Load-Adaptive Base-Station Management for Energy Reduction including Operation-Cost and Turn-On-Cost

Jiashang Liu*,Yang Yang†, Prasun Sinha‡ and Ness B. Shroff*‡
* Department of Electrical and Computer Engineering, The Ohio State University
† Qualcomm Cooperate Research and Development
‡ Department of Computer Science and Engineering, The Ohio State University

***Abstract*—The energy consumption of cellular networks has increased dramatically due to high demand for wireless communication. Base-stations (BSs) use about 60% to 80% of the energy consumed by these networks. An attractive way to reduce energy consumption is to turn the BSs off during periods of under-utilization. However, turning a BS back on typically consumes a lot of energy, which has not been considered in previous works, but critical to good energy management strategies. In this work, we dynamically determine the on-off schedule of these base-stations by taking both operation-cost and turn-on-cost into account. We develop the first online algorithm that only uses future information to decide the on-off status of each BS and characterize its performance using competitive ratio analysis. We extend it by utilizing history information which helps improve the competitive ratio. A heuristic adaptive online algorithm is then designed to balance the utilization of history and future information. We then show via simulation results that the adaptive algorithm works well under a wide range of traffic intensities.**

I. Introduction

The ever increasing traffic demand created by the proliferation of wireless devices and the development of new applications and services have made cellular networks a non-trivial source of energy consumption. Base-stations use about 60% to 80% of the energy consumed by cellular networks [1]. Base-stations are often deployed with a high density to handle peak load demands, which also means that during off-peak periods, the base-stations are highly underutilized [2]. Since the amount of energy consumed by these base-stations is large even when they are underutilized [3], an easy way to reduce energy consumption is to turn them off during off-peak period. Increasingly, base-stations with small setup times (time from off till they are operational) are becoming popular [4], which opens up the possibility of switching the base-station states more frequently and reduce the overall energy consumption.

Many prior works have focused on reducing the energy consumption of cellular networks. For example, [5] and [6] have proposed base-station on-off strategies by exploiting the energy-delay tradeoffs in cellular network. Another approach has been to equip base-stations with renewable energy. [7] and [8] have designed both a resource allocation scheme and an energy control scheme in green cellular networks with renewable energy. However, none of these works have considered the *turn-on-cost* of base-stations [9], which is a key factor in modeling the energy consumption. The key question here is: *how should the control plane decide on the on-off sequence of each base-station when both operation-cost and turn-on-cost are considered, while at the same time meeting the user traffic demand.*

In this work, we consider a multi-cell network, where a user may be in range of multiple base-stations. We focus on the problem of determining the status of these base-stations by taking both operation-cost and turn-on-cost into consideration. The objective is to reduce energy consumption of base-stations in the system, with the constraint that the traffic demands are met in each time-slot.

The contributions of this work are as follows:

- We develop the first online algorithm and evaluate its *competitive ratio*, a metric that captures the worst case performance, for any topology of the cellular network with any number of base-stations and users.
- We extend the first algorithm by adding a count-down scheme that utilizes history information, and surprisingly show that in a limited setting, history information can actually help improve the competitive ratio when limited future information is available.
- Inspired by the theoretical results, we develop a heuristic adaptive version of the first algorithm with count-down scheme, which strikes the right balance between the utilization of history and future information when making on-off decisions, and shows good performance via simulations under a wide range of loads.
- We show how the core component of all the three algorithms are reduced to solving a *minimum weighted set cover* problem, which can be solved by well-known approximation algorithms with low complexity.

Our paper is organized as follows. In Section II we discuss our system model and formulate our problem. In Section III we develop two online algorithms and present their competitive ratio analysis. Our heuristic adaptive online algorithm is discussed in Section IV. Detailed simulation results are shown in Section V. Finally we conclude our paper in Section VI.

II. System Model

A. Network Model

We consider a cellular network system with J base-stations and I users. The base-station to user connectivity is captured by $\mathbf{U} \triangleq \{U_j\}_{1\leq j\leq J}$, with U_j denoting the subset of users that base-station j can associate with.

