entropy

MDPI

Article

Hierarchical Cache-Aided Networks for Linear Function Retrieval

Lingyu Zhang [1], Yun Kong [2], Youlong Wu [3] and Minquan Cheng [1,*]

[1] Guangxi Key Laboratory of Multi-Source Information Mining & Security, Guangxi Normal University, Guilin 541004, China; zhangmm@stu.gxnu.edu.cn
[2] The Department of Electrical Engineering, University of North Texas, Denton, TX 76207, USA; yunkong@my.unt.edu
[3] The School of Information Science and Technology, ShanghaiTech University, Shanghai 201210, China; wuyl1@shanghaitech.edu.cn
* Correspondence: mqcheng@gxnu.edu.cn

Abstract: In a hierarchical caching system, a server is connected to multiple mirrors, each of which is connected to a different set of users, and both the mirrors and the users are equipped with caching memories. All the existing schemes focus on single file retrieval, i.e., each user requests one file. In this paper, we consider the linear function retrieval problem, i.e., each user requests a linear combination of files, which includes single file retrieval as a special case. We propose a new scheme that reduces the transmission load of the first hop by jointly utilizing the two layers' cache memories, and we show that our scheme achieves the optimal load for the second hop in some cases.

Keywords: linear function retrieval; hierarchical coded caching scheme; transmission load

check for updates

Citation: Zhang, L.; Kong, Y.; Wu, Y.; Cheng, M. Hierarchical Cache-Aided Networks for Linear Function Retrieval. *Entropy* **2024**, *26*, 195. https://doi.org/10.3390/e26030195

Academic Editor: Gergely Palla

Received: 23 January 2024
Revised: 14 February 2024
Accepted: 19 February 2024
Published: 25 February 2024

1. Introduction

In order to reduce the transmission pressure of wireless networks during peak traffic times, Maddah-Ali and Niesen in [1] provided a (K, M, N) coded caching scheme (MN Scheme) where a single server has N files and connects K cache-aided users with the cache memories of M files through an error-free shared link. A coded caching scheme consists of two phases: (1) the placement phase, where the server is equipped with the data and each user's cache is also equipped with the size of at most M files without knowledge of the users' future demands; (2) the delivery phase, where each user randomly requests one file and then the server sends the coded signal to the users such that each user can decode its requested file with the help of its cached packets. It is shown that the MN Scheme is generally order-optimal within a factor of 2 [2] and optimal under the uncoded data placement when $K \leq N$ [3]. The MN Scheme is also widely used in different networks, such as combination networks [4,5], device-to-device networks [6], etc.

In practical scenarios, caching systems are transformed into multiple layers in order to make transmission more efficient, such as the hierarchical edge caching architecture for Internet of Vehicles [7], the three-tier mobile cloud-edge computing structure [8], and so on. In this paper, we particularly study the hierarchical caching system [9], a two-layer network as illustrated in Figure 1. A $(K_1, K_2; M_1, M_2; N)$ hierarchical caching system consists of a single server with a library of N files, K_1 cache-aided mirror sites, and K_1K_2 cache-aided users. For the first layer, the K_1 mirror sites are connected to the server through an error-free shared link, and for the second layer, each user connects to only one mirror. Our goal is to design a scheme to decrease the first load R_1 in the first hop (i.e., from the server to all the mirror sites) and the second load R_2 in the second hop (i.e., from each mirror site to its connected users).

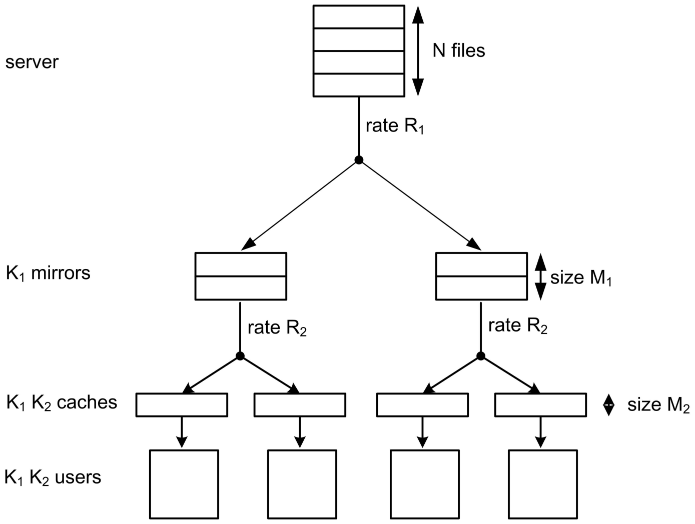

Figure 1. The $(K_1, K_2; M_1, M_2; N)$ hierarchical caching system with $N = 4$, $K_1 = K_2 = 2$, $M_1 = 2$, and $M_2 = 1$.

The authors in [9] proposed the first hierarchical coded caching scheme (KNMD Scheme). The MN Scheme is applied two times in two layers consecutively. Although the KNMD Scheme achieves the optimal transmission load for the second hop, it involves a significant increase in R_1 since it ignores the users' cache memory when designing the multicast message sent from the server. To improve the first load R_1, the authors in [10,11] proposed new schemes that jointly use the two types of the MN Scheme together for the mirror sites and users, respectively.

It is worth noting that all the schemes consider the single file retrieval case, i.e., each user requests one file. The authors in [12] first considered the linear function retrieval scheme (WSJT Scheme), i.e., a linear combination of files is requested from each user through the shared link broadcast network. Clearly, linear function retrieval includes the single file retrieval case. In this paper, we study the linear function retrieval scheme for hierarchical networks and obtain the following results.

- We first propose a baseline scheme via the WSJT Scheme and KNMD Scheme where the second-layer load achieves the optimal transmission load. However, we achieve this by sacrificing the first-layer load.
- Then, in order to reduce the first-layer load, we propose another scheme whose second load also achieves optimality at the expense of increased subpacketization. Our scheme also aids in reducing the redundancy for some special demand distributions.

The rest of this paper is organized as follows. Section 2 formally introduces the system model and some existing schemes. Section 3 presents the main results. Section 4 gives an example and the general description of our scheme, i.e., the scheme for Theorem 2. The conclusion of this paper is given in Section 5.

Notations: For any positive integers a and b with $a < b$, let $[a : b] \triangleq \{a, \ldots, b\}$ and $[a] \triangleq [1 : a]$. Let $\binom{[b]}{t} \triangleq \{\mathcal{V} | \mathcal{V} \subseteq [b], |\mathcal{V}| = t\}$, for any positive integer $t \leq b$. For a positive integer n, the n-dimensional vector space over the field $\mathbb{F}_q$ is denoted by $\mathbb{F}_q^n$. For a given matrix $\mathbf{P}$ with row size X_1, we divide it into X_1 parts by row, which is represented by $\mathbf{P} = \{\mathbf{P}^{(x_1)} | \, x_1 \in [X_1]\}$. For any integer set $\mathcal{T}$, define $\mathbf{P}_{\mathcal{T}}$ as the sub-matrix of $\mathbf{P}$ by selecting some rows from $\mathbf{P}$, where the rows have indices in $\mathcal{T}$. The rank of matrix $\mathbf{P}$ is denoted as $\text{rank}(\mathbf{P})$. The transpose of $\mathbf{P}$ is represented by $\mathbf{P}^\top$.

2. Preliminary

In this section, we give a formal description of the hierarchical caching system and review some existing related schemes for the hierarchical caching problem.

2.1. System Model

Consider a hierarchical network as shown in Figure 1. It consists of a single server with a library of N files, K_1 cache-aided mirror sites, and K_1K_2 cache-aided users. For the first layer, the K_1 mirror sites are connected to the server through an error-free shared link, and for the second layer, each user connects to only one mirror. M_{k_1} represents the k_1-th mirror and the k_2-th user attached to M_{k_1} as U_{k_1,k_2}, $k_1 \in [K_1], k_2 \in [K_2]$, and the set of users attached to M_{k_1} as $\mathcal{U}_{k_1}$. The server contains a collection of N files, denoted by $\mathcal{W} = \{W^{(1)}, W^{(2)}, \ldots, W^{(N)}\}$, each of which is uniformly distributed over $\mathbb{F}_2^B$, where $B \in \mathbb{N}^+$. Each mirror and user is equipped with M_1 and M_2 files, respectively, where M_1, $M_2 \geq 0$. A $(K_1, K_2; M_1, M_2; N)$ hierarchical caching system contains two phases.

