Traffic Off-Loading With Energy-Harvesting Small Cells and Coded Content Caching

Tao Li, Mehdi Ashraphijuo, Xiaodong Wang, *Fellow, IEEE*, and Pingyi Fan, *Senior Member, IEEE*

***Abstract*— We consider content delivery to users in a system consisting of a macro base station (BS), several energy-harvesting small cells (SCs), and many users. Each SC has a large cache and stores a copy of all contents in the BS. A user's content request can be either handled by the SC for free if it has enough energy, or by the BS that has a cost. Each user has a finite cache and can store some most popular contents. We propose an efficient coded content caching schemes and an optimal transmission schemes for this system to maximally off-load the data traffic from the macro BS to the energy-harvesting SCs, and therefore minimize the power consumption from the grid. Specifically, the proposed coded caching scheme stores fractions of some most popular contents, such that contents requested from multiple users can be simultaneously delivered by the BS or SC. Moreover, the optimal transmission policy is formulated and solved as a Markov decision process. Extensive simulation results are provided to demonstrate that the proposed coded caching and transmission schemes can provide significantly higher traffic off-loading capability compared with systems with no caching or with uncoded caching, as well as systems that employ heuristic-based transmission schemes.**

***Index Terms*— Traffic off-load, energy harvesting, macro base station (BS), small cells (SCs), proactive coded content caching, Markov decision process (MDP), value iteration.**

I. Introduction

THE demands for wireless multimedia data services grow dramatically in recent years due to the rapid development of mobile internet and smart phones [1]. In the current cellular architecture, these contents are typically delivered to users by unicast, which may lead to severe network congestion in the near future due to the explosive growth of mobile data. On the other hand, it has already been observed that only a small portion of the most popular multimedia contents account for a significant amount of the traffic load and consume a large portion of the energy from the power grid [2]. Thus, it is essential to design appropriate strategies to deliver these popular contents to alleviate the traffic load and to reduce the energy consumption.

For popular contents that are requested frequently by many users, broadcast (or multicast) is a more efficient way to deliver them to different users simultaneously. A rich body of prior works investigated the hybrid scheduling schemes of multicast and unicast in cellular networks, see, e.g., [3]–[5]. Moreover, AT&T announced that it will use a 700 MHz channel for LTE broadcast networks to remove video from its wireless cellular networks, clearing those airwaves up for other data services [6], [7]. However, different users may request a particular content at different times, which means that the base station (BS) needs to delay the responses to earlier requests in order to employ broadcast, leading to the quality-of-service (QoS) degradation.

Proactive content caching is another approach to exploit the non-uniform popularity of contents and to off-load the unicast traffic [8]–[10]. More specifically, when a new popular content appears, the BS will broadcast it to all users under the coverage and store it in each user's cache. Later if a user requests this particular content, then it can directly retrieve the content from its cache, so that the unicast traffic on the link between the BS and this user can be offloaded and the associated energy consumption from the power grid can be saved. However, in practice, each user terminal can only proactvely cache a small quantity of contents due to size limit [11].

This paper considers a system where each user has a finite-size cache to proactively store some portion of the popular contents. Furthermore, energy-harvesting-powered small cells (SCs) [12] are employed to cache popular contents and deliver them to users [13], [14]. This paper only focuses on the delivery problem of popular contents that may be requested by different users frequently, not all contents available on the Internet. Thus, it is reasonable to assume that SC does not have cache size constraint due to its larger form factor. The basic idea is to utilize the free harvested energy and the caches of the SCs and users to reduce the content delivery traffic from the macro BS to users, and to save the energy consumption from the power grid. Some prior works have investigated the delivery strategies at SCs in the content delivery system [15]–[19]. For instance, the optimal distributed caching strategy has been proposed in [15] and [16] when there are multiple available SCs for every user. Reference [17] discussed the caching problem in a multi-tier network, in which an information-centric based edge caching scheme was introduced. Some specific energy man-

Manuscript received June 7, 2016; revised October 1, 2016 and November 19, 2016; accepted November 27, 2016. Date of publication December 7, 2016; date of current version February 14, 2017. This work was supported in part by the National Natural Science Foundation of China under grant 61329101, the Leading Talents Program of Guangdong Province under grant 00201510, the Basic Research Program of Shenzhen under grant JCYJ20151117161854942. the China Major State Basic Research Development Program (973 Program) under Grant 2012CB316100(2), the European Union under Project CoNHealth 294923. The associate editor coordinating the review of this paper and approving it for publication was Z. Zhang.

T. Li and P. Fan are with Tsinghua National Laboratory for Information Science and Technology, Department of Electronic Engineering, Tsinghua University, Beijing 100084, China (e-mail: litao12@mails.tsinghua.edu.cn; fpy@tsinghua.edu.cn).

M. Ashraphijuo and X. Wang are with the Electrical Engineering Department, Columbia University, New York City, NY 10027 USA (e-mail: mehdi@ee.columbia.edu; wangx@ee.columbia.edu).

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

Digital Object Identifier 10.1109/TCOMM.2016.2636283

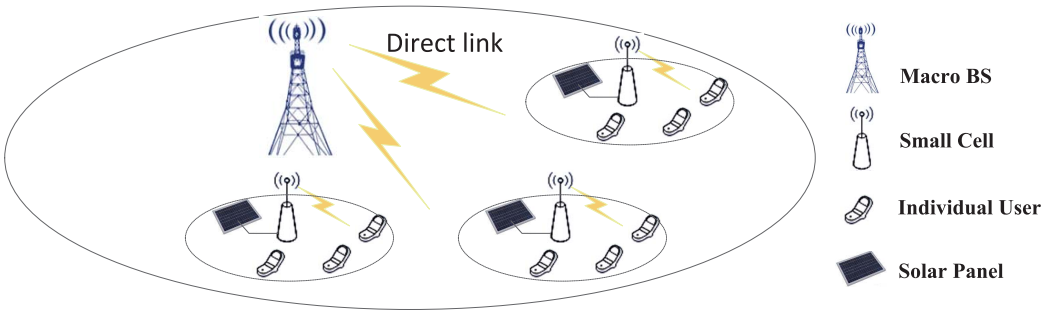

Fig. 1. The system model that consists of a macro BS, multiple SCs and many users. The SCs are energy-harvesting-based while the macro BS is grid powered.

agement strategies for energy-harvesting SCs were discussed in [18] and [19].

In this paper, we consider content delivery to users in a system with energy-harvesting-based SCs and finite-size cache at the user end. Note that the problem studied here is different from some prior works, such as [18]–[19]. In particular, in [18] each user has an infinite cache; while in [19] only the SC can cache information. First, in order to make efficient use of the finite-size cache at the user end, we develop a proactive coded content caching scheme that can significantly reduce the traffic load from SC or macro BS to users compared with the conventional uncoded caching scheme. We then formulate the content delivery procedure as a Markov decision process (MDP) and obtain the optimal policy via value iteration. In addition, the effects of repetitive requests and the basic energy consumption at SCs are also discussed. Extensive simulation results are provided to demonstrate the traffic off-load performance of the proposed system, and to compare with conventional systems with no caching, or uncoded caching, as well as with heuristic-based transmission schemes.

The remainder of this paper is organized as follows. The system under consideration is described and the concept of coded content caching is briefly reviewed in Section II. The proposed proactive coded content caching scheme is presented in Section III. Then, the optimal content delivery problem is formulated and solved in Section IV. The effects of repetitive requests and a holistic energy consumption model are discussed in Section V. Simulation results are given in Sections VI and finally Section VII concludes the paper.

II. System Descriptions and Background

A. System Model

Fig. 1 depicts the general structure of the system under consideration, which consists of a macro BS, multiple SCs and many users. The SCs are energy-harvesting-based, while the macro BS is grid powered. The users can potentially receive data from both the macro BS and the SCs that they are associated with. For a specific SC under the coverage of the macro BS, it is assumed that there are K individual active users associated with it. In addition, it is assumed that each user is associated with exactly one SC, so that we can focus on a single SC and the corresponding K users.

We denote the content set at the macro BS by $\mathcal{C} = \{c_1, c_2, \ldots, c_{N_0}\}$, where c_n is the n-th ranked content and is of unit size. The popularity of c_n refers to the probability that a user's request corresponds to this content, which follows the ZipF distribution [20], given by

$$f_n = \Pr\{c_n \text{ is requested by a particular user}\} = \frac{(1/n)^{\nu}}{\sum_{j=1}^{N_0}(1/j)^{\nu}}, \quad n = 1, 2, \ldots, N_0, \tag{1}$$

where $\nu > 0$ is the ZipF parameter, and larger ν means that fewer contents account for the most popular ones.

The SC is powered solely by the harvested energy. During the i-th time slot, the SC harvests a random amount of energy, denoted by e_i, and then stores it in a battery with finite capacity $B_{\max}$, where $e_i = E_l$ with probability p_l, $l = 1, 2, \ldots, L$ [21], [22]. Here, it is assumed that the harvested energy e_i cannot be immediately used in the current time slot. Denote b_i as the amount of energy stored in the battery and u_i as the energy consumption at the SC during the i-th time slot, (where $u_i \leq b_i$). Then, the available energy in the next time slot is

$$b_{i+1} = \min\{B_{\max}, b_i - u_i + e_i\}, \quad u_i \leq b_i. \tag{2}$$

B. Data Transmission Scheme

Similar to the system structure in [8], the macro BS proactively pushes some popular contents in set $\mathcal{C}$ to the SCs and the users through a broadcast channel, which can be accomplished energy-efficiently by low-rate transmission since there is no rigid delay requirement for this task. For tractability, most prior works on proactive caching in wireless networks assume that the popular contents can always be placed in the cache and hence effectively there is no limit on the cache size [8], [12]. This assumption is reasonable since at any time the hot contents that are cached are a small subset of all available contents on the Internet [23]–[25]. Thus, we assume that the SC does not have cache size constraint and can maintain a duplicated version of the popular content set $\mathcal{C}$ in its storage. On the other hand, every user has a finite buffer of size M and can only cache a subset of $\mathcal{C}$ for future usage. How to select appropriate contents for each user to cache is an important research problem.

