

Multi-Access Coded Caching Schemes from Maximal Cross Resolvable Designs

Niladri Das and B. Sundar Rajan, *Fellow, IEEE*

Abstract

We study the problem of multi-access coded caching (MACC): a central server has N files, K ($K \leq N$) caches each of which stores M out of the N files, K users each of which demands one out of the N files, and each user accesses z caches. The objective is to jointly design the placement, delivery, and user-to-cache association, to optimize the achievable rate. This problem has been extensively studied in the literature under the assumption that a user accesses only one cache. However, when a user accesses more caches, this problem has been studied only under the assumption that a user accesses z consecutive caches with a cyclic wrap-around over the boundaries. A natural question is how other user-to-cache associations fare against the cyclic wrap-around user-to-cache association. A bipartite graph can describe a general user-to-cache association. We identify a class of bipartite graphs that, when used as a user-to-cache association, achieves either a lesser rate or a lesser subpacketization than all other existing MACC schemes using a cyclic wrap-around user-to-cache association. The placement and delivery strategy of our MACC scheme is constructed using a combinatorial structure called maximal cross resolvable design.

I. INTRODUCTION

Consider the following scenario. A cellular cell is deployed with a base station (BS), and several access points (APs) are distributed within the cell. APs typically have a small coverage area than BS, cater to fewer users than BS, and may have a wire-line connection to the internet backbone. Each AP is served with memory and acts as a cache. This paper uses the word AP and cache alternatively, indicating the same device functionality.

A large number of files are stored with the BS. There are users in the cell connected (wireless) to the BS. A user may request to download files from the BS. Each user is also connected to one or more APs. The APs fetch data (before knowing the users' demands) from the BSs when internet traffic is low – this is called the cache placement phase. This cache filling strategy is centralized, meaning that the BS decides which cache stores which parts of the files. Furthermore, the cache filling strategy is uncoded, meaning that the data placed at the APs are copies of segments of the files present at the BS.

Manuscript received on date xx; revised on date xx; accepted on date xx. Date of publication xx.; date of current version xx. The editor coordinating the review of this paper and approving it for publication was Prof. xx.

Niladri Das and B. Sundar Rajan are with the Department of Electrical Communication Engineering, Indian Institute of Science, Bangalore, India e-mail: niladridas@iisc.ac.in and bsrajan@iisc.ac.in.

This work was supported partly by the Science and Engineering Research Board (SERB) of Department of Science and Technology (DST), Government of India, through J.C. Bose National Fellowship to B. Sundar Rajan, and through SERB-National Post Doctoral Fellowship (No. PDF/2019/000044) to Niladri Das.

User-to-cache association is a list that specifies which users access which caches. The central server may decide the user-to-cache association. The users inform of their demands (a demand is a request for a single file) to the BS. Based on a predetermined delivery strategy, the BS broadcasts a set of files to all users to fulfill their demands of the users. The term delivery load quantifies the number of files broadcast by the BS. The goal is to design the placement and the delivery policy such that the least pressure is put on the internet traffic during peak hours, which amounts to minimizing the number of files broadcast during peak hours. We assume that very high-speed links connect a user to the APs it accesses. Hence, the burden on the internet traffic is due to the files broadcast by the BS. Furthermore, we assume that the number of files is much more than the number of users. Hence, it is beneficial to minimize the delivery load assuming all users demand distinct files.

From the works conducted in the literature towards solving this problem, we see a trade-off between the delivery load and the size of the files. As in, a delivery load may be achievable only if the size of each file is greater than a certain value. Shanmugam *et al.* in [3] established a trade-off between the size of the files and the delivery load. Yan *et al.* in [4] also showed a trade-off between file size and delivery load, albeit for a particular placement. The size of the file is quantified by a term called subpacketization level. If a solution to the problem has a subpacketization level F , the solution requires each file to be split into F subfiles. Furthermore, the higher the subpacketization level, the more computations are necessary to decode at the users. So it is also beneficial to minimize the subpacketization level.

Traditionally, the content has been cached based on the file's popularity. Maddah-Ali and Niesen, in the seminal paper [1], showed that by jointly designing the cache placement phase and the file delivery phase, an additional gain (coding gain) can be achieved. The work of [1] significantly improved the previous caching schemes (which may be referred to as caching schemes with uncoded delivery). Maddah-Ali *et al.* also showed a trade-off between the memory size of the caches and the achievable rate. Furthermore, they showed that the rate achieved by their scheme is within 12 times the information-theoretic optimum value. The same authors provided a decentralized solution to the coded caching problem in [2].

Maddah-Ali and Niesen's caching model inherently assumes that each user accesses only one cache. We call such schemes as dedicated cache coded coding schemes. If a user can connect to several APs available in its vicinity, then restricting the user to connect to only one AP may not be the best choice. Since granting access to multiple APs does not require extra hardware, such a scheme may provide better performance at the cost of more signaling.

Hachem *et al.* in [6] improvised the model (considered in [1]) to include scenarios where a user accesses more than one cache. They called this problem as the multi-access coded caching (MACC) problem. In their model, there are K caches and K users, each user accesses z consecutive caches with a cyclic wrap-around over the boundaries, and each cache is accessed by z caches. We call such a user-to-cache association a cyclic wrap-around user-to-cache association. Hachem *et al.* showed a MACC scheme for any cache memory size M files and access degree z .

Subsequently, all works on MACC (which considers the number of users to be the same as the number of caches) assume a cyclic wrap-around user-to-cache association. However, reference [6] quotes the following: "we assume also for simplicity that the caches are arranged linearly and that users connect to d_i consecutive caches, with a cyclic wrap-around for symmetry." Although we acknowledge that studying a simpler version of a problem can

provide significant insights, until now, to the best of our knowledge, no work has justified the assumption of cyclic wrap-around user-to-cache association with any reason other than simplicity.

A bipartite graph can represent a general user-to-cache association of the MACC problem. It is an open problem to find out which bipartite graph provides the best performance when used as a user-to-cache association. The criteria for best performance may vary with applications—most applications would likely desire to achieve the least delivery load possible for a given upper limit on the subpacketization level.

A. State of the art

Hachem *et al.* were the first to study the MACC problem [6]. They studied the problem assuming a cyclic wrap-around user-to-cache association. Subsequent to the work of Hachem *et al.*, the MACC problem under the assumption of a cyclic wrap-around user-to-cache association has been studied by the following authors in the respective references: Serbetci *et al.* in [7], Reddy *et al.* in [8] and [13], Cheng *et al.* in [9], Sasi *et al.* in [10] and [11], Mahesh *et al.* in [12].

There have been some works in the literature that studied a version of the MACC problem where the number of users is more than the number of caches. Katyal *et al.* in [14] studied a version of a MACC problem where the user-to-cache association is decided by combinatorial structure called cross resolvable design (CRD). Using the CRD, they constructed a MACC scheme for this problem. The authors identified two infinite classes of CRDs from the existing literature in combinatorial designs and showed the performance of the MACC schemes constructed using these CRDs. Subsequent to [14], Muralidhar *et al.* in [15] showed a generalization of the scheme in [14]. In reference [16], Muralidhar *et al.* showed a new class of CRDs and the corresponding MACC schemes constructed using these CRDs.

Muralidhar *et al.* in [17] studied a MACC problem where the user-to-cache association mimics the combination network, which is well-studied in the network coding literature. They showed that a generalization of the MAN scheme provides a solution to this problem. Recently Brunero *et al.* in [18] studied a generalization of the user-to-cache association considered by [17]. They showed a solution and proved that it achieves the optimal delivery load under uncoded placement. The authors of [18] also address a scenario where the server is unaware of the user-to-cache association during cache placement.

It is to be noted that the works of [14]–[18] do not intersect with our work except for the trivial cases, which is when either each user accesses only one cache or each user accesses all caches. For the rest of this document, we limit ourselves to MACC problems where the number of users is the same as caches.

B. Contributions

- 1) We identify a class of bipartite graphs. If the server chooses the user-to-cache association (which can be represented by a bipartite graph) of the MACC problem from this class of bipartite graphs, we show a MACC scheme that provides a solution to the MACC problem. This contribution is shown in Theorem 2 of Section VI.

- 2) We show that our MACC scheme achieves a better trade-off between the achievable rate and subpacketization level compared to all other existing MACC schemes, at least for some memory sizes. The comparison is shown in Section VI-B.
- 3) The placement and delivery strategy of our MACC scheme is based on a combinatorial design called maximal cross resolvable design (MCRD). In Theorem 1 of Section IV we show a construction of MCRDs with general parameters.

C. Organization of the paper

In Section II we state the system model. In Section III we reproduce some definitions and lemmas from the literature on graph theory. In Section IV we first reproduce the definition of MCRD from the literature on combinatorial design theory, and then show a construction of MCRDs with general parameters. In Section V we isolate a class of user-to-cache association bipartite graphs. In Section VI we show our MACC scheme. In Section VI-B we compare our scheme with other relevant MACC schemes. In Section VII we provide two examples of the placement and delivery strategy of our MACC scheme. The paper is concluded in Section VIII.

II. SYSTEM MODEL

There is a central server, K caches, and K users. Let $U = \{k_1, k_2, \dots, k_K\}$ be the set of the users, and $C = \{c_1, c_2, \dots, c_K\}$ be the set of caches. So we have $|U| = |C| = K$.

Definition 1. A user-to-cache association bipartite graph is a bipartite graph $G = (U, C, E)$ where $U \cup C$ is the vertex set of G , sets U and C are disjoint, every edge in E connects one vertex in U to another vertex in C , an edge connects $u \in U$ to $v \in C$ if and only if user u accesses cache c .

The central server decides a user-to-cache association bipartite graph such that each user accesses $z \geq 1$ caches. Here z is called the access degree.

The central server contains N files W^1, W^2, \dots, W^N . Each file W^i is split into F subfiles $W^i(1), W^i(2), \dots, W^i(F)$. Here F is called the subpacketization level. The number j in $W^i(j)$ is called as the index of the subfile $W^i(j)$. Each subfile contains s bits. Each of the K caches can store MF subfiles in its memory. The ratio M/N is called normalized memory size. It is assumed that each user can instantaneously download the contents of the caches it accesses.

We describe a (K, z, M, R) multi-access coded caching scheme with uncoded placement. There are two phases: the placement phase and the delivery phase. The placement phase comes first. During the placement phase, the cache memory is filled up without knowing the users' future demands. Each cache c_i for $1 \leq i \leq K$ stores MF subfiles out of the NF subfiles available with the server. That is, if Z_i denote the content of the cache c_i , then $Z_i \subseteq \{W^1(1), W^1(2), \dots, W^N(F)\}$ and $|Z_i| = MF$.

During the delivery phase, each user makes a demand of a single file. Say the user k_i for $1 \leq i \leq K$ demands the file $W^{d_{k_i}}$ where $1 \leq d_{k_i} \leq N$. On knowing the demands of all users, the server uses a function (called encoding function)

$$f(d_{k_1}, d_{k_2}, \dots, d_{k_K}) : 2^{sNF} \rightarrow 2^{sRF} \quad (1)$$

to broadcast R files. Here R is called the (achievable) rate or the delivery load. Each user k_i for $1 \leq i \leq K$ uses a function (called decoding function)

$$f_{(k_i, d_{k_1}, d_{k_2}, \dots, d_{k_K})} : 2^{sRF} \times 2^{zsMF} \rightarrow 2^{sF} \quad (2)$$

in an attempt to retrieve its demanded file of F subfiles from the RF subfiles broadcast by the server and the memory contents of the z caches that user k_i accesses.

Suppose each user can successfully retrieve its demanded file using the R files broadcast by the server and the memory contents of the z caches it accesses. In that case, the placement and delivery policy and the user-to-cache association together is called a (K, z, M, R) MACC scheme. The objective is to jointly optimize the achievable rate (R files) and the subpacketization level by designing the placement and delivery policy and the user-to-cache association. The coding gain of a (K, z, M, R) MACC scheme is the number of users who benefit from each transmission made by the server during the delivery phase.

III. PRELIMINARIES ON GRAPH THEORY

An edge e of a graph is said to connect vertex u to vertex v if and only if one end point of the edge is u and the other edge point of the edge is v . Equivalently, one can say that the edge e is adjacent to vertices u and v .

Definition 2. Bipartite graph: A bipartite graph $G = (V_1, V_2, E)$ is such that $V_1 \cap V_2 = \emptyset$, and there does not exist any edge that is adjacent to two vertices in V_i for $i = 1, 2$.

Definition 3. Matching of a bipartite graph $G = (V_1, V_2, E)$: a matching is a subset $E' \subseteq E$ such that if $e_1, e_2 \in E'$ then there exists no vertex $v \in V_1 \cup V_2$ such that both e_1 and e_2 are adjacent to v . The size of a matching is the number of edges in the matching.

Definition 4. Vertex cover of a matching of a bipartite graph $G = (V_1, V_2, E)$: let $E' \subseteq E$ be a matching of the bipartite graph $G = (V_1, V_2, E)$. Vertex cover of E' is the set of all vertices that are adjacent to some edge in E' . Vertex cover of E' is denoted by $\nu(E')$.

Definition 5. Partial matching of a bipartite graph $G = (V_1, V_2, E)$: let $E' \subseteq E$ be a matching of the bipartite graph $G = (V_1, V_2, E)$. If $\nu(E') \subseteq V_1 \cup V_2$ then $E' \subseteq E$ is a partial matching of $G = (V_1, V_2, E)$.

Definition 6. Perfect matching of a bipartite graph $G = (V_1, V_2, E)$: let $E' \subseteq E$ be a matching of the bipartite graph $G = (V_1, V_2, E)$. If $\nu(E') = V_1 \cup V_2$ then $E' \subseteq E$ is a perfect matching of $G = (V_1, V_2, E)$.

IV. MAXIMAL CROSS RESOLVABLE DESIGNS

Definition 7. A design (X, \mathcal{A}) is an ordered tuple of two finite sets X and $\mathcal{A} \in 2^X$. The elements of the set X are called points. The elements of the set \mathcal{A} are called blocks. Each block in \mathcal{A} is a subset of X . Any two blocks in \mathcal{A} contain the same number of points.

Example 1. Let $X = \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16\}$, and $\mathcal{A} = \{\{1, 2, 3, 4\}, \{1, 5, 6, 7\}, \{1, 8, 9, 10\}, \{2, 11, 12, 13\}, \{3, 14, 15, 16\}, \{4, 5, 6, 7\}, \{8, 9, 10, 11\}, \{12, 13, 14, 15\}\}$. Then (X, \mathcal{A}) is a design.

Definition 8. A parallel class \mathcal{P} of a design (X, \mathcal{A}) is a subset of \mathcal{A} such that \mathcal{P} partitions X .

Example 2. In Example 1, $\mathcal{P} = \{\{1, 8, 9, 10\}, \{2, 11, 12, 13\}, \{3, 14, 15, 16\}, \{4, 5, 6, 7\}\}$ is a parallel class.

Since any two blocks has the same number of elements, and each parallel class partitions X , any two parallel class must have the same number of blocks. We denote the number of blocks in a parallel class by b . In Example 2, $b = 4$.

Definition 9. A design (X, \mathcal{A}) is a resolvable design if for some positive integer m , the design has m parallel classes $\mathcal{P}_1, \mathcal{P}_2, \dots, \mathcal{P}_m$ such that $\mathcal{P}_1, \mathcal{P}_2, \dots, \mathcal{P}_m$ partitions \mathcal{A} .

Example 3. It can be seen that the design (X, \mathcal{A}) considered in Example 1 is not resolvable.

Example 4. Let $X = \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16\}$. And $\mathcal{A} = \{\{1, 2, 3, 4\}, \{5, 6, 7, 8\}, \{9, 10, 11, 12\}, \{13, 14, 15, 16\}, \{1, 2, 9, 13\}, \{5, 6, 10, 14\}, \{3, 7, 11, 15\}, \{4, 8, 12, 16\}\}$. Then (X, \mathcal{A}) is a design. Consider the two parallel classes of (X, \mathcal{A}) .

$$\mathcal{P}_1 = \{\{1, 2, 3, 4\}, \{5, 6, 7, 8\}, \{9, 10, 11, 12\}, \{13, 14, 15, 16\}\}$$

$$\mathcal{P}_2 = \{\{1, 2, 9, 13\}, \{5, 6, 10, 14\}, \{3, 7, 11, 15\}, \{4, 8, 12, 16\}\}.$$

It can be seen that each sets \mathcal{P}_1 and \mathcal{P}_2 partition X , and they together partition \mathcal{A} . So (X, \mathcal{A}) is a resolvable design. For this design, $b = 4$.

Definition 10. A resolvable design (X, \mathcal{A}) with m parallel classes is called a maximal cross resolvable design (MCRD) if for some positive integer μ_m , any m blocks chosen from m distinct parallel classes have μ_m elements in common. That is, if for some ordering of the blocks in parallel class \mathcal{P}_i the j^{th} block is denoted by $B(i, j_i)$ for $1 \leq i \leq m$, $1 \leq j_i \leq b$, then for any $1 \leq j_i \leq b$, $|\cap_{i=1}^m B(i, j_i)| = \mu_m$. The integer μ_m is called m^{th} cross intersection number.

Example 5. The resolvable design shown in Example 4 is not a maximal cross resolvable design as: between $B(1, 1) = \{1, 2, 3, 4\}$, $B(2, 1) = \{1, 2, 9, 13\}$ we have $|B(1, 1) \cap B(2, 1)| = 2$, but between $B(1, 2) = \{5, 6, 7, 8\}$ and $B(2, 1) = \{1, 2, 9, 13\}$ we have $B(1, 2) \cap B(2, 1) = 0$.

Example 6. Let $X = \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16\}$. And $\mathcal{A} = \{\{1, 2, 3, 4\}, \{5, 6, 7, 8\}, \{9, 10, 11, 12\}, \{13, 14, 15, 16\}, \{1, 5, 9, 13\}, \{2, 6, 10, 14\}, \{3, 7, 11, 15\}, \{4, 8, 12, 16\}\}$. Then (X, \mathcal{A}) is a design. Consider the two parallel classes of (X, \mathcal{A}) .

$$\mathcal{P}_1 = \{\{1, 2, 3, 4\}, \{5, 6, 7, 8\}, \{9, 10, 11, 12\}, \{13, 14, 15, 16\}\}$$

$$\mathcal{P}_2 = \{\{1, 5, 9, 13\}, \{2, 6, 10, 14\}, \{3, 7, 11, 15\}, \{4, 8, 12, 16\}\}.$$

It can be seen that each sets \mathcal{P}_1 and \mathcal{P}_2 partition X , and they together partition \mathcal{A} . Furthermore, the cardinality of the intersection between any two blocks selected from the two distinct parallel classes is always 1. So the design is a maximal cross resolvable design (MCRD) with $\mu_2 = 1$.

Lemma 1. *Let (X, \mathcal{A}) be an MCRD with m parallel classes and $\mu_m = 1$. Furthermore, say each parallel class contains b blocks. Then, each block has exactly b^{m-1} elements. Moreover, $|X| = b^m$.*

The proof of Lemma 1 is deferred till Appendix A-A.

