Benefits of Edge Caching With Coded Placement for Asymmetric Networks and Shared Caches

Abdelrahman M. Ibrahim, Ahmed A. Zewail, *Member, IEEE*, and Aylin Yener, *Fellow, IEEE*

***Abstract*—This paper considers a cache-aided network where the users have access to helper-caches with heterogeneous sizes. First, coded placement schemes are proposed that exploit the heterogeneity in cache sizes when one user is connected to each cache. In the proposed scheme, the unicast/multicast signals intended to serve users connected to small memories are utilized in decoding the contents of the larger memories. A reduction in delivery load with coded placement is shown compared to uncoded placement for three-user systems with arbitrary cache sizes and larger systems in the small total memory regime. Next, systems with equal-size caches where multiple users are associated with each cache are considered. It is shown that coded placement outperforms the best uncoded placement scheme. In the proposed scheme, the unicast/multicast signals sent to the overloaded helper-caches facilitate the decoding of the coded subfiles stored at the underloaded helper-caches. The gain from coded placement is explicitly characterized for two-cache systems. For larger systems, the parameters of the coded placement scheme are obtained by optimization. It is observed that the gain from coded placement becomes more evident with increasing asymmetry in users' connectivity. Finally, a unified coded placement scheme for two-cache systems that exploits the asymmetry in both the cache sizes and the connectivity pattern is presented.**

***Index Terms*—Coded caching, coded placement, shared caches, network load reduction, unequal cache sizes.**

I. Introduction

In CODED caching [1], cache memories at the network edge are utilized in reducing network congestion during peak-traffic hours. Judicious design of the cache contents during off-peak hours, known as the *placement phase*, facilitates serving multiple users using coded multicast signals and in turn reducing the delivery cost in peak-traffic hours, which is known as the *delivery phase*. Extensive efforts have transpired towards understanding the fundamental trade-off between the delivery load and the cache sizes in several network topologies with different considerations in the recent years, see for example [1]–[29] and many others. References [15], [18], [20], [22], [24], [25] in particular have considered users with uneven storage capabilities and analyzed the impact of heterogeneity in cache sizes on the delivery load. Optimal caching schemes for such systems with uncoded placement have been studied in our previous work [15], [16]. Next generation content delivery networks with cache-aided small-cells/access-points where the users share the helper-caches have been considered for uncoded placement in [26], and coded placement with message exchange between caches in [27], [30].[1] Caching with multiple (same number of) file requests per user have been investigated in [19], [31].

Coded caching schemes are classified according to whether the files are encoded or not before being cached. In caching with uncoded placement, cache memories are populated with uncoded pieces of the files [1], [15], [18]. In caching with *coded placement*, the server places coded pieces of the files in the cache memories which are later decoded using the transmissions in the delivery phase[6], [9], [24], [25], [27]. Coding over placed contents, in general, has the potential to outperform caching with uncoded placement, if such coding is feasible at the content distributor. For example, [6], [9] have studied the benefits of coded placement in systems with more users than files in the small memory regime. References [24], [25], [27] have proposed coded placement schemes for systems where a subset of users have no caches.

In this work, we investigate whether coded placement is beneficial in systems where the users are connected to cache-helpers. We consider that these helpers have heterogeneous capabilities, with varying cache sizes or the number of users they may be helping, see Fig. 1. First, we study systems with one user per cache and demonstrate that coded placement outperforms uncoded placement for unequal caches. We show that coded placement increases both the users' local caching gains and the multicast gain in the system. For three-user systems, we present a caching scheme with coded placement that achieves a lower worst-case delivery load compared to the best caching scheme with uncoded placement in [15]. The proposed scheme makes use of the transmissions intended to serve users associated with small cache memories in decoding the cached coded pieces at the larger cache memories. We observe that the gain from coded placement increases with the heterogeneity in cache sizes, and decreases with the number of files in the library. We extend the proposed scheme to systems

Manuscript received May 27, 2021; revised September 10, 2021; accepted November 3, 2021. Date of publication November 11, 2021; date of current version December 23, 2021. This work was supported in part by NSF under Grant CCF-2105872. This work was presented in part at the 52nd Asilomar Conference on Signals, Systems and Computers, Pacific Grove, CA, USA, 2018, and the IEEE International Conference on Communications (ICC), Shanghai, China, 2019. *(Corresponding author: Aylin Yener.)*

Abdelrahman M. Ibrahim and Ahmed A. Zewail were with The Pennsylvania State University, State College, PA 16801 USA (e-mail: ami137@psu.edu; zewail@psu.edu).

Aylin Yener is with the Department of Electrical and Computer Engineering, The Ohio State University, Columbus, OH 43210 USA (e-mail: yener@ece.osu.edu).

Digital Object Identifier 10.1109/JSAIT.2021.3127435

[1] As noted explicitly in [30] footnote 3, our work, first presented in Asilomar 2018 predates [27] and [30], which is independent work from ours. We would like to thank the authors of [30] for this acknowledgement.

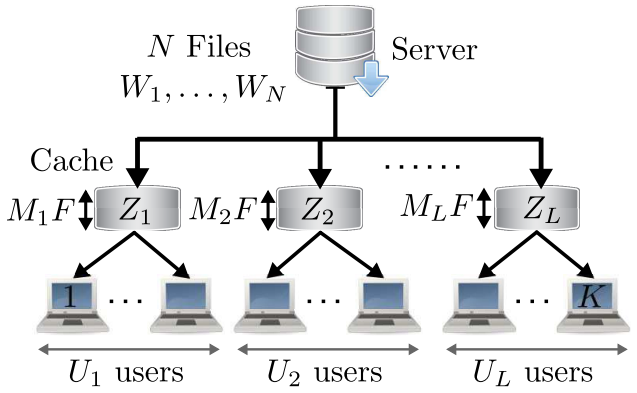

Fig. 1. Caching system with L unequal caches connected to K end-users.

with more than three users and show the reduction in delivery load with coded placement in the small total memory regime.

Motivated by this result, we next investigate systems with more users than helper-caches, i.e., each user can connect to one cache that it shares with multiple other users. We study the benefits of coded placement when the number of users connected to each helper-cache differ from one another. The asymmetry in the user-to-cache assignment is exploited by the proposed coded placement scheme. Based on the network connectivity pattern, in the proposed scheme, uncoded pieces of the files are placed in caches shared by a larger number of users, while storing coded pieces in the remaining caches. We first explain the coded placement scheme for two-cache systems with arbitrary number of users, then generalize the caching scheme to larger systems. Finally, we consider a system with two helper-caches which have different sizes. We propose a unified coded placement scheme that utilize the asymmetry in both the cache sizes and the number of users connected to each cache. Though demonstrated by two shared caches only, this part aims to close the loop towards establishing the benefit of coded placement in shared edge caches *with* asymmetry.

The remainder of this paper is organized as follows. In Section II, we describe the system model and the two operational phases in cache-aided networks. Systems where each helper-cache is connected to one user are analyzed in Section III. In Section IV, we consider systems with shared helper-caches. A unified coded placement scheme for heterogeneous helper-caches is presented in Section V. Finally, we draw our conclusions, comment on the scope of the proposed schemes and point to future directions in Section VI.

II. System Model

Notation: Vectors are represented by boldface letters, $\oplus$ refers to bitwise XOR operation, $|W|$ denotes size of W, $\mathcal{A} \setminus \mathcal{B}$ denotes the set of elements in $\mathcal{A}$ and not in $\mathcal{B}$, $[K] \triangleq \{1, \ldots, K\}$, and ϕ denotes the empty set. $\bigcup_{n=1}^{N} W_n$ denotes the union of the elements W_n, interpreted as the concatenation of the subfiles.

We consider a K-user system where a content server is connected to L cache-helpers via a shared error-free multicast link and each of the K end-users is connected to one of the L caches via an error-free link, as shown in Fig. 1. A library of N files, $\{W_1, \ldots, W_N\}$, is stored at the server, each with size F symbols. We consider a heterogeneous network with $L \leq K$ caches of unequal size and the number of users connected to each cache can be different. In particular, cache i is of size M_iF symbols and it is connected to the users in the set $\mathcal{U}_i$, where $U_i = |\mathcal{U}_i|$ and $K = \sum_{i=1}^{L} U_i$. Without loss of generality, we assume that $M_1 \leq M_2 \leq \cdots \leq M_K$ and $\mathcal{U}_1 = \{1, \ldots, U_1\}, \ldots, \mathcal{U}_L = \{K - U_L + 1, \ldots, K\}$. We also denote that fraction of the library stored at cache i by $m_i = M_i/N$, i.e., $m_i \in [0, 1]$ for $M_i \in [0, N]$.

Next, we explain the two operational phases of the system, namely, the placement phase and delivery phase.

A. Placement Phase

In the placement phase, the users' demands are unknown and the server populates the cache memories taking into account the cache sizes $\boldsymbol{m} \triangleq [m_1, \ldots, m_L]$ and the network topology, i.e., the number of users connected to each cache memory represented by $\boldsymbol{U} \triangleq [U_1, \ldots, U_L]$. More specifically, for given $\boldsymbol{U}$, and $\boldsymbol{m}$ the contents of cache i is defined as a function of the files

$$Z_i = \mu_i(W_1, \ldots, W_N; \boldsymbol{U}, \boldsymbol{m}), \tag{1}$$

which satisfies the cache size constraint $|Z_i| \leq Nm_iF$.

B. Delivery Phase

In the delivery phase, user k requests file W_{d_k} from the server. The delivery signal allows each cache to serve its associated users. The K users are served over $U_{\max}$ delivery rounds, such that we choose one of the users in $\mathcal{U}_i$, $\forall i$ that have not been served in each round, for example, for $L = 2$, $K = 3$, $\mathcal{U}_1 = \{1, 2\}$, and $\mathcal{U}_2 = \{3\}$, users 1 and 3 are served in the first round and user 2 in the second round. More specifically, in round r, the server sends a sequence of unicast/multicast signals, $X^{(r)}_{\mathcal{T},\boldsymbol{d}}$ to the caches in $\mathcal{T}$ in order to serve the users considered in this round.

At the end of the delivery phase, user k must be able to decode $\hat{W}_{d_k}$ reliably. Formally, for given cache sizes $\boldsymbol{m}$, number of files N, and network connectivity $\boldsymbol{U}$ with $U_{\max} \triangleq \max_i U_i$, the worst-case total delivery load $R(\boldsymbol{m}, N, \boldsymbol{U}) \triangleq \sum_{r=1}^{U_{\max}} \sum_{\mathcal{T}} |X^{(r)}_{\mathcal{T},\boldsymbol{d}}|/F$ is said to be achievable if for every $\epsilon > 0$ and large enough F, there exists a caching scheme such that $\max_{\boldsymbol{d},k\in[K]} Pr(\hat{W}_{d_k} \neq W_{d_k}) \leq \epsilon$.

Remark 1: A caching scheme designed for the shared-caches model in Fig. 1, can also be used in L-user systems where user k requests U_k files and the number of files requested by each user is known in advance. Caching systems where the users request multiple files have been investigated in [19], [31], where each user requests the same number of files.

In our achievability scheme, we utilize maximum distance separable (MDS) codes.

Definition 1 [32]: An (n, k) maximum distance separable (MDS) code is an erasure code that allows recovering k initial information symbols from any k out of the n coded symbols.

Furthermore, in a systematic (n, k)-MDS code the first k symbols in the output codeword is the information symbols. That is, we have

$$[i_1, \ldots, i_k]\boldsymbol{G}_{k\times n} = [i_1, \ldots, i_k]\big[\boldsymbol{I}_{k\times k}\ \boldsymbol{P}_{k\times n-k}\big] = \big[i_1, \ldots, i_k, c_{k+1}, \ldots, c_n\big], \quad (2)$$

where $\boldsymbol{G}_{k\times n}$ is the code generator matrix and $\boldsymbol{I}_{k\times k}$ is an identity matrix.

For an $(2N - j, N)$ MDS-code, we define

$$\sigma_j([i_1, \ldots, i_N]) \triangleq [i_1, \ldots, i_N]\boldsymbol{P}_{N\times N-j} \quad (3)$$

to denote the $N - j$ parity symbols in the output codeword. Note that $\sigma_j([i_1, \ldots, i_N])$ represents $N - j$ linearly independent equations in the information symbols $[i_1, \ldots, i_N]$. For example, $\sigma_1([i_1, \ldots, i_N]) = [i_1 \oplus i_2, i_2 \oplus i_3, \ldots, i_{N-1} \oplus i_N]$.

