entropy

MDPI

Article

Coded Caching for Broadcast Networks with User Cooperation [†]

Zhenhao Huang [1], Jiahui Chen [1], Xiaowen You [1], Shuai Ma [2] and Youlong Wu [1,*]

[1] School of Information Science and Technology, ShanghaiTech University, No. 393 Huaxia Middle Road, Pudong, Shanghai 201210, China; huangzhh@shanghaitech.edu.cn (Z.H.); chenjh1@shanghaitech.edu.cn (J.C.); youxw@shanghaitech.edu.cn (X.Y.)

[2] Information Processing and Communications Laboratory, Telecom Paris, IP Paris, 91120 Palaiseau, France; ma@telecom-paris.fr

[*] Correspondence: wuyl1@shanghaitech.edu.cn

[†] This paper was in part presented at the IEEE Information Theory Workshop, Visby, Gotland, Sweden, 2019 and at the IEEE 57th Annual Allerton Conference on Communication, Control, and Computing (Allerton), Monticello, IL, USA, 24–27 September 2019.

Abstract: Caching technique is a promising approach to reduce the heavy traffic load and improve user latency experience for the Internet of Things (IoT). In this paper, by exploiting edge cache resources and communication opportunities in device-to-device (D2D) networks and broadcast networks, two novel coded caching schemes are proposed that greatly reduce transmission latency for the centralized and decentralized caching settings, respectively. In addition to the multicast gain, both schemes obtain an additional *cooperation gain* offered by user cooperation and an additional *parallel gain* offered by the parallel transmission among the server and users. With a newly established lower bound on the transmission delay, we prove that the centralized coded caching scheme is *order-optimal*, i.e., achieving a constant multiplicative gap within the minimum transmission delay. The decentralized coded caching scheme is also order-optimal if each user's cache size is larger than a threshold which approaches zero as the total number of users tends to infinity. Moreover, theoretical analysis shows that to reduce the transmission delay, the number of users sending signals simultaneously should be appropriately chosen according to the user's cache size, and always letting more users send information in parallel could cause high transmission delay.

Keywords: coded cache; cooperation; device-to-device; transmission delay

check for updates

Citation: Huang, Z.; Chen, J.; You, X.; Ma, S.; Wu, Y. Coded Caching for Broadcast Networks with User Cooperation. *Entropy* **2022**, *24*, 1034. https://doi.org/10.3390/e24081034

Academic Editors: H. Vincent Poor, Holger Boche, Rafael F. Schaefer and Onur Günlü

Received: 22 June 2022
Accepted: 25 July 2022
Published: 27 July 2022

1. Introduction

With the rapid development of Internet of Things (IoT) technologies, IoT data traffic, such as live streaming and on-demand video streaming, has grown dramatically over the past few years. To reduce the traffic load and improve the user latency experience, the caching technique has been viewed as a promising approach that shifts the network traffic to low congestion periods. In the seminal paper [1], Maddah-Ali and Niesen proposed a coded caching scheme based on centralized file placement and coded multicast delivery that achieves a significantly larger global multicast gain compared to the conventional uncoded caching scheme.

The coded caching scheme has attracted wide and significant interest. The coded caching scheme was extended to a setup with decentralized file placement, where no coordination is required for the file placement [2]. For the cache-aided broadcast network, ref. [3] showed that the rate–memory tradeoff of the above caching system is within a factor of 2.00884. For the setting with uncoded file placement where each user stores uncoded content from the library, refs. [4,5] proved that Maddah-Ali and Niesen's scheme is optimal. In [6], both the placement and delivery phases of coded caching are depicted using a placement delivery array (PDA), and an upper bound for all possible regular PDAs was established. In [7], the authors studied a cached-aided network with heterogeneous setting where the user cache memories are unequal. More asymmetric network settings have been discussed, such as coded caching with heterogeneous user profiles [8], with distinct sizes of files [9], with asymmetric cache sizes [10–12] and with distinct link qualities [13]. The settings with varying file popularities have been discussed in [14–16]. Coded caching that jointly considers various heterogeneous aspects was studied in [17]. Other works on coded caching include, e.g., cache-aided noiseless multi-server network [18], cache-aided wireless/noisy broadcast networks [19–22], cache-aided relay networks [23–25], cache-aided interference management [26,27], coded caching with random demands [28], caching in combination networks [29], coded caching under secrecy constraints [30], coded caching with reduced subpacketization [31,32], the coded caching problem where each user requests multiple files [33], and a cache-aided broadcast network for correlated content [34], etc.

A different line of work is to study the cached-aided networks without the presence of a server, e.g., the device-to-device (D2D) cache-aided network. In [35], the authors investigated coded caching for wireless D2D network [35], where users locate in a fixed mesh topology wireless D2D network. A D2D system with selfish users who do not participate in delivering the missing subfiles to all users was studied in [36]. Wang et al. applied the PDA to characterize cache-aided D2D wireless networks in [37]. In [38], the authors studied the spatial D2D networks in which the user locations are modeled by a Poisson point process. For heterogeneous cache-aided D2D networks where users are equipped with cache memories of distinct sizes, ref. [39] minimized the delivery load by optimizing over the partition during the placement phase and the size and structure of D2D during the delivery phase. A highly dense wireless network with device mobility was investigated in [40].

In fact, combining the cache-aided broadcast network with the cache-aided D2D network can potentially reduce the transmission latency. This hybrid network is common in many practical distributed systems such as cloud network [41], where a central cloud server broadcasts messages to multiple users through the cellular network, and meanwhile users communicate with each other through a fiber local area network (LAN). A potential scenario is that users in a moderately dense area, such as a university, want to download files, such as movies, from a data library, such as a video service provider. It should be noted that the user demands are highly redundant, and the files need not only be stored by a central server but also partially cached by other users. Someone can attain the desired content through both communicating with the central server and other users such that the communication and storage resources can be used efficiently. Unfortunately, there is very little research investigating the coded caching problem for this hybrid network. In this paper, we consider such hybrid cache-aided network where a server consisting of $N \in \mathbb{Z}^+$ files connects with $K \in \mathbb{Z}^+$ users through a broadcast network, and meanwhile the users can exchange information via a D2D network. Unlike the settings of [35,38], in which each user can only communicate with its neighboring users via spatial multiplexing, we consider the D2D network as either an error-free shared link or a flexible routing network [18]. In particular, for the case of the shared link, all users exchange information via a shared link. In the flexible routing network, there exists a routing strategy adaptively partitioning all users into multiple groups, in each of which one user sends data packets error-free to the remaining users in the corresponding group. Let $\alpha \in \mathbb{Z}$ be the number of groups who send signals at the same time, then the following fundamental questions arise for this hybrid cache-aided network:

- *How does α affect the system performance?*
- *What is the (approximately) optimal value of α to minimize the transmission latency?*
- *How can communication loads be allocated between the server and users to achieve the minimum transmission latency?*

In this paper, we try to address these questions, and our main contributions are summarized as follows:

- We propose novel coded caching schemes for this hybrid network under centralized and decentralized data placement. Both schemes efficiently exploit communication opportunities in D2D and broadcast networks, and appropriately allocate communication loads between the server and users. In addition to multicast gain, our schemes achieve much smaller transmission latency than both that of Maddah-Ali and Niesen's scheme for a broadcast network [1,2] and the D2D coded caching scheme [35]. We characterize a *cooperation gain* and a *parallel gain* achieved by our schemes, where the cooperation gain is obtained through cooperation among users in the D2D network, and the parallel gain is obtained through the parallel transmission between the server and users.
- We prove that the centralized scheme is order-optimal, i.e., achieving the optimal transmission delay within a constant multiplicative gap in all regimes. Moreover, the decentralized scheme is also optimal when the cache size of each user M is larger than the threshold $N(1-\sqrt[K-1]{1/(K+1)})$ that is approaching zero as $K \to \infty$.
- For the centralized data placement case, theoretical analysis shows that α should decrease with the increase of the user caching size. In particular, when each user's caching size is sufficiently large, only one user should be allowed to send information, indicating that the D2D network can be just a simple shared link connecting all users. For the decentralized data placement case, α should be dynamically changing according to the sizes of subfiles created in the placement phase. In other words, always letting more users parallelly send information can cause a high transmission delay.

Please note that the decentralized scenario is much more complicated than the centralized scenario, since each subfile can be stored by $s = 1, 2, \ldots, K$ users, leading to a dynamic file-splitting and communication strategy in the D2D network. Our schemes, in particular the decentralized coded caching scheme, differ greatly with the D2D coded caching scheme in [35]. Specifically, ref. [35] considered a fixed network topology where each user connects with a fixed set of users, and the total user cache sizes must be large enough to store all files in the library. However, in our schemes, the user group partition is dynamically changing, and each user can communicate with any set of users via network routing. Moreover, our model has the server share communication loads with the users, resulting in an allocation problem on communication loads between the broadcast network and D2D network. Finally, our schemes achieve a tradeoff between the cooperation gain, parallel gain and multicast gain, while the schemes in [1,2,35] only achieve the multicast gain.

The remainder of this paper is as follows. Section 2 presents the system model, and defines the main problem studied in this paper. We summarize the obtained main results in Section 3. Following that is a detailed description of the centralized coded caching scheme with user cooperation in Section 4. Section 5 extends the techniques we developed for the centralized caching problem to the setting of decentralized random caching. Section 6 concludes this paper.

2. System Model and Problem Definition

Consider a cache-aided network consisting of a single server and K users as depicted in Figure 1. The server has a library of N independent files $W_1, \ldots, W_N$. Each file W_n, $n = 1, \ldots, N$, is uniformly distributed over

$$[2^F] \triangleq \{1, 2, \ldots, 2^F\},$$

for some positive integer F. The server connects with K users through a noisy-free shared link but rate-limited to a network speed of C_1 bits per second (bits/s). Each user $k \in [K]$ is equipped with a cache memory of size MF bits, for some $M \in [0, N]$, and can communicate with each other via a D2D network.

We mainly focus on two types of D2D networks: a shared link as in [1,2] and a flexible routing network introduced in [18]. In the case of a shared link, all users connect with each other through a shared error-free link but rate-limited to C_2 bits/s. In the flexible routing network, K users can arbitrarily form multiple groups via network routing, in each of which at most one user can send error-free data packets at a network speed C_2 bits/s to the remaining users within the group. To unify these two types of D2D networks, we introduce an integer $\alpha_{\max} \in \{1, \lfloor \frac{K}{2} \rfloor\}$, which denotes the maximum number of groups allowed to send data parallelly in the D2D network. For example, when $\alpha_{\max} = 1$, the D2D network degenerates into a shared link, and when $\alpha_{\max} = \lfloor \frac{K}{2} \rfloor$, it turns to be the flexible network.

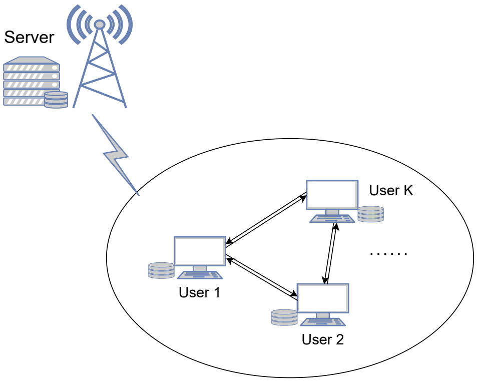

Figure 1. Caching system considered in this paper. A server connects with K cache-enabled users and the users can cooperate through a flexible network.

The system works in two phases: a placement phase and a delivery phase. In the placement phase, all users will access the entire library $W_1, \ldots, W_N$ and fill the content to their caching memories. More specifically, each user k, for $k \in [K]$, maps $W_1, \ldots, W_N$ to its cache content:

$$Z_k \triangleq \phi_k(W_1, \ldots, W_N), \tag{1}$$

for some caching function

$$\phi_k : [2^F]^N \to [\lfloor 2^{MF} \rfloor]. \tag{2}$$

In the delivery phase, each user requests one of the N files from the library. We denote the demand of user k as $d_k \in [N]$, and its desired file as W_{d_k}. Let $\mathbf{d} \triangleq (d_1, \ldots, d_K)$ denotes the request vector. In this paper, we investigate the worst request case where each user makes a unique request.

Once the request vector $\mathbf{d}$ is informed to the server and all users, the server produces the symbol

$$X \triangleq f_{\mathbf{d}}(W_1, \ldots, W_N), \tag{3}$$

and broadcasts it to all users through the broadcast network. Meanwhile, user $k \in \{1, \ldots, K\}$ produces the symbol (Each user k can produce X_k as a function of Z_k and the received signals sent by the server, but because all users can access to the server's signal due to the fact that the server broadcasts its signals to the network, it is equivalent to generating X_k as a function Z_k).

$$X_k \triangleq f_{k,\mathbf{d}}(Z_k), \tag{4}$$

and sends it to a set of intended users $\mathcal{D}_k \subseteq [K]$ through the D2D network. Here, $\mathcal{D}_k$ represents the set of destination users served by node k, $f_{\mathbf{d}}$ and $f_{k,\mathbf{d}}$ are some encoding functions

$$f_{\mathbf{d}} : [2^F]^N \to [\lfloor 2^{R_1 F} \rfloor], \; f_{k,\mathbf{d}} : [\lfloor 2^{MF} \rfloor] \to [\lfloor 2^{R_2 F} \rfloor], \tag{5}$$

where R_1 and R_2 denote the *transmission rate* sent by the server in the broadcast network and by each user in the D2D network, respectively. Here we focus on the symmetric case where all users have the same transmission rate. Due to the constraint of $\alpha_{\max}$, at most

$\alpha_{\max}$ users can send signals parallelly in each channel use. The set of $\alpha_{\max}$ users who send signals in parallel could be adaptively changed in the delivery phase.

At the end of the delivery phase, due to the error-free transmission in the broadcast and D2D networks, user k observes symbols sent to them, i.e., $(X_j : j \in [K], k \in \mathcal{D}_j)$, and decodes its desired message as $\hat{W}_{d_k} = \psi_{k,\mathbf{d}}(X, (X_j : j \in [K], k \in \mathcal{D}_j), Z_k)$, where $\psi_{k,\mathbf{d}}$ is a decoding function.

We define the worst-case probability of error as

$$P_e \triangleq \max_{\mathbf{d}\in\mathcal{F}^n} \max_{k\in[K]} \Pr\big(\hat{W}_{d_k} \neq W_{d_k}\big). \tag{6}$$

A coded caching scheme (M, R_1, R_2) consists of caching functions $\{\phi_k\}$, encoding functions $\{f_{\mathbf{d}}, f_{k,\mathbf{d}}\}$ and decoding functions $\{\psi_{k,\mathbf{d}}\}$. We say that the rate region (M, R_1, R_2) is *achievable* if for every $\epsilon > 0$ and every large enough file size F, there exists a coded caching scheme such that P_e is less than ϵ.