We consider a function $I : \{1, 2, \dots, b\}^m \rightarrow X$ defined as $I(l_1, l_2, \dots, l_m) = \cap_{n=1}^m B(n, l_n)$. The set $\{l_1, l_2, \dots, l_{i-1}, l_{i+1}, \dots, l_m\}$ can be chosen in b^{m-1} ways, and for each such choice there is one instance of the set $I(l_1, l_2, \dots, l_{i-1}, j, l_{i+1}, \dots, l_m) = \cap_{n=1, n \neq i}^m B(n, l_n) \cap B(i, j)$.

Corollary 1. *Let (X, \mathcal{A}) be an MCRD with m parallel classes. For $1 \leq i \leq m$, $1 \leq j \leq b$, say $B(i, j)$ is the j^{th} block in the i^{th} parallel class for some ordering of the parallel classes and the blocks in each parallel class. Then*

$$B(i, j) = \cup_{l_1=1}^b \cup_{l_2=1}^b \cdots \cup_{l_{i-1}=1}^b \cup_{l_{i+1}=1}^b \cdots \cup_{l_m=1}^b I(l_1, l_2, \dots, l_{i-1}, j, l_{i+1}, \dots, l_m). \quad (3)$$

The proof of Corollary 1 is deferred to Appendix A-B.

Example 7. In Example 6, as per Lemma 1, $m = 2$, $b = 4$, $\mu_m = 1$, and each block has $\mu_m b^{m-1} = 4$ elements. In congruence to Corollary 1, we have:

$$\begin{aligned} \cup_{l=1}^4 I(1, l) &= \cup_{l=1}^4 B(1, 1) \cap B(2, l) \\ &= \{B(1, 1) \cap B(2, 1)\} \cup \{B(1, 1) \cap B(2, 2)\} \cup \{B(1, 1) \cap B(2, 3)\} \cup \{B(1, 1) \cap B(2, 4)\} \\ &= B(1, 1) \cap \{B(2, 1) \cup B(2, 2) \cup B(2, 3) \cup B(2, 4)\} \\ &= B(1, 1) \cap X = B(1, 1). \end{aligned}$$

Example 8. Let $X = \{1, 2, \dots, 27\}$. And \mathcal{A} be the sets contained in the following parallel classes $\mathcal{P}_1, \mathcal{P}_2, \mathcal{P}_3$.

$$\mathcal{P}_1 = \{\{1, 2, 3, 4, 5, 6, 7, 8, 9\}, \{10, 11, 12, 13, 14, 15, 16, 17, 18\}, \{19, 20, 21, 22, 23, 24, 25, 26, 27\}\}$$

$$\mathcal{P}_2 = \{\{1, 2, 3, 10, 11, 12, 19, 20, 21\}, \{4, 5, 6, 13, 14, 15, 22, 23, 24\}, \{7, 8, 9, 16, 17, 18, 25, 26, 27\}\}$$

$$\mathcal{P}_3 = \{\{1, 4, 7, 10, 13, 16, 19, 22, 25\}, \{2, 5, 8, 11, 14, 17, 20, 23, 26\}, \{3, 6, 9, 12, 15, 18, 21, 24, 27\}\}.$$

It can be seen (X, \mathcal{A}) is a design with $b = 3$, $\mu_3 = 1$. Consistent with Lemma 1, each block has $\mu_3 b^2$ elements. Furthermore, in congruence to Corollary 1 we have:

$$\begin{aligned} &\cup_{l_2=1}^3 \cup_{l_3=1}^3 I(1, l_2, l_3) \\ &= \cup_{l_2=1}^3 \{I(1, l_2) \cap B(3, 1)\} \cup \{I(1, l_2) \cap B(3, 1)\} \cup \{I(1, l_2) \cap B(3, 1)\} \\ &= \cup_{l_2=1}^3 I(1, l_2) \cap \{B(3, 1) \cup B(3, 2) \cup B(3, 3)\} \\ &= \cup_{l_2=1}^3 I(1, l_2) = \cup_{l_2=1}^3 \{B(1, 1) \cap B(2, l_2)\} \\ &= B(1, 1) \cap \{B(2, 1) \cup B(2, 2) \cup B(2, 3)\} = B(1, 1). \end{aligned}$$

Theorem 1. *For any positive integer m , n , and b , there exists an MCRD (X, \mathcal{A}) with $X = \{1, 2, \dots, nb^m\}$, $|\mathcal{A}| = mb$, m parallel classes, b blocks in each parallel class, and $\mu_m = n$.*

The proof of Theorem 1 is shown in Appendix B.

V. A CLASS OF USER-TO-CACHE ASSOCIATION BIPARTITE GRAPHS

Let m and b be integers where $K = mb$. Consider the class of all user-to-cache association bipartite graphs $G = (U, C, E)$ that follows the following three conditions **C1**, **C2**, **C3**.

- Condition **C1**: $G = (U, C, E)$ is a disjoint union of m user-to-cache association bipartite graphs $G_i = (K_i, C_i, E_i)$ for $1 \leq i \leq m$ such that $|K_i| = |C_i| = b$, if $j \neq i$, $K_i \cap K_j = \emptyset$, $C_i \cap C_j = \emptyset$.
- Condition **C2**: For $1 \leq i \leq m$ the vertices C_i can be partitioned into z disjoint subsets $C_{(i,1)}, C_{(i,2)}, \dots, C_{(i,z)}$ such that (i) for $1 \leq l \leq z-1$ the subset $C_{(i,l)}$ contains $\lfloor \frac{b}{z} \rfloor$ vertices, the subset $C_{(i,z)}$ contains $b - (z-1)\lfloor \frac{b}{z} \rfloor$ vertices; (ii) any vertex $u \in K_i$ connects to at most one vertex in $C_{(i,l)}$ for $1 \leq l \leq z$.
- Condition **C3**: There is a perfect matching in G_i for $1 \leq i \leq m$.

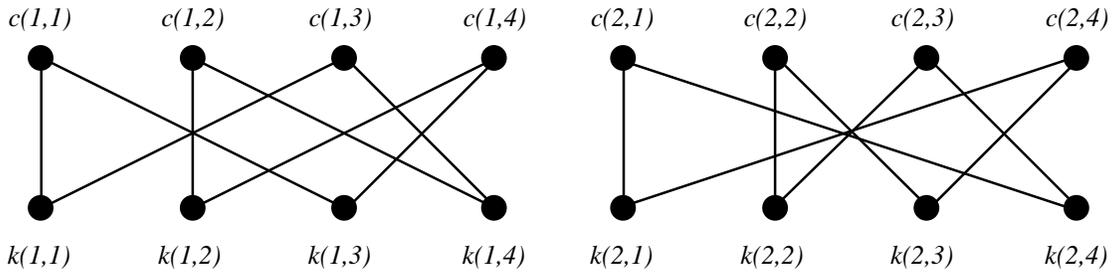


Fig. 1. An example user-to-cache association bipartite graph $G = (U, C, E)$ that satisfies conditions **C1**, **C2**, **C3**. There are 8 users $U = \{k(i, j) \mid 1 \leq i \leq 2, 1 \leq j \leq 4\}$, 8 caches $C = \{c(i, j) \mid 1 \leq i \leq 2, 1 \leq j \leq 4\}$, $b = 4$, $m = 2$, and we have $z = 2$. A vertex labeled by $k(i, j)$ is connected to a vertex labeled by $c(i, j)$ if and only if user $k(i, j)$ accesses cache $c(i, j)$. For instance, user $k(1, 1)$ accesses caches $c(1, 1)$ and $c(1, 3)$.

Figure 1 shows an example user-to-cache association bipartite graph that satisfies conditions **C1**, **C2**, **C3**.

As a result of condition **C1**, a user $k \in K_i$ cannot access any caches from C_j for $j \neq i$. As it will be shown in Section D-B, condition **C1** enforces a matching of size m , which in turn allows our scheme to achieve a coding gain of m : in our delivery algorithm, each transmission from the central server will be useful to one user in each K_i for $1 \leq i \leq m$.

Let the j^{th} user for $1 \leq j \leq b$ in K_i be denoted by $k(i, j)$ (for some ordering of the users in K_i). So $K_i = \{k(i, 1), k(i, 2), \dots, k(i, b)\}$. Similarly, let the j^{th} cache for $1 \leq j \leq b$ in C_i be denoted by $c(i, j)$ (for some ordering of the caches in C_i). So $C_i = \{c(i, 1), c(i, 2), \dots, c(i, b)\}$.

Conditions **C2**, **C3** apply on $G_i = (K_i, C_i, E_i)$ for $1 \leq i \leq m$. Condition **C2** ensures that each user accesses at most z caches. Furthermore, a user $k(i, j)$ can access at most one cache in $C_{(i,l)}$ for $1 \leq l \leq z$. Specifically, if $(k(i, j), c(i, j_1)), (k(i, j), c(i, j_2)) \in E_i$ and $c(i, j_1) \in C_{(i,n_1)}$, $c(i, j_2) \in C_{(i,n_2)}$, then $C_{(i,n_1)} \neq C_{(i,n_2)}$. Note, caches contained in $C_{(i,z)}$ are the caches that are in C_i but not in the sets $C_{(i,l)}$ for $1 \leq l \leq z-1$. The intuitive reasoning behind such partitioning is as follows. In the placement phase, we will place contents such a way that caches belonging to the separate partitions will never have any files in common; and caches belonging to the same partition may have files in common. Condition **C2** helps maximize the local coding gain.

Condition **C3** ensures that there is some set $M_i \subseteq E_i$ such that for each user $k(i, j)$ there is exactly one cache $c(i, j')$ such that $(k(i, j), c(i, j')) \in M_i$, and for each cache $c(i, l)$ there is only one user $k(i, l')$ such that $(k(i, l'), c(i, l)) \in M_i$. So, condition **C3** ensures that there is a bijection f_{M_i} from K_i to C_i : if $(u, v) \in M_i$, u maps to v .

VI. MAIN RESULT AND COMPARISONS

Theorem 2. Consider a MACC problem with K users, K caches, access degree z , such that for some integers m and b where $K = mb$ the user-to-cache association bipartite graph satisfies the conditions **C1**, **C2**, **C3**. Let the central server have N files where $N \geq K$, and the cache memory size be M files. For $M = \frac{tN}{b}$ ($t \in \mathbb{N}$) and a subpacketization level b^m a rate of R files is achievable where

$$\begin{aligned} R &= (b - tz) \text{ for } 1 \leq t \leq \lfloor \frac{b}{z} \rfloor \\ R &= (b - (z - 1)\lfloor \frac{b}{z} \rfloor - t) \text{ for } \lfloor \frac{b}{z} \rfloor < t < b - (z - 1)\lfloor \frac{b}{z} \rfloor \\ R &= 0 \text{ for } t \geq b - (z - 1)\lfloor \frac{b}{z} \rfloor. \end{aligned}$$

The proof of Theorem 2 shown in Appendix D.

A. Rate-subpacketization trade-off

It can be seen from Theorem 2 that the achievable rate increases with b . Since $K = mb$, increasing m decreases b , and thereby the achievable rate decreases with increasing m . On the other hand, the subpacketization level increases exponentially with m and increases polynomially with b . So our scheme has a trade-off: reducing rate leads to a higher subpacketization, and reducing subpacketization leads to a higher rate. It is to be noted that our scheme requires $z \leq b$, and so decreasing b limits the value of z .

B. Comparison

In this subsection, we compare the rate and subpacketization level of our scheme with the MACC schemes shown in references [6]–[13], and show that at least for some memory sizes our scheme achieves either a lesser rate or a lesser subpacketization.

The scheme shown in [7] is denoted by SPE scheme, the scheme in [8] is denoted by RK scheme, the scheme in [9] is denoted by NT scheme, the scheme in [13] is denoted by SICPS scheme, the scheme in [10] is denoted by SR1 scheme, the scheme in [11] is denoted by SR2 scheme, and the scheme in [12] is denoted by MR scheme.

Consider the case when $K = 100$ and $z = 5$. For every m and b such that $mb = 100$ Theorem 2 shows a solution. Fig. 2a shows the plot of achievable rate R against normalized cache memory size M/N and Fig. 2b shows the plot of base 10 logarithm of subpacketization level against the M/N . In Fig. 2a, the point $(0, 100)$ is trivially achieved, the point $(1/50, 45)$ is achieved for $b = 50$, the point $(1/25, 20)$ is achieved for $b = 25$, the point $(1/20, 15)$ is achieved for $b = 20$, the point $(1/10, 5)$ is achieved for $b = 10$, and the point $(1/5, 0)$ is achieved for any legitimate value of b . To note that the user-to-cache association bipartite graph depends upon b . So, for

instance, the user-to-cache association bipartite graph when $b = 50$ is different from the user-to-cache association bipartite graph when $b = 25$.

Fig. 2a shows that our scheme may achieve a better rate than SPE scheme, SR1 scheme, SR2 scheme, MR scheme, and RK scheme for some ranges of M/N . On the other hand, Figure 2b shows that our scheme achieves a lesser subpacketization level than NT scheme, SICPS scheme, and RK scheme for some values of M/N .

At $M/N = 0.1$ our scheme achieves $R = 5$ files, NT scheme achieves $R \approx 4.5$ files, SICPS scheme achieves $R \approx 4.3$ files, while at the same value of M/N the three schemes achieve a subpacketization level 10^{10} , $10^{12.8}$, $10^{11.1}$ respectively (approximate values for the latter two).

While the memory-rate trade-off curve of the NT scheme and the SICPS scheme (In Fig. 2a) is below our scheme for the entire range of M/N , the RK scheme and the SR1 scheme seems to have a crossover point after $M/N = 0.16$. For clarity, we show these proximal points in Table VI-B.

TABLE I
SOME PROXIMAL POINTS IN FIG. 2A SHOWN FOR CLARITY.

| M/N | SR1 Scheme | RK Scheme | Our Scheme |
|------|------------|-----------|------------|
| 0.16 | 3.4965 | 4 | 2 |
| 0.17 | 1.6953 | 2.25 | 1.5 |
| 0.18 | 0.9528 | 1 | 1 |
| 0.19 | 0.2103 | 0.25 | 0.5 |
| 0.2 | 0 | 0 | 0 |

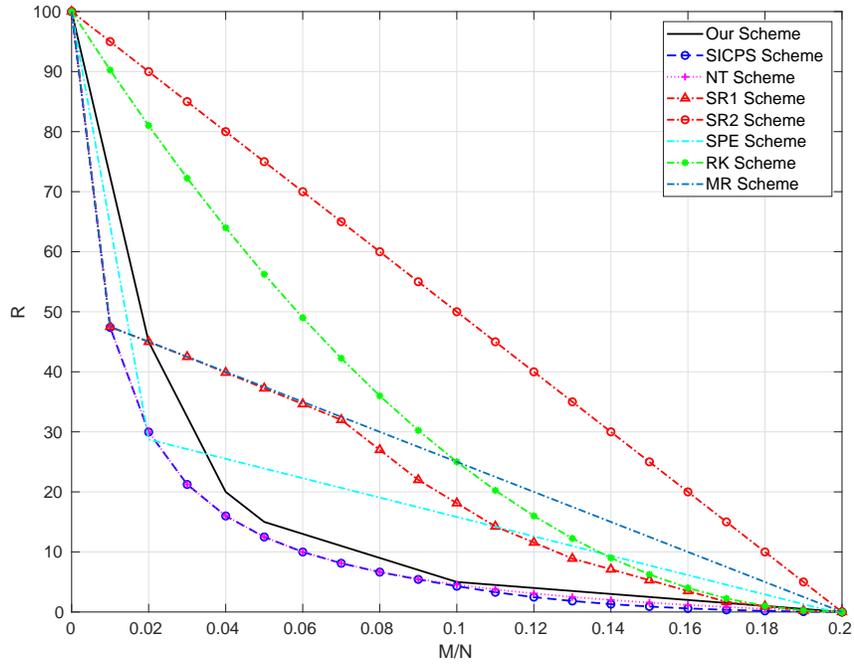
In Table VI-B we show the achievable rate and subpacketization level of all the relevant schemes. It has been shown in [9] that their NT scheme performs at least as good as the scheme in [6] in both achievable rate and subpacketization level for all memory sizes, and hence the latter scheme is not considered for comparison. Some of the equations in Table VI-B are sizable, in such cases the reader has been referred to see the equation in the original paper.

For the rest of this section $t \in \{1, 2, \dots, \lfloor \frac{b}{z} \rfloor\}$, $t' \in \{1, 2, \dots, \lfloor \frac{K}{z} \rfloor\}$, and $t'' \in \{1, 2, \dots, K\}$. It is to be noted that in Table VI-B the SR2 scheme is applicable only if t'' divides K and $(K - t''z + t')$ divides K ; and SR1 scheme is applicable only if $\gcd(t'', K) = 1$.

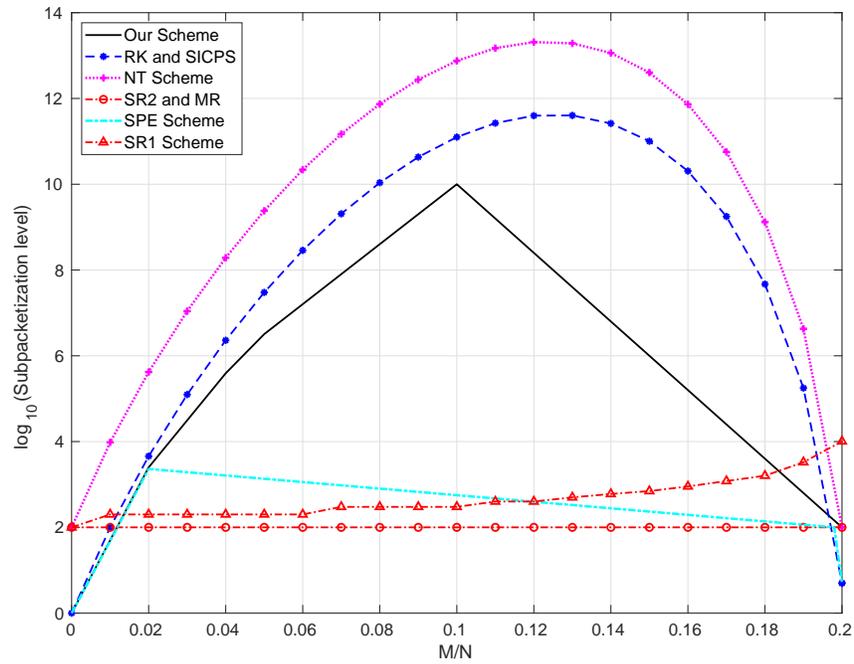
1) *Comparison with the SPE scheme:* Due to the complexity of equation (2) of [7] we could not provide an analytical comparison between the achievable rate of the SPE scheme and our scheme. For the case considered in Fig. 2a it can be seen that our scheme achieves a lesser rate for a large range of the normalized memory size.

For a comparison of the subpacketization level, if in our scheme we take $m = 1$, then $b = K$ and at $\frac{M}{N} = \frac{2}{K}$ our scheme achieves a lesser (linear) subpacketization level than the SPE scheme (though it is at the cost of a lesser coding gain and thus a higher rate).