In this paper, we employ a coded content caching method to improve the efficiency of finite-size content caching. Specifically, denote $\mathcal{C}_e$ as a subset of $\mathcal{C}$ (where $|\mathcal{C}_e| = N$ and $M \leq N \leq N_0$), which contains the N most popular contents

among the total N_0 contents in $\mathcal{C}$. Instead of caching the M most popular contents as in the conventional caching scheme, each user partially caches the N most popular contents, i.e., $\frac{M}{N}$ portion of each content in $\mathcal{C}_e$ is stored in each user. At the beginning of each time slot, each user requests a particular content from the content set $\mathcal{C}$ of the macro BS. Then, we have the following three possible cases.

- *Case 1*: The requested content is in $\mathcal{C}_e$, and the SC has enough energy. The SC will send a coded signal to the user. With this signal and the partially cached contents, the user can recover the requested content.
- *Case 2*: The requested content is not in $\mathcal{C}_e$, and the SC has enough energy. The SC will then send the entire requested content to the user.
- *Case 3*: The SC does not have enough energy for transmission. Then, the macro BS will transmit the requested content to the user.

Hence, by appropriately designing the caching and transmission protocols, a large portion of the traffic on the direct links between the BS and users can be offloaded, and a significant amount of energy can be saved at the BS side.

In this paper, our assumption is that the SC always broadcasts contents to users in the cell, for both cached contents and uncached contents. Hence the SC does not adjust its power for individual users, but has a broadcast power to cover all users in the cell. This way we can make full use of coded multicast transmission for cached contents. A more sophisticated SC would adjust its transmission based on the requested contents as well as the channel conditions, e.g., to employ multicast beamforming for cached contents and unicast beamforming for uncached contents. But this is beyond the scope of this paper and is left as a future work. Besides, similar to many prior works (such as [26]), we assume that the duration of the scheduling slot is long enough to average out the small-scale channel fading process, and hence the ergodic capacity can be achieved using channel coding. Thus, we can determine the required energy E_0 to deliver one unit content.

C. Background on Coded Content Caching

We next briefly summarize the coded content caching scheme introduced in [11]. It is assumed that all contents in $\mathcal{C}$ are requested by users with equal probability (namely $\nu = 0$). Then, all contents in $\mathcal{C}$ should participate in caching, i.e., $\mathcal{C}_e = \mathcal{C}$ (namely $N = N_0$). Denote d_k as the requested content by the k-th user and $\mathcal{D} = \{d_1, d_2, \ldots, d_K\}$. On the assumption that all elements in $\mathcal{D}$ are distinct, the procedures for the proactive coded caching phase and the corresponding content delivery phase have been given in [11], which are shown in Algorithm 1 and Algorithm 2, respectively.

As shown in Algorithm 1, each content in $\mathcal{C}_e$ is split into $\binom{K}{T}$ (where $T = KM/N$) non-overlapping fragments with equal size of $1/\binom{K}{T}$. Then, in Step 10 of Algorithm 1, each user selects $\frac{T}{K}\binom{K}{T}$ among the $\binom{K}{T}$ fragments of each content to cache during the proactive caching phase. The size of the total cached contents of each user is $\frac{T}{K}\binom{K}{T} \cdot 1/\binom{K}{T} \cdot N = M$, which meets the storage size constraint at the user end. Besides, it can be seen that each fragment has T copies stored in T different users' caches after the content caching phase.

Algorithm 1 Partially Caching the Contents $\{c_1, c_2, \ldots, c_N\}$ into the Storages of the Users

1: $T \leftarrow MK/N$;
2: $\Phi \leftarrow \{\phi | \phi \subset \{1, 2, \ldots, K\}, |\phi| = T\}$;
3: **for** $n \in \{1, 2, \ldots, N\}$ **do**
4: split the content c_n into $|\Phi|$ fragments with equal size, which are indexed by $\{c_{n,\phi} | \phi \in \Phi\}$, where $\cap c_{n,\phi} = \varnothing$ and $\cup c_{n,\phi} = c_n$;
5: **end for**
6: **for** $k \in \{1, 2, \ldots, K\}$ **do**
7: **for** $n \in \{1, 2, \ldots, N\}$ **do**
8: **for** $\phi \in \Phi$ **do**
9: **if** $k \in \phi$ **then**
10: cache the content fragment $c_{n,\phi}$ into the storage of the k-th user;
11: **end if**
12: **end for**
13: **end for**
14: **end for**

Algorithm 2 Delivering the Requested Contents $\{d_1, d_2, \ldots, d_K\}$ to the Users

1: $T \leftarrow MK/N$;
2: $\Psi \leftarrow \{\psi | \psi \subset \{1, 2, \ldots, K\}, |\psi| = T + 1\}$;
3: $\{l_1, l_2, \ldots, l_K\} \leftarrow \{l_k | d_k = c_{l_k}\}$;
4: **for** $\psi \in \Psi$ **do**
5: transmit the coded data flow $(\oplus_{k \in \psi} c_{l_k, \psi \backslash k})$, where $\oplus$ denotes the xor operation;
6: **end for**

In the content delivery phase, the following lemma from [11] can be used to calculate the required traffic load under the coded caching.

Lemma 1: Provided that each of the K users independently asks for one of the N contents in $\mathcal{C}$ at the beginning of each time slot, the minimum required traffic load for these K distinct requested contents is

$$R = K(1 - M/N) \cdot \min\{\frac{1}{1 + KM/N}, N/K\}. \tag{3}$$

Essentially, the coded caching scheme enables coded multicast to $T + 1$ users that request different contents during the delivery phase. Each content fragment is of size $1/\binom{K}{T}$ and the number of multicasts during a time slot is $\binom{K}{T+1}$. So the required traffic load is $\binom{K}{T+1} \cdot 1/\binom{K}{T} = K(1 - M/N)\frac{1}{1+KM/N}$. Besides, N/K factor is multicast benefit if $N < K$.

III. Proposed Coded Content Caching

In practical systems, where the popularity of the contents is far away from uniform but follows the ZipF distribution, it has been shown in [27] that involving all contents in the coded caching is not an efficient way to reduce the traffic load. Reference [27] proposed a multi-group coded caching scheme and analyzed the expected data rate requirement for content

delivery for each cycle. To make use of the non-uniform content distribution, this section proposes a popularity-based coded content caching scheme, in which content set $\mathcal{C}$ is divided into two subsets based on their popularity: coded one and uncoded one. For the coded subset, coded transmission similar to that is described in Section II.C is employed, while unicast transmission is used for the uncoded subset. Note that here we propose a new coded caching scheme that has the following advantages over the scheme in [27]. Firstly, there is no explicit traffic load expression for multi-group coded caching, whereas the traffic load of our scheme is given in Lemma 2. Secondly, in multi-group coded caching, the contents are divided into multiple groups, and different groups employ different coding schemes with different cache sizes, which are determined by multiple exhaustive search. Our proposed scheme is much simpler. Thirdly, our proposed scheme offers a higher traffic off-loading performance than that in [27] under the ZipF content distribution, as will be seen in Section VI.A.

More specifically, only the most popular subset of the content set $\mathcal{C}$, denoted as $\mathcal{C}_e = \{c_1, c_2, \ldots, c_N\}$ (where $N \leq N_0$), should be involved in the coded caching process at the user end. The proactive coded caching process is the same as Algorithm 1. That is, every content in $\mathcal{C}_e$ is split into $\binom{K}{T}$ non-overlapping fragments with equal size. Then, each user selects $\frac{T}{K}\binom{K}{T}$ fragments of each content to cache according to Step 10 of Algorithm 1. On the other hand, the content delivery phase becomes different. A requested content $d_k \in \mathcal{C}$ from the k-th user is either coded ($d_k \in \mathcal{C}_e$) or uncoded ($d_k \in \mathcal{C}\backslash\mathcal{C}_e$). Since $\mathcal{C}_e$ contains the N most popular contents in $\mathcal{C}$ whose probability distribution is given in (1), the probability that d_k is coded is

$$p_0 \triangleq \Pr\{d_k \in \mathcal{C}_e\} = \sum_{n=1}^{N} f_n = \frac{\sum_{n=1}^{N}(1/n)^v}{\sum_{j=1}^{N_0}(1/j)^v}. \tag{4}$$

Suppose that the number of coded elements in $\mathcal{D} = \{d_1, d_2, \ldots, d_K\}$ is k_i, which is binomial distributed, i.e.,

$$\Pr\{k_i = k\} = \binom{K}{k} \cdot p_0^k (1-p_0)^{K-k}, \quad k \in \{0, 1, 2, \ldots, K\}. \tag{5}$$

Denote the set of k requested coded contents as $\mathcal{D}' = \{d_{n_1}, d_{n_2}, \ldots, d_{n_k}\}$ and the $K-k$ requested uncoded contents as $\mathcal{D}'' = \mathcal{D}\backslash\mathcal{D}'$. The content delivery process is given in Algorithm 3. It is seen that the coded contents are simultaneously transmitted to the corresponding users as in Algorithm 2, whereas the each uncoded content is individually transmitted. The following lemma gives the traffic loads for transmitting the k coded and the $K-k$ uncoded contents.