- **Placement phase:** The mirror site M_{k_1} caches some parts of the files by using a cache function $\varphi_{k_1} : \mathbb{F}_2^{NB} \to \mathbb{F}_2^{M_1B}$, where $\frac{M_1}{N}$ is the memory ratio of the mirror in the first layer, $M_1 \in [0 : N]$. The cache contents of mirror M_{k_1} are

$$\mathcal{Z}_{k_1} = \varphi_{k_1}(\mathcal{W}), k_1 \in [K_1].$$

  The user U_{k_1,k_2} caches some parts of the files by using a cache function $\phi_{k_1,k_2} : \mathbb{F}_2^{NB} \to \mathbb{F}_2^{M_2B}$, where $\frac{M_2}{N}$ is the memory ratio of users in the second layer, $M_2 \in [0 : N]$. Then, the cache contents of U_{k_1,k_2} are

$$\widetilde{\mathcal{Z}}_{k_1,k_2} = \phi_{k_1,k_2}(\mathcal{W}), k_1 \in [K_1], k_2 \in [K_2].$$

- **Delivery phase:** Each user U_{k_1,k_2} randomly requests a linear combination of the files

$$L_{k_1,k_2} = d_{k_1,k_2}^{(1)} W^{(1)} + d_{k_1,k_2}^{(2)} W^{(2)} + \ldots + d_{k_1,k_2}^{(N)} W^{(N)}.$$

  for any $k_1 \in [K_1]$, $k_2 \in [K_2]$, where $\mathbf{d}_{k_1,k_2} = (d_{k_1,k_2}^{(1)}, \ldots, d_{k_1,k_2}^{(N)}) \in \mathbb{F}_2^N$ denotes the demand vector of user U_{k_1,k_2}. When each user requests a single file, the demand vector d_{k_1,k_2} is a N-length unit vector. For example, if user U_{k_1,k_2} only requests the 1-st file, then the demand vector is set as $\mathbf{d}_{k_1,k_2} = (1, \ldots, 0) \in \mathbb{F}_2^N, k_1 \in [K_1], k_2 \in [K_2]$, which is a special case of our proposed scheme. We can obtain the demand matrices of all users as follows:

$$\mathbf{D}^{(k_1)} = \begin{pmatrix} \mathbf{d}_{k_1,1} \\ \vdots \\ \mathbf{d}_{k_1,K_2} \end{pmatrix}, \quad \mathbf{D} = \begin{pmatrix} \mathbf{D}^{(1)} \\ \vdots \\ \mathbf{D}^{(K_1)} \end{pmatrix}. \tag{1}$$

  where $\mathbf{D}^{(k_1)}$ represents the demand vectors of $\mathcal{U}_{k_1}$. Given the demand matrix $\mathbf{D}$, we should consider the following two types of messages.

  - **The messages sent by the server:** The server generates signal X^{server} by using an encoding function $\chi : \mathbb{F}_2^{K_1K_2N} \times \mathbb{F}_2^{NB} \to \mathbb{F}_2^{R_1B}$, where

$$X^{\text{server}} = \chi(\mathbf{D}, \mathcal{W}).$$

    and then the server sends X^{server} to the mirrors. The normalized number of transmissions R_1 is called the transmission load for the first layer.
  - **The messages sent by the mirror:** Based on X^{server}, $\mathcal{Z}_{k_1}$, and $\mathbf{D}$, each mirror M_{k_1} generates a signal $X_{k_1}^{\text{mirror}}$ by using the encoding function $\kappa : \mathbb{F}_2^{K_2N} \times \mathbb{F}_2^{M_1B} \times X^{\text{server}} \to \mathbb{F}_2^{R_2B}$, where

$$X_{k_1}^{\text{mirror}} = \kappa(\mathbf{D}^{(k_1)}, \mathcal{Z}_{k_1}, X^{\text{server}}).$$

and then mirror M_{k_1} sends $X_{k_1}^{\text{mirror}}$ to its connected users. The normalized number of transmissions R_2 is called the transmission load for the second layer.

For the retrieval process, each user U_{k_1,k_2} can decode its required linear combination of files from $(\mathbf{D}, \widetilde{\mathcal{Z}}_{k_1,k_2}, X_{k_1}^{\text{mirror}})$, which means that there exist decoding functions $\xi_{k_1,k_2} : \mathbb{F}_2^{K_1K_2N} \times \mathbb{F}_2^{M_2B} \times \mathbb{F}_2^{R_2B} \rightarrow \mathbb{F}_2^{B}, k_1 \in [K_1], k_2 \in [K_2]$, such that

$$\xi_{k_1,k_2}(\mathbf{D}, \widetilde{\mathcal{Z}}_{k_1,k_2}, X_{k_1}^{\text{mirror}}) = d_{k_1,k_2}^{(1)} W^{(1)} + \ldots + d_{k_1,k_2}^{(N)} W^{(N)}.$$

We define the optimal transmission loads for the two layers as R_1^* and R_2^* separately.

$$\begin{aligned} R_1^* &= \inf_{\chi,\kappa,(\xi_{k_1,k_2})_{k_1\in[K_1],k_2\in[K_2]}} \{R_1\}, \\ R_2^* &= \inf_{\chi,\kappa,(\xi_{k_1,k_2})_{k_1\in[K_1],k_2\in[K_2]}} \{R_2\}. \end{aligned}$$

Our goal is to design schemes in which the transmission loads R_1 and R_2 are as small as possible.

2.2. Existing Schemes

In the following, we review the KNMD Scheme for the hierarchical caching problem and the WSJT Scheme over the binary field $\mathbb{F}_2$, which will be useful for the hierarchical caching system with linear function retrieval. First, let us outline the MN Scheme.

(1) MN Scheme [1]: Set $t \triangleq MK/N$, when $t \in [0 : K]$, $N \geq K$, each file is partitioned into $F = \binom{K}{t}$ packets, i.e., for each $n \in [N]$, $W^{(n)} = (W_{\mathcal{T}}^{(n)})$, where $\mathcal{T} \in \binom{[K]}{t}$. In the placement phase, for each user U_k, $k \in [K]$. The cache content of user U_k is $\mathcal{Z}_k = \{W_{\mathcal{T}}^{(n)} | n \in [N], k \in \mathcal{T}, \mathcal{T} \in \binom{[K]}{t}\}$. In the delivery phase, the file W_{d_k} is requested by each user U_k, where $d_k \in [N]$. Fixing a user $k \in \mathcal{S}$, the user k requests the subfiles $W_{d_k,\mathcal{S}\setminus\{k\}}$ when it is presented in the cache of any user $k' \in \mathcal{S} \setminus \{k\}$. Then, the server transmits the coded signal $\bigoplus_{k\in\mathcal{S}} W_{d_k,\mathcal{S}\setminus\{k\}}$, where $\mathcal{S} \subseteq [K]$ of $|\mathcal{S}| = t + 1$. The transmission load $R_{\text{MN}} = \frac{K(1-M/N)}{KM/N+1}$.

(2) KNMD Scheme [9]: This scheme uses the MN Scheme in each layer of the hierarchical network. More specifically, for the first layer between the server and K_1 mirrors, it uses the (K_1, M_1, N) MN Scheme K_2 times to recover all K_1K_2 requested files, and then each mirror $\mathbf{M}_{k_1}$, $k_1 \in [K_1]$ works as a server whose library contains K_2 files that are requested by users in $\mathcal{U}_{k_1}$, and finally it utilizes the (K_2, M_2, N) MN Scheme between $\mathbf{M}_{k_1}$ and $\mathcal{U}_{k_1}$. Then, each user can retrieve its requested file with the transmission load as follows.

$$R_1 = K_2 \frac{K_1 - K_1M_1/N}{K_1M_1/N + 1}, \quad R_2 = \frac{K_2 - K_2M_2/N}{K_2M_2/N + 1}.$$

However, the MN Scheme only works for single file retrieval. The authors in [12] proposed a scheme (WSJT Scheme) that is suitable for the linear function retrieval problem.

(3) WSJT Scheme [12]: Using the placement strategy of the MN Scheme, each user U_k where $k \in [K]$ requests a linear combination of files with demand vector $\mathbf{d}_k \in \mathbb{F}_2^N$. After revealing the demand matrix $\mathbf{D} = (\mathbf{d}_1^\top, \ldots, \mathbf{d}_K^\top)^\top$ with dimension $K \times N$, the server broadcasts some coded packets by modifying the transmission strategy of the MN Scheme such that each user is able to recover its demanded linear combination of files with the transmission load

$$R_{\text{WSJT}} = \frac{\binom{K}{t+1} - \binom{K-\text{rank}(\mathbf{D})}{t+1}}{\binom{K}{t}}, \quad t \in [0 : K].$$

It is worth noting that when $\mathbf{D}$ is row full rank, R_{WSJT} is optimal under the uncoded placement.