2) *Comparison with the RK scheme, NT scheme, and SICPS scheme: Comparison of the achievable rate:* It has been shown in [13] that the achievable rate of the SICPS scheme is lesser than that of the RK scheme and NT scheme. However, due to the complexity of equation (6) of [13] prescribing the achievable rate of the SICPS



(a)



(b)

Fig. 2. Comparison of rate and subpacketization level of Our Scheme, the RK scheme, the SICPS scheme, the NT scheme, the SPE scheme, the SR1 scheme, the SR2 scheme, and the MR scheme for MACC problem with $K = 100$, $z = 5$. The subpacketization level in Fig. 2b is shown in \log_{10} scale. It is to be noted that the achievable points in Fig. 2b are connected by a line for aesthetics.

TABLE II
COMPARISON OF SCHEMES

| | M/N | Rate | Subpacketization level |
|--------------|-----------------|--|---|
| Our Scheme | $\frac{t}{b}$ | $b - tz$ | b^m |
| SPE Scheme | $\frac{2}{K}$ | see eqn. (2) of [7] | $\frac{K(K-2z+2)}{4}$ |
| RK Scheme | $\frac{t'}{K}$ | $\frac{(K-t'z)^2}{K}$ | $\frac{K}{t'} \binom{K-t'z+t'-1}{t'-1}$ |
| NT Scheme | same as above | $\frac{K-t'z}{t'+1}$ | $K \binom{K-t'z+t'}{t'}$ |
| SICPS Scheme | same as above | see eqn. (6) of [13] | $\frac{K}{t'} \binom{K-t'z+t'-1}{t'-1}$ |
| SR1 Scheme | $\frac{t''}{K}$ | see Theorem 1 of [10] | cK ($1 \leq c \leq K$) |
| SR2 Scheme | same as above | $\frac{(K-t''z)(K-t''z+t'')}{2K}$ | K |
| MR Scheme | $\frac{1}{K}$ | $\frac{1}{K} \lceil \frac{K(K-z)}{2 + \lfloor \frac{z}{K-z+1} \rfloor + \lfloor \frac{z-1}{K-z+1} \rfloor} \rceil$ | K |

scheme, we are not able to provide an analytical comparison. For the example considered in Fig. 2a, it can be seen that the achievable rate of the NT scheme and the SICPS scheme is less than our scheme, but the rate of the RK scheme is higher than our scheme for the most part of the memory region.

Lemma 2. At $\frac{M}{N} = \frac{t}{b} = \frac{t'}{K}$ where $\frac{M}{N} < \min\{\frac{1}{b} \lfloor \frac{b}{z} \rfloor, \frac{1}{K} \lfloor \frac{K}{z} \rfloor, \frac{K-b}{Kz}\}$ our scheme achieves a lesser rate than that of the RK scheme.

The proof is shown in Appendix C-A.

At $\frac{M}{N} = \frac{t}{b} = \frac{t'}{K}$ where $\frac{t}{b} < \min\{\frac{1}{b} \lfloor \frac{b}{z} \rfloor, \frac{1}{K} \lfloor \frac{K}{z} \rfloor\}$, the NT scheme achieves a rate $\frac{m(b-tz)}{mt+1}$, whereas our scheme achieves a rate $(b - tz)$. So the achievable rate of the NT scheme is lesser by a factor of $(t + \frac{1}{m}) = b(\frac{M}{N} + \frac{1}{K})$. So for a given $\frac{M}{N}$ and K , the less the value of b the less is the gap between the rate of our scheme and the NT scheme.

Comparison of the Subpacketization level: We show that even though the SICPS scheme and the NT scheme achieve a lesser rate than our scheme, at least for some values $\frac{M}{N}$ our scheme achieves a lesser subpacketization level. Analytical comparison of the subpacketization level of these two schemes with our schemes is complicated due to the disparate nature of the closed form expression of the respective subpacketization levels, and hence we could provide an analytical result only under a restricted setting.

Lemma 3. At $\frac{M}{N} = \frac{1}{b} = \frac{m}{K}$ if $b \geq \sqrt{K(z-1)} + 1$, $b > z$, and $m \leq \lfloor \frac{K}{z} \rfloor$ our scheme achieves a lesser subpacketization level than the RK scheme, NT scheme, and the SICPS scheme.

The proof is shown in Appendix C-B. For the case considered in Fig. 2b it can be seen that our scheme achieves a lesser subpacketization level than these two schemes at several values $\frac{M}{N}$.

3) *Comparison with the SR1 scheme:* For the SR1 scheme the non-trivial corner points has $\frac{M}{N} = \frac{t''}{K}$. For our scheme the non-trivial corner points are $\frac{M}{N} = \frac{t}{b} = \frac{mt}{K}$. So for the $\frac{M}{N}$ value of the two schemes to coincide we must have $\frac{t''}{K} = \frac{t}{b}$, which leads to $t'' = mt$. However, in such a case $\gcd(t'', K) = \gcd(mt, mb) \geq m$. So unless $m = 1$ and $\gcd(t, b) = 1$, the $\frac{M}{N}$ value of the two schemes do not coincide.

Due to the complexity of the achievable rate expression of the SR1 scheme and its disparity with our scheme, we could only provide the following analytical comparison under a restricted setting.

Lemma 4. *For some positive integers m_1, m_2 such that both m_1 and m_2 divides K , $m_1 < m_2$, $b_1 = \frac{K}{m_1}$, $b_2 = \frac{K}{m_2}$, $b_1, b_2 \geq z$, $\lambda \in \mathbb{R}$, $0 \leq \lambda \leq 1$, $t'' = m_1 + \lambda(m_2 - m_1)$, $t'' \neq \frac{K-1}{z}$, $\gcd(t'', K) = 1$, at $\frac{M}{N} = \frac{t''}{K}$ our scheme achieves a better rate than the SR2 scheme if*

$$b_1 + \lambda(b_2 - b_1) \leq \frac{(K - t''z)(K - t''z + 2)}{K + 2} + z.$$

The proof is shown in Appendix C-C.

For example, for $K = 100$, $z = 5$, if we take $m_1 = 4$, $m_2 = 10$, $\lambda = \frac{1}{2}$, then all the conditions of Lemma 4 gets satisfied, and for the point $\frac{t''}{K} = \frac{7}{100}$ the lemma claims that our scheme achieves a lesser rate. Indeed, at $\frac{M}{N} = \frac{7}{100}$, the SR1 scheme a rate of 32 files, whereas our scheme achieves a rate of 12.5 files (the point $(\frac{7}{100}, 12.5)$ lies on the line connecting the points $(\frac{1}{25}, 20)$ and $(\frac{1}{10}, 5)$). (The comparison is shown in Fig. 2a.)

4) *Comparison with the SR2 scheme:*

Lemma 5. *At $\frac{M}{N} = \frac{t}{b} = \frac{t''}{K}$ where t divides b , $(b - tz + t)$ divides b , and $t \leq \lfloor \frac{b}{z} \rfloor$ our scheme achieves a lesser rate than that of the SR2 scheme when $\frac{M}{N} \leq \frac{m-2}{m(z-1)}$.*

The proof is shown in Appendix C-D.

For example, for $K = 120$, $z = 5$, at $\frac{M}{N} = \frac{15}{120}$, the SR1 scheme achieves a rate 11.25 files, whereas our scheme for $t = 3, b = 24, m = 5$ achieves a rate 9 files. For the case considered in Fig. 2a, for no $\frac{M}{N} = \frac{t}{b}$ the conditions t divides b and $(b - tz + t)$ divides b satisfied, so the plot of the rate achieved by the SR2 scheme is a straight line connecting the points $(0, 100)$ and $(1/5, 0)$.

The SR2 scheme has a subpacketization level of K . For $b = K$ and $m = 1$ our scheme also produces a scheme with subpacketization level K . If the subpacketization of our scheme and the SR2 scheme is kept the same, then SR2 scheme achieves a lesser rate.

5) *Comparison with the MR scheme:*

Lemma 6. *At $\frac{M}{N} = \frac{1}{b}$ where $K = mb$ and $m \geq 3$, our scheme achieves a lesser rate than the MR scheme.*

The proof is shown in Appendix C-E. For $m = 2$ the rates of our scheme and the MR scheme are equal. In Fig. 2a it can be seen that for $\frac{M}{N} > \frac{1}{50}$ our scheme achieves a lesser rate.

VII. EXAMPLES

A. *First Example:* $K = 8$, $z = 2$, $\frac{M}{N} = \frac{1}{4}$

In the first example, we consider a MACC problem with $K = 8$, $z = 2$, $\frac{M}{N} = \frac{1}{4}$. For $b = 4$, $m = 2$, let $G = (U, C, E)$ shown in Fig. 1 be the user-to-cache association bipartite graph. It is to be noted that there are other user-to-cache association bipartite graphs as well that satisfies the criteria set by this example problem. We have $U = K_1 \cup K_2$ where $K_1 = \{k(1, 1), k(1, 2), k(1, 3), k(1, 4)\}$, $K_2 = \{k(2, 1), k(2, 2), k(2, 3), k(2, 4)\}$. And, $C = C_1 \cup C_2$ where $C_1 = \{c(1, 1), c(1, 2), c(1, 3), c(1, 4)\}$, $C_2 = \{c(2, 1), c(2, 2), c(2, 3), c(2, 4)\}$.

Condition **C3** ensures the existence of a bijection f_{M_i} from K_i to C_i . Let us consider the following bijection $f_{M_1}(k(1,1)) = c(1,1)$, $f_{M_1}(k(1,2)) = c(1,2)$, $f_{M_1}(k(1,3)) = c(1,4)$, $f_{M_1}(k(1,4)) = c(1,3)$, $f_{M_2}(k(2,1)) = c(2,4)$, $f_{M_2}(k(2,2)) = c(2,3)$, $f_{M_2}(k(2,3)) = c(2,2)$, $f_{M_2}(k(2,4)) = c(2,1)$.

The map f_{M_i} could have been defined in other ways as well. For instance, we could have defined $f_{M_1}(k(1,1)) = c(1,3)$, $f_{M_1}(k(1,2)) = c(1,4)$, $f_{M_1}(k(1,3)) = c(1,1)$, $f_{M_1}(k(1,4)) = c(1,2)$.

As per condition **C2**, $C_i = C_{(i,1)} \cup C_{(i,2)}$, where

$$C_{(i,1)} = \{c(i,1), c(i,2)\}, C_{(i,2)} = \{c(i,3), c(i,4)\} \text{ for } 1 \leq i \leq 2.$$

For $i = 1, 2$, $j = 1, 2, 3, 4$, set $C_{k(i,j)}$ denotes the set of caches user $k(i,j)$ accesses. We have

$$\begin{aligned} C_{k(1,1)} &= \{c(1,1), c(1,3)\}, C_{k(1,2)} = \{c(1,2), c(1,4)\}, C_{k(1,3)} = \{c(1,1), c(1,4)\}, C_{k(1,4)} = \{c(1,2), c(1,3)\}, \\ C_{k(2,1)} &= \{c(2,1), c(2,4)\}, C_{k(2,2)} = \{c(2,2), c(2,3)\}, C_{k(2,3)} = \{c(2,2), c(2,4)\}, C_{k(2,4)} = \{c(2,1), c(2,3)\}. \end{aligned}$$

We consider the following MCRD to be used for placement and delivery.

$$\begin{aligned} \mathcal{P}_1 &= \{\{1, 2, 3, 4\}, \{5, 6, 7, 8\}, \{9, 10, 11, 12\}, \{13, 14, 15, 16\}\} \\ \mathcal{P}_2 &= \{\{1, 5, 9, 13\}, \{2, 6, 10, 14\}, \{3, 7, 11, 15\}, \{4, 8, 12, 16\}\}. \end{aligned}$$

It is to be noted that we could have used any MCRD which has $m = 2$ parallel classes, $b = 4$ blocks in each parallel class, and $\mu_2 = 1$. Let

$$\begin{aligned} B(1,1) &= \{1, 2, 3, 4\}, B(1,2) = \{5, 6, 7, 8\}, B(1,3) = \{9, 10, 11, 12\}, B(1,4) = \{13, 14, 15, 16\} \\ B(2,1) &= \{1, 5, 9, 13\}, B(2,2) = \{2, 6, 10, 14\}, B(2,3) = \{3, 7, 11, 15\}, B(2,4) = \{4, 8, 12, 16\}. \end{aligned}$$

Partition sets \mathcal{P}_1 and \mathcal{P}_2 as following

$$\mathcal{P}_{(i,1)} = \{B(i,1), B(i,2)\}, \mathcal{P}_{(i,2)} = \{B(i,3), B(i,4)\} \text{ for } 1 \leq i \leq 2.$$

Placement: Split each file into $b^m = 16$ subfiles. A cache stores the subfiles indexed by the elements of some of the blocks of the MCRD considered above. Let the cache $c(i,j)$ store the subfiles (of all files) indexed by the elements of the blocks contained in $B_{c(i,j)}$, where

$$B_{c(i,j)} = \{B(i,j)\} \text{ for } 1 \leq i \leq 2, 1 \leq j \leq 4. \quad (4)$$

Let $B_{k(i,j)}$ denote the set of blocks such that if $B \in B_{k(i,j)}$ then for some cache $c(i,j') \in C_{k(i,j)}$ we have $B \in B_{c(i,j')}$. So we have

$$\begin{aligned} B_{k(1,1)} &= \{B(1,1), B(1,3)\}, & B_{k(1,2)} &= \{B(1,2), B(1,4)\}, & B_{k(1,3)} &= \{B(1,1), B(1,4)\}, \\ B_{k(1,4)} &= \{B(1,2), B(1,3)\}, & B_{k(2,1)} &= \{B(2,1), B(2,4)\}, & B_{k(2,2)} &= \{B(2,2), B(2,3)\}, \\ B_{k(2,3)} &= \{B(2,2), B(2,4)\}, & B_{k(2,4)} &= \{B(2,1), B(2,3)\}. \end{aligned}$$

We now construct a new graph that we call as the demand graph $\bar{G} = (V_1, V_2, \bar{E})$ shown in Fig. 3. We have for $l = 1, 2$, $V_l = \{v_{(l,i,j)} | 1 \leq i \leq 2, 1 \leq j \leq 4\}$. There is a bijection from the vertices in V_1 to the caches in

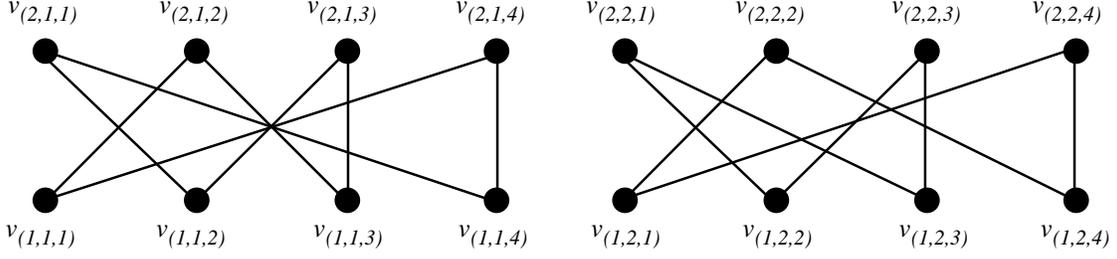


Fig. 3. The demand graph $\bar{G} = (V_1, V_2, \bar{E})$ for the user-to-cache association bipartite graph $G = (U, C, E)$ shown in Fig. 1 when $\frac{M}{N} = \frac{1}{4}$ (set $B_{c(i,j)}$ defined in equation (4)). An edge connects $v_{(1,i,j)}$ to $v_{(2,i',j^*)}$ if and only if $i = i'$ and the set $B_{f_{M_i}^{-1}(c(i,j))}$ does not contain the block $B(i, j^*)$.

C , and there is another bijection from the vertices in V_2 to the blocks in $\mathcal{P}_1 \cup \mathcal{P}_2$. Say for $i = 1, 2, 1 \leq j \leq 4$, the vertex $v_{(1,i,j)}$ maps to the cache $c(i, j)$, similarly, say that the vertex $v_{(2,i,j)}$ maps to the block $B(i, j)$. For $i = 1, 2, 1 \leq j \leq 4$ there exists an edge in \bar{E} connecting vertex $v_{(1,i,j)} \in V_1$ to vertex $v_{(2,\bar{i},j^*)} \in V_2$ if and only if $i = \bar{i}$ and $B(i, j^*) \notin B_{k(i,j')}$ where $k(i, j') = f_{M_i}^{-1}(c(i, j))$.