Let us assume a time-slotted system, where the base-station-user association can evolve on a per time-slot basis. Let matrix

$\mathbf{A}(t) = [A_{ij}(t)]_{1\le i\le I, 1\le j\le J}$ denote the association decision in time-slot t, with $A_{ij}(t) = 1$ if user i is associated with base-station j and $A_{ij}(t) = 0$ otherwise. For each user i, it needs to be associated with a base-station in a given time-slot if there is a traffic request for user i in that time-slot. We use a binary variable $W_i(t) \in \{0, 1\}$ to indicate the traffic status of user i in time-slot t, with $W_i(t) = 1$ if user i needs to be served in time-slot t. For the rest of the paper, we refer to $\{W_i(t)\}_{t\in\mathbb{N}, 1\le i\le I}$ as the traffic arrival process of the system. Based on the above discussion, an associate decision $\mathbf{A}(t)$ is valid if it satisfies the following conditions:

$$\text{For any } j \text{ and any } i \notin U_j, A_{ij}(t) = 0. \quad \text{(1a)}$$

$$\text{For any } i,\ \sum_{j=1}^{J} A_{ij}(t) \in \{0,1\},\ \text{and}$$
$$(1 - \sum_{j=1}^{J} A_{ij}(t))W_i(t) = 0. \quad \text{(1b)}$$

The last equation indicates that a user needs to be associated with some base-station in a certain time-slot if there is traffic arrival for that user in that time-slot.

B. Energy Model

Each base-station can be turned off in some time-slots to reduce energy consumption if there is no user for it to associate with. Let $O_j(t) \in \{1, 0\}$ denote the on-off status of base-station j in time-slot t, with $O_j(t) = 1$ indicating that base-station j is in the on-state in time-slot t. Then the following conditions must be satisfied:

$$\text{For any } j,\ O_j(t) \in \{0,1\},\ \text{and}$$
$$(1 - O_j(t)) \sum_{i=1}^{I} A_{ij}(t) = 0. \quad (2)$$

The last equation indicates that a base-station needs to be turned on if it is associated with some users. Let $\mathbf{O}(t_1, t_2) \triangleq [O_j(t)]_{\forall j, t_1\le t\le t_2}$, and $\mathbf{O}(t) \triangleq \mathbf{O}(t, t)$. For any base-station, if it is in an on-state in time-slot t, then it incurs one unit of energy consumption, also referred to as the *operation-cost*. To better capture the energy consumption in the system, we also consider the *turn-on-cost* of the base-station, i.e., the cost incurred when a base-station is switched from the off-state to the on-state. Let the *turn-on-cost* for each base-station be K unit(s) of energy. Since the energy required for turning on a base-station is normally much larger than the energy required to keep the base-station in operating mode [3], we assume that $K > 1$. Further, for ease of analysis, we assume that K is an integer, although all the analysis can be easily generalized to the case when K is not an integer.

Given a base-station on-off decision $\mathbf{O}(1, T)$ and an initial condition $\mathbf{O}(0)$, the energy cost of the system from time-slot 1 to time-slot T, denoted as $c(\mathbf{O}(0), \mathbf{O}(1, T))$, can be expressed as

$$c(\mathbf{O}(0), \mathbf{O}(1,T)) = \underbrace{\sum_{t=1}^{T}\sum_{j=1}^{J} O_j(t)}_{\text{operation-cost}} + \underbrace{\sum_{t=1}^{T}\sum_{j=1}^{J} K(O_j(t) - O_j(t-1))^{+}}_{\text{turn-on-cost}},$$

where $x^+ \triangleq \max\{x, 0\}$. So, in time-slot t a *turn-on-cost* is incurred only when $O_j(t) = 1$ and $O_j(t - 1) = 0$.

C. Problem Formulation

Let us first define the *feasibility* conditions of on-off decisions based on the discussions in the previous two subsections.

Definition 1 (Feasibility of the base-station on-off decision)**.** A base-station on-off decision $\mathbf{O}(t_1, t_2)$ is said to be *feasible* if for any $t_1 \le t \le t_2$, there exists $\mathbf{A}(t)$ such that (1) and (2) are satisfied.

The objective of this work is to design algorithms that control the base-station on-off decision, with the aim of achieving the minimum energy cost, which we formally state below.

