Coded Caching with Heterogeneous Cache Sizes and Link Qualities: The Two-User Case

Daming Cao*, Deyao Zhang†, Pengyao Chen†, Nan Liu†, Wei Kang*, Deniz Gündüz‡
*School of Information Science and Engineering, Southeast University, Nanjing, China, {dmcao,wkang}@seu.edu.cn
†National Mobile Communications Research Laboratory, Southeast University, China, {dyzhang, chenpy, nanliu}@seu.edu.cn
‡Imperial College London, London SW7 2AZ, U.K., d.gunduz@imperial.ac.uk

***Abstract*—The centralized coded caching problem is studied under heterogeneous cache sizes and channel qualities from the server to the users, focusing on the two-user case. A server holding N files is considered to be serving two users with arbitrary cache capacities of M_1 and M_2, and it is assumed that in addition to a shared common link, each user also has a private link from the server available during the delivery phase. Optimal caching and delivery strategies that minimize the worst-case delivery latency are presented for an arbitrary N. The converse proof benefits from Tian's observation that it suffices to consider file-index symmetric caching schemes, while the achievability is obtained through memory-sharing among certain special (M_1, M_2) pairs. The optimal scheme is shown to exploit the private link capacities by transmitting part of the corresponding user's request in an uncoded fashion. When there are no private links, the results presented here improve upon the two known results in the literature, namely, i) equal cache capacities and arbitrary number of files; and ii) unequal cache capacities and $N = 2$ files.**

I. Introduction

In their seminal paper [1], Maddah-Ali and Niesen proposed a framework for coded caching and delivery to better exploit the cache memories available at user devices to relieve the traffic burden at peak times. They considered a server holding N files of equal size, serving K users, each equipped with a cache capable of storing M files. Users' caches are filled before they reveal their demands, called the *placement phase*, over a low-traffic period. In the ensuing *delivery phase*, each user requests a single file from the library, which are delivered simultaneously over an error-free shared link. The coded caching scheme proposed in [1] creates *multicasting* opportunities by jointly designing the content placement and delivery, resulting in a global caching gain.

The optimal caching and delivery scheme for the general coded caching problem remains open despite ongoing research efforts. While various achievability schemes have been proposed in [2]–[8], and converse results are presented in [1], [9]–[13], the bounds obtained do not match in general except in some special cases, i.e., $N = K = 2$ [1], $N = 2$ and arbitrary K [12], $N = 3$ and $K = 2$ [12]. Recently, the optimal caching and delivery strategy is characterized in [9] when the cache placement is constrained to be *uncoded*.

Due to the difficulty of the problem, most of the literature follows the symmetric setting of [1], in which all the users are equipped with the same cache size, and the link between the server and the users is a common shared link. However, in practice, owing to the heterogeneous nature of devices, the equal cache assumption is often not realistic. Furthermore, the delivery channel quality may be different for different users, and limiting the model to a single common shared link is equivalent to considering the worst channel quality. Heterogeneous cache sizes with a shared common link has been considered in [14]–[16], heterogeneous link qualities has been considered in [17], [18], while a few works have studied heterogeneity in *both* the cache sizes and link qualities [19]–[23]. References [20]–[24] consider cache allocation among users with different delivery channel qualities, where it is shown that a general rule of thumb is to assign more cache to users with weaker links. We note that, in practice, the cache capacity cannot be distributed across user devices dynamically, but rather given as a parameter of the problem. For example, a cellphone with a weak link to the server probably will not have a larger cache than a laptop with a stronger link. Hence, in this work, we assume that both the cache sizes and the link qualities are given, and we aim to find the best caching and delivery strategy that minimizes the worst-case delivery latency.

To model the heterogeneous link qualities of K users, we consider orthogonal common and private links from the server to the users. For example, for $K = 2$ users, the multicast rate tuple is (R_c, R_{p1}, R_{p2}), where R_c is the rate of the common message that can be reliably transmitted to both users, and R_{pk} is the communication rate of the private message to User k, $k = 1, 2$. In practice, this might model a scenario in which there are orthogonal error-free finite-capacity channels for each subset of users, either because a frequency band is allocated for every subset of users, or because the underlying physical layer coding and modulation schemes that dictate these rates are fixed, and the coded caching scheme is implemented in the MAC layer.