3. Main Results

In this section, we first propose a baseline scheme via the WSJT Scheme where R_2 achieves optimality when the sub-matrix $\mathbf{D}^{(k_1)}$, $k_1 \in [K_1]$, is full rank. Then, we propose another scheme that improves R_1 while the R_2 remains unchanged compared with the Baseline Scheme. Finally, some theoretical and numerical comparisons are provided.

For the sake of convenience in proposing another scheme for some special demand distributions, the following definitions of the leader mirror and user sets are necessary.

Definition 1 (Leader mirror set)**.** *For a $K_1K_2 \times N$ demand matrix $\mathbf{D}$ in (1), we call a subset of mirrors the leader mirror set, which is represented by $\mathcal{L}_M = \{l_1, \ldots, l_{|\mathcal{L}_M|}\}$, $\mathcal{L}_M \subseteq [K_1]$, if it satisfies the following condition for $k_1 \in [K_1]$, $k_2 \in [K_2]$*

$$\mathbf{d}_{k_1,k_2} = \alpha_1^{(k_1)} \mathbf{d}_{l_1,k_2} + \ldots + \alpha_{|\mathcal{L}_M|}^{(k_1)} \mathbf{d}_{l_{|\mathcal{L}_M|},k_2}. \tag{2}$$

and it has the minimum cardinality among all the subsets satisfying (2), where $(\alpha_1^{(k_1)}, \ldots, \alpha_{|\mathcal{L}_M|}^{(k_1)}) \in \mathbb{F}_2^{|\mathcal{L}_M|}$.

Definition 2 (Leader user set)**.** *For a $K_2 \times N$ demand matrix $\mathbf{D}^{(k_1)}$ in (1), we call a subset of users the leader user set, which is represented by $\mathcal{L}_{k_1} = \{l'_1, \ldots, l'_{|\mathcal{L}_{k_1}|}\}$, $\mathcal{L}_{k_1} \subseteq [K_2]$, if, for any $k_1 \in [K_1]$, $k_2 \in [K_2]$, it satisfies the condition (3) and it has the minimum cardinality among all the subsets satisfying (3), where $(\alpha_1, \ldots, \alpha_{|\mathcal{L}_{k_1}|}) \in \mathbb{F}_2^{|\mathcal{L}_{k_1}|}$:*

$$\mathbf{d}_{k_1,k_2} = \alpha_1 \mathbf{d}_{k_1,1} + \ldots + \alpha_{|\mathcal{L}_{k_1}|} \mathbf{d}_{k_1,l'_{|\mathcal{L}_{k_1}|}}. \tag{3}$$

Now, we introduce the Baseline Scheme, which is generated by using the KNMD Scheme in [9] and the WSJT Scheme in [12]. We utilize the WSJTC Scheme to replace the MN Scheme in the KNMD Scheme, and then we obtain the Baseline Scheme, which is suitable for the linear function retrieval problem in the hierarchical network.

Theorem 1 (Baseline Scheme)**.** *For any positive integers K_1, K_2, $t_1 \in [K_1]$, $t_2 \in [K_2]$ and the demand matrix $\mathbf{D}$ in (1), there exists a $(K_1, K_2; M_1, M_2; N)$ hierarchical coded caching scheme for a linear function retrieval problem with memory ratios $\frac{M_1}{N} = \frac{t_1}{K_1}$, $\frac{M_2}{N} = \frac{t_2}{K_2}$ and transmission loads*

$$R_{base1} = K_2\left(\binom{K_1}{t_1+1} - \binom{K_1 - |\mathcal{L}_M|}{t_1+1}\right) / \binom{K_1}{t_1}.$$

$$R_{base2} = \max_{k_1 \in [K_1]} \left\{ \left(\binom{K_2}{t_2+1} - \binom{K_2 - rank(\mathbf{D}^{(k_1)})}{t_2+1}\right) / \binom{K_2}{t_2} \right\}.$$

where $\mathcal{L}_M$ is defined in Definition 1.

In fact, the transmission loads are related to the placement strategy and demand distribution, respectively. The KNMD Scheme considers the first and second layers separately and ignores the users' and mirrors' cache memories, which leads to good performance on R_2 but results in a large transmission load R_1. For the second layer, it can be regarded as a (K_2, M_2, N) shared link caching problem in which the WSJT Scheme achieves the optimal transmission load under certain circumstances, i.e., when the sub-matrix $\mathbf{D}^{(k_1)}$, $k_1 \in [K_1]$, is full rank, $R_{\text{base2}} = R_2^*$. For the purpose of improving R_1, we propose another scheme, stated below, and the proof is included in Section 4.

Theorem 2. *For any positive integers K_1, K_2, $t_1 \in [K_1]$, $t_2 \in [K_2]$ and the demand matrix* $\mathbf{D}$ *in* (1), *there exists a $(K_1, K_2; M_1, M_2; N)$ hierarchical coded caching scheme for a linear function retrieval problem with memory ratios $\frac{M_1}{N} = \frac{t_1}{K_1}$, $\frac{M_2}{N} = \frac{t_2}{K_2}$ and transmission loads*

$$\begin{aligned} R_1 &= \left(\binom{K_1}{t_1+1} - \binom{K_1 - |\mathcal{L}_M|}{t_1+1}\right)\binom{K_2}{t_2+1} \Big/ \binom{K_1}{t_1}\binom{K_2}{t_2}, \\ R_2 &= \max_{k_1 \in [K_1]} \left\{ \left(\binom{K_2}{t_2+1} - \binom{K_2 - rank(\mathbf{D}^{(k_1)})}{t_2+1}\right) \Big/ \binom{K_2}{t_2} \right\}. \end{aligned} \tag{4}$$

Now, let us consider the performance of our two schemes. For the first layer, we claim that $R_1 < \frac{1}{t_2+1} R_{\text{base1}}$, where $t_2 \geq 0$ since

$$\begin{aligned} \frac{R_1}{R_{\text{base1}}} &= \frac{\left(\binom{K_1}{t_1+1} - \binom{K_1-|\mathcal{L}_M|}{t_1+1}\right)\binom{K_2}{t_2+1}\binom{K_1}{t_1}}{K_2\left(\binom{K_1}{t_1+1} - \binom{K_1-|\mathcal{L}_M|}{t_1+1}\right)\binom{K_1}{t_1}\binom{K_2}{t_2}} = \frac{\binom{K_2}{t_2+1}}{K_2\binom{K_2}{t_2}} \\ &= \frac{K_2!\ t_2!(K_2-t_2)!}{K_2\ K_2!(t_2+1)!(K_2-t_2-1)!} \\ &= \frac{t_2!(K_2-t_2)(K_2-t_2-1)!}{K_2\ t_2!(t_2+1)(K_2-t_2-1)!} \\ &= \frac{K_2-t_2}{K_2} \cdot \frac{1}{t_2+1} \leq \frac{1}{t_2+1}. \end{aligned}$$

Obviously, this scheme has the same performance as the Baseline Scheme, i.e., $R_2 = R_{\text{base2}}$, which also achieves the optimal transmission load when the demand matrix $\mathbf{D}^{(k_1)}$, $k_1 \in [K_1]$, is full rank.

Finally, we perform a numerical comparison to further show the performance of our scheme. In Figure 2, we compare the Baseline Scheme with the scheme for Theorem 2 with fixed parameters $(K_1, K_2, |\mathcal{L}_M|, N) = (20, 10, 10, 200)$ and varying the memory ratio M_1/N from 0 to 1 with a step size 0.1. As seen in Figure 2, compared to the Baseline Scheme, the scheme for Theorem 2 can reduce the transmission load R_1 significantly, as this scheme utilizes both the user's cache and the mirror's cache when constructing the multicast message sent by the server. The scheme for Theorem 2 achieves the same R_2 as the Baseline Scheme, while our scheme has a lower transmission load R_1.