Since the server and the users send signals in parallel, the total transmission delay, denoted by T, can be defined as

$$T \triangleq \max\{\frac{R_1 F}{C_1}, \frac{R_2 F}{C_2}\}. \tag{7}$$

The *optimal* transmission delay is $T^* \triangleq \inf\{T : T \text{ is achievable}\}$. For simplicity, we assume that $C_1 = C_2 = F$, and then from (7) we have

$$T = \max\{R_1, R_2\}. \tag{8}$$

When $C_1 \neq C_2$, e.g., $C_1 : C_2 = 1/k$, one small adjustment allowing our scheme to continue to work is multiplying λ by $1/(k(1-\lambda)+\lambda)$, where λ is a devisable parameter introduced later.

Our goal is to design a coded caching scheme to minimize the transmission delay. Finally, in this paper we assume $K \leq N$ and $M \leq N$. Extending the results to other scenarios is straightforward, as mentioned in [1].

3. Main Results

We first establish a general lower bound on the transmission delay for the system model described in Section 2, then present two upper bounds of the optimal transmission delay achieved by our centralized and decentralized coded caching schemes, respectively. Finally, we present the optimality results of these two schemes.

Theorem 1 (Lower Bound)**.** *For memory size $0 \leq M \leq N$, the optimal transmission delay is lower bounded by*

$$T^* \geq \max\left\{\frac{1}{2}\left(1 - \frac{M}{N}\right), \max_{s\in[K]}\left(s - \frac{KM}{\lfloor N/s \rfloor}\right), \max_{s\in[K]}\left(s - \frac{sM}{\lfloor N/s \rfloor}\right)\frac{1}{1+\alpha_{\max}}\right\}. \tag{9}$$

Proof. See the proof in Appendix A. □

3.1. Centralized Coded Caching

In the following theorem, we present an upper bound on the transmission delay for the centralized caching setup.

Theorem 2 (Upper Bound for the Centralized Scenario). *Let $t \triangleq KM/N \in \mathbb{Z}^+$, and $\alpha \in \mathbb{Z}^+$. For memory size $M \in \{0, \frac{N}{K}, \frac{2N}{K}, \dots, N\}$, the optimal transmission delay T^* is upper bounded by $T^* \leq T_{\text{central}}$, where*

$$T_{\text{central}} \triangleq \min_{\alpha \leq \alpha_{\max}} K\left(1-\frac{M}{N}\right)\frac{1}{1+t+\alpha \min\{\lfloor \frac{K}{\alpha} \rfloor -1, t\}}. \tag{10}$$

For general $0 \leq M \leq N$, the lower convex envelope of these points is achievable.

Proof. See scheme in Section 4. □

The following simple example shows that the proposed upper bound can greatly reduce the transmission delay.

Example 1. *Consider a network described in Section 2 with $KM/N = K-1$. The coded caching scheme without D2D communication [1] has the server multicast an XOR message useful for all K users, achieving the transmission delay $K(1-\frac{M}{N})\frac{1}{1+t} = \frac{1}{K}$. The D2D coded caching scheme [35] achieves the transmission delay $\frac{N}{M}(1-\frac{M}{N}) = \frac{1}{K-1}$. The achievable transmission delay in Theorem 2 equals $\frac{1}{2K-1}$ by letting $\alpha = 1$, almost twice as short as the transmission delay of previous schemes if K is sufficiently large.*

From (10), we obtain that the optimal value of α, denoted by α^*, equals 1 if $t \geq K-1$ and to $\alpha_{\max}$ if $t \leq \lfloor \frac{K}{\alpha_{\max}} \rfloor -1$. When ignoring all integer constraints, we obtain $\alpha^* = \frac{K}{t+1}$. We rewrite this choice as follows:

$$\alpha^* = \begin{cases} 1, & t \geq K-1, \\ K/(t+1), & \lfloor K/\alpha_{\max} \rfloor -1 < t < K-1, \\ \alpha_{\max}, & t \leq \lfloor K/\alpha_{\max} \rfloor -1. \end{cases} \tag{11}$$

Remark 1. *From (11), we observe that when M is small such that $t \leq \lfloor K/\alpha_{\max} \rfloor -1$, we have $\alpha^* = \alpha_{\max}$. As M is increasing, α^* becomes $K/(t+1)$, smaller than $\alpha_{\max}$. When M is sufficiently large such that $M \geq (K-1)N/K$, only one user should be allowed to send information, i.e., $\alpha^* = 1$. This indicates that letting more users parallelly send information could be harmful. The main reason for this phenomenon is the existence of a tradeoff between the multicast gain,* cooperation gain *and* parallel gain, *which will be introduced below in this section.*

Comparing T_{central} with the transmission delay achieved by Maddah-Ali and Niesen's scheme for the broadcast network [1], i.e., $K(1-\frac{M}{N})\frac{1}{1+t}$, T_{central} consists of an additional factor

$$G_{\text{central,c}} \triangleq \frac{1}{1+\frac{\alpha}{1+t}\min\{\lfloor \frac{K}{\alpha} \rfloor -1, t\}}, \tag{12}$$

referred to as *centralized cooperation gain*, as it arises from user cooperation. Comparing T_{central} with the transmission delay achieved by the D2D coded caching scheme [35], i.e., $\frac{N}{M}(1-\frac{M}{N})$, T_{central} consists of an additional factor

$$G_{\text{central,p}} \triangleq \frac{1}{1+\frac{1}{t}+\frac{\alpha}{t}\min\{\lfloor \frac{K}{\alpha} \rfloor -1, t\}}, \tag{13}$$

referred to as *centralized parallel gain*, as it arises from parallel transmission among the server and users. Both gains depend on K, M/N and $\alpha_{\max}$.

Substituting the optimal α^* into (12), we have

$$G_{\text{central,c}} = \begin{cases} \dfrac{1+t}{K+t}, & t \geq K-1, \\ \dfrac{1+t}{K-\frac{K}{t+1}+t}, & \lfloor \dfrac{K}{\alpha_{\max}} \rfloor - 1 < t < K-1, \\ \dfrac{1+t}{\alpha_{\max} t + t + 1}, & t \leq \lfloor \dfrac{K}{\alpha_{\max}} \rfloor - 1. \end{cases} \tag{14}$$

When fixing $(K, N, \alpha_{\max})$, $G_{\text{central,c}}$ in general is not a monotonic function of M. More specifically, when M is small enough such that $t < \lfloor \frac{K}{\alpha_{\max}} \rfloor - 1$, the function $G_{\text{central,c}}$ is monotonically decreasing, indicating that the improvement caused by introducing D2D communication. This is mainly because relatively larger M allows users to share more common data with each other, providing more opportunities on user cooperation. However, when M grows larger such that $t \geq \lfloor \frac{K}{\alpha_{\max}} \rfloor - 1$, the local and global caching gains become dominant, and less improvement can be obtained from user cooperation, turning $G_{\text{central,c}}$ to a monotonic increasing function of M,

Similarly, substituting the optimal α^* into (13), we obtain

$$G_{\text{central,p}} = \begin{cases} \dfrac{t}{K+t}, & t \geq K-1, \\ \dfrac{t}{\frac{t \cdot K}{t+1} + t + 1}, & \lfloor \dfrac{K}{\alpha_{\max}} \rfloor - 1 < t < K-1, \\ \dfrac{t}{\alpha_{\max} t + t + 1}, & t \leq \lfloor \dfrac{K}{\alpha_{\max}} \rfloor - 1. \end{cases} \tag{15}$$

Equation (15) shows that $G_{\text{central,p}}$ is monotonically increasing with t, mainly due to the fact that when M increases, more content can be sent through the D2D network without the help of the central server, decreasing the improvement from parallel transmission between the server and users.

The centralized cooperation gain (12) and parallel gain (13) are plotted in Figure 2 when $N = 40$, $K = 20$ and $\alpha_{\max} = 5$.

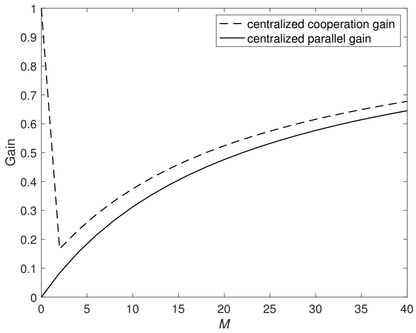

Figure 2. Centralized cooperation gain and parallel gain when $N = 40$, $K = 20$ and $\alpha_{\max} = 5$.

Remark 2. *Larger α could lead to better parallel and cooperation gain (more uses can concurrently multicast signals to other users), but will result in worse multicast gain (signals are multicast to fewer users in each group). The choice of α in (11) is in fact a tradeoff between the multicast gain, parallel gain and cooperation gain.*

The proposed scheme achieving the upper bound in Theorem 2 is order-optimal.

Theorem 3. *For memory size $0 \leq M \leq N$,*

$$\frac{T_{\text{central}}}{T^*} \leq 31. \tag{16}$$

Proof. See the proof in Appendix B. □

The exact gap of T_{central}/T^* could be much smaller. One could apply the method proposed in [3] to obtain a tighter lower bound and shrink the gap. In this paper, we only prove the order optimality of the proposed scheme, and leave the work of finding a smaller gap as the future work.

Figure 3 plots the lower bound (9) and upper bounds achieved by various schemes, including the proposed scheme, the scheme *Maddah-Ali 2014* in [1] which considers the broadcast network without D2D communication, and the scheme *Ji 2016* in [35], which considers the D2D network without server. It is obvious that our scheme outperforms the previous schemes and approaches closely to the lower bound.

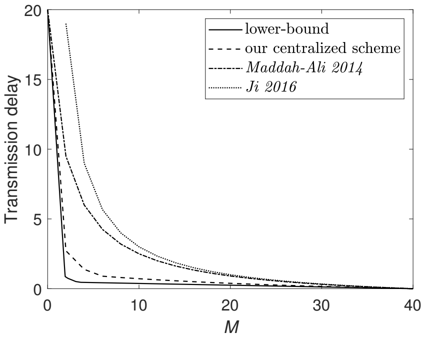

Figure 3. Transmission delay when $N = 40$, $K = 20$ and $\alpha_{\max} = 5$. The upper bounds are achieved under the centralized caching scenario.

3.2. Decentralized Coded Caching

We exploit the multicast gain from coded caching, D2D communication, and parallel transmission between the server and users, leading to the following upper bound.

Theorem 4 (Upper Bound for the Decentralized Scenario)**.** *Define $p \triangleq M/N$. For memory size $0 \le M \le N$, the optimal transmission delay T^* is upper bounded by*

$$T^* \le T_{\text{decentral}} \triangleq \max\left\{ R_\emptyset, \frac{R_{\mathrm{s}} R_{\mathrm{u}}}{R_{\mathrm{s}} + R_{\mathrm{u}} - R_\emptyset} \right\}, \tag{17}$$

where

$$R_\emptyset \triangleq K(1-p)^K, \tag{18}$$

$$R_{\mathrm{s}} \triangleq \frac{1-p}{p}\left(1-(1-p)^K\right), \tag{19}$$

$$R_{\mathrm{u}} \triangleq \frac{1}{\alpha_{\max}} \sum_{s=2}^{\lceil \frac{K}{\alpha_{\max}} \rceil - 1} \left(\frac{s\binom{K}{s}}{s-1} p^{s-1}(1-p)^{K-s+1} \right) + \sum_{s=\lceil \frac{K}{\alpha_{\max}} \rceil}^{K} \left(\frac{K\binom{K-1}{s-1}}{f(K,s)} p^{s-1}(1-p)^{K-s+1} \right), \tag{20}$$

with

$$f(K,s) \triangleq \begin{cases} \lfloor \frac{K}{s} \rfloor (s-1), & (K \bmod s) < 2, \\ K - 1 - \lfloor K/s \rfloor, & (K \bmod s) \ge 2. \end{cases} \tag{21}$$

Proof. Here, $R_\emptyset$ represents the transmission rate of sending contents that are not cached by any user, R_{s} and R_{u} represent the transmission rate sent by the server via the broadcast network, and the transmission rate sent by users via the D2D network, respectively. Equation (17) balances the communication loads assigned to the server and users. See more detailed proof in Section 5. □

The key idea of the scheme achieving (17) is to partition K users into $\lceil \frac{K}{s} \rceil$ groups for each communication round $s \in [K-1]$, and let each group perform the D2D coded caching scheme [35] to exchange information. The main challenge is that that among all $\lceil \frac{K}{s} \rceil$ groups, there are $\lfloor \frac{K}{s} \rfloor$ groups of the same size s, and an *abnormal* group of size $(K \bmod s)$ if $(K \bmod s) \neq 0$, leading to an asymmetric caching setup. One may use the scheme [35] for the groups of size s, for the group of size $(K \bmod s) \geq 2$, but how to exploit the caching resource and communication capability of all groups while balancing communication loads among the two types of groups to minimize the transmission delay remains elusive and needs to be carefully designed. Moreover, this challenge poses complexities both in establishing the upper bound and in optimality proof.

Remark 3. *The upper bound in Theorem 4 is achieved by setting the number of users that exactly send signals in parallel as follows:*

$$\alpha_{\mathrm{D}} = \begin{cases} \alpha_{\max}, & \textit{case } 1, \\ \lfloor \dfrac{K}{s} \rfloor, & \textit{case } 2, \\ \lceil \dfrac{K}{s} \rceil, & \textit{case } 3. \end{cases} \tag{22}$$

If $\lceil \frac{K}{s} \rceil > \alpha_{\max}$, the number of users who send data in parallel is smaller than α_{max}, indicating that always letting more users parallelly send messages could cause higher transmission delay. For example, when $K \geq 4$, $s = K-1$ and $\alpha_{max} = \lfloor \frac{K}{2} \rfloor$, we have $\alpha_D = 1 < \alpha_{max}$.

Remark 4. *From the definitions of $T_{\mathrm{decentral}}$, R_{s}, R_{u} and $R_{\emptyset}$, it is easy to obtain that $R_{\emptyset} \leq T_{\mathrm{decentral}} \leq R_{\mathrm{s}}$,*

$$T_{\mathrm{decentral}} = \begin{cases} \dfrac{R_{\mathrm{s}} R_{\mathrm{u}}}{R_{\mathrm{s}} + R_{\mathrm{u}} - R_{\emptyset}}, & R_{\mathrm{u}} \geq R_{\emptyset}, \\ R_{\emptyset}, & R_{\mathrm{u}} < R_{\emptyset}, \end{cases} \tag{23}$$

$T_{\mathrm{decentral}}$ decreases as $\alpha_{\max}$ increases, and $T_{\mathrm{decentral}}$ increases as R_{u} increases if $R_{\mathrm{u}} \geq R_{\emptyset}$.

Due to the complex term R_{u}, $T_{\mathrm{decentral}}$ in Theorem 4 is hard to evaluate. Since $T_{\mathrm{decentral}}$ is increasing as R_{u} increases (see Remark 4), substituting the following upper bound of R_{u} into (17) provides an efficient way to evaluate $T_{\mathrm{decentral}}$.