We define a map $f_{(i,j)} : \{1, 2\} \rightarrow \{1, 2, 3, 4\} \setminus \{l \mid B(i, l) \in B_{k(i,j)}\}$ for $1 \leq i \leq 2, 1 \leq j \leq 4$ as following.

$$\begin{aligned} f_{(1,1)}(1) &= 2, & f_{(1,1)}(2) &= 4, & f_{(1,2)}(1) &= 1, & f_{(1,2)}(2) &= 3, & f_{(1,3)}(1) &= 2, & f_{(1,3)}(2) &= 3, \\ f_{(1,4)}(1) &= 1, & f_{(1,4)}(2) &= 4, & f_{(2,1)}(1) &= 2, & f_{(2,1)}(2) &= 3, & f_{(2,2)}(1) &= 1, & f_{(2,2)}(2) &= 4, \\ f_{(2,3)}(1) &= 1, & f_{(2,3)}(2) &= 3, & f_{(2,4)}(1) &= 2, & f_{(2,4)}(2) &= 4. \end{aligned}$$

We define a partial matching M_{j_1, j_2}^n for $n = 1, 2, 1 \leq j_1, j_2 \leq 4$ as following.

$$M_{j_1, j_2}^n = \{(v_{(1,i,j_i)}, v_{(2,i,f_{(i,j_i)}(n))}) \mid i = 1, 2\} \text{ where } k(i, j_i) = f_{M_i}^{-1}(c(i, j_i)).$$

So we have

$$\begin{aligned} M_{1,1}^1 &= \{(v_{(1,1,1)}, v_{(2,1,2)}), (v_{(1,2,1)}, v_{(2,2,2)})\} & M_{1,2}^1 &= \{(v_{(1,1,1)}, v_{(2,1,2)}), (v_{(1,2,2)}, v_{(2,2,1)})\} \\ M_{1,3}^1 &= \{(v_{(1,1,1)}, v_{(2,1,2)}), (v_{(1,2,3)}, v_{(2,2,1)})\} & M_{1,4}^1 &= \{(v_{(1,1,1)}, v_{(2,1,2)}), (v_{(1,2,4)}, v_{(2,2,2)})\} \\ M_{2,1}^1 &= \{(v_{(1,1,2)}, v_{(2,1,1)}), (v_{(1,2,1)}, v_{(2,2,2)})\} & M_{2,2}^1 &= \{(v_{(1,1,2)}, v_{(2,1,1)}), (v_{(1,2,2)}, v_{(2,2,1)})\} \\ M_{2,3}^1 &= \{(v_{(1,1,2)}, v_{(2,1,1)}), (v_{(1,2,3)}, v_{(2,2,1)})\} & M_{2,4}^1 &= \{(v_{(1,1,2)}, v_{(2,1,1)}), (v_{(1,2,4)}, v_{(2,2,2)})\} \\ M_{3,1}^1 &= \{(v_{(1,1,3)}, v_{(2,1,1)}), (v_{(1,2,1)}, v_{(2,2,2)})\} & M_{3,2}^1 &= \{(v_{(1,1,3)}, v_{(2,1,1)}), (v_{(1,2,2)}, v_{(2,2,1)})\} \\ M_{3,3}^1 &= \{(v_{(1,1,3)}, v_{(2,1,1)}), (v_{(1,2,3)}, v_{(2,2,1)})\} & M_{3,4}^1 &= \{(v_{(1,1,3)}, v_{(2,1,1)}), (v_{(1,2,4)}, v_{(2,2,2)})\} \\ M_{4,1}^1 &= \{(v_{(1,1,4)}, v_{(2,1,2)}), (v_{(1,2,1)}, v_{(2,2,2)})\} & M_{4,2}^1 &= \{(v_{(1,1,4)}, v_{(2,1,2)}), (v_{(1,2,2)}, v_{(2,2,1)})\} \\ M_{4,3}^1 &= \{(v_{(1,1,4)}, v_{(2,1,2)}), (v_{(1,2,3)}, v_{(2,2,1)})\} & M_{4,4}^1 &= \{(v_{(1,1,4)}, v_{(2,1,2)}), (v_{(1,2,4)}, v_{(2,2,2)})\} \\ M_{1,1}^2 &= \{(v_{(1,1,1)}, v_{(2,1,4)}), (v_{(1,2,1)}, v_{(2,2,4)})\} & M_{1,2}^2 &= \{(v_{(1,1,1)}, v_{(2,1,4)}), (v_{(1,2,2)}, v_{(2,2,3)})\} \\ M_{1,3}^2 &= \{(v_{(1,1,1)}, v_{(2,1,4)}), (v_{(1,2,3)}, v_{(2,2,4)})\} & M_{1,4}^2 &= \{(v_{(1,1,1)}, v_{(2,1,4)}), (v_{(1,2,4)}, v_{(2,2,3)})\} \\ M_{2,1}^2 &= \{(v_{(1,1,2)}, v_{(2,1,3)}), (v_{(1,2,1)}, v_{(2,2,4)})\} & M_{2,2}^2 &= \{(v_{(1,1,2)}, v_{(2,1,3)}), (v_{(1,2,2)}, v_{(2,2,3)})\} \end{aligned}$$

$$\begin{aligned}
M_{2,3}^2 &= \{(v_{(1,1,2)}, v_{(2,1,3)}), (v_{(1,2,3)}, v_{(2,2,4)})\} & M_{2,4}^2 &= \{(v_{(1,1,2)}, v_{(2,1,3)}), (v_{(1,2,4)}, v_{(2,2,3)})\} \\
M_{3,1}^2 &= \{(v_{(1,1,3)}, v_{(2,1,4)}), (v_{(1,2,1)}, v_{(2,2,4)})\} & M_{3,2}^2 &= \{(v_{(1,1,3)}, v_{(2,1,4)}), (v_{(1,2,2)}, v_{(2,2,3)})\} \\
M_{3,3}^2 &= \{(v_{(1,1,3)}, v_{(2,1,4)}), (v_{(1,2,3)}, v_{(2,2,4)})\} & M_{3,4}^2 &= \{(v_{(1,1,3)}, v_{(2,1,4)}), (v_{(1,2,4)}, v_{(2,2,3)})\} \\
M_{4,1}^2 &= \{(v_{(1,1,4)}, v_{(2,1,3)}), (v_{(1,2,1)}, v_{(2,2,4)})\} & M_{4,2}^2 &= \{(v_{(1,1,4)}, v_{(2,1,3)}), (v_{(1,2,2)}, v_{(2,2,3)})\} \\
M_{4,3}^2 &= \{(v_{(1,1,4)}, v_{(2,1,3)}), (v_{(1,2,3)}, v_{(2,2,4)})\} & M_{4,4}^2 &= \{(v_{(1,1,4)}, v_{(2,1,3)}), (v_{(1,2,4)}, v_{(2,2,3)})\}.
\end{aligned}$$

For each matchings M_{j_1, j_2}^n for $n = 1, 2$, $1 \leq j_1, j_2 \leq 4$, we transmit one subfile Y_{j_1, j_2}^n . For a matching $M_{j_1, j_2}^n = \{(v_{(1,1,j_1)}, v_{(2,1,j_1^*)}), (v_{(1,2,j_2)}, v_{(2,2,j_2^*)})\}$, define $S_{j_1, j_2}^{n,1} = B(1, j_1^*) \cap B(2, j_2)$ and $S_{j_1, j_2}^{n,2} = B(1, j_1) \cap B(2, j_2^*)$. We transmit the following subfiles

$$Y_{j_1, j_2}^n = W^{d_{f_{M_1}^{-1}(c(1, j_1))}}(S_{j_1, j_2}^{n,1}) + W^{d_{f_{M_2}^{-1}(c(2, j_2))}}(S_{j_1, j_2}^{n,2}) \text{ for } 1 \leq n \leq 2, 1 \leq j_1, j_2 \leq 4.$$

The transmissions Y_{j_1, j_2}^n along with the set $S_{j_1, j_2}^{n,i}$ for $n, i = 1, 2$, $1 \leq j_1, j_2 \leq 4$, are shown in the following.

$$\begin{aligned}
S_{1,1}^{1,1} &= B(1, 2) \cap B(2, 1) = \{5\}, & S_{1,1}^{1,2} &= B(1, 1) \cap B(2, 2) = \{2\}, & Y_{1,1}^1 &= W^{d(1,1)}(5) + W^{d(2,4)}(2) & (5) \\
S_{1,2}^{1,1} &= B(1, 2) \cap B(2, 2) = \{6\}, & S_{1,2}^{1,2} &= B(1, 1) \cap B(2, 1) = \{1\}, & Y_{1,2}^1 &= W^{d(1,1)}(6) + W^{d(2,3)}(1) \\
S_{1,3}^{1,1} &= B(1, 2) \cap B(2, 3) = \{7\}, & S_{1,3}^{1,2} &= B(1, 1) \cap B(2, 1) = \{1\}, & Y_{1,3}^1 &= W^{d(1,1)}(7) + W^{d(2,2)}(1) \\
S_{1,4}^{1,1} &= B(1, 2) \cap B(2, 4) = \{8\}, & S_{1,4}^{1,2} &= B(1, 1) \cap B(2, 2) = \{2\}, & Y_{1,4}^1 &= W^{d(1,1)}(8) + W^{d(2,1)}(2) \\
S_{2,1}^{1,1} &= B(1, 1) \cap B(2, 1) = \{1\}, & S_{2,1}^{1,2} &= B(1, 2) \cap B(2, 2) = \{6\}, & Y_{2,1}^1 &= W^{d(1,2)}(1) + W^{d(2,4)}(6) \\
S_{2,2}^{1,1} &= B(1, 1) \cap B(2, 2) = \{2\}, & S_{2,2}^{1,2} &= B(1, 2) \cap B(2, 1) = \{5\}, & Y_{2,2}^1 &= W^{d(1,2)}(2) + W^{d(2,3)}(1) \\
S_{2,3}^{1,1} &= B(1, 1) \cap B(2, 3) = \{3\}, & S_{2,3}^{1,2} &= B(1, 2) \cap B(2, 1) = \{5\}, & Y_{2,3}^1 &= W^{d(1,2)}(3) + W^{d(2,2)}(5) \\
S_{2,4}^{1,1} &= B(1, 1) \cap B(2, 4) = \{4\}, & S_{2,4}^{1,2} &= B(1, 2) \cap B(2, 2) = \{6\}, & Y_{2,4}^1 &= W^{d(1,2)}(4) + W^{d(2,1)}(6) \\
S_{3,1}^{1,1} &= B(1, 1) \cap B(2, 1) = \{1\}, & S_{3,1}^{1,2} &= B(1, 3) \cap B(2, 2) = \{10\}, & Y_{3,1}^1 &= W^{d(1,4)}(1) + W^{d(2,4)}(10) \\
S_{3,2}^{1,1} &= B(1, 1) \cap B(2, 2) = \{2\}, & S_{3,2}^{1,2} &= B(1, 3) \cap B(2, 1) = \{9\}, & Y_{3,2}^1 &= W^{d(1,4)}(2) + W^{d(2,3)}(9) \\
S_{3,3}^{1,1} &= B(1, 1) \cap B(2, 3) = \{3\}, & S_{3,3}^{1,2} &= B(1, 3) \cap B(2, 1) = \{9\}, & Y_{3,3}^1 &= W^{d(1,4)}(3) + W^{d(2,2)}(9) \\
S_{3,4}^{1,1} &= B(1, 1) \cap B(2, 4) = \{4\}, & S_{3,4}^{1,2} &= B(1, 3) \cap B(2, 2) = \{10\}, & Y_{3,4}^1 &= W^{d(1,4)}(4) + W^{d(2,1)}(10) \\
S_{4,1}^{1,1} &= B(1, 2) \cap B(2, 1) = \{5\}, & S_{4,1}^{1,2} &= B(1, 4) \cap B(2, 2) = \{14\}, & Y_{4,1}^1 &= W^{d(1,3)}(5) + W^{d(2,4)}(14) \\
S_{4,2}^{1,1} &= B(1, 2) \cap B(2, 2) = \{6\}, & S_{4,2}^{1,2} &= B(1, 4) \cap B(2, 1) = \{13\}, & Y_{4,2}^1 &= W^{d(1,3)}(6) + W^{d(2,3)}(13) \\
S_{4,3}^{1,1} &= B(1, 2) \cap B(2, 3) = \{7\}, & S_{4,3}^{1,2} &= B(1, 4) \cap B(2, 1) = \{13\}, & Y_{4,3}^1 &= W^{d(1,3)}(7) + W^{d(2,2)}(13) \\
S_{4,4}^{1,1} &= B(1, 2) \cap B(2, 4) = \{8\}, & S_{4,4}^{1,2} &= B(1, 4) \cap B(2, 2) = \{14\}, & Y_{4,4}^1 &= W^{d(1,3)}(8) + W^{d(2,1)}(14) \\
S_{1,1}^{2,1} &= B(1, 4) \cap B(2, 1) = \{13\}, & S_{1,1}^{2,2} &= B(1, 1) \cap B(2, 4) = \{4\}, & Y_{1,1}^2 &= W^{d(1,1)}(13) + W^{d(2,4)}(4) \\
S_{1,2}^{2,1} &= B(1, 4) \cap B(2, 2) = \{14\}, & S_{1,2}^{2,2} &= B(1, 1) \cap B(2, 3) = \{3\}, & Y_{1,2}^2 &= W^{d(1,1)}(14) + W^{d(2,3)}(3) \\
S_{1,3}^{2,1} &= B(1, 4) \cap B(2, 3) = \{15\}, & S_{1,3}^{2,2} &= B(1, 1) \cap B(2, 4) = \{4\}, & Y_{1,3}^2 &= W^{d(1,1)}(15) + W^{d(2,2)}(4) \\
S_{1,4}^{2,1} &= B(1, 4) \cap B(2, 4) = \{16\}, & S_{1,4}^{2,2} &= B(1, 1) \cap B(2, 3) = \{3\}, & Y_{1,4}^2 &= W^{d(1,1)}(16) + W^{d(2,1)}(3)
\end{aligned}$$

$$\begin{aligned}
S_{2,1}^{2,1} &= B(1,3) \cap B(2,1) = \{9\}, & S_{2,1}^{2,2} &= B(1,2) \cap B(2,4) = \{8\}, & Y_{2,1}^2 &= W^{d(1,2)}(9) + W^{d(2,4)}(8) \\
S_{2,2}^{2,1} &= B(1,3) \cap B(2,2) = \{10\}, & S_{2,2}^{2,2} &= B(1,2) \cap B(2,3) = \{7\}, & Y_{2,2}^2 &= W^{d(1,2)}(10) + W^{d(2,3)}(7) \\
S_{2,3}^{2,1} &= B(1,3) \cap B(2,3) = \{11\}, & S_{2,3}^{2,2} &= B(1,2) \cap B(2,4) = \{8\}, & Y_{2,3}^2 &= W^{d(1,2)}(11) + W^{d(2,2)}(8) \\
S_{2,4}^{2,1} &= B(1,3) \cap B(2,4) = \{12\}, & S_{2,4}^{2,2} &= B(1,2) \cap B(2,3) = \{7\}, & Y_{2,4}^2 &= W^{d(1,2)}(12) + W^{d(2,1)}(7) \\
S_{3,1}^{2,1} &= B(1,4) \cap B(2,1) = \{13\}, & S_{3,1}^{2,2} &= B(1,3) \cap B(2,4) = \{12\}, & Y_{3,1}^2 &= W^{d(1,4)}(13) + W^{d(2,4)}(12) \\
S_{3,2}^{2,1} &= B(1,4) \cap B(2,2) = \{14\}, & S_{3,2}^{2,2} &= B(1,3) \cap B(2,3) = \{11\}, & Y_{3,2}^2 &= W^{d(1,4)}(14) + W^{d(2,3)}(11) \\
S_{3,3}^{2,1} &= B(1,4) \cap B(2,3) = \{15\}, & S_{3,3}^{2,2} &= B(1,3) \cap B(2,4) = \{12\}, & Y_{3,3}^2 &= W^{d(1,4)}(15) + W^{d(2,2)}(12) \\
S_{3,4}^{2,1} &= B(1,4) \cap B(2,4) = \{16\}, & S_{3,4}^{2,2} &= B(1,3) \cap B(2,3) = \{11\}, & Y_{3,4}^2 &= W^{d(1,4)}(16) + W^{d(2,1)}(11) \\
S_{4,1}^{2,1} &= B(1,3) \cap B(2,1) = \{9\}, & S_{4,1}^{2,2} &= B(1,4) \cap B(2,4) = \{16\}, & Y_{4,1}^2 &= W^{d(1,3)}(9) + W^{d(2,4)}(16) \\
S_{4,2}^{2,1} &= B(1,3) \cap B(2,2) = \{10\}, & S_{4,2}^{2,2} &= B(1,4) \cap B(2,3) = \{15\}, & Y_{4,2}^2 &= W^{d(1,3)}(10) + W^{d(2,3)}(15) \\
S_{4,3}^{2,1} &= B(1,3) \cap B(2,3) = \{11\}, & S_{4,3}^{2,2} &= B(1,4) \cap B(2,4) = \{16\}, & Y_{4,3}^2 &= W^{d(1,3)}(11) + W^{d(2,2)}(16) \\
S_{4,4}^{2,1} &= B(1,3) \cap B(2,4) = \{12\}, & S_{4,4}^{2,2} &= B(1,4) \cap B(2,3) = \{15\}, & Y_{4,4}^2 &= W^{d(1,3)}(12) + W^{d(2,1)}(15).
\end{aligned}$$

Each transmission benefits two users, for instance, from equation (5), since $2 \in B(1,1)$, $B(1,1) \in B_{c(1,1)}$, $c(1,1) \in C_{k(1,1)}$, user $k(1,1)$ knows $W^{d(2,4)}(2)$, so it can retrieve $W^{d(1,1)}(5)$; similarly, and since $5 \in B(2,1)$, $B(2,1) \in B_{c(2,1)}$, $c(2,1) \in C_{k(2,4)}$, user $k(2,4)$ knows $W^{d(1,1)}(5)$, so it can retrieve $W^{d(2,4)}(2)$. After all the above transmissions, it can be verified that all the users can retrieve its demanded files.

B. Second Example: $K = 14$, $z = 3$, $\frac{M}{N} = \frac{2}{7}$

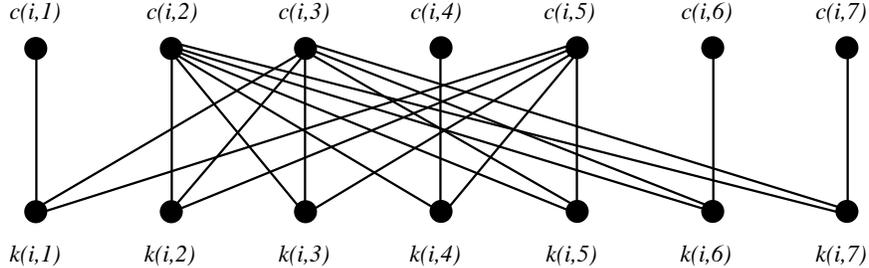


Fig. 4. Let the graph shown in the figure be denoted by J_i . The user-to-cache association bipartite graph $J = (U, C, E)$ considered in the second example in Section VII-B is the disjoint union of J_1 and J_2 . It can be verified that $J = (U, C, E)$ satisfies conditions **C1**, **C2**, **C3** for $b = 7$, $m = 2$, $z = 3$.

In the second example, we consider a MACC problem with $K = 14$, $z = 3$, $\frac{M}{N} = \frac{2}{7}$. For $b = 7$, $m = 2$, $z = 3$, let the user-to-cache association bipartite graph be $J = (U, C, E)$, an abstraction of which is shown in Fig. 4. It is to be noted that there are other user-to-cache association bipartite graphs that satisfies the criteria set by this example problem. We have $U = K_1 \cup K_2$ where for $i = 1, 2$, $K_i = \{k(i, j) \mid 1 \leq j \leq 7\}$. And, $C = C_1 \cup C_2$ where for $i = 1, 2$, $C_i = \{c(i, j) \mid 1 \leq j \leq 7\}$.

Let us consider the following definition of f_{M_i} : for $i = 1, 2$, $1 \leq j \leq 7$, $f_{M_i}(k(i, j)) = c(i, j)$. As per condition **C2**, $C_i = C_{(i,1)} \cup C_{(i,2)} \cup C_{(i,3)}$, where

$$C_{(i,1)} = \{c(i, 1), c(i, 2)\}, C_{(i,2)} = \{c(i, 3), c(i, 4)\}, C_{(i,3)} = \{c(i, 5), c(i, 6), c(i, 7)\} \text{ for } 1 \leq i \leq 2.$$

For $i = 1, 2$, $1 \leq j \leq 7$, set $C_{k(i,j)}$ denotes the set of caches user $k(i, j)$ accesses. We have

$$\begin{aligned} C_{k(i,1)} &= \{c(i, 1), c(i, 3), c(i, 5)\}, & C_{k(i,2)} &= \{c(i, 2), c(i, 3), c(i, 5)\}, & C_{k(i,3)} &= \{c(i, 2), c(i, 3), c(i, 5)\}, \\ C_{k(i,4)} &= \{c(i, 2), c(i, 4), c(i, 5)\}, & C_{k(i,5)} &= \{c(i, 2), c(i, 3), c(i, 5)\}, & C_{k(i,6)} &= \{c(i, 2), c(i, 3), c(i, 6)\}, \\ C_{k(i,7)} &= \{c(i, 2), c(i, 3), c(i, 7)\}. \end{aligned}$$

We consider the MCRD with the following parallel classes to be used for placement and delivery.

