Coded Placement for Systems with Shared Caches

Abdelrahman M. Ibrahim, Ahmed A. Zewail, and Aylin Yener

Wireless Communications and Networking Laboratory (WCAN)
Electrical Engineering and Computer Science Department
The Pennsylvania State University, University Park, PA 16802.
{*ami137,zewail*}*@psu.edu* *yener@engr.psu.edu*

***Abstract*—In this work, we consider a cache-aided network where the users share the end-caches. In particular, a user has access to only one of the caches and the number of caches is less than the number of users. We propose a coded placement scheme that exploits the asymmetry in the number of users associated with each cache. Some of the signals sent to the overloaded caches facilitate the decoding of the coded subfiles stored at the underloaded caches. We present an explicit caching scheme and fully characterize the coded placement gain for two-cache systems. Then, we generalize our scheme to larger networks, where the optimal parameters are characterized by solving a linear program. We observe that, with the proposed scheme, as the asymmetry in the users' connectivity increases, the gain from coded placement is more evident.**

I. Introduction

Caching reduces the end-to-end delay of content delivery during peak-traffic hours, known as the *delivery phase*, by pre-allocating some of the data in the cache memories at the network nodes during off-peak hours, known as the *placement phase*. Reference [1] has proposed a caching paradigm where the joint design of the two phases results in a reduction in the delivery load that is proportional to the total memory in the system which is known as the global caching gain. The cache contents are designed in a manner that allows serving multiple users simultaneously in the delivery phase. The fundamental trade-off between the delivery load on the server and the cache sizes in the network has been studied in several setups with different network topologies [1]–[10]. In particular, references [2]–[4], [11], [12] have investigated the effect of heterogeneity in the users' cache sizes on the delivery load memory trade-off. Taking into consideration the heterogeneity in cache sizes in the joint design of the cache placement and delivery schemes provides significant improvements over schemes tailored to uniform cache sizes. Optimal caching schemes with respect to uncoded placement have been studied in [3]. References [5], [13] considered setups where the cache sizes at the end-users can be optimized for further gain depending on the channel conditions.

Coded caching schemes are often categorized according to whether we consider coded or uncoded cache placement schemes. In caching with uncoded placement, the server places uncoded pieces of each file in the cache memories of the network nodes [1]–[4]. In contrast, in systems with *coded placement*, the server places coded pieces of the files in the users' caches [6], [7]. While uncoded placement is sufficient for some systems, coding over files, in general, has the potential to perform better. For instance, in systems with equal cache sizes, references [6], [7] have shown that coded placement is beneficial in the small memory regime when the number of files is less than or equal the number of users. Reference [14] has generalized the coded placement scheme in [7] to exploit the repeated requests in systems with multiple requests per cache. Recent reference [15] has shown that coded placement is essential in achieving the optimal delivery load in a two-user system with unequal cache sizes. More recently, for larger systems, we have proposed coded placement schemes that illustrate the role of coding over files in the placement phase in enhancing the utilization of cache memories and achieving lower delivery load [16]. In [16], users cache both uncoded and coded pieces of the files, and users with large memories recover the cached coded pieces using the transmissions intended to serve users with smaller memories. We observe that the gain from coded placement increases as the differences between the cache sizes grow, and decreases as the number of files grows.

In this work, we propose a caching scheme for networks where multiple users share the same cache memory, motivated by our coded placement in systems with unequal cache sizes [16]. In particular, we consider a system with K users who share $L \leq K$ helper-caches, each of which assists an arbitrary number of different users. A similar model has been considered recently in [17] where the authors focused on caching policies with uncoded placement. It has been shown that applying the placement scheme in [1] and adjusting the delivery strategy by grouping the users into multiple groups, where each group contains at most L users and the users from the same group are connected to different caches is sufficient to achieve the optimal delivery load under uncoded placement. In our proposed scheme, the caches are populated with both coded and uncoded pieces of the library files. In particular, depending on the network connectivity pattern, we place uncoded pieces in the cache which is shared by a larger number of users, while storing coded pieces in the remaining caches. We show how coded placement exploits the asymmetry in the cache assignment in minimizing the delivery load. We first explain the coded placement scheme for two-cache systems with arbitrary number of users, then generalize the caching scheme to larger systems.