Problem$(1, T)$:

$$\underset{\mathbf{O}(1,T)}{\arg\min}\ c(\mathbf{O}(0), \mathbf{O}(1,T))$$
$$\text{s.t. } \mathbf{O}(1,T) \text{ is a feasible on-off decision.}$$

Note that the problem itself is parameterized by the time-horizon T, and the optimal solution can only be found when the traffic arrival pattern $\{W_i(t)\}_{t\in\mathbb{N}, 1\le i\le I}$ for the entire time-horizon, i.e., $1 \le t \le T$, is known a prior. However, in a practical system, base-stations may only know the traffic arrival pattern of the users up to a limited number of time-slots in the future, and thus in this work our assumption is that the system can only foresee the traffic pattern of all users up to M time-slots in the future. More precisely, at the start of any time-slot τ, the system only knows $\{W_i(t)\}_{1\le i\le I}$ for $t = \tau, \tau + 1, \tau + 2, \ldots, \tau + M - 1$, which, for the ease of exposition, is denoted as $\mathbf{W}(\tau, \tau + M - 1)$.

III. Design of Online Algorithms with Competitive Ratio Analysis

In this section, we develop the first online algorithm only utilizing future information and extend it to the second algorithm which also consider history information. We evaluate their performance in terms of their competitive ratios, which we formally define as the following.

Definition 2 (Competitive Ratio)**.** The competitive ratio of an online algorithm α is defined as the maximum ratio of its energy cost over that of the offline optimal solution across all possible arrival patterns, which is formally stated as

$$\max_{T, \mathbf{W}(1,T), \mathbf{O}(0)} \frac{c(\mathbf{O}(0), \mathbf{O}^{\alpha}(1,T))}{c(\mathbf{O}(0), \mathbf{O}^{*}(1,T))},$$

where $\mathbf{O}^{\alpha}(1, T)$ denotes the solution of algorithm α.

As can be seen from the definition above, the *competitive ratio* captures the worst-case performance of an online algorithm across all possible arrival patterns.

A. Sliding-Window-Algorithm

Our first online algorithm utilizes the traffic information from the future M time-slots. The algorithm works in a sliding window fashion. It focuses on a window of M consecutive time-slots at a time. For each window, it calculates the on-off decision of the first $L \leq M$ time-slots in the window, utilizing the arrival pattern of the entire M time-slots. After the on-off decision for the first L time-slots in the window are made, the algorithm advances the window by L time-slots and repeats the procedure. L is the step size of the algorithm. The detailed procedure of the algorithm on a window starting from time-slot τ to time-slot $\tau + M - 1$ is defined in Algorithm 1. We denote the output of the *Sliding-Window-Algorithm with step size* L as $\mathbf{O}^{s(L)}$.

Algorithm 1: Sliding-Window-Algorithm with step size L

Input: $\mathbf{W}(\tau, \tau + M - 1), \mathbf{O}(\tau - 1)$.
Output: $\mathbf{O}(\tau, \tau + L - 1)$.

```
Find the optimal solution to Problem(τ, τ + M − 1)
  given the initial condition O(τ − 1), which we denote as
  O*(τ, τ + M − 1);
for t = τ to τ + L − 1 do
  assign O(t) = O*(t);
```

The total computation overhead is directly dependent on L. We obtain the following result for $L = M$.

Theorem 1. The competitive ratio of the Sliding-Window-Algorithm with step size $L = M$ is no larger than $\max\,\{1 + K/M,\, 2\}$.

Proof: Based on the Definition 2, it suffices to show that for any $\mathbf{W}(1,T)$ and any feasible on-off decision under $\mathbf{W}(1,T)$, which we denote as $\mathbf{O}^f(1,T)$, the following equation holds.

$$\frac{c\left(\mathbf{O(0)}, \mathbf{O}^{s(M)}(1,T)\right)}{c\left(\mathbf{O(0)}, \mathbf{O}^f(1,T)\right)} \leq \max\left\{1 + \frac{K}{M}, 2\right\}.$$

We divide the proof in two steps. In the first step, we construct a feasible on-off decision $\mathbf{O}^m(1,T)$ by modifying $\mathbf{O}^f(1,T)$ and show that

$$\frac{c\left(\mathbf{O(0)}, \mathbf{O}^m(1,T)\right)}{c\left(\mathbf{O(0)}, \mathbf{O}^f(1,T)\right)} \leq \max\left\{1 + \frac{K}{M}, 2\right\}. \quad (3)$$

In the second step, we argue that

$$c\left(\mathbf{O(0)}, \mathbf{O}^{s(M)}(1,T)\right) \leq c\left(\mathbf{O(0)}, \mathbf{O}^m(1,T)\right),$$

which together with Equation (3), completes the proof.

