Fundamental Limits of Cache-aided Shared-link Broadcast Networks and Combination Networks

Kai Wan

université PARIS-SACLAY

NNT : 2018SACLS217

Thèse de doctorat

UNIVERSITÉ PARIS SUD
Comprendre le monde, construire l'avenir

Limites Fondamentales de Stockage pour les Réseaux de Diffusion de Liens Partagés et les Réseaux de Combinaison

Thèse de doctorat de l'Université Paris-Saclay
préparée à l'Université Paris-Sud

École doctorale n°580 : STIC
Spécialité de doctorat: Réseaux, Information et Communications

Thèse présentée et soutenue à Gif-sur-Yvette, 29/06/2018, par

Kai WAN

Rapporteurs:

Petros ELIA Professeur Associé, Eurecom	Rapporteur
Giuseppe CAIRE Professeur, TU Berlin	Rapporteur

Composition du Jury:

Michel KIEFFER Professeur, Université Paris-Sud	Président
Petros ELIA Professeur Associé, Eurecom	Rapporteur
Charly POULLIAT Professeur, Université de Toulouse	Examinateur
Deniz GUNDUZ Professeur Associé, Imperial College London	Examinateur
Daniela TUNINETTI Professeur, Université of Illinois at Chicago	Directeur de thèse
Pablo PIANTANIDA Professeur Associé, CentraleSupélec	Co-Directeur de thèse
Armelle WAUTIER Professeur, CentraleSupélec	Invité
Mireille SARKISS Ingénieur-Chercheur, CEA	Invité

Acknowledgements

Completion of this doctoral dissertation was possible with the support of several people. I would like to express my sincere gratitude to all of them.

I am extremely grateful to my advisors, Prof. Daniela Tuninetti and Prof. Pablo Piantanida for their solid support. I would like to thank them for trusting me and giving this position to me. I would like to thank them for the detailed instruction on this topic which was totally new for me when I started. I know I was not good at speaking and writing in English. I would like to thank them for their patience while discussing with me and correcting my papers. I hope that after these three years, they find that they did not make a bad choice. I am so happy and proud to work with them.

I would like to thank our cooperator Prof. Mingyue Ji, who gave us many interesting research directions, useful ideas and detailed corrections on our work.

I am also grateful to my administrative advisors, Prof. Pierre Duhamel and Prof. Armelle Wautier, who gave me a lot of help on the administrative affairs.

Besides my advisors and cooperator, I would like to thank the rest of my thesis committee: Prof. , Prof. , and Dr. , for their insightful comments.

We also would like to thank Zhangchi Chen from Mathematical Department of University Paris-sud who proved Appendix A.9 and Tang Liu from ECE Department of University of Illinois at Chicago (UIC) who inspired us to derive Appendix A.7.

I gratefully acknowledge the funding sources that made my Ph.D. work possible. I was funded by Labex Digicosme for my first three years and three mouths and was funded by UIC for the last four mouths.

For my fellow labmates in Laboratoire des Signaux et Système (L2S), Weichao, Chen, Li, Maggie, Faton, Wafa, two Chao's', Xiaoxia, Xiaojun, Jian, Xuewen and many others, I would like to thank them for the stimulating discussions, for all the fun we have had in the last years, for everyday we were working together. I would like to thank all my 'brothers' (Siqi, Yunsong, Yao, Nan, Zhichao, etc.) in France with whom I did not feel alone even if I was away from my hometown. I also would like to thank my officemates (Tang, Kenneth, Meysam, Ahmad, Sara, Narueporn) at UIC where I visited three times. My time at Chicago was made enjoyable because of them. I would like to thank all the colleagues and secretaries of both L2S and UIC, for their friendly assistant.

I owe a lot to my parents, who encouraged and helped me at every stage of my personal and academic life, and longed to see this achievement come true. I would like to thank them for their strong love and encouragement.

Last but not the least, I am very much indebted to my wife, who is willing to accompany me in France during these years. I can not find any words to express my gratitude to her. I also would like to thank our lovely dog and cat, who make our life really happy and colorful.

Contents

List of Acronyms

CICS	Concatenated Inner Code delivery Scheme
cMAN	centralized caching scheme proposed by Maddah-Ali and Niesen
cYMA	centralized caching scheme proposed by Yu, Maddah-ali and Avestimehr
dMAN	decentralized caching scheme proposed by Maddah-Ali and Niesen
DIS	Direct Independent delivery Scheme
dYMA	decentralized caching scheme proposed by Yu, Maddah-ali and Avestimehr
e.g.	for example
IC	Index Coding
ICICS	Improved Concatenated Inner Code delivery Scheme
IES	Interference Elimination delivery Scheme
i.e.	that is
MDS code	Maximum Distance Separable code
RAM	Random Access Memory
ROM	Read Only Memory
SRDS	Separate Relay Decoding delivery Scheme

List of Notations

$\mathcal{A}$	Calligraphic symbols denote sets or collections, where a collection is a set of sets, e.g., $\big\{\{1,2\},\{1,3\}\big\}$
A	sans-serif symbols denote caching system parameters
$\mathbf{a}$	symbols in bold font denote vectors
$\mathbb{A}$	blackboard symbols denote matrices or fields
$\mathcal{A}\setminus\mathcal{B}$	$\{x\in\mathcal{A}\|x\notin\mathcal{B}\}$
$\mathbb{A}^T$	the transpose of matrix $\mathbb{A}$
$[a:b:c]$	$\{a,a+b,a+2b,...,c\}$
$[a:c]$	$[a:1:c]$
$[a]$	$[1:a]$
B	number of bits in one file
$dim_{\mathbb{Q}}$	degree of the field extension over $\mathbb{Q}$
$\mathbb{E}[\cdot]$	expectation operator
$\mathcal{G}_{\mathcal{Y}}$	set of users connected to at least two relays in $\mathcal{Y}$
g	coded caching gain compared to conventional uncoded caching scheme
$\gcd(j,n)$	greatest common divisor of j and n
$H(\cdot)$	entropy of a random variable
H	number of relays
$\mathcal{H}_{\mathcal{J}}$	set of relays connected to at least one user in $\mathcal{J}$
$\mathcal{H}_k$	set of connected relays of user k
$I(\cdot;\cdot)$	mutual information between two random variables
$\mathrm{im}(L)$	image of a linear map L
K	number of users in a caching problem
K'	number of users in an index coding problem
K_i	$\binom{\mathsf{H}-i}{\mathsf{r}-i}$
$\mathcal{K}_{\mathcal{Y}}$	set of users whose connected relays are all in $\mathcal{Y}$
$\ker(L)$	kernel of a linear map L

M, M^{user}	normalized memory size of each user
$\mathsf{M}^{\text{relay}}$	normalized memory size of each relay
N	number of files in a caching problem
N'	number of messages in an index coding problem
$\Pr$	probability measure
$\mathbb{Q}$	field of rational numbers
$\mathbb{Q}[\zeta]$	the cyclotomic field obtained by adjoining a primitive root of unit ζ to the rational numbers
r	number of relays connected to each user
R	worst-case max link-load
$\mathsf{R}_{\text{c}}^{\star}$	optimal worst-case max link-load in centralized system
$\mathsf{R}_{\text{c,u}}^{\star}$	optimal worst-case max link-load with uncoded cache placement in centralized system
$\mathsf{R}_{\text{d}}^{\star}$	optimal worst-case max link-load in decentralized system
$\mathsf{R}_{\text{d,u}}^{\star}$	optimal worst-case max link-load with uncoded cache placement in decentralized system
$\mathsf{R}^{\mathsf{s}\to\mathsf{r}}$	max link-load from the server to relays
$\mathsf{R}^{\mathsf{r}\to\mathsf{u}}$	max link-load from relays to users
$\mathcal{R}_{\mathcal{J}}$	set of relays connected to all the users in $\mathcal{J}$
$\text{RLC}(m, \mathcal{S})$	m random linear combinations of the equal-length packets indexed by $\mathcal{S}$;
$T_{\varepsilon}^{(n)}(\cdot)$	set of weak typical n-length sequences
$\mathbf{p}(\mathcal{J})$	$\mathbf{p}(\mathcal{J}) := \big(p_1(\mathcal{J}), \ldots, p_{\lvert\mathbf{p}(\mathcal{J})\rvert}(\mathcal{J})\big)$, a permutation of elements of the set $\mathcal{J}$
$\mathcal{U}_h$	set of connected users of relay h
$\mathcal{U}_{\mathcal{Y}}$	set of common connected users to all the relays in Yc
$\text{Var}()$	variance
$\mathcal{Z}_t$	collection of $t-$subsets of users to which at least one relay is connected to
$\mathbb{Z}, \mathbb{Z}_+$	fields of integers and positive integers
$\oplus$	bit-wise XOR operation between binary bits or vectors with same lengths
$\lvert\cdot\rvert$	the cardinality of a set, or the length of a vector, or the length of a (sub)file in bits
ϕ	the *Euler phi function*
$\binom{x}{y}$	binomial coefficient; for two integers x and y, we let $\binom{x}{y} = 0$ if $x < y$ or $x < 0$ or $y < 0$.

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Background and Motivation

1.1.1 Motivation for Caching

In the last decade, there has been a massive increase of Internet devices connected through the wireless access, and a steep increase in mobile traffic because of multimedia streaming, web-browsing applications and socially-interconnected networks [1], [2]. At the same time, the evolution of mobile devices (in computation power, size and speed of RAM and ROM , etc.) has open possibility for more advanced and sophisticated transmission techniques to reduce the network traffic. For example, Fig. 1.1 illustrates the exponential evolution of micro-processors in mobile devices in terms of computation power. In addition, the users' demands may concentrate on a relatively limited number of files (e.g., latest films and musics). Moreover, in real communication systems, the high temporal variability of network traffic results in congestions during peak-traffic times and underutilization of the network during off-peak times (e.g., see the daily video consumption variation in Chicago provided by 'AT& T' in Fig. 1.2). All of the above together motivates the use of *caching* [3].

Cache is a network component that leverages the device memory to transparently store data so that future requests for that data can be served faster. Caching reduces peak traffic by taking advantage of memories distributed across the network to duplicate content during off-peak times. By doing so, caching effectively allows to shift traffic from peak to peak-off hours, thereby smoothing out traffic variability and reducing congestion. Two phases are included in a caching system: i) *placement phase*: each user stores some bits in its cache without knowledge of later demands; ii) *delivery phase*: after each user has made its request and according to cache contents, the server transmits packets in order to satisfy the user demands. The goal is to minimize the number of transmitted bits (load) such that user demands can be satisfied.

1.1.2 Cache-aided Shared-link Broadcast Networks

In this thesis, we first consider the fundamental performance of cache-aided broadcast systems (also known as shared-link model illustrated in Fig. 1.3) originally proposed by Maddah-Ali and Niesen (MAN) in their seminar works [4], [5]. In the MAN model, a server is connected to K users, or clients, via a shared broadcast error-free link. The server has N files (F_1,..., F_{N}) each of size B bits. Each user has a local cache of size MB bits to store parts of the files available at a server. In the *placement phase*, pieces of the files available at a server are stored within the users' cache. The placement phase is done during off-peak times and thus without knowledge of the users' demands. In this phase, the cache size M is the main limitation. When

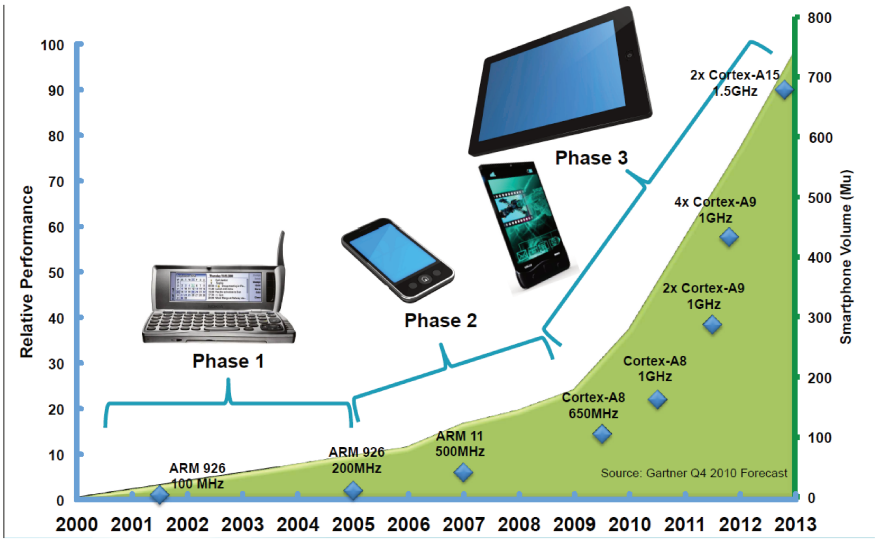

FIGURE 1.1: Evolution of micro-processors in mobile devices.

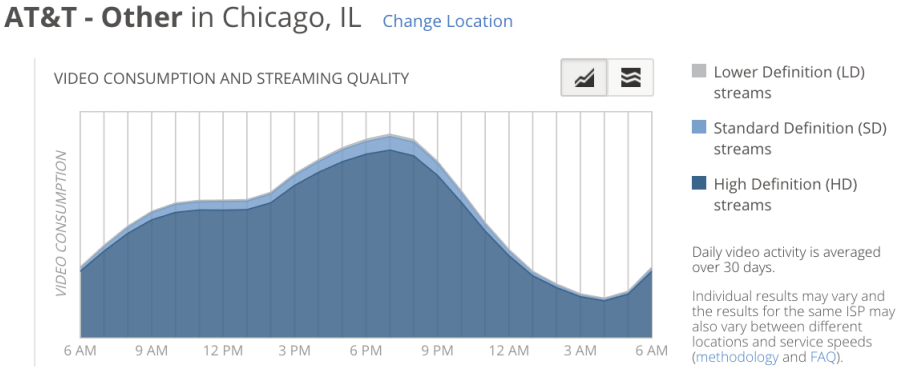

FIGURE 1.2: Daily variation of video consumptions in Chicago provided by 'AT& T'.

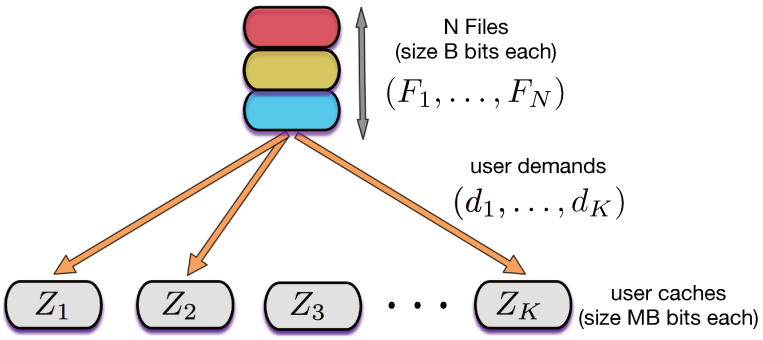

FIGURE 1.3: A cache-aided shared-link broadcast system where a server with N files of size B bits is connected to K users equipped with a cache of size MB bits.

each user directly copies some bits of the files within its cache, the placement phase is said to be *uncoded*, otherwise that it is *coded*. In the *delivery phase*, each user $k \in [\mathsf{K}]$ demands a specific file F_{d_k} from the server. The server, based on the demands and the cache contents, broadcasts RB bits to all users so that each user can recover the demanded file. The delivery phase occurs during peak traffic times, thus the broadcast transmission rate from the server to the users is the main limitation. Therefore, the objective is to design a two-phase scheme so that the load is minimized regardless of the demands, i.e., to minimize the *worst-case load*, or just *load* for short in the rest of thesis.

Cache-aided systems are divided into two classes, *centralized* in [4] and *decentralized* in [5], depending on whether users can coordinate the content stored locally in their caches during the placement phase. In centralized cache-aided systems, the users in the two phases are assumed to be the same; therefore, coordination among users is possible in the placement phase. In practice coordination may not be possible, for example due to users' mobility; thus a user may be connected to a server during its placement phase but to a different one during its delivery phase. In this decentralized scenario, coordination among users during the placement phase is thus not possible.

As pointed out in [4], when content is stored locally uncoded, the delivery phase of the caching problem in shared-link systems can be related to an index coding problem.

The Index Coding (IC) problem, originally proposed in [6], has connections to caching as already pointed out in [4, p. 2865, Section VIII.A. Connection to Index and Network Coding]: " [...] Now, for fixed *uncoded* content placement and for fixed demands, the delivery phase of the caching problem induces a so-called IC problem. However, it is important to realize that the caching problem actually consists of exponentially many parallel such IC problems, one for each of the N^{K} possible user demands. Furthermore, the IC problem itself is computationally hard to solve even only approximately. [...]"

In an IC problem, a sender has N' independent messages and is connected to K' users through an error-free broadcast link. Each user has already some messages stored locally as side information, and demands a subset of the remaining messages. The server broadcasts packets such that each user can recover the desired messages reliably. The objective is to determine the largest message rate region such that messages can be delivered reliably. If each user demands a different message, the IC problem is called a *multiple unicast* IC, and just IC otherwise. The difference between caching and IC is that the side information sets are fixed in IC, while they represent the cache contents that has to be properly designed in caching; moreover, in IC the demands are also fixed, while in caching one must consider all possible demands. In caching, if the cache placement phase is uncoded, the delivery phase is an IC problem for appropriately defined message and demand sets. Hence, IC results can be leveraged for the caching problem. This is exactly our approach in this thesis.