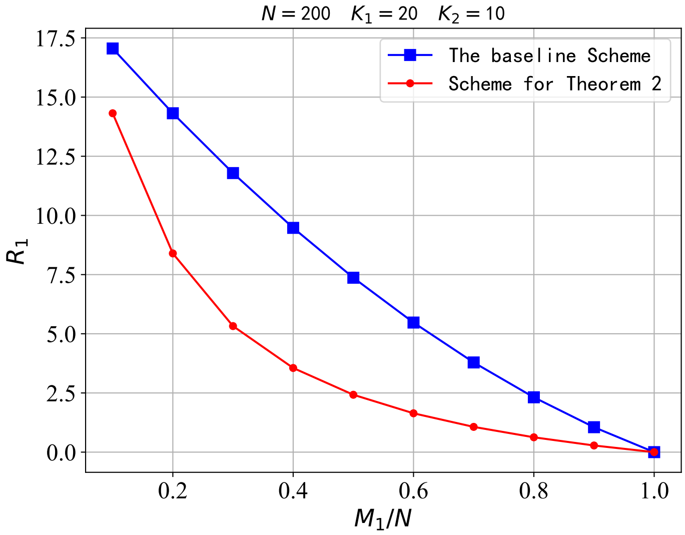

Figure 2. R_1 on $N = 200$, $K_1 = 20$, $K_2 = 10$.

4. Scheme for Theorem 2

In this section, we first give an illustrative example of our scheme. Then, the general description of the scheme is provided. Before the description, we first introduce the following lemmas regarding the message sent by the server and mirrors, whose proofs are included in Appendices A and B, respectively.

Lemma 1 (The messages sent by the server)**.** *Given a demand matrix* $\mathbf{D}$ *in* (1)*, the leader mirror set* $\mathcal{L}_M$*, and a user set* $\mathcal{B} \in \binom{[K_2]}{t_2+1}$*, if there exists a mirror set* $\mathcal{C} \in \binom{[K_1]}{|\mathcal{L}_M|+t_1+1}$*, where* $\mathcal{L}_M \subseteq \mathcal{C}$*, let* $\mathfrak{V}_{\mathcal{C}}$ *be the family of mirror set* $\mathcal{V}$*,* $\mathcal{V} \subseteq \mathcal{C}$*, where each* $\mathcal{V}$ *satisfies Definition 1. Then, we have* $\sum_{\mathcal{V}\in\mathfrak{V}_{\mathcal{C}}} X_{\mathcal{C}\setminus\mathcal{V},\mathcal{B}} = 0$*, where* $X_{\mathcal{C}\setminus\mathcal{V},\mathcal{B}}$ *represents the message sent by the server, which is defined in (8).*

Lemma 2 (The messages sent by the mirror)**.** *Given a sub-matrix* $\mathbf{D}^{(k_1)}$*,* $k_1 \in [K_1]$ *of* $\mathbf{D}$*, the leader user set* $\mathcal{L}_{k_1}$*, and a mirror set* $\mathcal{T}_1 \in \binom{[K_1]}{t_1}$*, if there exists a user set* $\mathcal{C}' \in \binom{[K_2]}{|\mathcal{L}_{k_1}|+t_2+1}$*, where* $\mathcal{L}_{k_1} \subseteq \mathcal{C}'$*, let* $\mathfrak{V}'_{\mathcal{C}'}$ *be the family of all set* $\mathcal{V}'$*,* $\mathcal{V}' \subseteq \mathcal{C}'$*, where each* $\mathcal{V}'$ *satisfies Definition 2. Then, we have* $\sum_{\mathcal{V}'\in\mathfrak{V}'_{\mathcal{C}'}} X^{(k_1)}_{\mathcal{T}_1,\mathcal{C}'\setminus\mathcal{V}'} = 0$*, where* $X^{(k_1)}_{\mathcal{T}_1,\mathcal{C}'\setminus\mathcal{V}'}$ *represents the message sent by the mirror* M_{k_1}*, which is defined in (9).*

By Lemma 1, for any mirror set $\mathcal{A} \in \binom{[K_1]}{t_1+1}$, $\mathcal{L}_{\mathrm{M}} \bigcap \mathcal{A} = \emptyset$, and the message $X_{\mathcal{A},\mathcal{B}}$, $\mathcal{B} \in \binom{[K_2]}{t_2+1}$ can be computed directly from the broadcast messages by using the following equation

$$X_{\mathcal{A},\mathcal{B}} = \sum_{\mathcal{V}\in\mathfrak{V}_{\mathcal{C}}\setminus\mathcal{L}_{\mathrm{M}}} X_{\mathcal{C}\setminus\mathcal{V},\mathcal{B}} \tag{5}$$

where $\mathcal{C} = \mathcal{A} \bigcup \mathcal{L}_{\mathrm{M}}$.

By Lemma 2, for any user set $\mathcal{B} = \binom{[K_2]}{t_2+1}$, $\mathcal{L}_{k_1} \bigcap \mathcal{B} = \emptyset$, the message $X^{(k_1)}_{\mathcal{T}_1,\mathcal{B}}$ can be computed directly from the broadcast messages by using the equation

$$X^{(k_1)}_{\mathcal{T}_1,\mathcal{B}} = \sum_{\mathcal{V}'\in\mathfrak{V}'_{\mathcal{C}'}\setminus\mathcal{L}_{k_1}} X^{(k_1)}_{\mathcal{T}_1,\mathcal{C}'\setminus\mathcal{V}'} \tag{6}$$

where $\mathcal{C}' = \mathcal{B} \bigcup \mathcal{L}_{k_1}$. After receiving the messages sent by the mirror M_{k_1}, user U_{k_1,k_2} is able to recover its desired linear combination of files.

4.1. An Example for Theorem 2

When $K_1 = 3$, $K_2 = 2$, $t_1 = t_2 = 1$, we can obtain an F-$(K_1, K_2; M_1, M_2; N) = 6 - (3, 2; 2, 3; 6)$ coded caching scheme as follows.

- **Placement phase**: Each file from $\mathbb{F}_2^B$ is divided into $\binom{3}{1}\binom{2}{1} = 6$ subfiles with equal size, i.e., $W^{(n)} = \{W^{(n)}_{1,1}, W^{(n)}_{1,2}, \ldots, W^{(n)}_{3,1}, W^{(n)}_{3,2}\}$, $n \in [6]$. For simplicity, we represent a set that is the subscript of some studied object by a string. For example, $\mathcal{T}_{\{1,2\}}$ is represented by $\mathcal{T}_{12}$. The contents cached by the mirrors are as follows:

$$\mathcal{Z}_1 = \{W^{(n)}_{1,1}, W^{(n)}_{1,2} \mid n \in [6]\}, \mathcal{Z}_2 = \{W^{(n)}_{2,1}, W^{(n)}_{2,2} \mid n \in [6]\}, \mathcal{Z}_3 = \{W^{(n)}_{3,1}, W^{(n)}_{3,2} | n \in [6]\}.$$

The subfiles cached by the users are as follows:

$$\widetilde{\mathcal{Z}}_{1,1} = \widetilde{\mathcal{Z}}_{2,1} = \widetilde{\mathcal{Z}}_{3,1} = \{W^{(n)}_{1,1}, W^{(n)}_{2,1}, W^{(n)}_{3,1} | n \in [6]\},$$
$$\widetilde{\mathcal{Z}}_{1,2} = \widetilde{\mathcal{Z}}_{2,2} = \widetilde{\mathcal{Z}}_{3,2} = \{W^{(n)}_{1,2}, W^{(n)}_{2,2}, W^{(n)}_{3,2} | n \in [6]\}.$$

- **Delivery phase**: In the delivery phase, the demand vectors with length 12 are $\mathbf{d}_{k_1,1} = (1, 1, 0, 0, 0, 0)$, $\mathbf{d}_{k_1,2} = (0, 0, 1, 1, 0, 0), k_1 \in [3]$. As we can see, $\mathbf{D}^{(1)} = \mathbf{D}^{(2)} = \mathbf{D}^{(3)}$.

Without loss of generality, we set $\mathcal{L}_{\rm M}=\{1\}$. We denote a linear combination of subfiles as

$$L_{\mathbf{d}_{k_1,k_2}\mathcal{T}_1,\mathcal{T}_2}=\sum_{n\in[N]}\mathbf{d}^{(n)}_{k_1,k_2}\cdot W^{(n)}_{\mathcal{T}_1,\mathcal{T}_2}.$$

where $\mathcal{T}_1\in\{\{1\},\{2\},\{3\}\}$ and $\mathcal{T}_2\in\{\{1\},\{2\}\}$, $k_1\in[3]$, $k_2\in[2]$. Then, the messages sent in this hierarchical system consist of the following two parts.

