A New Design Framework on D2D Coded Caching with Optimal Rate and Less Subpacketizations

Xiang Zhang*, Xianfeng (Terry) Yang†, and Mingyue Ji*
Department of Electrical and Computer Engineering, University of Utah*
Department of Civil & Environmental Engineering, University of Utah†
Email: *{xiang.zhang, mingyue.ji}@utah.edu, †x.yang@utah.edu

***Abstract*—In this paper, we propose a new design framework on Device-to-Device (D2D) coded caching networks with optimal communication load (rate) but significantly less file subpacketizations compared to that of the well-known D2D coded caching scheme proposed by Ji, Caire and Molisch (JCM). The proposed design framework is referred to as the *Packet Type-based (PTB) design*, where each file is partitioned into packets according to their pre-defined types while the cache placement and user multicast grouping are based on the packet types. This leads to the so-called *raw packet saving gain* for the subpacketization levels. By a careful selection of transmitters within each multicasting group, a so-called *further splitting ratio gain* of the subpacketizatios can also be achieved. By the joint effect of the *raw packet saving gain* and the *further splitting ratio gain*, an order-wise subpacketization reduction can be achieved compared to the JCM scheme while preserving the optimal rate. In addition, as the first time presented in the literature according to our knowledge, we find that unequal subpacketizaton is a key to achieve subpacketization reductions when the number of users is odd. As a by-product, instead of directly translating shared link caching schemes to D2D caching schemes, at least for the sake of subpackeitzation, a new design framework is indeed needed.**

I. Introduction

Coded caching has been shown to be an efficient approach to handle dramatically increased traffic in the current Internet. In [1], Maddah-Ali and Niesen (MAN) introduced a centralized shared-link caching network model, where a central controller serves K users, each of which is equipped with a cache of size M files from a library of N files, via an errorless broadcast link (shared link). In order to achieve the optimal worst-case rate (i.e., the total number of file transmissions in the network) under uncoded cache placement, a cache placement and a coded delivery scheme were proposed [1] and required to partition each file into $\binom{K}{t}$ packets where $t = \frac{KM}{N} \in \mathbb{Z}^+$. Later, [2] shows that this file subpacketization level is necessary to achieve the optimal rate under a so-called Placement Delivery Arrary (PDA) design based on uncoded cache placement. In order to reduce the subpacketization level, the authors in [3]–[7] proposed schemes based on various combinatorial designs and showed that the subpacketization can be reduced at a cost of a higher transmission rate (or higher traffic load). Ji, Molisch and Caire (JCM) extended the shared-link caching model to Device-to-Device (D2D) coded caching networks, where no central controller is present and all users serve each other via individual shared links [8]. This network model has been studied extensively in the literature and a few examples of the information-theoretic study are given in [9]–[12]. Under uncoded cache placement, [8] proposed a caching scheme referred to as the *JCM scheme* that achieves the optimal worst-case rate of $R(M) = \frac{N}{M}\left(1 - \frac{M}{N}\right)$ when $N \geq K$. In this case, $R(M)$ is surprisingly not a function of K and hence it is scalable. In order to achieve this rate, the required number of packets (*subpacketization level*) is $F^{\text{JCM}} = t\binom{K}{t}$, which can be impractical for large K. Efforts have been made in reducing the subpacketization levels for the D2D coded caching problem [13]–[16]. For example, a design approach named *D2D placement delivery array* (DPDA) was introduced in [14], which designed new DPDA schemes when $t = 2, t = K - 2$, for which the JCM scheme is actually not optimal in terms of subpacketization although it achieves the optimal rate.

In this paper, we propose a new design framework called *Packet Type-based (PTB) design* tailored for subpacketization reduction in D2D coded caching while preserving the optimal rate. Specifically, in the PTB design, D2D users (or nodes) are first partitioned into multiple groups. Then packet types are designed based on the user grouping and different types of multicasting groups are designed based on the packet types. Note that the JCM scheme contains all the packet types and all multicasting group types; any $t + 1$ users form a multicasting group and every user in each multicasting group transmit "symmetric" coded multicast message that is useful to all the group members. In contrast, the proposed PTB scheme excludes certain packet types which leads to a reduction of the subpacketization level. This is referred to as the *raw packet saving gain*. In addition, in each multicasting group, it is possible that not all nodes will perform as transmitters. Hence, based on a careful selection of the transmitters within each type of multicasting group, a so-called *further splitting ratio gain* can also be obtained. While preserving the optimal rate, the joint effect of the *raw packet saving gain* and the *further splitting ratio gain* can lead to an order-wise subpacketization reduction compared to the JCM scheme, where none of these gains is available. In fact, the PTB design problem can be cast into an integer optimization problem subject to node cache memory constraints and the design variables are the choices of possible transmitters within each multicasting group type. Moreover, according to our knowledge, it is the first time in the literature showing that unequal subpacketizaton is one of the keys to achieve a subpacketization gain when K is odd.

In [8], in order to achieve the optimal rate, the JCM scheme proposed a direct translation from MAN scheme [1] by splitting each packet further into t packets. It turns out that when the cache placement is uncoded and the delivery scheme is one-shot (i.e., every user can decode one requested packet from each coded transmission), this translation holds in general and it seems that the design procedure for the D2D coded caching scheme should be that 1) designing a shared-link coded caching scheme; 2) translating it into D2D coded caching scheme. As a by-product of the PTB design, we show that the above design methodology is not optimal in terms of subpacketization in general. Hence, in order to achieve good subpackeitzations in D2D coded caching networks, a new design framework is indeed needed.