Given the cache sizes $(M_1, M_2, \ldots, M_K)$, and the multicast rate tuple $(R_{\mathcal{D}})_{\mathcal{D}\subseteq_\phi\{1,2,\ldots,K\}}$ for the delivery phase, we are interested in finding the optimal *centralized* caching and delivery

This work was partially supported by the National Natural Science Foundation of China under Grants 61571123, 61571122, and 61521061, Six talent peaks project in Jiangsu Province, the Research Fund of National Mobile Communications Research Laboratory, Southeast University (No. 2018A03). The work of D. Gunduz was supported by the European Research Council (ERC) through Starting Grant BEACON (agreement 677854).

strategy that minimizes the worst-case delivery latency across all possible demand combinations. The optimal strategy will show us how to best utilize the heterogeneous caches at the users, and what to transmit over the shared and private links for the most efficient use of the communication resources.

In this paper, we focus on the special case of $K = 2$ users, while the number of files, N, is arbitrary. We characterize the optimal cache and delivery strategy for a generic scenario defined with five parameters $(M_1, M_2, R_c, R_{p1}, R_{p2})$. While the achievability is obtained by memory-sharing among certain special points, the converse proof benefits from Tian's observation that it suffices to consider *file-index symmetric* caching schemes [12]. Our main contributions are:

1) For $K = 2$ users with heterogeneous caches and one shared common link only, we identify the optimal cache and delivery strategy for any $N \geq 3$ files. Previously only the case of $M_1 = M_2, N \geq 2$ [12], and $M_1 \neq M_2$, and $N = 2$ [25] cases were solved.
2) For the general case of $K = 2$ users with one common and two private links, we find the optimal caching and delivery strategy for $N \geq 2$ files: i) the private links are used to transmit part of the requested files in an uncoded fashion; ii) when part of the file is required by one user only, either that part of all the files should be cached entirely in the said user's cache, or that part of the requested file should be transmitted over the shared common link in an uncoded fashion.
3) By identifying the parallels between the coded caching problem with one common and two private links studied here, and the coded caching problem with heterogeneous distortion requirements studied in [25], for the case of $K = 2$ users with heterogeneous caches, we prove the optimal caching and delivery strategy also for that problem for $N \geq 3$ files. In [25], the optimal cache and delivery strategy is characterized for $N = 2$.

II. System Model

We consider a coded caching problem with one server connected to $K = 2$ users. The server has access to a database of N independent files of equal size F, which are denoted by $W_1, W_2, \ldots, W_N$. The system operates in two phases. In the placement phase, the users are given access to the entire database and fill their caches in an error-free manner. The normalized cache capacities at the two users are given by M_1 and M_2, respectively, and the content of the caches after the placement phase are denoted by Z_1 and Z_2. In the delivery phase, each user requests a single file from the server, where d_k denotes the index of the file requested by User k, $k = 1, 2$. After receiving the demands $D \triangleq (d_1, d_2)$, the server transmits messages over the available channels to the two users to satisfy their demands.

In most of the previous literature, the delivery channel is modeled as an error-free shared common link of limited capacity. However, in practice, the channels between the server and the users are typically of different quality. Thus, in this work, we model the delivery channel as consisting of two private error-free links with capacities R_{p1} and R_{p2} to User 1 and User 2, respectively, in addition to an error-free shared common link of capacity R_c.