Lemma 2: The required traffic loads for delivering the k coded contents and the $K-k$ uncoded contents are given respectively by

$$r_1(k) = \min\left\{\frac{\binom{K}{T+1} - \binom{K-k}{T+1}}{\binom{K}{T}}, (k - kM/N)\cdot\frac{N}{k}\right\}, \tag{6}$$

$$r_2(K-k) = K-k, \quad k \in \{0, 1, 2, \ldots, K\}, \tag{7}$$

where $T = \frac{KM}{N}$ and $\binom{K-k}{T+1} = 0$ if $K - k < T + 1$.

Algorithm 3 Delivering the Requested Contents $\{d_1, d_2, \ldots, d_K\}$ to the Users

1: Split the set $\mathcal{D} = \{d_1, d_2, \ldots, d_K\}$ into two subsets: $\mathcal{D}' = \{d_{n_1}, d_{n_2}, \ldots, d_{n_k}\}$ and $\mathcal{D}'' = \{d_{m_1}, d_{m_2}, \ldots, d_{m_{K-k}}\}$, where $\mathcal{D}' \subset \mathcal{C}_e$ and $\mathcal{D}'' \subset (\mathcal{C}\backslash\mathcal{C}_e)$;
2: $T \leftarrow MK/N$;
3: $\Psi \leftarrow \{\psi | \psi \subset \{1, 2, \ldots, K\}, |\psi| = T + 1\}$;
4: $\{l_1, l_2, \ldots, l_k\} \leftarrow \{l_j | d_{n_j} = c_{l_j}\}$;
5: **for** $\psi \in \Psi$ **do**
6: transmit the coded data flow $(\oplus_{n_j \in \psi} c_{l_j, \psi\backslash n_j})$, where $\oplus$ denotes the xor operation;
7: **end for**
8: $\{l'_1, l'_2, \ldots, l'_{K-k}\} \leftarrow \{l'_j | d_{m_j} = c_{l'_j}\}$;
9: **for** each $d_{m_j} \in \mathcal{D}''$ **do**
10: transmit the entire requested content $c_{l'_j}$ to the corresponding m_j-th user;
11: **end for**

Proof: First, consider the special case of $k = K$. Since M/N portion of each content in $\mathcal{C}_e$ has already been cached in each user in advance, according to Step 5 of Algorithm 2, the transmitted signal is the sum of $T+1$ information flows that are requested by $T+1$ different users. Each user can decode its requested content from the sum signal with the help of the cached contents. For instance, to recover the content d_i (namely c_{l_i}) for the i-th user, the received signal at the user is $\oplus_{k\in\psi} c_{l_k, \psi\backslash k}$ ($i \in \psi, \psi \in \Psi$). Note that $c_{l_i, \psi\backslash i}$ is the content of interest and $c_{l_k, \psi\backslash k} (k \neq i)$ are the contents requested by other users which have already been cached in the i-th user according to Step 10 of Algorithm 1. As a result, the i-th user can extract $c_{l_i, \psi\backslash i}$ from the sum signal $\oplus_{k\in\psi} c_{l_k, \psi\backslash k}$. Hence, $T+1$ simultaneous transmissions to $T+1$ users are achieved. Each time, the system selects $T+1$ out of the K users to implement multicast transmission, so that the total number of multicast operations is $\binom{K}{T+1}$. Each content fragment is of size $1/\binom{K}{T}$, so the required traffic load is $\binom{K}{T+1} \cdot 1/\binom{K}{T} = K(1 - M/N)\frac{1}{1+KM/N}$. On the other hand, if $N < K$ that the total number of popular contents is less than the number of requests, the system then enjoys an improvement of N/K from broadcast. The corresponding traffic load is $K(1 - M/N) * (N/K)$. Thus, the final traffic expression in this condition is $K(1 - M/N) \cdot \min\{\frac{1}{1+KM/N}, N/K\}$.

Next, consider the general case that $k \leq K$ users request the contents in the coded set $\mathcal{C}_e$. In this case, we randomly select $K-k$ additional coded contents from $\mathcal{C}_e$ in addition to the k requested coded contents. In order to deliver these k real requests and $K-k$ virtual requests, $\binom{K}{T+1}$ multicast operations are enough based on discussion above. However, the multicast operations among those $K-k$ virtual requests are actually not needed. The decreased number of multicast operations is $\binom{K-k}{T+1}$. Hence, the total number of multicast operations becomes $\binom{K}{T+1} - \binom{K-k}{T+1}$. Since the size of each multicast operation is $1/\binom{K}{T}$, the required traffic load is $r_1(k) = \min\left\{\frac{\binom{K}{T+1} - \binom{K-k}{T+1}}{\binom{K}{T}}, (k - kM/N)\cdot\frac{N}{k}\right\}$. On the other hand, the $K-k$ uncoded contents in $\mathcal{D}''$ that are not in the

cache need to be transmitted to the users directly. Thus, the required transmission load is $r_2(k) = K - k$. ■

IV. Transmission Strategy Design

In general, the delivery of the requested contents from users can be handled by either the associated SC or the macro BS. Since the macro BS is powered by grid while the SC by the free harvested energy, we next investigate the transmission scheme that minimizes the transmissions handled by the macro BS. In particular, we formulate an MDP problem and then solve it by the value iteration algorithm [26], [28], [29].

A. Problem Formulation

A standard MDP consists of the following elements: system state, action set, cost function and state transition matrix, which will be described as follows.

1) System State: The system state at the i-th time slot can be expressed as $s_i = (\tilde{b}_i, k_i)$, where $k_i \in \{0, 1, 2, \ldots, K\}$ is the number of coded requests and $\tilde{b}_i$ is the discretized battery level.

2) Action Set: On the condition that $k_i = k$, the minimum required traffic load is $r_1(k)$ coded content units and $r_2(K-k)$ uncoded content units given in (6)–(7). The SC can decide to sleep, send coded contents, send uncoded contents or send both of them based on the available energy level $\tilde{b}_i$ and the traffic load requirement of the users $(r_1(k), r_2(K-k))$. We denote the feasible action as $\pi_i = (\pi_{i,1}, \pi_{i,2})$. Specifically, if $\pi_{i,1} = 1$, it indicates that the SC will send $r_1(k)$ coded content units to the users. Otherwise, $\pi_{i,1} = 0$ and SC does not send any coded contents. On the other hand, $\pi_{i,2}$ indicates that the SC will send $\pi_{i,2}$ uncoded content units, (where $\pi_{i,2} \in \{0, 1, 2, \ldots, r_2(K-k)\}$). Besides, any feasible action needs to satisfy the following energy constraint

$$u_i = E_0 \cdot (r_1(k) \cdot \pi_{i,1} + \pi_{i,2}) \le \tilde{b}_i, \tag{8}$$

where u_i is the energy consumption under SC's action π_i, and E_0 is the energy consumption for delivering one unit content.

3) Cost Function: Similar to [12], the cost function for each time slot corresponds to the transmissions handled by the macro BS, which depends on both the system state and the adopted action. Depending on whether or not the macro BS knows the coded caching information at the user end, we consider two different types of system models, which are named as distributed coded caching system and centralized coded caching system, respectively.

In the distributed coded caching system, the macro BS does not know the cache content at the user end. So it needs to transmit the entire $r_1(k)$ contents for the corresponding users on the condition $\pi_{i,1} = 0$. The cost is then

$$g_i(s_i, \pi_i) = k_i \cdot (1 - \pi_{i,1}) + (r_2(K-k) - \pi_{i,2}). \tag{9}$$

In the centralized coded caching system, the macro BS knows the contents in the caches of users, so that it can directly transmit coded content data for these k_i requested coded contents when $\pi_{i,1} = 0$. In this case, the cost becomes

$$g_i(s_i, \pi_i) = r_1(k) \cdot (1 - \pi_{i,1}) + (r_2(K-k) - \pi_{i,2}). \tag{10}$$

4) State Transition Probability: The probability distribution for the next state depends on the current state and the action adopted by the SC, which is expressed as

$$\begin{aligned}\Pr\{s_{i+1}|s_i, \pi_i\} &= \Pr\{\tilde{b}_{i+1}, k_{i+1}|\tilde{b}_i, k_i, \pi_i\} \\ &= \Pr\{k_{i+1}\} \cdot \Pr\{\tilde{b}_{i+1}|\tilde{b}_i, k_i, \pi_i\},\end{aligned} \tag{11}$$

where $\Pr\{k_{i+1}\}$ is given in (5), and the expression for $\Pr\{\tilde{b}_{i+1}|\tilde{b}_i, k_i, \pi_i\}$ is given as follows.