Corollary 1. *For memory size $0 \leq p \leq 1$, the upper bound of R_{u} is given below:*

- $\alpha_{\max} = 1$ *(a shared link):*

$$R_{\mathrm{u}} \leq \frac{1-p}{p} \left[1 - \frac{5}{2} K p (1-p)^{K-1} - 4(1-p)^K + \frac{3(1-(1-p)^{K+1})}{(K+1)p} \right]; \tag{24}$$

- $\alpha_{\max} = \lfloor \frac{K}{2} \rfloor$ *(a flexible network):*

$$R_{\mathrm{u}} \leq \frac{K(1-p)}{(K-1)} \left[1 - (1-p)^{K-1} - \frac{2/p}{K-2} (1-(1-p)^K - Kp(1-p)^{K-1}) \right]. \tag{25}$$

Proof. See the proof in Appendix C. □

Recall that the transmission delay achieved by the decentralized scheme without D2D communication [2] is equal to R_{s} given in (19). We define the ratio between $T_{\mathrm{decentral}}$ and R_{s} as *decentralized cooperation gain*:

$$G_{\mathrm{decentral,c}} \triangleq \max\{ \frac{R_{\emptyset}}{R_{\mathrm{s}}}, \frac{R_{\mathrm{u}}}{R_{\mathrm{s}} + R_{\mathrm{u}} - R_{\emptyset}} \}, \tag{26}$$

with $G_{\text{decentral,c}} \in [0,1]$ because of $R_{\emptyset} \leq R_{\text{s}}$. Similar to the centralized scenario, this gain arises from the coordination between users in the D2D network. Moreover, we also compare $T_{\text{decentral}}$ with the transmission delay $(1-p)/p$, achieved by the D2D decentralized coded caching scheme [35], and define the ratio between R_{s} and $(1-p)/p$ as *decentralized parallel gain*:

$$G_{\text{decentral,p}} \triangleq G_{\text{decentral,c}} \cdot \left(1-(1-p)^{K}\right), \tag{27}$$

where $G_{\text{decentral,p}} \in [0,1]$ arises from the parallel transmission between the server and the users.

We plot the decentralized cooperation gain and parallel gain for the two types of D2D networks in Figure 4 when $N=20$ and $K=10$. It can be seen that $G_{\text{decentral,c}}$ and $G_{\text{decentral,p}}$ in general are not monotonic functions of M. Here $G_{\text{decentral,c}}$ performs in a way similar to $G_{\text{central,c}}$. When M is small, the function $G_{\text{decentral,c}}$ is monotonically decreasing from value 1 until reaching the minimum. For larger M, the function $G_{\text{decentral,c}}$ turns to monotonically increase with M. The reason for this phenomenon is that in the decentralized scenario, when M increases, the proportion of subfiles that are not cached by any user and must be sent by the server is decreasing. Thus, there are more subfiles that can be sent parallelly via D2D network as M increases. Meanwhile, the decentralized scheme in [2] offers an additional multicasting gain. Therefore, we need to balance these two gains to reduce the transmission delay.

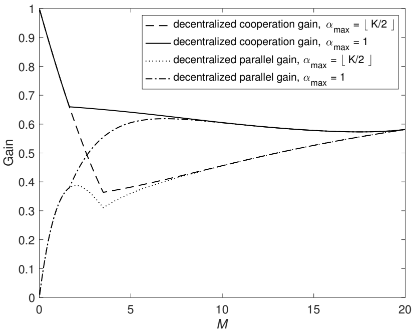

Figure 4. Decentralized cooperation gain and parallel gain when $N=20$ and $K=10$.

The function $G_{\text{decentral,p}}$ behaves differently as it monotonically increases when M is small. After reaching the maximal value, the function $G_{\text{decentral,p}}$ decreases monotonically until meeting the local minimum (The abnormal bend in parallel gain when $\alpha_{\text{max}} = \lfloor \frac{K}{2} \rfloor$ comes from a balance effect between the $G_{\text{decentral,c}}$ and $1-(1-p)^{K}$ in (27)), then $G_{\text{decentral,p}}$ turns to be a monotonic increasing function for large M. Similar to the centralized case, as M increases, the impact of parallel transmission among the server and users becomes smaller since more data can be transmitted by the users.

Theorem 5. *Define $p \triangleq M/N$ and $p_{\text{th}} \triangleq 1-\left(\frac{1}{K+1}\right)^{\frac{1}{K-1}}$, which tends to 0 as K tends to infinity. For memory size $0 \leq M \leq N$,*

- *if $\alpha_{\text{max}} = 1$ (shared link), then*
$$\frac{T_{\text{decentral}}}{T^{*}} \leq 24;$$
- *if $\alpha_{\text{max}} = \lfloor \frac{K}{2} \rfloor$, then*
$$\frac{T_{\text{decentral}}}{T^{*}} \leq \begin{cases} \max\left\{6, 2K\left(\dfrac{2K}{2K+1}\right)^{K-1}\right\}, & p < p_{\text{th}}, \\ 6, & p \geq p_{\text{th}}. \end{cases}$$

Proof. See the proof in Appendix D. □

Figure 5 plots the lower bound in (9) and upper bounds achieved by various decentralized coded caching schemes, including our scheme, the scheme *Maddah-Ali 2015* in [2] which considers the case without D2D communication, and the scheme *Ji 2016* in [35] which considers the case without server.

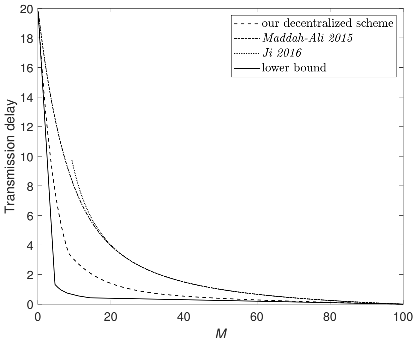

Figure 5. Transmission delay when $N = 100$, $K = 20$ and $\alpha_{\max} = 3$. The upper bounds are achieved under the decentralized random caching scenario.

4. Coding Scheme under Centralized Data Placement

In this section, we describe a novel centralized coded caching scheme for arbitrary K, N and M such that $t = KM/N$ is a positive integer. The scheme can be extended to the general case $1 \leq t \leq K$ by following the same approach as in [1].

We first use an illustrative example to show how we form D2D communication groups, split files and deliver data, and then present our generalized centralized coding caching scheme.

4.1. An Illustrative Example

Consider a network consisting of $K = 6$ users with cache size $M = 4$, and a library of $N = 6$ files. Thus, $t = KM/N = 4$. Divide all six users into two groups of equal size, and choose an integer $L_1 = 2$ that guarantees $\frac{K\binom{K-1}{t}L_1}{\min\{\alpha(\lfloor K/\alpha \rfloor - 1), t\}}$ to be an integer. (According to (11) and (29), one optimal choice could be ($\alpha = 1$, $L_1 = 4$, $\lambda = 5/9$), here we choose ($\alpha = 2$, $L_1 = 2$, $\lambda = 1/3$) for simplicity, and also in order to demonstrate that even with a suboptimal choice, our scheme still outperforms that in [1,35]). Split each file W_n, for $n = 1, \ldots, N$, into $3\binom{6}{4} = 45$ subfiles:

$$W_n = (W^l_{n,\mathcal{T}} : l \in [3], \mathcal{T} \subset [6], |\mathcal{T}| = 4).$$

We list all the requested subfiles uncached by all users as follows: for $l = 1, 2, 3$,

$$\begin{gathered}
W^l_{d_1,\{2,3,4,5\}}, W^l_{d_1,\{2,3,4,6\}}, W^l_{d_1,\{2,3,5,6\}}, W^l_{d_1,\{2,4,5,6\}}, W^l_{d_1,\{3,4,5,6\}}; \\
W^l_{d_2,\{1,3,4,5\}}, W^l_{d_2,\{1,3,4,6\}}, W^l_{d_2,\{1,3,5,6\}}, W^l_{d_2,\{1,4,5,6\}}, W^l_{d_2,\{3,4,5,6\}}; \\
W^l_{d_3,\{1,2,4,5\}}, W^l_{d_3,\{1,2,4,6\}}, W^l_{d_3,\{1,2,5,6\}}, W^l_{d_3,\{1,4,5,6\}}, W^l_{d_3,\{2,4,5,6\}}; \\
W^l_{d_4,\{1,2,3,5\}}, W^l_{d_4,\{1,2,3,6\}}, W^l_{d_4,\{1,2,5,6\}}, W^l_{d_4,\{1,3,5,6\}}, W^l_{d_4,\{2,3,5,6\}}; \\
W^l_{d_5,\{1,2,3,4\}}, W^l_{d_5,\{1,2,3,6\}}, W^l_{d_5,\{1,2,4,6\}}, W^l_{d_5,\{1,3,4,6\}}, W^l_{d_5,\{2,3,4,6\}}; \\
W^l_{d_6,\{1,2,3,4\}}, W^l_{d_6,\{1,2,3,5\}}, W^l_{d_6,\{1,2,4,5\}}, W^l_{d_6,\{1,3,4,5\}}, W^l_{d_6,\{2,3,4,5\}}.
\end{gathered}$$

The users can finish the transmission in different partitions. Table 1 shows the transmission in four different partitions over the D2D network.

Table 1. Subfiles sent by users in different partition, $l = 1, 2$.

$\{1,2,3\}$	$\{4,5,6\}$
user 2: $W^1_{d_1,\{2,3,4,5\}} \oplus W^1_{d_3,\{1,2,4,5\}}$	user 5: $W^1_{d_4,\{2,3,5,6\}} \oplus W^1_{d_6,\{2,3,4,5\}}$
user 2: $W^1_{d_1,\{2,3,4,6\}} \oplus W^1_{d_3,\{1,2,4,6\}}$	user 5: $W^1_{d_4,\{1,2,5,6\}} \oplus W^1_{d_6,\{1,2,4,5\}}$
user 1: $W^1_{d_2,\{1,3,4,6\}} \oplus W^1_{d_3,\{1,2,5,6\}}$	user 4: $W^1_{d_5,\{2,3,4,6\}} \oplus W^1_{d_6,\{1,3,4,5\}}$
user 3: $W^1_{d_1,\{2,3,5,6\}} \oplus W^1_{d_2,\{1,3,5,6\}}$	user 6: $W^1_{d_4,\{1,3,5,6\}} \oplus W^1_{d_5,\{1,3,4,6\}}$
$\{1,2,4\}$	$\{3,5,6\}$
user 2: $W^l_{d_1,\{2,4,5,6\}} \oplus W^l_{d_4,\{1,2,3,5\}}$	user 5: $W^l_{d_3,\{1,4,5,6\}} \oplus W^l_{d_6,\{1,2,3,5\}}$
$\{1,4,6\}$	$\{2,3,5\}$
user 6: $W^l_{d_1,\{3,4,5,6\}} \oplus W^l_{d_4,\{1,2,3,6\}}$	user 3: $W^l_{d_2,\{3,4,5,6\}} \oplus W^l_{d_5,\{1,2,3,4\}}$
$\{1,2,5\}$	$\{3,4,6\}$
user 1: $W^l_{d_2,\{1,4,5,6\}} \oplus W^l_{d_5,\{1,2,3,6\}}$	user 4: $W^l_{d_3,\{2,4,5,6\}} \oplus W^l_{d_6,\{1,2,3,4\}}$
$\{1,2,3\}$	$\{4,5,6\}$
user 3: $W^2_{d_1,\{2,3,4,5\}} \oplus W^2_{d_2,\{1,3,4,5\}}$	user 4: $W^2_{d_5,\{2,3,4,6\}} \oplus W^2_{d_6,\{2,3,4,5\}}$
user 3: $W^2_{d_1,\{2,3,4,6\}} \oplus W^2_{d_2,\{1,3,4,6\}}$	user 4: $W^2_{d_5,\{1,2,4,6\}} \oplus W^2_{d_6,\{1,2,4,5\}}$
user 2: $W^2_{d_1,\{2,3,5,6\}} \oplus W^2_{d_3,\{1,2,4,5\}}$	user 5: $W^2_{d_4,\{1,3,5,6\}} \oplus W^2_{d_6,\{1,3,4,5\}}$
user 1: $W^2_{d_3,\{1,2,4,6\}} \oplus W^2_{d_2,\{1,3,5,6\}}$	user 6: $W^2_{d_4,\{1,2,5,6\}} \oplus W^2_{d_5,\{1,3,4,6\}}$
user 1: $W^2_{d_3,\{1,2,5,6\}} \oplus W^1_{d_2,\{1,3,4,5\}}$	user 6: $W^1_{d_5,\{1,2,4,6\}} \oplus W^2_{d_4,\{2,3,5,6\}}$

In Table 1, all users first send XOR symbols with superscript $l = 1$. Please note that the subfiles $W^1_{d_2,\{1,3,4,5\}}$ and $W^1_{d_5,\{1,2,4,6\}}$ are not delivered at the beginning since $\frac{K\binom{K-1}{t}}{\alpha(\lfloor K/\alpha \rfloor - 1)}$ is not an integer. Similarly, for subfiles with $l = 2$, $W^2_{d_3,\{1,2,5,6\}}$ and $W^2_{d_4,\{2,3,5,6\}}$ remain to be sent to user 3 and 4. In the last transmission, user 1 delivers the XOR message $W^2_{d_3,\{1,2,5,6\}} \oplus W^1_{d_2,\{1,3,4,5\}}$ to user 2 and 3, and user 6 multicasts $W^1_{d_5,\{1,2,4,6\}} \oplus W^2_{d_4,\{2,3,5,6\}}$ to user 5 and 6. The transmission rate in the D2D network is $R_2 = \frac{1}{3}$.

For the remaining subfiles with superscript $l = 3$, the server delivers them in the same way as in [1]. Specifically, it sends symbols $\oplus_{k \in \mathcal{S}} W^3_{d_k, \mathcal{S} \setminus \{k\}}$, for all $\mathcal{S} \subseteq [K] : |\mathcal{S}| = 5$. Thus, the rate sent by the server is $R_1 = \frac{2}{15}$, and the transmission delay $T_{\text{central}} = \max\{R_1, R_2\} = \frac{1}{3}$, which is less than the delay achieved by the coded caching schemes for the broadcast network [1] and the D2D communication [35], respectively.