$\mathcal{P}_i = \{B(i, j) \mid 1 \leq j \leq 7\}$ for $i = 1, 2$ where

$$\begin{aligned} B(1, 1) &= \{1, 2, 3, 4, 5, 6, 7\}, & B(1, 2) &= \{8, 9, 10, 11, 12, 13, 14\}, & B(1, 3) &= \{15, 16, 17, 18, 19, 20, 21\}, \\ B(1, 4) &= \{22, 23, 24, 25, 26, 27, 28\}, & B(1, 5) &= \{29, 30, 31, 32, 33, 34, 35\}, & B(1, 6) &= \{36, 37, 38, 39, 40, 41, 42\}, \\ B(1, 7) &= \{43, 44, 45, 46, 47, 48, 49\}, & B(2, 1) &= \{1, 8, 15, 22, 29, 36, 43\}, & B(2, 2) &= \{2, 9, 16, 23, 30, 37, 44\}, \\ B(2, 3) &= \{3, 10, 17, 24, 31, 38, 45\}, & B(2, 4) &= \{4, 11, 18, 25, 32, 39, 46\}, & B(2, 5) &= \{5, 12, 19, 26, 33, 40, 47\}, \\ B(2, 6) &= \{6, 13, 20, 27, 34, 41, 48\}, & B(2, 7) &= \{7, 14, 21, 28, 35, 42, 49\}. \end{aligned}$$

Partition sets \mathcal{P}_1 and \mathcal{P}_2 as following

$$\mathcal{P}_{(i,1)} = \{B(i, 1), B(i, 2)\}, \mathcal{P}_{(i,2)} = \{B(i, 3), B(i, 4)\}, \mathcal{P}_{(i,3)} = \{B(i, 5), B(i, 6), B(i, 7)\} \text{ for } 1 \leq i \leq 2.$$

Placement: Split each file into $b^m = 49$ subfiles. The cache $c(i, j)$ stores the subfiles (of all files) indexed by the elements of the blocks contained in $B_{c(i,j)}$, where for $i = 1, 2$,

$$\begin{aligned} B_{c(i,1)} &= B_{c(i,2)} = \{B(i, 1), B(i, 2)\}, & B_{c(i,3)} &= B_{c(i,4)} = \{B(i, 3), B(i, 4)\}, \\ B_{c(i,5)} &= B_{c(i,6)} = \{B(i, 5), B(i, 6)\}, & B_{c(i,7)} &= \{B(i, 5), B(i, 7)\}. \end{aligned}$$

So we have for $i = 1, 2$,

$$\begin{aligned} B_{k(i,1)} &= \mathcal{P}_i \setminus B(i, 7), & B_{k(i,2)} &= \mathcal{P}_i \setminus B(i, 7), & B_{k(i,3)} &= \mathcal{P}_i \setminus B(i, 7), & B_{k(i,4)} &= \mathcal{P}_i \setminus B(i, 7), \\ B_{k(i,5)} &= \mathcal{P}_i \setminus B(i, 7), & B_{k(i,6)} &= \mathcal{P}_i \setminus B(i, 7), & B_{k(i,7)} &= \mathcal{P}_i \setminus B(i, 6). \end{aligned}$$

We now construct a new graph that we call as the demand graph $\bar{J} = (V_1, V_2, \bar{E})$, an abstraction of which is shown in Fig. 5. We have for $l = 1, 2$, $V_l = \{v_{(l,i,j)} \mid 1 \leq i \leq 2, 1 \leq j \leq 7\}$. There is a bijection from the vertices in V_1 to the caches in C , and there is another bijection from the vertices in V_2 to the blocks in $\mathcal{P}_1 \cup \mathcal{P}_2$. Say for $i = 1, 2$, $1 \leq j \leq 7$, the vertex $v_{(1,i,j)}$ maps to the cache $c(i, j)$, similarly, say that the vertex $v_{(2,i,j)}$ maps to the block $B(i, j)$. For $i = 1, 2$, $1 \leq j \leq 7$ there exists an edge in \bar{E} connecting vertex $v_{(1,i,j)} \in V_1$ to vertex $v_{(2,\bar{i},j^*)} \in V_2$ if and only if $i = \bar{i}$ and $B(i, j^*) \notin B_{k(i,j')}$ where $k(i, j') = f_{M_i}^{-1}(c(i, j))$.

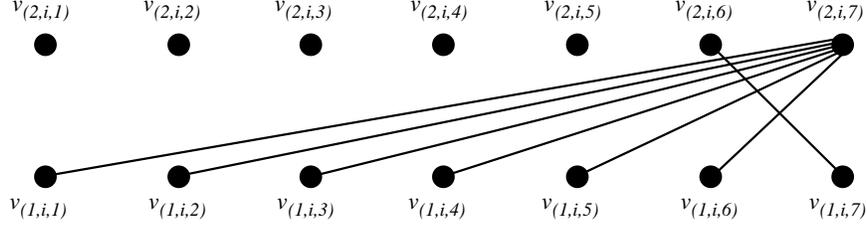


Fig. 5. Let the graph shown in the figure be denoted by \bar{J}_i . The demand graph $\bar{J} = (V_1, V_2, \bar{E})$ for the user-to-cache association bipartite graph $J = (U, C, E)$ considered in the second example in Section VII-B is the disjoint union of \bar{J}_1 and \bar{J}_2 . An edge connects $v_{(1,i,j)}$ to $v_{(2,i',j^*)}$ if and only if $i = i'$ and the set $B_{f_{M_i}^{-1}(c(i,j))}$ does not contain the block $B(i, j^*)$.

We define a map $f_{(i,j)} : \{1\} \rightarrow \{1, 2, 3, 4, 5, 6, 7\} \setminus \{l \mid B(i, l) \in B_{k(i,j)}\}$ for $1 \leq i \leq 2, 1 \leq j \leq 7$ as following.

$$f_{(i,j)}(1) = 7 \text{ for } i = 1, 2, 1 \leq j \leq 6$$

$$f_{(i,7)}(1) = 6 \text{ for } i = 1, 2.$$

We define a partial matching M_{j_1, j_2}^1 for $1 \leq j_1, j_2 \leq 7$ as following.

$$M_{j_1, j_2}^i = \{(v_{(1,i,j_i)}, v_{(2,i,f_{(i,j_i)}(1))}) \mid i = 1, 2\} \text{ where } k(i, j_i) = f_{M_i}^{-1}(c(i, j_i)).$$

So we have

$$M_{j_1, j_2}^1 = \{(v_{(1,1,j_1)}, v_{(2,1,7)}), (v_{(1,2,j_2)}, v_{(2,2,7)})\} \text{ for } 1 \leq j_1, j_2 \leq 6$$

$$M_{j_1, 7}^1 = \{(v_{(1,1,j_1)}, v_{(2,1,7)}), (v_{(1,2,7)}, v_{(2,2,6)})\} \text{ for } 1 \leq j_1 \leq 6$$

$$M_{7, j_2}^1 = \{(v_{(1,1,7)}, v_{(2,1,6)}), (v_{(1,2,j_2)}, v_{(2,2,7)})\} \text{ for } 1 \leq j_2 \leq 6$$

$$M_{7, 7}^1 = \{(v_{(1,1,7)}, v_{(2,1,6)}), (v_{(1,2,7)}, v_{(2,2,6)})\}.$$

For each matching M_{j_1, j_2}^1 for $1 \leq j_1, j_2 \leq 7$, we transmit one subfile Y_{j_1, j_2}^1 . For a matching $M_{j_1, j_2}^1 = \{(v_{(1,1,j_1)}, v_{(2,1,j_1^*)}), (v_{(1,2,j_2)}, v_{(2,2,j_2^*)})\}$, define $S_{j_1, j_2}^{1,1} = B(1, j_1^*) \cap B(2, j_2)$ and $S_{j_1, j_2}^{1,2} = B(1, j_1) \cap B(2, j_2^*)$. We transmit the following subfiles

$$Y_{j_1, j_2}^1 = W_{f_{M_1}^{-1}(c(1,j_1))}^{d_{f_{M_1}^{-1}(c(1,j_1))}}(S_{j_1, j_2}^{1,1}) + W_{f_{M_2}^{-1}(c(2,j_2))}^{d_{f_{M_2}^{-1}(c(2,j_2))}}(S_{j_1, j_2}^{1,2}) \text{ for } 1 \leq j_1, j_2 \leq 7.$$

The transmissions Y_{j_1, j_2}^1 along with the set $S_{j_1, j_2}^{1,i}$ for $i = 1, 2, 1 \leq j_1, j_2 \leq 7$, are shown in the following.

$$S_{1,1}^{1,1} = B(1, 7) \cap B(2, 1) = \{43\}, \quad S_{1,1}^{1,2} = B(1, 1) \cap B(2, 7) = \{7\}, \quad Y_{1,1}^1 = W^{d_{(1,1)}}(43) + W^{d_{(2,1)}}(7)$$

$$S_{1,2}^{1,1} = B(1, 7) \cap B(2, 2) = \{44\}, \quad S_{1,2}^{1,2} = B(1, 1) \cap B(2, 7) = \{7\}, \quad Y_{1,2}^1 = W^{d_{(1,1)}}(44) + W^{d_{(2,2)}}(7)$$

$$S_{1,3}^{1,1} = B(1, 7) \cap B(2, 3) = \{45\}, \quad S_{1,3}^{1,2} = B(1, 1) \cap B(2, 7) = \{7\}, \quad Y_{1,3}^1 = W^{d_{(1,1)}}(45) + W^{d_{(2,3)}}(7)$$

$$S_{1,4}^{1,1} = B(1, 7) \cap B(2, 4) = \{46\}, \quad S_{1,4}^{1,2} = B(1, 1) \cap B(2, 7) = \{7\}, \quad Y_{1,4}^1 = W^{d_{(1,1)}}(46) + W^{d_{(2,4)}}(7)$$

$$S_{1,5}^{1,1} = B(1, 7) \cap B(2, 5) = \{47\}, \quad S_{1,5}^{1,2} = B(1, 1) \cap B(2, 7) = \{7\}, \quad Y_{1,5}^1 = W^{d_{(1,1)}}(47) + W^{d_{(2,5)}}(7)$$

$$S_{1,6}^{1,1} = B(1, 7) \cap B(2, 6) = \{48\}, \quad S_{1,6}^{1,2} = B(1, 1) \cap B(2, 7) = \{7\}, \quad Y_{1,6}^1 = W^{d_{(1,1)}}(48) + W^{d_{(2,6)}}(7)$$

$$\begin{aligned}
S_{6,1}^{1,1} &= B(1,7) \cap B(2,1) = \{43\}, & S_{6,1}^{1,2} &= B(1,6) \cap B(2,7) = \{42\}, & Y_{6,1}^1 &= W^{d(1,6)}(43) + W^{d(2,1)}(42) \\
S_{6,2}^{1,1} &= B(1,7) \cap B(2,2) = \{44\}, & S_{6,2}^{1,2} &= B(1,6) \cap B(2,7) = \{42\}, & Y_{6,2}^1 &= W^{d(1,6)}(44) + W^{d(2,2)}(42) \\
S_{6,3}^{1,1} &= B(1,7) \cap B(2,3) = \{45\}, & S_{6,3}^{1,2} &= B(1,6) \cap B(2,7) = \{42\}, & Y_{6,3}^1 &= W^{d(1,6)}(45) + W^{d(2,3)}(42) \\
S_{6,4}^{1,1} &= B(1,7) \cap B(2,4) = \{46\}, & S_{6,4}^{1,2} &= B(1,6) \cap B(2,7) = \{42\}, & Y_{6,4}^1 &= W^{d(1,6)}(46) + W^{d(2,4)}(42) \\
S_{6,5}^{1,1} &= B(1,7) \cap B(2,5) = \{47\}, & S_{6,5}^{1,2} &= B(1,6) \cap B(2,7) = \{42\}, & Y_{6,5}^1 &= W^{d(1,6)}(47) + W^{d(2,5)}(42) \\
S_{6,6}^{1,1} &= B(1,7) \cap B(2,6) = \{48\}, & S_{6,6}^{1,2} &= B(1,6) \cap B(2,7) = \{42\}, & Y_{6,6}^1 &= W^{d(1,6)}(48) + W^{d(2,6)}(42) \\
S_{6,7}^{1,1} &= B(1,7) \cap B(2,7) = \{49\}, & S_{6,7}^{1,2} &= B(1,6) \cap B(2,6) = \{41\}, & Y_{6,7}^1 &= W^{d(1,6)}(49) + W^{d(2,7)}(41) \\
S_{7,1}^{1,1} &= B(1,6) \cap B(2,1) = \{36\}, & S_{7,1}^{1,2} &= B(1,7) \cap B(2,7) = \{49\}, & Y_{7,1}^1 &= W^{d(1,7)}(36) + W^{d(2,1)}(49) \\
S_{7,2}^{1,1} &= B(1,6) \cap B(2,2) = \{37\}, & S_{7,2}^{1,2} &= B(1,7) \cap B(2,7) = \{49\}, & Y_{7,2}^1 &= W^{d(1,7)}(37) + W^{d(2,2)}(49) \\
S_{7,3}^{1,1} &= B(1,6) \cap B(2,3) = \{38\}, & S_{7,3}^{1,2} &= B(1,7) \cap B(2,7) = \{49\}, & Y_{7,3}^1 &= W^{d(1,7)}(38) + W^{d(2,3)}(49) \\
S_{7,4}^{1,1} &= B(1,6) \cap B(2,4) = \{39\}, & S_{7,4}^{1,2} &= B(1,7) \cap B(2,7) = \{49\}, & Y_{7,4}^1 &= W^{d(1,7)}(39) + W^{d(2,4)}(49) \\
S_{7,5}^{1,1} &= B(1,6) \cap B(2,5) = \{40\}, & S_{7,5}^{1,2} &= B(1,7) \cap B(2,7) = \{49\}, & Y_{7,5}^1 &= W^{d(1,7)}(40) + W^{d(2,5)}(49) \\
S_{7,6}^{1,1} &= B(1,6) \cap B(2,6) = \{41\}, & S_{7,6}^{1,2} &= B(1,7) \cap B(2,7) = \{49\}, & Y_{7,6}^1 &= W^{d(1,7)}(41) + W^{d(2,6)}(49) \\
S_{7,7}^{1,1} &= B(1,6) \cap B(2,7) = \{42\}, & S_{7,7}^{1,2} &= B(1,7) \cap B(2,6) = \{48\}, & Y_{7,7}^1 &= W^{d(1,7)}(42) + W^{d(2,7)}(48).
\end{aligned}$$

After all the above transmissions, it can be verified that all the users can retrieve its demanded files.

VIII. CONCLUSION

Most works on the MACC problem (which assumes the number of users to be the same as the number of caches) in the literature assumes a cyclic wrap around the user-to-cache association, which has been introduced in [6]. A natural question is whether some other user-to-cache association can provide better results. This question has been studied very recently in [18] for the MACC problem where the number of users is more than the number of caches (other than the trivial cases). We address this question for the MACC problem, which has the same number of users and caches. We show that the user-to-cache associations considered in this paper can provide better results than all other existing MACC schemes in some aspects.

APPENDIX A

PROOFS OF LEMMA 1 AND COROLLARY 1

A. Proof of Lemma 1

Proof: For $1 \leq i \leq m$, $0 \leq j \leq b-1$, let $B(i, j)$ be the j^{th} block in the i^{th} parallel class for some ordering of the parallel classes and the blocks in each parallel class. For $1 \leq n \leq m$, $n \neq i$, $1 \leq l_n \leq b$, we have $\bigcap_{n=1, n \neq i}^m B(n, l_n) \cap B(i, j) = \mu_m$. Furthermore, if $l \neq l'$, then $B(n, l) \cap B(n, l') = \emptyset$, as blocks in the parallel class \mathcal{P}_x partitions X . As a result, for $1 \leq n \leq m$, $n \neq i$, $1 \leq l'_n \leq b$, if at least for some value of $1 \leq x \leq m$, $x \neq i$, we have $l_x \neq l'_x$, then $\{\bigcap_{n=1, n \neq i}^m B(n, l_n) \cap B(i, j)\} \cap \{\bigcap_{n=1, n \neq i}^m B(n, l'_n) \cap B(i, j)\} = \emptyset$. Since for $1 \leq n \leq m$,

$n \neq i$, l_n can be chose in b ways, there are b^{m-1} instances of the set $\cap_{n=1, n \neq i}^m B(n, l_n)$. Furthermore these sets are all disjoint as argued above. With each such set, $B(i, j)$ has μ_m distinct elements in common. So block $B(i, j)$ must have at least $\mu_m b^{m-1}$ distinct elements. Let V be the set of these $\mu_m b^{m-1}$ elements.

We now show that block $B(i, j)$ does not contain any other element. Let $a \in X$ be an element such that $a \notin V$ but $a \in B(i, j)$. Then in each parallel class \mathcal{P}_n , $1 \leq n \leq m$, $n \neq i$, there is a block $B(n, l_n)$, where $1 \leq l_n \leq b$, such that $a \in B(n, l_n)$ (as blocks in \mathcal{P}_n partitions X). So $a \in \cap_{n=1, n \leq i}^m B(n, l_n) \cap B(i, j)$. Then a must belong to the set V . This contradicts the assumption that $a \notin V$.

Since there are b blocks in each parallel class and each block has $\mu_m b^{m-1}$ elements, X must have exactly $\mu_m b^m$ elements. ■

B. Proof of Corollary 1

Proof: Even though the proof this corollary is contained in the proof Lemma 1, we provide an alternate proof here. Say for any positive integer m , $[m]_i = \{1, 2, \dots, m\} \setminus \{i\}$. Let $[x]_i \subseteq [m]_i$ where $x \leq m$. Then, $I([x]_i, j) = I(\{l_y | y \in [x]_i\} \cup j)$. We show the following results holds: $I([x]_i, j) = I([x-1]_i, j)$.

$$\begin{aligned}
& \cup_{\forall y \in [x]} \cup_{l_y=1}^b I([x]_i, j) \\
&= \cup_{\forall y \in [x-1]} \cup_{l_y=1}^b \cup_{l_x=1}^b \{I([x-1]_i, j) \cap B(x, l_x)\} \\
&= \cup_{\forall y \in [x-1]} \cup_{l_y=1}^b \{I([x-1]_i, j) \cap B(x, 1)\} \cup \{I([x-1]_i, j) \cap B(x, 2)\} \cup \dots \cup \{I([x-1]_i, j) \cap B(x, m)\} \\
&= \cup_{\forall y \in [x-1]} \cup_{l_y=1}^b I([x-1]_i, j) \cap \{B(x, 1) \cup B(x, 2) \cup \dots \cup B(x, m)\} \\
&= \cup_{\forall y \in [x-1]} \cup_{l_y=1}^b I([x-1]_i, j) \cap X \\
&= \cup_{\forall y \in [x-1]} \cup_{l_y=1}^b I([x-1]_i, j).
\end{aligned}$$

Similarly, it can be shown that $I([x-1]_i, j) = I([x-2]_i, j)$. And hence $I([x]_i, j) = I([x-2]_i, j)$. Thus, proceeding in the same manner would yield $I([x]_i, j) = I([x-(x-1)]_i, j)$. For some $w \in [x]_i$, we have $I([x-(x-1)]_i, j) = I(w, j)$. Now,

$$\cup_{l_w=1}^b I(l_w, j) = B(i, j) \cap \{B(w, 1) \cup B(w, 2) \cup \dots \cup B(w, m)\} = B(i, j) \cap X = B(i, j)$$
■

APPENDIX B

PROOF OF THEOREM 1

In reference [16] the authors show a new class of CRD. A special case of these CRDs result in MCRD. These MCRDs has the same parameters as that claimed in this theorem. So there already exists a proof for the theorem. In this section we provide an alternate proof. We remark that we have obtained our proof independently to [16]. We also state that our proof is motivated by the work of Tang and Ramamoorthy in [19].