III. Systems With One User per Cache

In this section, we focus on systems where only one user is connected to each cache, i.e., $U_i = 1, \forall i$ and $K = L$. Note that R_{uncoded} represents the load with *uncoded placement and linear delivery*.

A. Results

Next, we present two theorems of achievable loads. The first theorem is specific to $K = 3$ and general to all memory regimes. The goal is to contrast to previous results showing the reduction in the delivery load when coded placement is utilized. The second one is general in K, but is specific the small memory regime defined therein.

Theorem 1: For a three-user system with $N \geq 4$ and $m_1 \leq m_2 \leq m_3$, the worst-case delivery load given by

$$\begin{aligned} &R_{\text{coded}}(\boldsymbol{m}, N) \\ &= \max\Big\{3 - 3m_1 - 2m_2 - m_3 - \frac{3(m_2 - m_1)}{N-1} - \frac{2(m_3 - m_2)}{N-2}, \\ &\quad \frac{5}{3} - \frac{3m_1 - 2m_2 - m_3}{3} - \frac{m_2 - m_1}{3(N-1)}, \\ &\quad 2 - 2m_1 - m_2 - \frac{m_2 - m_1}{N-1},\ 1 - m_1\Big\}, \end{aligned} \quad (4)$$

is achievable with coded placement.

Proof: The reduction in the delivery load in (4) compared to (5) is achieved by placing coded pieces of the files at users 2 and 3, which are decoded in the delivery phase. For example, in order to achieve $R_{\text{coded}}(\boldsymbol{m}, N) = R^*_{uncoded}(\boldsymbol{m}) - \frac{m_2 - m_1}{3(N-1)}$, part of the multicast signal to users $\{1, 2\}$ is utilized in decoding the cached pieces at user 3. The proposed caching scheme is detailed in the Appendix. ∎

Remark 2: The achievable delivery load in Theorem 1 is lower than the minimum worst-case delivery load under uncoded placement and linear delivery, given by

$$\begin{aligned} &R^*_{\text{uncoded}}(\boldsymbol{m}) \\ &= \max\Big\{3 - 3m_1 - 2m_2 - m_3, \frac{5}{3} - \frac{3m_1 - 2m_2 - m_3}{3}, \\ &\quad 2 - 2m_1 - m_2,\ 1 - m_1\Big\}, \end{aligned} \quad (5)$$

which has been characterized in [15].

Next theorem characterizes the gain achieved by coded placement in the small memory regime, where the unicast signals intended for users $\{1, \ldots, k\}$ are utilized in decoding the cache content at users $\{k + 1, .., K\}$.

Theorem 2: For a K-user system with $N \geq K + 1$, $m_1 \leq m_2 \leq \cdots \leq m_K$, and

$$\sum_{i=1}^{K} m_i + \sum_{i=2}^{K} \frac{(i-1)(K-i+1)(m_i - m_{i-1})}{N-i+1} \leq 1,$$

the worst-case delivery load

$$\begin{aligned} R_{\text{coded}}(\boldsymbol{m}, N, K) &= R^*_{uncoded}(\boldsymbol{m}, K) \\ &\quad - \sum_{i=2}^{K} \frac{(i-1)(K-i+1)(K-i+2)(m_i - m_{i-1})}{2(N-i+1)}, \end{aligned} \quad (6)$$

is achievable with coded placement, where

$$R^*_{\text{uncoded}}(\boldsymbol{m}, K) = K - \sum_{i=1}^{K}(K-i+1)m_i, \quad (7)$$

is the minimum worst-case delivery load with uncoded placement and linear delivery for $\sum_{i=1}^{K} m_i \leq 1$ [15].

Proof (Placement Phase): File W_n is divided into $K(K+1)/2 + 1$ subfiles, which we denote by $W_{n,\phi}, W^{(1)}_{n,1}$, $\{W^{(1)}_{n,2}, W^{(2)}_{n,2}\}, \ldots, \{W^{(1)}_{n,K}, \ldots, W^{(K)}_{n,K}\}$, such that

$$\big|W^{(1)}_{n,k}\big| = m_1 F,\ \forall n, \quad (8)$$

$$\big|W^{(i)}_{n,k}\big| = \frac{N(m_i - m_{i-1})}{(N-i+1)}F,\ i = 2, \ldots, K, \forall n, \quad (9)$$

$$\big|W_{n,\phi}\big| = \left(1 - \sum_{i=1}^{K} m_i - \sum_{i=2}^{K} \frac{(i-1)(K-i+1)(m_i - m_{i-1})}{N-i+1}\right)F, \quad (10)$$

where $W_{n,\phi}$ is available only at the server. User k caches subfiles $W^{(1)}_{1,k}, \ldots, W^{(1)}_{N,k}$ uncoded and the MDS encoded pieces $\sigma_{i-1}([W^{(i)}_{1,k}, \ldots, W^{(i)}_{N,k}])$ for $i = 2, \ldots, k$. In turn, the cache content of user k is defined as

$$Z_k = \sigma_0\Big(\Big[W^{(1)}_{1,k}, \ldots, W^{(1)}_{N,k}\Big]\Big) \bigcup \left(\bigcup_{i=2}^{k} \sigma_{i-1}\Big(\Big[W^{(i)}_{1,k}, \ldots, W^{(i)}_{N,k}\Big]\Big)\right), \quad (11)$$

where $\sigma_0([W^{(1)}_{1,k}, \ldots, W^{(1)}_{N,k}]) = \bigcup_{n=1}^{N} W^{(1)}_{n,k}$ denotes the uncoded cached subfiles. The placement is illustrated in Fig. 2 for $K = 3$.

Delivery Phase: The server sends the following unicast signals

$$X_{\{K\},\boldsymbol{d}} = W_{d_K,\phi}, \quad (12)$$

$$X_{\{k\},\boldsymbol{d}} = \bigcup_{i=k+1}^{K} \bigcup_{j=i}^{K} W^{(i)}_{d_k,j},\ \forall k \in [K-1], \quad (13)$$

where the unicast signals $X_{\{1\},\boldsymbol{d}}, \ldots, X_{\{k\},\boldsymbol{d}}$ are used by users $\{k + 1, \ldots, K\}$ in decoding their cache contents. Next, the server sends the pairwise multicast signals

$$X_{\{k,j\},\boldsymbol{d}} = \left(\bigcup_{i=1}^{k} W^{(i)}_{d_j,k}\right) \oplus \left(\bigcup_{i=1}^{k} W^{(i)}_{d_k,j}\right), \quad (14)$$

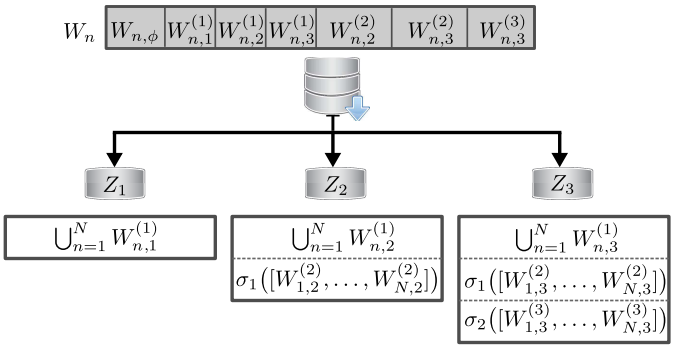

Fig. 2. Illustration of the placement scheme in Theorem 2 for a three-user system.

for $k = 1, \ldots, K$ and $j = k+1, \ldots, K$. In turn, the delivery load is given as

$$R_{\text{coded}} = \sum_{k=1}^{K} \left|X_{\{k\},\boldsymbol{d}}\right|/F + \sum_{k=1}^{K}\sum_{j=k+1}^{K} \left|X_{\{k,j\},\boldsymbol{d}}\right|/F, \tag{15}$$

$$= K - \sum_{i=1}^{K}(K-i+1)m_i - \sum_{i=2}^{K} \frac{(i-1)(K-i+1)(K-i+2)(m_i - m_{i-1})}{2(N-i+1)}. \tag{16}$$

Achievability: First, the cache size constraints are satisfied, since

$$|Z_k| = Nm_1F + \sum_{i=2}^{k}(N-i+1)\left|W_{n,k}^{(i)}\right|$$

$$= Nn_1F + \sum_{i=2}^{k} N(m_i - m_{i-1})F = Nm_kF. \tag{17}$$

In the delivery phase, user k reconstructs W_{d_k} by going through the following steps.

- Subfile $W_{d_k,k}^{(1)}$ is uncoded and in turn can be directly retrieved from the cache memory.
- Subfiles $W_{d_1,k}^{(i)}, \ldots, W_{d_{i-1},k}^{(i)}$ are extracted from the unicast signals $X_{\{1\},\boldsymbol{d}}, \ldots, X_{\{k-1\},\boldsymbol{d}}$.
- Subfile $W_{d_k,k}^{(i)}$ is retrieved from $\sigma_{i-1}([W_{1,k}^{(i)}, \ldots, W_{N,k}^{(i)}])$ using $W_{d_1,k}^{(i)}, \ldots, W_{d_{i-1},k}^{(i)}$.
- Subfiles $W_{d_k,j}^{(k+1)}, \ldots, W_{d_k,j}^{(j)}$ where $j \in \{k+1, \ldots, K\}$ and $W_{d_k,\phi}$ are retrieved from the unicast signal $X_{\{k\},\boldsymbol{d}}$.
- Subfiles $W_{d_k,j}^{(1)}, \ldots, W_{d_k,j}^{(k)}$ are retrieved from the multicast signals $X_{\{k,j\},\boldsymbol{d}}, j \in [K] \setminus \{k\}$. ■

Remark 3: For given K and $\boldsymbol{m}$, $\lim_{N\to\infty} R_{\text{coded}}(\boldsymbol{m}, N, K) = R^*_{\text{uncoded}}(\boldsymbol{m}, K)$. That is, the gain due to coded placement decreases with N and is negligible for $N \gg K$.

B. Numerical Results

In Fig. 3, we compare the worst-case delivery load achieved by exploiting coded placement with the minimum worst-case delivery load assuming uncoded placement in a three-user system where $N = 4$ and $m_k = \alpha\ m_{k+1}$. Fig. 3 shows that the gain achieved by coded placement increases with the heterogeneity in cache sizes. We also observe that the gain is higher when the total memory is small and the gain can be observed from the relative slopes of the two schemes.

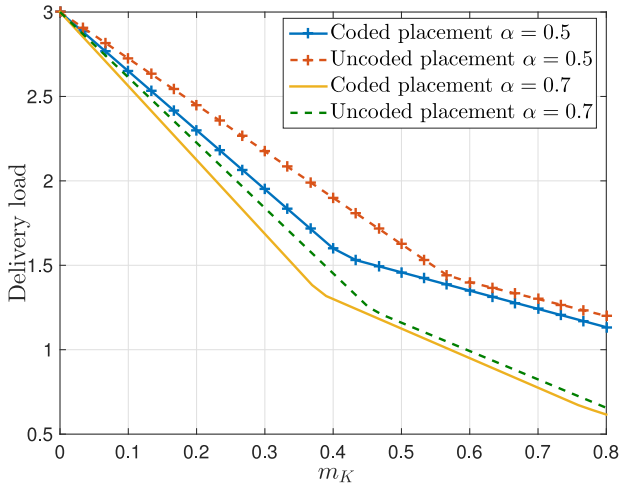

Fig. 3. Comparing the achievable delivery load assuming coded placement with the minimum delivery load under uncoded placement, for $K = 3$, $N = 4$, and $m_k = \alpha\ m_{k+1}$.

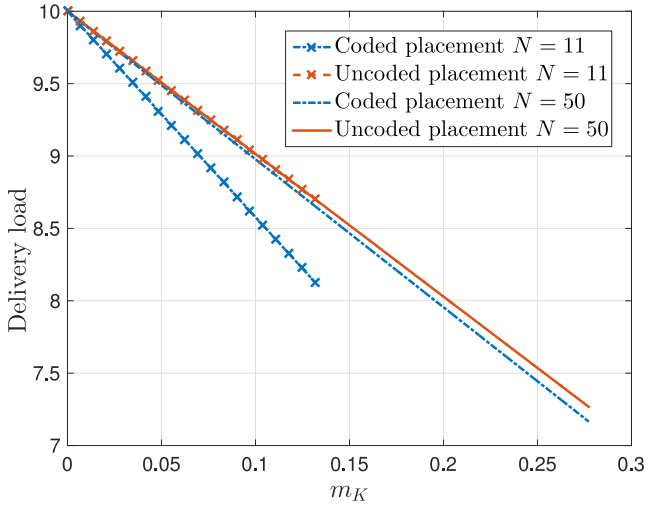

Fig. 4. Comparing the achievable delivery load assuming coded placement with the minimum delivery load under uncoded placement, for $K = 10$, $\alpha = 0.7$, and $m_k = \alpha\ m_{k+1}$.