4.2. The Generalized Centralized Coding Caching Scheme

In the placement phase, each file is first split into $\binom{K}{t}$ subfiles of equal size. More specifically, split W_n into subfiles as follows: $W_n = (W_{n,\mathcal{T}} : \mathcal{T} \subset [K], |\mathcal{T}| = t)$. User k caches all the subfiles if $k \in \mathcal{T}$ for all $n = 1, ..., N$, occupying the cache memory of MF bits. Then split each subfile $W_{n,\mathcal{T}}$ into two mini-files as $W_{n,\mathcal{T}} = \left(W^{\text{s}}_{n,\mathcal{T}}, W^{\text{u}}_{n,\mathcal{T}}\right)$, where

$$\begin{aligned} |W^{\text{s}}_{n,\mathcal{T}}| &= \lambda \cdot |W_{n,\mathcal{T}}| = \lambda \cdot \frac{F}{\binom{K}{t}}, \\ |W^{\text{u}}_{n,\mathcal{T}}| &= (1-\lambda) \cdot |W_{n,\mathcal{T}}| = (1-\lambda) \cdot \frac{F}{\binom{K}{t}}, \end{aligned} \tag{28}$$

with

$$\lambda = \frac{1+t}{\alpha \min\{\lfloor \frac{K}{\alpha} \rfloor - 1, t\} + 1 + t}. \tag{29}$$

Here, the mini-file $W^{\mathrm{s}}_{n,\mathcal{T}}$ and $W^{\mathrm{u}}_{n,\mathcal{T}}$ will be sent by the server and users, respectively. For each mini-file $W^{\mathrm{u}}_{n,\mathcal{T}}$, split it into L_1 pico-files of equal size $(1-\lambda)\cdot\frac{F}{L_1\binom{K}{t}}$, i.e., $W^{\mathrm{u}}_{n,\mathcal{T}} = \left(W^{\mathrm{u},1}_{n,\mathcal{T}},\dots,W^{\mathrm{u},L_1}_{n,\mathcal{T}}\right)$, where L_1 satisfies

$$\frac{K\cdot\binom{K-1}{t}\cdot L_1}{\alpha\min\{\lfloor\frac{K}{\alpha}\rfloor-1,t\}}\in\mathbb{Z}^+. \tag{30}$$

As we will see later, condition (29) ensures that communication loads can be optimally allocated between the server and the users, and (30) ensures that the number of subfiles is large enough to maximize multicast gain for the transmission in the D2D network.

In the delivery phase, each user k requests file W_{d_k}. The request vector $\mathbf{d}=(d_1,d_2,\dots,d_K)$ is informed by the server and all users. Please note that different parts of file W_{d_k} have been stored in the user cache memories, and thus the uncached parts of W_{d_k} can be sent both by the server and users. Subfiles

$$\left(W^{\mathrm{u},1}_{d_k,\mathcal{T}},\dots,W^{\mathrm{u},L_1}_{d_k,\mathcal{T}}:\mathcal{T}\subset[K],|\mathcal{T}|=t,k\notin\mathcal{T}\right)$$

are requested by user k and will be sent by the users via the D2D network. Subfiles

$$\left(W^{\mathrm{s}}_{d_k,\mathcal{T}}:\mathcal{T}\subset[K],|\mathcal{T}|=t,k\notin\mathcal{T}\right)$$

are requested by user k and will be sent by the server via the broadcast network.

First consider the subfiles sent by the users. Partition the K users into α groups of equal size:

$$\mathcal{G}_1,\dots,\mathcal{G}_\alpha,$$

where for $i,j=1,\dots,\alpha$, $\mathcal{G}_i\subseteq[K]:|\mathcal{G}_i|=\lfloor K/\alpha\rfloor$, and $\mathcal{G}_i\cap\mathcal{G}_j=\emptyset$, if $i\neq j$. In each group $\mathcal{G}_i$, one of $\lfloor K/\alpha\rfloor$ users plays the role of server and sends symbols based on its cached contents to the remaining $(\lfloor K/\alpha\rfloor-1)$ users within the group.

Focus on a group $\mathcal{G}_i$ and a set $\mathcal{S}\subset[K]:|\mathcal{S}|=t+1$. If $\mathcal{G}_i\subseteq\mathcal{S}$, then all nodes in $\mathcal{G}_i$ share subfiles

$$(W^{\mathrm{u},l}_{n,\mathcal{T}}:l\in[L_1],n\in[N],\mathcal{G}_i\subseteq\mathcal{T},|\mathcal{T}|=t).$$

In this case, user $k\in\mathcal{G}_i$ sends XOR symbols that contains the requested subfiles useful to all remaining $\lfloor K/\alpha\rfloor-1$ users in $\mathcal{G}_i$, i.e., $\oplus_{j\in\mathcal{G}_i\backslash\{k\}}W^{\mathrm{u},l(k,\mathcal{G}_i,\mathcal{S})}_{d_j,\mathcal{S}\backslash\{j\}}$, where $l(k,\mathcal{G}_i,\mathcal{S})\in[L_1]$ is a function of $(k,\mathcal{G}_i,\mathcal{S})$ which avoids redundant transmission of any fragments. .

If $\mathcal{S}\subseteq\mathcal{G}_i$, then the nodes in $\mathcal{S}$ share subfiles

$$(W^{\mathrm{u},l}_{n,\mathcal{T}}:l\in[L_1],n\in[N],\mathcal{T}\subset\mathcal{S},|\mathcal{T}|=t).$$

In this case, user $k\in\mathcal{S}$ sends an XOR symbol that contains the requested subfiles for all remaining $t-1$ users in $\mathcal{S}$, i.e., $\oplus_{j\in\mathcal{S}\backslash\{k\}}W^{\mathrm{u},l(k,\mathcal{G}_i,\mathcal{S})}_{d_j,\mathcal{S}\backslash\{j\}}$. Other groups perform the similar steps and concurrently deliver the remaining requested subfiles to other users.

By changing group partition and performing the delivery strategy described above, we can send all the requested subfiles

$$(W^{\mathrm{u},1}_{d_k,\mathcal{T}},\dots,W^{\mathrm{u},L_1}_{d_k,\mathcal{T}}:\mathcal{T}\subset[K],|\mathcal{T}|=t,k\notin\mathcal{T})_{k=1}^{K} \tag{31}$$

to the users.

Since α groups send signals in a parallel manner (α users can concurrently deliver contents), and each user in a group delivers a symbol containing $\min\{\lfloor K/\alpha\rfloor-1,t\}$ nonrepeating pico-files requested by other users, in order to send all requested subfiles in (31), we need to send in total

$$\frac{K \cdot \binom{K-1}{t} \cdot L_1}{\alpha \min\{\lfloor \frac{K}{\alpha} \rfloor - 1, t\}} \tag{32}$$

XOR symbols, each of size $\frac{1-\lambda}{\binom{K}{t}} F$ bits. Notice that L_1 is chosen according to (30), ensuring that (32) equals to an integer. Thus, we obtain R_2 as

$$\begin{aligned} R_2 = & \frac{K L_1 \cdot \binom{K-1}{t}}{\alpha \min\{\lfloor \frac{K}{\alpha} \rfloor - 1, t\}} \cdot \frac{1-\lambda}{L_1 \binom{K}{t}} \\ = & K\left(1-\frac{M}{N}\right) \frac{1}{1+t+\alpha \min\{\lfloor \frac{K}{\alpha} \rfloor - 1, t\}}, \end{aligned} \tag{33}$$

where the last equality holds by (29).

Now consider the delivery of the subfiles sent by the server. Apply the delivery strategy as in [1], i.e., the server broadcasts

$$\oplus_{k \in \mathcal{S}} W^{\mathrm{s}}_{d_k, \mathcal{S} \backslash \{k\}}$$

to all users, for all $\mathcal{S} \subseteq [K] : |\mathcal{S}| = t+1$. We obtain the transmission rate of the server

$$\begin{aligned} R_1 = & \lambda \cdot K\left(1-\frac{M}{N}\right) \cdot \frac{1}{1+t} \\ = & K\left(1-\frac{M}{N}\right) \frac{1}{1+t+\alpha \min\{\lfloor \frac{K}{\alpha} \rfloor - 1, t\}}. \end{aligned} \tag{34}$$

From (33) and (34), we can see that the choice λ in (29) guarantees equal communication loads at the server and users. Since the server and users transmit the signals simultaneously, the transmission delay of the whole network is the maximum between R_1 and R_2, i.e., $T_{\text{central}} = \max\{R_1, R_2\} = \frac{K(1-M/N)}{1+t+\alpha \min\{\lfloor K/\alpha \rfloor - 1, t\}}$, for some $\alpha \in [\alpha_{\max}]$.

5. Coding Scheme under Decentralized Data Placement

In this section, we present a novel decentralized coded caching scheme for joint broadcast network and D2D network. The decentralized scenario is much more complicated than the centralized scenario, since each subfile can be stored by $s = 1, 2, \ldots, K$ users, leading to a dynamic file-splitting and communication strategy in the D2D network. We first use an illustrative example to demonstrate how we form D2D communication groups, split data and deliver data, and then present our generalized coding caching scheme.

5.1. An Illustrative Example

Consider a joint broadcast and D2D network consisting of $K = 7$ users. When using the decentralized data placement strategy, the subfiles cached by user k can be written as

$$\left(W_{n,\mathcal{T}} : n \in [N], k \in \mathcal{T}, \mathcal{T} \subseteq [7]\right). \tag{35}$$

We focus on the delivery of subfiles $W_{n,\mathcal{T}} : n \in [N], k \in \mathcal{T}, |\mathcal{T}| = s = 4$, i.e., each subfile is stored by $s = 4$ users. A similar process can be applied to deliver other subfiles with respect to $s \in [K] \backslash \{4\}$.

To allocate communication loads between the server and users, we divide each subfile into two mini-files $W_{n,\mathcal{T}} = \left(W^{\mathrm{s}}_{n,\mathcal{T}}, W^{\mathrm{u}}_{n,\mathcal{T}}\right)$, where mini-files $\{W^{\mathrm{s}}_{n,\mathcal{T}}\}$ and $\{W^{\mathrm{u}}_{n,\mathcal{T}}\}$ will be sent by the server and users, respectively. To reduce the transmission delay, the size of

$W^{\mathrm{s}}_{n,\mathcal{T}}$ and $W^{\mathrm{u}}_{n,\mathcal{T}}$ need to be chosen properly such that $R_1 = R_2$, i.e., the transmission rate of the server and users are equal; see (37) and (39) ahead.

Divide all the users into two non-intersecting groups $(\mathcal{G}^r_1, \mathcal{G}^r_2)$, for $r \in [35]$ which satisfies

$$\mathcal{G}^r_1 \subset [K], \mathcal{G}^r_2 \subset [K], |\mathcal{G}^r_1| = 4, |\mathcal{G}^r_2| = 3, \mathcal{G}^r_1 \cap \mathcal{G}^r_2 = \emptyset.$$

There are $\binom{7}{4} = 35$ kinds of partitions in total, thus $r \in [35]$. Please note that for any user $k \in \mathcal{G}^r_i$, $|\mathcal{G}^r_i| - 1$ of its requested mini-files are already cached by the rest users in $\mathcal{G}^r_i$, for $i = 1, 2$.

To avoid repetitive transmission of any mini-file, each mini-file in

$$(W^{\mathrm{u}}_{d_k,\mathcal{T}\setminus\{k\}} : \mathcal{T} \subseteq [7], k \in [7])$$

is divided into non-overlapping pico-files $W^{\mathrm{u}_1}_{d_k,\mathcal{T}\setminus\{k\}}$ and $W^{\mathrm{u}_2}_{d_k,\mathcal{T}\setminus\{k\}}$, i.e.,

$$W^{\mathrm{u}}_{d_k,\mathcal{T}\setminus\{k\}} = (W^{\mathrm{u}_1}_{d_k,\mathcal{T}\setminus\{k\}}, W^{\mathrm{u}_2}_{d_k,\mathcal{T}\setminus\{k\}}).$$

The sizes of $W^{\mathrm{u}_1}_{n,\mathcal{T}}$ and $W^{\mathrm{u}_2}_{n,\mathcal{T}}$ need to be chosen properly to have equal transmission rate of group $\mathcal{G}^r_1$ and $\mathcal{G}^r_2$; see (51) and (52) ahead.

To allocate communication loads between the two different types of groups, split each $W^{\mathrm{u}_1}_{d_k,\mathcal{T}\setminus\{k\}}$ and $W^{\mathrm{u}_2}_{d_k,\mathcal{T}\setminus\{k\}}$ into 3 and two equal fragments, respectively, e.g.,

$$\begin{aligned} W^{\mathrm{u}_1}_{d_2,\{1,3,4\}} &= \left(W^{\mathrm{u}_1,1}_{d_2,\{1,3,4\}}, W^{\mathrm{u}_1,2}_{d_2,\{1,3,4\}}, W^{\mathrm{u}_1,3}_{d_2,\{1,3,4\}}\right), \\ W^{\mathrm{u}_2}_{d_2,\{1,3,4\}} &= \left(W^{\mathrm{u}_2,1}_{d_2,\{1,3,4\}}, W^{\mathrm{u}_2,2}_{d_2,\{1,3,4\}}\right). \end{aligned}$$

During the delivery phase, in each round, one user in each group produces and multicasts an XOR symbol to all other users in the same group, as shown in Table 2.

Table 2. Parallel user delivery when $K = 7, s = 4, \mathcal{G}^r_1 = 4$ and $\mathcal{G}^r_2 = 3, r \in [35]$.