A *caching and delivery code* for this system consists of[1]

1) two caching functions
$$\phi_k : [2^{nF}]^N \to [2^{nM_kF}], \quad k = 1, 2,$$
which map the database into cache contents of the users, denoted by $Z_k = \phi_k(W_1, W_2, \ldots, W_N)$, $k = 1, 2$.
2) N^2 encoding functions
$$f^D : [2^{nF}]^N \to [2^{nr_c^D F}] \times [2^{nr_{p1}^D F}] \times [2^{nr_{p2}^D F}],$$
that map the files into the messages transmitted over the shared common link, private link to User 1 and private link to User 2, denoted by X_c^D (with rate r_c^D), X_{p1}^D (with rate r_{p1}^D), and X_{p2}^D (with rate r_{p2}^D), respectively, i.e., $(X_c^D, X_{p1}^D, X_{p2}^D) \triangleq f^D(W_1, W_2, \ldots, W_N)$.
3) $2N^2$ decoding functions
$$g_k^D : [2^{nM_kF}] \times [2^{nr_c^D F}] \times [2^{nr_{pk}^D F}] \to [2^{nF}], k = 1, 2,$$
which decodes the desired file W_{d_k} at User k from the cached content at User k, the message on the shared common link and the message on the private link to User k, $k = 1, 2$.

The performance metric we are interested in is the worst-case delivery time, which is defined as $T = \max_D T^D$, and $T^D \triangleq \max\{T_c^D, T_{p1}^D, T_{p2}^D\}$, where $T_c^D \triangleq \frac{r_c^D}{R_c}$ and $T_{pk}^D \triangleq \frac{r_{pk}^D}{R_{pk}}$, $k = 1, 2$. In other words, T^D is the time, under demand D, it takes for X_c^D to be received at both users, and X_{pk}^D to be received at User k, $k = 1, 2$.

We provide a discussion on the worst case as follows. Similarly to the proof of [12, Proposition 1], we have the following lemma for the above problem.

Lemma 1: For any caching and delivery code, there exists a file-index-symmetric caching and delivery code with an equal or smaller worst-case delivery time.

File-index-symmetric codes are codes for which the permutations of the file index set does not lead to variations in the entropies [12]. Hence, Lemma 1 states that, it suffices to consider only file-index-symmetric caching and delivery codes. File-index-symmetric caching and delivery codes have the following property: for any pair of distinct demands (d_1, d_2), i.e., $d_1 \neq d_2$, $(r_c^D, r_{p1}^D, r_{p2}^D)$ takes the same value, denoted by (r_c, r_{p1}, r_{p2}); similarly, for all the cases in which the two users demand the same file, i.e., $d_1 = d_2$, $(r_c^D, r_{p1}^D, r_{p2}^D)$ takes the same value, denoted by $(r_c^0, r_{p1}^0, r_{p2}^0)$. We are interested in the worst-case performance; hence, for the rest of the paper, we will assume $d_1 \neq d_2$. Hence, we have

$$T = \max\left\{\frac{r_c}{R_c}, \frac{r_{p1}}{R_{p1}}, \frac{r_{p2}}{R_{p2}}\right\}. \tag{1}$$

We seek the minimum achievable worst-case delivery latency $T^*(M_1, M_2, R_c, R_{p1}, R_{p2})$ across all caching and

[1]For $X \in \mathbb{R}^+$, $[X]$ denotes the set $\{1, \ldots, \lceil X \rceil\}$.

delivery codes, where $(M_1, M_2, R_c, R_{p1}, R_{p2})$ are the parameters of the problem. We denote this problem by $\mathcal{Q}(M_1, M_2, R_c, R_{p1}, R_{p2})$. Note that for the problem of shared common link only, i.e., $\mathcal{Q}(M_1, M_2, R_c, 0, 0)$, the rate R_c is of no significance as $r_c = T \times R_c$. So rather than minimizing T given R_c, we would just like to minimize the data rate on the shared common link, i.e., r_c. As a result, we denote the problem $\mathcal{Q}(M_1, M_2, R_c, 0, 0)$ shortly by $\mathcal{Q}^c(M_1, M_2)$, and the minimal achievable data rate on the shared common link is denoted by $r_c^*(M_1, M_2)$.

To further simplify the notation, for the rest of the paper, we consider a normalized file size of $F = 1$.

III. Shared Link Problem: $\mathcal{Q}^c(M_1, M_2)$

We start by studying the case with heterogenous cache sizes and a shared common link only, i.e., the problem $\mathcal{Q}^c(M_1, M_2)$. For this problem, we would like to minimize the data rate on the shared common link, i.e., $r_c^*(M_1, M_2)$.