The delivery load achieved by utilizing coded placement in Theorem 2 is compared to the best uncoded placement scheme in Fig. 4, for $K = 10$ and $m_k = 0.7m_{k+1}$. From Fig. 4, we observe that the reduction in the delivery load due to coded placement decreases with the number of files N. In turn, for a system where $N >> K$, the delivery load achieved with our coded placement scheme is approximately equal to the minimum delivery load under uncoded placement. That is, the coded placement gain is negligible when $N >> K$.

IV. Systems With Equal Caches

In this section, we consider systems with shared caches of equal size, i.e., $M_i = M$, $\forall i$ and without loss of generality we assume that $U_1 \geq U_2 \geq \cdots \geq U_L$. We first provide two concrete examples for two and three-cache systems

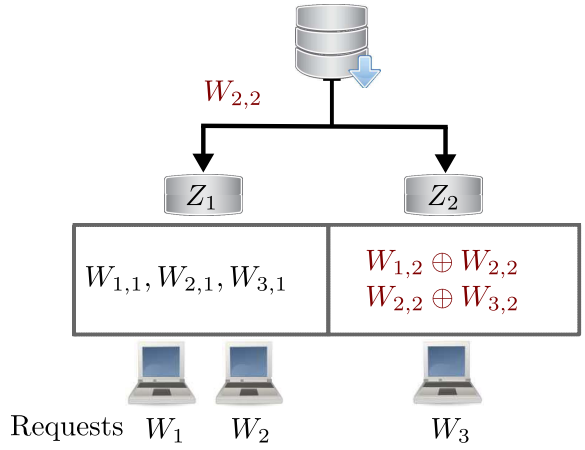

Fig. 5. The coded placement scheme for Example 1.

respectively to clarify the main ideas of the proposed coded placement and multi-round delivery, noting the relative simplicity of the scheme in the two-cache case. We then formalize the two-cache system moving on to the extension to $L > 2$. Finally, we provide the remarks for further improvement via placement through increased subpacketization for $L = 2$.

A. Example 1: Two-Cache System

Consider a system with $K = 3$, $N \geq 3$, $L = 2$ and $M \leq 1$, where $\mathcal{U}_1 = \{1, 2\}$ and $\mathcal{U}_2 = \{3\}$ as illustrated in Fig. 5.

1) The Uncoded Placement Scheme [26]: Each file is divided into 3 subfiles, $W_{n,1}$, $W_{n,2}$ and $W_{n,\phi}$, such that $|W_{n,1}| = |W_{n,2}| = \frac{M}{N}F$ bits and $|W_{n,\phi}| = (1 - 2\frac{M}{N})F$ bits [1]. The cached contents are given by

$$Z_k = \bigcup_n W_{n,k}, \tag{18}$$

Without loss of generality, we assume that user k requests file k. The delivery phase consists of two rounds. In round 1, the server sends the following signals to users 1 and 3:

$$W_{1,0}, W_{3,0}, W_{1,2} \oplus W_{3,1}. \tag{19}$$

In round 2, the unicast signals $W_{2,0}, W_{2,2}$ are sent to user 2. In turn, all the users recover their requested files using the cached contents. The total delivery load is given by

$$R_{\text{uncoded}} = 3\left(1 - \frac{2M}{N}\right) + \frac{2M}{N}, \tag{20}$$

where the first term represents the unicast transmission of the contents available only at the server, and the second term represents the multicast signal to users 1 and 3 in addition to the unicast signal to user 2.

2) The Proposed Coded Placement Scheme: We divide each file W_n into three pieces $W_{n,1}$ of size $\frac{MF}{N}$ bits, $W_{n,2}$ of size $\frac{MF}{N} + \frac{MF}{N(N-1)}$ bits, and $W_{n,\phi}$ of size $F - \frac{2MF}{N} - \frac{MF}{N-1}$ bits. The stored contents at the caches are given by

$$Z_1 = \bigcup_{n=1}^{3} W_{n,1}, \tag{21}$$

$$Z_2 = \left(W_{1,2} \oplus W_{2,2}\right) \bigcup \left(W_{2,2} \oplus W_{3,2}\right), \tag{22}$$

which is illustrated in Fig. 5. Assuming that user k requests file k, the server transmits the following signals over the two rounds.

$$W_{1,0}, W_{2,0}, W_{3,0}, W_{2,2}, W_{1,2} \oplus W_{3,1}. \tag{23}$$

Note that since $W_{1,2}$ is larger than $W_{3,1}$, we append zeros to the end of $W_{3,1}$ to equalize their lengths before XORing them. In order for the users to recover the requested files, first we need to decode Z_2. In particular, the unicast signal $W_{2,2}$ along with Z_2 enable cache 2 to recover $W_{1,2}, W_{2,2}, W_{3,2}$, which is illustrated in Fig. 5. In turn, we have

$$\begin{aligned} R_{\text{coded}} &= 3\left(1 - \frac{2M}{N} - \frac{M}{N-1}\right) + \frac{2M}{N(N-1)} + \frac{2M}{N} \\ &= R_{\text{uncoded}} - \frac{M}{N(N-1)}, \end{aligned} \tag{24}$$

where $\frac{M}{N(N-1)}$ is the gain from coded placement.

B. Example 2: Three-Cache System

Consider a system with $K = 6$, $N \geq 6$, $L = 3$ and $\frac{1}{3} \leq M/N \leq \frac{2}{3}$, where users 1, 2 and 3 are connected to cache 1, users 4 and 5 are connected to cache 2 and user 6 is connected to cache 3, i.e., $\mathcal{U}_1 = \{1, 2, 3\}$, $\mathcal{U}_2 = \{4, 5\}$ and $\mathcal{U}_3 = \{6\}$.

1) The Uncoded Placement Scheme [26]: Each file is divided into 6 subfiles, $W_{n,1}$, $W_{n,2}$, $W_{n,3}$ $W_{n,12}$, $W_{n,13}$, and $W_{n,23}$, such that $|W_{n,1}| = |W_{n,2}| = |W_{n,3}| = (\frac{2}{3} - \frac{M}{N})F$ bits and $|W_{n,12}| = |W_{n,13}| = |W_{n,23}| = (\frac{M}{N} - \frac{1}{3})F$ bits [1]. The cached contents are given by

$$Z_1 = \bigcup_n \left(W_{n,1} \bigcup W_{n,12} \bigcup W_{n,13}\right), \tag{25}$$

$$Z_2 = \bigcup_n \left(W_{n,2} \bigcup W_{n,12} \bigcup W_{n,23}\right), \tag{26}$$

$$Z_3 = \bigcup_n \left(W_{n,3} \bigcup W_{n,13} \bigcup W_{n,23}\right). \tag{27}$$

Suppose that user k requests file k during the delivery phase. In this example, we have three delivery rounds. In round 1, users $\{1, 4, 6\}$ are served by sending the signals

$$\begin{aligned} &W_{1,2} \oplus W_{4,1}, \ W_{1,3} \oplus W_{6,1}, \ W_{4,3} \oplus W_{6,2}, \\ &W_{1,23} \oplus W_{4,13} \oplus W_{6,12}. \end{aligned} \tag{28}$$

In round 2, users $\{2, 5\}$ are served by sending the signals

$$W_{2,2} \oplus W_{5,1}, W_{2,3}, W_{5,3}, W_{2,23} \oplus W_{5,13}. \tag{29}$$

In round 3, user 3 receives the following signals

$$W_{3,23}, W_{3,2}, W_{3,3}. \tag{30}$$

With the help of the cached contents, all the users recover their requested files. The delivery load is given by

$$R_{\text{uncoded}}F = 3|W_{n,23}| + 3|W_{n,2}| + 5|W_{n,3}| = \left(\frac{13}{3} - \frac{5M}{N}\right)F. \tag{31}$$

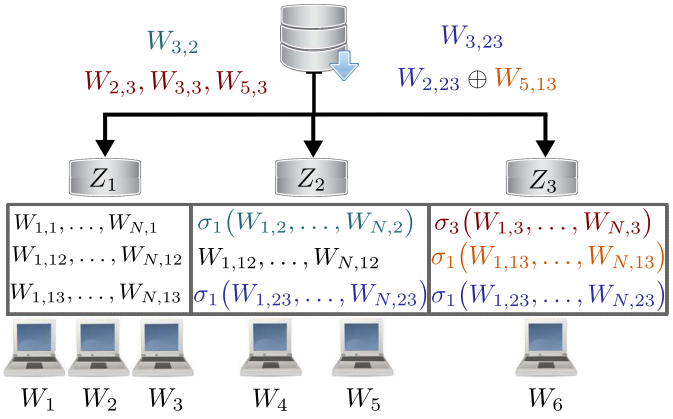

Fig. 6. The coded placement scheme for Example 2.

2) The Proposed Coded Placement Scheme: Similarly, each file is divided into 6 subfiles, $W_{n,1}$, $W_{n,2}$, $W_{n,3}$ $W_{n,12}$, $W_{n,13}$, and $W_{n,23}$, such that $|W_{n,\mathcal{S}}| = a_{\mathcal{S}}F$ bits that will be specified later. The stored contents at the caches are given by

$$Z_1 = \bigcup_n \Big(W_{n,1} \bigcup W_{n,12} \bigcup W_{n,13} \Big), \tag{32}$$

$$Z_2 = \sigma_1([W_{1,2}, \ldots, W_{N,2}]) \bigcup \sigma_1([W_{1,23}, \ldots, W_{N,23}]) \bigcup \Big(\bigcup_n W_{n,12} \Big), \tag{33}$$

$$Z_3 = \sigma_3([W_{1,3}, \ldots, W_{N,3}]) \bigcup \sigma_1([W_{1,13}, \ldots, W_{N,13}]) \bigcup \sigma_1([W_{1,23}, \ldots, W_{N,23}]), \tag{34}$$

i.e., we store $N-1$ independent equations of $W_{1,2}, \ldots, W_{N,2}$ and $N-1$ independent equations of $W_{1,23}, \ldots, W_{N,23}$ at cache 2. At cache 3, we store $N-3$ independent equations of $W_{1,3}, \ldots, W_{N,3}$, $N-1$ independent equations of $W_{1,13}, \ldots, W_{N,13}$, and $N-1$ independent equations of $W_{1,23}, \ldots, W_{N,23}$.

In the delivery phase, user k requests file k and the server constructs the signals defined in (28)-(30); again, if the subfiles forming a signal differ in size, then the server appends zeros to equalize their length before XORing them. In order to decode the subfiles stored at caches 2 and 3, we utilize the signals $W_{2,23} \oplus W_{5,13}$, $W_{2,3}$, $W_{5,3}$, $W_{3,3}$, $W_{3,2}$, $W_{3,23}$. For instance, the multicast signal $W_{2,23} \oplus W_{5,13}$ can be used in decoding $2N-1$ equations in $W_{n,23}$ and $W_{n,13}$. In our scheme, we assume that the subfiles are decoded successively at the caches. In particular, first we decode $W_{n,23}$, then the multicast signal can be used in decoding $W_{n,13}$. Fig. 6 depicts the scheme.

In order to minimize the total delivery load, we optimize over the subfile sizes, as follows

$$\min_{a_{\mathcal{S}} \geq 0} \quad R_{\text{coded}} = 3a_2 + 5a_3 + 3a_{23} \tag{35a}$$

$$\text{subject to} \quad \sum_{\mathcal{S} \subset [L]} a_{\mathcal{S}} = 1, \tag{35b}$$

$$N(a_1 + a_{12} + a_{13}) \leq M, \tag{35c}$$

$$(N-1)(a_2 + a_{13}) + Na_{12} \leq M, \tag{35d}$$

$$(N-3)a_3 + (N-1)(a_{13} + a_{23}) \leq M, \tag{35e}$$

$$a_1 \leq a_2 \leq a_3, \;\; a_{12} \leq a_{13} \leq a_{23}. \tag{35f}$$

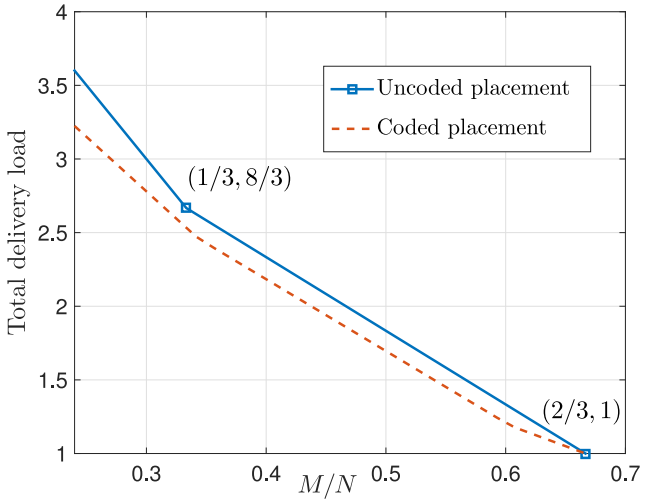

Fig. 7. The normalized delivery load for $L = 3$, $N = K = 6$, $U_1 = 3$, $U_2 = 2$ and $U_3 = 1$.

