Coded Caching based on Combinatorial Designs

by

Shailja Agarwal, K V Sushena Sree, Prasad Krishnan

in

2019 IEEE International Symposium on Information Theory

Paris

Report No: IIIT/TR/2019/-1

Centre for Communications
International Institute of Information Technology
Hyderabad - 500 032, INDIA
July 2019

Coded Caching based on Combinatorial Designs

Shailja Agrawal, K V Sushena Sree, Prasad Krishnan
International Institute of Information Technology, Hyderabad
Email: {shailja.agrawal@research. , sushena.sree@research. , prasad.krishnan@}iiit.ac.in

***Abstract*—We consider the standard broadcast setup with a single server broadcasting information to a number of clients, each of which contains local storage (called *cache*) of some size, which can store some parts of the available files at the server. The centralized coded caching framework, consists of a caching phase and a delivery phase, both of which are carefully designed in order to use the cache and the channel together optimally. In prior literature, various combinatorial structures have been used to construct coded caching schemes. In this work, we propose a binary matrix model to construct the coded caching scheme. The ones in such a *caching matrix* indicate uncached subfiles at the users. Identity submatrices of the caching matrix represent transmissions in the delivery phase. Using this model, we then propose several novel constructions for coded caching based on the various types of combinatorial designs. While most of the schemes constructed in this work (based on existing designs) have a high cache requirement (uncached fraction being $\Theta(\frac{1}{\sqrt{K}})$, K being the number of users), they provide a rate R that is upper bounded by a constant ($R \leq 1$) with increasing K, and moreover require extremely small levels of subpacketization (being $O(K)$), which is an extremely important parameter in practical applications of coded caching.**

I. Introduction

Coded caching, which uses coded transmission for broadcast communication where each of the clients have local storage (called *cache*), was proposed in [1] and has emerged as a valuable technique to use the broadcast communication channel efficiently. The coded caching scenario as in [1], consists of clients indexed by some set $\mathcal{K}$ of size K and possessing some cache, connected to a single server through an error free shared link. The library of files at the server consists of N files of same size, which are denoted as $W_i : \forall i \in [N]$. Each file consists of F non-overlapping subfiles of same size, where F is known as the *subpacketization level*. The subfiles of W_i are labelled as $W_{i,f} : f \in \mathcal{F}$, where $\mathcal{F}$ is a set of size F and $W_{i,f}$ takes values in some Abelian group. The centralized coded caching framework consists of two phases : the *placement phase* and the *delivery phase*. The placement phase occurs during non-peak hours. In the placement phase, the communication channel is utilized so that each client stores some $\frac{M}{N}$ fraction of each file in the library in its cache, where M reflects the cache size. The delivery phase corresponds to peak-hours. In the delivery phase, the demands of the users pop up. In the coded caching paradigm, the server broadcasts *coded* transmissions such that the demands of all the users are satisfied. As in [1], the rate R of the coded caching scheme is defined as the ratio of the number of bits transmitted to the size of each file, and can be calculated as

$$\text{Rate } R = \frac{\text{Number of transmissions in the delivery phase}}{\text{Subpacketization level}},$$

when each transmission is of the same size as any subfile.

Though the coded caching framework presented in [1] achieves an optimal rate, its exponential increase in subpacketization with respect to the number of users at constant $\frac{M}{N}$ is a major setback for its practical implementation. The scheme presented in [2] gives reduced subpacketization by using a combinatorial structure that designs both the placement and the delivery phase together, called as the Placement Delivery Array (PDA). In [3], resolvable designs derived from linear block codes have been used to reduce subpacketization. All of these schemes offered reductions in subpacketization as compared to [1], at the cost of some increase in the rate, for constant memory fraction $\frac{M}{N}$. However, to the best of our knowledge, most of the schemes (for reasonable values of K) available in literature require subpacketization exponential in K. A subpacketization subexponential in K has been obtained in [4] using a line graph model for coded caching along with a projective geometry based scheme. For constant rate, the scheme in [4] achieves a subpacketization level of $O(q^{(log_q K)^2})$, however demanding that the uncached fraction, $(1 - \frac{M}{N}) = \Theta(\frac{1}{\sqrt{K}})$.