- **The messages sent by the server:** The server generates signal $X_{\mathcal{A},\mathcal{B}}$ satisfying $\mathcal{A}\in\binom{[3]}{2}$, $\mathcal{B}\in\binom{[2]}{2}$ and $\mathcal{A}\bigcap\mathcal{L}_{\rm M}\neq\varnothing$ as follows:

$$\begin{aligned}X_{12,12}=&L_{\mathbf{d}_{1,1},2,2}\oplus L_{\mathbf{d}_{1,2},2,1}\oplus L_{\mathbf{d}_{2,1},1,2}\oplus L_{\mathbf{d}_{2,2},1,1}\\=&(\oplus_{i\in[2]}(W^{(i)}_{2,2}\oplus W^{(i)}_{1,2}))\oplus(\oplus_{i\in[3:4]}(W^{(i)}_{2,1}\oplus W^{(i)}_{1,1})),\\X_{13,12}=&L_{\mathbf{d}_{1,1},3,2}\oplus L_{\mathbf{d}_{1,2},3,1}\oplus L_{\mathbf{d}_{3,1},1,2}\oplus L_{\mathbf{d}_{3,2},1,1}\\=&(\oplus_{i\in[2]}(W^{(i)}_{3,2}\oplus W^{(i)}_{1,2}))\oplus(\oplus_{i\in[3:4]}(W^{(i)}_{3,1}\oplus W^{(i)}_{1,1})).\end{aligned}$$

  In this example, we have $\mathcal{L}_{\rm M}=\{1\}$, and $\mathcal{A}=\{2,3\}$ has no intersection with $\mathcal{L}_{\rm M}$. Here, we have $\mathcal{C}=\mathcal{L}_{\rm M}\bigcup\mathcal{A}=\{1,2,3\}$ and $\mathfrak{V}_{\mathcal{C}}=\{\{1\},\{2\},\{3\}\}$. By Lemma 2, we can generate $X_{23,12}=X_{12,12}+X_{13,12}=(\oplus_{i\in[2]}(W^{(i)}_{3,2}\oplus W^{(i)}_{2,2}))\oplus(\oplus_{i\in[3:4]}(W^{(i)}_{3,1}\oplus W^{(i)}_{2,1}))$. Thus, the transmission load of the first layer is $R_1=\frac{3-1}{6}=1/3$.

- **The messages sent by mirror $\mathbf{M}_{k_1}$:** Here, we take mirror M_1 as an example. From $D^{(1)}$, we have $\mathcal{L}_1=\{1,2\}$, and M_1 transmits $X^{(1)}_{\mathcal{T}_1,\mathcal{B}}$, where $\mathcal{T}_1\in\binom{[3]}{1}$, $\mathcal{B}\in\binom{[2]}{2}$, $\mathcal{B}\bigcap\mathcal{L}_1\neq\emptyset$, i.e.,

$$\begin{aligned}&X^{(1)}_{2,12}=X_{12,12}-X^{(2)}_{1,12}=(\oplus_{i\in[2]}W^{(i)}_{2,2})\oplus(\oplus_{i\in[3:4]}W^{(i)}_{2,1}),\\&X^{(1)}_{3,12}=X_{13,12}-X^{(3)}_{1,12}=(\oplus_{i\in[2]}W^{(i)}_{3,2})\oplus(\oplus_{i\in[3:4]}W^{(i)}_{3,1}),\\&X^{(1)}_{1,12}=W^{(1)}_{1,2}\oplus W^{(2)}_{1,2}\oplus W^{(3)}_{1,1}\oplus W^{(4)}_{1,1}.\end{aligned}$$

  Then, the transmission amount by mirror M_1 is 3 packets, and the transmission load of the second layer is $R_2=\frac{3}{6}=\frac{1}{2}$.

User $\mathrm{U}_{1,1}$ can decode $W^{(1)}_{1,2}\oplus W^{(2)}_{1,2}$, $W^{(1)}_{2,2}\oplus W^{(2)}_{2,2}$, $W^{(1)}_{3,2}\oplus W^{(2)}_{3,2}$, from $X^{(1)}_{1,12}$, $X^{(1)}_{2,12}$, $X^{(1)}_{3,12}$, respectively, as it has cached $\{W^{(n)}_{1,1},W^{(n)}_{2,1},W^{(n)}_{3,1}|n\in[6]\}$.

Compared with the Baseline Scheme, which achieves $R_{\rm base1}=\frac{4}{3}$, $R_{\rm base2}=\frac{1}{2}$, our scheme has a significant improvement in R_1.

4.2. General Description of Scheme for Theorem 2

Given a $(K_1,K_2;M_1,M_2;N)$ hierarchical caching system, we have an F-(K_1,K_2,M_1,M_2,N) coded caching scheme where $F=\binom{K_1}{t_1}\binom{K_2}{t_2}$, $t_1\in[0:K_1]$, $t_2\in[0:K_2]$. The scheme consists of two phases.

- **Placement phase:** Firstly, we divide each file into $\binom{K_1}{t_1}$ equal-size subfiles; then, we further divide each subfile into $\binom{K_2}{t_2}$ sub-subfiles. The index of subfiles consists of two parts, $\mathcal{T}_1$ and $\mathcal{T}_2$, i.e., $W^{(n)}=\{W^{(n)}_{\mathcal{T}_1,\mathcal{T}_2}\mid\mathcal{T}_1\in\binom{[K_1]}{t_1},\mathcal{T}_2\in\binom{[K_2]}{t_2}\}$, $n\in[N]$. Each mirror site M_{k_1}, $k_1\in[K_1]$ caches subfiles $W^{(n)}_{\mathcal{T}_1,\mathcal{T}_2}$ according to the following rule, which is mainly related to the first subscript $\mathcal{T}_1$.

$$\mathcal{Z}_{k_1}=\left\{W^{(n)}_{\mathcal{T}_1,\mathcal{T}_2}\middle|\mathcal{T}_1\in\binom{[K_1]}{t_1},\mathcal{T}_2\in\binom{[K_2]}{t_2},k_1\in\mathcal{T}_1,n\in[N]\right\}.$$

Similarly, each user U_{k_1,k_2}, $k_1 \in [K_1], k_2 \in [K_2]$ caches subfiles $W^{(n)}_{\mathcal{T}_1,\mathcal{T}_2}$ according to the following rule, which is mainly related to the second subscript $\mathcal{T}_2$.

$$\widetilde{\mathcal{Z}}_{k_1,k_2} = \left\{ W^{(n)}_{\mathcal{T}_1,\mathcal{T}_2} \middle| \mathcal{T}_1 \in \binom{[K_1]}{t_1}, \mathcal{T}_2 \in \binom{[K_2]}{t_2}, k_2 \in \mathcal{T}_2, n \in [N] \right\}.$$

Under this caching strategy, we can verify that it satisfies the memory constraints stated in Theorem 2. Each mirror caches $\binom{K_1-1}{t_1-1}\binom{K_2}{t_2}N$ subfiles and each user caches $\binom{K_1}{t_1}\binom{K_2-1}{t_2-1}N$ subfiles, where each subfile is $B/\binom{K_1}{t_1}\binom{K_2}{t_2}$ bits. Thus, the memory ratios of the mirror and user are $\frac{M_1}{N} = \frac{t_1}{K_1}$ and $\frac{M_2}{N} = \frac{t_2}{K_2}$, respectively. For any user's demand vector $\mathbf{d_{k_1,k_2}} = (d^{(1)}_{k_1,k_2}, \ldots, d^{(N)}_{k_1,k_2})$ of N-length, we use the notation as follows to denote a linear combination of subfiles:

$$L_{\mathbf{d}_{k_1,k_2}\mathcal{T}_1,\mathcal{T}_2} = \sum_{n \in [N]} \mathbf{d}^{(n)}_{k_1,k_2} W^{(n)}_{\mathcal{T}_1,\mathcal{T}_2}, \mathcal{T}_1 \in \binom{[K_1]}{t_1}, \mathcal{T}_2 \in \binom{[K_2]}{t_2}. \tag{7}$$