Step 1: For any given $\mathbf{O}^f(1,T)$, we focus on the i^{th} consecutive on period of base-station j, and suppose this period lasts from s_i^j to e_i^j, that is, base-station j is turned on at the beginning of s_i^j and is kept on until the end of e_i^j. Let $p_i^j = s_i^j - e_i^j + 1$, then we have

$$c\left(O_j^f(s_i^j - 1), O_j^f(s_i^j, e_i^j)\right) = p_i + K.$$

We divide the whole time horizon into blocks where each block has length M (where the last block can have size less

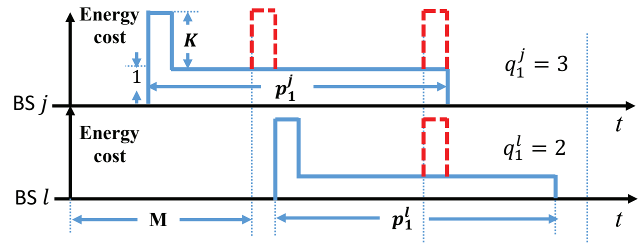


Fig. 1: On-off decision for base-station j and l, with blue solid-line indicating the energy cost of one consecutive on-period in the given feasible on-off decision $\mathbf{O}^f(1,T)$ and red dash-line indicating the additional turn-on-cost in $\mathbf{O}^m(1,T)$.

than M). Suppose from s_i^j to e_i^j these time-slots are spread across $q_i^j\,(q_i^j \geq 1)$ blocks. Then, we have,

$$\max\{M(q_i^j - 2) + 2, 1\} \leq p_i^j \leq M q_i^j.$$

We modify $O_j^f(s_i^j, e_i^j)$ by turning base-station j off right before the first time-slot of each block, and turning it on again. For example, Fig. 1 shows a feasible on-off decision for two base-stations. Before the first time-slot of the second and third blocks, we force base-station j off and turn it on again. By doing this, we can get a new feasible solution $O_j^m(s_i^j, e_i^j)$ based on the previous feasible solution by adding *turn-on-cost* times $(q_i - 1)$, and

$$\begin{aligned} c\left(O_j^m(s_i^j - 1), O_j^m(s_i^j, e_i^j)\right) &= p_i + K + (q_i - 1)K \\ &= p_i + q_i K. \end{aligned}$$

When $q_i \geq 2$,

$$\frac{c\left(O_j^m(s_i^j - 1), O_j^m(s_i^j, e_i^j)\right)}{c\left(O_j^f(s_i^j - 1), O_j^f(s_i^j, e_i^j)\right)} = \frac{p_i + q_i K}{p_i + K}$$

$$\leq 1 + \frac{(q_i - 1)K}{M(q_i - 2) + 2 + K} \leq \begin{cases} 1 + \frac{K}{M} & \text{if } M \leq K \\ 2 & \text{if } M > K \end{cases}$$

$$= \max\{1 + \frac{K}{M}, 2\}.$$

When $q_i = 1$, we do not modify $O_j^f(s_i^j, e_i^j)$, and thus

$$\frac{c\left(O_j^m(s_i^j - 1), O_j^m(s_i^j, e_i^j)\right)}{c\left(O_j^f(s_i^j - 1), O_j^f(s_i^j, e_i^j)\right)} = 1 \leq \max\left\{1 + \frac{K}{M}, 2\right\}.$$

Then for all base-stations across all the time-slots from 1 to T, by doing such modifications, the cost of modified on-off decision becomes,

$$c\left(\mathbf{O(0)}, \mathbf{O}^m(1,T)\right) = \sum_j \sum_i c\left(O_j^m(s_i^j - 1), O_j^m(s_i^j, e_i^j)\right)$$

$$\leq \sum_j \sum_i c\left(O_j^f(s_i^j - 1), O_j^f(s_i^j, e_i^j)\right) \times \max\left\{1 + \frac{K}{M}, 2\right\}$$

$$= c\left(\mathbf{O(0)}, \mathbf{O}^f(1,T)\right) \times \max\left\{1 + \frac{K}{M}, 2\right\}.$$