In the case of $K = N = 2$, the problem has been solved in [25]. Note that [25] studies the problem with heterogeneous cache sizes and distortion requirements. Thus, if we consider the special case of the problem studied in [25], in which the distortion requirements of the two users are the same, we obtain the problem $\mathcal{Q}^c(M_1, M_2)$, and [25, Corollary 1] provides the minimum delivery rate.

In the case of $K = 2$ and $N \geq 3$, we provide the following optimal data rate on the shared common link, which was previously unknown.

Theorem 1: In the cache and delivery problem $\mathcal{Q}^c(M_1, M_2)$, when $N \geq 3$, we have:

$$r_c^*(M_1, M_2) = \max\Bigg\{1 - \frac{M_1}{N}, 1 - \frac{M_2}{N}, 2 - \frac{3M_1}{N} - \frac{M_2 - M_1}{N-1}, 2 - \frac{3M_2}{N} - \frac{M_1 - M_2}{N-1}\Bigg\}. \quad (2)$$

In the special case of $M_1 = M_2 = M$, the problem $\mathcal{Q}^c(M, M)$ has been solved in [12], where the achievability follows from [1], while the converse proof utilizes the symmetry of optimal codes. We provide the proofs of the converse and the achievability below for the general case.

A. The converse proof of Theorem 1

The first two terms of (2) follow from the cut-set bound [1]. The third and fourth terms follow from the following lemma.

Lemma 2: In the $\mathcal{Q}^c(M_1, M_2)$ problem with $N \geq 3$, the common delivery rate r_c of any achievable scheme satisfies

$$NM_1 + (2N-3)M_2 + N(N-1)r_c \geq 2N(N-1),$$
$$NM_2 + (2N-3)M_1 + N(N-1)r_c \geq 2N(N-1).$$

The details of the proof of Lemma 2 is given in [26]. In the following, we comment on some of the proof ideas. First of all, the proof of Lemma 2 is simplified by Lemma 1. Secondly, it is proved by two major steps stated in the following two lemmas.

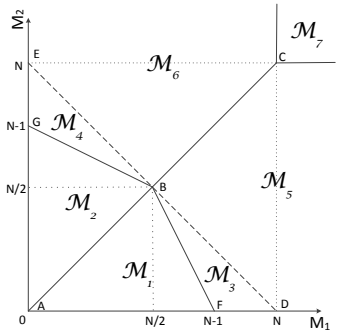

Fig. 1. The optimal tradeoff between $r_c^*(M_1, M_2, R_c, 0, 0)$ and (M_1, M_2) with $N \geq 3$

Lemma 3: In the caching problem $\mathcal{Q}^c(M_1, M_2)$, for file-index-symmetric caching schemes, we have:

$$H(X_c^{(1,2)}|Z_1, W_1) \geq 1 - \frac{1}{N-1}[H(Z_1|W_1) + H(Z_2|W_1)],$$
$$H(X_c^{(2,1)}|Z_2, W_1) \geq 1 - \frac{1}{N-1}[H(Z_1|W_1) + H(Z_2|W_1)].$$

Lemma 4: For file-index symmetric codes, we have

$$NH(Z_1|W_1) \geq (N-1)H(Z_1),$$
$$NH(Z_2|W_1) \geq (N-1)H(Z_2).$$

Please note that Lemma 4 is true for any file-index symmetric caching scheme, irrespective of the problem, i.e., it works for the more general problem of $\mathcal{Q}(M_1, M_2, R_c, R_{p1}, R_{p2})$.

As can be seen, Lemma 3 provides a way to lower bound complicated terms such as $H(X_c^{(1,2)}|Z_1, W_1)$ with simpler terms such as $H(Z_1|W_1)$, and Lemma 4 further lower bounds terms such as $H(Z_1|W_1)$ with even simpler terms such as $H(Z_1)$, which is equal to the size of the cache of User 1, i.e., M_1. Hence, the main aim of the two lemmas is to provide a lower bound that only depends on the placement scheme, and is independent of the delivery scheme. The same idea appeared in [13, Lemma 1]. The proofs of Lemmas 3 and 4 are provided in [26]. The converse of Theorem 1 can be proved using Lemma 2.