The remainder of the paper is organized as follows. In

978-1-5386-8088-9/19/$31.00 ©2019 IEEE

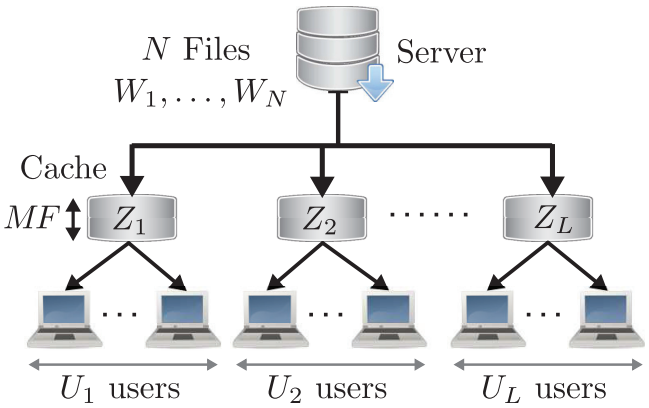

Fig. 1: Caching system with heterogeneous cache sizes.

Section II, we describe the system model. In Section III, we present examples to illustrate the key idea of our scheme. In Section IV, we detail the achievability technique for a two-cache system. Section V generalizes our achievability for a K-cache system. Section VI summarizes our conclusions.

Notation: Vectors are represented by boldface letters, $\oplus$ refers to the binary XOR operation, $|W|$ denotes cardinality 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.

II. System Model and Preliminaries

We consider a caching system where a single server is connected to K users via a shared error-free multicast link [1], as shown in Fig. 1. The server has access to a library $\{W_1, \ldots, W_N\}$ of N files, each with size F symbols over the field $\mathbb{F}_{2^r}$. There are L equal-size cache memories in the system, each of size MF bits where $L < K$. Each user is connected to one of the L caches via an error-free link. That is, each of the users has direct access to one of the caches of the system. The network topology is known to the server. We define $\mathcal{U}_i$ to denote the set of users connected to cache i, $i = \{1, ..., L\}$ and $U_i = |\mathcal{U}_i|$ is the number of users connected to cache i, i.e., $K = \sum_{i=1}^{L} U_i$. Without loss of generality, we assume that $U_1 \geq U_2 \geq ... \geq U_L$. The system operates over two phases: placement phase and delivery phase.

A. Placement Phase

In the placement phase, the server populates the cache memories without the knowledge of the users' demands which will be known in the delivery phase. The server designs the cache contents taking into account the network topology, i.e., the number of users connected to each cache memory. More specifically, the network connectivity is represented by $\boldsymbol{U} \triangleq [U_1, \ldots, U_L]$ and the contents of cache i is defined as

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

which satisfies the cache size constraint $H(Z_i) \leq MF$.

B. Delivery Phase

In the delivery phase, user k requests file W_{d_k} from the server. The users' demands are uniform and independent as in [1]. The K users are served over U_1 delivery rounds such that in each round, we choose one of the users in $\mathcal{U}_i$, $\forall i$ that needs to be served. In particular, 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 size M and network connectivity $\boldsymbol{U}$, the worst-case total delivery load $R(M, \boldsymbol{U}) \triangleq \sum_{r=1}^{U_1} \sum_{\mathcal{T}} |X^{(r)}_{\mathcal{T},\boldsymbol{d}}|/F$ 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 [11], [12], where each user requests the same number of files.* ■

In our achievability scheme, we utilize maximum distance separable (MDS) codes which are defined as follows.

Definition 1. *[18] 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*

$$\begin{aligned}[i_1, \ldots, i_k]\boldsymbol{G}_{k\times n} &= [i_1, \ldots, i_k][\boldsymbol{I}_{k\times k}\ \boldsymbol{P}_{k\times n-k}] \\ &= [i_1, \ldots, i_k, c_{k+1}, \ldots, c_n],\end{aligned} \tag{2}$$

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

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

to denote the $N - j$ parity symbols in the codeword. Note that $\sigma_j([i_1, \ldots, i_N])$ represents $N - j$ 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. Examples

A. Example 1: Two-cache system

Consider a system with $K = 3$, $N \geq 3$, $L = 2$ and $M \leq 1$, where users 1 and 2 are connected to cache 1 and user 3 is connected to cache 2, i.e., $\mathcal{U}_1 = \{1, 2\}$ and $\mathcal{U}_2 = \{3\}$ as illustrated in Fig. 2.