The first contribution of this work is to present a new binary matrix model for coded caching. In Section II, we introduce the concept of using a constant row-weight binary matrix for describing the coded caching scheme. We call these as *caching matrices*. The 1s in the binary matrix indicate uncached subfiles in the users. Identity submatrices of the caching matrix correspond to transmissions which enable the clients (involved in any transmission) to decode precisely one missing subfile each from that transmission. Thus, 'covering' the 1s in the caching matrix using identity submatrices provides a valid delivery scheme. The framework we present using binary matrices are closely (and obviously, as the reader shall see) related to the PDA schemes. However the advantage is that this viewpoint opens up a much larger space, viz. the space of all constant row-weight binary matrices, for searching for good caching schemes.

Following this, we use the binary matrix model for constructing novel caching schemes derived from a variety of combinatorial designs. Towards that end, Section III describes important terminologies related to combinatorial designs. In Sections V and VI, we elaborate on the construction of caching matrices using different combinatorial designs. When we employ existing constructions of designs from combinatorics literature, the caching schemes which we get demand a low uncached fraction, i.e., $1 - \frac{M}{N} = \Theta(\frac{1}{\sqrt{K}})$. This is a disadvantage. However this disadvantage is traded off by a deep reduction in the rate as well as the subpacketization levels, with the schemes achieving a rate bounded by a constant ($R \leq 1$), with subpacketization levels being $O(K)$. Section IV summarizes all our constructions and discusses the asymptotics of each. We end the paper in Section VII with some promising directions that can possibly help us to remedy the issue of high cache requirements at the users.

Due to space restrictions, we omit some constructions, examples, and proofs in this five page version. However these are all made available in the longer version in [5].

Notations and Terminology: For any positive integer N, we denote by $[N]$ the set $\{1, \ldots, N\}$. For a set $\mathcal{X}$ and some positive integer $t \leq |\mathcal{X}|$, we denote the set of all t-sized subsets of $\mathcal{X}$ by $\binom{\mathcal{X}}{t}$. For a matrix A whose rows are indexed by a finite set $\mathcal{R}$ and columns are indexed by a finite set $\mathcal{C}$, the element in the r^{th} row ($r \in \mathcal{R}$) and l^{th} column ($l \in \mathcal{C}$) is denoted as $A(r, l)$. For sets A, B, $A \backslash B$ denotes the elements in A but not in B. For some element i, we also denote $A \backslash \{i\}$ by $A \backslash i$.

II. A BINARY MATRIX MODEL FOR CODED CACHING

In this section we describe how a coded caching scheme can be derived from a binary matrix with constant row weight.

Definition 1 (Caching Matrix)**.** *Consider a matrix C with entries from $\{0, 1\}$ with rows indexed by a K-sized set $\mathcal{U}$ and columns indexed by a F-sized set $\mathcal{F}$ such that the number of 1's in each row is constant (say Q). Then the matrix C defines a caching scheme with K users (indexed by $\mathcal{U}$), subpacketization F (indexed by $\mathcal{F}$) and $(1 - \frac{M}{N}) = \frac{Q}{F}$ as follows:*

- *User $u \in \mathcal{U}$ caches $W_{i,f} : \forall i \in [N]$ if $C(u, f) = 0$ and does not cache it if $C(u, f) = 1$.*

We then call the matrix C as a $(\mathcal{U}, \mathcal{F}, (1 - \frac{M}{N}))$ - Caching Matrix.

A subfile $W_{i,f}$ is said to be *missing* at a user u if it is not available at its cache. The demand of a user u in the delivery phase is denoted by W_{d_u} for some $d_u \in [N]$. In order to construct a transmission scheme, we first describe one transmission based on the above described matrix based caching scheme, which will serve a number of users. Note that a submatrix of C can be specified by a subset of the row indices $\mathcal{U}$ and a subset of column indices $\mathcal{F}$. We now define an *identity submatrix* C_i of matrix C.

Definition 2 (Identity Submatrix)**.** *An $l \times l$ submatrix C_i of the matrix C is an identity submatrix of size l if it contains the columns corresponding to the identity matrix of size l permuted in some way.*

The proof of the following lemma is straightforward and is made available in [5].

Lemma 1. *Consider an identity submatrix of C given by rows $\{u_1, u_2, .., u_l : u_i \in \mathcal{U}\}$ and columns $\{f_1, f_2, .., f_l : f_i \in \mathcal{F}\}$, such that $C(u_i, f_i) = 1, \forall i \in [l]$, while $C(u_i, f_j) = 0, \forall i, j \in [l]$ where $i \neq j$. For each $i \in [l]$, the subfile $W_{d_{u_i}, f_i}$ is not available at user u_i and can be decoded from the transmission $\sum_{i=1}^{l} W_{d_{u_i}, f_i}$.*

We shall use Lemma 1 to describe the complete transmission scheme. For that purpose we introduce few more terminologies. For a *caching matrix* C, suppose $C(u, f) = 1$ for some $u \in \mathcal{U}$ and $f \in \mathcal{F}$. The entry $C(u, f) = 1$ is said to be *covered* by the identity submatrix B if u and f correspond to some row and column index of B respectively.