1) If $\tilde{b}_i - u_i + e_i < B_{\max}$, then

$$\Pr\{\tilde{b}_{i+1}|\tilde{b}_i, k_i, \pi_i\} = \begin{cases} \Pr\{e_i\}, & \text{if } \tilde{b}_{i+1} = \tilde{b}_i - u_i + e_i, \\ 0, & \text{Otherwise,} \end{cases} \tag{12}$$

2) If $\tilde{b}_i - u_i + e_i \ge B_{\max}$, then

$$\Pr\{\tilde{b}_{i+1}|\tilde{b}_i, k_i, \pi_i\} = \begin{cases} \Pr\{e_i \in \Xi\}, & \text{if } \tilde{b}_{i+1} = B_{\max}, \\ 0, & \text{Otherwise,} \end{cases} \tag{13}$$

where $\Xi = \{e_i | \tilde{b}_i - u_i + e_i \ge B_{\max},\ e_i \in \{E_1, E_2, \ldots, E_L\}\}$.

B. Optimal Transmission Policy by Value Iteration

The transmission policy design goal is to find a policy that minimizes the expected discounted cumulative cost, which is

$$\arg\min_{\pi_i} \left\{ \lim_{I_0 \to \infty} \sum_{i=0}^{I_0} \alpha^i \mathbb{E}\{g_i(s_i, \pi_i)\} \right\}, \tag{14}$$

where α $(0 \le \alpha \le 1)$ is a discount factor.

Denote Π and V as two vectors indexed by the state, with $V(s_i)$ being the discounted cumulative cost by the policy in state s_i, while $\Pi(s_i)$ being the corresponding action for state s_i. Then, the original problem in (14) can be written as the following two-step problem in the stationary state [30]

$$\begin{aligned}\Pi(s_i) = \arg\min_{\pi_i \in \mathcal{A}} \Big\{ &\sum_{s_{i+1} \in \mathcal{S}} \Pr\{s_{i+1}|s_i, \pi_i\} \\ &\cdot \big[g_i(s_i, \pi_i) + \alpha V(s_{i+1})\big]\Big\},\end{aligned} \tag{15}$$

$$\begin{aligned}V(s_i) = &\sum_{s_{i+1} \in \mathcal{S}} \Pr\{s_{i+1}|s_i, \Pi(s_i)\} \\ &\cdot \big[g_i(s_i, \Pi(s_i)) + \alpha V(s_{i+1})\big],\ s_i \in \mathcal{S},\end{aligned} \tag{16}$$

where $\mathcal{S}$ is the set of system states and $\mathcal{A}$ is the set of feasible actions.

By repeating these two steps in some order for all the states until convergence, we can obtain the final optimal policy. Value iteration and policy iteration are two typical algorithms for solving (15)–(16) [28, Secs. 2.3 and 2.4]. In this paper, we use the former. It is known that the error bound of the value iteration algorithm monotonically decreases with iterations and the algorithm can terminate with arbitrarily small error [28, Proposition 1.3.1].

The value iteration procedure is given in Algorithm 4. It starts with an initial value $V^0(s_i)$, e.g., $V^0(s_i) = 0$, and iteratively solves (15)–(16) until $V(s_i)$ $(s_i \in \mathcal{S})$ converges. Upon convergence, the optimal policy $\Pi^*(s_i)$ is obtained, under which the original MDP degrades to a discrete Markov chain and the transition probability is given in (11). We can then compute the stationary distribution of this Markov chain,

Algorithm 4 Value Iteration Algorithm for Computing the Optimal Transmission Policy

Input:

$\mathcal{S}$ is the set of system states;
$\mathcal{A}$ is the set of feasible action strategies;
$\Pr\{s_{i+1}|s_i, \pi_i\}$ is the transition probability (where $s_i, s_{i+1} \in \mathcal{S}$ and $\pi_i \in \mathcal{A}$);
$g_i(s_i, \pi_i)$ is the cost by the macro BS (where $s_i \in \mathcal{S}$ and $\pi_i \in \mathcal{A}$);
ϵ is the convergence threshold value;

Output:

The optimal transmission policy $\Pi^*(s_i)$ (where $s_i \in \mathcal{S}$);

$j = 0$
$V^j = \mathbf{0}$, $V^{j+1} = \mathbf{2}\epsilon$
while $|V^{j+1} - V^j| > \epsilon$ **do**
  $j \leftarrow j + 1$
  for $s_i \in \mathcal{S}$ **do**
    $\Pi^{j+1}(s_i) = \arg\min_{\pi_i \in \mathcal{A}} \Big\{ \sum_{s_{i+1} \in \mathcal{S}} \Pr\{s_{i+1}|s_i, \pi_i\} \cdot \big[g_i(s_i, \pi_i) + \alpha V^j(s_{i+1})\big]\Big\}$
    $V^{j+1}(s_i) = \sum_{s_{i+1} \in \mathcal{S}} \Pr\{s_{i+1}|s_i, \Pi^{j+1}(s_i)\} \cdot \big[g_i(s_i, \Pi^{j+1}(s_i)) + \alpha V^j(s_{i+1})\big]$
  end for
end while
$\Pi^*(s_i) = \Pi^{j+1}(s_i)$ (where $s_i \in \mathcal{S}$);

denoted as $\Pr(s_i)$. Thus, the expected number of transmissions handled by the macro BS under the optimal policy is

$$\eta = \sum_{s_i \in \mathcal{S}} \Pr(s_i) \cdot g_i(s_i, \Pi^*(s_i)). \tag{17}$$

Note that both $r_1(k)$ and $r_2(K-k)$ depend on the size of the coded content set $\mathcal{C}_e$, i.e., $N = |\mathcal{C}_e|$. We should select the optimal N that minimizes the energy consumption at the macro BS, i.e.,

$$N^* = \arg \min_{N \in \{1,2,\ldots,N_0\}} \eta. \tag{18}$$

However, in order to solve (18), we need to run Algorithm 4 N_0 times. Alternatively, we can choose N to minimize the average required traffic load for these K requests from the users, i.e.,

$$\begin{aligned} &\arg \min_{N \in \{1,2,\ldots,N_0\}} \mathbb{E}\{r_1(k) + r_2(K-k)\} \\ &= \arg \min_{N \in \{1,2,\ldots,N_0\}} \sum_{k=0}^{K} \Pr\{k_i = k\} \cdot \big(r_1(k) + r_2(K-k)\big), \end{aligned} \tag{19}$$

where $\Pr\{k_i = k\}$ is given in (5), $r_1(k)$ and $r_2(K-k)$ are given in (6)–(7).

It will be shown in Section VI numerically that (18) and (19) give similar performance in terms of the average cost η while the complexity of (19) is much lower.

C. Optimal Transmission Policy for Uncoded Caching Systems

To serve as a comparison, this subsection will discuss the system in which the conventional uncoded caching scheme is employed, where all users store the M most popular contents to maximize the probability that the real-time requests can be satisfied by the proactively cached contents. For the specific case of $M = 0$, it becomes the system with no cache at the user end discussed in [19]. With respect to a particular request d_k from the k-th user, the probability that d_k has already been cached is

$$\begin{aligned} q_0 &\triangleq \Pr\{d_k \text{ has been cached at the user end}\} \\ &= \sum_{m=1}^{M} f_m = \frac{\sum_{m=1}^{M} (1/m)^{\nu}}{\sum_{j=1}^{N_0} (1/j)^{\nu}}. \end{aligned} \tag{20}$$

The probability that $k_i = k$ of K requested contents have already been stored in the cache of each user is

$$\Pr\{k_i = k\} = \binom{K}{k} \cdot q_0^k (1-q_0)^{K-k}, \quad k \in \{0, 1, 2, \ldots, K\}. \tag{21}$$

On the condition that k of K requested contents have been cached, the corresponding traffic load requirement for delivering these K requested contents to users is $r_2(K-k) = K-k$ and the possible action policy $\pi_i \in \{0, 1, 2, \ldots, r_2(K-k)\}$. The energy constraint at the SC is $E_0 \cdot \pi_i \le \tilde{b}_i$. So, the feasible action set in this system during the i-th time slot can be defined as

$$\mathcal{A}_i = \{\pi_i | \pi_i \in \{0, 1, 2, \ldots, r_2(K-k)\} \text{ and } E_0 \cdot \pi_i \le \tilde{b}_i\}. \tag{22}$$

And the cost, namely the number of transmissions that need to be handled by the macro BS, can be expressed as

$$g_i(s_i, \pi_i) = r_2(K-k) - \pi_i. \tag{23}$$

By plugging (20)–(23) into Algorithm 4, the optimal policy for the uncoded caching system can be obtained. In Section VI, we will compare its performance with that of the coded caching system via simulations.

V. Further Discussions

A. The Impact of Repetitive Requests

In the previous sections, similar to prior work in [11], we assumed that the requested contents by different users are distinct, namely the elements in requesting vector $\mathcal{D} = \{d_1, d_2, \ldots, d_K\}$ are different with each other. This corresponds to the worst-case scenario in terms of traffic off-loading, under which both the average required traffic load and the corresponding optimal transmission strategy have been obtained.

However, in practice, several users under the coverage of an SC may request the same content at the same time. Obviously, these requests can be fulfilled simultaneously by multicast. If we make full use of this property, the traffic load can be further reduced. Thus, this subsection intends to analyze

the impacts of repetitive requests among $\mathcal{D}$ upon the traffic load. More specifically, we need to derive the required traffic load distribution firstly. Then, by substituting the results into Algorithm 4, we can get the optimal transmission strategy for this case. Thus, the goal of this subsection is to obtain the required traffic load in the presence of repetitive requests.