1) The uncoded placement scheme [17]: Each file is divided into 3 subfiles, $W_{n,1}$, $W_{n,2}$ and $W_{n,0}$, such that $|W_{n,1}| = |W_{n,2}| = \frac{M}{N}F$ bits and $|W_{n,0}| = (1 - 2\frac{M}{N})F$ bits [1]. The cached contents are given by

$$Z_1 = \{W_{1,1}, W_{2,1}, W_{3,1}\}, \tag{4}$$

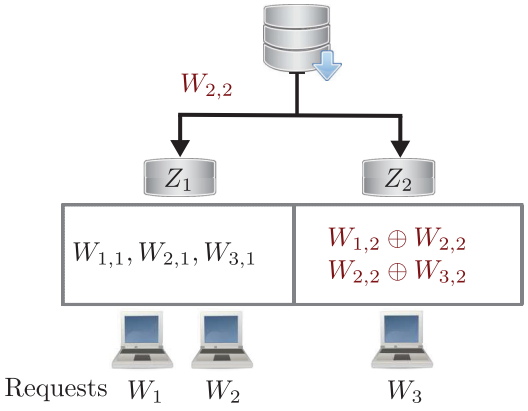

Fig. 2: The coded placement scheme for Example 1.

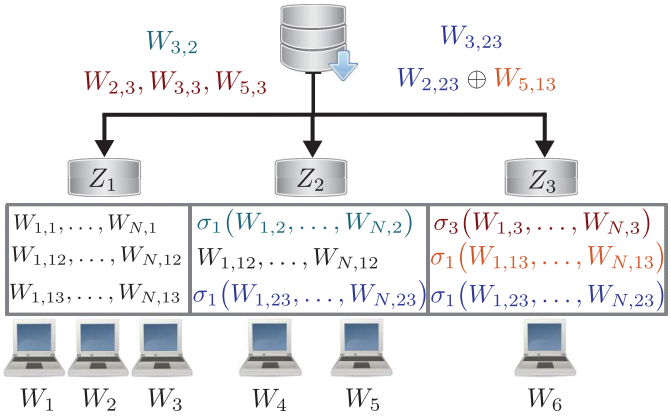

Fig. 3: The coded placement scheme for Example 2.

$$Z_2 = \{W_{1,2}, W_{2,2}, W_{3,2}\}. \tag{5}$$

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:

$$W_{1,0}, W_{3,0}, W_{1,2} \oplus W_{3,1}. \tag{6}$$

In round 2, the following unicast signals are sent to user 2.

$$W_{2,0}, W_{2,2}. \tag{7}$$

In turn, all the users recover their requested files using the cached contents. The total delivery load is given by

$$R_{\text{uncoded}} = 3\left(1 - \frac{2M}{N}\right) + \frac{2M}{N}, \tag{8}$$

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 $W_{n,0}$ of size $F - \frac{2MF}{N} - \frac{MF}{N-1}$ bits. The stored contents at the caches are given by

$$Z_1 = \{W_{1,1}, W_{2,1}, W_{3,1}\}, \tag{9}$$

$$Z_2 = \{W_{1,2} \oplus W_{2,2}, W_{2,2} \oplus W_{3,2}\}, \tag{10}$$

which is illustrated in Fig. 2.