Notation Convention: $|\cdot|$ represents the cardinality of a set. $\mathbb{Z}^+$ denotes the non-negative integer set. $[n] := \{1, \cdots, n\}$, $[m : n] := \{m, m+1, \cdots, n\}$ for some integers $m < n$. $\mathbf{a}^n := (a, a, \cdots, a)$ and $|\mathbf{a}^n| = n$.

II. Problem Formulation and Illustrations

A. General Problem Description

Consider a D2D caching network with a user set $\mathcal{U}$ where $|\mathcal{U}| = K$. Each file from a library $\{W_n : n \in [N]\}$ of N files consists of F packets with equal length.[1] The system operates in two separate phases, *i.e.*, the *cache placement phase* and *delivery phase* as described in [8]. In the cache placement phase, each user k stores up to MF packets from the file library. This phase is done without the knowledge of the users' requests. In the delivery phase, each user k reveals its request for a specific file $W_{d_k}, d_k \in [N]$ to other users. Let $\mathbf{d} := (d_1, d_2, \cdots, d_K)$ denote the user demand vector. Since users have already cached some of the files, the task in the delivery phase is to design a corresponding transmission scheme for each user based on the cache placement and the user demand vector so that the users' demands can be satisfied with vanishing error probability. The objective is to minimize the transmission rate defined as the total number of transmitted bits normalized by the file size. In this paper, our goal is to propose a new design framework based on combinatorial optimizations such that the subpacketization level of each file is significantly reduced while preserving the optimal rate. In the rest of this paper, we use F^{JCM} and F^{PTB} to represent the subpacketization level of the JCM scheme and the proposed PTB scheme.

B. JCM D2D Coded Caching Scheme

The cache placement in JCM scheme is the same as MAN cache placement scheme introduced in [1]. Let $t := \frac{MK}{N}$, each file W_n is divided into $\binom{K}{t}$ disjoint sub-files denoted by $W_{n,\mathcal{T}}$, where $\mathcal{T} \subset [K]$ and $|\mathcal{T}| = t$. The size of each sub-file is $F/\binom{K}{t}$ packets. Each user u caches all the packets in sub-files $W_{n,\mathcal{T}}$, for all $n \in [N]$ and $u \in \mathcal{T}$.

[1] The equal length assumption is for the ease of presentation. In practice, the length of each packet may be a design parameter since only the file length should be fixed. In the main results, we will see that if this assumption is relaxed, it is possible to achieve additional gain in terms of the subpacketizations.

In the delivery phase, for each multicasting group with $t+1$ users, it can be seen that there exists a unique sub-file that is available at t users and is requested by the remaining user. In order to symmetrize all transmissions (minimize the communication load), each sub-file needs be divided into t packets, where $W_{n,\mathcal{T}} = \{W_{n,\mathcal{T}}^{(j)}\}$, where $j \in [t]$. It can be seen that each file needs to be partitioned into $t\binom{K}{t}$ packets. Hence, the transmitted coded multicast message for user u is given by $Y_{\mathcal{A}^u}^u = \bigoplus_{k \in \mathcal{A}^u} W_{d_k, \{\mathcal{A}^u \cup \{u\} \setminus \{k\}\}}^{(j)}$, for all the user groups $\mathcal{A}^u \subset [K] \setminus \{u\}$ with $|\mathcal{A}^u| = t$ and for $j \in [t]$ that is not chosen yet. It can be seen that all users can decode all the desired sub-files and the transmission rate is given by $R = \frac{N}{M}(1 - \frac{M}{N})$, which is optimal if $N > K$.