Similar to the statement in Section III, assuming k_i of the K requests belong to the coded set $\mathcal{C}_e$, we denote the coded content set as $\mathcal{C}_e = \{c_1, c_2, \ldots, c_N\}$, and the subset of $\mathcal{D}$ that belongs to $\mathcal{C}_e$ as $\mathcal{D}' = \{d_{n_1}, d_{n_2}, \ldots, d_{n_{k_i}}\} \subseteq \mathcal{C}_e$. Given $d_j \in \mathcal{C}_e$, the conditional popularity that d_j corresponds to the l-th ranked content of $\mathcal{C}_e$ is

$$f_l' = \Pr\{d_j = c_l | d_j \in \mathcal{C}_e\} = \frac{1}{p_0} \cdot \frac{(1/l)^v}{\sum_{j=1}^{N_0} (1/j)^v}, \quad (24)$$

where $p_0 = \Pr\{d_j \in \mathcal{C}_e\}$ is given in (4).

Next, given that k_i of the K requests belong to the coded set $\mathcal{C}_e$, we need to find out how many non-repetitive requests there are in $\mathcal{D}'$ and $\mathcal{D}/\mathcal{D}'$, respectively. Denote the number of non-repetitive requests in $\mathcal{D}'$ as υ_1 ($\upsilon_1 \leq k_i$), and $\mathcal{D}' = \{d_{n_1}, d_{n_2}, \ldots, d_{n_{k_i}}\} = \{c_{l_1}, c_{l_2}, \ldots, c_{l_{k_i}}\}$ (where $d_{n_j} = c_{l_j}$). Then the number of contents in $\mathcal{D}'$ that need to be transmitted to users is υ_1, not k_i due to the repetition of requests. It is apparent that $\upsilon_1 \in \{1, 2, \ldots, k_i\}$. We have

$$\Pr\{\upsilon_1 | k_i = k\} = \sum_{(l_1, l_2, \ldots, l_k) \in \Theta_{\upsilon_1}} f_{l_1}' f_{l_2}' \ldots f_{l_k}', \quad (25)$$

where Θ_{υ_1} is defined as the set of all possible instances that meet the following constraint: $\Theta_{\upsilon_1} = \{(l_1, l_2, \ldots, l_k) \mid l_1, l_2, \ldots, l_k \in \{1, 2, \ldots, N\}, \|l_1, l_2, \ldots, l_k\|^\diamond = \upsilon_1\}$, with the operation $\|l_1, l_2, \ldots, l_k\|^\diamond$ denoting the number of distinct elements among $l_1, l_2, \ldots, l_k$.

Similarly, suppose the number of non-repetitive requests in $\mathcal{D}/\mathcal{D}'$ is υ_2 and denote $\mathcal{C}/\mathcal{C}_e = \{c_1', c_2', \ldots, c_{N_0-N}'\}$. Given that $d_j \in \mathcal{C}/\mathcal{C}_e$, the conditional popularity that d_j is the l-th ranked content of $\mathcal{C}/\mathcal{C}_e$ is

$$\begin{aligned} f_l'' &= \Pr\{d_j = c_l' | d_j \in \mathcal{C}/\mathcal{C}_e\} \\ &= \frac{1}{1 - p_0} \cdot \frac{1/(l+N)^v}{\sum_{j=1}^{N_0} 1/j^v}, \; l = 1, 2, \ldots, N_0 - N. \end{aligned} \quad (26)$$

We have

$$\begin{aligned} \Pr\{\upsilon_2 | k_i = k\} = \sum_{(l_1', l_2', \ldots, l_{(K-k)}') \in \Theta_{\upsilon_2}} f_{l_1'}'' f_{l_2'}'' \ldots f_{l_{(K-k)}'}'', \\ \upsilon_2 = 1, 2, \ldots, K - k. \end{aligned} \quad (27)$$

Finally, since υ_1 and υ_2 are independent, thus

$$\begin{aligned} \Pr\{\upsilon_1, \upsilon_2\} = \sum_{k=0}^{K} \Pr\{k_i = k\} \cdot \Pr\{\upsilon_1 | k_i = k\} \\ \cdot \Pr\{\upsilon_2 | k_i = k\}. \end{aligned} \quad (28)$$

By now, the exact number of distinct requests in $\mathcal{D}$ and the joint distribution of (υ_1, υ_2) have been derived. Then, replacing k and $K - k$ in the expressions of r_1 and r_2 in (6)–(7) by υ_1 and υ_2 respectively, the required traffic loads for delivering υ_1 non-repetitive coded contents and υ_2 non-repetitive uncoded contents can be derived. Lastly, by using the derived results in Algorithm 4, the optimal transmission strategy for systems with repetitive requests can be obtained, the performance of which will be illustrated via simulations in Section VI-D.

B. A Holistic Energy Consumption Model for Energy-Harvesting-Based SC

To make system model further more practical, we write the total energy consumption in each time slot as

$$u_i = P_T + P_F + P_P, \quad (29)$$

where P_T, P_F and P_P denote energy consumptions for signal transmission, fetching and the baseband processing, respectively.

P_T has been considered before, which is proportional to the number of transmissions. We now consider the effects of P_F and P_P. Fig. 2 shows a complete round of the content delivery process to handle these K requests from the users. Specifically, the K requests by the users are sent to both the associated SC and the macro BS at the beginning of the time slot. If the SC has enough energy to handle part or all of these requests, it will send a message to the macro BS, and delivers the corresponding contents to the users. If there are some remaining requests that have not been delivered by the SC, they will be handled by the macro BS. On the other hand, if the SC does not have sufficient energy, it is not able to send the message to the macro BS and the macro BS will handle all requests.

In (29), P_F is the energy consumption by the SC on transmitting message to the macro BS, so that the macro BS can know whether the SC is active and if so how many requests it can handle [19]. On the other hand, the baseband processing energy consumption P_P is essential for SC to remain in the active state, so that it can process and store data, and then transmit message. If the available energy level of the battery at the beginning of the time-slot is more than the basic energy consumption $E_u = P_F + P_P$, the SC can send the message to the macro BS and deliver part or all of the K requests depending on the energy level of the battery. Otherwise, the SC will be in the sleep state without consuming any energy, and of course cannot deliver any of the requests or update its cache.

C. Transmission Policy Under Repetitive Requests and Holistic Energy Model

When the effects of repetitive requests and the basic energy consumption are taken into account, the system state becomes $s_i = (\tilde{b}_i, \upsilon_{i,1}, \upsilon_{i,2}, q_i)$, in which the new indicator variable q_i represents the activation status of SC during the i-th time slot. Specifically, if $q_i = 1$, the SC is active during the current time-slot, and E_u amount of energy needs to be consumed firstly to keep the SC active. Otherwise, $q_i = 0$, the SC is inactive and no energy is consumed. Besides, the energy constraint in (8) should be rewritten as

$$u_i = \left(E_0 \cdot (r_1(\upsilon_1) \cdot \pi_{i,1} + \pi_{i,2}) + E_u\right) \cdot q_i \leq \tilde{b}_i, \quad (30)$$

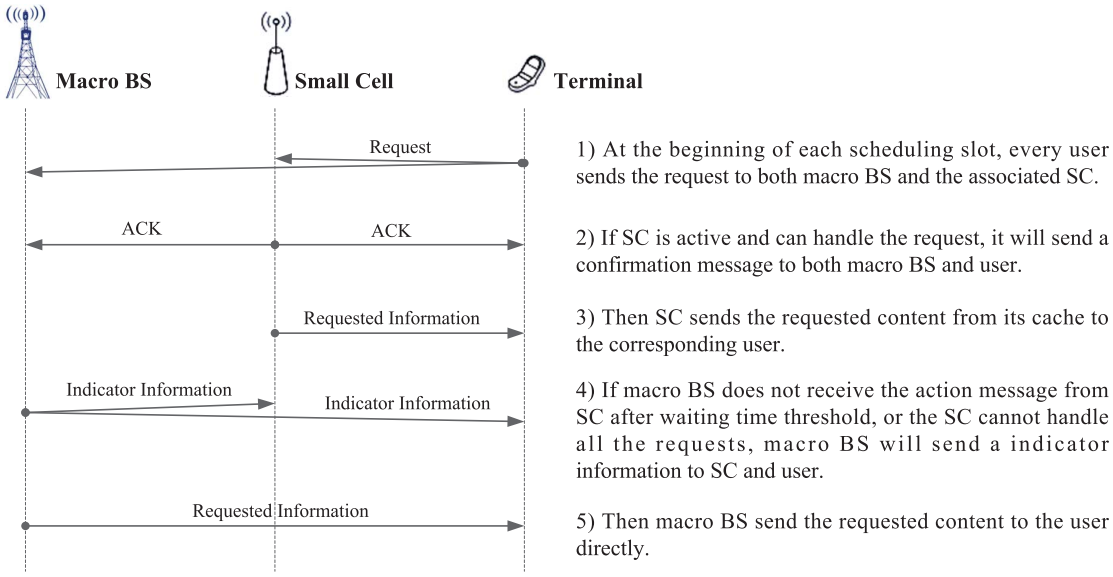

Fig. 2. The sequence diagram for a complete content delivery process during a scheduling slot.