Assuming that user k requests file k, the server transmits the following signals over the two rounds.

$$W_{1,0}, W_{2,0}, W_{3,0}, W_{2,2}, W_{1,2} \oplus W_{3,1}. \tag{11}$$

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. 2. In turn, we have

$$R_{\text{coded}} = 3\left(1 - \frac{2M}{N} - \frac{M}{N-1}\right) + \frac{2M}{N(N-1)} + \frac{2M}{N}, \tag{12}$$

which can be expressed as

$$R_{\text{coded}} = R_{\text{uncoded}} - \frac{M}{N(N-1)}, \tag{13}$$

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 [17]: 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

$$Z_1 = \{W_{n,1}, W_{n,12}, W_{n,13} : \quad \forall n\}, \tag{14}$$

$$Z_2 = \{W_{n,2}, W_{n,12}, W_{n,23} : \quad \forall n\}, \tag{15}$$

$$Z_3 = \{W_{n,3}, W_{n,13}, W_{n,23} : \quad \forall n\}. \tag{16}$$

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

$$W_{1,2} \oplus W_{4,1}, W_{1,3} \oplus W_{6,1}, W_{4,3} \oplus W_{6,2}, \\ W_{1,23} \oplus W_{4,13} \oplus W_{6,12}. \tag{17}$$

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

$$W_{2,2} \oplus W_{5,1}, W_{2,3}, W_{5,3}, W_{2,23} \oplus W_{5,13}. \tag{18}$$

Finally, in round 3, we serve user 3 by sending

$$W_{3,23}, W_{3,2}, W_{3,3}. \tag{19}$$

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

$$R_{\text{uncoded}}F = 3|W_{n,23}| + 3|W_{n,2}| + 5|W_{n,3}| = \left(\frac{13}{3} - \frac{5M}{N}\right)F. \tag{20}$$

2) The proposed coded placement scheme: Similarly, each file is divided into 6 subfiles, $W_{n,1}$, $W_{n,2}$, $W_{n,3}$ $W_{n,12}$, $W_{n,13}$,

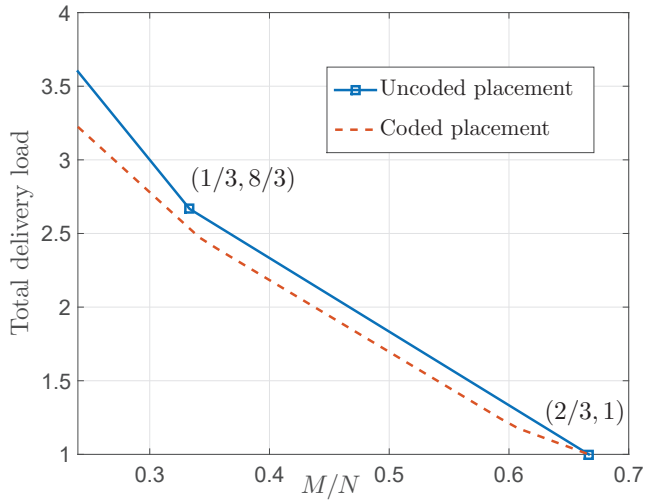

Fig. 4: The achievable normalized delivery load for $L = 3$, $N = K = 6$, $U_1 = 3$, $U_2 = 2$ and $U_3 = 1$.

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

$$Z_1 = \{W_{n,1}, W_{n,12}, W_{n,13} : \quad \forall n\}, \tag{21}$$

$$Z_2 = \{\sigma_1([W_{1,2}, ..., W_{N,2}]), \sigma_1([W_{1,23}, ..., W_{N,23}]), W_{1,12}, ..., W_{N,12}\}, \tag{22}$$

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

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

In the delivery phase, user k requests file k and the server constructs the signals defined in (17)-(19); 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

$$W_{2,23} \oplus W_{5,13}, W_{2,3}, W_{5,3}, W_{3,3}, W_{3,2}, W_{3,23}. \tag{24}$$

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 $W_{n,13}$. The decoding is illustrated in Fig. 3.

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

$$\min_{a_{\mathcal{S}} \geq 0} \quad R_{\text{coded}} = 3a_2 + 5a_3 + 3a_{23} \tag{25a}$$

$$\text{subject to} \quad \sum_{\mathcal{S} \subset [L]} a_{\mathcal{S}} = 1, \tag{25b}$$

$$N(a_1 + a_{12} + a_{13}) \leq M, \tag{25c}$$

$$(N-1)(a_2 + a_{13}) + Na_{12} \leq M, \tag{25d}$$

$$(N-3)a_3 + (N-1)(a_{13} + a_{23}) \leq M, \tag{25e}$$

$$a_1 \leq a_2 \leq a_3, \; a_{12} \leq a_{13} \leq a_{23}. \tag{25f}$$

(25b) ensures the feasibility of the file partitioning. Assuming that $a_{123} = 0$, conditions (25c)-(25e) 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. 4, we show that the proposed scheme achieves a lower total delivery load compared with the uncoded placement scheme in [17].