C. An Example to Illustrate PTB schemes

In this section, we present an example to show the key ideas of the PTB design approach and its difference from the JCM scheme. We consider a D2D coded caching network with parameters $K = 9, N = 3, M = 2$ and $t = KM/N = 6$. The user set is $\mathcal{U} = [9]$. We evenly partition $\mathcal{U}$ to 3 groups, denoted as $\mathcal{Q}_1, \mathcal{Q}_2$ and $\mathcal{Q}_3$, where $\mathcal{Q}_1 = \{1, 2, 3\}$, $\mathcal{Q}_2 = \{4, 5, 6\}$ and $\mathcal{Q}_3 = \{7, 8, 9\}$ (see Fig. 1). We use a partition vector $\mathbf{q} = (3, 3, 3)$ to indicate such a node grouping, where each element in $\mathbf{q}$ denotes the number of nodes in each group. One of the key ideas of the proposed approach is to introduce the *packet type*, which is a partition of $t = 6$ nodes from all user groups. In this example, there are three partition vectors (i.e, three packet types) $\mathbf{v}_1 = (2, 2, 2)$, $\mathbf{v}_2 = (3, 2, 1)$ and $\mathbf{v}_3 = (3, 3, 0)$, where the sum of all the elements in each partition vector is 6. The packets of type $\mathbf{v}_1$ are cached by 2 users in $\mathcal{Q}_1$, 2 users in $\mathcal{Q}_2$ and 2 users in $\mathcal{Q}_3$ respectively. The meaning of packet types $\mathbf{v}_2$ and $\mathbf{v}_3$ follows similarly. Fig. 1 shows an example for all three packet types. For instance, for packet type $\mathbf{v}_1$, Fig. 1 illustrates the case where the packets are cached by users $\mathcal{T} = \{2, 3, 5, 6, 8, 9\}$. When all packets have the same size, in order to design the cache placement, we need to decide the number of packets for each type. It can be shown that packet type $\mathbf{v}_3$ can be *excluded*. This means that only the packet types $\mathbf{v}_1$ and $\mathbf{v}_2$ are needed for designing the D2D coded caching scheme. This results in $\binom{3}{2}^3 + \binom{3}{1}\binom{3}{2}\binom{2}{1}\binom{3}{1} = 81$ sub-files while the JCM scheme requires $\binom{9}{6} = 84$. At this stage, the saving of the PTB design in terms of sub-files is $84 - 81 = 3$, which is referred to as the *raw packet saving gain*. It can also be seen that the JCM scheme includes all possible packet types $\mathbf{v}_1, \mathbf{v}_2$, and $\mathbf{v}_3$. [2] Moreover, each sub-file of type $\mathbf{v}_1$ needs to be partitioned into 4 packets and each sub-file of type $\mathbf{v}_2$ needs to be partitioned into 3 packets according to the PTB scheme. Note that each sub-file in the JCM scheme needs to be partitioned into $t = 6$ packets. This further reduction of subpacketization of PTB scheme is referred to as the *further splitting ratio gain*. Hence, using the PTB design, we can compute the number of total packets needed for each file as $\binom{3}{2}^3 \cdot 4 + \binom{3}{1}\binom{3}{2}\binom{2}{1}\binom{3}{1} \cdot 3 = 270$

[2] Note that the subpacketization is not final yet, we call the unit of the file partition as "sub-files", not packet, which is the smallest unit in a file.

while the number of packet per file required by JCM is $t\binom{K}{t} = 6\binom{9}{6} = 504$. Clearly, the cache placement, determined by the packet type, is that node k stores any packet $W_{n,\mathcal{T}}^{(j)}$ for $k \in \mathcal{T}$, $j \in [4]$ $n \in [N]$. For example, Fig.1 shows the four packets $\{W_{n,\mathcal{T}}^{[j]} : j \in [4]\}$ derived from the type $\mathbf{v}_1$ sub-file $W_{n,\mathcal{T}}$ where $\mathcal{T} = \{2,3,5,6,8,9\}$.

In the delivery phase, the JCM scheme exploits the "symmetric" multicasting group structure , where any $t+1$ out of K users can form a multicasting group and any node in each multicasting group is a transmitter. For the proposed PTB scheme, this "symmetry" breaks, which means that the multicasting groups have to be designed according to the packet types and the transmitter in each multicasting group has to be designated specifically. This means that we can have different "types" of multicasting group. In this example, we have two types of multicasting groups denoted as $\mathbf{s}_1 = (3,3,1^*)$ and $\mathbf{s}_2 = (3,2^*,2^*)$, where each element in $\mathbf{s}_1$ and $\mathbf{s}_2$ means the number of users from the corresponding user group and the symbol "$*$" means that all nodes in the corresponding user group are selected as transmitters. For example, for a multicasting group $\mathcal{S} = [7]$ of type $\mathbf{s}_1$, all users in $\mathcal{Q}_1$ and $\mathcal{Q}_2$ are included and one node 7 from $\mathcal{Q}_3$ is included. The only transmitter in this multicasting group is node 7 from $\mathcal{Q}_3$. Note that the JCM scheme includes all types of multicasting groups in general and every node in each multicasting group is a transmitter.

Next, we will illustrate the design of the coded multicast message. For a type-$\mathbf{s}_1$ multicasting group $\mathcal{S}_1 = [7]$, node 7 is the only transmitter and it transmits three coded multicast messages $\bigoplus_{k\in[6]} W_{d_k,\mathcal{S}_1\backslash\{k\}}^{(j)}$, $j = 1,2,3$ to other nodes in $\mathcal{S}_1$. Each node k recovers its desired packets $\{W_{d_k,\mathcal{S}_1\backslash\{k\}}^{(j)} : j = 1,2,3\}$ with the help of the cached packets while node 7 itself only transmits but receives nothing. For a type-$\mathbf{s}_2$ multicasting group $\mathcal{S}_2 = [9]\backslash\{6,9\}$, the set of type-$\mathbf{v}_1$ and $\mathbf{v}_2$ subpackets involved are $\{W_{d_k,\mathcal{S}_2\backslash\{k\}}^{(j)} : j \in [4], k \in [3]\}$ and $\{W_{d_k,\mathcal{S}_2\backslash\{k\}}^{(j)} : j \in [3], k \in \mathcal{S}_2\backslash\mathcal{Q}_1\}$ respectively. Denote $W^{(j)} := \bigoplus_{k\in[3]} W_{d_k,\mathcal{S}_2\backslash\{k\}}^{(j)}, j \in [4]$. Each node $k \in \{4,5,7,8\}$ sends a coded multicast message Y_k as follows:

$$
\begin{aligned}
Y_4 &= W^{(1)} \oplus W_{d_5,\mathcal{S}_2\backslash\{5\}}^{(1)} \oplus W_{d_7,\mathcal{S}_2\backslash\{7\}}^{(1)} \oplus W_{d_8,\mathcal{S}_2\backslash\{8\}}^{(1)}\\
Y_5 &= W^{(2)} \oplus W_{d_4,\mathcal{S}_2\backslash\{4\}}^{(1)} \oplus W_{d_7,\mathcal{S}_2\backslash\{7\}}^{(2)} \oplus W_{d_8,\mathcal{S}_2\backslash\{8\}}^{(2)}\\
Y_7 &= W^{(3)} \oplus W_{d_4,\mathcal{S}_2\backslash\{4\}}^{(2)} \oplus W_{d_5,\mathcal{S}_2\backslash\{5\}}^{(2)} \oplus W_{d_8,\mathcal{S}_2\backslash\{8\}}^{(3)}\\
Y_8 &= W^{(4)} \oplus W_{d_4,\mathcal{S}_2\backslash\{4\}}^{(3)} \oplus W_{d_5,\mathcal{S}_2\backslash\{5\}}^{(3)} \oplus W_{d_7,\mathcal{S}_2\backslash\{7\}}^{(3)}
\end{aligned}
$$

from which we can see that all nodes can recover their desired packets. Since each coded message is simultaneously useful for $t = 6$ nodes, the transmission rate is $\frac{N}{M} - 1$. The transmission procedure for other multicasting groups is similar.

D. General Packet Type Based (PTB) Design Framework

The specific design proposed in the previous example is not unique and can be generalized by solving the following optimization problem, where the solutions are called PTB designs.

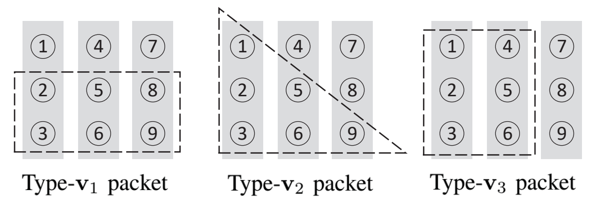

Fig. 1. An illustration of packet types under node grouping $\mathbf{q} = (3,3,3)$ with three specific groups $\mathcal{Q}_1 = \{1,2,3\}$, $\mathcal{Q}_2 = \{4,5,6\}$ and $\mathcal{Q}_3 = \{7,8,9\}$.

$$
\begin{aligned}
\min_{\boldsymbol{\alpha}^{\mathrm{LCM}},\mathbf{F}} \quad & F^{\mathrm{PTB}} = \boldsymbol{\alpha}^{\mathrm{LCM}}\mathbf{F}^{\mathrm{T}} && \text{(1a)}\\
\text{s.t.} \quad & \boldsymbol{\alpha}^{\mathrm{LCM}} \in \Phi, && \text{(1b)}\\
& \boldsymbol{\alpha}^{\mathrm{LCM}}\Delta\mathbf{F}_i^{\mathrm{T}} = 0, i \in [N_d - 1] && \text{(1c)}
\end{aligned}
$$

where $\mathbf{F}$ denotes the *raw packet number vector*; $\boldsymbol{\alpha}^{\mathrm{LCM}}$ denotes the *further splitting vector* and all the notations will be explained in detail in the Appendix. It can be seen that this is an integer optimization problem and cannot be solved analytically in general. In the following, we will provide some solutions in some specific parameter regimes. Note that the solutions of this combinatorial problem achieve the transmission rate $\frac{N}{M}(1 - \frac{M}{N})$, which is optimal when $K > N$.

III. Main Results

In this section, we present the main theorems. First, we consider the scenario when K is even and t is large.

Theorem 1: For even $\bar{t} := K - t$, where $K = 2m$, using the PTB design framework, the rate of $\frac{N}{M} - 1$ in D2D coded caching networks is achievable and

$$
\frac{F^{\mathrm{PTB}}}{F^{\mathrm{JCM}}} = \Theta\left(\frac{f(\bar{t})}{K - \bar{t}}\right), \tag{2}
$$

where $f(\bar{t}) := \prod_{i=1}^{\frac{\bar{t}}{2}}(2i-1)$ is a function which depends only on $\bar{t}$. Moreover, $\forall K \geq 2\bar{t}$, and $\bar{t} = O(\log\log K)$, $\frac{F}{F^{\mathrm{JCM}}}$ vanishes as K goes to infinity. □

From Theorem 1, it can be seen that when $K \leq 2t$ is even and t is large enough (*i.e.*, $t = K - O(\log\log K)$), an order gain in terms of subpacketization can be obtained using the PTB design compared to the JCM scheme while preserving the optimal rate. However, it can be seen that for small t, the PTB design achieving Theorem 1 may result in an even worse subpacketization compared to the JCM scheme. In the following theorem, we provide a general result for even K and t based on a specific PTB design demonstrating subpacketization gains compared to the JCM scheme when t is small.

Theorem 2: For $(K,t) = (2q,2r)$ with $q \geq t+1$ and $r \geq 1 (r \in \mathbb{Z}^+)$, using the PTB design framework with the two-group equal grouping, *i.e.*, $\mathbf{q} = (\frac{K}{2}, \frac{K}{2})$, the rate of $\frac{N}{M} - 1$ of D2D coded caching networks is achievable by the further splitting ratio vector $\boldsymbol{\alpha}^{\mathrm{LCM}} = (0,1,2,\cdots,r)$. Further, when $r \geq 2$, we have $\frac{F^{\mathrm{PTB}}}{F^{\mathrm{JCM}}} < \frac{1}{2}\left(1 - \frac{1}{2^{t-1}}\right)$. □

When K is odd, it is surprisingly more difficult than the case of even K. In Section II-C, we provided an example showing that when $K = 3m$ and $t = K - 3$, $m \in \mathbb{Z}^+$, it is possible to exploit an equal user grouping to achieve an order gain of subpacketization level compared to the JCM scheme. However, in general, we may need to use the general PTB design framework that exploits the heterogeneous packet size.