B. The achievability proof for Theorem 1

In Figure 1, we show the 2-dimensional plane of possible (M_1, M_2) values. For the following points on this figure, the minimum data rate on the shared common link, r_c^*, is known: i) at Point A, $(M_1, M_2) = (0, 0)$ and we have $r_c^* = 2$; ii) at Point B: $(M_1, M_2) = (\frac{N}{2}, \frac{N}{2}$, and $r_c^* = \frac{1}{2})$; iii) at Point C: $(M_1, M_2) = (N, N)$ and $r_c^* = 0$; iv) at Point D: $(M_1, M_2) = (N, 0)$, and $r_c^* = 1$; v) at Point E: $(M_1, M_2) = (0, N)$ and $r_c^* = 1$.

We now provide the achievability scheme for Point F, i.e., $(M_1, M_2) = (N-1, 0)$ and $r_c^* = 1$, and the scheme for Point G follows by symmetry. A similar caching and delivery scheme was also used in [7].

- **Placement phase**: User 1 fills its cache with the modulo sum of every two label-adjacent files, i.e. $Z_1 = \{W_1 \oplus W_2, W_2 \oplus W_3, \ldots, W_{N-1} \oplus W_N\}$.

- **Delivery phase**: The server transmit $X_c^{(d_1,d_2)} = \{W_{d_2}\}$. Therefore, User 2 can directly get W_{d_2}, while user 1 can decode W_{d_1} with the help of its cache contents by successive cancellation. For example, if $(d_1, d_2) = (1, 4)$, we have $X_c^{(1,4)} = W_4$. User 1 can first recover W_3 from $W_3 \oplus W_4$, then W_2 from $W_2 \oplus W_3$, and finally, it recovers its requested file W_1 from $W_1 \oplus W_2$ in a successive manner.

By performing memory-sharing [1], [25], [27] among these 7 points, we can obtain the following achievable data rate on the shared common link:

$$r_c(M_1, M_2) = \begin{cases} 2 - \frac{3M_2}{N} - \frac{M_1 - M_2}{N-1} & (M_1, M_2) \in \mathcal{M}_1 \\ 2 - \frac{3M_1}{N} - \frac{M_2 - M_1}{N-1} & (M_1, M_2) \in \mathcal{M}_2 \\ 1 - \frac{M_2}{N} & (M_1, M_2) \in \mathcal{M}_3, \mathcal{M}_5 \\ 1 - \frac{M_1}{N} & (M_1, M_2) \in \mathcal{M}_4, \mathcal{M}_6 \end{cases}.$$

This completes the achievability proof of Theorem 1.

C. Comparison with known results

As we mentioned before, for the problem $\mathcal{Q}^c(M_1, M_2)$, in the case of $N = 2$, the problem has been solved in [25]. But for $N \geq 3$, the best known achievability schemes [25, Section III-C], [27] perform memory sharing between the five points of Fig. 1, i.e., Point A to Point E. Since we performed memory-sharing with two more points, i.e., Points F and G, we obtain a strictly lower delivery rate than [25, Section III-C].

As for the converse, when $N \geq 3$, the best known converse to date is given in [25, Lemma 1], when we set $r_1 = r_2 = 1$, which is the minimum of the five terms

$$r_c(M_1, M_2) \geq \max\Bigg\{1 - \frac{M_1}{N}, 1 - \frac{M_2}{N}, 2 - \frac{M_1 + M_2}{\lfloor N/2 \rfloor}, \frac{3}{2} - \frac{M_1 + M_2}{2\lfloor N/2 \rfloor}, 2 - \frac{M_1 + M_2}{2\lfloor N/3 \rfloor}\Bigg\},$$

where the first two terms follow from the cut-set bound, the third and fourth terms follow from the straightforward generalization of the proof when $N \geq 2$, and in the proof of the last term, the step [25, (40c)] may be loose because the content of two caches may not be independent even conditioned on the knowledge of some files. We transform terms like $H(X_{i,j}, Z_k|W_i)$ into $H(X_{i,j}|Z_k, W_i)$ and $H(Z_k|W_i)$, and then bound these two terms via Lemmas 3 and 4 to obtain a tighter converse.