Our main contribution for cache-aided shared-link broadcast systems is to leverage known, and derived novel, results for the index coding problem to determine the fundamental limits of cache-aided systems with uncoded cache placement.

1.1.3 Cache-aided Combination Networks

In practice most communication systems include a network structure other than a single broadcast link. Caching problem were considered in relay networks ([7]–[12]), device-to-device systems ([13]), small cell

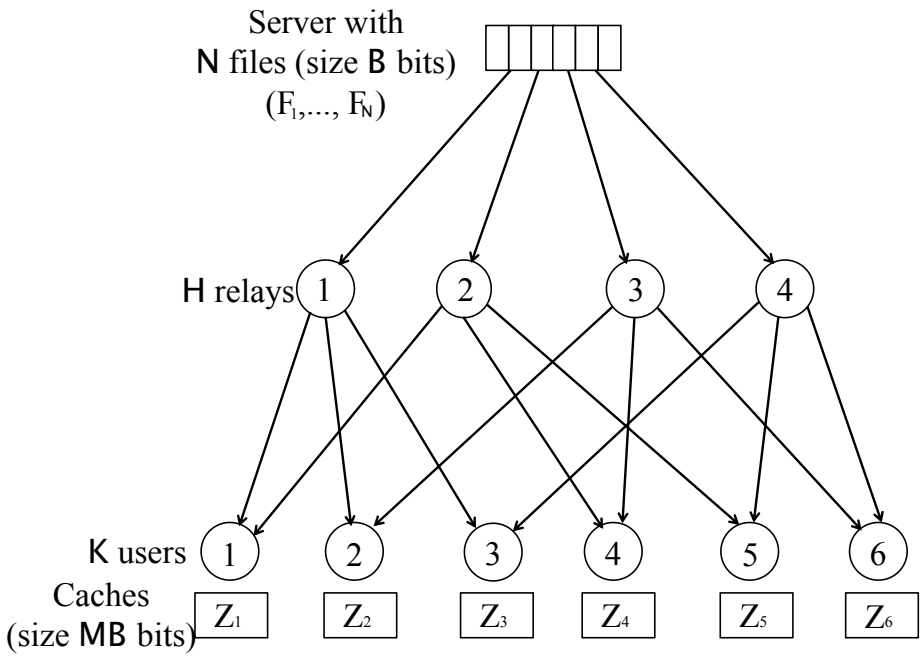

FIGURE 1.4: A combination network with end-user-caches, with $\mathsf{H} = 4$ relays and $\mathsf{K} = 6$ users, i.e., $\mathsf{r} = 2$.

networks ([14]), interference channel ([15]), etc. In this thesis, we consider caching in two-hop relay networks, where the users communicate with the server through some intermediate relays. Since the analysis of relay networks with arbitrary topologies is challenging, a 'symmetric' version of this general problem is known as the *combination networks with end-user-caches* proposed in [12], [16] as shown in Fig. 1.4: a server with N files is connected to H relays (without caches), which in turns are connected to $\mathsf{K} = \binom{\mathsf{H}}{\mathsf{r}}$ users equipped with caches of size M files and where each user is connected to a different subset of the r relays. All links are assumed to be error-free and interference-free. The objective is to determine the optimal *max-link load* $\mathsf{R}^\star$, defined as the smallest max-rate (the maximum rate among all the links, proportional to the download time) for the worst case demands. Lying at the intersection of multi-hop networks and single-hop broadcast channels, combination networks are an interesting class of networks to build up intuition and understanding, and develop new caching converse and achievable bounds applicable to general relay networks.

Our main contribution for combination networks with end-user-caches is to characterize the optimality in terms of the *download time* under the constraint of some system parameter regimes, by deriving improved converse bounds and achievable bounds based on the network topology compared to the state-of-the-art schemes.

1.2 State-of-the-Art and Main Contributions for Cache-aided Shared-link Networks

1.2.1 Past Work on Cache-aided Shared-link Networks

We denote the optimal load as $\mathsf{R}_t^\star$ (i.e., $\mathsf{BR}_t^\star$ is the maximum number of bits transmitted for the-worst case demands) for $t \in \{\mathrm{c}, \mathrm{d}\}$, where 'c' stands for centralized and 'd' for decentralized.

Centralized Cache-aided Systems In [4] Maddah-Ali and Niesen proposed a coded caching scheme that utilizes an uncoded combinatorial cache construction in the placement phase and a binary linear network code in the delivery phase, where content in the caches is stored in a coordinated manner. For the centralized MAN scheme, denoted as cMAN in the following for short, for $\mathsf{M} = t\frac{\mathsf{N}}{\mathsf{K}}$ with $t \in [0 : \mathsf{K}]$, the worst-case load was shown to satisfy

$$\mathsf{R}_{\mathrm{c}}^\star \leq \left(1 - \frac{\mathsf{M}}{\mathsf{N}}\right) \min\left\{\frac{\mathsf{K}}{1 + \mathsf{K}\frac{\mathsf{M}}{\mathsf{N}}}, \mathsf{N}\right\} \tag{1.1a}$$

$$\leq \frac{\mathsf{K} - t}{1 + t} =: \mathsf{R}_{\mathrm{c,u,MAN}}[t], \tag{1.1b}$$

where the subscript "c,u,MAN" in (1.1b) stands for "centralized uncoded-placement Maddah-Ali and Niesen' and where for $\mathsf{M}\frac{\mathsf{K}}{\mathsf{N}} \neq t \in [0 : \mathsf{K}]$ one takes the lower convex envelope of the set of points $(\mathsf{M}, \mathsf{R}) = \left(t\frac{\mathsf{N}}{\mathsf{K}}, \mathsf{R}_{\mathrm{c,u,MAN}}[t]\right)$. In (1.1a) the factor $\left(1 - \frac{\mathsf{M}}{\mathsf{N}}\right) \in [0, 1]$ is interpreted as a 'local caching gain'. Each user can store locally the same fraction $\frac{\mathsf{M}}{\mathsf{N}}$ of each file in the placement phase; the server delivers uncoded the missing fraction $1 - \frac{\mathsf{M}}{\mathsf{N}}$ of the requested file to each user in the delivery phase. This type of gain was well know before the work of Maddah-Ali and Niesen and scales with the size of the cache M. What Maddah-Ali and Niesen surprisingly shown was that a 'global caching gain' is also possible with careful cache placement, by which a single transmission is useful to satisfy the requests of more than one user. This gain is represented by the first term within the minimum in (1.1a). The second term within the minimum in (1.1a) represents the 'natural multicasting gain' that arises when one file is demanded by several users in the regime $\mathsf{N} < \mathsf{K}$.

In [4], the authors derived a cut-set type converse bound as well. The optimal load $\mathsf{R}_{\mathrm{c}}^\star$ must satisfy

$$\mathsf{R}_{\mathrm{c}}^\star \geq \max_{s \in [1:\min(\mathsf{N},\mathsf{K})]} \left(s - \frac{s}{\lfloor \mathsf{N}/s \rfloor}\mathsf{M}\right). \tag{1.2}$$

Comparing this cut-set bound and the load of cMAN in (1.1b), Maddah-Ali and Niesen proved that cMAN caching scheme is optimal within factor 12, but it was noted in [17] that it numerically appeared to be optimal to within a factor 4.7.

The pioneering work in [4] was extended in several ways. In [18], the authors proposed a novel delivery phase with the same placement phase as in cMAN for the case $\mathsf{N} < \mathsf{K}$. This novel group-based delivery is however only valid for the special case of $\mathsf{M} = \mathsf{N}/\mathsf{K}$ (that is, $t = 1$) where it achieves the load $\mathsf{N} - \frac{\mathsf{N}(\mathsf{N}+1)}{2\mathsf{K}}$.

Recently, in [19] the optimal load for centralized cache-aided systems with uncoded cache placement was shown to be achieved essentially following the cMAN original idea, but with the fundamental observation that certain transmitted linear combinations sent by cMAN are redundant (i.e., they can be obtained as

linear combinations of other transmitted messages) leading to the bound

$$\mathsf{R}_{\mathsf{c}}^{\star} \leq \mathsf{R}_{\mathrm{c,u,YMA}}[t] := \frac{\binom{\mathsf{K}}{t+1} - \binom{\mathsf{K}-\min(\mathsf{K},\mathsf{N})}{t+1}}{\binom{\mathsf{K}}{t}}, \tag{1.3}$$

for $\mathsf{M} = t\frac{\mathsf{N}}{\mathsf{K}}$ with $t \in [0 : \mathsf{K}]$, where the subscript "c,u,YMA" in (1.3) stands for "centralized uncoded-placement Yu Maddah-Ali Avestimehr" and where for $\mathsf{M}\frac{\mathsf{K}}{\mathsf{N}} \neq t \in [0 : \mathsf{K}]$ one takes the lower convex envelope of set of points $(\mathsf{M}, \mathsf{R}) = \big(t\frac{\mathsf{N}}{\mathsf{K}}, \mathsf{R}_{\mathrm{c,u,YMA}}[t]\big)$. Notice that the load in (1.3) is strictly smaller than the one in (1.1b) for $\mathsf{N} < \mathsf{K}$. The converse bound under the constraint of uncoded cache placement used in [19] to characterize the optimality of $\mathsf{R}_{\mathrm{c,u,YMA}}$ was first derived in our previous conference works [20], [21], which are the short versions of this present thesis.

In [22]–[25], converse bounds on $\mathsf{R}_{\mathsf{c}}^{\star}$ tighter than cut-set bound provided in [4] were proposed. An improved lower bound compared to the cut-set bound was given in [23, Theorem 1], which was used to show that the effectiveness of caching become small when the number of files becomes comparable to the square of the number of users. An algorithm that generalizes [23] was proposed in [22] to generate bounds of the form $\alpha\mathsf{R} + \beta\mathsf{M} \leq \mathsf{R}_{\mathsf{c}}^{\star}$, for positive integers α and β, and used to show that the upper bound in (1.1b) is optimal to within a factor 4. Another lower bound was obtained in [24] by leveraging [26, Theorem 17.6.1] as a 'symmetrization argument' over demands and used to show that the bound in (1.1b) is optimal to within a factor 8; the lower bound applies to the case where users can request multiple files from the server as well. An approach based on solving a linear program derived from the sub-modularity of entropy and simplified by leveraging certain inherent symmetries in the caching problem was put forth in [25] as a means to computationally generate lower bounds; the approach allowed to solve the case of $\mathsf{K} = 2$ users and any number of files, and gives at present the tightest bounds for problems with small K and N (beyond which the computational approach becomes practically unfeasible).

An enhanced converse bound was proposed in [27] by adding an additional non-negative term in the cut-set converse bound,

$$\mathsf{R}_{\mathsf{c}}^{\star} \geq s - 1 + \alpha - \frac{s(s-1) - l(l-1) + 2\alpha s}{2(\mathsf{N} - l + 1)}\mathsf{M}, \tag{1.4}$$

for any $\alpha \in \{1, \ldots, \min\{\mathsf{N}, \mathsf{K}\}\}$, $\alpha \in [0, 1]$, where $l \in [1 : s]$ is the minimum value such that $\frac{s(s-1)-l(l-1)}{2} + \alpha s \leq (\mathsf{N}-l+1)l^2$. The multiplicative gap between the achieved load in (1.3) and the converse bound in (1.4) was proved to be within factor 2. In other words, coded cache placement can at most half the network load compared to the caching schemes with uncoded cache placement.

The cMAN scheme uses uncoded cache placement and linear network coding on the binary field for the delivery phase. Improvements on cMAN have been obtained by considering coded cache placement and linear network codes on fields of large size. The example of coded cache placement, originally proposed in [4] for the case $\mathsf{N} = \mathsf{K} = 2$, was generalized in [28] to the case $\mathsf{N} \leq \mathsf{K}$ and $\mathsf{MK} \leq 1$ and shown to achieve the optimal point $(\mathsf{M}, \mathsf{R}) = (\mathsf{M}, \mathsf{N}(1 - \mathsf{M}))$, i.e., to coincide with the cut-set bound. In [29, Theorem 1] the authors proposed a coded cache placement scheme, where the packets for the delivery phase are constructed based on the idea of interference elimination and linear combinations on a finite field of sufficiently large size; the proposed method combines rank metric codes and maximum distance separable codes and recovers the result of [28] for $\mathsf{M} = 1/\mathsf{K}$. The authors in [30] proposed a coded caching scheme for $\mathsf{K} \geq \mathsf{N} = 2$ and

$\mathsf{M} = m/\mathsf{K}$ where $m \in [1 : \mathsf{K}-1]$; in the placement phase, coded and symmetric cache placement is used while the delivery phase comprises the transmission of four kinds of linear combinations. Another caching scheme with coded cache placement was proposed in [31] when N and K satisfy $\mathsf{N} < \mathsf{K} \leq 3\mathsf{N}/2$ and (N, K) have a common divisor greater than 1, and $\mathsf{M} = (\mathsf{N}-1)/\mathsf{K}$; the memory-load tradeoff point $(\mathsf{M}, \mathsf{R}) = (\frac{\mathsf{N}-1}{\mathsf{K}}, \frac{\mathsf{K}}{2} + \frac{\mathsf{N}}{\mathsf{K}} - 1)$ was shown to be achievable thus improving on cMAN in (1.1b). In [32], a scheme was proposed for the case of coded cache placement with $\mathsf{N} \leq \mathsf{K}$ and $\frac{\mathsf{MK}}{\mathsf{N}} \leq 1$ that provably has the best load to-date for $\mathsf{N} \leq \mathsf{K} \leq (\mathsf{N}^2+1)/2$.

Decentralized Cache-aided Systems The previously mentioned works leverage coordination among users in the placement phase which is not possible in decentralized systems. In [5], Maddah-Ali and Niesen proposed a scheme, denoted as dMAN in the following, where each user fills its cache randomly and independently of the others. During the delivery phase of dMAN, the bits of the N files are organized into sub-files depending on which subsets of users cached them. With the assumption that the file size B goes to infinity (while N, K, and M are kept fixed), the normalized length (by the file size B) of each sub-file converges in probability to a value that only depends on the number of users who stored this sub-file in the local cache. By repeating K times the delivery phase of cMAN (one for each group of sub-files known by k users, with $k \in [0 : \mathsf{K}-1]$) the following load was shown to be achievable

$$\mathsf{R}_{\mathrm{d}}^{\star} \leq \mathsf{R}_{\mathrm{d,u,MAN}} := \left(1 - \frac{\mathsf{M}}{\mathsf{N}}\right) \min\left\{\frac{\mathsf{N}}{\mathsf{M}}\left[1 - \left(1 - \frac{\mathsf{M}}{\mathsf{N}}\right)^{\mathsf{K}}\right], \mathsf{N}\right\} \tag{1.5a}$$

$$\leq \frac{1 - \frac{\mathsf{M}}{\mathsf{N}}}{\frac{\mathsf{M}}{\mathsf{N}}}\left[1 - \left(1 - \frac{\mathsf{M}}{\mathsf{N}}\right)^{\mathsf{K}}\right], \tag{1.5b}$$

where the factor $\frac{\mathsf{N}}{\mathsf{M}}\left[1 - \left(1 - \frac{\mathsf{M}}{\mathsf{N}}\right)^{\mathsf{K}}\right]$ in (1.5a) represents an additional 'global caching gain' compared to the conventional 'local caching gain' of $\left(1 - \frac{\mathsf{M}}{\mathsf{N}}\right)$; the subscript "d,u,MAN" in (1.5a) stands for "decentralized uncoded-placememnt MAN."

In general, existing centralized caching schemes with uncoded cache placements can be extended to decentralized systems in the same manner as cMAN was used to get dMAN. For example, in [19] the optimal load for decentralized cache-aided systems with uncoded cache placement was shown to be achieved essentially following the dMAN original idea, but with the fundamental observation that certain transmitted linear combinations sent by dMAN are redundant, leading to the upper bound

$$\mathsf{R}_{\mathrm{d}}^{\star} \leq \mathsf{R}_{\mathrm{d,u,YMA}} := \frac{1 - \frac{\mathsf{M}}{\mathsf{N}}}{\frac{\mathsf{M}}{\mathsf{N}}}\left[1 - \left(1 - \frac{\mathsf{M}}{\mathsf{N}}\right)^{\min(\mathsf{K},\mathsf{N})}\right], \tag{1.6}$$

where the subscript "d,u,YMA" stands for "decentralized uncoded-placememnt Yu Maddah-Ali and Avestimehr." The load in (1.6) is strictly smaller than the one in (1.5b) for $\mathsf{N} < \mathsf{K}$ and coincides with the converse bound for decentralized cache-aided systems under the constraint of random, independent and uniform caching, a result that was originally derived in the first author's preliminary Ph.D. defense document [33].