Step 2: According to Step 1, $\mathbf{O}^m(1,T)$ is constructed from $\mathbf{O}^f(1,T)$ by dividing the time horizon into blocks of size M, turning all the base stations off at the end of each block, and turning on the base-stations that are in the on-state at the beginning of each block as specified in $\mathbf{O}^f(1,T)$. Meanwhile, since *Sliding-Window-Algorithm with step size M* finds the local optimal solution for the whole block considering both base-stations on-off status at the beginning of each block and the user arrivals during the whole block, $c\left(\mathbf{O}(\mathbf{0}), \mathbf{O}^{s(M)}(1,T)\right)$ should not be greater than $c\left(\mathbf{O}(\mathbf{0}), \mathbf{O}^m(1,T)\right)$. Thus we have

$$c\left(\mathbf{O}(\mathbf{0}), \mathbf{O}^{s(M)}(1,T)\right) \leq c\left(\mathbf{O}(\mathbf{0}), \mathbf{O}^m(1,T)\right),$$

which completes the proof of Step 2. ■

B. Sliding-Window-Algorithm with Count-Down

Intuitively, one may think that in order to make an informed base-station on-off decision for the current time-slot, only the future information regarding the user traffic would be helpful. In other words, to make the on-off decision for the current time-slot, the decisions that have already been made in the previous time-slots seem quite irrelevant and are perhaps not useful. One may even tend to think that it is unlikely that there is a way to improve the competitive ratio of Algorithm 1, as it already tries to exploit all the future traffic information.

Surprisingly, neither conjecture is true. In this subsection, we will introduce a new algorithm, namely *Sliding-Window-Algorithm with Count-Down*, which utilizes history decision information and contains Algorithm 1 as a special case, and show that it can achieve a smaller competitive ratio than Algorithm 1.

The new algorithm basically modifies the base-station on-off decisions of Algorithm 1. In the new algorithm, each base-station keeps track of the number of time-slots since it was last decided to be in the on-state by Algorithm 1. If it is no more than a fixed threshold C, then the base-station will be kept on under the new algorithm for the current time-slot, otherwise it will be turned off.

To implement this new algorithm, we add a count-down timer for each base-station, which will be reset to C whenever it is decided to be in the on-state by Algorithm 1, and decremented by 1 for each time-slot that it is decided to be in the off-state by Algorithm 1, until it reaches zero. Under the new algorithm, a base-station is in the on-state whenever the count-down timer is non-zero. We let $H_j(t)$ denote the value of the count-down timer of base-station j in time-slot t, and refer to C as the count-down threshold. The detailed procedure of the algorithm on a window starting from time-slot τ to time-slot $\tau + M - 1$ is shown in Algorithm 2.

In Algorithm 2, the history information for each base-station j in time-slot t is captured in its timer $H_j(t)$. The larger the value of C, the more history information is utilized. In the extreme case when $C = 1$, it is easy to check that the algorithm reduces to Algorithm 1. In a simplified network where there is only one base-station, we are able to obtain the competitive ratio of the Sliding-Window-Algorithm with step size $L = 1$ and any count-down threshold C.

Algorithm 2: Sliding-Window-Algorithm with step size L and count-down threshold C

Input: $\mathbf{W}(\tau, \tau+M-1)$, $\mathbf{O}(\tau-1)$, $\mathbf{H}(\tau-1)$.
Output: $\mathbf{O}(\tau, \tau+L-1)$, $\mathbf{H}(\tau+L-1)$.

$\mathbf{O}(\tau, \tau+L-1) = \mathbf{O}^{s(L)}(\tau, \tau+L-1)$
for $t = \tau$ *to* $\tau + L - 1$ **do**
  /* Update count-down timer */
  for $j = 1$ *to* J **do**
    if $O_j(t) = 1$ **then**
      $H_j(t) = C$
    else if $O_j(t) = 0$ **then**
      $H_j(t) = \max\{H_j(t-1) - 1, 0\}$
  /* Modify the on-off decision */
  for $j = 1$ *to* J **do**
    if $H_j(t) > 0$ **then**
      $O_j(t) = 1$

Theorem 2. In the single-base-station network ($J = 1$), *Sliding-Window-Algorithm with step size $L = 1$ and fixed count-down threshold* **C** achieves the competitive ratio of

$$\begin{cases} 1 + (C-1)/(K+1) & \text{when } C \geq \max\{K - M + 1, 1\} \\ \max\{(C+K)/(C+M), 1\} & \text{otherwise} \end{cases}$$

Proof: See technical report [10]. ■

Corollary 2.1. In the single-base-station network ($J = 1$), *Sliding-Window-Algorithm with step size $L = 1$ and fixed count-down threshold* **C** achieves lowest possible competitive ratio when $C = \max\{K - M + 1, 1\}$.