And the state transition probability in (11) should be modified as

$$\begin{aligned}&\Pr\{s_{i+1}|s_i,\pi_i\}\\&\quad=\Pr\{\tilde{b}_{i+1},\upsilon_{i+1,1},\upsilon_{i+1,2},q_{i+1}|\tilde{b}_i,\upsilon_{i,1},\upsilon_{i,2},q_i,\pi_i\}\\&\quad=\Pr\{\upsilon_{i+1,1},\upsilon_{i+1,2}\}\cdot\Pr\{\tilde{b}_{i+1}|\tilde{b}_i,\upsilon_{i,1},\upsilon_{i,2},q_i,\pi_i\}\\&\qquad\cdot\Pr\{q_{i+1}|\tilde{b}_{i+1}\},\end{aligned}\tag{31}$$

in which

$$\begin{aligned}\Pr\{q_{i+1}=1|\tilde{b}_{i+1}\}&=1-\Pr\{q_{i+1}=0|\tilde{b}_{i+1}\}\\&=\begin{cases}1, & \text{when } \tilde{b}_{i+1}\geq E_u,\\0, & \text{Otherwise.}\end{cases}\end{aligned}\tag{32}$$

And $\Pr\{\tilde{b}_{i+1}|\tilde{b}_i,\upsilon_{i,1},\upsilon_{i,2},q_i,\pi_i\}$ is expressed similarly as in (12)–(13) with u_i given in (30).

Then, substituting the transition probability matrix in (31) and other parameters into Algorithm 4, the optimal transmission strategy can be obtained. The effects of E_u and repetitive requests on the average number of transmissions handled by the macro BS, η, will be investigated via simulations in Section VI.

VI. Numerical Results

Firstly, we study the effects of the cache size and the content popularity on the traffic load requirements for different caching schemes. Then, the performance of the optimal transmission policy obtained by Algorithm 4 is compared with some other heuristic-based policies. Lastly, we illustrate the effects of the holistic energy consumption model as well as the repetitive requests.

A. Comparison of Different Caching Schemes

Suppose that the number of available contents at the macro BS is $N_0 = 200$, and the number of users under the coverage of each SC is $K = 50$. The content popularity follows the ZipF distribution with parameter ν. We compare the average required traffic load for successfully decoding these K requests from users under several different caching schemes. In particular, the average required traffic load under our proposed coded caching $\mathbb{E}\{r_1(k)+r_2(K-k)\}$ is given by (6)–(7) (where the expectation is with respect to the variable k), in which the value of N is optimized over $\{0,1,2,\ldots,N_0\}$. On the other hand, the traffic load under no caching is always K. In the uncoded caching scheme, the M most popular contents are entirely stored in each user's cache and the average traffic load is $\sum_{k=0}^{K}[(K-k)\cdot\Pr\{k_i=k\}]$, where $\Pr\{k_i=k\}$ is given in (21). Besides, since optimal caching scheme for non-uniform content popularity distribution is still an open problem, and the multi-group coded caching proposed by [27] is the best one among the existing schemes, it is also considered in this subsection to serve as a baseline. According to multi-group coded caching, the content set $\mathcal{C}$ should be divided into four groups with group sizes $\{4,16,64,116\}$ when $\nu=0.5$ and divided into seven groups with group sizes $\{2,4,8,16,32,64,74\}$ when $\nu=1$. Since there is no explicit traffic load expression for it, we use the upper bound in [27].

Fig. 3 shows the average required traffic loads as a function of the cache size M at the user end for $\nu=0.5$ and $\nu=1$. It can be seen that the data loads decrease rapidly with the cache size. Compared with the no caching case, even a small cache, e.g., $M=1$ or 2, can substantially decrease the data loads. Moreover, the coded system has a smaller data load than the uncoded system, and the difference is more significant with relatively small cache size. When the cache size becomes large, the performance gap of the two caching systems become smaller. Moreover, it is seen that the proposed coded caching scheme outperforms the multi-group coded caching scheme in [27]. In multi-group coded caching scheme, the cache size M is divided into pieces for different content groups

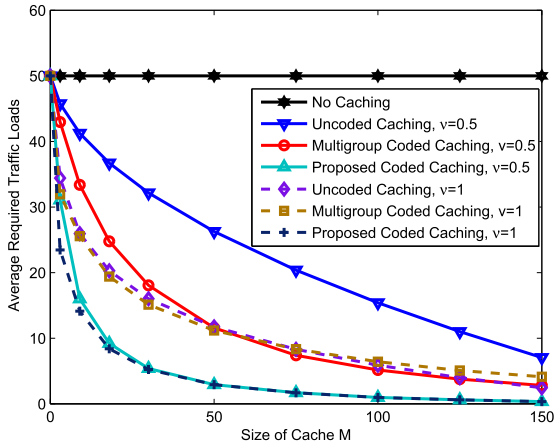

Fig. 3. The traffic load versus the user cache size.

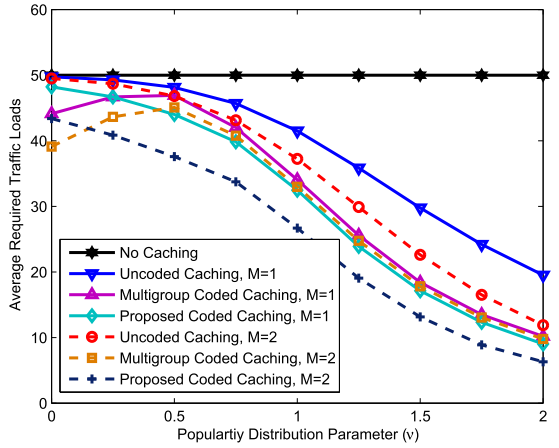

Fig. 4. The traffic load versus the popularity distribution parameter ν.

to cache proactively. Regarding the results in (3), it can be seen that reduced value of cache size weakens the benefit from multicast transmission. It is why the proposed scheme performs better.

Fig. 4 shows the traffic load as a function of the popularity distribution parameter ν for $M = 1$ and $M = 2$. According to (1), it is easy to see that the contents are equiprobable when $\nu = 0$. As ν becomes larger, less contents will be requested by the K users with higher probability. Both the uncoded and coded caching schemes can exploit this nonuniform distribution to reduce the traffic load. But the performance of the coded caching scheme is uniformly better than that of the uncoded scheme for the entire range of ν. For multigroup coded caching, the traffic load increases with ν and then decreases. It may be because that we just used the upper bound to evaluate the performance as in [27]. If we search over the whole multiple parameter dimensions, better performance can be obtained.

B. Performance of the Optimal Transmission Policy

To serve as comparisons, we also consider some simple heuristic policies. For instance, the greed policy [31], which tries to minimize the current cost value as much as possible

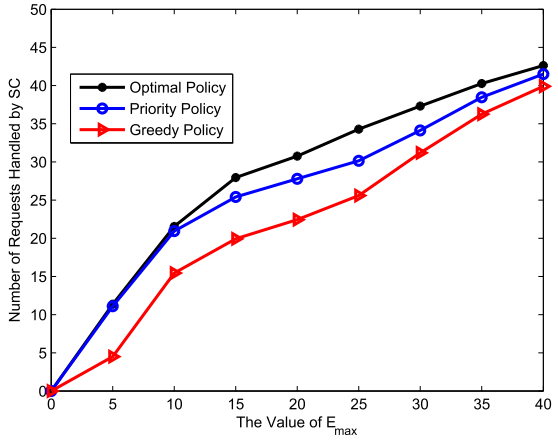

Fig. 5. Comparison of different transmission policies.

and ignores the cost of future. It means that the SC under greed policy will handle the users' requests unless the energy is exhausted. Another example is the priority policy, under which the SC only delivers coded content and ignores the uncoded content requests.

Suppose that the coded caching scheme in Section III is employed with the size of the content set $N_0 = 200$, popularity parameter $\nu = 1$, the size of cache $M = 2$, the battery capacity of SC $B_{\max} = 100$. The energy harvesting profile is $L = 2$, namely the widely used Bernoulli distribution model $\{0, E_{\max}\}$ with probability $\{0.5, 0.5\}$ [21], [22]. The size of the coding set N is optimized over $\{0, 1, 2, \ldots, N_0\}$ based on (19). The value space of the discretized available energy level in the battery $\tilde{b}_i$ of SC is $\{0, 1, 2, \ldots, B_{\max}\}$, and the value space of k_i is $\{0, 1, 2, \ldots, K\}$. So the total number of states of the MDP is $(B_{\max}+1)\times(K+1)$. Fig. 5 illustrates the performance, i.e., the average number of requests that can be handled by the SC, under the greedy policy, priority policy and optimal policy obtained by Algorithm 4, respectively. It is seen that the proposed transmission scheme outperforms the other two schemes, since Algorithm 4 obtain the optimal action strategy for SC depending on not only the current cost but also the discounted future cost. Fig. 6 illustrates the convergence of Algorithm 4, with $B_{\max} = 100$ and $E_{\max} = 30$, where the relative error $\frac{|V^{j+1}-V^j|}{|V^j|}$ is plotted against the iteration number. It is seen that convergence is reached within about 10 iterations.

C. Overall System Performance

We next study the performance of the overall system with coded content caching and optimal transmission policy. Both the distributed coded caching and centralized coded caching as well as the conventional no caching and uncoded caching systems are considered with their corresponding optimal transmission policies. The simulation parameters are listed in Table I.