1.2.2 Past Work on Index Coding

A converse bound for the IC problem based on the polymatroid properties of the entropy function was proposed in [34, Theorem 3.1]. For a multiple unicast IC problem, a looser version of [35, Theorem 1] but easier to evaluate was given in [35, Corollary 1]. In [35, Corollary 1] one needs knowledge of the subgraphs not containing a directed cycle in the directed graph generated based on the side information sets (more detailed can be found in Section 2.3). **In this work, we shall use this looser converse bound, referred to as acyclic index coding converse bound, to lower bound the performance of caching schemes with uncoded cache placement.**

For the multiple unicast IC problem several achievable bounds are known: based on the minimization of the rank of certain matrices [36]; based on interference alignment [37], [38]; based on graph proprieties such as clique-cover, partial clique-cover, local clique cover, and partial local clique covering in [36], [39]–[41], respectively; and based on linear coding for a class of interlinked-cycle structure graphs [42].

Information-theoretic random coding arguments inspired by source coding have also been used to derive achievable schemes for the IC problem. The authors of [43]–[45] proposed achievable bounds based on the Heegard-Berger coding scheme 'where side information may be absent' [46]. An achievable bound, referred to as *composite coding*, was proposed in [35, Theorem 2] by combining Slepian-Wolf coding [47] and binning (this last step referred to as flat coding). Composite coding matches the converse bound in [35, Theorem 1] for the multiple unicast IC problem with no more than five messages / users. **In this work, since the composite (index) coding is not optimal in general for the caching problem with uncoded placement, we shall derive a novel IC achievable bound which strictly improves on the composite (index) coding rate region.**

1.2.3 Main Contributions for Cache-aided Shared-link Networks

The exact memory-load tradeoff under the constraint of uncoded cache placement was recently characterized in [19] where the converse bound for centralized cache-aided systems and the its optimality for $K \leq N$ were originally proposed in our conference papers [20], [21], and the converse bound for decentralized cache-aided systems was originally derived in [33]. In this thesis, we focus on cache-aided systems with uncoded placement (both centralized and decentralized) and study their performance based on the IC problem.

Our main contributions for cache-aided shared-link Networks (detailed in Chapter 2) are:

1. *Novel index coding achievable bound.* In Section 3.2, we propose an IC achievable bound based on Han's coding scheme [48], Slepian-Wolf coding [47] and non-unique decoding. This achievable scheme is shown to strictly outperform composite (index) coding and is, to the best our knowledge, the best random coding achievable bound for the general IC problem to date.

2. *Centralized cache-aided systems with uncoded cache placement.* By exploiting [35, Corollary 1], in Section 4.2 we derive a lower bound for the load in centralized cache-aided systems with uncoded cache placement. We originally presented this bound in [20], where we also showed that it matches the load of cMAN in (1.1b) when $K \leq N$. We then show that our novel IC scheme matches our converse bound for cache-aided systems when $K > N$.

3. *Decentralized cache-aided systems with uncoded cache placement.* By again exploiting [35, Corollary 1], in Section 4.3 we derive a lower bound for the load in decentralized cache-aided systems under the constraint that each user randomly, uniformly and independently chooses MB bits of the N files to store. We then prove that our centralized caching scheme can be extended to achieve the proposed converse bound for decentralized systems.

1.3 State-of-the-Art and Main Contributions for Cache-aided Combination Networks

1.3.1 Past Work on Cache-aided Combination Networks

Achievable bounds

For combination networks with end-user-caches, a straightforward way is that in the placement each user caches the same MB/N bits of each file and in the delivery the server uses routing to deliver the non-cached bits of the demanded file to each user. The achieved max link-load by routing is $\mathsf{R}_{\text{routing}} = \mathsf{K}(1-\mathsf{M}/\mathsf{N})/\mathsf{H}$.

A coded caching strategy, referred to as *two-step separation approach*, was formalized in [10] for cache-aided relay networks: (a) cMAN-type uncoded cache placement and multicast message generation (i.e., the generation of the multicast messages is independent of the network topology), and (b) message delivery that aims to match the network multicast capacity region. With MAN placement and cMAN multicast messages generation, the authors in [12] proposed to use an combination network linear coding to deliver cMAN multicast messages, which can achieve the max link-load

$$\mathsf{R}_{\text{MJ}} = \min\left\{\frac{\mathsf{K}(1-\mathsf{M}/\mathsf{N})}{\mathsf{H}}, \frac{\mathsf{K}-t}{\mathsf{r}(1+t)}\right\} \tag{1.7}$$

for $\mathsf{M} = \mathsf{N}t/\mathsf{K}$ where $t \in [0:\mathsf{K}]$.

The another approach is to design the placement and/or the multicast generation based on the network topology, referred to as *non-separation approach*. The existing schemes belonging to this approach were proposed in [49], [50], where the combination network was split into H uncoordinated shared-link networks and then cMAN delivery was used in each one. With a coded cache placement based on MDS (maximum separable distance) codes, the scheme in [50] achieves the following max link-load which is the same as [49] but without the constraint in [49] that r divides H,

$$\mathsf{R}_{\text{ZY}} = \frac{\mathsf{K}-t'}{\mathsf{H}(1+t')} \tag{1.8}$$

for $\mathsf{M} = \mathsf{N}\mathsf{H}t'/\mathsf{K}\mathsf{r}$ where $t' \in [0:\mathsf{K}\mathsf{r}/\mathsf{H}]$. The limitation of this coded placement is that the coded caching gain is now that of a network with $\binom{\mathsf{H}-1}{\mathsf{r}-1} < \mathsf{K}$ equivalent users, which appears to be suboptimal in light of known results for shared-link networks (i.e., the coded caching gain *fundamentally* scales linearly with the number of users K).

Placement Delivery Array (PDA) originally proposed in [51] to reduce the sub-packetization of MAN scheme in the shared-link models was also extended in [52] to combination networks where r divides H to achieve the same load as [12] and [50] but with lower sub-packetization.

Information Theoretic Converse Bounds

The cut-set converse bound in (1.2) for shared-link broadcast Networks was extended to combination networks in [12]. The optimal load $\mathsf{R}_{\mathsf{c}}^{\star}$ must satisfy

$$\mathsf{R}_{\mathsf{c}}^{\star} \geq \max_{\alpha\in[1,\frac{\mathsf{H}}{\mathsf{r}}]} \frac{1}{\lceil \alpha \mathsf{r} \rceil} \max_{l\in\left\{1,\cdots,\min\left\{\mathsf{N},\binom{\lceil \alpha \mathsf{r} \rceil}{\mathsf{r}}\right\}\right\}} \left(l - \frac{l}{\lfloor \frac{\mathsf{N}}{l} \rfloor} \mathsf{M} \right), \tag{1.9}$$

By analysing the above converse bound and the achieved max-link load in (1.7), the caching schemes in [12] is order optimal within factor $\max\{6\sqrt{3}, 6\log(\mathsf{N}/\mathsf{M})\}$. So when $\mathsf{M} \geq \mathsf{N}/6$, the caching schemes in [12] is order optimal within factor $6\sqrt{3}$. However, the gap might be extremely large when the memory size M is small.

1.3.2 Combination Networks with Cache-aided Relays and Users

Combination networks with caches at both the relays and the end-users has recently been considered in [50], where each relay and user can store $\mathsf{M}^{\text{relay}}\mathsf{B}$ and $\mathsf{M}^{\text{user}}\mathsf{B}$ bits of all the N file in the placement phase, respectively. The objective of this problem is to find the capacity region of $(\mathsf{R}^{\mathsf{s}\to\mathsf{r}}, \mathsf{R}^{\mathsf{r}\to\mathsf{u}})$ for each $(\mathsf{M}^{\text{relay}}, \mathsf{M}^{\text{user}})$, where $\mathsf{R}^{\mathsf{s}\to\mathsf{r}}$ is the max link-load from the server to each relay (first layer load) and $\mathsf{R}^{\mathsf{r}\to\mathsf{u}}$ is the max link-load from each relay to each user (second layer load). In [50], the authors proposed a caching scheme where each file is divided into two parts. Each relay only caches the content of first part and each user caches the content of the both parts. The first part is totally cached in the relays such that in the delivery phase, the server does not need to transmit the first part of each file. The proposed caching scheme in [50] for combination networks with end-user-caches is used to let each user recover the second part of his demanded file. For

$$\mathsf{M}^{\text{user}} = \frac{(t_1 - t_2)\mathsf{M}^{\text{relay}}\mathsf{r}}{\mathsf{K}_1} + \frac{t_2\mathsf{N}}{\mathsf{K}_1} \in \left[0, \frac{\mathsf{N}}{\mathsf{r}}\right] \tag{1.10}$$

where $\mathsf{K}_1 = \mathsf{K}\mathsf{r}/\mathsf{H}$, $t_1, t_2 \in [0 : \mathsf{K}_1]$, the achieved first and second layer loads are

$$\mathsf{R}_{\mathsf{ZY}}^{\mathsf{s}\to\mathsf{r}} := \frac{\mathsf{K}_1 - t_2}{\mathsf{r}(t_2+1)}\left(1 - \frac{\mathsf{M}^{\text{relay}}\mathsf{r}}{\mathsf{N}}\right), \tag{1.11}$$

$$\mathsf{R}_{\mathsf{ZY}}^{\mathsf{r}\to\mathsf{u}} := \frac{1}{\mathsf{r}}\left(1 - \frac{\mathsf{M}^{\text{user}}}{\mathsf{N}}\right). \tag{1.12}$$

It was proved [50] that the achieved second layer load of the proposed scheme is information theoretically optimal.

1.3.3 Main Contributions for Cache-aided Combination Networks

Our main contributions for cache-aided combination networks are:

1. For combination networks with end-user-caches, in Section 6.2 we study converse bounds on the max link-load under the constraint of uncoded cache placement when $\mathsf{N} \geq \mathsf{K}$ (the case $\mathsf{N} < \mathsf{K}$ can be treated by similar ideas and is not treated here for simplicity). Based on the cut-set strategy in [12], we first extend the enhanced cut-set converse bound in (1.4) and the converse bound under the constraint

of uncoded placement for shared-link models to combination networks with end-user-caches. We then extend the acyclic index coding converse bound to combination networks and propose a converse bound based on this extended acyclic index coding converse bound. Furthermore, by deriving bounds on the joint entropy of the various random variables that define the problem, we provide two novel ways to tighten the "acyclic index coding converse bound". The combination of these two ideas produces the best known converse bound to date, to the best of our knowledge.

As a result of independent interest, an inequality that generalizes the well-known sub-modularity of entropy is derived, which may find applications in other network information theory problems.

2. Based on the two-step separation approach, in Section 7.2 we propose four delivery schemes for combination networks with end-user-caches to deliver cMAN multicast messages to the corresponding users:

   - We first propose a novel delivery scheme, Direct Independent delivery Scheme (DIS), by exploiting the fact that not all the linear combinations of the MAN delivery scheme are useful to every user.
   - For $\mathsf{M} = \mathsf{N}/\mathsf{K}$, Interference Elimination delivery Scheme (IES), use interference elimination (a form of interference alignment) to rid the users of the MAN coded messages that are not of interest.
   - Concatenated Inner Code delivery Scheme (CICS) proposes a delivery coding scheme including two phases: in the first phase we directly transmit each MAN multicast message to some relays, which forward them to their connected users; such messages are simultaneously useful for $t = \mathsf{KM}/\mathsf{N}$ users and will be used as 'side information' in the next phase; Since not all users are able to decode their desired file at the end of the first phase, in the second phase, we thus deliver the MAN multicast messages through a carefully design network code, to let the remaining 'unsatisfied' users recover their demanded file. In this phase, the multiplicative coding gain
   - By leveraging the multicasting opportunities which are ignored in CICS, we propose, Improved Concatenated Inner Code delivery Scheme (ICICS).

3. By comparing the proposed achievable schemes with cMAN placement and the proposed converse bounds under the constraint of uncoded cache placement, we get the (order) optimality results illustrated in Table 1.1. Since the achieved max link-load in (1.7) was proved in [12] to be order optimal within factor $6\sqrt{3}$, it can be seen that one of our main contributions is the order optimality within a constant value when M is small. The aforementioned optimality results can extended to the general case (coded or uncoded cache placement) by multiplying the order optimality factors by 2.

4. Based on the separation approach, with cMAN placement, in Section 7.3 we propose a novel delivery scheme which generates multicast messages by leveraging the network topology. This delivery scheme could be used with any uncoded cache placement, with the proposed coded cache placements in this thesis and with decentralized cache placements.

5. The main limitation of cMAN placement based schemes is that, due to the combination network topology, the multicasting opportunities (directly related to the overall coded caching gain compared

TABLE 1.1: Order optimality results under the constraint of uncoded cache placement

Delivery Schemes	Constraint of system parameters	Optimality under the constraint of uncoded cache placement
DIS, [12]	$\mathsf{N} \geq \mathsf{K}$	order optimal within factor $\min\{\mathsf{H}/\mathsf{r}, t+1\}$ for $\mathsf{M} = \mathsf{N}t/\mathsf{K}$
DIS,CICS, ICICS	$\mathsf{N} \geq \mathsf{K}$, $\mathsf{r} > \mathsf{H}t/(t+1)$, $\mathsf{M} = \mathsf{N}t/\mathsf{K}$	optimal
DIS,CICS, ICICS	$\mathsf{N} \geq \mathsf{K}$, $\mathsf{r} = \mathsf{H}-1$	optimal
CICS	$\mathsf{N} \geq \mathsf{K}$, $\mathsf{M} = \mathsf{N}t/\mathsf{K}$	order optimal within factor $1 + t/\mathsf{r}$
CICS	$\mathsf{N} \geq \mathsf{K}$, $\mathsf{M} \leq \mathsf{r}t/\mathsf{K}$	order optimal within factor 2
CICS	$\mathsf{N} \geq \mathsf{K}$, $\mathsf{M} = \mathsf{N}t/\mathsf{K}$, $t/\mathsf{r} \to 0$	optimal
IES	$\mathsf{H} \leq 2\mathsf{r}$, $\mathsf{N} \geq \mathsf{K}$, $\mathsf{M} \leq \mathsf{N}/\mathsf{K}$	optimal
IES	$\mathsf{H} > 2\mathsf{r}$, $\mathsf{N} \geq \mathsf{K}$, $\mathsf{M} \leq \mathsf{N}/\mathsf{K}$	order optimal within factor $2\mathsf{r}/(2\mathsf{r}-1) \leq 4/3$

to the uncoded routing scheme) to transmit the various subfiles are not "symmetric" across subfiles (because relays are connected to different sets of users). For example, consider the combination network in Fig. 1.4 and $t = 2$, since users 1 and 2 are connected to one relay (relay 1) while no relays is connected to users 1 and 6 simultaneously, it costs less load to transmit one multicast messages to users in $\{1, 2\}$ from the server than to users $\{1, 6\}$. Hence, in Section 7.4 we fix one coded caching gain $g \in \left[1 : \binom{\mathsf{H}-1}{\mathsf{r}-1}\right]$ compared to uncoded routing scheme and propose an asymmetric coded placement scheme to achieve this coded caching gain, where 'asymmetric' means that it is not necessary to generate one subfile known by each $(g-1)-$subset of users and 'coded' means that we use an MDS-precoded placement such that each user needs only to decode the subfiles with highest multicast opportunities. It is interesting to see that with this asymmetric placement, the delivery phase is symmetric. This scheme is proved to be information theoretically optimal when $\mathsf{M} \geq \frac{(\mathsf{K}-\mathsf{H}+\mathsf{r}-1)\mathsf{N}}{\mathsf{K}}$.

6. Comparing to all the existing schemes on combination networks with end-user-caches, DIS is strictly better than the caching schemes in [12], and the asymmetric coded placement scheme is strictly better than the caching scheme in [50].

7. In Chapter 9, we extend the proposed results to more general models, including combination networks with cache-aided relays and users, and more general relay networks. Different to the existing scheme in [50] where the packets transmitted from the server are independent of the cached contents of relays, we propose a caching scheme where the cached contents in relays can also help users to decode the coded messages transmitted from the server and thus can further reduce the transmitted load from the server to relays. We prove that for some memory size regime, the achieved first layer load of the proposed scheme is information theoretically optimal. In the end, we extend the proposed schemes to more general relay networks with a further step to 'balance' the link-loads transmitted to relays. Numerical results show that in both extended models, the proposed scheme improve the existing schemes.

1.4 Publications

The following lists are the papers produced during my Ph.D. candidature.