Definition 3 (Identity Submatrix Cover)**.** *Consider a set $\mathfrak{C} = \{C_1, ..., C_S\}$ consisting of S identity submatrices of a caching matrix C such that any $C(u, f) = 1$ in C is covered by atleast one C_i such that $i \in S$. Then, $\mathfrak{C}$ is called an Identity Submatrix Cover of C.*

We now describe how an identity submatrix cover is used to form a transmission scheme.

Theorem 1. *Consider an identity submatrix cover $\mathfrak{C} = \{C_1, C_2, .., C_S\}$ of a caching matrix C. Then the transmission corresponding to $C_i : i \in [S]$ according to Lemma 1, is a valid transmission scheme (i.e the scheme satisfies all the user demands) for the caching scheme defined by C and the rate of the transmission scheme, $R = \frac{S}{F}$.*

Proof: Pick some arbitrary missing subfile $W_{d_u, f}$ of user u. Then $C(u, f) = 1$ and this entry of C will be covered by atleast one of the identity submatrices, say C_i in $\mathfrak{C}$ since $\mathfrak{C}$ is an identity submatrix cover of C. The transmission corresponding to the identity submatrix C_i given by Lemma 1 will ensure that the subfile $W_{d_u, f}$ will be decoded by the corresponding user u where it is missing. Hence, the transmissions corresponding to $C_i \in \mathfrak{C}$ enables decoding of any arbitrary missing subfile. Since the number of identity submatrices in $\mathfrak{C}$ is S, the rate of the transmission scheme is, $R = \frac{S}{F}$. ■

We also need the idea of an *overlap* between identity submatrices of C, which enables us to prove some results in this paper.

Definition 4 (Overlap)**.** *An overlap between identity submatrices occurs when some entry $C(u, f) = 1$ in matrix C is covered by more than one identity submatrix of C.*

III. INTRODUCTION TO COMBINATORIAL DESIGNS

In Section V and VI, we will use combinatorial designs to construct caching matrices. For that purpose we first review some of the basic definitions related to designs and their constructions. For more details reader is referred to [6] [7].

Definition 5 (Design $(\mathcal{X}, \mathcal{A})$)**.** *A design is a pair $(\mathcal{X}, \mathcal{A})$ such that the following properties are satisfied:*
(D1). $\mathcal{X}$ is a set of elements called points, and
(D2). $\mathcal{A}$ is a collection (i.e., multiset) of nonempty subsets of $\mathcal{X}$ called blocks.

We now define t-designs.

Definition 6 (t-designs)**.** *Let v, k, λ, and t be positive integers such that $v > k \geq t$. A t-(v, k, λ)-design (or simply t-design)*

is a design $(\mathcal{X}, \mathcal{A})$ such that the following properties are satisfied:
(T1). $|\mathcal{X}| = v$,
(T2). Each block contains exactly k points, and
(T3). Every set of t distinct points is contained in exactly λ blocks.

Consider a nonempty $Y \subseteq \mathcal{X}$ such that $|Y| = s \leq t$. Then there are exactly

$$\lambda_s = \lambda \frac{\binom{v-s}{t-s}}{\binom{k-s}{t-s}} \tag{1}$$

blocks in $\mathcal{A}$ that contain all the points in Y. It can also be shown that $b = \lambda_0 = \lambda \frac{\binom{v}{t}}{\binom{k}{t}}$ is the number of blocks in t-designs.

Example 1. *[Parametrized Constructions] A t-design with $\lambda = 1$ (i.e t-$(v, k, 1)$ design) is called a Steiner system and its existence is discussed in [8]. Some general constructions for Steiner systems can be found in [7]. Here we use a specific construction.*

- *A construction of Steiner system with parameters $t = 3,\ v = q^2 + 1,\ k = q + 1$ is presented in [6], where q is a prime power such that $q \geq 2$.*

In the following examples and some others in this paper, we drop the parentheses and the commas in writing the blocks explicitly (for instance block $\{l, m, n\}$ is written as lmn).