$\{1,2,3,4\}$	$\{5,6,7\}$
user 1: $W^{\mathrm{u}_1,1}_{d_2,\{1,3,4\}} \oplus W^{\mathrm{u}_1,1}_{d_3,\{1,2,4\}} \oplus W^{\mathrm{u}_1,1}_{d_4,\{1,2,3\}}$	user 5: $\bigcup_{x\in\{1,2,3,4\}} W^{\mathrm{u}_2,1}_{d_6,\{5,7,x\}} \oplus W^{\mathrm{u}_2,1}_{d_7,\{5,6,x\}}$
user 2: $W^{\mathrm{u}_1,1}_{d_1,\{2,3,4\}} \oplus W^{\mathrm{u}_1,2}_{d_3,\{1,2,4\}} \oplus W^{\mathrm{u}_1,2}_{d_4,\{1,2,3\}}$	user 6: $\bigcup_{x\in\{1,2,3,4\}} W^{\mathrm{u}_2,1}_{d_5,\{6,7,x\}} \oplus W^{\mathrm{u}_2,2}_{d_7,\{5,6,x\}}$
user 3: $W^{\mathrm{u}_1,2}_{d_2,\{1,3,4\}} \oplus W^{\mathrm{u}_1,2}_{d_1,\{2,3,4\}} \oplus W^{\mathrm{u}_1,3}_{d_4,\{1,2,3\}}$	user 7: $\bigcup_{x\in\{1,2,3,4\}} W^{\mathrm{u}_2,2}_{d_6,\{5,7,x\}} \oplus W^{\mathrm{u}_2,2}_{d_5,\{6,7,x\}}$
user 4: $W^{\mathrm{u}_1,3}_{d_2,\{1,3,4\}} \oplus W^{\mathrm{u}_1,3}_{d_3,\{1,2,4\}} \oplus W^{\mathrm{u}_1,3}_{d_1,\{2,3,4\}}$	
$\{1,2,3,5\}$	$\{4,6,7\}$
user 1: $W^{\mathrm{u}_1,1}_{d_2,\{1,3,5\}} \oplus W^{\mathrm{u}_1,1}_{d_3,\{1,2,5\}} \oplus W^{\mathrm{u}_1,1}_{d_5,\{1,2,3\}}$	user 4: $\bigcup_{x\in\{1,2,3,5\}} W^{\mathrm{u}_2,y(..)}_{d_6,\{4,7,x\}} \oplus W^{\mathrm{u}_2,y(..)}_{d_7,\{4,6,x\}}$
user 2: $W^{\mathrm{u}_1,1}_{d_1,\{2,3,5\}} \oplus W^{\mathrm{u}_1,2}_{d_3,\{1,2,5\}} \oplus W^{\mathrm{u}_1,2}_{d_5,\{1,2,3\}}$	user 6: $\bigcup_{x\in\{1,2,3,5\}} W^{\mathrm{u}_2,1}_{d_4,\{6,7,x\}} \oplus W^{\mathrm{u}_2,y(..)}_{d_7,\{4,6,x\}}$
user 3: $W^{\mathrm{u}_1,2}_{d_2,\{1,3,5\}} \oplus W^{\mathrm{u}_1,2}_{d_1,\{2,3,5\}} \oplus W^{\mathrm{u}_1,3}_{d_5,\{1,2,3\}}$	user 7: $\bigcup_{x\in\{1,2,3,5\}} W^{\mathrm{u}_2,y(..)}_{d_6,\{4,7,x\}} \oplus W^{\mathrm{u}_2,2}_{d_4,\{6,7,x\}}$
user 5: $W^{\mathrm{u}_1,3}_{d_2,\{1,3,5\}} \oplus W^{\mathrm{u}_1,3}_{d_3,\{125\}} \oplus W^{\mathrm{u}_1,3}_{d_1,\{235\}}$	
$\{1,2,3,6\}$	$\{4,5,7\}$
user 1: $W^{\mathrm{u}_1,1}_{d_2,\{1,3,6\}} \oplus W^{\mathrm{u}_1,1}_{d_3,\{1,2,6\}} \oplus W^{\mathrm{u}_1,1}_{d_6,\{1,2,3\}}$	user 4: $\bigcup_{x\in\{1,2,3,6\}} W^{\mathrm{u}_2,y(..)}_{d_5,\{4,7,x\}} \oplus W^{\mathrm{u}_2,y(..)}_{d_7,\{4,5,x\}}$
user 2: $W^{\mathrm{u}_1,1}_{d_1,\{2,3,6\}} \oplus W^{\mathrm{u}_1,2}_{d_3,\{1,2,6\}} \oplus W^{\mathrm{u}_1,2}_{d_6,\{1,2,3\}}$	user 5: $\bigcup_{x\in\{1,2,3,6\}} W^{\mathrm{u}_2,1}_{d_4,\{5,7,x\}} \oplus W^{\mathrm{u}_2,y(..)}_{d_7,\{4,5,x\}}$
user 3: $W^{\mathrm{u}_1,2}_{d_2,\{1,3,6\}} \oplus W^{\mathrm{u}_1,2}_{d_1,\{2,3,6\}} \oplus W^{\mathrm{u}_1,3}_{d_6,\{1,2,3\}}$	user 7: $\bigcup_{x\in\{1,2,3,6\}} W^{\mathrm{u}_2,y(..)}_{d_5,\{4,7,x\}} \oplus W^{\mathrm{u}_2,2}_{d_4,\{5,7,x\}}$
user 6: $W^{\mathrm{u}_1,3}_{d_2,\{1,3,6\}} \oplus W^{\mathrm{u}_1,3}_{d_3,\{1,2,6\}} \oplus W^{\mathrm{u}_1,3}_{d_1,\{2,3,6\}}$	
$\cdots$ $\cdots\cdots$	$\cdots$ $\cdots\cdots$

There should be 35 partitions in total while the table only shows three partitions.

Please note that in this example, each group only appears one time among all partitions. However, for some other values of s, each group could appear multiple times in different partitions. For example, when $s = 2$, group $\{1,2\}$ appears in both partitions $\{\{1,2\},\{3,4\},\{5,6,7\}\}$ and $\{\{1,2\},\{3,5\},\{4,6,7\}\}$. To reduce the transmission delay, one should balance communication loads between all groups, and between the server and users as well.

5.2. The Generalized Decentralized Coded Caching Scheme

In the placement phase, each user k applies the caching function to map a subset of $\frac{MF}{N}$ bits of file $W_n, n = 1,...,N$, into its cache memory at random: $W_n = \Big(W_{n,\mathcal{T}} : \mathcal{T} \subseteq [K]\Big)$. The subfiles cached by user k can be written as $\Big(W_{n,\mathcal{T}} : n \in [N], k \in \mathcal{T}, \mathcal{T} \subseteq [K]\Big)$. When the size of file F is sufficiently large, by the law of large numbers, the subfile size with high probability can be written by

$$|W_{n,\mathcal{T}}| \approx p^{|\mathcal{T}|}(1-p)^{K-|\mathcal{T}|}. \tag{36}$$

The delivery procedure can be characterized into three different levels: allocating communication loads between the server and user, inner-group coding (i.e., transmission in each group) and parallel delivery among groups.

5.2.1. Allocating Communication Loads between the Server and User

To allocate communication loads between the server and users, split each subfile $W_{n,\mathcal{T}}$, for $\mathcal{T} \subseteq [K] : \mathcal{T} \neq \emptyset$, into two non-overlapping mini-files

$$W_{n,\mathcal{T}} = \Big(W^{\mathrm{s}}_{n,\mathcal{T}}, W^{\mathrm{u}}_{n,\mathcal{T}}\Big),$$

where

$$\begin{aligned} |W^{\mathrm{s}}_{n,\mathcal{T}}| &= \lambda \cdot |W_{n,\mathcal{T}}|, \\ |W^{\mathrm{u}}_{n,\mathcal{T}}| &= (1-\lambda) \cdot |W_{n,\mathcal{T}}|, \end{aligned} \tag{37}$$

and λ is a design parameter whose value is determined in Remark 5.

Mini-files $(W^{\mathrm{s}}_{d_k,\mathcal{T}\setminus\{k\}} : k \in [K])$ will be sent by the server using the decentralized coded caching scheme for the broadcast network [2], leading to the transmission delay

$$\lambda R_{\mathrm{s}} = \lambda \frac{1 - M/N}{M/N}\Big(1 - (1 - \frac{M}{N})^K\Big), \tag{38}$$

where R_{s} is defined in (19).

Mini-files $(W^{\mathrm{u}}_{d_k,\mathcal{T}\setminus\{k\}} : k \in [K])$ will be sent by users using *parallel user delivery* described in Section 5.2.3. The corresponding transmission rate is

$$R_2 = (1-\lambda)R_{\mathrm{u}}, \tag{39}$$

where R_{u} represents the transmission bits sent by each user normalized by F .

Since subfile $W_{d_k,\emptyset}$ is not cached by any user and must be sent exclusively from the server, the corresponding transmission delay for sending $(W_{d_k,\emptyset} : k \in [K])$ is

$$R_{\emptyset} = K(1 - \frac{M}{N})^K, \tag{40}$$

where $R_{\emptyset}$ coincides with the definition in (18).

By (38)–(40), we have

$$R_1 = R_{\emptyset} + \lambda R_{\mathrm{s}}, \quad R_2 = (1-\lambda)R_{\mathrm{u}}. \tag{41}$$

According to (8), we have $T_{\text{decentral}} = \max\{R_1, R_2\}$.

Remark 5 (Choice of λ)**.** *The parameter λ is chosen such that $T_{\text{decentral}}$ is minimized. If $R_{\text{u}} < R_{\emptyset}$, then the inequality $R_2 \le R_1$ always holds and $T_{\text{decentral}}$ reaches the minimum $T_{\text{decentral}} = R_{\emptyset}$ with $\lambda = 0$. If $R_{\text{u}} \ge R_{\emptyset}$, solving $R_1 = R_2$ yields $\lambda = \frac{R_{\text{u}} - R_{\emptyset}}{R_{\text{s}} + R_{\text{u}}}$ and $T_{\text{decentral}} = \frac{R_{\text{s}} R_{\text{u}}}{R_{\text{s}} + R_{\text{u}} - R_{\emptyset}}$.*

5.2.2. Inner-Group Coding

Given parameters $(s, \mathcal{G}, \text{i}, \gamma)$ where $s \in [K-1]$, $\mathcal{G} \subseteq [K]$, $\text{i} \in \{\text{u}, \text{u}_1, \text{u}_2\}$ with indicators $\text{u}, \text{u}_1, \text{u}_2$ described in (37) and (51), and $\gamma \in \mathbb{Z}^+$, we present how to successfully deliver

$$(W^{\text{i}}_{d_k, \mathcal{S} \setminus \{k\}} : \forall \mathcal{S} \subseteq [K], |\mathcal{S}| = s, \mathcal{G} \subseteq \mathcal{S})$$

to every user $k \in \mathcal{G}$ via D2D communication.

Split each $W^{\text{i}}_{d_k, \mathcal{S} \setminus \{k\}}$ into $(|\mathcal{G}| - 1)\gamma$ non-overlapping fragments of equal size, i.e.,

$$W^{\text{i}}_{d_k, \mathcal{S} \setminus \{k\}} = \left(W^{\text{i}, l}_{d_k, \mathcal{S} \setminus \{k\}} : l \in [(|\mathcal{G}| - 1)\gamma] \right), \tag{42}$$

and each user $k \in \mathcal{G}$ takes turn to broadcast XOR symbol

$$X^{\text{i}}_{k, \mathcal{G}, s} \triangleq \oplus_{j \in \mathcal{G} \setminus \{k\}} W^{\text{i}, l(j, \mathcal{G}, \mathcal{S})}_{d_j, \mathcal{S} \setminus \{j\}}, \tag{43}$$

where $l(k, \mathcal{G}, \mathcal{S}) \in [(|\mathcal{G}| - 1)\gamma]$ is a function of $(k, \mathcal{G}, \mathcal{S})$ which avoids redundant transmission of any fragments. The XOR symbol $X^{\text{i}}_{k, \mathcal{G}, s}$ will be received and decoded by the remaining users in $\mathcal{G}$.

For each group $\mathcal{G}$, inner-group coding encodes in total $\binom{K - |\mathcal{G}|}{s - |\mathcal{G}|}$ of $W^{\text{i}}_{d_k, \mathcal{S} \setminus \{k\}}$, and each XOR symbol $X^{\text{i}}_{k, \mathcal{G}, s}$ in (43) contains fragments required by $|\mathcal{G}| - 1$ users in $\mathcal{G}$.

5.2.3. Parallel Delivery among Groups

The parallel user delivery consists of $(K-1)$ rounds characterized by $s = 2, \ldots, K$. In each round s, mini-files

$$(W^{\text{u}}_{d_k, \mathcal{T} \setminus \{k\}} : \forall \mathcal{T} \subseteq [K], |\mathcal{T}| = s, k \in [K])$$

are recovered through D2D communication.

The key idea is to partition K users into $\lceil \frac{K}{s} \rceil$ groups for each communication round $s \in \{2, ..., K\}$, and let each group perform the D2D coded caching scheme [35] to exchange information. If $(K \bmod s) \neq 0$, there will be $\lfloor \frac{K}{s} \rfloor$ numbers of groups of the same size s, and an *abnormal* group of size $(K \bmod s)$, leading to an asymmetric caching setup. We optimally allocate the communication loads between the two types of groups, and between the broadcast network and D2D network as well.

Based on K, s and $\alpha_{\max}$, the delivery strategy in the D2D network is divided into 3 cases:

- Case 1: $\lceil \frac{K}{s} \rceil > \alpha_{\max}$. In this case, $\alpha_{\max}$ users are allowed to send data simultaneously. Select $s \cdot \alpha_{\max}$ users from all users and divide them into $\alpha_{\max}$ groups of equal size s. The total number of such kinds of partition is

$$\beta_1 \triangleq \frac{\binom{K}{s}\binom{K-s}{s} \cdots \binom{K - s(\alpha_{\max} - 1)}{s}}{\alpha_{\max}!}. \tag{44}$$

  In each partition, $\alpha_{\max}$ users, selected from $\alpha_{\max}$ groups, respectively, send data in parallel via the D2D network.

- Case 2: $\lceil\frac{K}{s}\rceil \leq \alpha_{\max}$ and $(K \bmod s) < 2$. In this case, choose $(\lfloor\frac{K}{s}\rfloor - 1)s$ users from all users and partition them into $(\lfloor\frac{K}{s}\rfloor - 1)$ groups of equal size s. The total number of such kind partition is

$$\beta_2 \triangleq \frac{\binom{K}{s}\binom{K-s}{s}\cdots\binom{K-s(\lfloor\frac{K}{s}\rfloor-1)}{s}}{\lfloor\frac{K}{s}\rfloor!}. \tag{45}$$

In each partition, $(\lfloor\frac{K}{s}\rfloor - 1)$ users selected from $(\lfloor\frac{K}{s}\rfloor - 1)$ groups of equal size s, respectively, together with an extra user selected from the *abnormal* group of size $K - s(\lfloor\frac{K}{s}\rfloor - 1)$ send data in parallel via the D2D network.

- Case 3: $\lceil\frac{K}{s}\rceil \leq \alpha_{\max}$ and $(K \bmod s) \geq 2$. In this case, every s users form a group, resulting in $\lfloor\frac{K}{s}\rfloor$ groups consisting of $s\lfloor\frac{K}{s}\rfloor$ users. The remaining $(K \bmod s)$ users form an *abnormal* group. The total number of such kind of partition is

$$\beta_3 = \beta_2. \tag{46}$$

In each partition, $\lfloor\frac{K}{s}\rfloor$ users selected from $\lfloor\frac{K}{s}\rfloor$ groups of equal size s, respectively, together with an extra user selected from the abnormal group of size $(K \bmod s)$ send data in parallel via the D2D network.

Thus, the exact number of users who parallelly send signals can be written as follows:

$$\alpha_{\mathrm{D}} = \begin{cases} \alpha_{\max}, & \text{case 1,} \\ \lfloor\dfrac{K}{s}\rfloor, & \text{case 2,} \\ \lceil\dfrac{K}{s}\rceil, & \text{case 3.} \end{cases} \tag{47}$$

Please note that each group $\mathcal{G}$ re-appears

$$N_{\mathcal{G}} \triangleq \frac{\binom{K-s}{s}\cdots\binom{K-s\cdot(\alpha_{\mathrm{D}}-1)}{s}}{(\alpha_{\mathrm{D}}-1)!} \tag{48}$$

times among $[\beta_c]$ partitions .

Now we present the decentralized scheme for these three cases as follows.

Case 1 $(\lceil\frac{K}{s}\rceil > \alpha_{\max})$: Consider a partition $r \in [\beta_1]$, denoted by

$$\mathcal{G}_1^r, \ldots, \mathcal{G}_{\alpha_{\mathrm{D}}}^r,$$

where $|\mathcal{G}_i^r| = s$ and $\mathcal{G}_i^r \cap \mathcal{G}_j^r = \emptyset, \forall i, j \in [\alpha_{\mathrm{D}}]$ and $i \neq j$.