1.4.1 Journal Articles

Submitted

1. Kai Wan, Daniela Tuninetti and Pablo Piantanida, "An Index Coding Approach to Caching with Uncoded Cache Placement", to IEEE Trans. on Information Theory, 2017. [53]

In preparation

1. Kai Wan, Mingyue Ji, Pablo Piantanida and Daniela Tuninetti, "Caching in Combination Networks: Novel Asymmetric Coded Cache Placement and Multicast Message Generation by Leveraging Network Topology", to IEEE Trans. on Information Theory. [54]

2. Kai Wan, Mingyue Ji, Pablo Piantanida and Daniela Tuninetti, "Novel Outer Bounds and Inner Bounds with Uncoded Cache Placement for Combination Networks with End-User- Caches", to IEEE Trans. on Information Theory. [55]

1.4.2 Conference Papers

Published or Accepted

1. Kai Wan, Mingyue Ji, Pablo Piantanida and Daniela Tuninetti, "On the Benefits of Asymmetric Coded Cache Placement in Combination Networks with End-User Caches", in IEEE Int. Symp. Inf. Theory (ISIT), Jun. 2018. [56]

2. Kai Wan, Mingyue Ji, Pablo Piantanida and Daniela Tuninetti, "Caching in Combination Networks: Novel Multicast Message Generation and Delivery by Leveraging the Network Topology", in IEEE Intern. Conf. Commun. (ICC), May 2018. [57]

3. Kai Wan, Daniela Tuninetti, Pablo Piantanida and Mingyue Ji, "On Combination Networks with Cache-aided Relays and Users", in Proceedings of Workshop on Smart Antennas (WSA), Mar. 2018. [58]

4. Kai Wan, Daniela Tuninetti, Pablo Piantanida and Mingyue Ji, "A Novel Asymmetric Coded Placement in Combination Networks with end-user Caches", in Proceedings of IEEE Infor. Theory Application Workshop (ITA), Feb. 2018. [59]

5. Kai Wan, Mingyue Ji, Pablo Piantanida and Daniela Tuninetti, "Novel Outer Bounds for Combination Networks with End-User-Caches", in Proceedings of the IEEE Information Theory Workshop (ITW), Nov. 2017. [60]

6. Kai Wan, Daniela Tuninetti, Mingyue Ji, and Pablo Piantanida, "State-of-the-art in Cache-aided Combination Networks", in Proceedings of the IEEE Asilomar Conf., Nov 2017. [61]

7. Kai Wan, Mingyue Ji, Pablo Piantanida and Daniela Tuninetti, "Novel Inner Bounds for Combination Networks with End-User-Caches", in Proceedings of the 55th Allerton Conf. Commun., Control, Comp., Oct. 2017. [62]

8. Kai Wan, Daniela Tuninetti and Pablo Piantanida, "Novel Delivery Schemes for Decentralized Coded Caching in the Finite File Size Regime", in Proceedings of IEEE Intern. Conf. Commun. Workshops (ICCW), May 2017. [63]

9. Kai Wan, Daniela Tuninetti and Pablo Piantanida, "A Novel Index Coding Scheme and its Application to Coded Caching", in Proceedings of IEEE Infor. Theory Application Workshop (ITA), Feb. 2017. [64]

10. Kai Wan, Daniela Tuninetti and Pablo Piantanida, "On the Optimality of Uncoded Cache Placement", in Proceedings of the IEEE Information Theory Workshop (ITW), pp. 161-165, Sep. 2016. [20]

11. Kai Wan, Daniela Tuninetti and Pablo Piantanida, "On Caching with More Users than Files", in IEEE Int. Symp. Inf . Theory (ISIT), pp. 135-139, Jul. 2016. [21]

In preparation

1. Kai Wan, Mingyue Ji, Pablo Piantanida and Daniela Tuninetti, "Caching in Combination Networks: A Novel Delivery by Leveraging the Network Topology", to IEEE Inf. Theory Workshop (ITW), available at arXiv:1802.10479, Apr. 2018. [65]

Part 1

Coded Caching in Shared-link Broadcast Networks

Chapter 2

System Model and Some Known Results

2.1 The Index Coding Problem: Definition

In this section, we provide the information-theoretic formulation of the IC problem shown in Fig. 2.1. A sender wishes to communicate N' independent messages to K' users. Each user $j \in [\mathsf{K}']$ demands a set of messages indexed by $\mathcal{D}_j \subseteq [\mathsf{N}']$ and knows a set of messages indexed by $\mathcal{A}_j \subseteq [\mathsf{N}']$. In order to avoid trivial problems, it is assumed that $\mathcal{D}_j \neq \emptyset$, $\mathcal{A}_j \neq [\mathsf{N}']$, and $\mathcal{D}_j \cap \mathcal{A}_j = \emptyset$. The server is connected to the users through a noiseless channel with finite input alphabet $\mathcal{X}$.

A $(2^{nR_1}, \ldots, 2^{nR_{\mathsf{N}'}}, n, \epsilon_n)$-code for this IC problem is defined as follows. Each message M_i, $i \in [\mathsf{N}']$, is uniformly distributed on $[2^{nR_i}]$, where n is the block-length and $R_i \geq 0$ is the transmission rate in bits per channel use. In order to satisfy the users' demands, the server broadcasts $X^n = \mathsf{enc}(M_1, \ldots, M_{\mathsf{N}'}) \in \mathcal{X}^n$ where enc is the encoding function. Each user $j \in [\mathsf{K}']$ estimates the messages indexed by $\mathcal{D}_j$ by the decoding function $\mathsf{dec}_j\big(X^n, (M_i : i \in \mathcal{A}_j)\big)$. The probability of error is

$$\epsilon_n := \max_{j \in [\mathsf{K}']} \Pr\left[\mathsf{dec}_j\big(X^n, (M_i : i \in \mathcal{A}_j)\big) \neq (M_i : i \in \mathcal{D}_j)\right]. \tag{2.1}$$

A rate vector $(R_1, \ldots, R_{\mathsf{N}'})$ is said to be achievable if there exists a family of $(2^{nR_1}, \ldots, 2^{nR_{\mathsf{N}'}}, n, \epsilon_n)$-codes for which $\lim_{n\to\infty} \epsilon_n = 0$.

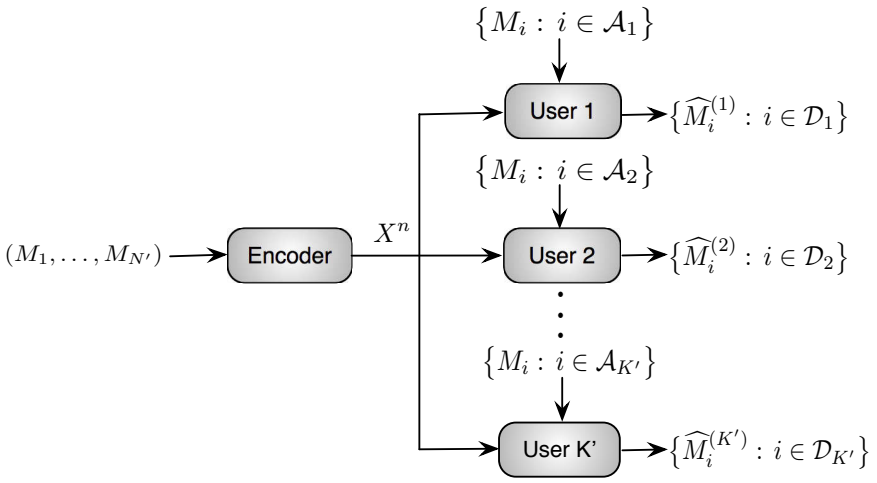

FIGURE 2.1: An IC problem with N' files and K' users.

2.2 The Index Coding Problem: Composite (Index) Coding Achievable Bound

The composite (index) coding achievable bound proposed in [35] is a two-stage scheme based on binning and non-unique decoding. In the first encoding stage, for each $\mathcal{J} \subseteq [\mathsf{N}']$, the messages $(M_i : i \in \mathcal{J})$ are encoded into the 'composite index' $W_{\mathcal{J}} \in [2^{nS_{\mathcal{J}}}]$ based on random binning at some rate $S_{\mathcal{J}} \geq 0$. By convention $S_{\emptyset} = 0$. In the second encoding stage, the collection of all composite indices $(W_{\mathcal{J}} : \mathcal{J} \subseteq [\mathsf{N}'])$ is mapped into a length-n sequence X^n which is received error-free by all users.

In the first decoding stage, every user recovers all composite indices. In the second decoding stage, user $j \in [\mathsf{K}']$ chooses a set $\mathcal{K}_j$ such that $\mathcal{D}_j \subseteq \mathcal{K}_j \subseteq [\mathsf{N}']\backslash\mathcal{A}_j$ and simultaneously decodes all messages $(M_i : i \in \mathcal{K}_j)$, but uniquely only those in $\mathcal{D}_j$, based on the recovered $(W_{\mathcal{J}} : \mathcal{J} \subseteq \mathcal{K}_j \cup \mathcal{A}_j)$.

The achievable rate region with composite (index) coding is stated next.

Theorem 1 (Composite (Index) Coding Achievable Bound, generalization of [35] to allow for multicast messages)**.** *A non-negative rate tuple* $\mathbf{R} := (R_1, \ldots, R_{\mathsf{N}'})$ *is achievable for the IC problem* $\Big((\mathcal{A}_j, \mathcal{D}_j) : j \in [\mathsf{K}']\Big)$ *with* $\mathsf{N}' = \big|\cup_{j\in[\mathsf{K}']}\mathcal{A}_j \cup \mathcal{D}_j\big|$ *if*

$$\mathbf{R} \in \bigcap_{j\in[\mathsf{K}']} \bigcup_{\mathcal{K}_j:\mathcal{D}_j\subseteq\mathcal{K}_j\subseteq[\mathsf{N}']\backslash\mathcal{A}_j} \mathscr{R}(\mathcal{K}_j|\mathcal{A}_j, \mathcal{D}_j), \tag{2.2a}$$

$$\mathscr{R}(\mathcal{K}|\mathcal{A}, \mathcal{D}) := \bigcap_{\mathcal{J}:\mathcal{J}\subseteq\mathcal{K}} \left\{\sum_{i\in\mathcal{J}} R_i < v_{\mathcal{J}}\right\}, \tag{2.2b}$$

where in (2.2b) *the value* $v_{\mathcal{J}}$ *is defined as*

$$v_{\mathcal{J}} := \sum_{\mathcal{P}:\mathcal{P}\subseteq\mathcal{A}\cup\mathcal{K},\mathcal{P}\cap\mathcal{J}\neq\emptyset} S_{\mathcal{P}}, \tag{2.2c}$$

and where in (2.2c) *the non-negative quantities* $(S_{\mathcal{J}} : \mathcal{J} \subseteq [\mathsf{N}'])$ *must satisfy*

$$\sum_{\mathcal{J}:\mathcal{J}\in[\mathsf{N}'],\mathcal{J}\nsubseteq\mathcal{A}_j} S_{\mathcal{J}} \leq \log_2(|\mathcal{X}|), \quad \forall j \in [\mathsf{K}']. \tag{2.2d}$$

Note that the constrain in (2.2d) is from the first decoding stage at receiver $j \in [\mathsf{K}']$, and (2.2c) from the second decoding stage at receiver $j \in [\mathsf{K}']$.

2.3 The Index Coding Problem: Acyclic Subgraph Converse Bound for Multiple Unicast Index Coding

If $\mathsf{K}' = \mathsf{N}'$ and $\mathcal{D}_j = \{j\}$ and $j \notin \mathcal{A}_j$ for each $j \in [\mathsf{N}']$, the multiple unicast IC problem can be represented as a directed graph G, where each node in the graph represents one user and its demanded message, and where a directed edge connects node i to node j if user j knows message M_i. By the submodularity of entropy, a converse bound was proposed in [34, Theorem 3.1] for the symmetric rate case and extended in [35, Theorem 1] to the case where messages can have different rates. Due to the high computational complexity, the converse bound [35, Theorem 1] can only be evaluated for IC problems with limited number

of messages. A looser (compared to [35, Theorem 1]) converse bound was proposed in [35, Corollary 1] and is stated next.

Theorem 2. *(Acyclic Subgraph Converse Bound [35, Corollary 1]) If $(R_1, \ldots, R_{\mathsf{N}'})$ is achievable for the multiple unicast IC problem $\left((\mathcal{A}_j, \mathcal{D}_j = \{j\}) : j \in [\mathsf{N}']\right)$ represented by the directed graph G then it must satisfy*

$$\sum_{j \in \mathcal{J}} R_j \leq \log_2(|\mathcal{X}|), \tag{2.3}$$

for all $\mathcal{J} \subseteq [\mathsf{N}']$ where the sub-graph of G over the vertices in $\mathcal{J}$ does not contain a directed cycle.

2.4 The Caching Problem: Definition

Centralized Cache-aided Systems The information-theoretic formulation of the centralized coded caching problem, as originally formulated by Maddah-Ali and Niesen in [4], is as follows:

- The system comprises a server with N independent files, denoted by $(F_1, F_2, \ldots, F_{\mathsf{N}})$, and K users connected to it through an error-free link. Each file has B equally likely bits. The system is illustrated in Fig. 1.3.
- In the placement phase, user $k \in [\mathsf{K}]$ stores content from the N files in its cache of size MB bits without knowledge of later demands, where $\mathsf{M} \in [0, \mathsf{N}]$. We denote the content in the cache of user $k \in [\mathsf{K}]$ by $Z_k = \phi_k(F_1, \ldots, F_{\mathsf{N}})$, where

$$\text{(placement functions) } \phi_k : [0:1]^{\mathsf{NB}} \to [0:1]^{\lfloor \mathsf{MB} \rfloor}, \ \forall k \in [\mathsf{K}]. \tag{2.4}$$

  We also denote by $\mathbf{Z} := (Z_1, \ldots, Z_{\mathsf{K}})$ the content of all K caches.
- In the delivery phase, each user demands one file and the demand vector $\mathbf{d} := (d_1, d_2, \ldots, d_{\mathsf{K}})$, where $d_k \in [\mathsf{N}]$ corresponds to the file demanded by user $k \in [\mathsf{K}]$, is revealed to the server and all users. Given $(\mathbf{Z}, \mathbf{d})$, the server broadcasts the message $X_{\mathbf{d}} = \psi(F_1, \ldots, F_{\mathsf{N}}, \mathbf{Z}, \mathbf{d})$, where

$$\text{(encoding function) } \psi : [0:1]^{\mathsf{NB}} \times [0:1]^{\mathsf{K}\lfloor \mathsf{MB} \rfloor} \times [\mathsf{N}]^{\mathsf{K}} \to [0:1]^{\lceil \mathsf{RB} \rceil}. \tag{2.5}$$

- Each user $k \in [\mathsf{K}]$ estimates the demanded file as $\widehat{F}_k = \mu_k(X_{\mathbf{d}}, Z_k)$, where

$$\text{(decoding functions) } \mu_k : \ [0:1]^{\lceil \mathsf{RB} \rceil} \times [0:1]^{\lfloor \mathsf{MB} \rfloor} \to [0:1]^{\mathsf{B}}, \ \forall k \in [\mathsf{K}]. \tag{2.6}$$

- The (worst-case over all possible demands) probability of error is

$$P_{\text{err,c}}^{(\mathsf{B})} := \max_{\mathbf{d} \in [\mathsf{N}]^{\mathsf{K}}} \Pr\left[\bigcup_{k=1}^{\mathsf{K}} \left\{\widehat{F}_k \neq F_{d_k}\right\}\right]. \tag{2.7}$$

- A pair (M, R) is said to be achievable if there exit placement functions, encoding function and decoding functions such that $\lim_{\mathsf{B} \to \infty} P_{\text{err,c}}^{(\mathsf{B})} = 0$, where $P_{\text{err,c}}^{(\mathsf{B})}$ was defined in (2.7).

- The objective is to determine, for a fixed M, the (worst-case) load

$$\mathsf{R}_{\mathrm{c}}^{\star} := \inf\{\mathsf{R} : (\mathsf{M}, \mathsf{R}) \text{ is achievable}\}. \tag{2.8}$$

In the following, we say that the placement phase is *uncoded* if the bits of the various files are simply copied within the caches, as formally defined next.

Definition 1 (Uncoded cache placement)**.** *The placement phase is said to be uncoded if the cache contents in* (2.4) *are* $Z_k = (A_{1,k}, A_{2,k}, \ldots, A_{\mathsf{N},k})$ *where* $A_{i,k} \subseteq F_i$ *for all files* $i \in [\mathsf{N}]$ *and such that* $\sum_{i\in[\mathsf{N}]} |A_{i,k}| \leq \mathsf{MB}$*, for all users* $k \in [\mathsf{K}]$.

The (worst-case) load with uncoded cache placement is

$$\mathsf{R}_{\mathrm{c,u}}^{\star} := \inf\{\mathsf{R} : \ (\mathsf{M}, \mathsf{R}) \text{ is achievable with uncoded cache placement}\}. \tag{2.9}$$

Decentralized Cache-aided Systems Centralized systems allow for coordination among users in the placement phase, while decentralized systems do not. So in decentralized systems the caching functions in (2.4) are replaced by a random function applied independently by the users. Here we only give the formal problem definition for uncoded placement as per Definition 1.

- Each user caches bits of the files independently of the other users, that is, for some placement probability mass function $q := \big(q_{(A_1,A_2,\ldots,A_{\mathsf{N}})} : A_j \subseteq F_j,\ j \in [\mathsf{N}]\big)$, we let

$$\Pr\left[Z_k = (A_{1,k}, A_{2,k}, \ldots, A_{\mathsf{N},k}), k \in [\mathsf{K}]\right] = \prod_{u\in[\mathsf{K}]} q_{(A_{1,u},A_{2,u},\ldots,A_{\mathsf{N},u})}, \tag{2.10}$$

  for any $A_{j,k} \subseteq F_j,\ j \in [\mathsf{N}], k \in [\mathsf{K}]$.

- For a demand vector $\mathbf{d} = (d_1, d_2, \ldots, d_{\mathsf{K}})$ and a realization of the caches $\mathbf{Z} = (Z_1, \ldots, Z_{\mathsf{K}})$, the server sends $X_{\mathbf{d}}$ (defined as in (2.5)) and receiver $k \in [\mathsf{K}]$ decodes F_{d_k} (defined as in (2.6)).