For any positive integer q , let \mathbb{Z}_q be the ring of integers modulo q . Let \mathbb{Z}_q^m be the set of all m -length column vectors over \mathbb{Z}_q . It can be seen that \mathbb{Z}_q^m has total q^m distinct column vectors. Create a matrix $M_{\{m, q\}}$ by listing all

of these q^m distinct column vectors (in any arbitrary order) as columns of $M_{\{m,q\}}$. Hence $M_{\{m,q\}}$ is an $m \times q^m$ matrix. For any matrix M , let $M(i, j)$ denote the value at the i^{th} row and j^{th} column of M (with the row and column indices starting from 1 and not 0). For $1 \leq i \leq m$, $l \in \mathbb{Z}_q$ we define the following mq sets.

$$B(i, l+1) = \{j \mid j \in \{1, 2, \dots, q^m\}, M_{m,q}(i, j) = l\}.$$

In words, $B(i, l+1)$ contains all the column indices j such that the value of $M_{\{m,q\}}$ at the i^{th} row and j^{th} column is l .

Lemma 7. For any $1 \leq i \leq m$, and $l \in \mathbb{Z}_q$, set $B(i, l+1)$ contains q^{m-1} elements, i.e. $|B(i, l+1)| = q^{m-1}$.

Proof: First note \mathbb{Z}_q^m is group under vector addition. Since all distinct vectors of \mathbb{Z}_q^m are present as a column of $M_{\{m,q\}}$, the columns of $M_{\{m,q\}}$ form a group. Let V_i^l be the set of all columns of $M_{\{m,q\}}$ such that the value at the i^{th} row of all columns in V_i^l is $l \in \mathbb{Z}_q$. It can be seen that V_i^0 is a proper subgroup of \mathbb{Z}_q^m . Furthermore, V_i^0 has q cosets namely $V_i^0, V_i^1, V_i^2, \dots, V_i^{q-1}$. And, any of the q^m column vectors of $M_{\{m,q\}}$ must be contained in one of the q cosets V_i^l for $l = 0, 1, \dots, q-1$. As all cosets are of same size, each coset must have q^{m-1} column vectors.

As $B(i, l+1)$ contains all the column indices of the columns present in V_i^l , it is immediate that $|B(i, l+1)| = q^{m-1}$. ■

From $M_{\{m,q\}}$, we construct a design (X, \mathcal{A}) where $X = \{1, 2, \dots, q^m\}$, and $\mathcal{A} = \{B(i, l+1) \mid 1 \leq i \leq m, l \in \mathbb{Z}_q\}$.

Lemma 8. (X, \mathcal{A}) is a resolvable design.

Proof: We first define m parallel classes of (X, \mathcal{A}) . For $1 \leq i \leq m$, let $\mathcal{P}_i = \{B(i, l+1) \mid l \in \mathbb{Z}_q\}$. Since the i^{th} row of every column must contain one value from \mathbb{Z}_q , each column index in $\{1, 2, \dots, q^m\}$ must be present in exactly one set contained in \mathcal{P}_i . Thus \mathcal{P}_i partitions X .

Furthermore, as for $1 \leq i \leq m$, $l \in \mathbb{Z}_q$, $B(i, l+1) \in \mathcal{P}_i$, the sets $\{\mathcal{P}_1, \mathcal{P}_2, \dots, \mathcal{P}_m\}$ partitions \mathcal{A} . So (X, \mathcal{A}) is a resolvable design. ■

Lemma 9. (X, \mathcal{A}) is a maximal cross resolvable design (MCRD) with $\mu_m = 1$.

Proof: We have already shown that (X, \mathcal{A}) is a resolvable design. Say $l_1, l_2, \dots, l_m \in \mathbb{Z}_q$ be any m values from \mathbb{Z}_q . We show that $|\cap_{i=1}^m B(i, l_i+1)| = 1$.

Since every m -length vector over \mathbb{Z}_q appear exactly once as a column vector of $M_{\{m,q\}}$, for some $c \in \{1, 2, \dots, q^m\}$, the column vector at the c^{th} column is $\begin{bmatrix} l_1 & l_2 & \dots & l_m \end{bmatrix}^T$. Hence $c \in B(i, l_i+1)$ for $1 \leq i \leq m$, and as every vector occurs exactly once, there $\nexists c' \in \{1, 2, \dots, q^m\}$, $c' \neq c$, and $c' \in B(i, l_i+1)$ for $1 \leq i \leq m$. So $\cap_{i=1}^m B(i, l_i+1) = c$, and the lemma is proved. ■

Example 9.

$$M_{\{3,2\}} = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \end{bmatrix}$$

We have $B(1, 1) = \{1, 2, 3, 4\}$, $B(1, 2) = \{5, 6, 7, 8\}$, $B(2, 1) = \{1, 2, 5, 6\}$, $B(2, 2) = \{3, 4, 7, 8\}$,
 $B(3, 1) = \{1, 3, 5, 7\}$, $B(3, 2) = \{2, 4, 6, 8\}$. We now show the parallel classes.

$$\mathcal{P}_1 = \{\{1, 2, 3, 4\}, \{5, 6, 7, 8\}\}$$

$$\mathcal{P}_2 = \{\{1, 2, 5, 6\}, \{3, 4, 7, 8\}\}$$

$$\mathcal{P}_3 = \{\{1, 3, 5, 7\}, \{2, 4, 6, 8\}\}.$$

For $X = \{1, 2, \dots, 8\}$, $\mathcal{A} = \{\mathcal{P}_1, \mathcal{P}_2, \mathcal{P}_3\}$, it can be seen that (X, \mathcal{A}) is an MCRD with $\mu_3 = 1$.

Example 10.

$$M_{\{4,2\}} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \end{bmatrix}$$

We have $B(1, 1) = \{1, 2, 3, 4, 5, 6, 7, 8\}$, $B(1, 2) = \{9, 10, 11, 12, 13, 14, 15, 16\}$, $B(2, 1) = \{1, 2, 3, 4, 9, 10, 11, 12\}$,
 $B(2, 2) = \{5, 6, 7, 8, 13, 14, 15, 16\}$, $B(3, 1) = \{1, 2, 5, 6, 9, 10, 13, 14\}$, $B(3, 2) = \{3, 4, 7, 8, 11, 12, 15, 16\}$,
 $B(4, 1) = \{1, 3, 5, 7, 9, 11, 13, 15\}$, $B(4, 2) = \{2, 4, 6, 8, 10, 12, 14, 16\}$. We now show the parallel classes.

$$\mathcal{P}_1 = \{\{1, 2, 3, 4, 5, 6, 7, 8\}, \{9, 10, 11, 12, 13, 14, 15, 16\}\}$$

$$\mathcal{P}_2 = \{\{1, 2, 3, 4, 9, 10, 11, 12\}, \{5, 6, 7, 8, 13, 14, 15, 16\}\}$$

$$\mathcal{P}_3 = \{\{1, 2, 5, 6, 9, 10, 13, 14\}, \{3, 4, 7, 8, 11, 12, 15, 16\}\}$$

$$\mathcal{P}_4 = \{\{1, 3, 5, 7, 9, 11, 13, 15\}, \{2, 4, 6, 8, 10, 12, 14, 16\}\}.$$

For $X = \{1, 2, \dots, 16\}$, $\mathcal{A} = \{\mathcal{P}_1, \mathcal{P}_2, \mathcal{P}_3, \mathcal{P}_4\}$, it can be seen that (X, \mathcal{A}) is an MCRD with $\mu_4 = 1$.

Example 11.

$$M_{3,3} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 2 & 2 & 2 & 2 & 2 & 2 & 2 & 2 \\ 0 & 0 & 0 & 1 & 1 & 1 & 2 & 2 & 2 & 0 & 0 & 0 & 1 & 1 & 1 & 2 & 2 & 2 & 0 & 0 & 0 & 1 & 1 & 1 & 2 & 2 & 2 \\ 0 & 1 & 2 & 0 & 1 & 2 & 0 & 1 & 2 & 0 & 1 & 2 & 0 & 1 & 2 & 0 & 1 & 2 & 0 & 1 & 2 & 0 & 1 & 2 & 0 & 1 & 2 \end{bmatrix}$$

We have $B(1, 1) = \{1, 2, 3, 4, 5, 6, 7, 8, 9\}$, $B(1, 2) = \{10, 11, 12, 13, 14, 15, 16, 17, 18\}$, $B(1, 3) = \{19, 20, 21, 22, 23, 24, 25, 26, 27\}$, $B(2, 1) = \{1, 2, 3, 10, 11, 12, 19, 20, 21\}$, $B(2, 2) = \{4, 5, 6, 13, 14, 15, 22, 23, 24\}$, $B(2, 3) = \{7, 8, 9, 16, 17, 18, 25, 26, 27\}$, $B(3, 1) = \{1, 4, 7, 10, 13, 16, 19, 22, 25\}$, $B(3, 2) = \{2, 5, 8, 11, 14, 17, 20, 23, 26\}$,
 $B(3, 3) = \{3, 6, 9, 12, 15, 18, 21, 24, 27\}$. We now show the parallel classes.

$$\mathcal{P}_1 = \{\{1, 2, 3, 4, 5, 6, 7, 8, 9\}, \{10, 11, 12, 13, 14, 15, 16, 17, 18\}, \{19, 20, 21, 22, 23, 24, 25, 26, 27\}\}$$

$$\mathcal{P}_2 = \{\{1, 2, 3, 10, 11, 12, 19, 20, 21\}, \{4, 5, 6, 13, 14, 15, 22, 23, 24\}, \{7, 8, 9, 16, 17, 18, 25, 26, 27\}\}$$

$$\mathcal{P}_3 = \{\{1, 4, 7, 10, 13, 16, 19, 22, 25\}, \{2, 5, 8, 11, 14, 17, 20, 23, 26\}, \{3, 6, 9, 12, 15, 18, 21, 24, 27\}\}.$$

For $X = \{1, 2, \dots, 27\}$, $\mathcal{A} = \{\mathcal{P}_1, \mathcal{P}_2, \mathcal{P}_3\}$, it can be seen that (X, \mathcal{A}) is an MCRD with $\mu_3 = 1$.

Example 12.

$$M_{\{2,4\}} = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 2 & 2 & 2 & 2 & 3 & 3 & 3 & 3 \\ 0 & 1 & 2 & 3 & 0 & 1 & 2 & 3 & 0 & 1 & 2 & 3 & 0 & 1 & 2 & 3 \end{bmatrix}$$

We have $B(1, 1) = \{1, 2, 3, 4\}$, $B(1, 2) = \{5, 6, 7, 8\}$, $B(1, 3) = \{9, 10, 11, 12\}$, $B(1, 4) = \{13, 14, 15, 16\}$, $B(2, 1) = \{1, 5, 9, 13\}$, $B(2, 2) = \{2, 6, 10, 14\}$, $B(2, 3) = \{3, 7, 11, 15\}$, $B(2, 4) = \{4, 8, 12, 16\}$. We now show the parallel classes.

$$\mathcal{P}_1 = \{\{1, 2, 3, 4\}, \{5, 6, 7, 8\}, \{9, 10, 11, 12\}, \{13, 14, 15, 16\}\}$$

$$\mathcal{P}_2 = \{\{1, 5, 9, 13\}, \{2, 6, 10, 14\}, \{3, 7, 11, 15\}, \{4, 8, 12, 16\}\}.$$

For $X = \{1, 2, \dots, 16\}$, $\mathcal{A} = \{\mathcal{P}_1, \mathcal{P}_2\}$, it can be seen that (X, \mathcal{A}) is an MCRD with $\mu_2 = 1$. We considered the same design in Example 6.

We now construct an MCRD which has $\mu_m = n$, where $n \geq 1$. Construct a matrix $M_{\{m,q\}}^n$ where each vector belonging to \mathbb{Z}_q^m repeats in exactly n columns (in any arbitrary order). Hence $M_{\{m,q\}}^n$ is an $m \times nq^m$ matrix. For $1 \leq i \leq m$, $l \in \mathbb{Z}_q$ we re-define $B(i, l+1)$.

$$B(i, l+1) = \{j \mid j \in \{1, 2, \dots, nq^m\}, M_{m,q}^n(i, j) = l\}$$

The following lemma can be proven proceeding similarly to Lemma 7.

Lemma 10. For any $1 \leq i \leq m$, and $l \in \mathbb{Z}_q$, set $B(i, l+1)$ contains nq^{m-1} elements, i.e. $|B(i, l+1)| = nq^{m-1}$.

From $M_{\{m,q\}}^n$, we construct a design (X, \mathcal{A}) where $X = \{1, 2, \dots, nq^m\}$, and $\mathcal{A} = \{B(i, l+1) \mid 1 \leq i \leq m, l \in \mathbb{Z}_q\}$. Similar to Lemmas 8 and 9 it can be shown that (X, \mathcal{A}) is an MCRD with $\mu_m = n$.

Example 13.

$$M_{\{3,2\}}^2 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \end{bmatrix}$$

$M_{\{3,2\}}^2$ is the same as the first three rows of $M_{4,2}$ considered in Example 10. For the same blocks and parallel classes (\mathcal{P}_4 does not exist in this case), (X, \mathcal{A}) is an MCRD with $\mu_3 = 2$, where $X = \{1, 2, \dots, 16\}$, $\mathcal{A} = \{\mathcal{P}_1, \mathcal{P}_2, \mathcal{P}_3\}$.

APPENDIX C

PROOFS OF LEMMAS 2, 3, 4, 5, 6

A. Proof of Lemma 2

Proof: Since $\frac{t}{b} < \frac{1}{b} \lfloor \frac{b}{z} \rfloor$ our scheme achieves a rate $b - tz$. Then, the values $\frac{M}{N}$ where our scheme achieves a lesser rate that the RK scheme must satisfy

$$b - tz \leq \frac{(K - mtz)^2}{K} \implies b - tz \leq \frac{(mb - mtz)^2}{mb} \quad (6)$$

$$\implies b - tz \leq \frac{m(b - tz)^2}{b} \implies 1 \leq \frac{m(b - tz)}{b} \quad (7)$$

$$\implies b \leq mb - mtz = K - mtz \implies t \leq \frac{K - b}{mz} \implies \frac{t}{b} \leq \frac{K - b}{Kz}.$$

In equation (6) t' has been replaced by mt as $t/b = t'/K$ (see Table VI-B). Since $\frac{M}{N} = \frac{t}{b}$, we have $\frac{t}{b} < \frac{1}{b} \lfloor \frac{b}{z} \rfloor$, which implies $\frac{t}{b} < \frac{1}{z}$, or $b - tz \neq 0$, and hence equation (7) holds. ■

B. Proof of Lemma 3

Proof: The subpacketization level of the RK scheme and the SICPS scheme are equal. At $\frac{M}{N} = \frac{1}{b} = \frac{t'}{K}$ the SICPS scheme achieves a subpacketization level $\frac{K}{t'} \binom{K - t'z + t' - 1}{t' - 1} = b \binom{mb - mz + m - 1}{m - 1}$.

Let n and r be positive integers where $n \geq r$.

$$\begin{aligned} \binom{n}{r} &= \frac{n(n-1)(n-2)\cdots(n-(r-1))}{r(r-1)(r-2)\cdots(r-(r-1))} = \frac{n}{r} \frac{(n-1)}{(r-1)} \frac{(n-2)}{(r-2)} \cdots \frac{(n-(r-1))}{(r-(r-1))} \\ &\geq \frac{n}{r} \frac{n}{r} \frac{n}{r} \cdots \frac{n}{r} = \left(\frac{n}{r}\right)^r. \end{aligned} \quad (8)$$

Since $b \geq z$, we have $mb - mz + m - 1 \geq m - 1$. Then, using equation (8) we have

$$b \binom{mb - mz + m - 1}{m - 1} = b \left(\frac{mb - mz + m - 1}{m - 1}\right)^{m-1}.$$

Our scheme is guaranteed to achieve a lesser subpacketization than the SICPS scheme when

$$\begin{aligned} b^m &\leq b \left(\frac{mb - mz + m - 1}{m - 1}\right)^{m-1} \implies b^{m-1} \leq \left(\frac{mb - mz + m - 1}{m - 1}\right)^{m-1} \\ \implies b &\leq \frac{mb - mz + m - 1}{m - 1} \implies b(m - 1) \leq mb - mz + m - 1 \implies mb - b \leq mb - mz + m - 1 \\ \implies b &\geq mz - m + 1 = m(z - 1) + 1 \implies b \geq \frac{K}{b}(z - 1) + 1. \end{aligned} \quad (9)$$

Now,

$$\begin{aligned} \sqrt{K(z-1)} + 1 &\geq \sqrt{K(z-1)} \implies 1 \geq \frac{\sqrt{K(z-1)}}{\sqrt{K(z-1)} + 1} \\ \implies \sqrt{K(z-1)} &\geq \frac{K(z-1)}{\sqrt{K(z-1)} + 1} \implies \sqrt{K(z-1)} + 1 \geq \frac{K(z-1)}{\sqrt{K(z-1)} + 1} + 1. \end{aligned}$$

So $b = \sqrt{K(z-1)} + 1$ satisfies equation (9). As the right hand side of equation (9) decreases with increasing b , all values of b greater than or equal to $\sqrt{K(z-1)} + 1$ (where b must divide K as well) must satisfy equation (9).

Now, we know that $\binom{n}{r} \geq \binom{n-1}{r-1}$. So the subpacketization level of the NT scheme is always at least as much as the SCIPS scheme. This completes the proof of the lemma. ■

C. Proof of Lemma 4

Proof: It can be seen that for our scheme the corner points $(\frac{1}{b_1}, b_1 - z)$ and $(\frac{1}{b_2}, b_2 - z)$ are achievable. So any point (x, y) on the straight line connecting the two points is also achievable. We have

$$y = -b_1 b_2 x + b_2 + b_1 - z. \quad (10)$$

Let (x, y) be the point such that

$$x = \frac{1}{b_1} + \lambda \left(\frac{1}{b_2} - \frac{1}{b_1} \right) = \frac{m_1 + \lambda(m_2 - m_1)}{K} = \frac{t''}{K}. \quad (11)$$

Since m_1 and m_2 are such that $\gcd(t'', K) = 1$, at $\frac{M}{N} = x$ the SR1 scheme has a non-trivial achievable corner point.