Fig. 7 depicts the number of transmissions handled by the macro BS versus the battery capacity for various systems with $E_{\max} = 50$. It is seen that compared with the system with no caching, the systems with caching can offload a significant amount of traffic from the macro BS. Moreover, although the

TABLE I
PARAMETER SETTINGS FOR THE SYSTEM SIMULATION

PARAMETER	DESCRIPTIONS	VALUE
N_0	The size of available contents at the macro BS	200
ν	The parameter for Zipf distribution	1
K	The number of individual users under each SC	50
M	The size of the cache at each user	2
E_0	The energy consumption for delivering one unit content	1
$\{p_1, p_2, ..., p_L\}$	The energy harvesting probability distribution	$\{0.5, 0.5\}$
$\{E_1, E_2, ..., E_L\}$	The energy harvesting profile	$\{0, E_{\max}\}$
$B_{\max}$	The capacity of the energy battery at SCs	–

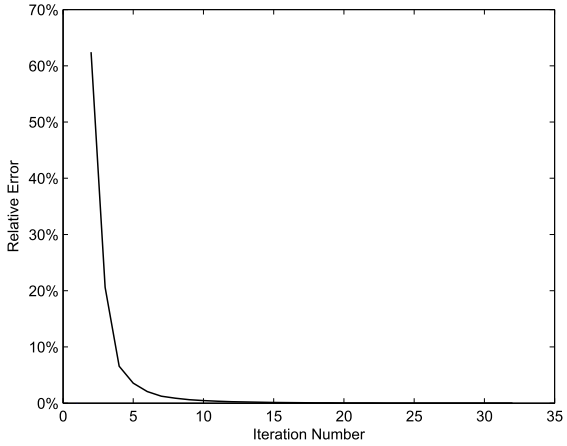

Fig. 6. Convergence of Algorithm 4.

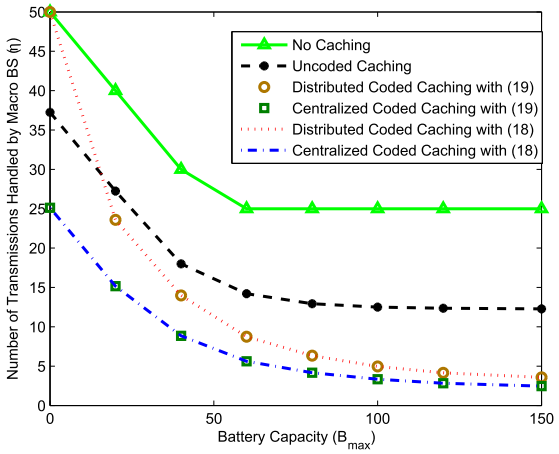

Fig. 7. Overall system performance versus battery capacity.

coded caching systems outperform the uncoded system when $B_{\max}$ is large, for small $B_{\max}$, the uncoded system actually outperforms the distributed coded system. It is because all requests need to be entirely transmitted by the macro BS to the K users in the no caching system and distributed coded caching system when $B_{\max}$ is small, since the SC cannot work well due to lack of energy. So the number of transmissions in both cases is nearly K. For the uncoded caching, M most popular contents have already been entirely cached by all users, and no BS transmission is needed if these contents are requested. Thus, even when $B_{\max}$ is small, the number

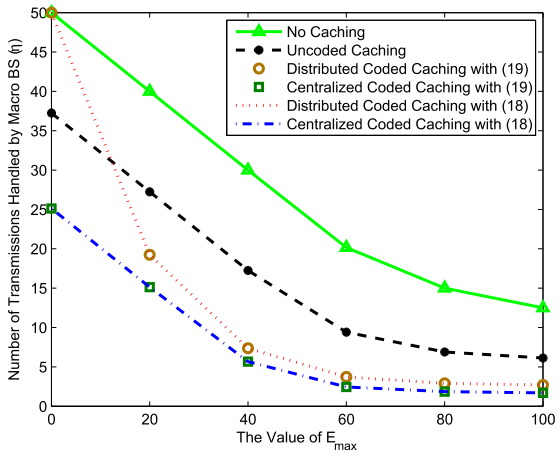

Fig. 8. The overall system performance versus the energy-harvesting level at the SCs.

of transmissions of macro BS is further less than K, which is superior to that of no caching and distributed coded caching systems.

On the other hand, in the centralized coded caching system, the macro BS also knows the cache contents at the user end, so that it can transmit coded data instead of uncoded data to the users. As a result, its performance is superior to that of the other three systems. Even when $B_{\max} = 0$, the centralized coded caching can benefit from coded caching contents at the user end. So, the number of transmissions by BS in this condition is also less than K. Furthermore, it is seen that for both the centralized and distributed coded systems, the performance using the encoding size given by (19) is essentially the same as that using the optimal encoding size given by (18), even though the former has a much lower computational complexity.

Fig. 8 depicts the number of transmissions handled by the macro BS versus the energy-harvesting level $E_{\max}$ at the SCs with $B_{\max} = 100$. Compared with the no caching system, again it is seen that the caching at the user end can lead to significant traffic offload.

D. The Effects of the Repetitive Requests and the Basic Energy Consumptions at SCs

As discussed in Section V, the SC needs to consume E_u energy units during a time slot to remain in the active state (more specifically, sending the feedback information to

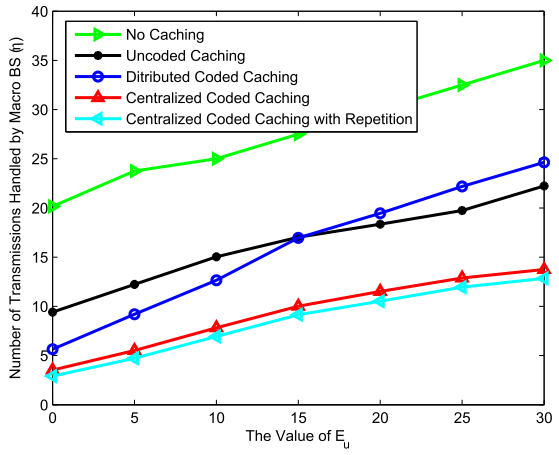

Fig. 9. Overall system performance with basic energy consumptions and repetitive requests.

the macro BS, storing the information in the buffer, baseband signal processing, etc). Suppose that $E_{\max} = 60$ and $B_{\max} = 100$, and other parameters are just the same as those in Table I. Fig. 9 shows the number of transmissions handled by the macro BS versus the basic energy consumption E_u in several different systems. It can be observed that the number of transmissions increase, i.e., the data offload decreases with the basic energy consumption in all systems, since the SCs have less energy to transmit data than before. The centralized coded system still performs the best, whereas the uncoded system eventually outperforms the distributed coded system as E_u increases. Further, if the repetitive requests among the K requests of users are also considered, the distribution for the traffic load requirement has been given in (28). The performance of the centralized coded caching system with repeated requests is also shown in Fig. 9.

VII. Conclusions

We have developed a proactive coded content caching scheme and an optimal transmission scheme for a system consisting of a macro base station (BS) and several energy-harvesting small cells (SCs). The key feature of the proposed coded caching strategy is that it enables multiple contents requested by different users to be delivered simultaneously by the BS or the SC. The optimal transmission policy is obtained by solving a Markov decision process (MDP) via value iteration. Compared with the conventional no caching and uncoded caching systems, much traffic can be offloaded from the link between the macro BS and users to that between the energy-harvesting SC and users, leading to a significant reduction of energy consumption from the power grid.

References

[1] "Cisco visual networking index: Global mobile data traffic forecast update, 2015–2020," Cisco, San Jose, CA, USA, White Paper, Feb. 2016. [Online]. Available: http://www.cisco.com/c/en/us/solutions/collateral/service-provider/visual-networking-index-vni/mobile-white-paper-c11-520862.html

[2] M. Cha, H. Kwak, P. Rodriguez, Y.-Y. Ahn, and S. Moon, "I tube, you tube, everybody tubes: Analyzing the world's largest user generated content video system," in *Proc. 7th ACM SIGCOMM Conf. Internet Meas.*, New York, NY, USA, 2007, pp. 1–14.

[3] B. Bai and Z. Cao, "A convergence scheme for digital video/audio broadcasting network and broadband wireless access network," in *Proc. IEEE Int. Wireless Commun. Netw. Conf.*, Mar. 2007, pp. 3291–3295.

[4] G. RAN. (Mar. 2008). *Introduction of the Multimedia Broadcast Multicast Service (MBMS) in the Radio Access Network (RAN); Stage 2 (Release 7), 3g ts 25.346 v7.7.0*. [Online]. Available: www.qtc.jp/3GPP/Specs/25346-750.pdf

[5] D. Lecompte and F. Gabin, "Evolved multimedia broadcast/multicast service (eMBMS) in LTE-advanced: Overview and Rel-11 enhancements," *IEEE Commun. Mag.*, vol. 50, no. 11, pp. 68–74, Nov. 2012.

[6] J. Inofuentes. (Feb. 2013). *AT&T Announces Plans to Use 700Mhz Channels for LTE Broadcast*. [Online]. Available: http://arstechnica.com/gadgets/2013/09/att-announces-plans-to-use-700mhz-channels-for-lte-broadcast/

[7] E. Alvarez. (Jan. 2015). *How AT&T Will Deliver TV (and More) Over Crowded LTE*. [Online]. Available: https://www.engadget.com/2015/01/14/att-lte-broadcast/

[8] K. Wang, Z. Chen, and H. Liu, "Push-based wireless converged networks for massive multimedia content delivery," *IEEE Trans. Wireless Commun.*, vol. 13, no. 5, pp. 2894–2905, May 2014.

[9] E. Bastug, M. Bennis, and M. Debbah, "Living on the edge: The role of proactive caching in 5G wireless networks," *IEEE Commun. Mag.*, vol. 52, no. 8, pp. 82–89, Aug. 2014.

[10] J. G. Andrews *et al.*, "What will 5G be?" *IEEE J. Sel. Areas Commun.*, vol. 32, no. 6, pp. 1065–1082, Jun. 2014.