- The (worst-case over all possible demands) probability of error is

$$P_{\mathrm{err,d}}^{(\mathsf{B})} := \max_{\mathbf{d}\in[\mathsf{N}]^{\mathsf{K}}} \Pr\left[\bigcup_{k=1}^{\mathsf{K}} \left\{\widehat{F}_k \neq F_{d_k}\right\} \cup \bigcup_{k=1}^{\mathsf{K}} \{|Z_k| > \mathsf{MB}\}\right], \tag{2.11}$$

  where the error event in (2.11) includes both the 'decoding error' $\bigcup_{k=1}^{\mathsf{K}} \left\{\widehat{F}_k \neq F_{d_k}\right\}$ and the 'caching error' $\bigcup_{k=1}^{\mathsf{K}} \{|Z_k| > \mathsf{MB}\}$, which is the event that at least one user fetches more bits that allowed by its cache size. Note that this definition of probability of error is not limited to uncoded cache placement (i.e., only the definition of the caching function in (2.10) is actually only valid for uncoded cache placement).

- A pair (M, R) is said to be achievable if there exit placement probability mass function, encoding function and decoding functions such that $\lim_{\mathsf{B}\to\infty} P_{\mathrm{err,d}}^{(\mathsf{B})} = 0$, where $P_{\mathrm{err,d}}^{(\mathsf{B})}$ was defined in (2.11).

- The (worst-case) load with uncoded cache placement is

$$\mathsf{R}^{\star}_{\mathrm{d,u}} := \inf\{\mathsf{R} : \ (\mathsf{M}, \mathsf{R}) \text{ is achievable with uncoded cache placement}\}. \tag{2.12}$$

2.5 The Caching Problem: Achievability of (1.1b) and (1.3), and of (1.5b) and (1.6)

Centralized Cache-aided Systems We start with the description of cMAN. Let $\mathsf{M} = t\frac{\mathsf{N}}{\mathsf{K}}$ for some positive integer $t \in [0 : \mathsf{K}]$. In the placement phase, each file is split into $\binom{\mathsf{K}}{t}$ non-overlapping equal size sub-files of $\frac{\mathsf{B}}{\binom{\mathsf{K}}{t}}$ bits. The sub-files of F_i are denoted by $F_{i,\mathcal{W}}$ for $\mathcal{W} \subseteq [\mathsf{K}]$ where $|\mathcal{W}| = t$. User $k \in [K]$ fills its cache as

$$Z_k = \Big(F_{i,\mathcal{W}} : k \in \mathcal{W},\ \mathcal{W} \subseteq [\mathsf{K}],\ |\mathcal{W}| = t,\ i \in [\mathsf{N}]\Big). \tag{2.13}$$

In the delivery phase, given the demand vector $\mathbf{d}$, for each set of users $\mathcal{J} \subseteq [\mathsf{K}]$ where $|\mathcal{J}| = t + 1$, the server transmits the cMAN multicast messages

$$W_{\mathcal{J}} = \oplus_{k\in\mathcal{J}} F_{d_k,\mathcal{J}\setminus\{k\}}. \tag{2.14}$$

Hence, totally the server broadcasts

$$X_{\mathbf{d}} = \Big(W_{\mathcal{J}} : \mathcal{J} \subseteq [\mathsf{K}], |\mathcal{J}| = t + 1\Big), \tag{2.15}$$

which requires broadcasting $\mathsf{B}\binom{\mathsf{K}}{t+1}/\binom{\mathsf{K}}{t}$ bits. Note that user $k \in \mathcal{J}$, for $\mathcal{J}$ as in (2.15), wants $F_{d_k,\mathcal{J}\setminus\{k\}}$ and knows/cached $F_{d_s,\mathcal{J}\setminus\{s\}}$ for all $s \neq k$, so he can recover $F_{d_k,\mathcal{J}\setminus\{k\}}$ from $X_{\mathbf{d}}$ in (2.15) and the cache content in (2.13). Thus for the cMAN caching scheme the load is given by (1.1b).

The optimal load for centralized cache-aided systems with uncoded cache placement was shown to be achieved essentially following the cMAN original idea, but with the fundamental observation that, of the $\binom{\mathsf{K}}{t+1}$ transmitted linear combinations in (2.15), $\binom{\mathsf{K}-\min(\mathsf{K},\mathsf{N})}{t+1}$ can be obtained as linear combinations of other transmissions. Therefore, by removing these redundant transmissions for the case $\mathsf{N} < \mathsf{K}$, the load becomes the one in (1.3).

Decentralized Cache-aided Systems We now give the description of dMAN. In decentralized systems, coordination during the placement phase is not possible. The dMAN scheme in [5] lets each user cache a subset of MB/N bits of each file, chosen uniformly and independently at random (which guarantees that every users does not violate its cache size limit). Given the cache content of all the users, the bits of the files are naturally grouped into sub-files $F_{i,\mathcal{W}}$, where $F_{i,\mathcal{W}}$ is the set of bits of file $i \in [\mathsf{N}]$ that are only known/cached only by the users in $\mathcal{W} \subseteq [\mathsf{K}]$. It can be shown that

$$\frac{|F_{i,\mathcal{W}}|}{\mathsf{B}} \to \left(\frac{\mathsf{M}}{\mathsf{N}}\right)^{|\mathcal{W}|}\left(1 - \frac{\mathsf{M}}{\mathsf{N}}\right)^{\mathsf{K}-|\mathcal{W}|} \text{ in probability when } \mathsf{B} \to \infty. \tag{2.16}$$

In the delivery phase, for each $t \in [0 : \mathsf{K}-1]$, all the $\binom{\mathsf{K}}{t+1}$ sub-files $(F_{i,\mathcal{W}} : |\mathcal{W}| = t,\ i \in [\mathsf{N}])$ are gathered together; since they all have approximately the same normalized length as in (2.16), the sender uses the cMAN scheme for $\mathsf{M} = t\mathsf{N}/\mathsf{K}$ to deliver them. As a result of the dMAN caching scheme, the load converges in probability when $\mathsf{B} \to \infty$ to

$$\sum_{t\in[0:\mathsf{K}-1]} \binom{\mathsf{K}}{t+1} \left(\frac{\mathsf{M}}{\mathsf{N}}\right)^t \left(1-\frac{\mathsf{M}}{\mathsf{N}}\right)^{\mathsf{K}-t} = \frac{1-\frac{\mathsf{M}}{\mathsf{N}}}{\frac{\mathsf{M}}{\mathsf{N}}} \left[1-\left(1-\frac{\mathsf{M}}{\mathsf{N}}\right)^{\mathsf{K}}\right], \tag{2.17}$$

which coincides with (1.5b). The optimal load for decentralized cache-aided systems with uncoded cache placement can be achieved following the dMAN original idea without the redundant transmissions in the underlying cMAN scheme, which leads to the load in (1.6) (i.e., as in (2.17) but with K replaced by $\min(\mathsf{N}, \mathsf{K})$).

2.6 Mapping the Caching Problem with Uncoded Cache Placement into an Index Coding Problem

Under the constraint of uncoded cache placement, when the cache contents and the demands are fixed, the delivery phase of the caching problem is equivalent to the following IC problem. Denote the set of distinct demanded files in the demand vector $\mathbf{d}$ by $\mathcal{N}(\mathbf{d})$. For each $i \in \mathcal{N}(\mathbf{d})$ and for each $\mathcal{W} \subseteq [\mathsf{K}]$, the sub-file $F_{i,\mathcal{W}}$ (containing the bits of file F_i only within the cache of the users indexed by $\mathcal{W}$) is an independent message in the IC problem with user set $[\mathsf{K}]$. Hence, by using the notation introduced in Sections 2.1 and 2.4, we have $\mathsf{K}' = \mathsf{K}$ and $\mathsf{N}' \leq |\mathcal{N}(\mathbf{d})|(2^K - 1)$. For each user $k \in [\mathsf{K}]$ in this general IC problem, the desired message set and the side information sets are given by

$$\mathcal{D}_k = \left\{F_{d_k,\mathcal{W}} : \mathcal{W} \subseteq [\mathsf{K}], k \notin \mathcal{W}\right\}, \tag{2.18}$$

$$\mathcal{A}_k = \left\{F_{i,\mathcal{W}} : \mathcal{W} \subseteq [\mathsf{K}], i \in \mathcal{N}(\mathbf{d}), k \in \mathcal{W}\right\}. \tag{2.19}$$

Converse bound We want to leverage the acyclic converse bound in Theorem 2 from [35, Corollary 1]. However, in the equivalent multicast IC problem resulting from the caching problem with uncoded cache placement, the converse bound in [35, Corollary 1] is not suitable because valid for the multiple unicast IC case only. Hence, we generate a multiple unicast IC problem from the general IC problem as follows. We choose $|\mathcal{N}(\mathbf{d})| \leq \min(\mathsf{N}, \mathsf{K})$ users with different demands in the user set $[\mathsf{K}]$; if multiple such choices exist, then we consider the intersection of the resulting converse bound regions. The chosen user set is denoted by $\mathcal{C}$. For each $k \in \mathcal{C}$ and each $\mathcal{W} \subseteq [\mathsf{K}]$ such that $k \notin \mathcal{W}$, $F_{d_k,\mathcal{W}}$ is a message in a multiple unicast IC problem and is demanded by a virtual user with the side information identical to the cache content of user k in the caching problem. The inverse of the largest symmetric achievable rate for the this IC problem provides a lower bound for $\mathsf{R}^\star_{\mathrm{c,u}}$ if the cache-aided system is centralized or $\mathsf{R}^\star_{\mathrm{d,u}}$ if the cache-aided system is decentralized. In general, the lower bound derived in this way is not necessarily achievable in the original caching problem. A converse bound derived as just described and based on Theorem 2 will be derived in Section 4.2 for centralized cache-aided systems, and in Section 4.3 for decentralized cache-aided systems.

Achievable bound For the delivery phase in a cache-aided system with uncoded cache placement, which can be seen as a general multicast IC problem, the composite (index) coding achievable bound in Theorem 1 could be used. However, in our conference work [20], [21] we reported that Theorem 1 is not tight in general. In order to overcome this limitation, in Section 3.2, we propose a novel IC achievable bound which will be proved to be strictly better than composite (index) coding.

Chapter 3

Novel Index Coding Achievable Bound

3.1 Chapter Overview and Related Publications

In this chapter, based on Han's coding scheme [48], Slepian-Wolf coding [47] and non-unique decoding, we propose a novel index coding achievable bound. This achievable scheme is shown to strictly outperform composite (index) coding and is, to the best our knowledge, the best random coding achievable bound for the general IC problem to date.

Related Publications

1. Kai Wan, Daniela Tuninetti and Pablo Piantanida, "An Index Coding Approach to Caching with Uncoded Cache Placement", *submitted* to IEEE Trans. on Information Theory, 2017. [53]

2. Kai Wan, Daniela Tuninetti and Pablo Piantanida, "A Novel Index Coding Scheme and its Application to Coded Caching", in Proceedings of IEEE Infor. Theory Application Workshop (ITA), Feb. 2017. [64]

3.2 Novel Index Coding Achievable Bound

We first introduce a novel IC achievable scheme based on coding for the Multi-Access Channel (MAC) with correlated messages [48], Slepian-Wolf coding [47], and non-unique decoding [66]. We then prove that the rate region of the proposed scheme not only strictly includes the region achieved by composite (index) coding in Theorem 1 but it also strictly outperforms the improved version of Theorem 1 from [67].

Intuitively, the improvements in our scheme come from:

- In (2.2b), for each subset $\mathcal{J} \subseteq [\mathsf{N}']$ in the composite (index) coding scheme, the composite index $W_{\mathcal{J}}$ is determined by the messages indexed by $\mathcal{J}$. Thus, composite indices are correlated among themselves. We leverage this correlation to loosen the required rate in the first decoding stage, which is given by (2.2d).

- In the composite (index) coding scheme, decoder $j \in [\mathsf{K}']$ wants to recover uniquely the messages in $\mathcal{K}_j$, and for that only the composite indices $(W_{\mathcal{J}} : \mathcal{J} \subseteq \mathcal{K}_j \cup \mathcal{A}_j)$ are used, as the constraint in (2.2c) indicates. In our proposed scheme, every user uses all the composite messages to uniquely recover the desired messages in the desired set $\mathcal{D}_j$, non-uniquely those in $\mathcal{K}_j \backslash \mathcal{D}_j$, and treats the remaining messages as noise.

Theorem 3 (Novel Achievable Scheme for the General Index Coding Problem)**.** *A non-negative rate tuple* $\mathbf{R} := (R_1, \ldots, R_{\mathsf{N}'})$ *is achievable for the IC problem* $\Big((\mathcal{A}_j, \mathcal{D}_j) : j \in [\mathsf{K}']\Big)$ *with* $\mathsf{N}' = \big|\cup_{j\in[\mathsf{K}']} \mathcal{A}_j \cup \mathcal{D}_j\big|$ *users if*

$$\mathbf{R} \in \bigcap_{j\in[\mathsf{K}']} \bigcup_{\mathcal{K}_j : \mathcal{D}_j \subseteq \mathcal{K}_j \subseteq [\mathsf{N}']\setminus\mathcal{A}_j} \mathscr{R}(\mathcal{K}_j|\mathcal{A}_j, \mathcal{D}_j), \tag{3.1a}$$

$$\mathscr{R}(\mathcal{K}|\mathcal{A}, \mathcal{D}) := \bigcap_{\mathcal{J}:\mathcal{J}\subseteq\mathcal{K}, \mathcal{D}\cap\mathcal{J}\neq\emptyset} \left\{ \sum_{i\in\mathcal{J}} R_i < \kappa_{\mathcal{J}} \right\}, \tag{3.1b}$$

where in (3.1b) $\kappa_{\mathcal{J}}$ *is defined as*

$$\kappa_{\mathcal{J}} := I\Big(\big(U_i : i \in \mathcal{J}\big); \big(X_{\mathcal{P}} : \mathcal{P} \subseteq [\mathsf{N}']\big)\Big|\big(U_i : i \in \mathcal{A}_j \cup \mathcal{K}_j \setminus \mathcal{J}\big)\Big), \tag{3.1c}$$

for some independent auxiliary random variables $(U_i : i \in [\mathsf{N}'])$ *and some functions* $\Big(f_{\mathcal{P}} : \mathcal{P} \subseteq [\mathsf{N}']\Big)$*, such that* $X_{\mathcal{P}} = f_{\mathcal{P}}\Big(\big(U_i : i \in \mathcal{P}\big)\Big)$ *and*

$$H\Big(\big(X_{\mathcal{P}} : \mathcal{P} \subseteq [\mathsf{N}']\big)\Big|\big(U_i : i \in \mathcal{A}_j\big)\Big) \leq \log_2(|\mathcal{X}|), \quad \forall j \in [\mathsf{K}']. \tag{3.1d}$$

The proof of Theorem 3 is given in Appendix A.1.

Corollary 1. *The composite (index) coding region in Theorem 1 is included in Theorem 3.*

Proof. In general, for a set $\mathcal{B} \subseteq [\mathsf{N}']$ and for the auxiliary random variables as defined in Theorem 3, we have

$$\begin{aligned} H\Big(\big(X_{\mathcal{P}} : \mathcal{P} \subseteq [\mathsf{N}']\big)\Big|\big(U_i : i \in \mathcal{B}\big)\Big) &\leq H\Big(\big(X_{\mathcal{P}} : \mathcal{P} \subseteq [\mathsf{N}'], \mathcal{P} \not\subseteq \mathcal{B}\big)\Big) && \text{(3.2a)} \\ &\leq \sum_{\mathcal{P}:\mathcal{P}\subseteq[\mathsf{N}'], \mathcal{P}\not\subseteq\mathcal{B}} H\big(X_{\mathcal{P}}\big) && \text{(3.2b)} \\ &\leq \sum_{\mathcal{P}:\mathcal{P}\subseteq[\mathsf{N}'], \mathcal{P}\not\subseteq\mathcal{B}} S_{\mathcal{P}}, \ \text{ where } S_{\mathcal{P}} = \log_2(|\mathcal{X}_{\mathcal{P}}|). && \text{(3.2c)} \end{aligned}$$

In the following, we choose the auxiliary random variables $(U_i : i \in [\mathsf{N}'])$ and $(X_{\mathcal{P}} : \mathcal{P} \subseteq [\mathsf{N}'])$ such that all the inequality leading to (3.2c) holds with equality for any $\mathcal{B} \subseteq [\mathsf{N}']$, that is, we construct random variables $(X_{\mathcal{P}} : \mathcal{P} \subseteq [\mathsf{N}'])$ that are independent and uniformly distributed, where the alphabet of $X_{\mathcal{P}}$ has support of size $|\mathcal{X}_{\mathcal{P}}| = 2^{S_{\mathcal{P}}}$. With this choice of auxiliary random variables we show that Theorem 3 reduces to Theorem 1.