IV. 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 $q = U_1 - U_2$.

A. Placement Phase

Divide each file into the subfiles: $W_{n,0}$, $W_{n,1}$, $W_{n,2}$ and $W_{n,12}$. The cache contents are given by

$$Z_1 = \{W_{n,1}, W_{n,12} : \quad \forall n\}, \tag{26}$$

$$Z_2 = \{\sigma_q([W_{1,2}, ..., W_{N,2}]), W_{n,12} : \quad \forall n\}. \tag{27}$$

B. Delivery Phase

Next, we describe the caching scheme in three memory regimes.

1) 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 $|W_{n,0}| = (1 - \frac{M}{N} - \frac{M}{N-q})F$. 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 $\{(1, U_1+1), (2, U_1+2), ..., (U_2, K)\}$. Second, we send q unicast signals in the form of $W_{d_x,2}$ to users $x \in \{U_2+1, ..., U_1\}$. 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 $\{W_{d_k,0} : \forall k\}$, 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

$$R_{\text{coded}} = K\left(1 - \frac{M}{N} - \frac{M}{N-q}\right) + \frac{qM}{N-q} + \frac{U_2 M}{N-q}, \tag{28}$$

$$= R_{\text{uncoded}} - \frac{qU_2 M}{N(N-q)}. \tag{29}$$

Remark 2. *The last term in (29) 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.* ■

2) Region $(\frac{N-q}{2N-q} \leq \frac{M}{N} < 0.5)$*:* In this case, we choose $|W_{n,0}| = |W_{n,12}| = 0$, $|W_{n,1}| = \frac{M}{N}F$, and $|W_{n,2}| = (1 -$

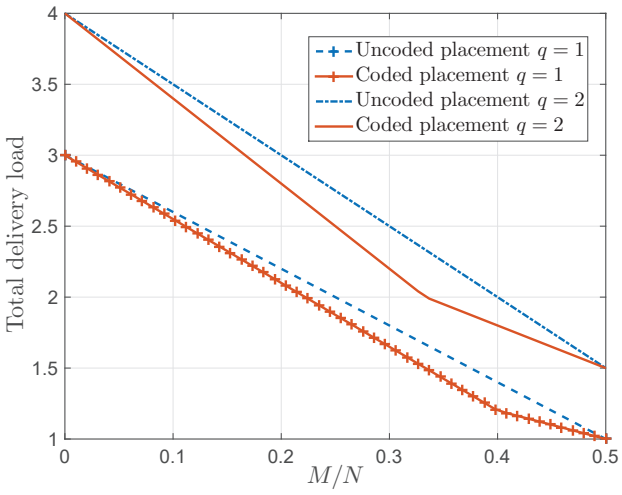

Fig. 5: The achievable normalized delivery load for $L=2$, $N=K=q+2$, $U_1=q+1$ and $U_2=1$.

$\frac{M}{N})F$. The delivery procedure is similar to the first case, and the total delivery load is given by

$$R_{\text{coded}} = U_1\left(1-\frac{M}{N}\right). \tag{30}$$

3) Region ($\frac{M}{N}\geq 0.5$)*:* In this case, there is no gain from coded placement and the total delivery load is given by

$$R_{\text{coded}} = R_{\text{uncoded}} = U_1\left(1-\frac{M}{N}\right). \tag{31}$$

Remark 3. *The proposed scheme is optimal with respect to the cut-set bound [11] for* $\frac{M}{N}\geq\frac{N-q}{2N-q}$. ■

In Fig. 5, 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 [17] increases with q, as explained in Remark 2.