Theorem 3: For $(K, t) = (2q + 1, 2r)$ with $q \geq 2r + 1, r \geq 1 (r \in \mathbb{Z}^+)$, using the two-group unequal grouping $\mathbf{q} = (\frac{K+1}{2}, \frac{K-1}{2})$, the rate of $\frac{N}{M} - 1$ in D2D coded caching networks is achievable by the further splitting ratio vector $\boldsymbol{\alpha}^{\text{LCM}} = (0, 2, 4, \cdots, t-2, t, t, \cdots, t)$ and we have

$$\frac{F^{\text{PTB}}}{F^{\text{JCM}}} < \frac{1}{t}\left(\frac{\binom{t}{r}}{2^t} - 1\right) + 1 \tag{3}$$

$\square$

From Theorem 3, it can be seen that by using the general PTB design framework with the consideration of heterogeneous subpacket size, when K is odd, a constant gain in terms of subpacketization compared to the JCM scheme can be achieved while preserving the optimal rate when t small.

IV. Conclusion

This paper proposed a new design framework called PTB designs for D2D coded caching networks. The key ideas of this specific design are 1) classifying packets by the packet types; 2) grouping users into multicasting groups based on the cached packet types and 3) asymmetrically assigning transmitters in each multicasting group. This new design approach is completely different from the original coded D2D caching schemes and shows that the coded D2D caching problem may need a new design idea in contrast to firstly designing a centralized coded caching problem and then convert it to a D2D coded caching scheme.

Acknowledgement

This work is supported by NITC 1235 - Connected Vehicle System Design for Signalized Arterials and is also supported through the INL Laboratory Directed Research & Development (LDRD) Program under DOE Idaho Operations Office Contract DE-AC07-05ID14517, and NSF Awards 1817154 and 1824558.

Appendix A
General PTB Design Framework

We will present the general PTB design framework by decomposing it into the concepts including *Node Grouping*, *Packet Type*, *Multicasting Group Type*, *Further Splitting Ratio (FSR)*, *Further Splitting Ratio Table (FSRT)*, *Memory Constraint Table (MCT)* and *PTB Design as an Integer Optimization Problem*.

1) Node Grouping: The user set $\mathcal{U}$ is partitioned into $m \in \mathbb{Z}^+$ non-empty groups denoted by $\mathcal{Q}_1, \mathcal{Q}_2, \cdots, \mathcal{Q}_m$, where the i-th group contains $|\mathcal{Q}_i| = q_i$ nodes. We use a *partition vector* $\mathbf{q} := (q_1, q_2, \cdots, q_m, 0, 0, \cdots, 0)(|\mathbf{q}| = K)$ to represent such a node grouping, satisfying $\sum_{i=1}^{m} q_i = K$ and $q_1 \geq q_2 \geq \cdots \geq q_m > 0$. For a specific partition $\mathbf{q}$, there are actually multiple ways to assign the set of K nodes, but they are all considered the same partition/grouping. The number of groups m and the number of nodes contained in each group, *i.e.*, $\{q_i\}_{i \in [m]}$ are parameters to be designed. Let N_d denote the number of distinct elements in $\mathbf{q}$. We define a *unique group* as the union of the non-empty groups containing the same number of nodes. The i-th ($i \in [N_d]$) *unique group*, denoted by $\mathcal{U}_i$, contains ψ_i groups and each of these groups contains β_i nodes, *i.e.*, $|\mathcal{U}_i| = \psi_i \beta_i$. It is clear that $\sum_{i=1}^{N_d} \beta_i \psi_i = K$ and $\sum_{i=1}^{N_d} \psi_i = m$. For example, let $K = 7$ and the user set be $\mathcal{U} = [K]$. $\mathbf{q} = (3, 2, 1, 1, 0, 0)$ is a partition vector representing a partition of $\mathcal{U}$ into $m = 4$ groups which are $\mathcal{Q}_1 = \{1, 2, 3\}$, $\mathcal{Q}_2 = \{4, 5\}$, $\mathcal{Q}_3 = \{6\}$ and $\mathcal{Q}_4 = \{7\}$. In this case, there are $N_d = 3$ unique groups, *i.e.*, $\mathcal{U}_1 = \mathcal{Q}_1 = \{1, 2, 3\}$, $\mathcal{U}_2 = \mathcal{Q}_2 = \{4, 5\}$, $\mathcal{U}_3 = \mathcal{Q}_3 \cup \mathcal{Q}_4 = \{6, 7\}$. According to the definitions, we also have $(\beta_1, \psi_1) = (3, 1), (\beta_2, \psi_2) = (2, 1), (\beta_3, \psi_3) = (1, 2)$. Using the definition of unique groups, we can represent a partition vector $\mathbf{q} = (\beta_1, \cdots, \beta_1, \beta_2, \cdots, \beta_2, \cdots, \beta_m, \cdots, \beta_m, 0, \cdots, 0)$ (each $\beta_i, i \in [m]$ has ψ_i terms) by a more compact form $\mathbf{q} = (\boldsymbol{\beta}_1^{(\psi_1)}, \boldsymbol{\beta}_2^{(\psi_2)}, \cdots, \boldsymbol{\beta}_m^{(\psi_m)}, \mathbf{0})$. Moreover, we call a node grouping an *equal grouping* if all the groups contain the same number of nodes, *i.e.*, $q_1 = q_2 = \cdots = q_m = \frac{K}{m}$. Otherwise, it is called an *unequal grouping*. Clearly, the example presented in Section II-C uses an unequal grouping.