(35b) ensures the feasibility of the file partitioning. Assuming that $a_{123} = 0$, conditions (35c)-(35e) ensure that the memory capacity constraints are satisfied. We also assume $a_1 \leq a_2 \leq a_3$ and $a_{12} \leq a_{13} \leq a_{23}$, since $U_1 > U_2 > U_3$. In Fig. 7, we show that the proposed scheme achieves a lower total delivery load compared with the uncoded placement scheme in [26].

C. Two-Cache System

In this section, we provide the proposed scheme for systems with two caches, i.e., $L = 2$. Without loss of generality, assume that the first U_1 users are connected to cache 1 and users $\{U_1 + 1, \ldots, K\}$ are connected to cache 2. Let $q = U_1 - U_2$.

1) Placement Phase: Divide each file into the subfiles: $W_{n,\phi}$, $W_{n,1}$, $W_{n,2}$ and $W_{n,12}$. The cache contents are given by

$$Z_1 = \bigcup_n \Big(W_{n,1} \bigcup W_{n,12} \Big), \tag{36}$$

$$Z_2 = \sigma_q([W_{1,2}, \ldots, W_{N,2}]) \bigcup \Big(\bigcup_n W_{n,12} \Big). \tag{37}$$

2) Delivery Phase: Next, we describe the caching scheme in three memory regimes.

Region ($\frac{M}{N} + \frac{M}{N-q} \leq 1$): In this case, we choose $|W_{n,12}| = 0$, $|W_{n,1}| = \frac{M}{N}F$, $|W_{n,2}| = \frac{M}{N-q}F$, and $|W_{n,\phi}| = (1 - \frac{M}{N} - \frac{M}{N-q})F$. During the delivery phase, first we send U_2 multicast signal in the form of $W_{d_x,2} \oplus W_{d_y,1}$ each of which is intended to a pair of users from the set $\{(1, U_1+1), (2, U_1+2), \ldots, (U_2, K)\}$. Second, we send q unicast signals in the form of $W_{d_x,2}$ to users $x \in \{U_2+1, \ldots, U_1\}$. Additionally, the q unicast signals facilitate decoding the coded cached contents in Z_2, i.e., cache 2 is able to retrieve $W_{n,2}$, $\forall n$. Finally, the server unicast the subfiles $\{W_{d_k,\phi} : \forall k\}$, which are not cached in the network. By the end of the delivery phase, each user is able to reconstruct its requested file. The total delivery load is given by

$$R_{\text{coded}} = K\left(1 - \frac{M}{N} - \frac{M}{N-q}\right) + \frac{qM}{N-q} + \frac{U_2 M}{N-q} = R_{\text{uncoded}} - \frac{qU_2 M}{N(N-q)}. \tag{38}$$

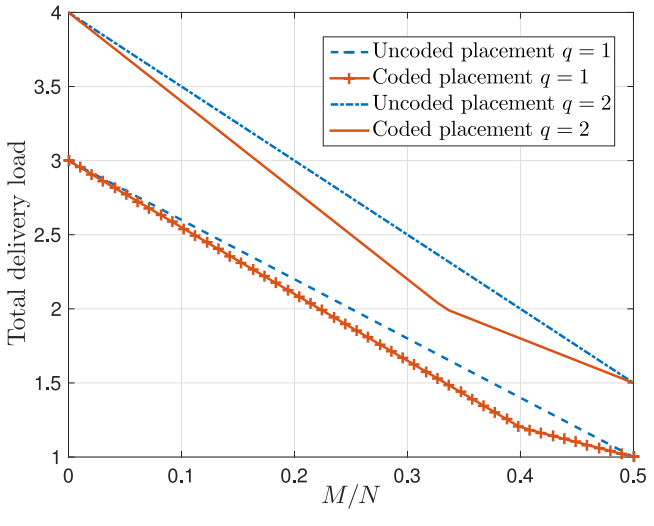

Fig. 8. The total delivery load for $L=2$, $N=K=q+2$, $U_1=q+1$ and $U_2=1$.

Remark 4: The last term in (38) represents the gain of coded placement. We observe that the gain from coded placement increases with q, which is the difference between the number of users connected to each of the two caches. In other words, as the asymmetry in the system increases the gain from the coded placement increases as well.

Region ($\frac{N-q}{2N-q} \leq \frac{M}{N} < 0.5$): We choose $|W_{n,\phi}| = |W_{n,12}| = 0$, $|W_{n,1}| = \frac{M}{N}F$, and $|W_{n,2}| = (1-\frac{M}{N})F$. The delivery procedure is similar to the first case, and the delivery load is given by

$$R_{\text{coded}} = U_1\left(1-\frac{M}{N}\right). \tag{39}$$

Region ($\frac{M}{N} \geq 0.5$): No gain from coded placement and the total delivery load is given by

$$R_{\text{coded}} = R_{\text{uncoded}} = U_1\left(1-\frac{M}{N}\right). \tag{40}$$

Remark 5: For $\frac{M}{N} \geq \frac{N-q}{2N-q}$, the proposed scheme is optimal with respect to cut-set bound.

In Fig. 8, we show the achievable delivery load for two-caches system. It is clear that the performance gap between the proposed scheme and the uncoded placement scheme [26] increases with q, as explained in Remark 4.

D. L-Cache System

In this section, we present our caching scheme for a general L-cache system.

1) Placement Phase: Each file W_n is divided into subfiles $W_{n,\mathcal{S}}, \mathcal{S} \subset [L]$, where $W_{n,\mathcal{S}}$ is stored (coded or uncoded) exclusively at the caches in $\mathcal{S}$ and $|W_{n,\mathcal{S}}| = a_{\mathcal{S}}F, \forall n$. In general, we assume that cache $s \in \mathcal{S}$ stores $(N-\lambda_{\mathcal{S}}^{(s)})$ independent equations in subfiles $W_{1,\mathcal{S}}, \ldots, W_{N,\mathcal{S}}$, i.e., Cache s is defined as

$$Z_s = \bigcup_{\mathcal{S}\subset[L]:s\in\mathcal{S}} \sigma_{\lambda_{\mathcal{S}}^{(s)}}([W_{1,\mathcal{S}}, \ldots, W_{N,\mathcal{S}}]), \tag{41}$$

where $\sigma_0(.)$ represents uncoded placement. In order to determine the coded placement parameters $\{\lambda_{\mathcal{S}}^{(s)}\}$, we need to analyze the unicast/multicast signals in the delivery procedure in [26] and characterize the signals that can be utilized in decoding the cached contents.

2) Delivery Phase: Our delivery scheme is based on the delivery procedure in [26], where the delivery rounds are grouped as follows:

1) Rounds (1 to U_L): In each round, we serve L out of the remaining users connected to the caches $[L]$.
2) Rounds (U_L+1 to U_{L-1}): In each round, we serve $L-1$ out of the remaining users connected to the caches $[L-1]$.

$\vdots$

l) Rounds ($U_{L-l+2}+1$ to U_{L-l+1}): In each round, we serve $L-l+1$ out of the remaining users connected to the caches $[L-l+1]$.

$\vdots$

L) Rounds (U_2+1 to U_1): In each round, we serve one out of the remaining users connected to cache 1.

Different from [26], the XORed subfiles in a multicast signal can have different size. In the lth group of rounds, a multicast signal serving the users connected to the caches in $\mathcal{T} \subset [L]$, where $\mathcal{T} \cap [L-l+1] \neq \phi$, is defined as

$$\tilde{X}_{\mathcal{T},l} = \oplus_{k\in\mathcal{T}\cap[L-l+1]} W_{d_k,\mathcal{T}\setminus\{k\}}, \tag{42}$$

where d_k is the file requested by the user connected to cache k and $|\tilde{X}_{\mathcal{T},l}| = \max_{k\in\mathcal{T}\cap[L-l+1]} a_{\mathcal{T}\setminus\{k\}}F$. In turn, the total delivery load is defined as

$$R = \sum_{l=1}^{L}(U_{L-l+1}-U_{L-l+2}) \sum_{\mathcal{T}\subset[L]\,:\,\mathcal{T}\cap[L-l+1]\neq\phi} \left|\tilde{X}_{\mathcal{T},l}\right|/F, \tag{43}$$

since the $(U_{L-l+1}-U_{L-l+2})$ delivery rounds in the lth group of rounds have same delivery load.

Remark 6: If we have $|\mathcal{T}\cap[L-l+1]| < |\mathcal{T}|$, the multicast signal defined in (42) can be utilized in decoding the caches in $\bigcap_{k\in\mathcal{T}\cap[L-l+1]} \mathcal{T}\setminus\{k\}$, e.g., for $L=3$, $\mathcal{T}=\{1,2,3\}$ and $l=2$, $W_{d_1,\{2,3\}} \oplus W_{d_2,\{1,3\}}$ can be utilized at cache 3 in decoding $2N-1$ equations in $\{W_{n,\{2,3\}}\}_{n=1}^{N}$ and $\{W_{n,\{1,3\}}\}_{n=1}^{N}$.

Coded placement parameters $\{\lambda_{\mathcal{S}}^{(s)}\}$ represent the overall coded placement gain facilitated by all the signals satisfying the condition $|\mathcal{T}\cap[L-l+1]| < |\mathcal{T}|$. Given $U_1 \geq U_2 \geq \cdots \geq U_L$, we assume that the subfile sizes satisfy[2]

$$\begin{aligned} a_{\{s_1,\ldots,s_{t-1},s_t\}} &\leq a_{\{s_1,\ldots,s_{t-1},s_t+1\}}, \\ a_{\{s_1,\ldots,s_{t-1},L\}} &\leq a_{\{s_1,\ldots,s_{t-1}+1,s_{t-1}+2\}}, \end{aligned} \tag{44}$$

for $\mathcal{S} \subset [L]$ where $\mathcal{S}=\{s_1,\ldots,s_t\}$ and $s_i < s_{i+1} \forall i$. That is, $s_1 \in \{1,\ldots,L-t+1\}$ and $s_i \in \{s_{i-1}+1,\ldots,L-t+i\}$ for $i>1$. In turn, we have

$$\left|\tilde{X}_{\mathcal{T},l}\right| = a_{\mathcal{T}\setminus\{k\}}F, \;\; k = \arg\min_{i\in\mathcal{T}\cap[L-l+1]} i, \tag{45}$$

[2]This assumption is intuitive: The more users share the caches the smaller the shares.

and the total normalized delivery load can be expressed as

$$R = \sum_{t=0}^{L-1} \sum_{\mathcal{S}\subset[L]:|\mathcal{S}|=t} \mu_{\mathcal{S}} a_{\mathcal{S}}, \tag{46}$$

where $\mu_{\mathcal{S}}$ is defined as

$$\begin{aligned}\mu_{\mathcal{S}} &= \sum_{l=1}^{L-s_1+1} (s_1-1)(U_{L-l+1}-U_{L-l+2}) \\ &\quad + \sum_{l=L-s_1+2}^{L} (L-l+1)(U_{L-l+1}-U_{L-l+2}) \\ &= \sum_{l=1}^{s_1-1} U_l. \end{aligned} \tag{47}$$

The coded subfiles are decoded successively at the caches starting with the subfile with the largest size. That is, the multicast signal defined in (42) facilitates the decoding of $\{W_{n,\mathcal{T}\setminus\{k\}}\}_{n=1}^{N}$, where $k = \arg\max_{i\in\mathcal{T}\cap[L-l+1]} i$, e.g., for $L = 3$, $\mathcal{T} = \{1,2,3\}$ and $l = 2$, $W_{d_1,\{2,3\}} \oplus W_{d_2,\{1,3\}}$ is used in decoding $\{W_{n,\{1,3\}}\}_{n=1}^{N}$ at cache 3, since $\{W_{n,\{2,3\}}\}_{n=1}^{N}$ are decoded first. Based on the aforementioned decoding order, the parameters $\{\lambda_{\mathcal{S}}^{(s)}\}$ are defined as follows

$$\lambda_{\mathcal{S}}^{(s_i)} = \lambda_{\mathcal{S}}^{(s_{i-1})} + \sum_{l=L-s_i+2}^{L-s_{i-1}} (L-l-s_{i-1}+1)(U_{L-l+1}-U_{L-l+2}), \tag{48}$$

where $s_0 = 0$, $\lambda_{\mathcal{S}}^{(0)} = 0$, and $U_{L+1} = 0$.