It can be seen from [10] that if $K - t''z$ is even, then the SR1 scheme achieves a rate

$$R_{SR1} = \sum_{r=\frac{K-t''z}{2}}^{K-t''z} \frac{2}{1 + \lceil \frac{t''z}{r} \rceil}.$$

And if $K - t''z > 1$ is odd (the case $K - t''z = 1$ never arise due to the conditions set in the statement of the lemma), then the SR1 scheme achieves a rate

$$R_{SR1} = \frac{1}{1 + \lceil \frac{2t''z}{K-t''z+1} \rceil} + \sum_{r=\frac{K-t''z+3}{2}}^{K-t''z} \frac{2}{1 + \lceil \frac{t''z}{r} \rceil}.$$

So irrespective of whether $K - t''z > 1$ is odd or even, we have

$$R_{SR1} \geq \sum_{r=\frac{K-t''z}{2}}^{K-t''z} \frac{2}{1 + \lceil \frac{t''z}{r} \rceil} \geq \sum_{r=\frac{K-t''z}{2}}^{K-t''z} \frac{2}{1 + \frac{t''z}{r} + 1} = (K - t''z) \frac{2}{2 + \frac{t''z}{\frac{K-t''z}{2}}} = \frac{(K - t''z)(K - t''z + 2)}{K + 2}. \quad (12)$$

Equation (12) has also been shown to hold in reference [9]. So for our scheme achieves a lesser rate at least when

$$\begin{aligned} y &\leq \frac{(K - t''z)(K - t''z + 2)}{K + 2} \\ \implies -b_1 b_2 x + b_2 + b_1 - z &\leq \frac{(K - t''z)(K - t''z + 2)}{K + 2} \end{aligned} \quad (13)$$

$$\implies -b_2 - \lambda(b_1 - b_2) + b_2 + b_1 - z = b_1 - \lambda(b_1 - b_2) - z \leq \frac{(K - t''z)(K - t''z + 2)}{K + 2}. \quad (14)$$

Equations (13) and (14) hold due to equations (10) and (11) respectively. ■

D. Proof of Lemma 5

Proof: The rate achieved by the SR2 scheme is

$$\frac{(mb - mtz)(mb - mtz + mt)}{2mb} = \frac{(b - tz)(mb - mtz + mt)}{2b}.$$

For $\frac{M}{N} \leq \frac{1}{b} \lfloor \lfloor \frac{b}{z} \rfloor \rfloor$ our scheme achieves a lesser rate if

$$\begin{aligned} b - tz &\leq \frac{(b - tz)(mb - mtz + mt)}{2b} \\ \implies 2b &\leq mb - mt(z - 1) \implies t \leq \frac{mb - 2b}{m(z - 1)} \\ \implies \frac{t}{b} &\leq \frac{m - 2}{m(z - 1)}. \end{aligned}$$

■

E. Proof of Lemma 6

For $z < \frac{K}{2}$, the MR scheme achieves a rate $\frac{1}{K} \lceil \frac{K(K-z)}{2} \rceil$ files at $\frac{M}{N} = \frac{1}{K}$. Trivially the point $(\frac{1}{K} \lceil \frac{K}{z} \rceil, 0)$ is also achievable. It can be seen that the line connecting these two achievable points is above the line connecting the points $(\frac{1}{K}, \frac{K-z}{2})$ and $(\frac{1}{z}, 0)$. If the point (x, y) is on the line connecting the latter two points then $2y = -Kzx + K$. So at $\frac{M}{N} = \frac{1}{b}$ for $b < K$ our scheme achieves a lesser rate if

$$2(b-z) \leq -Kz\frac{1}{b} + K \implies 2b - 2z \leq -mbz\frac{1}{b} + mb \implies z(m-2) \leq b(m-2) \implies z \leq b \quad (15)$$

Equation (15) holds when $m \neq 2$. Since $z \leq b$ always holds for our scheme, it means that at $\frac{M}{N} = \frac{1}{b}$, for $b < K$ and $m \geq 3$, our scheme achieves a lesser rate.

APPENDIX D PROOF OF THEOREM 2

We use a lot of notations in the proof. To enhance the readability of the paper, we have listed the most relevant notations in Table D.

Without loss of generality (w.l.o.g.) we assume that for $1 \leq l \leq z-1$, $C_{(i,l)} = \{c(i, y+1), c(i, y+2), \dots, c(i, y + \lfloor \frac{b}{z} \rfloor)\}$ where $y = (l-1)\lfloor \frac{b}{z} \rfloor$, and $C_{(i,z)} = \{c(i, y+1), c(i, y+2), \dots, c(i, b)\}$ where $y = (z-1)\lfloor \frac{b}{z} \rfloor$. The reasoning behind this w.l.o.g. statement is that we can always re-label the caches to achieve the above segregation.

Lemma 11. For $1 \leq i \leq m, 1 \leq j \leq b$, cache $c(i, j) \in C_{(i,n)}$ for $n = \min\{\lceil \frac{j}{x} \rceil, z\}$ where $x = \lfloor \frac{b}{z} \rfloor$.

Proof:

Case 1. $j > (z-1)\lfloor \frac{b}{z} \rfloor$.

In this case we know that $c(i, j) \in C_{(i,z)}$. As $\frac{j}{x} > z-1$, we have $n = z$, and hence the lemma hold for this case.

Case 2. $1 \leq j \leq (z-1)\lfloor \frac{b}{z} \rfloor$.

If $c(i, j) \in C_{(i,l)}$ for $1 \leq l \leq z-1$ then $(l-1)\lfloor \frac{b}{z} \rfloor < j \leq l\lfloor \frac{b}{z} \rfloor$. This implies $(l-1) < \frac{j}{x} \leq l$ (as $1 \leq z \leq b$). Hence, $\lceil j/x \rceil = l$. ■

In $G_i = (K_i, C_i, E_i)$, a vertex $u \in K_i$ can choose a vertex in $C_{(i,l)}$ in $\lfloor \frac{b}{z} \rfloor$ ways if $1 \leq l \leq z-1$ and in $b - (z-1)\lfloor \frac{b}{z} \rfloor$ ways if $l = z$. So total number of possible user-to-cache associations that satisfy conditions **C1** and **C2** is $(\lfloor \frac{b}{z} \rfloor^{z-1} (b - (z-1)\lfloor \frac{b}{z} \rfloor))^{bm}$.

For each user $k(i, j)$ where $1 \leq i \leq m, 1 \leq j \leq b$, we use the notation $C_{k(i,j)}$ to denote the set of caches user $k(i, j)$ accesses. Equivalently, $C_{k(i,j)}$ is the set of all vertices in C_i such that there exists an edge in E_i that connects the user $k(i, j)$ to the vertex. Note, condition **C3** ensures that $f_{M_i}(k(i, j)) \in C_{k(i,j)}$ for $1 \leq i \leq m, 1 \leq j \leq b$.

In Section VII we show the placement and delivery of the MACC problem whose user-to-cache association is given by the bipartite graph shown in Fig. 1.

TABLE III
NOTATIONS AND THEIR SIGNIFICANCE

| Notation | Range | Significance |
|-----------------------|------------------------------------|---|
| m | \mathbb{N} | mb caches, users, partitioned into m disjoint sets of b caches, users respectively; MCRD has m parallel classes, subpacketization level is b^m . |
| b | \mathbb{N} | mb caches, users, partitioned into m disjoint sets of b caches, users respectively; each parallel class of MCRD has b blocks, subpacketization level is b^m . |
| z | $1 \leq z \leq b$ | Each user accesses z caches. |
| W^i | $1 \leq i \leq N$ | The N files at the server. Each file W^i is split into b^m subfiles: $W^i(1), W^i(2), \dots, W^i(b^m)$. |
| K_i | $1 \leq i \leq m$ | mb users partitioned into m disjoint sets K_1, K_2, \dots, K_m each having b users. |
| $k(i, j)$ | $1 \leq i \leq m, 1 \leq j \leq b$ | j^{th} user in K_i for some ordering of the users in K_i . |
| C_i | $1 \leq i \leq m$ | mb caches partitioned into m disjoint sets C_1, C_2, \dots, C_m each having b caches. |
| $c(i, j)$ | $1 \leq i \leq m, 1 \leq j \leq b$ | j^{th} cache in C_i for some ordering of the caches in C_i . |
| $C_{(i,l)}$ | $1 \leq i \leq m, 1 \leq l \leq z$ | Set C_i is partitioned into z disjoint sets $C_{(i,1)}, C_{(i,2)}, \dots, C_{(i,z)}$. Set $C_{(i,l)}$ for $1 \leq l \leq z-1$ contains $\lfloor b/z \rfloor$ caches. Set $C_{(i,z)}$ contains $b - (z-1)\lfloor b/z \rfloor$ caches. |
| $C_{k(i,j)}$ | $1 \leq i \leq m, 1 \leq j \leq z$ | $C_{k(i,j)}$ is the set of caches user $k(i, j)$ accesses. $C_{k(i,j)} \subseteq C_i$. For $1 \leq l \leq z$, $ C_{(i,l)} \cap C_{k(i,j)} = 1$. |
| f_{M_i} | $1 \leq i \leq m$ | $f_{M_i} : \{1, 2, \dots, b\} \rightarrow \{1, 2, \dots, b\}$. The set $\{f_{M_i}(k(i, 1)), f_{M_i}(k(i, 2)), \dots, f_{M_i}(k(i, b))\}$ is a system of distinct representatives of the sets $C_{k(i,1)}, C_{k(i,2)}, \dots, C_{k(i,b)}$ where $f_{M_i}(k(i, j)) \in C_{k(i,j)}$ for $1 \leq j \leq b$ and $f_{M_i}(k(i, j_1)) \neq f_{M_i}(k(i, j_2))$. |
| \mathcal{P}_i | $1 \leq i \leq m$ | We consider MCRDs with m parallel classes denoted by $\mathcal{P}_1, \mathcal{P}_2, \dots, \mathcal{P}_m$. |
| $B(i, j)$ | $1 \leq i \leq m, 1 \leq j \leq b$ | Each parallel class \mathcal{P}_i has b blocks $B(i, 1), B(i, 2), \dots, B(i, b)$. |
| $\mathcal{P}_{(i,l)}$ | $1 \leq i \leq m, 1 \leq l \leq z$ | Set \mathcal{P}_i is partitioned into z disjoint sets $\mathcal{P}_{(i,1)}, \mathcal{P}_{(i,2)}, \dots, \mathcal{P}_{(i,z)}$. Set $\mathcal{P}_{(i,l)}$ for $1 \leq l \leq z-1$ contains $\lfloor b/z \rfloor$ blocks. Set $\mathcal{P}_{(i,z)}$ contains $b - (z-1)\lfloor b/z \rfloor$ blocks. |
| t, t', t_z | \mathbb{N} | If cache $c \in C_{(i,l)}$ where $1 \leq l \leq z-1$, then c stores t' blocks belonging to $\mathcal{P}_{(i,l)}$. If cache $c \in C_{(i,z)}$, then c stores t_z blocks belonging to $\mathcal{P}_{(i,z)}$. Value of t is provided with the coded caching problem. If $1 \leq t \leq \lfloor b/z \rfloor$, $t' = t_z = t$. If $\lfloor b/z \rfloor < t < b - (z-1)\lfloor \frac{b}{z} \rfloor$, $t' = \lfloor b/z \rfloor, t_z = t$. If $t \geq b - (z-1)\lfloor \frac{b}{z} \rfloor$, $t' = \lfloor b/z \rfloor, t_z = b - (z-1)\lfloor \frac{b}{z} \rfloor$. |
| $B_{c(i,j)}$ | $1 \leq i \leq m, 1 \leq j \leq b$ | $B_{c(i,j)} \subseteq \mathcal{P}_i$. During placement, cache $c(i, j)$ stores the subfiles indexed by the blocks contained in $B_{c(i,j)}$. If $c(i, j) \in C_{(i,l)}$ where $1 \leq l \leq z-1$, $B_{c(i,j)}$ contains $B(i, j)$ and another $t' - 1$ blocks from $\mathcal{P}_{(i,l)}$. If $c(i, j) \in C_{(i,z)}$, $B_{c(i,j)}$ contains $B(i, j)$ another $t_z - 1$ blocks from $\mathcal{P}_{(i,z)}$. |
| $B_{k(i,j)}$ | $1 \leq i \leq m, 1 \leq j \leq b$ | Set $B_{k(i,j)}$ denotes the set of blocks whose contents are the indices of the subfiles stored in the caches accessed by user $k(i, j)$. Since $k(i, j)$ accesses the caches in $C_{k(i,j)}$, and each cache $c(x, y)$ stores the subfiles indexed by the elements of the blocks contained in $B_{c(x,y)}$, we have $B_{k(i,j)} = \cup_{c(x,y) \in C_{k(i,j)}} B_{c(x,y)}$. |
| $f_{(i,j)}$ | $1 \leq i \leq m, 1 \leq j \leq b$ | $f_{(i,j)} : \{1, 2, \dots, b - t'(z-1) - t_z\} \rightarrow \{1, 2, \dots, b\} \setminus \{l \mid B(i, l) \in B_{k(i,j)}\}$. $f_{i,j}(\{1, 2, \dots, b - t'(z-1) - t_z\})$ is the set of all integers l such that $B(i, l) \notin B_{k(i,j)}$. |

A. Placement

Let (X, \mathcal{A}) be a maximal cross resolvable design (MCRD) with m parallel classes, b blocks in each parallel class, $X = \{1, 2, \dots, b^m\}$, $|\mathcal{A}| = mb$, and $\mu_m = 1$. The existence of such an MCRD is guaranteed by Theorem 1. As per Lemma 1, each block in \mathcal{A} has exactly b^{m-1} elements. Let $\mathcal{P}_1, \mathcal{P}_2, \dots, \mathcal{P}_m$ be the m parallel classes of (X, \mathcal{A}) . Let $B(i, 1), B(i, 2), \dots, B(i, b)$ be the b blocks contained in the parallel class \mathcal{P}_i .

For $1 \leq i \leq m$, $1 \leq j \leq b$, for each cache $c(i, j)$, we define a set $B_{c(i,j)} \subseteq \mathcal{P}_i$; cache $c(i, j)$ stores the subfiles

$W^n(l)$ for $\forall n \in \{1, 2, \dots, N\}$ and $l \in B(i, s)$ for $\forall B(i, s) \in B_{c(i,j)}$ (i.e., the cache $c(i, j)$ stores all files indexed by the elements of the blocks contained in $B_{c(i,j)}$). Towards describing $B_{c(i,j)}$, we segregate each parallel class in the following way.

We partition the set of blocks in \mathcal{P}_i for $1 \leq i \leq m$ into z disjoint subsets $\mathcal{P}_{(i,1)}, \mathcal{P}_{(i,2)}, \dots, \mathcal{P}_{(i,z)}$. For $1 \leq l \leq z - 1$, the subset $\mathcal{P}_{(i,l)}$ contains $\lfloor \frac{b}{z} \rfloor$ blocks. Whereas, the subset $\mathcal{P}_{(i,z)}$ contains $b - (z - 1)\lfloor \frac{b}{z} \rfloor$ blocks, i.e., the blocks contained in $\mathcal{P}_{(i,z)}$ are the blocks that are in \mathcal{P}_i , but not in the sets $\mathcal{P}_{(i,l)}$ for $1 \leq l \leq z - 1$. Without loss of generality (w.l.o.g.) we assume that, for $1 \leq l \leq z - 1$, $\mathcal{P}_{(i,l)} = \{B(i, y + 1), B(i, y + 2), \dots, B(i, y + \lfloor \frac{b}{z} \rfloor)\}$ where $y = (l - 1)\lfloor \frac{b}{z} \rfloor$, and $\mathcal{P}_{(i,z)} = \{B(i, y + 1), B(i, y + 2), \dots, B(i, b)\}$ where $y = (z - 1)\lfloor \frac{b}{z} \rfloor$. Similar to Lemma 11 we have the following.

Lemma 12. For $1 \leq i \leq m, 1 \leq j \leq b$, cache $B(i, j) \in \mathcal{P}_{(i,n)}$ if and only if $n = \min\{\lceil \frac{j}{x} \rceil, z\}$ where $x = \lfloor \frac{b}{z} \rfloor$.

We define two variables t', t_z . If $1 \leq t \leq \lfloor b/z \rfloor$, $t' = t_z = t$. If $\lfloor b/z \rfloor < t < b - (z - 1)\lfloor \frac{b}{z} \rfloor$, $t' = \lfloor b/z \rfloor, t_z = t$. If $t \geq b - (z - 1)\lfloor \frac{b}{z} \rfloor$, $t' = \lfloor b/z \rfloor, t_z = b - (z - 1)\lfloor \frac{b}{z} \rfloor$.

Due to Lemma 11 and Lemma 12, for $1 \leq i \leq m, 1 \leq j \leq b$, if $c(i, j) \in C_{(i,l)}$, then $B(i, j) \in \mathcal{P}_{(i,l)}$.

We now define $B_{c(i,j)}$. For $1 \leq i \leq m, 1 \leq j \leq b$, if $c(i, j) \in C_{(i,l)}$ where $1 \leq l \leq z - 1$, $B_{c(i,j)}$ contains $B(i, j)$ in addition to another $t' - 1$ arbitrarily chosen blocks from $\mathcal{P}_{(i,l)}$; if $c(i, j) \in C_{(i,z)}$, $B_{c(i,j)}$ contains $B(i, j)$ in addition to another $t_z - 1$ arbitrarily chosen blocks from $\mathcal{P}_{(i,z)}$.

$$B_{c(i,j)} = \begin{cases} B(i, j) \cup \{B(i, j_1), B(i, j_2), \dots, B(i, j_{t'-1})\} \subseteq \mathcal{P}_{(i,n)} \text{ where } n = \lceil j/\lfloor \frac{b}{z} \rfloor \rceil, j \notin \{j_1, j_2, \dots, j_{t'-1}\} \\ \text{if } \lceil j/\lfloor \frac{b}{z} \rfloor \rceil < z. \\ B(i, j) \cup \{B(i, j_1), B(i, j_2), \dots, B(i, j_{t_z-1})\} \subseteq \mathcal{P}_{(i,z)}, j \notin \{j_1, j_2, \dots, j_{t_z-1}\} \text{ if } \lceil j/\lfloor \frac{b}{z} \rfloor \rceil \geq z. \end{cases} \quad (16)$$

Because each block contains b^{m-1} subfiles, if $c(i, j) \in C_{(i,l)}$ where $1 \leq l \leq z - 1$ then $c(i, j)$ stores $b^{m-1}Nt'$ subfiles, and if $c(i, j) \in C_{(i,z)}$ then $c(i, j)$ stores $b^{m-1}Nt_z$ subfiles. Since $t', t_z \leq t$ and that each cache can store upto $M = \frac{tN}{b}$ files, which is equivalent to Mb^m subfiles or $b^{m-1}Nt$ subfiles, the placed content does not exceed available cache memory size.