The above corollary indicates that in the case when K is larger than M, Algorithm 2 can achieve a better worse-case guarantee when compared to Algorithm 1.

IV. Heuristic and Practical Algorithm Design

A. Insight from theoretical results

In Section III, we evaluated the performance of two online algorithms in terms of their competitive ratios, which considers their worst case performance across all possible arrival patterns. Theorem 2 shows that in the Sliding-Window-Algorithm with Count-Down, the count-down threshold $C = \max\{K - M + 1, 1\}$ achieves the lowest competitive ratio in a single-base-station system. However, it remains unknown whether $C = \max\{K - M + 1, 1\}$ yields good performance in the average case, especially when the traffic statistics information can be derived. Indeed, the optimal count-down threshold may vary as a function of the traffic intensity of the users in the system. To gain some insight, let us consider a single-user single base-station scenario. If the user has a very low traffic rate, it is more likely that the time interval between two arrivals is large, in which case it is better to reduce C so that the base-station can be shut off soon after serving the user, in order to avoid additional *operation-cost*. On the other hand, if the traffic intensity for that user is high, we should have a larger count-down threshold C such that the base-station is kept on for a longer time after it finishes serving the user, to help avoid any *turn-on-cost* that may be incurred when the user has another traffic request soon after its last one. In real systems, traffic statistics information can be derived from history. In the next subsection, we introduce a heuristic adaptive algorithm that exploits traffic statistics and through numerical simulation, it is shown that it can achieve better performance than Algorithms 1 and 2.

B. Sliding-Window-Algorithm with Adaptive Count-Down

Sliding-Window-Algorithm with Adaptive Count-Down adaptively chooses the count-down threshold for each base-station. Similar to Algorithm 2, it is designed by modifying the decision of Algorithm 1. For each base-station, it keeps track of the fraction of time-slots in which the base-station is to be kept on as decided by Algorithm 1, and determines the count-down threshold of each base-station based on this fraction. When the fraction is small, it indicates that user traffic served by this base-station is low, and the threshold should be set to a small value. For base-station j, let $C_j(t)$ denote the count-down threshold and $T_j^F(t)$ denote the number of time-slots in which it is decided to be in the on-state by Algorithm 1 in the previous F time-slots. In our proposed algorithm, we set $C_j(t)$ to be

$$C_j(t) = \begin{cases} (K-M+1)\rho_j^{\gamma(M)} & \text{when } M < K \\ 1 & \text{otherwise} \end{cases} \quad (4)$$

where $\rho_j = T_j^F(t)/F$ and $\gamma(M) = 1/(1-\frac{M}{K})$. The detailed procedure of the algorithm on a window starting from time-slot τ to time-slot $\tau + M - 1$ is shown in Algorithm 3.

Note that we use $\gamma(M)$ to force $C_j(t)$ to approach 1 when M is large. Below we provide some intuitive reasons for why this should help reduce energy cost.

There are two types of information that we can exploit in the system: future traffic information for the next M time-slots and the observed traffic statistics in the past. As M increases, meaning that more future information is readily available, we should lean towards the decision from Algorithm 1, which utilizes the future information. On the hand, when there is a lack of future traffic information, i.e., when M is small, we should rely more on the decision from Algorithm 2 with a count-down threshold that is optimized for the worst case.

C. Revisiting Sliding-Window-Algorithm

One problem remaining to be solved is how do we find the optimal solution of **Problem**$(1, M)$, which is needed in line 1 of Algorithm 1. In this subsection, we show that **Problem**$(1, M)$ can be reduced to the well-known minimum weighted set cover problem [11]. First, let us focus on the simple case when $M = 1$.

Lemma 1. $\mathbf{O}(1)$ is a feasible on-off decision *if and only if*

$$\{1 \le i \le I | W_i(1) = 1\} \subseteq \cup_{j:O_j(1)=1}\{U_j\}$$

Proof: See technical report [10] ∎

Algorithm 3: Sliding-Window-Algorithm with step size L and adaptive count-down threshold

Input: $\mathbf{W}(\tau, \tau+M-1), \mathbf{O}(\tau-1), \mathbf{H}(\tau-1)$.
Output: $\mathbf{O}(\tau, \tau+L-1), \mathbf{H}(\tau+L-1)$.