Since each group $\mathcal{G}_i^r$ re-appears $N_{\mathcal{G}_i^r}$ times among $[\beta_1]$ partitions, and $(|\mathcal{G}_i^r| - 1)$ users take turns to broadcast XOR symbols (43) in each group $\mathcal{G}_i^r$, in order to guarantee that each group can send a unique fragment without repetition, we split each mini-file $W^{\mathrm{u}}_{d_k,\mathcal{S}\backslash\{k\}}$ into $(|\mathcal{G}_i^r| - 1)N_{\mathcal{G}_i^r}$ fragments of equal size.

Each group $\mathcal{G}_i^r$, for $r \in [\beta_1]$ and $i \in [\alpha_{\mathrm{D}}]$, performs inner-group coding (see Section 5.2.2) with parameters

$$(s, \mathcal{G}_i^r, \mathrm{u}, N_{\mathcal{G}_i^r}),$$

for all s satisfying $\lceil\frac{K}{s}\rceil > \alpha_{\max}$. For each round r, all groups $\mathcal{G}_1^r, \ldots, \mathcal{G}_{\alpha_{\mathrm{D}}}^r$ parallelly send XOR symbols containing $|\mathcal{G}_i^r| - 1$ fragments required by other users of its group. By the fact that the partitioned groups traverse every set $\mathcal{T}$, i.e.,

$$\mathcal{T} \subseteq \{\mathcal{G}_1^r \cup \ldots \cup \mathcal{G}_{\alpha_{\mathrm{D}}}^r\}_{r=1}^{\beta_1}, \forall \mathcal{T} \subseteq [K] : |\mathcal{T}| = s,$$

and since inner-group coding enables each group $\mathcal{G}_i^r$ to recover

$$(W^{\mathrm{u}}_{d_k,\mathcal{S}\backslash\{k\}} : \forall\mathcal{S} \subseteq [K], |\mathcal{S}| = s, \mathcal{G}_i^r \subseteq \mathcal{S}, k \in [K]),$$

we can recover all required mini-files

$$(W^{\mathrm{u}}_{d_k,\mathcal{T}\backslash\{k\}} : \forall\mathcal{T} \subseteq [K], |\mathcal{T}| = s, k \in [K]).$$

The transmission delay of Case 1 in round s is thus

$$\begin{aligned} R^{\mathrm{u}}_{\mathrm{case1}}(s) &\triangleq \sum_{r\in[\beta_1]} \sum_{k\in\mathcal{G}_i^r} |X^{\mathrm{u}}_{k,\mathcal{G}_i^r,s}| \\ &\overset{(a)}{=} \frac{K\binom{K-1}{s-1}}{\alpha_{\mathrm{D}}(s-1)} |W^{\mathrm{u}}_{d_k,\mathcal{T}\backslash\{k\}}| \\ &= \frac{K\binom{K-1}{s-1}}{\alpha_{\max}(s-1)} (1-\lambda)p^{s-1}(1-p)^{K-s+1}, \end{aligned} \tag{49}$$

where (a) follows by (44) and (48).

Case 2 ($\lceil \frac{K}{s} \rceil \le \alpha_{\max}$ and $(K \bmod s) < 2$): We apply the same delivery procedure as Case 1, except that β_1 is replaced by β_2 and $\alpha_{\mathrm{D}} = \lfloor \frac{K}{s} \rfloor$. Thus, the transmission delay in round s is

$$\begin{aligned} R^{\mathrm{u}}_{\mathrm{case2}}(s) &= \frac{K\binom{K-1}{s-1}}{\alpha_D(s-1)} |W^{\mathrm{u}}_{d_k,\mathcal{T}\backslash\{k\}}| \\ &= \frac{K\binom{K-1}{s-1}}{\lfloor \frac{K}{s} \rfloor (s-1)} (1-\lambda)p^{s-1}(1-p)^{K-s+1}. \end{aligned} \tag{50}$$

Case 3 ($\lceil \frac{K}{s} \rceil \le \alpha_{\max}$ and $(K \bmod s) \ge 2$): Consider a partition $r \in [\beta_3]$, denoted as

$$\mathcal{G}_1^r, \ldots, \mathcal{G}^r_{\alpha_{\mathrm{D}}},$$

where $\mathcal{G}_i^r \subseteq [K]$, $\mathcal{G}_i^r \cap \mathcal{G}_j^r = \emptyset$, $\forall i,j \in [\alpha_{\mathrm{D}} - 1]$ and $i \neq j$ and $\mathcal{G}^r_{\alpha_{\mathrm{D}}} = [K] \backslash (\cup_{i=1}^{\alpha_{\mathrm{D}}-1} \mathcal{G}_i^r)$ with $|\mathcal{G}_i^r| = s$, $|\mathcal{G}^r_{\alpha_{\mathrm{D}}}| = (K \bmod s)$.

Since group $\mathcal{G}_i^r : i \in [\alpha_{\mathrm{D}} - 1]$ and $\mathcal{G}^r_{\alpha_{\mathrm{D}}}$ have different group sizes, we further split each mini-file $W^{\mathrm{u}}_{d_k,\mathcal{T}\backslash\{k\}}$ into two non-overlapping fragments such that

$$\begin{aligned} |W^{\mathrm{u}_1}_{d_k,\mathcal{T}\backslash\{k\}}| &= \lambda_2 |W^{\mathrm{u}}_{d_k,\mathcal{T}\backslash\{k\}}|, \\ |W^{\mathrm{u}_2}_{d_k,\mathcal{T}\backslash\{k\}}| &= (1-\lambda_2)|W^{\mathrm{u}}_{d_k,\mathcal{T}\backslash\{k\}}|, \end{aligned} \tag{51}$$

where $\lambda_2 \in [0,1]$ is a designed parameter satisfying (52).

Split each mini-file $W^{\mathrm{u}_1}_{d_k,\mathcal{S}\backslash\{k\}}$ and $W^{\mathrm{u}_2}_{d_k,\mathcal{S}\backslash\{k\}}$ into fragments of equal size:

$$\begin{aligned} W^{\mathrm{u}_1}_{d_k,\mathcal{S}\backslash\{k\}} &= \left(W^{\mathrm{u}_1,l}_{d_k,\mathcal{S}\backslash\{k\}} : l \in [(s-1)N_{\mathcal{G}_i^r}] \right), \\ W^{\mathrm{u}_2}_{d_k,\mathcal{S}\backslash\{k\}} &= \left(W^{\mathrm{u}_2,l}_{d_k,\mathcal{S}\backslash\{k\}} : l \in \left[(|\mathcal{G}^r_{\alpha_{\mathrm{D}}}| - 1) \binom{s-1}{|\mathcal{G}^r_{\alpha_{\mathrm{D}}}| - 1} N_{\mathcal{G}_i^r} \right] \right). \end{aligned}$$

Following the similar encoding operation in (43), group $\mathcal{G}_i^r : i \in [\alpha_{\mathrm{D}} - 1]$ and group $\mathcal{G}^r_{\alpha_{\mathrm{D}}}$ send the following XOR symbols, respectively:

$$(X^{\mathrm{u}_1}_{k,\mathcal{G}_i^r,s} : k \in \mathcal{G}_i^r)_{i=1}^{(\alpha_{\mathrm{D}}-1)}, \quad (X^{\mathrm{u}_2}_{k,\mathcal{G}^r_{\alpha_{\mathrm{D}}},s} : k \in \mathcal{G}^r_{\alpha_{\mathrm{D}}}).$$

For each $s \in \{2, \dots, K\}$, the transmission delay for sending the XOR symbols above by group $\mathcal{G}_i^r : i \in [\alpha_{\rm D} - 1]$ and group $\mathcal{G}^r_{\lceil \frac{K}{s} \rceil}$ can be written as

$$R^{\rm u_1}_{\rm case3}(s) = \frac{\lambda_2 K\binom{K-1}{s-1}}{(\alpha_{\rm D}-1)(s-1)} \cdot |W^{\rm u}_{d_k,\mathcal{T}\backslash\{k\}}|,$$
$$R^{\rm u_2}_{\rm case3}(s) = \frac{(1-\lambda_2) K\binom{K-1}{s-1}}{(K \bmod s)-1} \cdot |W^{\rm u}_{d_k,\mathcal{T}\backslash\{k\}}|,$$

respectively. Since $\mathcal{G}_i : i \in [\lfloor \frac{K}{s} \rfloor]$ and group $\mathcal{G}_{\lceil \frac{K}{s} \rceil}$ can send signals in parallel, by letting

$$R^{\rm u_1}_{\rm case3}(s) = R^{\rm u_2}_{\rm case3}(s), \tag{52}$$

we eliminate the parameter λ_2 and obtain the balanced transmission delay at users for Case 3:

$$R^{\rm u}_{\rm case3}(s) \triangleq \frac{K\binom{K-1}{s-1}}{K-1-\lfloor \frac{K}{s} \rfloor}(1-\lambda)p^{s-1}(1-p)^{K-s+1}. \tag{53}$$

Remark 6. *The condition $\lceil \frac{K}{s} \rceil > \alpha_{\max}$ in Case 1 implies that $s \le \lceil \frac{K}{\alpha_{\max}} \rceil - 1$. In this regime, the transmission delay is given in (49). If $s \ge \lceil \frac{K}{\alpha_{\max}} \rceil - 1$ and $(K \mod s) < 2$, scheme for Case 2 starts to work and the transmission delay is given in (50); If $s \ge \lceil \frac{K}{\alpha_{\max}} \rceil - 1$ and $(K \mod s) \ge 2$, scheme for Case 3 starts to work and the transmission delay is given in (53).*

In each round $s \in \{2, \dots, K\}$, all requested mini-files can be recovered by the delivery strategies above. By Remark 6, the transmission delay in the D2D network is

$$\begin{aligned} R_2 &= (1-\lambda)\frac{1}{\alpha_{\max}} \sum_{s=2}^{\lceil \frac{K}{\alpha_{\max}} \rceil - 1} [\frac{s\binom{K}{s}}{s-1}p^{s-1}(1-p)^{K-s+1}] + (1-\lambda) \sum_{s=\lceil \frac{K}{\alpha_{\max}} \rceil}^{K} [\frac{K\binom{K-1}{s-1}}{f(K,s)}p^{s-1}(1-p)^{K-s+1}] \\ &= (1-\lambda)R_{\rm u}, \end{aligned} \tag{54}$$

where $R_{\rm u}$ is defined in (20) and

$$f(K,s) \triangleq \begin{cases} \lfloor \frac{K}{s} \rfloor (s-1), & (K \bmod s) < 2, \\ K-1-\lfloor K/s \rfloor, & (K \bmod s) \ge 2. \end{cases} \tag{55}$$

6. Conclusions

In this paper, we considered a cache-aided communication via joint broadcast network with a D2D network. Two novel coded caching schemes were proposed for centralized and decentralized data placement settings, respectively. Both schemes achieve a parallel gain and a cooperation gain by efficiently exploiting communication opportunities in the broadcast and D2D networks, and optimally allocating communication loads between the server and users. Furthermore, we showed that in the centralized case, letting too many users parallelly send information could be harmful. The information theoretic converse bounds were established, with which we proved that the centralized scheme achieves the optimal transmission delay within a constant multiplicative gap in all regimes, and the decentralized scheme is also order-optimal when the cache size of each user is larger than a small threshold which tends to zero as the number of users tends to infinity. Our work indicates that combining the cache-aided broadcast network with the cache-aided D2D network can greatly reduce the transmission latency.

Author Contributions: Project administration, Y.W.; Writing—original draft, Z.H., J.C. and X.Y.; Writing—review & editing, S.M. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by National Natural Science Foundation of China grant number 61901267.

Conflicts of Interest: The authors declare no conflict of interest.

Appendix A. Proof of the Converse

Let T_1^* and T_2^* denote the optimal rate sent by the server and each user. We first consider an enhance system where every user is served by an exclusive server and user, which both store full files in the database, then we are easy to obtain the following lower bound:

$$T^* \geq \frac{1}{2}(1 - \frac{M}{N}). \tag{A1}$$

Another lower bound follows similar idea to [1]. However, due to the flexibility of D2D network, the connection and partitioning status between users can change during the delivery phase, prohibiting the direct application of the proof in [1] into the hybrid network considered in this paper. Moreover, the parallel transmission of the server and many users creates abundant different signals in the networks, making the scenario more sophisticated.

Consider the first s users with cache contents $Z_1, ..., Z_s$. Define $X_{1,0}$ as the signal sent by the server, and $X_{1,1}, \ldots, X_{1,\alpha_{\max}}$ as the signals sent by the $\alpha_{\max}$ users, respectively, where $X_{j,i} \in [\lfloor 2^{T_2^* F} \rfloor]$ for $j \in [s]$ and $i \in [\alpha_{\max}]$. Assume that $W_1, \ldots, W_s$ are determined by $X_{1,0}$, $X_{1,1}, \ldots, X_{1,\alpha_{\max}}$ and $Z_1, \ldots, Z_s$. Additionally, define $X_{2,0}$, $X_{2,1}, \ldots, X_{2,\alpha_{\max}}$ as the signals which enable the users to decode $W_{s+1}, ..., W_{2s}$. Continue the same process such that $X_{\lfloor N/s \rfloor,0}$, $X_{\lfloor N/s \rfloor,1}, \ldots, X_{\lfloor N/s \rfloor,\alpha_{\max}}$ are the signals which enable the users to decode $W_{s\lfloor N/s \rfloor - s+1}, ..., W_{s\lfloor N/s \rfloor}$. We then have $Z_1, \ldots, Z_s$, $X_{1,0}, \ldots, X_{\lfloor N/s \rfloor,0}$, and

$$X_{1,1}, \ldots, X_{1,\alpha_{\max}}, \ldots, X_{\lfloor N/s \rfloor,1}, \ldots, X_{\lfloor N/s \rfloor,\alpha_{\max}}$$

to determine $W_1, \ldots, W_{s\lfloor N/s \rfloor}$. Let

$$\mathbf{X}_{1:\alpha_{\max}} \triangleq (X_{1,1}, \ldots, X_{1,\alpha_{\max}}, \ldots, X_{\lfloor N/s \rfloor,1}, \ldots, X_{\lfloor N/s \rfloor,\alpha_{\max}}).$$

By the definitions of T_1^*, T_2^* and the encoding function (5), we have

$$H(X_{1,0}, \ldots, X_{\lfloor N/s \rfloor,0}) \leq \lfloor N/s \rfloor T_1^* F, \tag{A2}$$
$$H(\mathbf{X}_{1:\alpha_{\max}}) \leq \lfloor N/s \rfloor \alpha_{\max} T_2^* F, \tag{A3}$$
$$H(\mathbf{X}_{1:\alpha_{\max}}, Z_1, \ldots, Z_s) \leq KMF. \tag{A4}$$

Consider the cut separating $X_{1,0}, \ldots, X_{\lfloor N/s \rfloor,0}$, $\mathbf{X}_{1:\alpha_{\max}}$, and $Z_1, \ldots, Z_s$ from the corresponding s users. By the cut-set bound and (A2), we have

$$\lfloor \frac{N}{s} \rfloor sF \leq \lfloor \frac{N}{s} \rfloor T_1^* F + KMF, \tag{A5}$$
$$\lfloor \frac{N}{s} \rfloor sF \leq \lfloor \frac{N}{s} \rfloor T_1^* F + sMF + \lfloor \frac{N}{s} \rfloor \alpha_{\max} T_2^* F. \tag{A6}$$

Since we have $T^* \geq T_1^*$ and $T^* \geq \max\{T_1^*, T_2^*\}$ from the above definition, we obtain

$$T^* \geq \max_{s \in [K]} (s - \frac{KM}{\lfloor N/s \rfloor}), \tag{A7}$$
$$T^* \geq \max_{s \in [K]} (s - \frac{sM}{\lfloor N/s \rfloor}) \frac{1}{1 + \alpha_{\max}}. \tag{A8}$$

Appendix B

We prove that T_{central} is within a constant multiplicative gap of the minimum transmission delay T^* for all values of M. To prove the result, we compare them in the following regimes.