We have the following lemma as a result of $B_{c(i,j)}$.

Lemma 13. For any $1 \leq i \leq m, 1 \leq j_1, j_2 \leq b, 1 \leq l_1, l_2 \leq z, l_1 \neq l_2$, if $c(i, j_1) \in C_{(i,l_1)}$ and $c(i, j_2) \in C_{(i,l_2)}$, then $B_{c(i,j_1)} \cap B_{c(i,j_2)} = \emptyset$.

Proof: Since $l_1 \neq l_2$ we have $\mathcal{P}_{(i,l_1)} \neq \mathcal{P}_{(i,l_2)}$, and from Lemma 11 we have $j_1 \neq j_2$. Sets $\mathcal{P}_{(i,l_1)}$ and $\mathcal{P}_{(i,l_2)}$ are disjoint by definition. So from equation (16) we have $B_{c(i,j_1)} \cap B_{c(i,j_2)} = \emptyset$. ■

Note that our placement is partially flexible in nature (when $1 \leq t < \lfloor b/z \rfloor$) in terms of which blocks are contained in $B_{c(i,j)}$.

B. Delivery

In this subsection, we show both the delivery scheme and the correctness of the delivery scheme.

Lemma 14. User $k(i, j)$ where $1 \leq i \leq m$, $1 \leq j \leq b$ receives $(t'(z-1) + t_z)b^{m-1}$ subfiles of each file W^1, W^2, \dots, W^N from the z caches contained in $C_{k(i, j)}$.

Proof: Let $C_{k(i, j)} = \{c(i, j_1), c(i, j_2), \dots, c(i, j_z)\}$. Due to condition **C2**, if $c(i, j_x) \in C_{(i, l_x)}$ and $c(i, j_y) \in C_{(i, l_y)}$ for $x, y \in \{1, 2, \dots, z\}$, $x \neq y$, then $C_{(i, l_x)} \neq C_{(i, l_y)}$. Hence, as shown in Lemma 13, $B_{c(i, j_x)} \cap B_{c(i, j_y)} = \emptyset$. Hence, user $k(i, j)$ receives $(t'(z-1))b^{m-1}$ subfiles of each file from $\{c(i, j_1), c(i, j_2), \dots, c(i, j_{z-1})\}$ and an additional (non-overlapping) $t_z b^{m-1}$ subfiles of each file from $c(i, j_z)$ where $c(i, j_z) \in C_{(i, z)}$. ■

Lemma 15. When $t \geq b - (z-1)\lfloor \frac{b}{z} \rfloor$, we have $R = 0$ files.

Proof: In this case we have $t' = \lfloor b/z \rfloor$ and $t_z = b - (z-1)\lfloor \frac{b}{z} \rfloor$. As per Lemma 14, user $k(i, j)$ for $1 \leq i \leq m$, $1 \leq j \leq b$ receives $(t'(z-1) + t_z)b^{m-1} = b^m$ subfiles of each file from the caches. As subpacketization level is b^m , every user receives all subfiles of all files at the server. ■

In the remaining part of this subsection, we show the delivery when $t < b - (z-1)\lfloor \frac{b}{z} \rfloor$. For each user $k(i, j)$ where $1 \leq i \leq m$, $1 \leq j \leq b$, we define a set $B_{k(i, j)} = \cup_{c(x, y) \in C_{k(i, j)}} B_{c(x, y)}$ (it is the set of blocks whose elements are the indices of the subfiles stored by the caches accessed by user $k(i, j)$).

For $1 \leq i \leq m$, $1 \leq j \leq b$, each user $k(i, j)$ needs to retrieve the subfiles indexed by the contents of the blocks in $\mathcal{P}_i \setminus B_{k(i, j)}$. For this purpose the users use the broadcast from the central server along with the contents of the accessible caches. The central server broadcasts following the below procedure.

We now construct a new graph that we call the demand graph. For the user-to-cache association bipartite graph $G = (U, C, E)$ and the MCRD (X, \mathcal{A}) based placement scheme (that is, $B_{c(i, j)}$ is defined for all $c(i, j) \in C$), the demand graph $\bar{G} = (V_1, V_2, \bar{E})$ is a bipartite graph where $|V_1| = |V_2| = mb$, and for each cache in C there is a unique vertex in V_1 , for each block in \mathcal{A} there is a unique vertex in V_2 . In other words, there is a bijection from the vertices in V_1 to the caches in C , and there is another bijection from the vertices in V_2 to the blocks in \mathcal{A} . Say for $l = 1, 2$, $V_l = \{v_{(l, i, j)} | 1 \leq i \leq m, 1 \leq j \leq b\}$, and that the vertex $v_{(1, i, j)}$ maps to the cache $c(i, j)$, similarly, say that the vertex $v_{(2, i, j)}$ maps to the block $B(i, j)$. There exists an edge in \bar{E} connecting vertex $v_{(1, i, j)} \in V_1$ to vertex $v_{(2, \bar{i}, j^*)} \in V_2$ if and only if $i = \bar{i}$ and $B(i, j^*) \notin B_{k(i, j')}$ where $k(i, j') = f_{M_i}^{-1}(c(i, j))$. The existence of f_{M_i} and its inverse is ensured by condition **C3**.

We now show that conditions **C1** and **C2** enforces $(b - t'(z-1) - t_z)b^m$ partial matchings (each containing m edges) of a certain kind on \bar{G} . For $1 \leq i \leq m$, $l = 1, 2$ we define the set $V_{(l, i)} = \{v_{(l, i, j)} | 1 \leq j \leq b\}$. Each vertex $v_{(1, i, j)} \in V_{(1, i)}$ is adjacent to $(b - t'(z-1) - t_z)$ edges as there are $(b - t'(z-1) - t_z)$ blocks in \mathcal{P}_i that are not contained in $B_{f_{M_i}^{-1}(c(i, j))}$.

To construct a partial matching of m edges, pick m vertices from V_1 by selecting only one vertex from each $V_{(1, i)}$ for $1 \leq i \leq m$. There are b ways to choose a vertex from $V_{(1, i)}$, and hence b^m ways to choose these m vertices from V_1 . Consider one such chosen set to m vertices. For each chosen vertex $v_{(1, i, j_i)} \in V_{(1, i)}$, pick one vertex $v \in V_{(2, i)}$ such that there is an edge connecting $v_{(1, i, j_i)}$ and v . It can be seen that v can be chosen in $(b - t'(z-1) - t_z)$ ways.

For $1 \leq i \leq m$, $1 \leq j \leq b$, for each user $k(i, j)$ we define a bijection

$$f_{(i,j)} : \{1, 2, \dots, b - (t'(z-1) + t_z)\} \rightarrow \{1, 2, \dots, b\} \setminus \{l \mid B(i, l) \in B_{k(i,j)}\}.$$

The bijection $f_{(i,j)}$ can be defined arbitrarily without violating the domain, range, and bijectivity. For $1 \leq n \leq b - (t'(z-1) + t_z)$ define a set of partial matchings \mathcal{M}_n as following

$$\mathcal{M}_n = \{M_{j_1, j_2, \dots, j_m}^n \mid 1 \leq j_i \leq b, 1 \leq i \leq m\}, \text{ where}$$

$$M_{j_1, j_2, \dots, j_m}^n = \{(v_{(1, i, j_i)}, v_{(2, i, f_{(i, j_i)}(n))}) \mid 1 \leq i \leq m\} \text{ where } k(i, j_i) = f_{M_i}^{-1}(c(i, j_i)).$$

It can be seen that \mathcal{M}_n has b^m matchings. Let $\mathcal{M} = \bigcup_{n=1}^{b-(t'(z-1)+t_z)} \mathcal{M}_n$. Hence, \mathcal{M} has $(b - (t'(z-1) + t_z))b^m$ matchings.

During delivery, for each such partial matching in \mathcal{M} we make one transmission, and each of these transmissions benefit all m users associated with the matching (the association that is established by the inverse of f_{M_i} acting on the m chosen vertices (caches) from V_1). Say for a matching in $M_{j_1, j_2, \dots, j_m}^n \in \mathcal{M}$ the vertex $v_{(1, i, j_i)} \in V_{(1, i)}$ connects to the vertex $v_{(2, i, j_i^*)} \in V_{(2, i)}$ for $1 \leq i \leq m$. The transmission corresponding to this matching is given in the following. For $1 \leq i \leq m$ compute the set $S_{j_1, j_2, \dots, j_m}^{n, i}$ where

$$S_{j_1, j_2, \dots, j_m}^{n, i} = B(1, j_1) \cap B(2, j_2) \cap \dots \cap B(i-1, j_{i-1}) \cap B(i, j_i^*) \cap B(i+1, j_{i+1}) \cap \dots \cap B(m, j_m). \quad (17)$$

Since in (X, \mathcal{A}) we have $\mu_m = 1$, set $S_{j_1, j_2, \dots, j_m}^{n, i}$ for $1 \leq i \leq m$ is a singleton set. We transmit the following $(b - (t'(z-1) + t_z))b^m$ linear function of the subfiles to meet all users' demands.

$$Y_{j_1, j_2, \dots, j_m}^n = \sum_{i=1}^m W_{f_{M_i}^{-1}(c(i, j_i))}^{d_{f_{M_i}^{-1}(c(i, j_i))}}(S_{j_1, j_2, \dots, j_m}^{n, i}) \quad \text{for } 1 \leq n \leq b - (t'(z-1) + t_z), 1 \leq j_1, j_2, \dots, j_m \leq b. \quad (18)$$

Lemma 16. For the set $S_{j_1, j_2, \dots, j_m}^{n, i}$ defined in equation (17), if $e \in S_{j_1, j_2, \dots, j_m}^{n, i}$, then

- (i) e is not an element of any block contained in $B_{k(i, j_i')}$ where $k(i, j_i') = f_{M_i}^{-1}(c(i, j_i))$,
- (ii) e is an element of some block in $B_{k(x, j_x')}$ where $k(x, j_x') = f_{M_x}^{-1}(c(x, j_x))$ for $1 \leq x \leq m$, $x \neq i$,
- (iii) Using the transmission in equation (18) user $k(i, j_i')$ retrieves the subfile $W^{d_{k(i, j_i')}}(S_{j_1, j_2, \dots, j_m}^{n, i})$ where $f_{M_i}(k(i, j_i')) = c(i, j_i)$.

Proof: (i) Due to the construction of $\bar{G} = (V_1, V_2, \bar{E})$ we have $B(i, j_i^*) \notin B_{k(i, j_i')}$.

(ii) From equation (17), we also see that $e \in B(x, j_x)$ for all $1 \leq x \leq m$, $i \neq x$. Due to equation (16), $B(x, j_x) \in B_{c(x, j_x)}$. We know user $k(x, j_x')$ accesses the cache $c(x, j_x)$. So $B(x, j_x) \in B_{k(x, j_x')}$.

(iii) As per the proof of statement (ii), user $k(i, j_i')$ knows any subfile indexed by the element contained in $S_{j_1, j_2, \dots, j_m}^{n, x}$ for $1 \leq x \leq m$, $x \neq i$. So in the sum of equation (18) user $k(i, j_i')$ knows $\sum_{x=1, x \neq i}^m W^{d_{k(x, j_x')}}(S_{j_1, j_2, \dots, j_m}^{n, x})$. Hence user $k(i, j_i')$ can retrieve $W^{d_{k(i, j_i')}}(S_{j_1, j_2, \dots, j_m}^{n, i})$. ■

Lemma 17. After making one transmission (given by equation (18)) for each partial matchings in \mathcal{M} , each user receives all subfiles of the file it demands.

Proof: Say for user $k(i, j_i')$ for $1 \leq i \leq m$, $1 \leq j_i' \leq b$, $f_{M_i}(k(i, j_i')) = c(i, j_i)$. Let $B(i, j_i^*)$ be a block such that $B(i, j_i^*) \notin B_{k(i, j_i')}$. Then there exists an edge e connecting vertices $v_{(1, i, j_i)} \in V_1$ and $v_{(2, i, j_i^*)} \in V_2$. So for

some $1 \leq n \leq b - (t'(z-1) + t_z)$, $f_{(i,j'_i)}(n) = j_i^*$. Let $\mathcal{M}_{(n,e)} \subseteq \mathcal{M}_n$ be the set of all matchings in \mathcal{M}_n such that $M \in \mathcal{M}_{(n,e)}$ if and only if $e \in M$. It can be seen that $|\mathcal{M}_{(n,e)}| = b^{m-1}$. Say the vertices of $\mathcal{M}_{(n,e)}$ that belong to V_1 are $v_{(1,1,j_1)}, v_{(1,2,j_2)}, \dots, v_{(1,m,j_m)}$.

There are b^{m-1} transmissions (given by equation (18)) for all matchings in $\mathcal{M}_{(n,e)}$. Hence, as per statement (iii) of Lemma 16 and equation (17), user $k(i, j'_i)$ can retrieve $W^{d_{k(i,j'_i)}}(S_{j_1, j_2, \dots, j_m}^{n,i})$ for $1 \leq j_l \leq b$, $1 \leq l \leq m$, $l \neq i$.

Corollary 1 showed that the block $B(i, j_i^*) = \cup_{1 \leq j_l \leq b, 1 \leq l \leq m, l \neq i} S_{j_1, j_2, \dots, j_m}^{n,i}$. Hence, after the transmissions corresponding to all matchings in \mathcal{M}_n , user $k(i, j'_i)$ receives all subfiles whose indices belong to $B(i, j_i^*)$.

Since $f_{(i,j'_i)}$ is onto, after the transmissions corresponding to all matchings in \mathcal{M} , user $k(i, j'_i)$ receives all subfiles whose indices belong to any block not contained in $B_{k(i,j'_i)}$. ■

As a result of Lemma 17, and as subpacketization level is b^m , when $t < b - (z-1)\lfloor \frac{b}{z} \rfloor$, a rate of $b - (t'(z-1) + t_z)$ files is achievable.

REFERENCES

- [1] M. A. Maddah-Ali and U. Niesen, "Fundamental Limits of Caching," in *IEEE Transactions on Information Theory*, vol. 60, no. 5, pp. 2856-2867, May 2014, doi: 10.1109/TIT.2014.2306938.
- [2] M. A. Maddah-Ali and U. Niesen, "Decentralized Coded Caching Attains Order-Optimal Memory-Rate Tradeoff," in *IEEE/ACM Transactions on Networking*, vol. 23, no. 4, pp. 1029-1040, Aug. 2015, doi: 10.1109/TNET.2014.2317316.
- [3] K. Shanmugam, M. Ji, A. M. Tulino, J. Llorca and A. G. Dimakis, "Finite-Length Analysis of Caching-Aided Coded Multicasting," in *IEEE Transactions on Information Theory*, vol. 62, no. 10, pp. 5524-5537, Oct. 2016, doi: 10.1109/TIT.2016.2599110.
- [4] Q. Yan, M. Cheng, X. Tang and Q. Chen, "On the Placement Delivery Array Design for Centralized Coded Caching Scheme," in *IEEE Transactions on Information Theory*, vol. 63, no. 9, pp. 5821-5833, Sept. 2017.
- [5] K. Shanmugam, N. Golrezaei, A. G. Dimakis, A. F. Molisch and G. Caire, "FemtoCaching: Wireless Content Delivery Through Distributed Caching Helpers," in *IEEE Transactions on Information Theory*, vol. 59, no. 12, pp. 8402-8413, Dec. 2013, doi: 10.1109/TIT.2013.2281606.
- [6] J. Hachem, N. Karamchandani and S. N. Diggavi, "Coded Caching for Multi-level Popularity and Access," in *IEEE Transactions on Information Theory*, vol. 63, no. 5, pp. 3108-3141, May 2017, doi: 10.1109/TIT.2017.2664817.
- [7] B. Serbetci, E. Parrinello and P. Elia, "Multi-access coded caching: gains beyond cache-redundancy," *Proceedings of IEEE Information Theory Workshop (ITW)*, 2019, doi: 10.1109/ITW44776.2019.8989128.
- [8] K. S. Reddy and N. Karamchandani, "Rate-Memory Trade-off for Multi-Access Coded Caching With Uncoded Placement," in *IEEE Transactions on Communications*, vol. 68, no. 6, pp. 3261-3274, June 2020, doi: 10.1109/TCOMM.2020.2980817.
- [9] M. Cheng, K. Wan, D. Liang, M. Zhang and Giuseppe Caire, "A Novel Transformation Approach of Shared-Link Coded Caching Schemes for Multiaccess Networks," in *IEEE Transactions on Communications*, vol. 69, no. 11, pp. 7376-7389, Nov 2021, doi: 10.1109/TCOMM.2021.3104035.
- [10] S. Sasi and B. S. Rajan, "An Improved Multi-access Coded Caching with Uncoded Placement," Available on *arXiv:2009.05377v3 [cs.IT]*.
- [11] S. Sasi and B. S. Rajan, "Multi-Access Coded Caching Scheme With Linear Sub-Packetization Using PDAs," in *IEEE Transactions on Communications*, vol. 69, no. 12, pp. 7974-7985, Dec 2021, doi: 10.1109/TCOMM.2021.3116229.
- [12] A. A. Mahesh, and B. S. Rajan, "A Coded Caching Scheme with Linear Sub-packetization and its Application to Multi-Access Coded Caching," in *Proceedings of IEEE Information Theory Workshop (ITW)*, 2021, Riva del Garda, Italy.
- [13] K. S. Reddy and N. Karamchandani, "Structured Index Coding Problem and Multi-access Coded Caching," in *IEEE Journal on Selected Areas in Information Theory*, special Issue on Beyond Errors and Erasures: Coding for Data Management and Delivery in Networks, Dec. 2021
- [14] D. Katyal, P. N. Muralidhar and B. S. Rajan, "Multi-Access Coded Caching Schemes From Cross Resolvable Designs," in *IEEE Transactions on Communications*, vol. 69, no. 5, pp. 2997-3010, May 2021, doi: 10.1109/TCOMM.2021.3053048.
- [15] P. N. Muralidhar, D. Katyal and B. S. Rajan, "Improved Multi-access Coded Caching Schemes From Cross Resolvable Designs," in *Proceedings of IEEE Information Theory Workshop (ITW)*, October 2021, Kanazawa, Japan.

- [16] P. N. Muralidhar and B. S. Rajan, "Multi-access Coded Caching from a New Class of Cross Resolvable Designs," in *Proceedings of IEEE International Symposium on Information Theory (ISIT)*, July 2021, Melbourne, Australia.
- [17] P. N. Muralidhar, D. Katyal and B. S. Rajan, "Maddah-Ali-Niesen Scheme for Multi-access Coded Caching," in *Proceedings of IEEE Information Theory Workshop (ITW)*, October 2021, Kanazawa, Japan.
- [18] F. Brunero and P. Elia, "Fundamental Limits of Combinatorial Multi-Access Caching," Available on *arXiv:2110.07426 [cs.IT]*.
- [19] Li Tang and Aditya Ramamoorthy, "Coded Caching Schemes With Reduced Subpacketization From Linear Block Codes," in *IEEE Transactions on Information Theory*, vol. 64, no. 4, pp-3099–3120, 2018.