Finally, the total delivery load is minimized by optimizing over the subfile sizes.

$$\min_{a_{\mathcal{S}}\geq 0} \quad \sum_{t=0}^{L-1} \sum_{\mathcal{S}\subset[L]:|\mathcal{S}|=t} \mu_{\mathcal{S}} a_{\mathcal{S}} \tag{49a}$$

$$\text{subject to} \quad \sum_{\mathcal{S}\subset[L]} a_{\mathcal{S}} = 1, \tag{49b}$$

$$\sum_{\mathcal{S}\subset[L]:l\in\mathcal{S}} \left(N - \lambda_{\mathcal{S}}^{(s)}\right) a_{\mathcal{S}} \leq M, \forall l \in [L] \tag{49c}$$

$$\begin{aligned} &a_{\{s_1,\ldots,s_{t-1},s_t\}} \leq a_{\{s_1,\ldots,s_{t-1},s_t+1\}}, \\ &a_{\{s_1,\ldots,s_{t-1},L\}} \leq a_{\{s_1,\ldots,s_{t-1}+1,s_{t-1}+2\}}, \\ &\forall \{s_1,\ldots,s_t\} \subset [L] \text{ with } s_i < s_{i+1}. \end{aligned} \tag{49d}$$

Equation (49b) and (49c) above represent all feasible choices for the subfile sizes which satisfy the cache size constraints.

In Fig. 9, we compare the achievable delivery loads with uncoded and coded placement for $L = 4$, $N = 15$, and $\boldsymbol{U} = [8,\ 4,\ 2,\ 1]$, and observe the performance improvement due to coded placement.

E. Remark: Gains With Coded Placement With Increased Subpacketization

The caching schemes presented so far have the same subpacketization level as the uncoded placement scheme in [26], i.e., the number of subfiles in the placement phase is the same. Next, we show that a higher coded placement gain is achievable at the expense of increasing the number of subfiles in the placement phase.

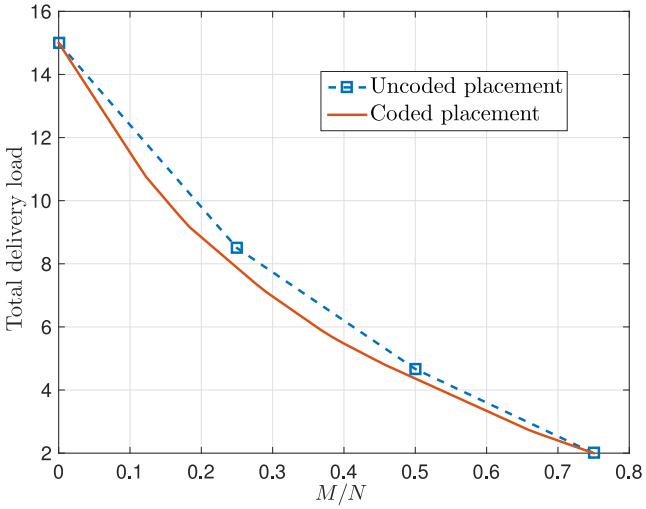

Fig. 9. The achievable delivery load for $L = 4$ and $N = 15$.

In Section IV-C, we presented a caching scheme for a two-cache system with $U_1 \geq U_2$, where the multicast signal $W_{d_x,\{2\}} \oplus W_{d_y,\{1\}}$ is formed by zero-padding $W_{d_y,\{1\}}$ to have the same length as $W_{d_x,\{2\}}$ before XORing them, since $|W_{d_y,\{1\}}| \leq |W_{d_x,\{2\}}|$. Alternatively, if $W_{n,\{2\}}$ is split into $W_{n,\{2\}}^{(1)}, W_{n,\{2\}}^{(2)}$ such that $|W_{n,\{1\}}| = |W_{n,\{2\}}^{(1)}|$, the multicast signals $W_{d_x,\{2\}} \oplus W_{d_y,\{1\}}$ can be decomposed into multicast signals $W_{d_x,\{2\}}^{(1)} \oplus W_{d_y,\{1\}}$ and unicast signals $W_{d_x,\{2\}}^{(2)}$. The unicast signals $W_{d_x,\{2\}}^{(2)}$ can be utilized in increasing the coded placement gain. In particular, at cache 2, we store $\sigma_{U_1-U_2}([W_{1,\{2\}}^{(1)},\ldots,W_{N,\{2\}}^{(1)}])$ and $\sigma_{U_1}([W_{1,\{2\}}^{(2)},\ldots,W_{N,\{2\}}^{(2)}])$, instead of $\sigma_{U_1-U_2}([W_{1,\{2\}},\ldots,W_{N,\{2\}}])$, and the U_1 unicast signals $W_{d_x,\{2\}}^{(2)}$ are used to decode $\sigma_{U_1}([W_{1,\{2\}}^{(2)},\ldots,W_{N,\{2\}}^{(2)}])$. The details of the caching scheme is explained in Section V-A for the general case where the caches are of different size. For equal caches, the improved delivery load is given as

- For $M/N \leq \frac{N-U_1}{2N-U_1-U_2}$, we have

$$\tilde{R}_{\text{coded}} = R_{\text{uncoded}} - \frac{U_2(U_1-U_2)M}{(N-U_1)N}. \tag{50}$$

  That is, the coded placement gain is increased by the multiplicative factor $\frac{(N-U_1+U_2)}{(N-U_1)}$.
- For $\frac{N-U_1}{2N-U_1-U_2} < M/N$, we have

$$\tilde{R}_{\text{coded}} = U_1\left(1-\frac{M}{N}\right). \tag{51}$$

In Fig. 10, we compare the total delivery load achievable with uncoded placement R_{uncoded}, the delivery load R_{coded} achievable with the coded placement scheme presented in Section IV-C, and the delivery load $\tilde{R}_{\text{coded}}$ achievable with the improved coded placement scheme, for a two-cache system with $N = K = 11$, $U_1 = 8$ and $U_2 = 3$.

V. Unified Coded Placement Scheme for Asymmetric Systems

In this section, we consider a system with both unequal caches and asymmetric user-to-cache connectivity. In particular, we consider a two-cache system with $M_1 \leq M_2$ and

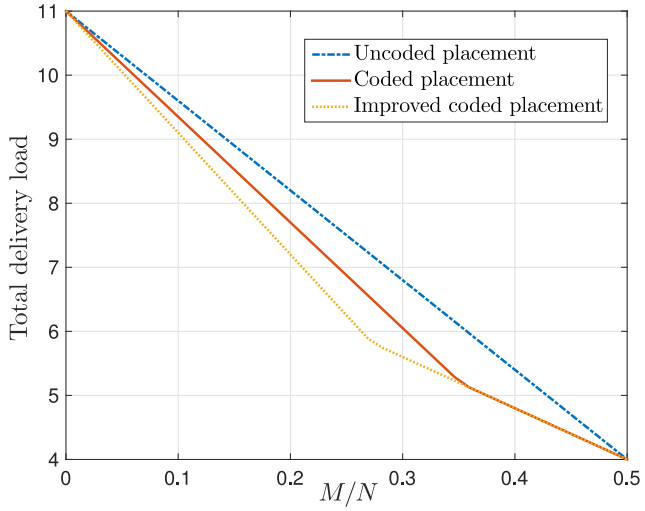

Fig. 10. The total delivery load for $L = 2$, $N = K = 11$, $U_1 = 8$ and $U_2 = 3$.

investigate the gain achievable with coded placement when $U_1 \geq U_2$ and $U_1 < U_2$.

A. Smaller Cache is Overloaded ($U_1 \geq U_2$)

In the placement phase, each file is divided into subfiles: $W_{n,\phi}$, $W_{n,\{1\}}$, $W^{(1)}_{n,\{2\}}$, $W^{(2)}_{n,\{2\}}$, and $W_{n,\{1,2\}}$. The cache contents are defined as

$$Z_1 = \bigcup_n \Big(W_{n,\{1\}} \bigcup W_{n,\{1,2\}} \Big), \tag{52}$$

$$Z_2 = \sigma_{U_1-U_2}\Big(\Big[W^{(1)}_{1,\{2\}}, \ldots, W^{(1)}_{N,\{2\}}\Big]\Big) \bigcup \sigma_{U_1}\Big(\Big[W^{(2)}_{1,\{2\}}, \ldots, W^{(2)}_{N,\{2\}}\Big]\Big) \bigcup \Big(\bigcup_n W_{n,\{1,2\}} \Big), \tag{53}$$

where $|W_{n,\{1\}}| = |W^{(1)}_{n,\{2\}}|$.

In the delivery phase, the server sends $U_1 + U_2$ unicast signals $W_{d_k,\phi}$, U_2 multicast signals $W^{(1)}_{d_x,\{2\}} \oplus W_{d_y,\{1\}}$, $U_1 - U_2$ unicast signals $W^{(1)}_{d_x,\{2\}}$, and U_1 unicast signals $W^{(2)}_{d_x,\{2\}}$.

More specifically, we have the following three memory regions:

- Region ($\frac{(N-U_1-U_2)m_1}{N-U_1} + \frac{Nm_2}{N-U_1} \leq 1$): We have $|W_{n,\{1,2\}}| = 0$ and

$$\Big|W^{(2)}_{n,\{2\}}\Big| = \frac{Nm_2 - (N - U_1 + U_2)m_1}{N - U_1} F, \tag{54}$$

$$\big|W_{n,\{1\}}\big| = m_1 F. \tag{55}$$

  In turn, the total delivery load is given by

$$R_{\text{coded}} = R_{\text{uncoded}} - \frac{U_2(U_1 m_2 - U_2 m_1)}{N - U_1}, \tag{56}$$

  where $R_{\text{uncoded}} = (U_1 + U_2) - (U_1 + U_2)m_1 - U_2 m_2$.
- Region ($\frac{(N-U_1-U_2)m_1}{N-U_1} + \frac{Nm_2}{N-U_1} > 1$) and ($m_1 + m_2 < 1$): We have $|W_{n,\phi}| = |W_{n,\{1,2\}}| = 0$, and

$$\Big|W^{(2)}_{n,\{2\}}\Big| = (1 - 2m_1)F, \tag{57}$$

$$\big|W_{n,\{1\}}\big| = m_1 F, \tag{58}$$

  In turn, the total delivery load is given by

$$R_{\text{coded}} = U_1(1 - m_1) = R_{\text{uncoded}} - U_2(1 - m_1 - m_2). \tag{59}$$

- Region ($m_1 + m_2 \geq 1$): No gain from coded placement and the delivery load is given by

$$R_{\text{coded}} = R_{\text{uncoded}} = U_1(1 - m_1). \tag{60}$$

B. Larger Cache is Overloaded ($U_1 < U_2$)

Next, we present two different coded placement schemes depending on whether we have $Nm_1 \leq (N - U_2 + U_1)m_2$ or $Nm_1 > (N - U_2 + U_1)m_2$.

1) Case ($Nm_1 \leq (N-U_2+U_1)m_2$): In the placement phase, each file is divided into subfiles: $W_{n,\phi}$, $W_{n,\{1\}}$, $W^{(1)}_{n,\{2\}}$, $W^{(2)}_{n,\{2\}}$, and $W_{n,\{1,2\}}$. The cache contents are defined as

$$Z_1 = \sigma_{U_2-U_1}\big([W_{1,\{1\}}, \ldots, W_{N,\{1\}}]\big) \bigcup \Big(\bigcup_n W_{n,\{1,2\}} \Big), \tag{61}$$

$$Z_2 = \bigcup_n \Big(W^{(1)}_{n,\{2\}} \bigcup W_{n,\{1,2\}} \Big) \bigcup \sigma_{U_1}\Big(\Big[W^{(2)}_{1,\{2\}}, \ldots, W^{(2)}_{N,\{2\}}\Big]\Big), \tag{62}$$

where $|W_{n,\{1\}}| = |W^{(1)}_{n,\{2\}}|$.