IV. General problem: $\mathcal{Q}(M_1, M_2, R_c, R_{p1}, R_{p2})$

In this section, we study the general problem $\mathcal{Q}(M_1, M_2, R_c, R_{p1}, R_{p2})$, i.e. the problem with one shared common link and two private links, one for each user. We characterize the optimal delivery latency $T^*(M_1, M_2, R_c, R_{p1}, R_{p2})$ in the following theorem.

Theorem 2: In the problem $\mathcal{Q}(M_1, M_2, R_c, R_{p1}, R_{p2})$ with $N = 2$, we have:

$$T^* = \max\Bigg\{\frac{1 - \frac{M_1}{2}}{R_c + R_{p1}}, \frac{1 - \frac{M_2}{2}}{R_c + R_{p2}}, \frac{2 - M_1 - M_2}{R_c + R_{p1} + R_{p2}}, \frac{3 - M_1 - M_2}{2(R_c + R_{p2}) + R_{p1}}, \frac{3 - M_1 - M_2}{2(R_c + R_{p1}) + R_{p2}}\Bigg\},$$

while, for $N \geq 3$, we have:

$$T^* = \max\Bigg\{\frac{1 - \frac{M_1}{N}}{R_c + R_{p1}}, \frac{1 - \frac{M_2}{N}}{R_c + R_{p2}}, \frac{2 - \frac{3M_2}{N} - \frac{M_1 - M_2}{N-1}}{R_c + R_{p1} + R_{p2}}, \frac{2 - \frac{3M_1}{N} - \frac{M_2 - M_1}{N-1}}{R_c + R_{p1} + R_{p2}}, \frac{N(2N-1) - 2(N-1)M_1 - NM_2}{N^2(R_c + R_{p2}) + N(N-1)R_{p1}}, \frac{N(2N-1) - 2(N-1)M_2 - NM_1}{N^2(R_c + R_{p1}) + N(N-1)R_{p2}}\Bigg\}.$$

A. Converse Proof of Theorem 2

The converse proof is a generalization of the one for $\mathcal{Q}^c(M_1, M_2)$, and can be found in [26]. We note here that in dealing with the maximum as in the definition of delivery latency in (1), we used the fact that for positive numbers a, b, c, d, α, we have $\max\left\{\frac{a}{b}, \frac{c}{d}\right\} \geq \frac{a + \alpha c}{b + \alpha d}$.

B. Achievability proof of Theorem 2

The private links: the private link to User k, $k = 1, 2$, is used to transmit part of the desired message W_{d_k} uncoded. The joint caching and common link delivery strategy is designed as if the file size is reduced by the amount of data transmitted over the private link. For example, in the case of $r_{p1} \geq r_{p2}$, we split each file into three parts W_i^c, W_i^{p1} and $W_i^{p12}, i = 1, \ldots, N$, with sizes of $l_1, l_2 - l_1, 1 - l_2$ bits, respectively, where $l_1 \triangleq 1 - r_{p1}$, $l_2 \triangleq 1 - r_{p2}$. In the delivery phase, the server transmits $\{W_{d_1}^{p1}, W_{d_1}^{p12}\}$ and $W_{d_2}^{p12}$ to Users 1 and 2 respectively via the private links. Thus, we only need to design the caching and common link delivery strategy for the recovery of $(W_{d_1}^c, W_{d_2}^c)$ for Users 1 and 2, and the recovery of $W_{d_2}^{p1}$ for User 2 only.