Let U_i, for $i \in [\mathsf{N}']$, be an independent and equally likely binary vector of length L_i. For all $\mathcal{P} \subseteq [\mathsf{N}']$, let $X_{\mathcal{P}}$ be a binary vector of length $\lfloor S_{\mathcal{P}} \log_2(|\mathcal{X}|)\rfloor$ obtained as a linear code for the collection of bits in $(U_i : i \in \mathcal{P})$. If $L_i \geq \sum_{\mathcal{P}\subseteq[\mathsf{N}']:i\in\mathcal{P}} \lfloor S_{\mathcal{P}} \log_2(|\mathcal{X}|)\rfloor$ for all $i \in [\mathsf{N}']$, then all the linear combinations that determine $X_{\mathcal{P}}$ can be chosen to be independent and therefore all the inequalities leading to (3.2c) holds with equality for such a choice of auxiliary random variables. As a result, we have that the bound in (3.1d) reduces to the one in (2.2d) by using (3.2c) with $\mathcal{B} = \mathcal{A}_j$, and that the bound in (3.1c) reduces to the one

in (2.2c) by using (3.2c) twice, once with $\mathcal{B} = \mathcal{A} \cup \mathcal{K} \setminus \mathcal{J}$ and once with $\mathcal{B} = \mathcal{A} \cup \mathcal{K}$, which is so because

$$\kappa_{\mathcal{J}} = \sum_{\mathcal{P}:\mathcal{P}\subseteq[\mathsf{N}']:\mathcal{P}\not\subseteq(\mathcal{A}\cup\mathcal{K}\setminus\mathcal{J})} S_{\mathcal{P}} - \sum_{\mathcal{P}:\mathcal{P}\subseteq[\mathsf{N}']:\mathcal{P}\not\subseteq(\mathcal{A}\cup\mathcal{K})} S_{\mathcal{P}} = \sum_{\mathcal{P}:\mathcal{P}\subseteq\mathcal{A}\cup\mathcal{K}:\mathcal{P}\cap\mathcal{J}\neq\emptyset} S_{\mathcal{P}}.$$

This concludes the proof. □

In the rest of this section, we give an example to show that the proposed scheme in Theorem 3 is strictly better than the composite (index) coding in Theorem 1 and its improvement from [67]

Example 1. Consider a multiple unicast IC problem with $K = 6$ equal rate messages and with

$$\begin{aligned}
\mathcal{D}_1 &= \{1\}, \quad \mathcal{A}_1 = \{3,4\},\\
\mathcal{D}_2 &= \{2\}, \quad \mathcal{A}_2 = \{4,5\},\\
\mathcal{D}_3 &= \{3\}, \quad \mathcal{A}_3 = \{5,6\},\\
\mathcal{D}_4 &= \{4\}, \quad \mathcal{A}_4 = \{2,3,6\},\\
\mathcal{D}_5 &= \{5\}, \quad \mathcal{A}_5 = \{1,4,6\},\\
\mathcal{D}_6 &= \{6\}, \quad \mathcal{A}_6 = \{1,2\}.
\end{aligned}$$

Composite (Index) Coding Achievable Bound In [67, Example 1] the authors showed that the largest symmetric rate with the composite (index) coding achievable bound in Theorem 1 for this problem is $R_{\text{sym,cc}} = 0.2963 \cdot \log_2(|\mathcal{X}|)$. In the same paper, the authors proposed an extension of the composite (index) coding idea (see [67, Section III.B]) and showed that this extended scheme for this problem gives $R_{\text{sym,enhanced cc}} = 0.2987 \cdot \log_2(|\mathcal{X}|)$.

Converse Give message M_5 as additional side information to receiver 1 so that the new side information set satisfied $\{3,4,5\} \subset \mathcal{A}_2$. With this receiver 1, in addition to message 1, can decode message 2 and then message 6 for a total of 3 messages. Thus

$$3R_{\text{sym}} \leq \lim_{n\to\infty} \frac{1}{n} H(X^n) \leq \log_2(|\mathcal{X}|), \tag{3.3}$$

where R_{sym} denotes the symmetric rate. Next we show that $R_{\text{sym}} \leq 1/3 \cdot \log_2(|\mathcal{X}|)$ is tight. This shows the strict sub-optimality of composite (index) coding and its extension.

Achievability Take the messages to be binary digits. It is not difficult to see that all users can be satisfied by the transmission of the three coded bits $X = (M_1 \oplus M_3 \oplus M_4,\ M_2 \oplus M_4 \oplus M_5,\ M_1 \oplus M_2 \oplus M_6)$.

We now map this scheme into a choice of auxiliary random variables in our novel IC scheme. Let each file be an independent bit, $\mathcal{K}_j = \mathcal{D}_j$ for $j \in [6]$, and

$$U_1 = M_1,\ U_2 = M_2, \cdots, U_6 = M_6, \tag{3.4}$$

$$\text{for all } \mathcal{P} \subseteq [6] \text{ set } X_{\mathcal{P}} = 0 \text{ except} \tag{3.5}$$

$$X_{\{1,3,4\}} = U_1 \oplus U_3 \oplus U_4, \tag{3.6}$$
$$X_{\{2,4,5\}} = U_2 \oplus U_4 \oplus U_5, \tag{3.7}$$
$$X_{\{1,2,6\}} = U_1 \oplus U_2 \oplus U_6, \tag{3.8}$$

the transmitted signal is

$$X = (X_{\{1,3,4\}}, X_{\{2,4,5\}}, X_{\{1,2,6\}}). \tag{3.9}$$

Here $\mathcal{X} = \text{GF}(2^3)$ so one channel use corresponds to three bits. From (3.1c), we have that for example the rate bound corresponding to receiver 5 is

$$\begin{aligned} R_{\text{sym}} &\leq I(U_5; U_1 \oplus U_3 \oplus U_4, U_2 \oplus U_4 \oplus U_5, U_1 \oplus U_2 \oplus U_6 | U_1, U_4, U_6) && \text{(3.10a)} \\ &= I(U_5; U_3, U_2 \oplus U_5, U_2) && \text{(3.10b)} \\ &= I(U_5; U_2, U_3, U_5) && \text{(3.10c)} \\ &= I(U_5; U_5) && \text{(3.10d)} \\ &= H(U_5) && \text{(3.10e)} \\ &= 1/3 \cdot \log_2(|\mathcal{X}|), && \text{(3.10f)} \end{aligned}$$

and similarly for all the other users. As a result, $R_{\text{sym}} = 1/3 \cdot \log_2(|\mathcal{X}|)$ is achievable by the proposed scheme and coincides with the converse bound. □

Chapter 4

Optimal Converse Bound under the Constraint of Uncoded Cache Placement for Cache-aided Shared-link Broadcast Networks

4.1 Chapter Overview and Related Publications

This chapter analyses the cache-aided shared-link broadcast networks. After an uncoded cache placement, the deliver phase is equivalent to an index coding problem. In Section 4.2 and 4.3, we leverage the acyclic index coding converse bound and the proposed index coding achievable bound, for both centralized and decentralized caching systems. We derive a converse bound on the load cache-aided systems under the constraint of uncoded cache placement. When $\mathsf{N} \geq \mathsf{K}$, we prove that cMAN (resp. dMAN) coincides with the proposed converse bound; while when $\mathsf{N} < \mathsf{K}$, the converse bound is attained by using the proposed novel IC scheme in the delivery phase with the same cMAN (resp. dMAN) placement phase.

Related Publications

1. Kai Wan, Daniela Tuninetti and Pablo Piantanida, "An Index Coding Approach to Caching with Uncoded Cache Placement", *submitted* to IEEE Trans. on Information Theory, 2017. [53]

2. Kai Wan, Daniela Tuninetti and Pablo Piantanida, "On the Optimality of Uncoded Cache Placement", in Proceedings of the IEEE Information Theory Workshop (ITW), pp. 161-165, Sep. 2016. [20]

3. Kai Wan, Daniela Tuninetti and Pablo Piantanida, "On Caching with More Users than Files", in IEEE Int. Symp. Inf . Theory (ISIT), pp. 135-139, Jul. 2016. [21]

4. Kai Wan, Daniela Tuninetti and Pablo Piantanida, "A Novel Index Coding Scheme and its Application to Coded Caching", in Proceedings of IEEE Infor. Theory Application Workshop (ITA), Feb. 2017. [64]

4.2 Centralized Cache-aided Systems with Uncoded Cache Placement

In this section, we leverage the connection between caching and IC problems outlined in Section 2.6 to investigate the fundamental limits of centralized cache-aided systems with uncoded cache placement. We first derive a converse bound (by using Theorem 2 in Section 2.3), and then an achievable bound (by using Theorem 3 in Section 2.2) that matches the converse bound.

Recall that we denote the optimal load for centralized systems and decentralized systems by $\mathsf{R}_{\mathrm{c}}^{\star}$ and $\mathsf{R}_{\mathrm{d}}^{\star}$, respectively. In addition, the loads for centralized systems and decentralized systems under the constraint of uncoded cache placement (see Definition 1) are denoted by $\mathsf{R}_{\mathrm{c,u}}^{\star}$ and $\mathsf{R}_{\mathrm{d,u}}^{\star}$, respectively.

4.2.1 Novel Converse Bound under the Constraint of Uncoded Cache Placement

In this section, we prove our converse bound on the load of centralized cache-aided systems under the constraint of uncoded cache placement, which we first presented in [20], [21]. The main result of this section is stated next.

Theorem 4. *In centralized cache-aided systems the load $\mathsf{R}_{c,u}^{\star}$ satisfies*

$$\mathsf{R}_{c,u}^{\star} \geq \frac{\binom{\mathsf{K}}{q+1} - \binom{\mathsf{K}-\min(\mathsf{K},\mathsf{N})}{q+1}}{\binom{\mathsf{K}}{q}} + s_q \left(\frac{\mathsf{KM}}{\mathsf{N}} - q \right), \tag{4.1a}$$

$$s_q := \frac{\binom{\mathsf{K}}{q+1} - \binom{\mathsf{K}-\min(\mathsf{K},\mathsf{N})}{q+1}}{\binom{\mathsf{K}}{q}} - \frac{\binom{\mathsf{K}}{q} - \binom{\mathsf{K}-\min(\mathsf{K},\mathsf{N})}{q}}{\binom{\mathsf{K}}{q-1}}, \quad \forall q \in [\mathsf{K}]. \tag{4.1b}$$

Moreover, this converse bound is a piecewise linear curve with corner points

$$(\mathsf{M},\mathsf{R}) = \left(t\frac{\mathsf{N}}{\mathsf{K}}, \frac{\binom{\mathsf{K}}{t+1} - \binom{\mathsf{K}-\min(\mathsf{K},\mathsf{N})}{t+1}}{\binom{\mathsf{K}}{t}} \right), \quad \forall t \in [0:\mathsf{K}]. \tag{4.1c}$$

Before we proceed to prove the general converse bound in Theorem 4, we give a specific example. This example introduces the main ideas in the proof.

Example 2. The reasoning in this example applies to the general case $\mathsf{K} \leq \mathsf{N}$. Assume that the server has $\mathsf{N} = \mathsf{K} = 3$ files, denoted as (F_1, F_2, F_3). The total file length in number of bits is $\sum_{i\in[3]} |F_i| = \mathsf{NB} = 3\mathsf{B}$. The total cache size in number of bits is $\sum_{i\in[3]} |Z_i| \leq \mathsf{KMB} = 3\mathsf{MB}$, for some $\mathsf{M} \in [0,\mathsf{N}] = [0,3]$. After the uncoded cache placement phase is done, each file F_i can be thought as having been divided into $2^{\mathsf{K}} = 2^3 = 8$ disjoint sub-files, denoted as $(F_{i,\mathcal{W}} : \mathcal{W} \subseteq [\mathsf{K}] = [3],\ i \in [\mathsf{N}] = [3])$ where $F_{i,\mathcal{W}}$ has been cached only by the users in $\mathcal{W}$. For simplicity in the following we omit the braces when we indicate sets, i.e., $F_{1,12}$ represents $F_{1,\{1,2\}}$, which dos not create any confusion.

For each demand vector $\mathbf{d} \in [N]^K = [3]^3$, we generate an IC problem with $|\mathcal{N}(\mathbf{d})|2^{\mathsf{K}-1} = 12$ independent messages; these messages are $(F_{d_i,\mathcal{W}} : i \notin \mathcal{W},\ i \in [\mathsf{K}])$ and represents the sub-files demanded by the users in $[\mathsf{K}]$ but not available in their caches. For this IC problem, we generate a directed graph with 12 vertices as follows. Each vertex corresponds to a different sub-file. We denote the user in the cache-aided system who wants the sub-file represented by vertex j by h_j. There is a directed edge from vertex i to vertex

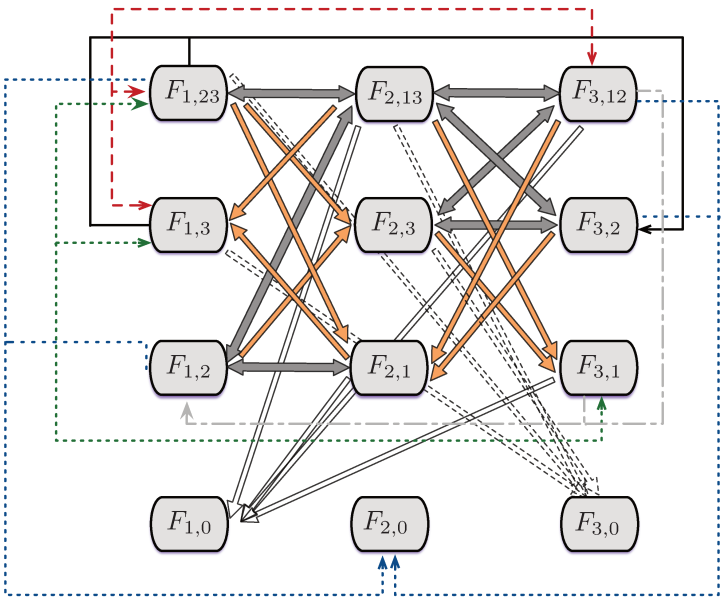

FIGURE 4.1: Directed graph for the equivalent IC problem to the caching problem with $\mathsf{N} = \mathsf{K} = 3$ and with demand vector $\mathbf{d} = (1, 2, 3)$.

j if user h_j knows the sub-file represented by vertex i. For example, Fig. 4.1 shows the directed graph representing this IC problem for the demand vector $\mathbf{d} = (1, 2, 3)$.

Consider the demand vector $\mathbf{d} = (d_1, d_2, d_3)$, where $d_i \in [\mathsf{N}] = [3]$, $i \in [\mathsf{K}] = [3]$. In order to apply Theorem 2, in the constructed directed graph we want to find sets of vertices $\mathcal{J}$ that do not form a directed cycle. No receiver has stored $F_{1,\emptyset}, F_{2,\emptyset}, F_{3,\emptyset}$, so there is no outgoing edge from $F_{1,\emptyset}, F_{2,\emptyset}, F_{3,\emptyset}$ to any other vertex in the graph. Therefore, $F_{1,\emptyset}, F_{2,\emptyset}, F_{3,\emptyset}$ are always in the such sets $\mathcal{J}$ when we evaluate (2.3).

We focus next on demand vectors $\mathbf{d}$ with distinct demands, that is, $|\mathcal{N}(\mathbf{d})| = \min(\mathsf{N}, \mathsf{K}) = \mathsf{K} = 3$; the worst case demand may not be in such a set of demand vectors, but this is not a problem as we aim to derive a lower bound on the load at this point. For a demand vector $\mathbf{d}$ with distinct demands, consider now a permutation $\mathbf{u} = (u_1, u_2, u_3)$ of $[\mathsf{K}] = [3]$. For each such a $\mathbf{u}$, a set of nodes not containing a cycle is as follows: $F_{d_{u_1},\mathcal{W}_1}$ for all $\mathcal{W}_1 \subseteq [\mathsf{K}]\backslash\{u_1\}$, and $F_{d_{u_2},\mathcal{W}_2}$ for all $\mathcal{W}_2 \subseteq [\mathsf{K}]\backslash\{u_1, u_2\}$, and $F_{d_{u_3},\mathcal{W}_3}$ for all $\mathcal{W}_3 \subseteq [\mathsf{K}]\backslash\{u_1, u_2, u_3\} = \emptyset$.