In the delivery phase, the server sends $U_1 + U_2$ unicast signals $W_{d_k,\phi}$, U_1 multicast signals $W^{(1)}_{d_x,\{2\}} \oplus W_{d_y,\{1\}}$, $U_2 - U_1$ unicast signals $W_{d_y,\{1\}}$, and U_1 unicast signals $W^{(2)}_{d_x,\{2\}}$. Therefore, the delivery load is given as $R_{\text{coded}}F = (U_1 + U_2)|W_{n,\phi}| + U_2|W_{n,\{1\}}| + U_1|W^{(2)}_{n,\{2\}}|$.

More specifically, we have the following three memory regions:

- Region ($\frac{N(N-2U_1)m_1}{(N-U_1)(N-U_2+U_1)} + \frac{Nm_2}{N-U_1} \leq 1$): We have $|W_{n,\{1,2\}}| = 0$ and

$$\big|W_{n,\{1\}}\big| = \Big|W^{(1)}_{n,\{2\}}\Big| = \frac{Nm_1}{N - U_2 + U_1} F, \tag{63}$$

$$\Big|W^{(2)}_{n,\{2\}}\Big| = \left(\frac{N}{N - U_1}\right)\left(m_2 - \frac{N}{N - U_2 + U_1} m_1\right) F. \tag{64}$$

  In turn, the delivery load is given by

$$R_{\text{coded}} = (U_1 + U_2) - \left(\frac{N}{N - U_1}\right)\left(\frac{U_1(2N - 2U_1 - U_2)}{N - U_2 + U_1} m_1 + U_2 m_2\right), \tag{65}$$

  compared to $R_{\text{uncoded}} = (U_1 + U_2) - 2U_1 m_1 - U_2 m_2$.
- Region ($\frac{N(N-2U_1)m_1}{(N-U_1)(N-U_2+U_1)} + \frac{Nm_2}{N-U_1} > 1$) and ($m_1 + \frac{N}{N-U_1}(m_2 - m_1) < 1$): The subfile sizes are obtained by solving

$$2\big|W_{n,\{1\}}\big| + \Big|W^{(2)}_{n,\{2\}}\Big| + \big|W_{n,\{1,2\}}\big| = F, \tag{66}$$

$$(N - U_2 + U_1)\big|W_{n,\{1\}}\big| + N\big|W_{n,\{1,2\}}\big| = Nm_1 F, \tag{67}$$

$$N\big|W_{n,\{1\}}\big| + (N - U_1)\Big|W^{(2)}_{n,\{2\}}\Big| + N\big|W_{n,\{1,2\}}\big| = Nm_2 F. \tag{68}$$

  and the corresponding delivery load is given by

$$R_{\text{coded}} = \frac{U_1 N(N-U_2)}{N(N-U_1)-U_1(U_2-U_1)}(1-m_1) + \frac{(U_2-U_1)N(N-U_1)}{N(N-U_1)-U_1(U_2-U_1)}(1-m_2) \quad (69)$$

- Region $(m_1 + \frac{N}{N-U_1}(m_2-m_1) \geq 1)$: We have $|W^{(2)}_{n,\{2\}}| = (1-m_1)F$, $|W_{n,\{1,2\}}| = m_1 F$, and the delivery load is given by

$$R_{\text{coded}} = U_1(1-m_1), \quad (70)$$

compared to $R_{\text{uncoded}} = U_1(1-m_1)+(U_2-U_1)(1-m_2)$.

2) Case ($Nm_1 > (N-U_2+U_1)m_2$): In the placement phase, each file is divided into subfiles: $W_{n,\phi}$, $W^{(1)}_{n,\{1\}}$, $W^{(2)}_{n,\{1\}}$, $W_{n,\{2\}}$, and $W_{n,\{1,2\}}$. The cache contents are defined as

$$Z_1 = \sigma_{U_2-U_1}\left(\left[W^{(1)}_{1,\{1\}},\ldots,W^{(1)}_{N,\{1\}}\right]\right) \bigcup \sigma_{U_2}\left(\left[W^{(2)}_{1,\{1\}},\ldots,W^{(2)}_{N,\{1\}}\right]\right) \bigcup \left(\bigcup_n W_{n,\{1,2\}}\right), \quad (71)$$

$$Z_2 = \bigcup_n \left(W_{n,\{2\}} \bigcup W_{n,\{1,2\}}\right), \quad (72)$$

where $|W^{(1)}_{n,\{1\}}| = |W_{n,\{2\}}|$.

In the delivery phase, the server sends $U_1 + U_2$ unicast signals $W_{d_k,\phi}$, U_1 multicast signals $W_{d_x,\{2\}} \oplus W^{(1)}_{d_y,\{1\}}$, U_2-U_1 unicast signals $W^{(1)}_{d_y,\{1\}}$, and U_2 unicast signals $W^{(2)}_{d_x,\{1\}}$. Therefore, the delivery load is given as $R_{\text{coded}}F = (U_1 + U_2)|W_{n,\phi}| + U_2|W_{n,\{2\}}| + U_2|W^{(2)}_{n,\{1\}}|$.

More specifically, we have the following three memory regions:

- Region $(\frac{Nm_1}{N-U_2} + \frac{(N-U_2-U_1)m_2}{N-U_2} \leq 1)$: We have $|W_{n,\{1,2\}}| = 0$ and

$$\left|W^{(1)}_{n,\{1\}}\right| = \left|W_{n,\{2\}}\right| = m_2 F, \quad (73)$$

$$\left|W^{(2)}_{n,\{1\}}\right| = \left(\frac{N}{N-U_2}\right)\left(m_1 - \frac{N-U_2+U_1}{N}m_2\right)F. \quad (74)$$

In turn, the delivery load is given by

$$R_{\text{coded}} = (U_1+U_2) - \left(\frac{N}{N-U_2}\right)\left(U_1 m_1 + \frac{(U_1+U_2)(N-U_2)-U_1^2}{N}m_2\right). \quad (75)$$

- Region $(\frac{Nm_1}{N-U_2} + \frac{(N-U_2-U_1)m_2}{N-U_2} > 1)$ and $(m_1 + \frac{N+U_2-U_1}{U_2-U_1}(m_2-m_1) < 1)$: We have $|W_{n,\phi}| = 0$, and the subfile sizes satisfy

$$\left|W_{n,\{2\}}\right| + \left|W_{n,\{1,2\}}\right| = m_2 F, \quad (76)$$

$$\left|W_{n,\{2\}}\right| + \left|W^{(2)}_{n,\{1\}}\right| = (1-m_2)F, \quad (77)$$

where $\frac{N-U_2}{N-U_1} + \frac{U_2}{N-U_1}m_2 - \frac{N}{N-U_1}m_1 \leq |W_{n,\{2\}}|/F \leq \min\{m_2, 1-m_2\}$. That is, the caching scheme is not unique and the delivery load is given by

$$R_{\text{coded}} = U_2(1-m_2). \quad (78)$$

- Region $(m_1 + \frac{N+U_2-U_1}{U_2-U_1}(m_2-m_1) \geq 1)$: We have

$$\left|W_{n,\{2\}}\right| = \frac{N}{N+U_2-U_1}(1-m_1)F, \quad (79)$$

$$\left|W_{n,\{1,2\}}\right| = \frac{N}{N+U_2-U_1}\left(2m_1 - \frac{N-U_2+U_1}{N}\right)F, \quad (80)$$

and the delivery load is given by

$$R_{\text{coded}} = U_2\left(\frac{N}{N+U_2-U_1}\right)(1-m_1). \quad (81)$$

VI. Conclusion

In this paper, we have studied the benefits of coded placement in scenarios in heterogeneous caching systems. First, we have demonstrated the benefits of coded placement in systems where the helper-caches have different sizes and each helper-cache is connected to one user. We have shown that coded placement schemes outperform the best uncoded placement scheme for three-user systems with arbitrary cache sizes and K-user systems in the small total memory regime. In the proposed schemes, some of the signals intended to serve the users with small cache sizes are utilized in decoding the cache contents of users with larger cache sizes. We have observed that the gain due to coded placement increases with the heterogeneity in cache sizes and decreases with the number of files.

Next, we have considered systems with L helper-caches which are shared by K users. In particular, there are more users than helper-caches in the system and each user has access to only one helper-cache. We have demonstrated that the asymmetry in users' connectivity to the helper-caches can be exploited in reducing the delivery load when coded placement is utilized. That is, the proposed coded placement scheme outperforms the best uncoded placement scheme [26], when the helper-caches are associated with different number of users. For a two-cache system, we have provided an explicit characterization of the gain from coded placement. Then, we have extended our scheme to L-cache systems, where the parameters of the caching scheme are optimized by solving a linear program. Finally, we have extended our coded placement scheme to two-cache systems where the shared caches have heterogeneous sizes. Identifying efficient solutions in this general setting to more than two caches remains open. Additional future directions include overlapping demands, small libraries, considering hierarchical cache-enabled networks and general network and shared cache topologies.

As a final remark, coding over networks and coded caching in general, and coded placement in particular, remain to be of interest to the network design and research communities alike, the latter evident from the independent and concurrent work [27], [30] to this. Conference presentations of reference [27] appeared in ISIT 2019, and of this work in part [33] in Asilomar 2018 and in part in ICC 2019 [34]. Reference [30] already appeared in print (and was on ArXiv in May 2019). A preliminary version of this manuscript in part appeared in [35]. We encourage the reader who found their way to this manuscript to also read and credit [30].

APPENDIX
PROOF OF THEOREM 1

In this section, we present our caching schemes for three-user systems. The achievable delivery load in Theorem 1 consists of the following regions

- Region *I*: If $\sum_{i=1}^{3} m_i + \frac{2(m_2-m_1)}{N-1} + \frac{2(m_3-m_2)}{N-2} \leq 1$, then
$$R_{\text{coded}} = 3 - 3m_1 - 2m_2 - m_3 - \frac{3(m_2-m_1)}{N-1} - \frac{2(m_3-m_2)}{N-2}. \quad (82)$$
- Region *II*: If $\sum_{i=1}^{3} m_i + \frac{2(m_2-m_1)}{N-1} + \frac{2(m_3-m_2)}{N-2} > 1$, $Nm_3 \leq (N+3)m_2 + 3(N-2)m_1 - (N-1)$, and $Nm_3 \leq 2(N-1) - (2N-3)m_2$, then
$$R_{\text{coded}} = \frac{5}{3} - m_1 - \frac{2m_2}{3} - \frac{m_3}{3} - \frac{m_2-m_1}{3(N-1)}. \quad (83)$$
- Region *III*: If $\sum_{i=1}^{3} m_i + \frac{2(m_2-m_1)}{N-1} + \frac{2(m_3-m_2)}{N-2} > 1$, $Nm_3 > (N+3)m_2 + 3(N-2)m_1 - (N-1)$, and $Nm_2 + (N-2)m_1 \leq N-1$, then
$$R_{\text{coded}} = 2 - 2m_1 - m_2 - \frac{m_2-m_1}{N-1}. \quad (84)$$
- Region *IV*: If $Nm_2 + (N-2)m_1 > N-1$, and $Nm_3 > 2(N-1) - (2N-3)m_2$, then
$$R_{\text{coded}} = 1 - m_1. \quad (85)$$

Region *I* is a special case of Theorem 2. Next, we consider regions II to IV.