2) Packet Type: A *packet type* refers to a partition of $t := \frac{KM}{N} \in \mathbb{Z}^+$ nodes and is represented by a partition vector $\mathbf{v} := (v_1, v_2, \cdots, v_t)$ satisfying $\sum_{i=1}^{t} v_i = t$ and $v_1 \geq v_2 \geq \cdots \geq v_t \geq 0$. Different partitions of t correspond to different packet types. A *raw packet* $W_{n,\mathcal{T}}$, for some $\mathcal{T} \subset \mathcal{U}$, $|\mathcal{T}| = t$ refers to a sub-file that is cached exclusively by a set of nodes in $\mathcal{T}$. Each packet type may contain multiple raw packets. Since not all packets types can appear under a given node grouping, we can exclude some invalid packet types, meaning that these packet types will not be used in the PTB design. This is called *raw packet saving gain*. In the delivery phase, raw packets (i.e, sub-files) might be further split into multiple *packets*, *i.e.*, $W_{n,\mathcal{T}} = \{W_{n,\mathcal{T}}^{(i)}\}_{i \in [\alpha(\mathbf{v})]}$ where $\mathbf{v}$ is the packet type and $\alpha(\mathbf{v})$ is called *further splitting ratio*. Raw packets of the same type must have the same further splitting ratio. Note that all raw packets have the same further splitting ratio $\alpha(\mathbf{v}) = t$, for any packet type $\mathbf{v}$ in the JCM scheme.

3) Multicasting Group Type: A *multicasting group* is a set of $t + 1$ nodes among which each node broadcasts some packets needed by the remaining t nodes. A *multicasting group type* refers to a specific partition of $t + 1$ nodes which is represented by a partition vector $\mathbf{s} := (s_1, s_2, \cdots, s_{t+1})$ satisfying $\sum_{i=1}^{t+1} s_i = t + 1$ and $s_1 \geq s_2 \geq \cdots s_{t+1} \geq 0$. Different partitions of $t + 1$ nodes correspond to different multicasting group types. A *unique group* in $\mathbf{s}$, denoted by $\bar{\mathcal{U}}_i (i \in \bar{N}_d)$, refers to the union of parts of $\mathbf{s}$ that contain the same number of nodes, where $\bar{N}_d$ denotes the number of distinct parts in $\mathbf{s}$. For a specific multicasting group $\mathcal{S}$ of type $\mathbf{s}$, the set of unique groups of $\mathbf{s}$ are represented by $\{\bar{\mathcal{U}}_i\}_{i \in [\bar{N}_d]}$ and we have $\mathcal{S} = \bigcup_{i \in [\bar{N}_d]} \bar{\mathcal{U}}_i$. We define a *involved packet type set*, denoted by $\boldsymbol{\rho}$, corresponding to a specific multicasting group type $\mathbf{s}$, as the set of packet types that can appear in the transmission process within multicasting groups of type $\mathbf{s}$.

4) Further Splitting Ratio (FSR): The *further splitting ratio* of a packet type $\mathbf{v}$, denoted by $\alpha(\mathbf{v})$, implies that all the type-$\mathbf{v}$ raw packets need to be split into $\alpha(\mathbf{v}) \in \mathbb{Z}^+$ packets in the PTB design. For a multicasiting group $\mathcal{S}$ of type $\mathbf{s}$ containing $\bar{N}_d$ different unique groups, a set of nodes $\mathcal{T}_\mathrm{x} \subseteq \mathcal{S}$ is selected to serve as transmitters for the coded multicasting transmissions in $\mathcal{S}$. We can select $\mathcal{T}_\mathrm{x}$ in such a way that it can be expressed as a union of $|\mathcal{D}_\mathrm{T}|$ different unique groups where, in the multicasting group $\mathcal{S}$ of type $\mathbf{s}$, $\mathcal{D}_\mathrm{T} \subseteq [\bar{N}_d]$ is defined as the set of the indices of the unique groups *i.e.*, $\mathcal{T}_\mathrm{x} = \bigcup_{i\in\mathcal{D}_\mathrm{T}} \bar{\mathcal{U}}_i$. Denote $g_i := |\bar{\mathcal{U}}_i|$ as the number of nodes contained in the unique group $\bar{\mathcal{U}}_i$, then we have $|\mathcal{T}_\mathrm{x}| = \sum_{i\in\mathcal{D}_\mathrm{T}} g_i$. The involved packet type set associated with $\mathcal{S}$ contains $\bar{N}_d$ different packet types, *i.e.*, $\boldsymbol{\rho} = \{\mathbf{v}_i\}_{i\in[\bar{N}_d]}$. The packet type $\mathbf{v}_i (i \in [\bar{N}_d])$ is composed of a set of raw packets $\{W_{d_{k_i}, \mathcal{S}\setminus\{k_i\}}\}_{k_i\in\bar{\mathcal{U}}_i}$. Under such a selection of transmitters, the further splitting ratios for the involved packet types are