[11] M. A. Maddah-Ali and U. Niesen, "Fundamental limits of caching," *IEEE Trans. Inf. Theory*, vol. 60, no. 5, pp. 2856–2867, May 2014.

[12] S. Zhou, J. Gong, Z. Zhou, W. Chen, and Z. Niu, "GreenDelivery: Proactive content caching and push with energy-harvesting-based small cells," *IEEE Commun. Mag.*, vol. 53, no. 4, pp. 142–149, Apr. 2015.

[13] N. Sharma, D. K. Krishnappa, D. Irwin, M. Zink, and P. Shenoy, "GreenCache: Augmenting off-the-grid cellular towers with multimedia caches," in *Proc. 4th ACM Multimedia Syst. Conf.*, Feb./Mar. 2013, pp. 271–280.

[14] E. Bastug, M. Bennis, and M. Debbah. (Apr. 2015). *Proactive Caching in 5G Small Cell Networks*. [Online]. Available: www.laneas.com/sites/default/files/publications/1/BastugEtAl-ProChapter2015-Final.pdf

[15] N. Golrezaei, K. Shanmugam, A. G. Dimakis, A. F. Molisch, and G. Caire, "FemtoCaching: Wireless video content delivery through distributed caching helpers," in *Proc. IEEE INFOCOM*, Mar. 2012, pp. 1107–1115.

[16] K. Shanmugam, N. Golrezaei, A. G. Dimakis, A. F. Molisch, and G. Caire, "FemtoCaching: Wireless content delivery through distributed caching helpers," *IEEE Trans. Inf. Theory*, vol. 59, no. 12, pp. 8402–8413, Dec. 2013.

[17] X. Wang, M. Chen, T. Taleb, A. Ksentini, and V. C. M. Leung, "Cache in the air: Exploiting content caching and delivery techniques for 5G systems," *IEEE Commun. Mag.*, vol. 52, no. 2, pp. 131–139, Feb. 2014.

[18] J. Gong, S. Zhou, Z. Zhou, and Z. Niu, "Proactive push with energy harvesting based small cells in heterogeneous networks," in *Proc. IEEE Int. Conf. Commun. (ICC)*, Jun. 2015, pp. 25–30.

[19] A. Kumar and W. Saad, "On the tradeoff between energy harvesting and caching in wireless networks," in *Proc. IEEE Int. Conf. Commun. Workshop (ICCW)*, Jun. 2015, pp. 1976–1981.

[20] N. Golrezaei, A. F. Molisch, A. G. Dimakis, and G. Caire, "Femtocaching and device-to-device collaboration: A new architecture for wireless video distribution," *IEEE Commun. Mag.*, vol. 51, no. 4, pp. 142–149, Apr. 2013.

[21] Y. Dong, F. Farnia, and A. Özgür, "Near optimal energy control and approximate capacity of energy harvesting communication," *IEEE J. Sel. Areas Commun.*, vol. 33, no. 3, pp. 540–557, Mar. 2015.

[22] M. Ashraphijuo, V. Aggarwal, and X. Wang, "On the capacity of energy harvesting communication link," *IEEE J. Sel. Areas Commun.*, vol. 33, no. 12, pp. 2671–2686, Dec. 2015.

[23] M. Leconte, G. Paschos, L. Gkatzikis, M. Draief, S. Vassilaras, and S. Chouvardas, "Placing dynamic content in caches with small population," in *Proc. 35th Annu. IEEE Int. Conf. Comput. Commun. (INFOCOM)*, Apr. 2016, pp. 1–9.

[24] S. Traverso, M. Ahmed, M. Garetto, P. Giaccone, E. Leonardi, and S. Niccolini, "Temporal locality in today's content caching: Why it matters and how to model it," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 43, no. 5, pp. 5–12, Oct. 2013.

[25] G. Paschos, E. Bastug, I. Land, G. Caire, M. Debbah, "Wireless caching: Technical misconceptions and business barriers," *IEEE Commun. Mag.*, vol. 54, no. 8, pp. 16–22, Aug. 2016.

[26] B. Zhou, Y. Cui, and M. Tao, "Optimal dynamic multicast scheduling for cache-enabled content-centric wireless networks," in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, Jun. 2015, pp. 1412–1416.
[27] U. Niesen and M. A. Maddah-Ali, "Coded caching with nonuniform demands," in *Proc. IEEE Conf. Comput. Commun. Workshops (INFOCOM WKSHPS)*, Apr. 2014, pp. 221–226.
[28] D. P. Bertsekas, *Dynamic Programming and Optimal Control*. Belmont, MA, USA: Athena Scientific, 1995.
[29] B. Zhou, Y. Cui, and M. Tao, "Stochastic content-centric multicast scheduling for cache-enabled heterogeneous cellular networks," *IEEE Trans. Wireless Commun.*, vol. 15, no. 9, pp. 6284–6297, Sep. 2016.
[30] R. Bellman, "A Markovian decision process," RAND CORP: SANTA MONICA, CA, USA, Tech. Rep. P-1066, 1957.
[31] Z. Wang, V. Aggarwal, and X. Wang, "Iterative dynamic water-filling for fading multiple-access channels with energy harvesting," *IEEE J. Sel. Areas Commun.*, vol. 33, no. 3, pp. 382–395, Mar. 2015.

Tao Li received the B.S. and M.S. degrees from the School of Electronic and Information Engineering, Harbin Institute of Technology, China, in 2010 and 2012, respectively. He is currently pursuing the Ph.D. degree with the Department of Electronic Engineering, Tsinghua University, Beijing, China. From 2014 to 2015, he visited the Department of Electrical Engineering, Columbia University, USA, as a Research Scholar. His current research interests include high mobility wireless communications, wireless cooperative networks, and energy harvesting. He received the Best Paper Awards of the 2013 International Workshop on High Mobility Wireless Communications and the 2014 IEEE Global Communications Conference (GLOBECOM). He is a Reviewer of several journals of the IEEE Communication Society, including the IEEE COMMUNICATION LETTERS, the IEEE WIRELESS COMMUNICATIONS LETTERS, and the IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS. He also has served as a Reviewer for several international conferences, including the IEEE Globecom and PIMRC.

Mehdi Ashraphijuo received the B.Sc. and M.Sc. degrees in electrical engineering from the Sharif University of Technology, Tehran, Iran, in 2010 and 2012, respectively. He is currently pursuing the Ph.D. degree with the Department of Electrical Engineering, Columbia University, New York City, NY, USA. His research interests lie in the general areas of network information theory and statistical signal processing with applications in wireless communication. His current research focuses primarily on developing fundamental principles for communication network design, with emphasis on developing interference management and understanding the role of feedback and user cooperation. He was a recipient of Qualcomm Innovation Fellowship Award in 2014.

Xiaodong Wang (S'98–M'98–SM'04–F'08) received the Ph.D. degree in electrical engineering from Princeton University. He is currently a Professor of Electrical Engineering with Columbia University, New York City. His research interests fall in the general areas of computing, signal processing, and communications. He has authored extensively in these areas. Among his publications the book entitled *Wireless Communication Systems: Advanced Techniques for Signal Reception*, (Prentice Hall, 2003). His current research interests include wireless communications, statistical signal processing, and genomic signal processing. He received the 1999 NSF CAREER Award, the 2001 IEEE Communications Society and Information Theory Society Joint Paper Award, and the 2011 IEEE Communication Society Award for Outstanding Paper on New Communication Topics. He has served as an Associate Editor of the IEEE TRANSACTIONS ON COMMUNICATIONS, the IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS, the IEEE TRANSACTIONS ON SIGNAL PROCESSING, and the IEEE TRANSACTIONS ON INFORMATION THEORY. He is listed as an ISI Highly-cited Author.

Pingyi Fan (M'03–SM'09) received the B.S. degree from the Department of Mathematics, Hebei University, in 1985, the M.S. degree from Nankai University in 1990, and the Ph.D. degree from the Department of Electronic Engineering, Tsinghua University, Beijing, China, in 1994. From 1997 to 1998, he visited The Hong Kong University of Science and Technology as a Research Associate. From 1998 to 1999, he visited the University of Delaware, USA, as a Research Fellow. In 2005, he visited NICT, Japan, as a Visiting Professor. From 2005 to 2011, he visited The Hong Kong University of Science and Technology for many times. In 2011, he was a Visiting Professor with the Institute of Network Coding, The Chinese University of Hong Kong. He is currently a Professor with the Department of EE, Tsinghua University.

Dr. Fan main research interests include 5G technology in wireless communications, such as massive MIMO, OFDMA, network coding, network information theory, and cross layer design. He is an overseas member of IEICE and a TPC Member of the IEEE ICC, Globecom, WCNC, VTC, and Inforcom. He has organized many international conferences, including as the General Co-Chair of the IEEE VTS HMWC2014 and the TPC Co-Chair of the IEEE International Conference on Wireless Communications, Networking and Information Security (2010). He has served as an Editor of the IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS, the *International Journal of Ad Hoc and Ubiquitous Computing* (Inderscience), and the *Journal of Wireless Communication and Mobile Computing* (Wiley). He is also a reviewer of more than 32 international journals, including the 18 IEEE journals and 8 EURASIP journals. He has received some academic awards, including the IEEE Globecom 2014 Best Paper Award, the IEEE WCNC'08 Best Paper Award, the ACM IWCMC'10 Best Paper Award, and the IEEE ComSoc Excellent Editor Award for the IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS in 2009.