For example, when $\mathbf{d} = (1, 2, 3)$ and $\mathbf{u} = (1, 3, 2)$, we have

$$d_{u_1} = d_1 = 1; \mathcal{W}_1 \subseteq [\mathsf{K}]\backslash\{u_1\} = [3]\backslash\{1\} = \{2, 3\}, \tag{4.2}$$

$$d_{u_2} = d_3 = 3; \mathcal{W}_2 \subseteq [\mathsf{K}]\backslash\{u_1, u_2\} = [3]\backslash\{1, 3\} = \{2\}, \tag{4.3}$$

$$d_{u_3} = d_2 = 2; \mathcal{W}_3 \subseteq [\mathsf{K}]\backslash\{u_1, u_2, u_3\} = \emptyset, \tag{4.4}$$

and the corresponding set not containing a cycle is $(F_{1,\emptyset}, F_{1,2}, F_{1,3}, F_{1,23}, F_{3,\emptyset}, F_{3,2}, F_{2,\emptyset})$, as it can be easily verified by inspection of Fig. 4.1. From (2.3), we have that this acyclic set of nodes can be used to write the bound

$$\mathsf{R}^{\star}_{\mathrm{c,u}} \geq \frac{|F_{1,\emptyset}| + |F_{1,2}| + |F_{1,3}| + |F_{1,23}| + |F_{3,\emptyset}| + |F_{3,2}| + |F_{2,\emptyset}|}{\mathsf{B}}. \tag{4.5}$$

In general, when $\mathsf{K} \leq \mathsf{N}$ we can find a bound such as the one in (4.5) for all possible pairs $\mathbf{d} \in \mathrm{Perm}(\mathsf{N}, \mathsf{K})$

and $\mathbf{u} \in \text{Perm}(\mathsf{K},\mathsf{K})$, where $\text{Perm}(n,k)$ denotes the set of all $k-$permutations of n elements (there are $\frac{n!}{(n-k)!}$ elements in the set $\text{Perm}(n,k)$ for $n \geq k$). If we sum all the $|\text{Perm}(\mathsf{N},\mathsf{K})||\text{Perm}(\mathsf{K},\mathsf{K})| = \binom{\mathsf{N}}{\mathsf{K}}(\mathsf{K}!)^2 = (3!)^2 = 36$ inequalities as in (4.5), we get

$$\mathsf{R}^{\star}_{\text{c,u}} \geq \frac{1}{(3!)^2} \sum_{\mathbf{d}\in\text{Perm}(3,3)} \sum_{\mathbf{u}\in\text{Perm}(3,3)} \sum_{j\in[3]} \sum_{\mathcal{W}_j\subseteq[3]\setminus\{u_1,\ldots,u_j\}} \frac{|F_{d_{u_j},\mathcal{W}_j}|}{\mathsf{B}} \tag{4.6a}$$

$$= \sum_{t\in[0:3]} x_t \frac{\binom{3}{t+1}}{\binom{3}{t}} \tag{4.6b}$$

$$= \sum_{t\in[0:3]} x_t \frac{3-t}{1+t} \tag{4.6c}$$

$$= 3\cdot x_0 + 1\cdot x_1 + \frac{1}{3}\cdot x_2 + 0\cdot x_3, \tag{4.6d}$$

where x_t in (4.6b) is defined as

$$0 \leq x_t := \sum_{j\in[\mathsf{N}]} \sum_{\mathcal{W}\subseteq[\mathsf{K}]:|\mathcal{W}|=t} \frac{|F_{j,\mathcal{W}}|}{\mathsf{NB}}, \quad t \in [0:\mathsf{K}], \tag{4.7}$$

is the fraction of the total number of bits across all files that are known/cached only by a subset of $t \in [0:\mathsf{K}] = [0:3]$ users.

The general proof of (4.6b) can be found in (4.16b). At this point we can offer the following intuitive interpretation for the case $\mathsf{K} \leq \mathsf{N}$, as it is the case in this example. The total number of sub-files cached by a subset of $t \in [0:\mathsf{K}]$ users is $\mathsf{N}\binom{\mathsf{K}}{t}$, where the factor $\binom{\mathsf{K}}{t}$ appears at the denominator of (4.6b) (here $\mathsf{K}=3$), and the factor N at the denominator of (4.7). The total number of sub-files cached by a subset of $t \in [0:\mathsf{K}]$ users in the bound in (4.5) (by the symmetry of the problem, all the other bound have the same structure) is

$$\sum_{i\in[\mathsf{K}]} \sum_{\mathcal{W}_i\subseteq[\mathsf{K}]\setminus\{u_1,\ldots,u_i\}} 1_{\{|\mathcal{W}_i|=t\}} = \binom{\mathsf{K}-1}{t} + \binom{\mathsf{K}-2}{t} + \ldots + \binom{t}{t} \tag{4.8a}$$

$$= \binom{\mathsf{K}}{t+1}, \tag{4.8b}$$

where $1_{\{\mathcal{A}\}}$ in the indicator function that is equal to one if and only if the condition in $\mathcal{A}$ is true, and where we have used the Pascal's triangle identity; the factor $\binom{\mathsf{K}}{t+1}$ (here $\mathsf{K}=3$) appears at the numerator of (4.6b).

In addition to the bounds in (4.6) and (4.7), we also have that the total number of bits in the files is

$$\sum_{j\in[\mathsf{N}]} \sum_{\mathcal{W}\subseteq[\mathsf{K}]} |F_{j,\mathcal{W}}| = \mathsf{NB} \iff \sum_{t\in[\mathsf{K}]} x_t = 1, \tag{4.9}$$

and that the total number of bits in the caches must satisfy

$$\sum_{j\in[\mathsf{N}]} \sum_{\mathcal{W}\subseteq[\mathsf{K}]:j\in\mathcal{W}} |F_{j,\mathcal{W}}| \leq \mathsf{KMB} \iff \sum_{t\in[\mathsf{K}]} t\, x_t \leq \frac{\mathsf{KM}}{\mathsf{N}}. \tag{4.10}$$

Note that the bound in (4.9) is the total number of bits across all files, which is a looser constraint than

imposing that each file contains the same number of bits; similarly, the bound in (4.10) is the total number of bits across all caches, which is a looser constraint than imposing that each cache has the same size; none of these is a problem as we aim to derive a lower bound on the load at this point. This implies that that the lower bound we derive applies to the case where the total number of bits across all files and the total number of bits across all caches are constrained, but not the size each individual file or each individual cache.

The constraints in (4.6)-(4.10) provide a converse bound for the load $\mathsf{R}^{\star}_{\text{c,u}}$ with uncoded cache placement. Since there are many inequalities in $\mathsf{K}+1=4$ unknowns, we proceed to eliminate (x_0, x_1, x_2, x_3) in the system of inequalities in (4.6)-(4.10). By doing so, we obtain

$$\mathsf{R}^{\star}_{\text{c,u}} \overset{\text{by eq.(4.6d)}}{\geq} 3x_0 + x_1 + \frac{1}{3}x_2 \tag{4.11a}$$

$$\overset{\text{by eq.(4.9)}}{=} 3(1 - x_1 - x_2 - x_3) + x_1 + \frac{1}{3}x_2 \tag{4.11b}$$

$$\overset{\text{by eq.(4.10)}}{\geq} 3 + 2(2x_2 + 3x_3 - \mathsf{M}) - \frac{8}{3}x_2 - 3x_3 \tag{4.11c}$$

$$\overset{\text{by eq.(4.7)}}{\geq} 3 - 2\mathsf{M}. \tag{4.11d}$$

Similarly, we can obtain

$$\mathsf{R}^{\star}_{\text{c,u}} \geq -\frac{2}{3}\mathsf{M} + \frac{5}{3}, \tag{4.12}$$

$$\mathsf{R}^{\star}_{\text{c,u}} \geq -\frac{1}{3}\mathsf{M} + 1. \tag{4.13}$$

The maximum among the right-hand sides of (4.11d), (4.12) and (4.13) give a piecewise linear curve with corner points: $(0,3), (1,1), (2,\frac{1}{3}), (3,0)$. Since these corner points are achieved by $\mathsf{R}_{\text{c,u,MAN}}[t]$ for $t \in [0:3]$, in (1.1b), we conclude that the two-phase strategy in [4] is optimal under the constraint of uncoded cache placement in this case. Note that this shows that demand vectors with distinct demands lead to the worst case load. □

We are now ready to extend the reasoning in Example 2 to a general setting, where not necessarily $\mathsf{K} \leq \mathsf{N}$.

Proof of Theorem 4. Consider a system with uncoded cache placement and a demand vector with $\min(\mathsf{K}, \mathsf{N})$ distinct demanded files. We treat the delivery phase of this caching scheme as an IC problem, as described in Section 2.6. We derive a converse bound on $\mathsf{R}^{\star}_{\text{c,u}}$ using Theorem 2. A directed graph can be generated for such IC problem as described in Section 2.6; Lemma 2 in Appendix A.4 gives the sets of nodes not containing a directed cycle as follows. Let $\mathbf{u} = (u_1, u_2, \ldots, u_{\min(\mathsf{K},\mathsf{N})})$ be a permutation of $\mathcal{C}$, where $\mathcal{C}$ is the chosen user set with different demands. A set of nodes not containing a directed cycle in the directed graph of the corresponding IC problem can be composed of sub-files

$$\big(F_{d_{u_i},\mathcal{W}_i} : \mathcal{W}_i \subseteq [\mathsf{K}] \backslash \{u_1, \ldots, u_i\},\ i \in [\min(\mathsf{K}, \mathsf{N})]\big). \tag{4.14}$$

Therefore, the bound in (2.3) reads

$$\mathsf{R}^{\star}_{\mathrm{c,u}} \geq \sum_{i\in[\min(\mathsf{K},\mathsf{N})]} \sum_{\mathcal{W}_i\subseteq[\mathsf{K}]\setminus\{u_1,\ldots,u_i\}} \frac{|F_{d_{u_i},\mathcal{W}_i}|}{\mathsf{B}}. \tag{4.15}$$

Note that, in the bound in (4.15), the number of $F_{k,\mathcal{W}}$'s such that $|\mathcal{W}| = t$ is $\sum_{i\in[\min(\mathsf{K},\mathsf{N})]}\binom{\mathsf{K}-i}{t}$, while the total number of $F_{k,\mathcal{W}}$'s such that $|\mathcal{W}| = t$ is $\mathsf{N}\binom{\mathsf{K}}{t}$. By considering all sets $\mathcal{C}$ of users with $\min(\mathsf{K},\mathsf{N})$ different demands, and all the permutations $\mathbf{u}$ of $\mathcal{C}$, we can list all the inequalities in the form of (4.15) and sum them together to obtain

$$\mathsf{R}^{\star}_{\mathrm{c,u}} \geq \sum_{t\in[0:\mathsf{K}]} \frac{\binom{\mathsf{K}-1}{t}+\binom{\mathsf{K}-2}{t}+\cdots+\binom{\mathsf{K}-\min(\mathsf{K},\mathsf{N})}{t}}{\binom{\mathsf{K}}{t}}\, x_t \tag{4.16a}$$

$$= \sum_{t\in[0:\mathsf{K}]} \frac{\binom{\mathsf{K}}{t+1}-\binom{\mathsf{K}-\min(\mathsf{K},\mathsf{N})}{t+1}}{\binom{\mathsf{K}}{t}}\, x_t, \tag{4.16b}$$

where from (4.16a) to (4.16b) we used the Pascal's triangle equality, where the set of coefficients $(x_0,\ldots,x_{\mathsf{K}})$ as defined in (4.7) can be interpreted as a probability mass function (see (4.9)) subject to the first-moment constraint given in (4.10).

Next, by Fourier-Motzkin elimination of x_q and x_{q-1} in (4.16b) for each $q\in[\mathsf{K}]$ (see Appendix A.2), we obtain the bound given in (4.1a). The bound in (4.1a) is a family of straight lines parameterized by $q\in[\mathsf{K}]$. The lines for $q=t$ and $q=t-1$ intersect at the point in (4.1c) because the coefficients $c_t := \frac{\binom{\mathsf{K}}{t+1}-\binom{\mathsf{K}-\min(\mathsf{K},\mathsf{N})}{t+1}}{\binom{\mathsf{K}}{t}}$ decreases monotonously in $t\in[\mathsf{K}-1]$. This concludes the proof. □

Remark 1. *In [19], the authors propose a genie-aided converse bound to arrive to the very same inequalities as in* (4.15)*, which were originally derived in [20], [21] by leveraging the IC acyclic converse bound. These two approaches are equivalent. Firstly, the acyclic converse bound can be proved by providing genie information to the receivers in the acyclic set. Secondly, the following steps in [19] are also the same as in our original work [20], [21], namely, summing all the inequalities, bounding the load by* $(x_t : t\in[0:\mathsf{K}])$ *defined in* (4.7)*, and eliminating the* x_t*'s to get the final bound. The only difference is that we used Fourier-Motzkin elimination while the authors in [19] treated* $(x_t : t\in[0:\mathsf{K}])$ *as a probability mass function and optimized the bound over all probability mass functions (see* (4.9)*) with a given constraint on the the first moment (see* (4.10)*).*

Remark 2. *Our proposed converse bound can be generalized to different memory sizes* $(\mathsf{M}_1,\ldots,\mathsf{M}_{\mathsf{K}})$ *or file lengths* $(\mathsf{B}_1,\ldots,\mathsf{B}_{\mathsf{N}})$*, where where the memory size of user* $i\in[\mathsf{K}]$ *is* M_i *and the file size of file* $j\in[\mathsf{N}]$ *is* L_j*, or average load. In the above cases, we define the optimal load under the constraint of uncoded cache placement as* $R_{c,low}$*, where* $R_{c,low}$ *is lower bounded as*

$$\sum_{\mathbf{d}\in[\mathsf{N}]^{\mathsf{K}}} \Pr[\mathbf{d}]\mathsf{R}_{\mathbf{d}} \leq \mathsf{R}_{c,low} \text{ (in case of average load)}, \tag{4.17}$$

or as

$$\mathsf{R}_{\mathbf{d}} \leq \mathsf{R}_{c,low} \text{ (in case worst-case load)}, \tag{4.18}$$

where we can optimize over the lengths of the subfiles subject to

$$\sum_{\mathcal{W}\subseteq[\mathsf{K}]} |F_{j,\mathcal{W}}| = \mathsf{B}_j,\ \forall j \in [\mathsf{N}],\ \textit{(file length)}, \tag{4.19}$$

$$\sum_{j\in[\mathsf{N}]} \sum_{\mathcal{W}\subseteq[\mathsf{K}]:u\in\mathcal{W}} |F_{j,\mathcal{W}}| \le \mathsf{M}_u,\ \forall u \in [\mathsf{K}],\ \textit{(cache size)}, \tag{4.20}$$

$$\sum_{i\in[\min(\mathsf{K},\mathsf{N})]} \sum_{\mathcal{W}_i\subseteq[\mathsf{K}]\setminus\{u_1,\ldots,u_i\}} |F_{d_{u_i},\mathcal{W}_i}| \le \mathsf{R}_{\mathbf{d}},\ \forall \mathbf{d} \in [\mathsf{N}]^{\mathsf{K}},$$

$$\forall \textit{permutation } \mathbf{u} \textit{ of each largest user set with distinct demands (acyclic outer bound)}. \tag{4.21}$$

4.2.2 Optimality of the Proposed Converse Bound

In this section, we prove that our converse bound in Theorem 4 is indeed achievable.

Theorem 5. *The caching converse bound in Theorem 4 is achievable by the cMAN cache placement and a delivery scheme based on the IC achievable bound in Theorem 3.*

Proof. For centralized cache-aided systems under the constraint of uncoded cache placement, for $\mathsf{N} \ge \mathsf{K}$ the claim is true because the converse bound in (4.1c) coincides with the cMAN scheme in (1.1b), and for $\mathsf{N} < \mathsf{K}$ because Theorem 3 achieves (4.1c) as showed next.

We use the same placement phase as cMAN for $\mathsf{M} = t\frac{\mathsf{N}}{\mathsf{K}}$, for $t \in [0:\mathsf{K}]$, so that the delivery phase is equivalent to an IC problem with K users in which each sub-file $F_{i,\mathcal{W}}$, for $i \in \mathsf{N}(\mathbf{d})$, $\mathcal{W} \subseteq [\mathsf{K}]$ and $|\mathcal{W}| = t$, is an independent message, and where the desired message and side information sets are given by (2.18) and (2.19), respectively. Note that the message rates in this equivalent IC problem are identical by construction and the number of messages for the worst case-load is $\mathsf{N}' = \min(\mathsf{N},\mathsf{K})\binom{\mathsf{K}}{t}$.

In Theorem 3, following Example 1, we let $\mathcal{K}_j = \mathcal{D}_j$ for $j \in [\mathsf{K}]$, we represent $F_{i,\mathcal{W}}$ as a binary vector of length $\mathsf{B}/\binom{\mathsf{K}}{t}$ bits (assumed to be an integer without loss of generality) and we let the corresponding random variable U to be equal to the message. We also let $X_{\mathcal{P}}$ to be non zero only for the linear combinations of messages sent by the scheme in [4]. With this we have $\mathsf{R}_{\text{sym}} = H(U) = \mathsf{B}/\binom{\mathsf{K}}{t}$ and

$$\log_2(|\mathcal{X}|) = H(X) = \frac{\mathsf{B}}{\binom{\mathsf{K}}{t}}\left(\binom{\mathsf{K}}{t+1} - \binom{\mathsf{K}-\min(\mathsf{N},\mathsf{K})}{t+1}\right), \tag{4.22}$$

so the symmetric rate is

$$\mathsf{R}_{\text{sym}} = \frac{1}{\binom{\mathsf{K}}{t+1} - \binom{\mathsf{K}-\min(\mathsf{N},\mathsf{K})|}{t+1}} \log_2(|\mathcal{X}|)\ \left[\frac{\text{bits}}{\text{ch.use}}\right]. \tag{4.23}$$

Each receiver in the original caching problem is interested in recovering $\binom{\mathsf{K}}{t}$ messages, or one file of B bits, thus the 'sum-rate rate' delivered to each user is

$$\mathsf{R}_{\text{sum-rate}} = \frac{\binom{\mathsf{K}}{t}}{\binom{\mathsf{K}}{t+1} - \binom{\mathsf{K}-\min(\mathsf{N},\mathsf{K})}{t+1}} \log_2(|\mathcal{X}|)\ \left[\frac{\text{bits}}{\text{ch.use}}\right]. \tag{4.24}$$

The load in the caching problem is the number of transmissions (channel uses) needed to deliver one file to each user, thus the inverse of $\mathsf{R}_{\text{sum-rate}}$ for $|\mathcal{X}| = 2$ indeed corresponds to the load in (1.3). □

Remark 3. *For the case* $\mathsf{N} \geq \mathsf{K}$*, the claim of Theorem 5 is trivially obvious because the converse bound in* (4.1c) *coincides with the cMAN scheme in* (1.1b)*, as already pointed out in the Introduction. Our proof for the case* $\mathsf{N} < \mathsf{K}$ *uses Theorem 3 and gives an interpretation of the achievable scheme proposed in [19] via the framework of 'source coding with side information.' Our approach has the advantage that it applies to any IC scheme, and when applied to the caching problem is not limited to binary linear codes for the specific cMAN placement as that of [19].*

4.3 Decentralized Cache-aided Systems with Uncoded Placement

In this section we consider decentralized cache-aided systems with uncoded cache placement. The delivery phase in the achievable scheme for such systems can be divided into several independent delivery phases of a centralized caching scheme, each one dealing with the sub-files known/cached by the same number of users, as originally proposed in dMAN.