Example 2. *A 3-(8,4,1) design (Steiner system)*
$\mathcal{X} = \{1, 2, 3, 4, 5, 6, 7, 8\}$
$\mathcal{A} = \{1256, 3478, 1357, 2468, 1458, 2367, 1234, 5678, 1278, 3456, 1368, 2457, 1467, 2358\}$.

Definition 7 (Balanced Incomplete Block Design)**.** *t-Designs with $t = 2$ are called Balanced Incomplete Block Designs (BIBD) denoted as $(v,\ k,\ \lambda)$-BIBD.*

By (1) it follows that for $(v,\ k,\ \lambda)$-BIBD every point occurs in exactly $r = \lambda_1 = \frac{\lambda(v-1)}{k-1}$ blocks and there are exactly $b = \frac{vr}{k}$ blocks.

Example 3. *Let $\mathcal{X} = \{1, 2, 3, 4, 5, 6, 7\}$, and $\mathcal{A} = \{127, 145, 136, 467, 256, 357, 234\}$. This gives a (7,3,1)-BIBD.*

Example 4. *[Parametrized Constructions] Some constructions of BIBD known in literature are given below:*

- *BIBDs with parameters $v = n^2 + n + 1$, $k = n + 1$, $\lambda = 1$ are constructed in [6] using a projective plane of order n, where n is a prime power such that $n \geq 2$.*
- *BIBDs with parameters $v = n^2$, $k = n$, $\lambda = 1$ are constructed in [6] using an affine plane of order n where, n is a prime power such that $n \geq 2$.*

We will use in some constructions the following idea of the *incidence matrix* of a design.

Definition 8 (Incidence Matrix)**.** *Let $(\mathcal{X}, \mathcal{A})$ be a design where $\mathcal{X} = \{x_1, ..., x_v\}$ and $\mathcal{A} = \{A_1, ..., A_b\}$. The incidence matrix of $(\mathcal{X}, \mathcal{A})$ is the $v \times b$ binary matrix $M = (M(i,j))$ defined by the rule $M(i,j) = 1$ if $x_i \in A_j$, and $M(i,j) = 0$ if $x_i \notin A_j$.*

IV. Summary of Results

Table I summarizes all the caching parameters related to the coded caching schemes to be constructed in Section V and Section VI, from the designs given in Section III. In the longer version of this work [5], we give three more constructions using *transversal designs*, t-designs, and BIBDs. The parameters are based on those of the designs using which they are constructed.

Combinatorial Designs	$\left(\mathbf{1 - \frac{M}{N}}\right)$	**K**	**F**	**R**
BIBD ($\lambda = 1$) Section V	$\frac{k}{v}$	v	$\frac{v(v-1)}{k(k-1)}$	$\frac{k(k-1)}{v-1}$
t-design ($\lambda = 1$) (Scheme 1) Section VI	$\frac{\binom{k}{t}(v-t+1)}{\binom{v}{t}k}$	$\binom{v}{t-1}$	$\frac{\binom{v}{t}k}{\binom{k}{t}}$	$\frac{\binom{k-1}{t-1}\binom{k}{t}v}{\binom{v}{t}k}$

TABLE I: Parameters of Coded Caching Scheme based on Combinatorial Designs

A. Specific Constructions and their asymptotics

Applying the results of Table I to the parameterized constructions of designs as given in Section III, we get the following results.

1) BIBDs: The parameters of the transmission scheme (described in Section V) for the constructions described in Example 4 are as follows:

- BIBDs with parameters $v = n^2 + n + 1$, $k = n + 1$, $\lambda = 1$ will give a coded caching scheme with parameters $F = n^2 + n + 1$, $K = n^2 + n + 1$, Rate $= 1$, $(1 - \frac{M}{N}) = \frac{n+1}{n^2+n+1}$.
- BIBDs with parameters $v = n^2$, $k = n$, $\lambda = 1$ will give a coded caching scheme with parameters $F = n^2 + n$, $K = n^2$, Rate $= \frac{n}{n+1}$, $(1 - \frac{M}{N}) = \frac{1}{n}$.

Note that for the above two schemes, we have $F = O(K)$, and $1 - \frac{M}{N} = \Theta(\frac{1}{\sqrt{K}})$, while $R \leq 1$.