- If $0.6393 < t < \lfloor K/\alpha \rfloor - 1$, from Theorem 1, we have

$$\begin{aligned} T^* &\geq (s - \frac{Ms}{\lfloor N/s \rfloor}) \frac{1}{1+\alpha_{\max}} \\ &\overset{(a)}{\geq} \frac{1}{12} \cdot K\Big(1 - \frac{M}{N}\Big) \frac{1}{1+t} \cdot \frac{1}{1+\alpha_{\max}}, \end{aligned} \tag{A9}$$

  where (a) follows from [1] [Theorem 3]. Then we have

$$\begin{aligned} \frac{T_{\text{central}}}{T^*} &\leq 12 \cdot \frac{(1+\alpha_{\max})(1+t)}{1+t+\alpha t} \\ &= 12 \cdot \frac{(1+\alpha_{\max})}{1+\alpha t/(1+t)} \\ &\leq 12 \cdot \frac{(1+\alpha_{\max})}{1+\alpha \cdot 0.6393/(1+0.6393)} \\ &\leq 31, \end{aligned} \tag{A10}$$

  where the last inequality holds by setting $\alpha = \alpha_{\max}$.
- If $t > \lfloor K/\alpha \rfloor - 1$, we have

$$\begin{aligned} \frac{T_{\text{central}}}{T^*} &\leq \frac{K(1-\frac{M}{N})\frac{1}{1+t+\alpha(\lfloor K/\alpha \rfloor - 1)}}{\frac{1}{2}(1-\frac{M}{N})} \\ &= \frac{2K}{1+t+\alpha(\lfloor K/\alpha \rfloor - 1)} \\ &\overset{(a)}{\leq} \frac{2K}{K+KM/N} \\ &\leq 2, \end{aligned} \tag{A11}$$

  where (a) holds by choosing $\alpha = 1$.
- If $t \leq 0.6393$, setting $s = 0.275N$, we have

$$\begin{aligned} T^* &\geq s - \frac{KM}{\lfloor N/s \rfloor} \\ &\overset{(a)}{\geq} s - \frac{KM}{N/s - 1} \\ &= 0.275N - t \cdot 0.3793N \\ &\geq 0.0325N > \frac{1}{31} \cdot N, \end{aligned} \tag{A12}$$

  where (a) holds since $\lfloor x \rfloor \geq x - 1$ for any $x \geq 1$. Please note that for all values of M, the transmission delay

$$T_{\text{central}} \leq \min\{K, N\}. \tag{A13}$$

  Combining with (A12) and (A13), we have $\frac{T_{\text{central}}}{T^*} \leq 31$.

Appendix C

Appendix C.1. Case $\alpha_{\max} = \lfloor \frac{K}{2} \rfloor$

When $\alpha_{\max} = \lfloor \frac{K}{2} \rfloor$, we have

$$R_{\text{u}} = R_{\text{u-f}} \triangleq \sum_{s=2}^{K} \frac{K\binom{K-1}{s-1}}{f(K,s)} p^{s-1} q^{K-s+1}, \tag{A14}$$

where $R_{\text{u-f}}$ denotes the user's transmission rate for a flexible D2D network with $\alpha_{\max} = \lfloor \frac{K}{2} \rfloor$. In the flexible D2D network, at most $\lfloor \frac{K}{2} \rfloor$ users are allowed to transmit messages simultaneously, in which the user transmission turns to unicast.

Please note that in each term of the summation:

$$\begin{aligned} \frac{K\binom{K-1}{s-1}}{f(K,s)} &\le \frac{K\binom{K-1}{s-1}}{K-1-\frac{K}{s}} \\ &= \left(\frac{K}{K-1} + \frac{\left(\frac{K}{K-1}\right)^2}{s-\frac{K}{K-1}}\right) \cdot \binom{K-1}{s-1} \\ &\le \frac{K\binom{K-1}{s-1}}{K-1} + \frac{2K\binom{K}{s}}{(K-1)(K-2)}, \end{aligned} \tag{A15}$$

where the last inequality holds by $s \ge \frac{K}{K-1} + \frac{K-2}{K-1} = 2$ and

$$\begin{aligned} \frac{\left(\frac{K}{K-1}\right)^2}{s-\frac{K}{K-1}} \binom{K-1}{s-1} &= \frac{K^2\binom{K-1}{s-1}}{(K-1)(K-2)} \cdot \frac{\frac{K-2}{K-1}}{s-\frac{K}{K-1}} \\ &\le \frac{K^2\binom{K-1}{s-1}}{(K-1)(K-2)} \cdot \frac{\frac{K-2}{K-1}+\frac{K}{K-1}}{s-\frac{K}{K-1}+\frac{K}{K-1}} \\ &= \frac{2K}{(K-1)(K-2)} \cdot \binom{K}{s}. \end{aligned}$$

Therefore, by (A15), $R_{\text{u-f}}$ can be rewritten as

$$\begin{aligned} R_{\text{u-f}} &\le \frac{K}{K-1} \sum_{s=2}^{K} \binom{K-1}{s-1} p^{s-1} q^{K-s+1} + \frac{2K}{(K-1)(K-2)} \sum_{s=2}^{K} \binom{K}{s} p^{s-1} q^{K-s+1} \\ &= \frac{Kq}{K-1} \cdot \sum_{i=1}^{K-1} \binom{K-1}{i} p^{i} q^{K-1-i} + \frac{2Kq/p}{(K-1)(K-2)} \cdot \sum_{s=2}^{K} \binom{K}{s} p^{s} q^{K-s} \\ &= \frac{Kq}{K-1}\left(1-q^{K-1}\right) + \frac{2Kq/p}{(K-1)(K-2)} \cdot \left(1-q^{K}-Kpq^{K-1}\right). \end{aligned}$$

Appendix C.2. Case $\alpha_{\max} = 1$

When $\alpha_{\max} = 1$, the cooperation network degenerates into a shared link where only one user acts as the server and broadcasts messages to the remaining $K-1$ users. A similar derivation is given in [35]. In this case, R_{u} can be rewritten as

$$\begin{aligned}
R_{\mathrm{u}} &= \sum_{s=2}^{K} \frac{s\binom{K}{s}}{s-1} p^{s-1} q^{K-s+1} \\
&\leq \sum_{s=2}^{K} \left(1+\frac{3}{s+1}\right)\binom{K}{s} p^{s-1} q^{K-s+1} \\
&= \sum_{s=2}^{K} \binom{K}{s} p^{s-1} q^{K-s+1} + \frac{3}{K+1}\sum_{s=2}^{K}\binom{K+1}{s+1} p^{s-1} q^{K-s+1} \\
&= \frac{q}{p}\left(1-q^{K}-Kpq^{K-1}\right) + \frac{3q/p^2}{K+1}\left(1-q^{K+1}-(K+1)pq^{K}-\frac{K(K+1)}{2}p^2q^{K-1}\right) \\
&= \frac{q}{p}\left(1-\frac{5}{2}Kpq^{K-1}-4q^{K}+\frac{3(1-q^{K+1})}{(K+1)p}\right),
\end{aligned}$$

where the inequality holds by the fact that $s \geq 2$.

Appendix D

Appendix D.1. When $\alpha_{\max} = \lfloor \frac{K}{2} \rfloor$

Recall that $p_{\mathrm{th}} \triangleq 1-\left(\frac{1}{K+1}\right)^{\frac{1}{K-1}}$, which tends to zero as K goes to infinity. We first introduce the following three lemmas.

Lemma A1. *Given arbitrary convex function $g_1(p)$ and arbitrary concave function $g_2(p)$, if they intersect at two points with $p_1 < p_2$, then $g_1(p) \leq g_2(p)$ for all $p \in [p_1, p_2]$.*

We omit the proof of Lemma A1 as it is straightforward.

Lemma A2. *For memory size $0 \leq p \leq 1$ and $\alpha_{\max} = \lfloor \frac{K}{2} \rfloor$, we have*

$$R_{\mathrm{u}} \geq R_{\emptyset},\ T_{\mathrm{decentral}} = \frac{R_{\mathrm{s}}R_{\mathrm{u}}}{R_{\mathrm{s}}+R_{\mathrm{u}}-R_{\emptyset}},\ \textit{for all } p \in [p_{\mathrm{th}}, 1].$$

Proof. When $\alpha_{\max} = \lfloor \frac{K}{2} \rfloor$, from Equation (20), we have

$$\begin{aligned}
R_{\mathrm{u}} &= \sum_{s=2}^{K} \frac{K\binom{K-1}{s-1}}{f(K,s)} p^{s-1}(1-p)^{K-s+1} \\
&\geq \frac{K}{K}\sum_{x=1}^{K-1}\binom{K-1}{x} p^{x}(1-p)^{K-x} \\
&= (1-p)\left(1-(1-p)^{K-1}\right), \qquad \text{(A16)}
\end{aligned}$$

where the first inequality holds by letting $x = s-1$ and $\frac{K}{K-1-\lfloor \frac{K}{s} \rfloor} > \frac{K}{K-1}$. It is easy to show that $(1-p)\left(1-(1-p)^{K-1}\right)$ is a concave function of p by verifying $\frac{\partial^2 (1-p)(1-(1-p)^{K-1})}{\partial p^2} \leq 0$. □

On the other hand, one can easily show that

$$R_{\emptyset} = K(1-p)^{K}$$

is a convex function of p by showing $\frac{\partial^2 R_\emptyset(p)}{\partial p^2} \geq 0$. Since the two functions $(1-p)\big(1-(1-p)^{K-1}\big)$ and $R_\emptyset$ intersect at $p_{\text{th}} = 1 - \left(\frac{1}{K+1}\right)^{\frac{1}{K-1}}$ and $p_2 = 1$ with $p_{\text{th}} \leq p_2$, from Lemma A1 and (A16), we have

$$R_{\text{u}} \geq (1-p)\big(1-(1-p)^{K-1}\big) \geq R_\emptyset,$$

for all $p \in [p_{\text{th}}, 1]$. From Remark 4, we know that $T_{\text{decentral}} = \frac{R_{\text{s}}R_{\text{u}}}{R_{\text{s}}+R_{\text{u}}-R_\emptyset}$ if $R_{\text{u}} \geq R_\emptyset$

Lemma A3. *For memory size* $0 \leq p \leq 1$ *and* $\alpha_{\max} = \lfloor \frac{K}{2} \rfloor$, *we have*

$$\frac{R_{\text{s}}R_{\text{u}}}{R_{\text{s}}+R_{\text{u}}-R_\emptyset} \leq 6T^*.$$

Proof. From (25) and (19), we have

$$\begin{aligned}
R_{\text{u}} &\leq \frac{K}{K-1}\cdot\Big(q-q^K\Big) + \frac{2K}{(K-1)(K-2)}\cdot\frac{q}{p}\Big(1-q^K-Kpq^{K-1}\Big) \\
&\overset{(a)}{\leq} \frac{K}{K-1}\cdot\Big(q-q^K\Big) + \frac{2K}{(K-1)(K-2)}\cdot\frac{q}{p}\Big(1-(1-Kp)-Kpq^{K-1}\Big) \\
&= \frac{K(3K-2)}{(K-1)(K-2)}\cdot\Big(q-q^K\Big),
\end{aligned} \tag{A17}$$

$$R_{\text{s}} = \frac{q}{p}\Big(1-q^K\Big) \overset{(b)}{\leq} \frac{q}{p}\Big(1-(1-Kp)\Big) = Kq, \tag{A18}$$

where (a) and (b) both follow from inequality

$$(1-p)^K \geq (1-Kp). \tag{A19}$$

□

Then, by Remark 4 and (A17), (A18) and definition of $R_\emptyset$ in (18), if $\alpha_{\max} = \lfloor \frac{K}{2} \rfloor$, then

$$\begin{aligned}
\frac{R_{\text{s}}R_{\text{u}}}{R_{\text{s}}+R_{\text{u}}-R_\emptyset} &\leq \frac{Kq\cdot\frac{K(3K-2)}{(K-1)(K-2)}\left(q-q^K\right)}{Kq+\frac{K(3K-2)}{(K-1)(K-2)}\left(q-q^K\right)-Kq^K} \\
&= \left(3-\frac{2}{K}\right)\cdot q.
\end{aligned} \tag{A20}$$

From Theorem 1, we have $T^* \geq \frac{1}{2}q$. Thus, we obtain

$$\frac{R_{\text{s}}R_{\text{u}}}{R_{\text{s}}+R_{\text{u}}-R_\emptyset}\cdot\frac{1}{T^*} \leq \frac{\Big(3-2/K\Big)\cdot q}{q/2} \leq 6-\frac{4}{K} < 6.$$

Next, we use Lemmas A2 and A3 to prove that when $\alpha_{\max} = \lfloor \frac{K}{2} \rfloor$,

$$\frac{T_{\text{decentral}}}{T^*} \leq \begin{cases} \max\left\{6, 2K\left(\frac{2K}{2K+1}\right)^{K-1}\right\}, & p < p_{\text{th}}, \\ 6, & p \geq p_{\text{th}}. \end{cases}$$

Appendix D.1.1. Case $\alpha_{\max} = \lfloor \frac{K}{2} \rfloor$ and $p \geq p_{\text{th}}$

In this case, from Lemma A2, we have

$$T_{\text{decentral}} = \frac{R_{\text{s}}R_{\text{u}}}{R_{\text{s}}+R_{\text{u}}-R_\emptyset}.$$

Thus, from Lemma A3,

$$T_{\text{decentral}} = \frac{R_{\text{s}}R_{\text{u}}}{R_{\text{s}} + R_{\text{u}} - R_{\emptyset}} \leq 6T^*.$$

Appendix D.1.2. Case $\alpha_{\max} = \lfloor \frac{K}{2} \rfloor$ and $p \leq p_{\text{th}}$

From the definition of $T_{\text{decentral}}$ in (17), we have

$$\frac{T_{\text{decentral}}}{T^*} = \max\{\frac{R_{\emptyset}}{T^*}, \frac{R_{\text{s}}R_{\text{u}}}{R_{\text{s}} + R_{\text{u}} - R_{\emptyset}} \cdot \frac{1}{T^*}\}. \tag{A21}$$

From Lemma A3, we know that

$$\frac{R_{\text{s}}R_{\text{u}}}{R_{\text{s}} + R_{\text{u}} - R_{\emptyset}} \cdot \frac{1}{T^*} \leq 6, \tag{A22}$$

and thus only focus on the upper bound of $R_{\emptyset}/T^*$.