How to deal with the part of the file that is requested by one user only, i.e., $\{W_1^{p1}, W_2^{p1}, \ldots, W_N^{p1}\}$: Memory-sharing is performed based on certain special achievable points. In each point, the achievable scheme is to either transmit $W_{d_2}^{p1}$ uncoded through the shared common link, or cache all files $\{W_1^{p1}, W_2^{p1}, \ldots, W_N^{p1}\}$ (of size $l_2 - l_1$ bits each) in the cache of User 2. The caching and delivery strategy over the common shared link for files $\{W_1^c, W_2^c, \ldots, W_N^c\}$ (of file size l_1 each) is the same as the one proposed for the $\mathcal{Q}^c(M_1, M_2)$ problem studied in the previous section.

The details of the proof, i.e., i) the memory-sharing among special points; and ii) obtaining the minimum delivery latency T^* based on parameters $(M_1, M_2, R_c, R_{p1}, R_{p2})$ can be found in [26].

Remark: In [25] the authors study the caching problem in which the users request different quality descriptions of the files in the common database, due to, for example, different processing or display capabilities. For given distortion targets of (D_1, D_2) for two users, assuming $D_1 \geq D_2$ without loss of generality, the authors suggest using scalable coding [28] of the files in the library at rates (r_1, r_2), such that the *base layer* of r_1 bits allows the first receiver to obtain

an average reconstruction distortion of D_1, while the base layer together with the *refinement layer* of r_2 bits allows an average reconstruction distortion of D_2 at the second receiver. This successive coding scheme is known to be rate-distortion optimal if the underlying sources are Gaussian distributed and the distortion measure is squared error between the original samples and the reconstruction at the receivers.

We note that, once we specify the way the private links are used, the (l_1, l_2) parameters in our problem correspond to (r_1, r_2) in the achievable scheme of [25], where r_1 corresponds to the number of bits transmitted over the common link, while $r_2 - r_1$ corresponds to the number of bits transmitted over the private links to the user that request a higher quality description. As such, we may make a comparison of the achievable scheme proposed here and that in [25] for $K = 2$ user with $N \geq 3$ files. The achievable scheme in [25] is a memory-sharing scheme between the points A, B, B', C', D, E', ignoring the three points G, G' and F, which are included in the achievable scheme proposed here. Hence, the achievable scheme in [25] is suboptimal in general for $N \geq 3$ files. In fact, we can show that the memory-sharing scheme among the nine points, A, B, B', C', D, E', G, G' and F is optimal for the coded caching problem with heterogeneous distortion requirements studied in [25] for $K = 2$ users. The details can be found in the longer version of this paper [26].

V. Conclusions

We have studied the problem of centralized coded caching for two users with different cache capacities, where, in addition to the shared common link, each user also has a private link from the server. We have characterized the optimal caching and delivery strategies for any number of files in the library. In the case of a shared common link only, we have improved upon known results in the literature by proposing a new achievable scheme for a special (M_1, M_2) pair, and performing memory-sharing among a total of 9 special memory pairs. In the case of two private links in addition to the shared common link, we have showed that it is optimal to use all the capacity available in the private link to each user to transmit part of its requested file in an uncoded fashion.

A connection between the problem of coded caching with a private link to each user considered here and that of coded caching with heterogeneous distortion requirements studied in [25] has also been established, which allowed us extending the proposed results to improve the state of the art in the latter problem as well.

References

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

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

[3] M. Ji, A. M. Tulino, J. Llorca, and G. Caire, "Caching and coded multicasting: Multiple groupcast index coding," in *IEEE Global Conference on Signal and Information Processing (GlobalSIP),*, 2014, pp. 881–885.

[4] K. Shanmugam, M. Ji, A. M. Tulino, J. Llorca, and A. G. Dimakis, "Finite-length analysis of caching-aided coded multicasting," *IEEE Transactions on Information Theory*, vol. 62, no. 10, pp. 5524–5537, 2016.

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

[6] C. Tian and J. Chen, "Caching and delivery via interference elimination," in *Information Theory (ISIT), 2016 IEEE International Symposium on*, 2016, pp. 830–834.

[7] M. M. Amiri, Q. Yang, and D. Gündüz, "Coded caching for a large number of users," in *IEEE Information Theory Workshop (ITW),*, 2016, pp. 171–175.