V. L-Cache System

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

A. 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 Section III, we have illustrated that coded placement outperforms uncoded placement when a subset of the unicast/multicast signals is utilized in decoding the cached subfiles. In general, we assume that cache $s\in\mathcal{S}$ stores $(N-\lambda_{\mathcal{S}}^{(s)})$ independent equations in subfiles $W_{1,\mathcal{S}},\dots,W_{N,\mathcal{S}}$, i.e., Cache s is defined as

$$Z_s = \left\{\sigma_{\lambda_{\mathcal{S}}^{(s)}}([W_{1,\mathcal{S}},\dots,W_{N,\mathcal{S}}]), \forall\mathcal{S}\subset[L],\, s\in\mathcal{S}\right\}, \tag{32}$$

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 [17] and characterize the signals that can be utilized in decoding the cached contents.

B. Delivery Phase

Our delivery scheme is based on the delivery procedure in [17], 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 $[L-1]$.

⋮

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 $[L-l+1]$.

⋮

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 [17], 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

$$\tilde{X}_{\mathcal{T},l} = \oplus_{k\in\mathcal{T}\cap[L-l+1]} W_{d_k,\mathcal{T}\setminus\{k\}}, \tag{33}$$

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

$$R = \sum_{l=1}^{L}(U_{L-l+1}-U_{L-l+2}) \sum_{\mathcal{T}\subset[L]:\mathcal{T}\cap[L-l+1]\neq\phi} |\tilde{X}_{\mathcal{T},l}|/F, \tag{34}$$

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

Remark 4. *If we have* $|\mathcal{T}\cap[L-l+1]|<|\mathcal{T}|$*, the multicast signal defined in (33) 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 $U_1\geq U_2\geq\dots\geq U_L$, we assume that the subfile sizes satisfy

$$a_{\{s_1,\dots,s_{t-1},s_t\}} \leq a_{\{s_1,\dots,s_{t-1},s_t+1\}}, \tag{35}$$

$$a_{\{s_1,\dots,s_{t-1},L\}} \leq a_{\{s_1,\dots,s_{t-1}+1,s_{t-1}+2\}}, \tag{36}$$

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

$$|\tilde{X}_{\mathcal{T},l}| = a_{\mathcal{T}\setminus\{k\}}F,\ k=\arg\min_{i\in\mathcal{T}\cap[L-l+1]} i, \tag{37}$$

and the total normalized delivery load can be expressed as

$$R = \sum_{t=0}^{L-1}\sum_{\mathcal{S}\subset[L]:|\mathcal{S}|=t}\mu_{\mathcal{S}}a_{\mathcal{S}}, \tag{38}$$

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

$$\mu_{\mathcal{S}} = \sum_{l=1}^{L-s_1+1} (s_1-1)(U_{L-l+1}-U_{L-l+2}) + \sum_{l=L-s_1+2}^{L} (L-l+1)(U_{L-l+1}-U_{L-l+2}) \quad (39)$$

$$= \sum_{l=1}^{s_1-1} U_l. \quad (40)$$

The coded subfiles are decoded successively at the caches starting with the subfile with the largest size. That is, the multicast signal defined in (33) 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

$$\lambda_{\mathcal{S}}^{(s_i)} = \lambda_{\mathcal{S}}^{(s_{i-1})} + \sum_{l=L-s_i+2}^{L-s_{i-1}} (L-l-s_{i-1}+1)(U_{L-l+1}-U_{L-l+2}), \quad (41)$$

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.

$$\min_{a_{\mathcal{S}}\geq 0} \quad \sum_{t=0}^{L-1} \sum_{\mathcal{S}\subset[L]:|\mathcal{S}|=t} \mu_{\mathcal{S}} a_{\mathcal{S}} \quad (42a)$$

$$\text{subject to} \quad \sum_{\mathcal{S}\subset[L]} a_{\mathcal{S}} = 1, \quad (42b)$$

$$\sum_{\mathcal{S}\subset[L]:l\in\mathcal{S}} (N-\lambda_{\mathcal{S}}^{(s)}) a_{\mathcal{S}} \leq M, \forall l\in[L] \quad (42c)$$

$$\text{and } (35), (36).$$

(42b) and (42c) above represent all feasible choices for the subfile sizes which satisfy the cache size constraints.

In Fig. 6, 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.