According to Theorem 1, T^* has the following two lower bounds: $T^* \geq \frac{1-p}{2}$, and

$$T^* \geq \max_{s\in[K]} \left(s - \frac{KM}{\lfloor N/s \rfloor}\right) \geq \max_{s\in[K]} \left(s - \frac{KM}{N/(2s)}\right).$$

Let $R_1^*(p) \triangleq \frac{1}{2}(1-p)$ and $R_2^*(p) \triangleq (K - 2K^2p)$, then we have

$$T^* \geq \max\{R_1^*(p), R_2^*(p)\}.$$

Here $R_{\emptyset}/R_1^*(p)$ and $R_{\emptyset}/R_2^*(p)$ both are monotonic functions of p according to the following properties:

$$\begin{aligned}
\frac{\partial\big(R_{\emptyset}/R_1^*(p)\big)}{\partial p} &= \frac{\partial\big(2K(1-p)^{K-1}\big)}{\partial p} \leq 0, \\
\frac{\partial\big(R_{\emptyset}/R_2^*(p)\big)}{\partial p} &= \frac{\partial\big(q^K/(1-2Kp)\big)}{\partial p} \\
&= \frac{Kq^{K-1}(1+2(K-1)p)}{(1-2Kp)^2} \geq 0.
\end{aligned}$$

Additionally, notice that if $p = 0$, then $\frac{R_{\emptyset}}{R_2^*(p)} = 1 < \frac{R_{\emptyset}}{R_1^*(p)}$, and if $p = 1$, $\frac{R_{\emptyset}}{R_2^*(p)} > \frac{R_{\emptyset}}{R_1^*(p)} = 1$. Therefore, the maximum value of $R_{\emptyset}/\max\{R_1^*, R_2^*\}$ is chosen at $p = \frac{1}{2K+1}$ which satisfying $R_1^*(\frac{1}{2K+1}) = R_2^*(\frac{1}{2K+1})$, implying that

$$\frac{R_{\emptyset}}{T^*} \leq \frac{R_{\emptyset}(\frac{1}{2K+1})}{R_1^*(\frac{1}{2K+1})} = 2K\left(\frac{2K}{2K+1}\right)^{K-1}. \tag{A23}$$

From (A21)–(A23), we obtain the following equality:

$$\frac{T_{\text{decentral}}}{T^*} \leq \max\left\{2K\left(\frac{2K}{2K+1}\right)^{K-1}, 6\right\}.$$

Appendix D.2. When $\alpha_{\max} = 1$

From Equation (24), we obtain that

$$\begin{aligned} R_{\mathrm{u}} &\leq \frac{q}{p}\left(1 - \frac{5}{2}Kpq^{K-1} - 4q^K + \frac{3(1-q^{K+1})}{(K+1)p}\right) \\ &\leq \frac{q}{p}\left(1 - \frac{5}{2}Kpq^{K-1} - 4q^K + \frac{3(K+1)p}{(K+1)p}\right) \\ &= \frac{q}{p}\left(4\cdot(1-q^K) - \frac{5}{2}Kpq^{K-1}\right) \\ &\leq \frac{q}{p}(4\cdot(1-q^K)) \\ &= 4R_{\mathrm{s}}, \end{aligned} \tag{A24}$$

where the second inequality holds by (A19) and the last equality holds by the definition $R_s \triangleq \frac{q}{p}(1-q^K)$ in (19). On the other hand, rewrite the second lower bound of T^*:

$$T^* \geq \max_{s\in[K]}\left(s - \frac{sM}{\lfloor N/s \rfloor}\right)\frac{1}{1+\alpha_{\max}}. \tag{A25}$$

From the result in [2] (Appendix B), we have

$$\frac{R_{\mathrm{s}}}{\max_{s\in[K]}\left(s - \frac{sM}{\lfloor N/s \rfloor}\right)} \leq 12. \tag{A26}$$

Combining (A24)–(A26), we have

$$\frac{R_{\mathrm{s}}}{T^*} \leq 12(1+\alpha_{\max}),\ \frac{R_{\mathrm{u}}}{T^*} \leq 48(1+\alpha_{\max}). \tag{A27}$$

If $p \leq p_{\mathrm{th}}$, by (A27) and since $R_{\emptyset} \leq T_{\mathrm{decentral}} \leq R_{\mathrm{s}}$ (see Remark 4), we have

$$\frac{T_{\mathrm{decentral}}}{T^*} \leq \frac{R_{\mathrm{s}}}{T^*} \leq 12(1+\alpha_{\max}) = 24, \tag{A28}$$

the last equality holds by the fact $\alpha_{\max} = 1$.

If $p \geq p_{\mathrm{th}}$, from Lemma A2, we have $R_{\mathrm{u}} \geq R_{\emptyset}$ and

$$\begin{aligned} \frac{T_{\mathrm{decentral}}}{T^*} &= \frac{\frac{R_{\mathrm{s}}R_{\mathrm{u}}}{R_{\mathrm{s}}+R_{\mathrm{u}}-R_{\emptyset}}}{T^*} \\ &\leq \frac{\min\{R_{\mathrm{u}}, R_{\mathrm{s}}\}}{T^*} \\ &\leq \min\{12(1+\alpha_{\max}), 48(1+\alpha_{\max})\} \\ &= 24, \end{aligned} \tag{A29}$$

where the second inequality holds by (A27) and the last equality is from the fact $\alpha_{\max} = 1$ in this case.

References

1. Maddah-Ali, M.A.; Niesen, U. Fundamental limits of caching. *IEEE Trans. Inf. Theory* **2014**, *60*, 2856–1867.
2. Maddah-Ali, M.A.; Niesen, U. Decentralized coded caching attains order-optimal memory-rate tradeoff. *IEEE/ACM Trans. Netw.* **2015**, *23*, 1029–1040.
3. Yu, Q.; Maddah-Ali, M.A.; Avestimehr, A.S. Characterizing the Rate-Memory Tradeoff in Cache Networks Within a Factor of 2. *IEEE Trans. Inf. Theory* **2019**, *65*, 647–663.
4. Wan, K.; Tuninetti, D.; Piantanida, P. On the optimality of uncoded cache placement. In Proceedings of the IEEE Information Theory Workshop (ITW), Cambridge, UK, 11–14 September 2016; pp. 161–165.

5. Yu, Q.; Maddah-Ali, M.A.; Avestimehr, A.S. The exact rate-memory tradeoff for caching with uncoded prefetching. *IEEE Trans. Inf. Theory* **2018**, *64*, 1281–1296.
6. Yan, Q.; Cheng, M.; Tang, X.; Chen, Q. On the placement delivery array design for centralized coded caching scheme. *IEEE Trans. Inf. Theory* **2017**, *63*, 5821–5833.
7. Zhang, D.; Liu, N. Coded cache placement for heterogeneous cache sizes. In Proceedings of the IEEE Information Theory Workshop (ITW), Guangzhou, China, 25–29 November 2018; pp. 1–5.
8. Wang, S.; Peleato, B. Coded caching with heterogeneous user profiles. In Proceedings of the IEEE International Symposium on Information Theory (ISIT), France, Paris, 7–12 July 2019; pp. 2619–2623.
9. Zhang, J.; Lin, X.; Wang, C.C. Coded caching for files with distinct file sizes. In Proceedings of the IEEE International Symposium on Information Theory (ISIT), Hong Kong, China, 14–19 June 2015; pp. 1686–1690.
10. Ibrahim, A.M.; Zewail, A.A.; Yener, A. Centralized coded caching with heterogeneous cache sizes. In Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC), San Francisco, CA, USA, 19–22 March 2017; pp. 1–6.
11. Ibrahim, A.M.; Zewail, A.A.; Yener, A. Coded caching for heterogeneous systems: An Optimization Perspective. *IEEE Trans. Commun.* **2019**, *67*, 5321–5335.
12. Amiri, M.M.; Yang, Q.; Gündüz, D. Decentralized caching and coded delivery with distinct cache capacities. *IEEE Trans. Commun.* **2017**, *65*, 4657–4669.
13. Cao, D.; Zhang, D.; Chen, P.; Liu, N.; Kang, W.; Gündüz, D. Coded caching with asymmetric cache sizes and link qualities: The two-user case. *IEEE Trans. Commun.* **2019**, *67*, 6112–6126.
14. Niesen, U.; Maddah-Ali, M.A. Coded caching with nonuniform demands. *IEEE Trans. Inf. Theory* **2017**, *63*, 1146–1158.
15. Zhang, J.; Lin, X.; Wang, X. Coded caching under arbitrary popularity distributions. *IEEE Trans. Inf. Theory* **2018**, *64*, 349–366.
16. Pedarsani, R.; Maddah-Ali, M.A.; Niesen, U. Online coded caching. *IEEE/ACM Trans. Netw.* **2016**, *24*, 836–845.
17. Daniel, A.M.; Yu, W. Optimization of heterogeneous coded caching. *IEEE Trans. Inf. Theory* **2020**, *66*, 1893–1919.
18. Shariatpanahi, S.P.; Motahari, S.A.; Khalaj, B.H. Multi-server coded caching. *IEEE Trans. Inf. Theory* **2016**, *62*, 7253–7271.
19. Zhang, J.; Elia, P. Fundamental limits of cache-aided wireless BC: Interplay of coded-caching and CSIT feedback. *IEEE Trans. Inf. Theory* **2017**, *63*, 3142–3160.
20. Bidokhti, S.S.; Wigger, M.; Timo, R. Noisy broadcast networks with receiver caching. *IEEE Trans. Inf. Theory* **2018**, *64*, 6996–7016.
21. Sengupta, A.; Tandon, R.; Simeone, O. Cache aided wireless networks: Tradeoffs between storage and latency. In Proceedings of the 2016 Annual Conference on Information Science and Systems (CISS), Princeton, NJ, USA, 15–18 March 2016; pp. 320–325.
22. Tandon, R.; Simeone, O. Cloud-aided wireless networks with edge caching: Fundamental latency trade-offs in fog radio access networks. In Proceedings of the IEEE International Symposium on Information Theory (ISIT), Barcelona, Spain, 10–15 July 2016; pp. 2029–2033.
23. Karamchandani, N.; Niesen, U.; Maddah-Ali, M.A.; Diggavi, S.N. Hierarchical coded caching. *IEEE Trans. Inf. Theory* **2016**, *62*, 3212–3229.
24. Wang, K.; Wu, Y.; Chen, J.; Yin, H. Reduce transmission delay for caching-aided two-layer networks. In Proceedings of the IEEE International Symposium on Information Theory (ISIT), France, Paris, 7–12 July 2019; pp. 2019–2023.
25. Wan, K.; Ji, M.; Piantanida, P.; Tuninetti, D. Caching in combination networks: Novel multicast message generation and delivery by leveraging the network topology. In Proceedings of the IEEE International Conference on Communications (ICC), Kansas City, MO, USA, 20–24 May 2018; pp. 1–6.
26. Naderializadeh, N.; Maddah-Ali, M.A.; Avestimehr, A.S. Fundamental limits of cache-aided interference management. *IEEE Trans. Inf. Theory* **2017**, *63*, 3092–3107.
27. Xu, F.; Tao, M.; Liu, K. Fundamental tradeoff between storage and latency in cache-aided wireless interference Networks. *IEEE Trans. Inf. Theory* **2017**, *63*, 7464–7491.
28. Ji, M.; Tulino, A.M.; Llorca, J.; Caire, G. Order-optimal rate of caching and coded multicasting with random demands. *IEEE Trans. Inf. Theory* **2017**, *63*, 3923–3949.
29. Ji, M.; Tulino, A.M.; Llorca, J.; Caire, G. Caching in combination networks. In Proceedings of the 2015 49th Asilomar Conference on Signals, Systems and Computers, Pacific Grove, CA, USA, 8–11 November 2015.
30. Ravindrakumar, V.; Panda, P.; Karamchandani, N.; Prabhakaran, V. Fundamental limits of secretive coded caching. In Proceedings of the IEEE International Symposium on Information Theory (ISIT), Barcelona, Spain, 10–15 July 2016; pp. 425–429.
31. Tang, L.; Ramamoorthy, A. Coded caching schemes with reduced subpacketization from linear block codes. *IEEE Trans. Inf. Theory* **2018**, *64*, 3099–3120.
32. Cheng, M.; Li, J.; Tang, X.; Wei, R. Linear coded caching scheme for centralized networks. *IEEE Trans. Inf. Theory* **2021**, *67*, 1732–1742.
33. Wan, K.; Caire, G. On coded caching with private demands. *IEEE Trans. Inf. Theory* **2021**, *67*, 358–372.
34. Hassanzadeh, P.; Tulino, A.M.; Llorca, J.; Erkip, E. Rate-memory trade-off for caching and delivery of correlated sources. *IEEE Trans. Inf. Theory* **2020**, *66*, 2219–2251.
35. Ji, M.; Caire, G.; Molisch, A.F. Fundamental limits of caching in wireless D2D networks. *IEEE Trans. Inf. Theory* **2016**, *62*, 849–869.
36. Tebbi, A.; Sung, C.W. Coded caching in partially cooperative D2D communication networks. In Proceedings of the 9th International Congress on Ultra Modern Telecommunications and Control Systems and Workshops (ICUMT), Munich, Germany, 6–8 November 2017; pp. 148–153.

37. Wang, J.; Cheng, M.; Yan, Q.; Tang, X. Placement delivery array design for coded caching scheme in D2D Networks. *IEEE Trans. Commun.* **2019**, *67*, 3388–3395.
38. Malak, D.; Al-Shalash, M.; Andrews, J.G. Spatially correlated content caching for device-to-device communications. *IEEE Trans. Wirel. Commun.* **2018**, *17*, 56–70.
39. Ibrahim, A.M.; Zewail, A.A.; Yener, A. Device-to-Device coded caching with distinct cache sizes. *arXiv* **2019**, arXiv:1903.08142.
40. Pedersen, J.; Amat, A.G.; Andriyanova, I.; Brännström, F. Optimizing MDS coded caching in wireless networks with device-to-device communication. *IEEE Trans. Wirel. Commun.* **2019**, *18*, 286–295.
41. Chiang, M.; Zhang, T. Fog and IoT: An overview of research opportunities. *IEEE Internet Things J.* **2016**, *3*, 854–864.