[8] M. Mohammadi Amiri and D. Gunduz, "Fundamental limits of coded caching: improved delivery rate-cache capacity trade-off," *IEEE Transactions on Communications*, vol. 65, no. 2, pp. 806–815, 2017.

[9] Q. Yu, M. A. Maddah-Ali, and A. S. Avestimehr, "The exact rate-memory tradeoff for caching with uncoded prefetching," in *Information Theory (ISIT), 2017 IEEE International Symposium on*. IEEE, 2017, pp. 1613–1617.

[10] A. Sengupta, R. Tandon, and T. C. Clancy, "Improved approximation of storage-rate tradeoff for caching via new outer bounds," in *Information Theory (ISIT), 2015 IEEE International Symposium on*. IEEE, 2015, pp. 1691–1695.

[11] H. Ghasemi and A. Ramamoorthy, "Improved lower bounds for coded caching," *IEEE Transactions on Information Theory*, 2017.

[12] C. Tian, "Symmetry, demand types and outer bounds in caching systems," in *IEEE International Symposium on Information Theory (ISIT),*, 2016, pp. 825–829.

[13] Q. Yu, M. A. Maddah-Ali, and A. S. Avestimehr, "Characterizing the rate-memory tradeoff in cache networks within a factor of 2," *arXiv preprint arXiv:1702.04563*, 2017.

[14] S. Wang, W. Li, X. Tian, and H. Liu, "Coded caching with heterogenous cache sizes," *arXiv:1504.01123 [cs.IT]*, 2015.

[15] M. M. Amiri, Q. Yang, and D. Gündüz, "Decentralized caching and coded delivery with distinct cache capacities," *IEEE Transactions on Communications*, vol. 65, no. 11, pp. 4657 – 4669, 2017.

[16] A. M. Ibrahim, A. A. Zewail, and A. Yener, "Centralized coded caching with heterogeneous cache sizes," in *Wireless Communications and Networking Conference (WCNC), 2017 IEEE*. IEEE, 2017, pp. 1–6.

[17] J. Zhang and P. Elia, "Wireless coded caching: A topological perspective," in *IEEE Int'l Symp. on Inform. Theory (ISIT)*, June 2017, pp. 401–405.

[18] N. Naderializadeh, M. A. Maddah-Ali, and A. S. Avestimehr, "On the optimality of separation between caching and delivery in general cache networks," *arXiv preprint arXiv:1701.05881*, 2017.

[19] A. Ghorbel, M. Kobayashi, and S. Yang, "Content delivery in erasure broadcast channels with cache and feedback," *IEEE Transactions on Information Theory*, vol. 62, no. 11, pp. 6407–6422, 2016.

[20] S. S. Bidokhti, M. Wigger, and R. Timo, "Erasure broadcast networks with receiver caching," in *Information Theory (ISIT), 2016 IEEE International Symposium on*. IEEE, 2016, pp. 1819–1823.

[21] M. Mohammadi Amiri and D. Gunduz, "Cache-aided content delivery over erasure broadcast channels," *IEEE Transactions on Communications*, vol. 66, no. 1, pp. 370–381, 2018.

[22] S. S. Bidokhti, M. Wigger, and R. Timo, "Noisy broadcast networks with receiver caching," *arXiv preprint arXiv:1605.02317*, 2016.

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

[24] M. Mohammadi Amiri and D. Gunduz, "Caching and coded delivery over gaussian broadcast channels for energy efficiency," *to appear, IEEE Journal on Selected Areas in Communications*, 2018.

[25] Q. Yang and D. Gündüz, "Coded caching and content delivery with heterogeneous distortion requirements," *IEEE Transactions on Information Theory*, 2018.

[26] 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 preprint arXiv:1802.02706*, 2018.

[27] A. Sengupta, R. Tandon, and T. C. Clanc, "Layered caching for heterogeneous storage," in *Signals, Systems and Computers, 2016 50th Asilomar Conference on*. IEEE, 2016, pp. 719–723.

[28] T. M. Cover and W. H. Equitz, "Successive refinement of information," *IEEE Trans. Inform. Theory*, vol. 37, no. 2, pp. 269–275, 1991.