- **Delivery phase:** For the convenience of the subsequent discussion, we first give the following two definitions of the signals transmitted in the first layer, say $X_{\mathcal{A},\mathcal{B}}$, and the second layer, say $X^{(k_1)}_{\mathcal{T}_1,\mathcal{B}}$. For any mirror set containing $t_1 + 1$ mirrors defined as $\mathcal{A} \in \binom{[K_1]}{t_1+1}$, any mirror site set containing t_1 mirror sites defined as $\mathcal{T}_1 \in \binom{[K_1]}{t_1}$, and any user set containing $t_2 + 1$ users defined as $\mathcal{B} \in \binom{[K_2]}{t_2+1}$, we define

$$X_{\mathcal{A},\mathcal{B}} = \sum_{k_1 \in \mathcal{A}} \sum_{k_2 \in \mathcal{B}} L_{\mathbf{d}_{k_1,k_2},\mathcal{A}\backslash\{k_1\},\mathcal{B}\backslash\{k_2\}}, \tag{8}$$

$$X^{(k_1)}_{\mathcal{T}_1,\mathcal{B}} = \sum_{k_2 \in \mathcal{B}} L_{\mathbf{d}_{k_1,k_2},\mathcal{T}_1,\mathcal{B}\backslash\{k_2\}}. \tag{9}$$

After the demand matrix $\mathbf{D}$ of size $K_1K_2 \times N$ and its sub-matrix $\mathbf{D}^{(k_1)}$ of size $K_2 \times N$ in (1) are revealed, we have the leader mirror set $\mathcal{L}_{\mathrm{M}}$ according to Definition 1. For each sub-matrix $\mathbf{D}^{(k_1)}$ of $\mathbf{D}$, $k_1 \in [K_1]$, we have the leader user set $\mathcal{L}_{k_1}$, $\mathcal{L}_{k_1} \subseteq [K_2]$ according to Definition 2. There are two types of messages transmitted by the server and mirror, respectively.

  - **The messages sent by the server:** For each $\mathcal{A} \in \binom{[K_1]}{t_1+1}$, $\mathcal{B} \in \binom{[K_2]}{t_2+1}$, $\mathcal{L}_{\mathrm{M}} \bigcap \mathcal{A} \neq \emptyset$, the server transmits $X_{\mathcal{A},\mathcal{B}}$ to the mirror sites.
  - **The messages sent by the mirror:** Mirror site M_{k_1} transmits $X^{(k_1)}_{\mathcal{T}_1,\mathcal{B}}$ via the following rules.
(1) For each $\mathcal{T}_1 \in \binom{[K_1]}{t_1}$, $k_1 \notin \mathcal{T}_1$, $\mathcal{A} = \mathcal{T}_1 \bigcup \{k_1\}$, $\mathcal{B} \in \binom{[K_2]}{t_2+1}$, $\mathcal{B} \bigcap \mathcal{L}_{k_1} \neq \emptyset$, mirror M_{k_1} transmits $X^{(k_1)}_{\mathcal{T}_1,\mathcal{B}}$ by subtracting $\sum_{k_1' \in \mathcal{T}_1} X^{(k_1')}_{\mathcal{A}\backslash\{k_1'\},\mathcal{B}}$ from $X_{\mathcal{A},\mathcal{B}}$, i.e.,

$$X^{(k_1)}_{\mathcal{T}_1,\mathcal{B}} = X_{\mathcal{A},\mathcal{B}} - \sum_{k_1' \in \mathcal{T}_1} X^{(k_1')}_{\mathcal{A}\backslash\{k_1'\},\mathcal{B}}.$$

(2) For each $\mathcal{T}_1 \in \binom{[K_1]}{t_1}$, $k_1 \in \mathcal{T}_1$, $\mathcal{B} \in \binom{[K_2]}{t_2+1}$, $\mathcal{B} \bigcap \mathcal{L}_{k_1} \neq \emptyset$, mirror M_{k_1} directly transmits $X^{(k_1)}_{\mathcal{T}_1,\mathcal{B}}$ to its connected users generated from its cached content $\mathcal{Z}_{k_1}$.

As regards the messages $X_{\mathcal{A},\mathcal{B}}$, $\mathcal{A} \bigcap \mathcal{L}_{\mathrm{M}} = \emptyset$, and $X^{(k_1)}_{\mathcal{T}_1,\mathcal{B}}$, $\mathcal{B} \bigcap \mathcal{L}_{k_1} = \emptyset$, which are also necessary for the users, these messages can be computed from the sent messages by using Lemmas 1 and 2. More precisely, $X_{\mathcal{A},\mathcal{B}}$, $\mathcal{A} \bigcap \mathcal{L}_{\mathrm{M}} = \emptyset$ can be obtained by (5), and $X^{(k_1)}_{\mathcal{T}_1,\mathcal{B}}$, $\mathcal{B} \bigcap \mathcal{L}_{k_1} = \emptyset$ can be obtained by (6).

Now, we prove that each message $X_{\mathcal{A},\mathcal{B}}$ transmitted by the server is decodable, i.e., after each mirror subtracts some packets from $X_{\mathcal{A},\mathcal{B}}$, the rest of the message only contains coded packets required by the users in $\mathcal{U}_{k_1}$. Then, we further prove that each message $X^{(k_1)}_{\mathcal{T},\mathcal{B}}$ transmitted by M_{k_1} is decodable, i.e., after user U_{k_1,k_2}, $k_1 \in [K_1]$, $k_2 \in [K_2]$, subtracting some packets from $X^{(k_1)}_{\mathcal{T},\mathcal{B}}$, the rest of the message only contains coded packets required by user U_{k_1,k_2}.

4.2.1. Decodability of Mirror

For each mirror M_{k_1}, $k_1 \in [K_1]$, it can receive or recover all the $X_{\mathcal{A},\mathcal{B}}$ $\mathcal{A} \subseteq [K_1]$, $\mathcal{B} \subseteq [K_2]$, from the server. By (8), we have

$$\begin{aligned}
X_{\mathcal{A},\mathcal{B}} &= \sum_{k_1 \in \mathcal{A}} \sum_{k_2 \in \mathcal{B}} L_{\mathbf{d}_{k_1,k_2},\mathcal{A}\setminus\{k_1\},\mathcal{B}\setminus\{k_2\}} \\
&= \sum_{k_1 \in \mathcal{A}} \sum_{k_2 \in \mathcal{B}} \sum_{n \in [N]} d^{(n)}_{k_1,k_2} W^{(n)}_{\mathcal{A}\setminus\{k_1\},\mathcal{B}\setminus\{k_2\}} \qquad (10) \\
&= \sum_{k_2 \in \mathcal{B}} \sum_{n \in [N]} d^{(n)}_{k_1,k_2} W^{(n)}_{\mathcal{A}\setminus\{k_1\},\mathcal{B}\setminus\{k_2\}} \\
&+ \sum_{k_1' \in \mathcal{A}\setminus\{k_1\}} \sum_{k_2 \in \mathcal{B}} \sum_{n \in [N]} d^{(n)}_{k_1',k_2} W^{(n)}_{\mathcal{A}\setminus\{k_1'\},\mathcal{B}\setminus\{k_2\}} \qquad (11) \\
&= \underbrace{X^{(k_1)}_{\mathcal{A}\setminus\{k_1\},\mathcal{B}}}_{\text{The coded packets required by users in } \mathcal{U}_{k_1}.} \\
&+ \underbrace{\sum_{k_1' \in \mathcal{A}\setminus\{k_1\}} \sum_{k_2 \in \mathcal{B}} \sum_{n \in [N]} d^{(n)}_{k_1',k_2} W^{(n)}_{\mathcal{A}\setminus\{k_1'\},\mathcal{B}\setminus\{k_2\}}}_{\text{The coded packets cached by } \mathrm{M}_{k_1}.}
\end{aligned}$$

where (10) holds directly from (7), and (11) holds by separating k_1 from $\mathcal{A}$. Moreover, (11) holds by (7) and (9). The first term of (11) denotes coded packets that will be transmitted to $\mathcal{U}_{k_1}$ and the second term denotes packets cached by M_{k_1} because $k_1 \in \mathcal{A} \setminus \{k_1'\}$.