2) Steiner systems: For the constructions described in Example 1, the parameters of the transmission scheme presented in Section VI are as follows:

a) Scheme 1: t-designs with parameters $t = 3,\ v = q^2 + 1,\ k = q + 1,\ \lambda = 1$, will give a coded caching scheme with parameters $F = (q^2 + 1)(q + 1)$, $K = \frac{(q^2+1)q^2}{2}$, Rate $= \frac{(q-1)}{2(q+1)}$, $(1 - \frac{M}{N}) = \frac{(q-1)}{q(q^2+1)}$.

We note that for the above construction of the coded caching scheme, we have $F = O(K^{\frac{3}{4}})$, $1 - \frac{M}{N} = \Theta(\frac{1}{\sqrt{K}})$ and $R \leq 1$.

In the forthcoming sections, we provide constructions for caching schemes based on the above combinatorial designs. Each construction is contingent on the existence of the design of the considered type. In each such case, we define the caching matrix using the given design and obtain its parameters K, F, and $(1 - \frac{M}{N})$ based on the design parameters. We then define and prove an identity submatrix cover of the caching matrix based on the properties of the design.

V. BIBD (WITH $\lambda = 1$) BASED CODED CACHING SCHEME

Consider a $(v, k, 1)$-BIBD $(\mathcal{X}, \mathcal{A})$. We order the elements in $\mathcal{X}$ in some arbitrary way. For distinct $x, y \in \mathcal{X}$, we say that $x < y$ if x comes before y in the ordering of $\mathcal{X}$. Let the elements in block $A \in \mathcal{A}$ be denoted as $\{A(0), A(1), \ldots, A(k-1)\}$ where $A(0) < A(0) < .. < A(k-1)$. Note that each element in $\mathcal{X}$ occurs in r blocks. Thus each row of the incidence matrix C has weight r. Therefore, $(1-\frac{M}{N}) = \frac{r}{b} = \frac{k}{v}$ ($\because vr = bk$). $F = b = \frac{v(v-1)}{k(k-1)}$. Hence, the incidence matrix C of design $(\mathcal{X}, \mathcal{A})$ will give a $\left(v, \frac{v(v-1)}{k(k-1)}, \frac{k}{v}\right)$-caching matrix.

For $x \in \mathcal{X}$, let $B_x \triangleq \{A \in \mathcal{A} : x \in A\}$. Note that $|B_x| = r$, by the property of $\lambda = 1$ BIBD. In the next lemma, we describe a single identity submatrix of matrix C. For $j \in \{0, 1, \ldots, (k-1)\}$, we denote $(j+1) mod\ k$ by $(j \oplus_k 1)$.

Lemma 2. *For any $x \in \mathcal{X}$, let us denote B_x as $B_x = \{A_1, ..., A_r\}$ where $x = A_i(j_i) : j_i \in \{0, 1, \ldots, k-1\}$ i.e x is the j_i^{th} element in the block A_i. Consider the submatrix C_x of C whose columns are indexed by B_x and rows are indexed by $\{A_i(j_i \oplus_k 1) : i \in [r], A_i \in B_x\}$. Then C_x is an identity submatrix of C of size r.*

Proof: Note that there are r columns in submatrix C_x of C. Now we show that there are r rows. Due to the fact that $(\mathcal{X}, \mathcal{A})$ is a BIBD with $\lambda = 1$, the elements $\{x, A_i(j_i \oplus_k 1)\} \subset A_i$ only and does not lie in any other $A_{i'}$ for any $i' \neq i$. Thus the elements $A_i(j_i \oplus_k 1) \neq A_{i'}(j_{i'} \oplus_k 1)$ for any $i' \neq i$. Hence, there are r rows in C_x. Let us consider the row indexed by $A_i(j_i \oplus_k 1)$. Again by the property that $\lambda = 1$, it holds in C_x that the only column index corresponding to which there is a 1 in the row indexed by $A_i(j_i \oplus_k 1)$ is A_i and no other column. Thus each row of C_x has a single entry 1 in some column. Now consider an arbitrary column of C_x, say indexed by A_i. Since $A_{i'}(j_{i'} \oplus_k 1) \in A_i$ only if $i = i'$, thus the column A_i has a 1 only in the row $A_i(j_i \oplus_k 1)$. Hence, C_x is an identity submatrix of C of size r. ■

In next two lemmas we will prove that there is no overlap between the identity submatrices $C_x : x \in \mathcal{X}$ and that these identity submatrices will cover all the entries where $C(x, A) = 1$ in matrix C.