$$\alpha(\mathbf{v}_i) = \begin{cases} \sum\limits_{j\in\mathcal{D}_\mathrm{T}} g_j - 1 & \text{if } i \in \mathcal{D}_\mathrm{T} \\ \sum\limits_{j\in\mathcal{D}_\mathrm{T}} g_j & \text{if } i \notin \mathcal{D}_\mathrm{T} \end{cases} \tag{4}$$

which means that each type-$\mathbf{v}_i$ raw packet needs to be split into $\alpha(\mathbf{v}_i)$ packets when considering the coded multicasting transmission within the multicasting group $\mathcal{S}$ of type $\mathbf{s}$ in the delivery phase. Since one packet type can possibly be contained in multiple involved packet type sets and the above further splitting ratios are derived when only one multicasting group type is considered, we refer to this further splitting ratio as *local further splitting ratio*.

5) Further Splitting Ratio Table (FSRT): Given a node grouping $\mathbf{q}$, denote V, S as the total number of different valid packets types and multicasting types respectively. A *further splitting ratio table* is a matrix $\boldsymbol{\Lambda} = [\alpha_{ij}]_{S\times V}$ which specifies the local further splitting ratios of packet types derived from all the S multicasting types. More specifically, the i-th ($i \in [S]$) row of the FSRT, which is referred to as the *local further splitting ratio vector* $\boldsymbol{\alpha}_i$, consists further splitting ratios $\alpha(\mathbf{v}_j)$ for all packet types $\mathbf{v}_j \in \boldsymbol{\rho}_i$ ($\boldsymbol{\rho}_i$ is the involved packet set corresponding to multicasting type $\mathbf{s}_i$) and is specified by Eq. (4). All the other entries $\{\alpha(\mathbf{v}_j) : \mathbf{v}_j \notin \boldsymbol{\rho}_i\}$ are left empty. Note that a further splitting ratio of $\alpha = 0$ is not the same as an empty entry. To determine the overall further splitting ratio for all the V types of packets, we need to derive the *Least Common Multiple (LCM) vector* $\boldsymbol{\alpha}^{\mathrm{LCM}}$ (defined below) of the S different local further splitting ratio vectors.

Definition 1: (**Least Common Multiple (LCM) Vector**) For a set of n vectors $\mathcal{A} = \{\boldsymbol{a}_i\}_{i\in[n]}$ in which $|\boldsymbol{a}_i| = V$ and $\boldsymbol{a}_i$ may contain 'empty' entries, the LCM vector of $\mathcal{A}$, denoted by $\boldsymbol{a}^{\mathrm{LCM}} := \mathrm{LCM}(\mathcal{A})$, is defined as: $\exists\, z_1, z_2, \cdots, z_n \in \mathbb{Z}^+$ such that: (1) $z_1\boldsymbol{a}_1 = z_2\boldsymbol{a}_2 = \cdots = z_n\boldsymbol{a}_n$ and (2) $\boldsymbol{a}^{\mathrm{LCM}} = \arg\min_{z_1\sim z_n} ||\boldsymbol{a}||_2^2 = \arg\min_{z_1\sim z_n} ||\mathit{combine}\left(\{z_i\boldsymbol{a}_i\}_{i\in[n]}\right)||_2^2$ in which the *combine* operation means that the j-th entry of $\boldsymbol{a}^{\mathrm{LCM}}$ takes the value of the non-zero and non-empty value among the j-th entries all the n vectors $\{z_i\boldsymbol{a}_i\}_{i\in[n]}$. We assume that 1) the product of any integer and an empty entry is still an empty entry; 2) entry '0' is equal to any other entries, including non-zero entries and empty entries; 3) empty entry is equal to any other zero/non-zero entries. $\triangle$

Note that the LCM vector may not always exist. If it exists, it must be unique. In a specific PTB design, the overall splitting ratio vector, denoted by $\boldsymbol{\alpha}^{\mathrm{LCM}}$, is obtained via deriving the LCM vector of the set of local splitting ratio vectors $\{\boldsymbol{\alpha}_i\}_{i\in[S]}$, *i.e.*, $\boldsymbol{\alpha}^{\mathrm{LCM}} := \mathrm{LCM}\left(\{\boldsymbol{\alpha}_i\}_{i\in[S]}\right)$.