4.2.2. Decodability of User

For each user U_{k_1,k_2}, $k_1 \in [K_1]$, $k_2 \in [K_2]$, it can receive all the $X^{(k_1)}_{\mathcal{T}_1,\mathcal{B}}$, $\mathcal{T}_1 \subseteq [K_1]$, $\mathcal{B} \subseteq [K_2]$, $k_2 \in \mathcal{B}$, from mirror M_{k_1}. By (9), we have

$$\begin{aligned}
X^{(k_1)}_{\mathcal{T}_1,\mathcal{B}} &= \sum_{k_2 \in \mathcal{B}} L_{\mathbf{d}_{k_1,k_2},\mathcal{T}_1,\mathcal{B}\setminus\{k_2\}} = \underbrace{\sum_{n \in [N]} d^{(n)}_{k_1,k_2} W^{(n)}_{\mathcal{T}_1,\mathcal{B}\setminus\{k_2\}}}_{\text{requested by user } \mathrm{U}_{k_1,k_2}} \qquad (12) \\
&+ \underbrace{\sum_{k_2' \in \mathcal{B}\setminus\{k_2\}} \sum_{n \in [N]} d^{(n)}_{k_1,k_2'} W^{(n)}_{\mathcal{T}_1,\mathcal{B}\setminus\{k_2'\}}}_{\text{cached by user } \mathrm{U}_{k_1,k_2}}. \qquad (13)
\end{aligned}$$

where (12) holds directly by separating k_2 from $\mathcal{B}$. It is clear that user U_{k_1,k_2} can decode its desired linear combination of packets, i.e., the first term of (12), by subtracting the cached contents, i.e., the second term of (12), as $k_2 \in \mathcal{B} \setminus \{k_2'\}$, which means that U_{k_1,k_2} has already cached the packets from $X^{(k_1)}_{\mathcal{T}_1,\mathcal{B}}$.

4.2.3. Performance

From the placement phase, each file is firstly divided into $\binom{K_1}{t_1}$ subfiles and then each subfile is further divided into $\binom{K_2}{t_2}$ subfiles, so the subpacketization is $\binom{K_1}{t_1}\binom{K_2}{t_2}$. Each subfile is $B/\binom{K_1}{t_1}\binom{K_2}{t_2}$ bits, each mirror caches $\binom{K_1-1}{t_1-1}\binom{K_2}{t_2}N$ subfiles, and each user caches $\binom{K_1}{t_1}\binom{K_2-1}{t_2-1}N$ subfiles. Thus, the memory ratios of the mirror and user are $\frac{M_1}{N} = \frac{t_1}{K_1}$

and $\frac{M_2}{N} = \frac{t_2}{K_2}$, which satisfy the memory constraints in Theorem 2. In total, the server transmits $\binom{K_1}{t_1+1} - \binom{K_1-|\mathcal{L}_{\rm M}|}{t_1+1}\binom{K_2}{t_2+1}$ multicast messages, and the mirror transmits $\binom{K_2}{t_2+1} - \binom{K_2-\text{rank}(\mathbf{D}^{(k_1)})}{t_1+1}\binom{K_1}{t_1}$ multicast messages. Each message contains $B/\binom{K_1}{t_1}\binom{K_2}{t_2}$ bits, so the transmission loads of the first layer and the second layer are as illustrated in (4). Although the scheme for Theorem 2 has a higher subpacketization level of $\binom{K_1}{t_1}\binom{K_2}{t_2}$ compared with $\binom{K_1}{t_1} + \binom{K_2}{t_2}$ of the Baseline Scheme, we achieve a much lower transmission load R_1 under the same transmission load R_2, where both schemes achieve the optimal transmission load of the second layer when the sub-demand matrix $\mathbf{D}^{(k_1)}$, $k_1 \in [K_1]$ is full rank.

5. Conclusions

In this paper, we studied the linear function retrieval problem for hierarchical cache-aided networks. We proposed two schemes, where the first scheme achieves the optimal transmission load for the second layer for some demand distribution and our second scheme further reduces the load of the first layer while maintaining the same transmission load in the second layer.

Author Contributions: Conceptualization, L.Z. and Y.K.; methodology, L.Z., Y.K., Y.W. and M.C.; writing—original draft preparation, L.Z.; writing—review and editing, Y.W. and M.C. All authors have read and agreed to the published version of the manuscript.

Funding: This work is in part supported by 2022GXNSFDA035087, NSFC (No. 62061004).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data are contained within the article.

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

Appendix A. Proof of Lemma 1

Proof. Without loss of generality, we assume that $\mathcal{L}_{\rm M} = [|\mathcal{L}_{\rm M}|]$. From (8) and (7), we have

$$\begin{aligned}\sum_{\mathcal{V}\in\mathfrak{V}_{\mathcal{C}}} X_{\mathcal{C}\setminus\mathcal{V},\mathcal{B}} &= \sum_{\mathcal{V}\in\mathfrak{V}_{\mathcal{C}}}\sum_{k_1\in\mathcal{C}\setminus\mathcal{V}}\sum_{k_2\in\mathcal{B}} L_{\mathbf{d}_{k_1,k_2},\mathcal{C}\setminus(\mathcal{V}\bigcup\{k_1\}),\mathcal{B}\setminus\{k_2\}}\\ &= \sum_{\mathcal{V}\in\mathfrak{V}_{\mathcal{C}}}\sum_{k_1\in\mathcal{C}\setminus\mathcal{V}}\sum_{k_2\in\mathcal{B}}\sum_{n\in[N]} d^{(n)}_{k_1,k_2}\cdot W^{(n)}_{\mathcal{C}\setminus(\mathcal{V}\bigcup\{k_1\}),\mathcal{B}\setminus\{k_2\}}.\end{aligned} \tag{A1}$$

If the occurrence number of each subfile $W^{(n)}_{\mathcal{T}_1,\mathcal{T}_2}$, $\mathcal{T}_1 \in \binom{[K_1]}{t_1}$, $\mathcal{T}_2 \in \binom{[K_2]}{t_2}$, $n \in [N]$ is even in $\sum_{\mathcal{V}\in\mathfrak{V}_{\mathcal{C}}} X_{C\setminus\mathcal{V},\mathcal{B}}$, then the coefficient of $W^{(n)}_{\mathcal{T}_1,\mathcal{T}_2}$ in the summation is 0. Note that if we focus on $W^{(n)}_{\mathcal{T}_1,\mathcal{T}_2}$, then $k_2 = \mathcal{B}\setminus\mathcal{T}_2$, which is a fixed user label. $W^{(n)}_{\mathcal{T}_1,\mathcal{T}_2}$ appears in $\sum_{\mathcal{V}\in\mathfrak{V}_{\mathcal{C}}} X_{C\setminus\mathcal{V},\mathcal{B}}$ if and only if there exist some mirrors $\mathbf{M}_{k_1}$, $k_1 \in \mathcal{C}\setminus\mathcal{T}_1$, which satisfy the two conditions: $d^{(n)}_{k_1,k_2} \neq 0, \mathcal{C}\setminus(\{k_1\}\bigcup\mathcal{T}_1) \in \mathfrak{V}_{\mathcal{C}}$. Moreover, for each $k_1 \in \mathcal{C}\setminus\mathcal{T}_1$ satisfying the two conditions, there exists one coded message that contains $W^{(n)}_{\mathcal{T}_1,\mathcal{T}_2}$. Thus, we only need to prove that the number of mirrors $\mathbf{M}_{k_1}$, $k_1 \in \mathcal{C}\setminus\mathcal{T}_1$ satisfying the two conditions is even.

Assume that mirror k_1 satisfies the two conditions; then, we have $d^{(n)}_{k_1,k_2} \neq 0$ and $\mathcal{C}\setminus(\{k_1\}\bigcup\mathcal{T}_1) \in \mathfrak{V}_{\mathcal{C}}$. Let $\mathcal{L}'_{\rm M} = \mathcal{C}\setminus(\{x\}\bigcup\mathcal{T}_1) = \{l_1,\ldots,l_{|\mathcal{L}'_{\rm M}|}\}$, and $\mathcal{L}'_{\rm M}$ is also a leader mirror set satisfying Definition 1. Then, by (2), we have $\mathbf{d}_{k_1,k_2} = \alpha^{(k_1)}_1\mathbf{d}_{l_1,k_2} + \ldots + \alpha^{(k_1)}_{|\mathcal{L}'_{\rm M}|}\mathbf{d}_{l_{|\mathcal{L}'_{\rm M}|},k_2}$. Then, there must be k'_1 mirrors in $\alpha^{(k_1)}_1\mathbf{d}_{l_1,k_2} + \ldots + \alpha^{(k_1)}_{|\mathcal{L}'_{\rm M}|}\mathbf{d}_{l_{|\mathcal{L}'_{\rm M}|},k_2}$ satisfying $d^{(n)}_{k_1,k_2} \neq 0$ and the corresponding coefficient $\alpha^{(k'_1)}_{l'_{\rm M}} \neq 0$, $l'_{\rm M} \in [|\mathcal{L}'_{\rm M}|]$. k'_1 is an odd number; otherwise, $d^{(n)}_{k_1,k_2} = 0$. It is easy to check that these k'_1 mirrors also satisfy constraints $d^{(n)}_{k_1,k_2} \neq 0, \mathcal{C}\setminus(\{k_1\}\bigcup\mathcal{T}_1) \in \mathfrak{V}_{\mathcal{C}}$.