Lemma 3. *For distinct $x_1, x_2 \in \mathcal{X}$, there is no $x \in \mathcal{X}$, $A \in \mathcal{A}$ with $C(x, A) = 1$ such that $C(x, A)$ is covered by both C_{x_1} and C_{x_2}, where C_{x_1} and C_{x_2} are as defined in Lemma 2.*

Proof: Suppose $C(x, A) = 1$ is covered by both C_{x_1}, C_{x_2}. By the definition of C_{x_1} and C_{x_2}, it implies that $x_1, x_2 \in A$. Let the element x_1 and x_2 be present in j and j' position of A where $j \neq j'$. Therefore, $x = A(j \oplus_k 1) = x = A(j' \oplus_k 1)$, by our construction. But $j \neq j'$. This gives a contradiction. Hence, there is no $C(x, A) = 1$ which is covered by both C_{x_1} and C_{x_2}. ■

Lemma 4. *The set of matrices $\{C_x : x \in \mathcal{X}\}$ forms an identity submatrix cover of C.*

Proof: The total number of 1's in matrix C is equal to the product of the number of 1's in each row and the number of rows, and thus equal to rv. For each $x \in \mathcal{X}$ (note that $|\mathcal{X}| = v$), there exists an identity submatrix C_x. From Lemma 2 and 3, we see that each identity submatrix is of size r and no two such identity submatrices have overlaps. These identity submatrices will cover vr number of 1's in matrix C which is equal to total number of 1's in C. Hence, $\{C_x : x \in \mathcal{X}\}$ forms an identity submatrix cover of C. ■

We thus have the following theorem summarizing the caching scheme.

Theorem 2. *The incidence matrix of a $(v, k, 1)$- BIBD forms a $\left(K = v, F = \frac{v(v-1)}{k(k-1)}, (1 - \frac{M}{N}) = \frac{k}{v}\right)$ caching matrix. Further there is a transmission scheme with rate $R = \frac{k(k-1)}{v-1}$.*

Proof: The parameters of the caching matrix C have already been defined. By Lemma 2, 3 and 4, we have an identity submatrix cover consisting of v identity submatrices. Hence in Theorem 1, $S = v$ and rate $R = \frac{S}{F} = \frac{k(k-1)}{v-1}$. ■

Example 5. *Consider the $(7, 3, 1)$-BIBD as given in Example 3. We describe the identity submatrix C_1 (as per Lemma 2) corresponding to element '1' in $\mathcal{X}$. Element '1' is present in blocks* $\{127, 145, 136\}$. *Then the columns of identity submatrix matrix C_1 are indexed by* $127, 145, 136$ *and rows are indexed by* $2, 4, 3$ *(since* $2, 4, 3$ *are the next elements present after '1' in blocks* $127, 145, 136$ *respectively) of matrix C. In this manner, the submatrices $C_i : i \in \mathcal{X}$, gives us a rate 1 transmission scheme.*

VI. t-DESIGN BASED CODED CACHING SCHEMES

We now describe a t-design based caching scheme. We call this *Scheme 1* because in the longer version [5], we discuss another t-design based scheme which we call Scheme 2.

A. *Scheme*-1

Let $(\mathcal{X}, \mathcal{A})$ denote a t-$(v, k, 1)$ design. Let the blocks in this t-design be denoted as $\mathcal{A} = \{B_1, ..., B_b\}$ where $b = \frac{\binom{v}{t}}{\binom{k}{t}}$. We construct a binary matrix T as follows. Let the rows of T be indexed by all the $(t-1)$-sized subsets of $\mathcal{X}$. Let the columns be indexed by $\{(y, B) : y \in B, B \in \mathcal{A}\}$. For some $D \in \binom{\mathcal{X}}{t-1}$, the matrix $T = (T(D, (y, B)))$ is defined by the rule,

$$T(D, (y, B)) = \begin{cases} 1, & \text{if } D \cup \{y\} \subset B, |D \cup \{y\}| = t \\ 0, & \text{otherwise.} \end{cases}$$

The number rows in matrix T is $\binom{v}{t-1}$. The number of columns in matrix T is $bk = \frac{\binom{v}{t}k}{\binom{k}{t}}$. The number of 1's in each row of T is $\lambda_{t-1}\binom{k-t+1}{1} = (v-t+1)$. Hence, the matrix T gives a $\left(\binom{v}{t-1}, \frac{\binom{v}{t}k}{\binom{k}{t}}, \frac{\binom{k}{t}(v-t+1)}{\binom{v}{t}k}\right)$-caching matrix.