A. Region II

1) Placement Phase: Each file W_n is split into subfiles $W_{n,1}$, $W_{n,\{2\}}$, $W_{n,\{3\}}$, $W_{n,\{1,2\}}$, $\{W^{(1)}_{n,\{1,3\}}, W^{(2)}_{n,\{1,3\}}\}$, and $\{W^{(1)}_{n,\{2,3\}}, W^{(2)}_{n,\{2,3\}}, W^{(3)}_{n,\{2,3\}}, W^{(4)}_{n,\{2,3\}}\}$, such that

$$\left|W_{n,\{1\}}\right| = \left(\frac{2}{3} - m_1 - \frac{N(m_3-m_2)}{3(N-1)}\right)F, \quad (86)$$
$$\left|W_{n,\{2\}}\right| = \left|W_{n,\{3\}}\right| = \left|W_{n,\{1\}}\right| - (m_2-m_1)F, \quad (87)$$
$$\left|W^{(3)}_{n,\{2,3\}}\right| = \left|W^{(4)}_{n,\{2,3\}}\right| = (m_2-m_1)F, \quad (88)$$
$$\left|W_{n,\{1,2\}}\right| = \left|W^{(1)}_{n,\{1,3\}}\right| = \left|W^{(1)}_{n,\{2,3\}}\right| = \left(m_1 - 1/3 - \frac{N(m_3-m_2)}{3(N-1)}\right)F, \quad (89)$$
$$\left|W^{(2)}_{n,\{1,3\}}\right| = \left|W^{(2)}_{n,\{2,3\}}\right| = \frac{N(m_3-m_2)}{(N-1)}F, \quad (90)$$

where all subfiles are cached uncoded except for $W^{(2)}_{n,\{2,3\}}, \forall n$ are encoded before being placed at user 3. More specifically, the cache contents are given as

$$Z_1 = \bigcup_{n=1}^{N}\left(W_{n,\{1\}} \bigcup W_{n,\{1,2\}} \bigcup W^{(1)}_{n,\{1,3\}} \bigcup W^{(2)}_{n,\{1,3\}}\right), \quad (91)$$
$$Z_2 = \bigcup_{n=1}^{N}\left(W_{n,\{2\}} \bigcup W_{n,\{1,2\}} \bigcup \left(\bigcup_{i=1}^{4} W^{(i)}_{n,\{2,3\}}\right)\right), \quad (92)$$
$$Z_3 = \bigcup_{n=1}^{N}\left(W_{n,\{3\}} \bigcup \left(\bigcup_{i=1}^{2} W^{(i)}_{n,\{1,3\}}\right) \bigcup \left(\bigcup_{i=1,i\neq 2}^{4} W^{(i)}_{n,\{2,3\}}\right)\right) \bigcup \sigma_1\left(\left[W^{(2)}_{1,\{2,3\}}, \ldots, W^{(2)}_{N,\{2,3\}}\right]\right). \quad (93)$$

2) Delivery Phase: The server sends the following multicast signals

$$X'_{\{1,2\},\boldsymbol{d}} = W_{d_2,\{1\}} \oplus \left(W_{d_1,\{2\}} \bigcup W^{(3)}_{d_1,\{2,3\}}\right), \quad (94)$$
$$X_{\{1,3\},\boldsymbol{d}} = W_{d_3,\{1\}} \oplus \left(W_{d_1,\{3\}} \bigcup W^{(4)}_{d_1,\{2,3\}}\right), \quad (95)$$
$$X_{\{2,3\},\boldsymbol{d}} = W_{d_3,\{2\}} \oplus W_{d_2,\{3\}}, \quad (96)$$
$$X_{\{1,2,3\},\boldsymbol{d}} = W_{d_3,\{1,2\}} \oplus W^{(1)}_{d_2,\{1,3\}} \oplus W^{(1)}_{d_1,\{2,3\}}, \quad (97)$$
$$X''_{\{1,2\},\boldsymbol{d}} = W^{(2)}_{d_2,\{1,3\}} \oplus W^{(2)}_{d_1,\{2,3\}}. \quad (98)$$

3) Achievability: The proposed placement scheme is valid since the cache sizes constraints are satisfied. In the delivery phase, the users retrieve the requested pieces from the multicast signals using the cached subfiles. Additionally, using the multicast signal $X''_{\{1,2\},\boldsymbol{d}}$ and the cached piece $W^{(2)}_{d_2,\{1,3\}}$, user 3 decodes $W^{(2)}_{d_1,\{2,3\}}$, which is used in retrieving $W^{(2)}_{d_3,\{2,3\}}$ from $\sigma_1([W^{(2)}_{1,\{2,3\}}, \ldots, W^{(2)}_{N,\{2,3\}}])$.

B. Region III

1) Placement Phase: File W_n is split into subfiles $W_{n,1}$, $\{W^{(1)}_{n,\{2\}}, W^{(2)}_{n,\{2\}}\}$, $\{W^{(1)}_{n,\{3\}}, W^{(2)}_{n,\{3\}}, W^{(3)}_{n,\{3\}}\}$, $W_{n,\{1,3\}}$, and $\{W^{(1)}_{n,\{2,3\}}, W^{(2)}_{n,\{2,3\}}\}$, such that

$$\left|W_{n,\{1\}}\right| + \left|W_{n,\{1,3\}}\right| = \left|W^{(1)}_{n,\{2\}}\right| + \left|W^{(1)}_{n,\{2,3\}}\right| = m_1F, \quad (99)$$
$$\left|W_{n,\{1,3\}}\right| = \left|W^{(1)}_{n,\{2,3\}}\right| = m_1F - \left|W^{(1)}_{n,\{3\}}\right|, \quad (100)$$
$$\left|W^{(2)}_{n,\{2\}}\right| = \left|W^{(2)}_{n,\{3\}}\right| = \frac{N(m_2-m_1)}{N-1}F - \left|W^{(2)}_{n,\{2,3\}}\right|. \quad (101)$$

In particular, we have the following three cases.

- For $m_3 \leq \frac{N-2}{N} + \frac{m_2}{N-1} - \frac{(2N-3)(N-2)m_1}{(N-1)(N)}$
$$\left|W^{(2)}_{n,\{2,3\}}\right| = \left(\sum_{i=1}^{3} m_i + \frac{2(m_2-m_1)}{N-1} + \frac{2(m_3-m_2)}{N-2} - 1\right)F, \quad (102)$$
$$\left|W^{(3)}_{n,\{3\}}\right| = \frac{N(m_3-m_2)}{N-2}F, \ \left|W^{(1)}_{n,\{3\}}\right| = m_1F. \quad (103)$$
- For $\frac{m_2}{N-1} - \frac{(2N-3)(N-2)m_1}{(N-1)(N)} < m_3 - \frac{N-2}{N} \leq \frac{m_2}{N-1} - \frac{m_1}{(N-1)(N)}$
$$\left|W^{(2)}_{n,\{2,3\}}\right| = \frac{N(m_2-m_1)}{N-1}F, \quad (104)$$
$$\left|W^{(3)}_{n,\{3\}}\right| = \left(1 - 2m_1 - \frac{N(m_2-m_1)}{N-1}\right)F - \left|W^{(1)}_{n,\{3\}}\right|, \quad (105)$$
$$\left|W^{(1)}_{n,\{3\}}\right| = \frac{N-2}{2N-3}\left(1 - \frac{(N-3)m_1}{(N-2)} - \frac{N(m_2-m_1)}{N-1} - \frac{N(m_3-m_2)}{N-2}\right)F. \quad (106)$$

- For $m_3 > \frac{N-2}{N} + \frac{m_2}{N-1} - \frac{m_1}{(N-1)(N)}$

$$\left|W^{(3)}_{n,\{3\}}\right| = \left(1 - 2m_1 - \frac{N(m_2-m_1)}{N-1}\right)F, \quad (107)$$

$$\left|W^{(1)}_{n,\{3\}}\right| = 0, \ \left|W^{(2)}_{n,\{2,3\}}\right| = \frac{N(m_2-m_1)}{N-1}F, \quad (108)$$

The cache contents are defined as

$$Z_1 = \bigcup_{n=1}^{N}\left(W_{n,\{1\}} \bigcup W_{n,\{1,3\}}\right), \quad (109)$$

$$Z_2 = \bigcup_{n=1}^{N}\left(W^{(1)}_{n,\{2\}} \bigcup W^{(1)}_{n,\{2,3\}}\right) \bigcup \sigma_1\left(\left[W^{(2)}_{1,\{2\}}, \ldots, W^{(2)}_{N,\{2\}}\right]\right) \bigcup \sigma_1\left(\left[W^{(2)}_{1,\{2,3\}}, \ldots, W^{(2)}_{N,\{2,3\}}\right]\right), \quad (110)$$

$$Z_3 = \bigcup_{n=1}^{N}\left(W^{(1)}_{n,\{3\}} \bigcup W^{(1)}_{n,\{2,3\}}\right) \bigcup \sigma_1\left(\left[W^{(2)}_{1,\{3\}}, \ldots, W^{(2)}_{N,\{3\}}\right]\right) \bigcup \sigma_1\left(\left[W^{(2)}_{1,\{2,3\}}, \ldots, W^{(2)}_{N,\{2,3\}}\right]\right) \bigcup \sigma_1\left(\left[W_{1,\{1,3\}}, \ldots, W_{N,\{1,3\}}\right]\right) \bigcup \sigma_2\left(\left[W^{(3)}_{1,\{3\}}, \ldots, W^{(3)}_{N,\{3\}}\right]\right). \quad (111)$$

2) Delivery Phase: The server sends the following multicast signals

$$X_{\{1,2\},\boldsymbol{d}} = \left(W_{d_2,\{1\}} \bigcup W_{d_2,\{1,3\}}\right) \oplus \left(W^{(1)}_{d_1,\{2\}} \bigcup W^{(1)}_{d_1,\{2,3\}}\right), \quad (112)$$

$$X_{\{2,3\},\boldsymbol{d}} = \left(W^{(1)}_{d_3,\{2\}} \bigcup W^{(2)}_{d_3,\{2\}}\right) \oplus \left(W^{(1)}_{d_2,\{3\}} \bigcup W^{(2)}_{d_2,\{3\}}\right), \quad (113)$$

$$X_{\{1,3\},\boldsymbol{d}} = W_{d_3,\{1\}} \oplus W^{(1)}_{d_1,\{3\}}. \quad (114)$$

The following unicast signals complete the requested files and help users $\{2, 3\}$ in decoding their cache contents.

$$X_{\{1\},\boldsymbol{d}} = W^{(2)}_{d_1,\{2\}} \bigcup W^{(2)}_{d_1,\{2,3\}} \bigcup W^{(2)}_{d_1,\{3\}} \bigcup W^{(3)}_{d_1,\{3\}}, \quad (115)$$

$$X_{\{2\},\boldsymbol{d}} = W^{(3)}_{d_2,\{3\}}. \quad (116)$$

3) Achievability: User 2 decodes its cache using $W^{(2)}_{d_1,\{2\}}, W^{(2)}_{d_1,\{2,3\}}$ from $X_{\{1\},\boldsymbol{d}}$. Similarly, user 3 decodes its cache using $W^{(2)}_{d_1,\{2,3\}}, W^{(2)}_{d_1,\{3\}}, W^{(3)}_{d_1,\{3\}}$ from $X_{\{1\},\boldsymbol{d}}$ and $W^{(3)}_{d_2,\{3\}}$ from $X_{\{2\},\boldsymbol{d}}$.

C. Region IV

Next, we consider the case where $m_1 + m_2 \leq 1$, since uncoded placement is optimal for $m_1 + m_2 > 1$ [15].

1) Placement Phase: W_n is split into $W_{n,\{1,2\}}$, $W_{n,\{1,3\}}$, and $\{W^{(1)}_{n,\{2,3\}}, W^{(2)}_{n,\{2,3\}}, W^{(3)}_{n,\{2,3\}}\}$, such that

$$\left|W_{n,\{1,2\}}\right| = m_1 F - \left|W_{n,\{1,3\}}\right|, \quad (117)$$

$$\left|W_{n,\{1,2\}}\right| = \left|W^{(2)}_{n,\{2,3\}}\right|, \ \left|W_{n,\{1,3\}}\right| = \left|W^{(1)}_{n,\{2,3\}}\right|, \quad (118)$$

$$\left|W^{(3)}_{n,\{2,3\}}\right| = (1 - 2m_1)F. \quad (119)$$

In particular, we have

$$\left|W_{n,\{1,2\}}\right| = \begin{cases} \left(\frac{Nm_2+(N-2)m_1}{N-1} - 1\right)F, & \text{if } m_3 \leq \frac{(N-1)+m_1}{N}, \\ 0, & \text{otherwise.} \end{cases} \quad (120)$$

The cache contents are defined as

$$Z_1 = \bigcup_{n=1}^{N}\left(W_{n,\{1,2\}} \bigcup W_{n,\{1,3\}}\right), \quad (121)$$

$$Z_2 = \bigcup_{n=1}^{N}\left(W^{(1)}_{n,\{2,3\}} \bigcup W^{(2)}_{n,\{2,3\}}\right) \bigcup \sigma_1\left(\left[W_{1,\{1,2\}}, \ldots, W_{N,\{1,2\}}\right]\right) \bigcup \sigma_1\left(\left[W^{(3)}_{1,\{2,3\}}, \ldots, W^{(3)}_{N,\{2,3\}}\right]\right), \quad (122)$$

$$Z_3 = \bigcup_{n=1}^{N}\left(W^{(1)}_{n,\{2,3\}} \bigcup W^{(2)}_{n,\{2,3\}}\right) \bigcup \sigma_1\left(\left[W_{1,\{1,3\}}, \ldots, W_{N,\{1,3\}}\right]\right) \bigcup \sigma_1\left(\left[W^{(3)}_{1,\{2,3\}}, \ldots, W^{(3)}_{N,\{2,3\}}\right]\right). \quad (123)$$

2) Delivery Phase: The server sends the following signals

$$X_{\{1,2\},\boldsymbol{d}} = W_{d_2,\{1,3\}} \oplus W^{(1)}_{d_1,\{2,3\}}, \quad (124)$$

$$X_{\{1,3\},\boldsymbol{d}} = W_{d_3,\{1,2\}} \oplus W^{(2)}_{d_1,\{2,3\}}, \quad (125)$$

$$X_{\{1\},\boldsymbol{d}} = W^{(3)}_{d_1,\{2,3\}}. \quad (126)$$

3) Achievability: User 2 retrieves $W_{d_2,\{1,2\}}$ from its cache using $W_{d_3,\{1,2\}}$ which is extracted from $X_{\{1,3\},\boldsymbol{d}}$. Similarly, user 3 retrieves $W_{d_3,\{1,3\}}$ by utilizing $X_{\{1,2\},\boldsymbol{d}}$.