Thus, there are in total $k'_1 + 1$ mirrors in $\mathcal{C} \setminus \mathcal{T}_1$, which is an even number, satisfying the two constraints. □

Appendix B. Proof of Lemma 2

Proof. Without loss of generality, we assume that $\mathcal{L}_{k_1} = [|\mathcal{L}_{k_1}|]$. From (9) and (7), we have

$$\sum_{\mathcal{V}'\in\mathfrak{V}'_{\mathcal{C}'}} X^{(k_1)}_{\mathcal{T}_1,\mathcal{C}'\setminus\mathcal{V}'} = \sum_{\mathcal{V}'\in\mathfrak{V}'_{\mathcal{C}'}} \sum_{k_2\in\mathcal{C}'\setminus\mathcal{V}'} L_{\mathbf{d}_{k_1,k_2},\mathcal{T}_1,\mathcal{C}'\setminus(\mathcal{V}'\cup\{k_2\})}$$
$$= \sum_{\mathcal{V}'\in\mathfrak{V}'_{\mathcal{C}'}} \sum_{k_2\in\mathcal{C}'\setminus\mathcal{V}'} \sum_{n\in[N]} d^{(n)}_{k_1,k_2} \cdot W^{(n)}_{\mathcal{T}_1,\mathcal{C}'\setminus(\mathcal{V}'\cup\{k_2\})}.$$

If the occurrence number of subfile $W^{(n)}_{\mathcal{T}_1,\mathcal{T}_2}$ is even in $\sum_{\mathcal{V}'\in\mathfrak{V}'_{\mathcal{C}'}} X^{(k_1)}_{\mathcal{T}_1,\mathcal{C}'\setminus\mathcal{V}'}$, then the coefficient of $W^{(n)}_{\mathcal{T}_1,\mathcal{T}_2}$ in the summation is 0. $W^{(n)}_{\mathcal{T}_1,\mathcal{T}_2}$ appears in $\sum_{\mathcal{V}'\in\mathfrak{V}'_{\mathcal{C}'}} X^{(k_1)}_{\mathcal{T}_1,\mathcal{C}'\setminus\mathcal{V}'}$ if and only if there exist some users $\mathrm{U}_{k_1,x}$, $x\in\mathcal{C}' \setminus \mathcal{T}_2$, which satisfy the following two conditions: $d^{(n)}_{k_1,x} \neq 0$ and $\mathcal{C}' \setminus (\{x\} \bigcup \mathcal{T}_2) \in \mathfrak{V}'_{\mathcal{C}'}$.

Moreover, for each $x \in \mathcal{C}' \setminus \mathcal{T}_2$ satisfying the two conditions, there exists one coded message that contains $W^{(n)}_{\mathcal{T}_1,\mathcal{T}_2}$. Thus, we only need to prove that the number of users $\mathrm{U}_{k_1,x}$, $x\in\mathcal{C}' \setminus \mathcal{T}_2$ satisfying the two conditions is even.

Assume that user $\mathrm{U}_{k_1,x}$ satisfies the two conditions; then, we have $d^{(n)}_{k_1,x} \neq 0$ and $\mathcal{C}' \setminus (\{x\} \bigcup \mathcal{T}_2) \in \mathfrak{V}'_{\mathcal{C}'}$. Let $\mathcal{L}'_{k_1} = \mathcal{C}' \setminus (\{x\} \bigcup \mathcal{T}_2) = \{l_1, \ldots, l_{|\mathcal{L}'_{k_1}|}\}$, and $\mathcal{L}'_{k_1}$ is also a leader user set among users in $\mathcal{U}_{k_1}$, where $\mathrm{rank}(\mathbf{D}^{(k_1)}) = \mathrm{rank}(\mathbf{D}_{\mathcal{L}'_{k_1}})$. Then, we have

$$\mathbf{d}_{k_1,k_2} = \alpha_1 \mathbf{d}_{k_1,l_1} + \ldots + \alpha_{|\mathcal{L}'_{k_1}|} \mathbf{d}_{k_1,l_{|\mathcal{L}'_{k_1}|}}, (\alpha_1, \ldots, \alpha_{|\mathcal{L}'_{k_1}|}) \in [\mathbb{F}_2]^{|\mathcal{L}'_{k_1}|}.$$

Then, there must be x' users in $\alpha_1 \mathbf{d}_{k_1,l_1} + \ldots + \alpha_{|\mathcal{L}'_{k_1}|} \mathbf{d}_{k_1,l_{|\mathcal{L}'_{k_1}|}}$, $(\alpha_1, \ldots, \alpha_{|\mathcal{L}'_{k_1}|}) \in [\mathbb{F}_2]^{|\mathcal{L}'_{k_1}|}$ satisfying $d^{(n)}_{k_1,x} \neq 0$ and the corresponding coefficient $\alpha_{l'_{k_1}} \neq 0$, $k_1 \in [|\mathcal{L}'_{k_1}|]$. x' is an odd number; otherwise, $d^{(n)}_{k_1,x} = 0$. It is easy to check that these x' users also satisfy $\mathcal{C}' \setminus (\{x\} \bigcup \mathcal{T}_2) \in \mathfrak{V}'_{\mathcal{C}'}$. Thus, there are in total $x' + 1$ users in $\mathcal{C}' \setminus \mathcal{T}'_1$, which is an even number, satisfying the two constraints. □

References

1. Maddah-Ali, M.A.; Niesen, U. Fundamental limits of caching. *IEEE Trans. Inf. Theory* **2014**, *60*, 2856–2867. [CrossRef]
2. 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. [CrossRef]
3. Wan, K.; Tuninetti, D.; Piantanida, P. An index coding approach to caching with uncoded cache placement. *IEEE Trans. Inf. Theory* **2020**, *66*, 1318–1332. [CrossRef]
4. 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 2018 IEEE International Conference on Communications, Kansas City, MO, USA, 20–24 May 2018; pp. 1–6.
5. Ji, M.; Tulino, A.M.; Llorca, J.; Caire, G. Caching in combination networks. In Proceedings of the 2015 49th Asilomar Conference on Signals, Systems & Computers, Pacific Grove, CA, USA, 8–11 November 2015; pp. 1269–1273.
6. Ji, M.; Caire, G.; Molisch, A.F. Fundamental limits of caching in wireless D2D networks. *IEEE Trans. Inf. Theory* **2016**, *62*, 849–869. [CrossRef]
7. Zhou, H.; Jiang, K.; He, S.B.; Min, G.Y.; Wu, J. Distributed Deep Multi-Agent Reinforcement Learning for Cooperative Edge Caching in Internet-of-Vehicles. *IEEE Trans. Wireless Commun.* **2023**, *22*, 9595–9609. [CrossRef]
8. Zhou, H.; Wang, Z.; Zheng, H.T.; He, S.B.; Dong, M.X. Cost Minimization-Oriented Computation Offloading and Service Caching in Mobile Cloud-Edge Computing: An A3C-Based Approach. *IEEE Trans. Netw. Sci. Eng* **2023**, *10*, 1326–1338. [CrossRef]
9. Karamchandani, N.; Niesen, U.; Maddah-Ali, M.A.; Diggavi, S.N. Hierarchical coded caching. *IEEE Trans. Inf. Theory* **2016**, *62*, 3212–3229. [CrossRef]

10. Wang, K.; Wu, Y.; Chen, J.; Yin, H. Reduce transmission delay for caching-aided two-layer networks. In Proceedings of the 2023 IEEE International Symposium on Information Theory, Paris, France, 7–12 July 2019; pp. 2019–2023.
11. Kong, Y.; Wu, Y.; Cheng, M. Combinatorial designs for coded caching on hierarchical networks. In Proceedings of the IEEE Conference on Wireless Communications and Networking, Glasgow, UK, 26–29 March 2023; pp. 1–6.
12. Wan, K.; Sun, H.; Ji, M.; Tuninetti, D.; Caire, G. On the optimal load-memory tradeoff of cache-aided scalar linear function retrieval. *IEEE Trans. Inf. Theory* **2021**, *67*, 4001–4018. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.