VI. Conclusion

In this paper, we have studied a cache-aided network with L cache memories and K end-users, where $L\leq K$. The end users are divided into L groups, and the users from each group have direct access to only one of the L cache memories. We have proposed a coded placement scheme that outperforms the best uncoded placement scheme [17]. In the placement phase, the proposed scheme stores both coded and uncoded data at the caches taking into consideration the users connectivity pattern. For a two-cache system, we have provided an explicit characterization of the gain from coded placement. Next, we have extended our scheme to L-cache systems, where the optimal parameters for the caching scheme are obtained by solving a linear program. We have shown that coded placement exploits the asymmetry in the users' connectivity and the coded placement gain increases with the heterogeneity in the system.

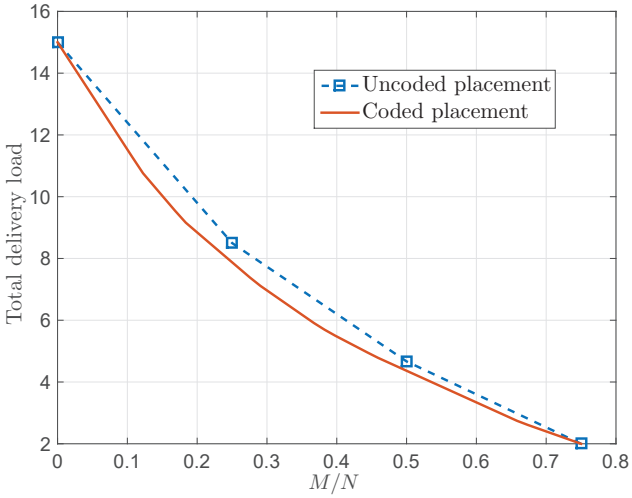

Fig. 6: The achievable delivery load for $L=4$ and $N=15$.

References

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

[2] A. Sengupta, R. Tandon, and T. C. Clancy, "Layered caching for heterogeneous storage," in *Proc. IEEE Asilomar*, 2016.

[3] A. M. Ibrahim, A. A. Zewail, and A. Yener, "Centralized coded caching with heterogeneous cache sizes," in *Proc. IEEE WCNC*, 2017.

[4] ——, "Coded caching for heterogeneous systems: An optimization prespective," *arXiv:1810.08187*, 2018.

[5] ——, "Optimization of heterogeneous caching systems with rate limited links," in *Proc. IEEE ICC*, 2017.

[6] C. Tian and K. Zhang, "From uncoded prefetching to coded prefetching in coded caching," *arXiv:1704.07901*, 2017.

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

[8] A. A. Zewail and A. Yener, "Combination networks with or without secrecy constraints: The impact of caching relays," *IEEE Journal on Selected Areas in Communications*, vol. 36, no. 6, pp. 1140–1152, Jun. 2018.

[9] 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.

[10] J. Hachem, N. Karamchandani, and S. N. Diggavi, "Coded caching for multi-level popularity and access," *IEEE Trans. Inf. Theory*, vol. 63, no. 5, pp. 3108–3141, 2017.

[11] 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, 2017.

[12] Y.-P. Wei and S. Ulukus, "Coded caching with multiple file requests," in *Proc. IEEE Allerton*. IEEE, 2017, pp. 437–442.

[13] S. S. Bidokhti, M. Wigger, and A. Yener, "Benefits of cache assignment on degraded broadcast channels," *arXiv:1702.08044*, 2017.

[14] H. Xu, C. Gong, and X. Wang, "Efficient file delivery for coded prefetching in shared cache networks with multiple requests per user," *arXiv:1803.09408*, 2018.

[15] D. Cao, D. Zhang, P. Chen, N. Liu, W. Kang, and D. Gündüz, "Coded caching with heterogeneous cache sizes and link qualities: The two-user case," *arXiv:1802.02706*, 2018.

[16] A. M. Ibrahim, A. A. Zewail, and A. Yener, "Benefits of coded placement for networks with heterogeneous cache sizes," in *Proc. IEEE Asilomar*, 2018.

[17] E. Parrinello, A. Ünsal, and P. Elia, "Coded caching with shared caches: Fundamental limits with uncoded prefetching," *arXiv:1809.09422*, 2018.

[18] F. J. MacWilliams and N. J. A. Sloane, *The theory of error-correcting codes*. Elsevier, 1977.