6) Memory Constraint Table (MCT): Given a node grouping $\mathbf{q}$ containing N_d different unique groups, a *memory constraint table* is a matrix $\boldsymbol{\Omega} = [\omega_{ij}]_{N_d\times V}$ with $\omega_{ij} := F_i(\mathbf{v}_j)$ where $F_i(\mathbf{v}_j)$ denotes the number of raw packets of type $\mathbf{v}_j$ cached by a node in the i-th unique group. Denote $\mathbf{F}_i := [F_i(\mathbf{v}_1), F_i(\mathbf{v}_2), \cdots, F_i(\mathbf{v}_V)]$ as the i-th row of $\boldsymbol{\Omega}$. Also denote the *raw packet number vector* as $\mathbf{F} := [F(\mathbf{v}_1), F(\mathbf{v}_2), \cdots, F(\mathbf{v}_V)]$ where $F(\mathbf{v}_j)$ represents the number of raw packets of type $\mathbf{v}_j (j \in [V])$ in a PTB design. Furthermore, $\forall i \in [N_\mathrm{d} - 1]$, we define the *node cache difference vector* as $\Delta\mathbf{F}_i := (f_{i1}, f_{i2}, \cdots, f_{iV}) = \mathbf{F}_{i+1} - \mathbf{F}_i$ in which $f_{ij} = F_{i+1}(\mathbf{v}_j) - F_i(\mathbf{v}_j), \forall j \in [V]$ is the difference of the number of type-$\mathbf{v}_j$ raw packets cached by nodes in the $(i+1)$-th and i-th unique group $\mathcal{U}_{i+1}$ and $\mathcal{U}_i$. Let all the packets have the same size, the memory constraint can be represented as $\boldsymbol{\alpha}^{\mathrm{LCM}}\Delta\mathbf{F}_i^{\mathrm{T}} = \boldsymbol{\alpha}^{\mathrm{LCM}}(\mathbf{F}_{i+1} - \mathbf{F}_i)^{\mathrm{T}} = 0, \forall i \in [1 : N_d - 1]$, *i.e.*, $\boldsymbol{\alpha}^{\mathrm{LCM}}\mathbf{F}_1^{\mathrm{T}} = \boldsymbol{\alpha}^{\mathrm{LCM}}\mathbf{F}_2^{\mathrm{T}} \cdots = \boldsymbol{\alpha}^{\mathrm{LCM}}\mathbf{F}_{N_d}^{\mathrm{T}}$, implying that nodes in all the N_d unique groups have cached the same number of packets. Since all the nodes have identical cache memory size, caching the same number of packets of equal length satisfies the memory constraint. The exact length of the packets can be determined by the fact that each node has a cache memory size of M files.

7) PTB Design as An Integer Optimization: With all the above definitions, under the condition of equal-length subpacketizations (all packets have identical length), the integer optimization problem that determines the optimal LCM vector which results in the minimum F is given by Eq. (1a) to (1c), where Φ represents the set of all possible LCM vectors derived from the S local further splitting vectors based on the set of all possible node grouping $\mathbf{q}$ and the set of all possible selections of transmitters within each multicasting group type under each $\mathbf{q}$. Although each feasible solution of the above optimization problem corresponds to a valid PTB design which may or may not yield a lower subpacketization level than the JCM scheme, in this paper, we present several PTB designs with order or constant reduction on the subpacketization levels compared to the JCM scheme, implying that the JCM scheme is far from optimal in terms of subpacketization. The extension of the optimization problem (1a) to (1c) to the case of unequal-length subpacketizations (different packets may have distinct packet length) can be found in [17].

References

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

[4] K. Shanmugam, A. M. Tulino, and A. G. Dimakis, "Coded caching with linear subpacketization is possible using ruzsa-szeméredi graphs," in *2017 IEEE International Symposium on Information Theory (ISIT)*, June 2017, pp. 1237–1241.

[5] C. Shangguan, Y. Zhang, and G. Ge, "Centralized coded caching schemes: A hypergraph theoretical approach," *IEEE Transactions on Information Theory*, vol. 64, no. 8, pp. 5755–5766, 2018.

[6] 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.

[7] S. Jin, Y. Cui, H. Liu, and G. Caire, "A new order-optimal decentralized coded caching scheme with good performance in the finite file size regime," *IEEE Transactions on Communications*, pp. 1–1, 2019.

[8] M. Ji, G. Caire, and A. F. Molisch, "Fundamental limits of caching in wireless d2d networks," *IEEE Transactions on Information Theory*, vol. 62, no. 2, pp. 849–869, Feb 2016.

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

[10] R. Karasik, O. Simeone, and S. Shamai, "Information- theoretic analysis of d2d-aided pipelined content delivery in fog-ran," in *2018 15th International Symposium on Wireless Communication Systems (ISWCS)*, Aug 2018, pp. 1–5.

[11] Ç. Yapar, K. Wan, R. F. Schaefer, and G. Caire, "On the optimality of d2d coded caching with uncoded cache placement and one-shot delivery," *IEEE Transactions on Communications*, vol. 67, no. 12, pp. 8179–8192, Dec 2019.

[12] Kai Wan, Hua Sun, Mingyue Ji, Daniela Tuninetti, and Giuseppe Caire, "Device-to-device private caching with trusted server," *arXiv preprint arXiv:1909.12748*, 2019.

[13] N. Woolsey, R. Chen, and M. Ji, "Device-to-device caching networks with subquadratic subpacketizations," in *GLOBECOM 2017*, Dec 2017, pp. 1–6.

[14] J. Wang, M. Cheng, Q. Yan, and X. Tang, "On the placement delivery array design for coded caching scheme in d2d networks," *arXiv:1712.06212*, 2017.

[15] N. Woolsey, R.-R. Chen, and M. Ji, "Towards practical file packetizations in wireless device-to-device caching networks," *arXiv preprint arXiv:1712.07221*, 2017.

[16] N. Woolsey, R. Chen, and M. Ji, "Coded caching in wireless device-to-device networks using a hypercube approach," in *2018 ICC Workshops*, May 2018, pp. 1–6.

[17] X. Zhang and M. Ji, "A new design framework on device-to-device coded caching with optimal rate and less subpacketizations," https://www.dropbox.com/s/ibldcixb4gndaga/LongVersion.pdf?dl=0.