$\mathbf{O}(\tau, \tau+L-1) = \mathbf{O}^{s(L)}(\tau, \tau+L-1)$
for $t = \tau$ *to* $\tau+L-1$ **do**
  for $j = 1$ *to* J **do**
    Update $C_j(t)$ using Equation (4)
    if $O_j(t) = 1$ **then**
      $H_j(t) = C_j(t)$
    else if $O_j(t) = 0$ **then**
      $H_j(t) = \max\{H_j(t-1) - 1, 0\}$
  for $j = 1$ *to* J **do**
    if $H_j(t) > 0$ **then**
      $O_j(t) = 1$

Based on Lemma 1, we can rewrite **Problem(1,1)** as the following,

$$\arg\min_{\mathbf{O}(1)} \sum_{j=1}^{J} \left(O_j(1) + K(O_j(1) - O_j(0))^+\right)$$
$$\text{s.t. } \{1 \le i \le I | W_i(1) = 1\} \subseteq \cup_{j:O_j(1)=1} U_j.$$

It is not hard to see that the above problem falls in the format of the classic minimum weighted set cover problem. The set to be covered is the set of users that need to be served in time-slot 1. Each base-station j covers a subset U_j of users. The weight on each base-station is $(1 + K(1 - O_j(0))^+)$. The on-off decision $\mathbf{O}(1)$ describes the set of base-stations to be selected to cover $\{1 \le i \le I | W_i(1) = 1\}$.

Similarly, we can show that **Problem**$(1, M)$ with $M > 1$ can also be mapped into a minimum weighted set cover problem (see our technical report [10]).

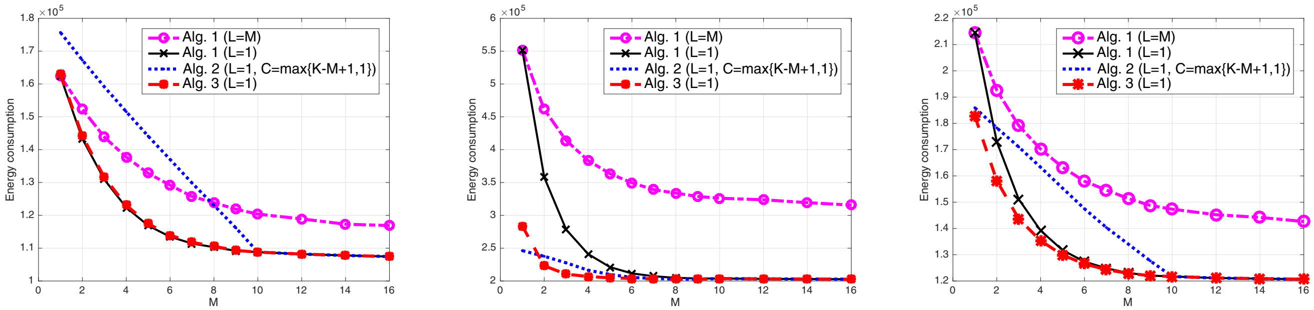
V. Numerical Result

A. Simulation Setup

We conduct simulations on a network topology with 10 base-stations and 35 users. Traffic requests are independent across different time-slots and across all users. In each time-slot, traffic arrival of user i is Bernoulli distributed with parameter p. We define three user traffic scenarios: in low traffic case $p = 0.01$, in heavy traffic case $p = 0.1$ and in mixed traffic case p is unbiased chosen between 0.01 and 0.1. For any base-station j and user i, whether $i \in U_j$ depends on their physical distance, with threshold $11.89m$. The turn-on-cost K is set to be 10 and the length of past period F to be 1000. We use greedy-approximation algorithm to solve the minimum weighted set cover problem [11].

B. Performance Evaluation and Analysis

We study the performance of four different algorithms: Algorithm 1 ($L = M$), Algorithm 1 ($L = 1$), Algorithm 2 ($L = 1$, $C = \max\{K - M + 1, 1\}$) and Algorithm 3 ($L = 1$).



(a) Low traffic case

(b) Heavy traffic case

(c) Mixed traffic case

Fig. 2: Energy consumption of the whole system as a function of the number of future information.

Notice that when $M \geq K$, the count-down thresholds for the last two algorithms are at most 1, and the last three algorithms tend to the same algorithm.

(1) **The impact of step size** L: Fig 2 shows that Algorithm 1 ($L = 1$) is much better than Algorithm 1 ($L = M$) in all three cases. The reason is that as L decreases, we make the decision more frequently and utilize more future information. However, a small L leads to higher computational complexity.