REFERENCES

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

[2] K. Wan, D. Tuninetti, and P. Piantanida, "On the optimality of uncoded cache placement," in *Proc. IEEE ITW*, 2016, pp. 161–165.

[3] Q. Yu, M. A. Maddah-Ali, and A. S. Avestimehr, "The exact rate-memory tradeoff for caching with uncoded prefetching," *IEEE Trans. Inf. Theory*, vol. 64, no. 2, pp. 1281–1296, Feb. 2018.

[4] Z. Chen, P. Fan, and K. B. Letaief, "Fundamental limits of caching: Improved bounds for small buffer users," *IET Commun.*, vol. 10, no. 17, pp. 2315–2318, Nov. 2016.

[5] M. M. Amiri and D. Gündüz, "Fundamental limits of coded caching: Improved delivery rate-cache capacity tradeoff," *IEEE Trans. Commun.*, vol. 65, no. 2, pp. 806–815, Feb. 2017.

[6] J. Gómez-Vilardebó, "Fundamental limits of caching: Improved bounds with coded prefetching," 2016, *arXiv:1612.09071*.

[7] H. Ghasemi and A. Ramamoorthy, "Improved lower bounds for coded caching," *IEEE Trans. Inf. Theory*, vol. 63, no. 7, pp. 4388–4413, Jul. 2017.

[8] S. H. Lim, C.-Y. Wang, and M. Gastpar, "Information-theoretic caching: The multi-user case," *IEEE Trans. Inf. Theory*, vol. 63, no. 11, pp. 7018–7037, Nov. 2017.

[9] K. Zhang and C. Tian, "Fundamental limits of coded caching: From uncoded prefetching to coded prefetching," *IEEE J. Sel. Areas Commun.*, vol. 36, no. 6, pp. 1153–1164, Jun. 2018.

[10] C.-Y. Wang, S. S. Bidokhti, and M. Wigger, "Improved converses and gap-results for coded caching," in *Proc. IEEE ISIT*, 2017 pp. 2428–2432.

[11] Q. Yu, M. A. Maddah-Ali, and A. S. Avestimehr, "Characterizing the rate-memory tradeoff in cache networks within a factor of 2," *IEEE Trans. Inf. Theory*, vol. 65, no. 1, pp. 647–663, Jan. 2019.

[12] M. Ji *et al.*, "On the fundamental limits of caching in combination networks," in *Proc. IEEE SPAWC*, 2015, pp. 695–699.

[13] A. A. Zewail and A. Yener, "Combination networks with or without secrecy constraints: The impact of caching relays," *IEEE J. Sel. Areas Commun.*, vol. 36, no. 6, pp. 1140–1152, Jun. 2018.
[14] M. Ji, G. Caire, and A. F. Molisch, "Fundamental limits of caching in wireless D2D networks," *IEEE Trans. Inf. Theory*, vol. 62, no. 2, pp. 849–869, Feb. 2016.
[15] A. M. Ibrahim, A. A. Zewail, and A. Yener, "Coded caching for heterogeneous systems: An optimization perspective," *IEEE Trans. Commun.*, vol. 67, no. 8, pp. 5321–5335, Aug. 2019.
[16] A. M. Ibrahim, A. A. Zewail, and A. Yener, "Device-to-device coded-caching with distinct cache sizes," *IEEE Trans. Commun.*, vol. 68, no. 5, pp. 2748–2762, May 2020.
[17] A. A. Zewail and A. Yener, "Device-to-device secure coded caching," *IEEE Trans. Inf. Forensics Security*, vol. 15, pp. 1513–1524, 2020.
[18] A. Sengupta, R. Tandon, and T. C. Clancy, "Layered caching for heterogeneous storage," in *Proc. IEEE Asilomar*, 2016, pp. 719–723.
[19] Y.-P. Wei and S. Ulukus, "Coded caching with multiple file requests," in *Proc. IEEE Allerton*, Monticello, IL, USA, 2017, pp. 437–442.
[20] B. Asadi, L. Ong, and S. J. Johnson, "Centralized caching with unequal cache sizes," in *Proc. IEEE ITW*, Guangzhou, China, 2018, pp. 1–5.
[21] S. S. Bidokhti, M. Wigger, and A. Yener, "Benefits of cache assignment on degraded broadcast channels," *IEEE Trans. Inf. Theory*, vol. 65, no. 11, pp. 6999–7019, Nov. 2019.
[22] Q. Yang and D. Gündüz, "Coded caching and content delivery with heterogeneous distortion requirements," *IEEE Trans. Inf. Theory*, vol. 64, no. 6, pp. 4347–4364, Jun. 2018.
[23] A. M. Ibrahim, A. A. Zewail, and A. Yener, "On coded caching with heterogeneous distortion requirements," in *Proc. IEEE ITA*, 2018, pp. 1–9.
[24] D. Cao, D. Zhang, P. Chen, N. Liu, W. Kang, and D. Gündüz, "Coded caching with asymmetric cache sizes and link qualities: The two-user case," *IEEE Trans. Commun.*, vol. 67, no. 9, pp. 6112–6126, Sep. 2019.
[25] D. Zhang and N. Liu, "Coded cache placement for heterogeneous cache sizes," in *Proc. IEEE ITW*, 2018, pp. 1–5.
[26] E. Parrinello, A. Ünsal, and P. Elia, "Fundamental limits of coded caching with multiple antennas, shared caches and uncoded prefetching," *IEEE Trans. Inf. Theory*, vol. 66, no. 4, pp. 2252–2268, Apr. 2020.
[27] K. Wan, D. Tuninetti, M. Ji, and G. Caire, "Novel inter-file coded placement and D2D delivery for a cache-aided fog-RAN architecture," 2018, *arXiv:1811.05498*.
[28] N. Karamchandani, U. Niesen, M. A. Maddah-Ali, and S. N. Diggavi, "Hierarchical coded caching," *IEEE Trans. Inf. Theory*, vol. 62, no. 6, pp. 3212–3229, Apr. 2016.
[29] M. Takita, M. Hirotomo, and M. Morii, "Coded caching for hierarchical networks with a different number of layers," in *Proc. 5th Int. Symp. Comput. Netw. (CANDAR)*, Aomori, Japan, 2017, pp. 249–255.
[30] K. Wan, D. Tuninetti, M. Ji, and G. Caire, "On the fundamental limits of fog-RAN cache-aided networks with downlink and sidelink communications," *IEEE Trans. Inf. Theory*, vol. 67, no. 4, pp. 2353–2378, Apr. 2021.
[31] A. Sengupta and R. Tandon, "Improved approximation of storage-rate tradeoff for caching with multiple demands," *IEEE Trans. Commun.*, vol. 65, no. 5, pp. 1940–1955, May 2017.
[32] F. J. MacWilliams and N. J. A. Sloane, *The Theory of Error-Correcting Codes*. Amsterdam, The Netherlands: Elsevier, 1977.
[33] A. M. Ibrahim, A. A. Zewail, and A. Yener, "Benefits of coded placement for networks with heterogeneous cache sizes," 2018, *arXiv:1811.04067*.
[34] A. M. Ibrahim, A. A. Zewail, and A. Yener, "Coded placement for systems with shared caches," in *Proc. IEEE ICC*, 2019, pp. 1–6.
[35] A. M. Ibrahim, "Data storage and energy management in emerging networks," Ph.D. dissertation, Dept. Elect. Eng., Pennsylvania State Univ., State College, PA, USA, Apr. 2019. [Online]. Available: libraries.psu.edu

Abdelrahman M. Ibrahim received the Ph.D. degree in electrical engineering from the Pennsylvania State University, University Park, PA, USA, in 2019, where he was a Research Assistant with the Wireless Communications and Networking Laboratory from 2014 to 2019. He is currently with Qualcomm.

Ahmed A. Zewail (Member, IEEE) received the Ph.D. degree from Pennsylvania State University, in 2019, where he was a Research Assistant with the Wireless Communications and Networking Laboratory from 2013 to 2019. He is currently with Qualcomm.

Aylin Yener (Fellow, IEEE) received the B.Sc. degree in electrical and electronics engineering and the B.Sc. degree in physics from Bogazici University, Istanbul, Turkey, and the M.S. and Ph.D. degrees in electrical and computer engineering from Rutgers University, New Brunswick, NJ, USA. She is the Roy and Lois Chope Chair of Engineering with The Ohio State University, Columbus, OH, USA, and is a Professor of Electrical and Computer Engineering, Computer Science and Engineering, and Integrated Systems Engineering. Prior to joining Ohio State in 2020, she was a University Distinguished Professor of Electrical Engineering and a Dean's Fellow with The Pennsylvania State University, University Park, PA, USA, where she joined the faculty as an Assistant Professor in 2002. She was a Visiting Professor of Electrical Engineering with Stanford University from 2016 to 2018, where he was a Visiting Associate Professor from 2008 to 2009. She was a Visiting Researcher with Telecom Paris Tech in 2016. Her current research interests are in information security and privacy, green communications, caching, 6G, and more generally in the fields of information theory, communication theory, and networked systems. She received the NSF CAREER Award in 2003, the Best Paper Award in Communication Theory from the IEEE International Conference on Communications in 2010, the Penn State Engineering Alumni Society (PSEAS) Outstanding Research Award in 2010, the IEEE Marconi Prize Paper Award in 2014, the PSEAS Premier Research Award in 2014, the Leonard A. Doggett Award for Outstanding Writing in Electrical Engineering at Penn State in 2014, the IEEE Women in Communications Engineering Outstanding Achievement Award in 2018, the IEEE Communications Society Best Tutorial Paper Award in 2019, and the IEEE Communications Society Communication Theory Technical Achievement Award in 2020. She has been a Distinguished Lecturer for the IEEE Information Theory Society (2019–2021), the IEEE Communications Society (2018–2019), and the IEEE Vehicular Technology Society (2017–2021). She is currently serving as the Junior Past President of the IEEE Information Theory Society. She was the President (2020), the Vice President (2019), the Second Vice President (2018), an Elected Member of the Board of Governors (2015–2018), and the Treasurer (2012–2014) of the IEEE Information Theory Society. She served as the Student Committee Chair for the IEEE Information Theory Society (2007–2011), and was the Co-Founder of the Annual School of Information Theory in North America in 2008. She was a Technical (Co)-Chair for various symposia/tracks at the IEEE ICC, PIMRC, VTC, WCNC, and Asilomar in 2005, 2008–2014, and 2018. She served as an Editor for IEEE TRANSACTIONS ON COMMUNICATIONS (2009–2012) and IEEE TRANSACTIONS ON MOBILE COMPUTING (2017–2018), and an Editor and an Editorial Advisory Board Member for IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS (2001–2012). She also served as a Guest Editor for IEEE TRANSACTIONS ON INFORMATION FORENSICS AND SECURITY in 2011, and IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS in 2015. She currently serving as a Senior Editor for IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS. She is on the Senior Editorial Board of IEEE JOURNAL ON SELECTED AREAS IN INFORMATION THEORY and is an Area Editor for Security and Privacy for the IEEE TRANSACTIONS ON INFORMATION THEORY.