In the next lemma we describe an identity submatrix of matrix T. Towards that end, we need to denote a few sets. For some $y \in \mathcal{X}$, define $\mathcal{B}_y \triangleq \{B \in \mathcal{A} : y \in B\}$, i.e the set of blocks containing y. By (1), $|\mathcal{B}_y| = \lambda_1 = \frac{\binom{v-1}{t-1}}{\binom{k-1}{t-1}}$. Denote $\mathcal{B}_y$ by $\mathcal{B}_y = \{B_1, \ldots, B_{\lambda_1}\}$. For any B_i, denote by $\{D_{i,j} : j \in$

$[\binom{k-1}{t-1}]\}$ the set $\binom{B_i\backslash y}{t-1}$, i.e the set of all $(t-1)$-sized subsets of $B_i\backslash y$.

Lemma 5. *For some $j \in [\binom{k-1}{t-1}]$ and $y \in \mathcal{X}$, consider the submatrix $T_{y,j}$ of T whose rows are indexed by $\{D_{i,j} : \forall i \in [\lambda_1]\}$ ($D_{i,j}$ as defined above) and the columns are indexed by $\{(y, B_i) : B_i \in \mathcal{B}_y\}$. Then $T_{y,j}$ is an identity submatrix of T of size λ_1.*

Proof: Clearly, the number of columns in $T_{y,j}$ is λ_1. First note that the rows in $\{D_{i,j} : \forall i \in [\lambda_1]\}$ are all distinct, i.e $D_{i,j} \neq D_{i',j}$ for $i \neq i'$. If not, note that $D_{i,j} \cup y = D_{i',j} \cup y \in B_i \cap B_{i'}$. But this contradicts the fact that any t- sized subset of $\mathcal{X}$ occurs in only one block. Thus $|\{D_{i,j} : i \in [\lambda_1]\}| = \lambda_1$, and $T_{y,j}$ is a square matrix of size λ_1. Now consider a row of $T_{y,j}$ indexed by $D_{i,j}$ for some particular $i \in [\lambda_1]$. Suppose a column indexed by (y, B) for some $B \in \mathcal{B}_y$ has a 1 in the row indexed by $D_{i,j}$. Then it means that $D_{i,j} \cup \{y\} \subset B$. But there is precisely one block B such that $D_{i,j} \cup \{y\} \subset B$, which is precisely $B = B_i$ (as $\lambda = 1$). Thus each row of $T_{y,j}$ has only one entry which is 1. Now, consider a column of $T_{y,j}$ indexed by (y, B_i) for some $B_i \in \mathcal{B}_y$. Suppose for some $D_{i',j}$, the row indexed by $D_{i',j}$ has 1 in the column indexed by (y, B_i). Then it must be that $D_{i',j} \subset B_i\backslash y$ and hence $D_{i',j} \cup \{y\} \subset B_i$. Once again, because of the property of t-design with $\lambda = 1$, we have that $i = i'$ (else $D_{i',j} \cup \{y\} \in B_i \cap B_{i'}$, which is a contradiction). Hence, each column of $T_{y,j}$ has precisely only one entry that is 1. This proves the lemma. ∎

In the next two lemmas we will prove that there is no overlap between the identity submatrices and that these identity submatrices will cover all the entries where $T(D, (y, B)) = 1$ in matrix T.

Lemma 6. *Any $T(D, (y, B)) = 1$ such that $y \in \mathcal{X}, B \in \mathcal{A}, D \in \binom{\mathcal{X}}{t-1}$ will be covered by exactly one identity submatrix of T (as defined in Lemma 5).*

Proof: Let $T(D, (y, B)) = 1$ be covered by an identity submatrix $T_{y',j}$ of T. As $T(D, (y, B)) = 1$, we have that $D \cup \{y\} \subset B$. By definition of $T_{y',j}$ in Lemma 5, we must first have $y = y'$. Further it must be that $D \subset B_i\backslash y$ for some B_i which contains y. Therefore, $\{y\} \cup D \subset B_i$, which means $B_i = B$ (as $\lambda = 1$). Hence the unique transmission which covers $T(D, (y, B)) = 1$ is $T_{y,j}$ where j is such that $D = D_{i,j}$ is the unique j^{th} set in $\binom{B\backslash y}{t-1}$. ∎

Lemma 7. *The set of matrices $\{T_{y,j} : \forall y \in \mathcal{X},\ \forall j \in [\binom{k-1}{t-1}]\}$ forms an identity submatrix cover of T.*