Although in the simulation result Algorithm 1 ($L = 1$) performs better than Algorithm 1 ($L = M$), it is not true for all arrival patterns. A counterexample can be found in our technical report [10].

(2) **The impact of *count-down scheme***: Fig 2a and Fig 2b show that with $L = 1$, when the traffic is low, Algorithm 2 performs worse than Algorithm 1. This is because when the traffic is low, it is more likely that the time interval between two arrivals is large, and operation-cost can be avoided if the base-station can be shut off soon after serving the user. However, when the traffic is heavy, the count-down scheme of Algorithm 2 actually helps avoid *turn-on-cost* that may be incurred when a user has another traffic request soon after its last one, especially when M is small.

(3) **The impact of the value** M: When M is small, as M increases, the performance of Algorithm 1 ($L = 1$) and Algorithm 1 ($L = M$) improves dramatically. The reason is that with small M, it is likely that Algorithm 1 turns off some base-stations which will probably be used in the next few time-slots. It avoids *operation-cost* but incurs expensive *turn-on-cost*. As M increases, more future information becomes available, and additional *turn-on-cost* due to the decisions made when M is small can be reduced. When M is large, such additional *turn-on-cost* is not much and increasing M only brings small gains.

VI. Conclusion

In this work, we focus on the on-off control scheme design of base-stations in multi-cell networks, with the objective of reducing energy consumption of base-stations by both considering the *operation-cost* as well as *turn-on-cost* in the energy model. We propose two online algorithms and study their competitive ratios. An adaptive online heuristic based on them has also been proposed to deal with performance in the average case. Simulation results show that our adaptive algorithm is capable of lowering the energy consumption for a wide range of traffic intensities.

Acknowledgments: This work was partially supported by NSF grant CNS-1409336.

References

[1] Marsan, M Ajmone and Chiaraviglio, Luca and Ciullo, Delia and Meo, Michela, "Optimal Energy Savings in Cellular Access Networks," in *Proc. of IEEE ICC Workshops*, 2009, pp. 1–5.

[2] 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, 2011.

[3] C. Peng, S.-B. Lee, S. Lu, H. Luo, and H. Li, "Traffic-driven Power Saving in Operational 3G Cellular Networks," in *Proc. of ACM Mobicom*, 2011, pp. 121–132.

[4] 3GPP, "LTE; Evolved Universal Terrestrial Radio Access (E-UTRA); Base Station (BS) Conformance Testing," 2011.

[5] J. Wu, S. Zhou, and Z. Niu, "Traffic-aware Base Station Sleeping Control and Power Matching for Energy-Delay Tradeoffs in Green Cellular Networks," *IEEE Transactions on Wireless Communications*, vol. 12, no. 8, pp. 4196–4209, 2013.

[6] E. Oh, K. Son, and B. Krishnamachari, "Dynamic Base Station Switching-On/Off Strategies for Green Cellular Networks," *IEEE Transactions on Wireless Communications*, vol. 12, no. 5, pp. 2126–2136, 2013.

[7] Y. Yang, J. Liu, P. Sinha, and N. B. Shroff, "Dynamic User Association and Energy Control in Cellular Networks with Renewable Resources," in *Proc. of IEEE CDC*, 2015, pp. 1643–1650.

[8] J. Gong, J. S. Thompson, S. Zhou, and Z. Niu, "Base Station Sleeping and Resource Allocation in Renewable Energy Powered Cellular Networks," *IEEE Transactions on Communications*, vol. 62, no. 11, pp. 3801–3813, 2014.

[9] O. Arnold, F. Richter, G. Fettweis, and O. Blume, "Power Consumption Modeling of Different Base Station Types in Heterogeneous Cellular Networks," *IEEE Future Network & Mobile Summit*, pp. 1–8, 2010.

[10] J. Liu, Y. Yang, P. Sinha, and N. B. Shroff. "Load-Adaptive Base-Station Management for Energy Reduction including Operation-Cost and Turn-On-Cost." Technical Report. [Online]. Available: https://www.dropbox.com/s/d26b5a3kqdsmmxp/technicalreport.pdf?dl=0

[11] V. Chvatal, "A Greedy Heuristic for the Set-Covering Problem," *Mathematics of operations research*, vol. 4, no. 3, pp. 233–235, 1979.