Proof: The total number of 1's in T is equal to the product of the number of 1's in each row and the number of rows, and thus equal to $(v-t+1)\binom{v}{t-1} = t\binom{v}{t}$. The size of each identity submatrix is $\frac{\binom{v-1}{t-1}}{\binom{k-1}{t-1}}$. Since we have an identity submatrix $T_{y,j}$ for each $j \in [\binom{k-1}{t-1}]$, $y \in \mathcal{X}$, the number of identity submatrices $= \binom{k-1}{t-1}v$. Moreover, from Lemma 6, there are no overlaps between the identity submatrices. Hence, the total number of 1's covered by all identity submatrices $= \binom{k-1}{t-1}\frac{\binom{v-1}{t-1}v}{\binom{k-1}{t-1}} = \binom{v-1}{t-1}v = t\binom{v}{t}$ which is equal to the total number of 1's in matrix T. This proves the lemma. ∎

We thus have the below theorem summarizing the caching scheme.

Theorem 3. *The matrix T of a t-$(v, k, 1)$-design forms a $\left(K = \binom{v}{t-1}, F = \frac{\binom{v}{t}k}{\binom{k}{t}}, (1 - \frac{M}{N}) = \frac{\binom{k}{t}(v-t+1)}{\binom{v}{t}k}\right)$-caching matrix. Further there is a transmission scheme with rate $R = \frac{\binom{k-1}{t-1}\binom{k}{t}v}{\binom{v}{t}k}$.*

Proof: The parameters of the caching matrix T have already been defined. By Lemma 5, 6 and 7, we have an identity submatrix cover consisting of $\binom{k-1}{t-1}v$ identity submatrices. Hence in Theorem 1, $S = \binom{k-1}{t-1}v$ and rate $R = \frac{S}{F} = \frac{\binom{k-1}{t-1}\binom{k}{t}v}{\binom{v}{t}k}$. ∎

Example 6. *Consider the 3-$(8, 4, 1)$ design as given in Example 2. We now describe an identity submatrix namely $T_{4,1}$. The set of blocks containing element '4' are denoted by $\mathcal{B}_4 = \{3478, 2468, 1458, 1234, 3456, 2457, 1467\}$. The identity submatrix $T_{4,1}$ has rows indexed by $\{37, 26, 15, 12, 35, 25, 16\}$ and columns indexed by $\{(4, B_i) : B_i \in \mathcal{B}_4\}$ of matrix T. In this manner we can obtain an identity submatrix cover of T using the matrices $T_{y,j} : \forall y \in \mathcal{X},\ \forall j \in [\binom{k-1}{t-1}]$, which gives a transmission scheme with rate $= \frac{3}{7}$.*

VII. Discussion

An important precursor to this work is [3], which discusses caching schemes via resolvable designs. However the approach of [3] is different from the approach we take here. Inspite of having small rates and subpacketization levels, the schemes constructed here still suffer from the drawback of requiring large local cache sizes, which is impractical. This can possibly be remedied by looking at designs with a higher value of λ.

VIII. ACKNOWLEDGMENT

This work was supported partly by the Early Career Research Award (ECR/2016/000447) and the MATRICS grant (MTR/2017/000827) from Science and Engineering Research Board (SERB) to Prasad Krishnan.

References

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

[2] Q. Yan, M. Cheng, X. Tang, and Q. Chen, "On the placement delivery array design for centralized coded caching scheme," *IEEE Transactions on Information Theory*, vol. 63, no. 9, pp. 5821–5833, 2017.

[3] L. Tang and A. Ramamoorthy, "Coded caching schemes with reduced subpacketization from linear block codes," *IEEE Transactions on Information Theory*, vol. 64, no. 4, pp. 3099–3120, 2018.

[4] P. Krishnan, "Coded caching via line graphs of bipartite graphs," in *2018 IEEE Information Theory Workshop (ITW)*, pp. 1–5, IEEE, 2018.

[5] S. Agrawal, K. Sree, and P. Krishnan, "Coded caching based on combinatorial designs," *arXiv preprint arXiv:1901.06383*, 2019.

[6] D. R. Stinson, *Combinatorial designs: constructions and analysis*. Springer Science & Business Media, 2007.

[7] C. J. Colbourn and J. H. Dinitz, *Handbook of combinatorial designs*. CRC press, 2006.

[8] P. Keevash, "The existence of designs," *arXiv preprint arXiv:1401.3665*, 2014.