In the following, we give the modelization of a decentralized caching scheme. Given a cache realization as a result of an uncoded placement function as defined in (2.10), let $X_{j,b,u} = 1$ if user $u \in [\mathsf{K}]$ cached bit $b \in [\mathsf{B}]$ of file $j \in [\mathsf{N}]$, and zero otherwise. The number of bits cached by user $u \in [\mathsf{K}]$ is

$$|Z_u| = \sum_{j\in[\mathsf{N}]} \sum_{b\in[\mathsf{B}]} X_{j,b,u}. \tag{4.25}$$

The number of bits of file $j \in [\mathsf{N}]$ cached only by the users in $\mathcal{W} \subseteq [\mathsf{K}]$ is

$$|F_{j,\mathcal{W}}| = \sum_{b\in[\mathsf{B}]} \left(\prod_{u\in\mathcal{W}} X_{j,b,u} \right) \left(\prod_{k\notin\mathcal{W}} (1 - X_{j,b,k}) \right). \tag{4.26}$$

Define

$$Y_t := \frac{1}{\mathsf{NB}} \max_{j\in[\mathsf{N}],\ \mathcal{W}\subseteq[\mathsf{K}]:|\mathcal{W}|=t} \{|F_{j,\mathcal{W}}|\},\ t \in [0:\mathsf{K}], \tag{4.27}$$

which is the length (normalized by NB) of the longest sub-files needed for cMAN-type linear combinations as in (2.15) involving the sub-files only cached by t users. Similarly define

$$X_t := \frac{1}{\mathsf{NB}} \sum_{j\in[\mathsf{N}]} \sum_{\mathcal{W}\subseteq[\mathsf{K}]:|\mathcal{W}|=t} |F_{j,\mathcal{W}}|,\ t \in [0:\mathsf{K}]. \tag{4.28}$$

Clearly $X_t \leq \mathsf{N}\binom{\mathsf{K}}{t} Y_t,\ t \in [0:\mathsf{K}]$, with equality if and only if $|F_{j,\mathcal{W}}|$ only depends on $t = |\mathcal{W}|$, but neither $j \in [\mathsf{N}]$ nor $\mathcal{W} \subseteq [\mathsf{K}]$. Based on the same reasoning we used for the centralized scheme in Section 4.2, the right hand side of (4.16b) with x_t replaced by X_t provides a lower bound on the load, and with x_t replaced by $\mathsf{N}\binom{\mathsf{K}}{t} Y_t$ provides an upper bound on the load.

For the probability of error in (2.11) we have that the 'caching error' event is

$$\mathcal{E}_{\text{caching error}} := \bigcup_{k=1}^{\mathsf{K}} \left\{ \frac{|Z_u|}{\mathsf{NB}} = \frac{1}{\mathsf{NB}} \sum_{j\in[\mathsf{N}]} \sum_{b\in[\mathsf{B}]} X_{j,b,u} > \frac{\mathsf{M}}{\mathsf{N}} \right\}; \tag{4.29}$$

and the 'decoding error' is

$$\mathcal{E}_{\text{decoding error}} := \bigcup_{k=1}^{\mathsf{K}} \left\{ \widehat{F}_k \neq F_{d_k} \right\}; \tag{4.30}$$

with this, the probability of error can be bounded as

$$P_{\text{err,d}}^{(\mathsf{B})} \leq \Pr[\mathcal{E}_{\text{caching error}}] + \Pr[\mathcal{E}_{\text{decoding error}} | \mathcal{E}_{\text{caching error}}^{c}]. \tag{4.31}$$

Assume there is an uncoded cache placement in (2.10) such that:

A1: $\Pr\left[\bigcup_{k=1}^{\mathsf{K}} \left\{ \frac{1}{\mathsf{NB}} \sum_{j\in[\mathsf{N}]} \sum_{b\in[\mathsf{B}]} X_{j,b,u} > \frac{\mathsf{M}}{\mathsf{N}} \right\}\right] \to 0$ as $\mathsf{B} \to \infty$, and

A2: for some deterministic non-negative vector $(\ell_0, \ldots, \ell_{\mathsf{K}})$ such that $\sum_{i\in[0:\mathsf{K}]} \ell_{\mathsf{K}} = 1$ we have that $\frac{|F_j, \mathcal{W}|}{\mathsf{B}} \to \ell_{|\mathcal{W}|}$ as $\mathsf{B} \to \infty$ in probability, for all $j \in [\mathsf{N}]$ and $\mathcal{W} \subseteq [\mathsf{K}]$ (which implies $\lim_{\mathsf{B}\to\infty} \Pr[|X_t - \mathsf{N}\binom{\mathsf{K}}{t} Y_t| > \epsilon] = 0$ for all $t \in [0 : \mathsf{K}]$ and all $\epsilon > 0$).

Then, clearly, the optimal delivery for this uncoded cache placement gives the load

$$\mathsf{R}(\ell_0, \ldots, \ell_{\mathsf{K}}) := \sum_{t\in[0:\mathsf{K}]} \frac{\binom{\mathsf{K}}{t+1} - \binom{\mathsf{K}-\min(\mathsf{K},\mathsf{N})}{t+1}}{\binom{\mathsf{K}}{t}} \ell_t. \tag{4.32}$$

Therefore finding the optimal uncoded placement for a decentralized cache-aided system is equivalent to find placement probability mass functions in (2.10) such that the conditions in A1 and A2 are satisfied and result in a PMF $(\ell_0, \ldots, \ell_{\mathsf{K}})$ that minimizes the rate in (4.32),

An example of placement probability mass functions in (2.10) such that the conditions in A1 and A2 are satisfied – that is, the one on dMAN is not the only one; possibly there are others. Fix an arbitrary $\epsilon \in (0, 1)$ and let each user cache each bit of each file in an iid Bernoulli(q) fashion with $q := \frac{\mathsf{M}}{\mathsf{N}} - \epsilon$. The probability that the placement violates the cache memory size is

$$\begin{aligned}
\Pr[\mathcal{E}_{\text{caching error}}] &= \Pr\left[\sum_{j=1}^{\mathsf{N}} \sum_{b=1}^{\mathsf{B}} X_{j,b,u} > \mathsf{MB} \text{ for some } u \in [\mathsf{K}]\right] && (4.33a)\\
&= 1 - \left(1 - \Pr\left[\frac{1}{\mathsf{NB}} \sum_{j=1}^{\mathsf{N}} \sum_{b=1}^{\mathsf{B}} X_{j,b,1} > \frac{\mathsf{M}}{\mathsf{N}}\right]\right)^{\mathsf{K}} && (4.33b)\\
&= 1 - \left(1 - e^{-\mathsf{NB}\cdot\left(D\left(\frac{\mathsf{M}}{\mathsf{N}} \| \frac{\mathsf{M}}{\mathsf{N}} - \epsilon\right) + O\left(\frac{1}{\mathsf{NB}}\right)\right)}\right)^{\mathsf{K}} && (4.33c)\\
&\leq 1 - \left(1 - e^{-\mathsf{NB}\cdot\left(\frac{\epsilon^2}{2\frac{\mathsf{M}}{\mathsf{N}}\left(1-\frac{\mathsf{M}}{\mathsf{N}}\right)} + O\left(\frac{1}{\mathsf{NB}}\right)\right)}\right)^{\mathsf{K}} && (4.33d)\\
&= \sum_{k=1}^{\mathsf{K}} \binom{\mathsf{K}}{k} (-1)^{k+1} e^{-k\cdot\mathsf{NB}\cdot\left(\frac{\epsilon^2}{2\frac{\mathsf{M}}{\mathsf{N}}\left(1-\frac{\mathsf{M}}{\mathsf{N}}\right)} + O\left(\frac{1}{\mathsf{NB}}\right)\right)} \to 0 \; as \; \mathsf{B} \to \infty. && (4.33e)
\end{aligned}$$

With this we have

$$\frac{|F_{j,\mathcal{W}}|}{\mathsf{B}} \to (q)^{|\mathcal{W}|}(1-q)^{\mathsf{K}-|\mathcal{W}|}; \tag{4.34}$$

$$X_t \to \binom{\mathsf{K}}{t}(q)^t(1-q)^{\mathsf{K}-t},\ t\in[0:\mathsf{K}]; \tag{4.35}$$

$$\mathsf{N}\binom{\mathsf{K}}{t}Y_t \to \binom{\mathsf{K}}{t}(q)^t(1-q)^{\mathsf{K}-t},\ t\in[0:\mathsf{K}]. \tag{4.36}$$

Therefore, we conclude that the load is

$$\sum_{t\in[0:\mathsf{K}]}\frac{\binom{\mathsf{K}}{t+1}-\binom{\mathsf{K}-\min(\mathsf{K},\mathsf{N})}{t+1}}{\binom{\mathsf{K}}{t}}\ell_t \tag{4.37a}$$

$$=\sum_{t\in[0:\mathsf{K}]}\left(\binom{\mathsf{K}}{t+1}-\binom{\mathsf{K}-\min(\mathsf{K},\mathsf{N})}{t+1}\right)(q)^t(1-q)^{\mathsf{K}-t}|_{q=\frac{\mathsf{M}}{\mathsf{N}}-\epsilon} \tag{4.37b}$$

$$=\frac{1-q}{q}\left[1-(1-q)^{\min(\mathsf{K},\mathsf{N})}\right]|_{q=\frac{\mathsf{M}}{\mathsf{N}}-\epsilon} \tag{4.37c}$$

$$=\frac{1-\frac{\mathsf{M}}{\mathsf{N}}}{\frac{\mathsf{M}}{\mathsf{N}}}\left[1-\left(1-\frac{\mathsf{M}}{\mathsf{N}}\right)^{\min(\mathsf{K},\mathsf{N})}\right]-O(\epsilon). \tag{4.37d}$$

Theorem 6. *In decentralized cache-aided systems under the constraint that each user randomly, uniformly and independently chooses* MB *bits of the* N *files to store, the optimal load is in* (1.6)*, which can be achieved by the novel IC achievable bound in Theorem 3.*

Proof. The delivery phase in the achievable scheme for such systems can be divided into several independent delivery phases of a centralized caching scheme, each one dealing with the sub-files known/cached by the same number of users, as originally proposed in [5] for the dMAN scheme.

As shown in our past work [33], assuming that each bit of each file has the same probability of being cached at each user's cache as in [5], then the load of dMAN in (1.5b) is optimal for $\mathsf{N}\geq\mathsf{K}$. In [19] the optimal load for $\mathsf{N}<\mathsf{K}$ was shown to be achieved essentially following the dMAN original idea, but with the fundamental observation that certain transmitted linear combinations sent by dMAN are redundant when $\mathsf{K}>\mathsf{N}$, leading to the load in (1.6).

For $\mathsf{N}<\mathsf{K}$, by following the same proof idea of [5] and by using as a building block for the delivery phase the novel IC achievable bound in Theorem 3, we can provide an alternative poof for the optimality of (1.6) to that given in [19]. The detailed steps are not reported because they are the same as those in Section 4.2.2 but repeated for every $t\in[0:\mathsf{K}-1]$ (as opposed to for a single value of t). □

Remark 4. *We conclude this section by noting that in [5] the authors claimed that when each user randomly and uniformly stores* MB/N *bits of each file and* $\mathsf{B}\to\infty$*, the length of each sub-file only depends on the number of users who cached it by the Law of Large Numbers. Recall that the placement phase is as follows. Each user* $k\in[\mathsf{K}]$ *randomly and uniformly choose a set of* MB *bits of the* N *files. For each user, there are* $\binom{\mathsf{NB}}{\mathsf{MB}}$ *choices with identical probability. After the placement phase, let* $F_{j,\mathcal{W}}$ *be the set of bits of* F_j *only known by users in* $\mathcal{W}$*, where* $j\in[\mathsf{N}]$ *and* $\mathcal{W}\subseteq[\mathsf{K}]$*; we have* $\sum_{\mathcal{W}\subseteq[\mathsf{K}]}|F_{j,\mathcal{W}}|=\mathsf{B}$ *for each* $j\in[\mathsf{N}]$*, and* $\sum_{j\in[\mathsf{N}]}\sum_{\mathcal{W}\subseteq[\mathsf{K}]:k\in\mathcal{W}}|F_{j,\mathcal{W}}|=\mathsf{MB}$*. In [5] it was claimed that the one can use the Law of Lange Numbers*

to determine the limit $|F_{j,\mathcal{W}}|/\mathsf{B}$ *for large* B*. However, since the number of stored bits in each cache is fixed, the events that a particular bit is stored are neither independent nor uncorrelated. Thus, the Law of Large Numbers can not be used directly. In Appendix A.5 we provide a proof for the claim in [5] without using of the Law of Large Numbers.*

Part 2

Coded Caching in Combination Networks

Chapter 5

System Model and Some Known Results

5.1 System Model of Combination Networks with End-user-caches

Consider the combination network with end-user-caches illustrated in Fig. 1.4. The server has access to N files denoted by $\{F_1, \cdots, F_{\mathsf{N}}\}$, each composed of B bits, and is connected to H relays through H error-free and interference-free links. The relays are connected to $\mathsf{K} = \binom{\mathsf{H}}{\mathsf{r}}$ users through $\mathsf{r}\,\mathsf{K}$ error-free and interference-free links.

We let

$$\mathsf{K}_i := \binom{\mathsf{H}-i}{\mathsf{r}-i},\ i \in [0:\mathsf{r}], \tag{5.1}$$

where $\mathsf{K}_0 = \mathsf{K}$ is the number of users in the system, K_1 is the number of users connected to each relay, and K_i represents the number of users that are simultaneously connected to i relays.

The subset of users connected to relay $h \in [\mathsf{H}]$ is denoted by $\mathcal{U}_h$, and the subset of relays connected to user $k \in [\mathsf{K}]$ by $\mathcal{H}_k$. For a subset of users $\mathcal{J} \subseteq [\mathsf{K}]$, the set of relays connected to all the users in $\mathcal{J}$ is denoted by

$$\mathcal{R}_{\mathcal{J}} := \cap_{k\in\mathcal{J}}\, \mathcal{H}_k. \tag{5.2}$$

For a subset of users $\mathcal{J} \subseteq [\mathsf{K}]$, the set of relays connected to at least one user in $\mathcal{J}$ is denoted by

$$\mathcal{H}_{\mathcal{J}} := \cup_{k\in\mathcal{J}}\, \mathcal{H}_k. \tag{5.3}$$

For a subset of relays $\mathcal{Y} \subseteq [\mathsf{H}]$, the set of users who are simultaneously connected to all the relays in $\mathcal{Y}$ is denoted by

$$\mathcal{U}_{\mathcal{Y}} := \cap_{h\in\mathcal{Y}}\, \mathcal{U}_h. \tag{5.4}$$

Note that $\mathcal{U}_{\{h\}} = \mathcal{U}_h$. For a subset of relays $\mathcal{Y} \subseteq [\mathsf{H}]$, the set of users who are connected to at least two relays in $\mathcal{Y}$ is denoted by

$$\mathcal{G}_{\mathcal{Y}} := \big\{k \in [\mathsf{K}] : |\mathcal{H}_k \cap \mathcal{Y}| \geq 2\big\}. \tag{5.5}$$

For a subset of relays $\mathcal{Y} \subseteq [\mathsf{H}]$, the set of users whose connected relays are all in $\mathcal{Y}$ is denoted by

$$\mathcal{K}_{\mathcal{Y}} := \{k \in [\mathsf{K}] : \mathcal{U}_k \subseteq \mathcal{Y}\}. \tag{5.6}$$

We define the collection of the common users of each two relays as

$$\mathsf{S} := \{\mathcal{U}_{\mathcal{J}} : \mathcal{J} \subseteq [\mathsf{H}], |\mathcal{J}| = 2\}. \tag{5.7}$$

For a given integer t, the t-subsets of users for which there exists at least one relay connected to all the users in this subset is denoted as

$$\mathcal{Z}_t := \big\{\mathcal{W} \subseteq [\mathsf{K}] : |\mathcal{W}| = t,\ \mathcal{R}_{\mathcal{W}} \neq \emptyset\big\}. \tag{5.8}$$

By the inclusion-exclusion principle [68, Theorem 10.1]

$$|\mathcal{Z}_t| = \sum_{n=1}^{\mathsf{r}} \binom{\mathsf{H}}{n} \binom{\mathsf{K}_n}{t} (-1)^{n-1}, \tag{5.9}$$

and moreover, from the definition of K_i in (5.1), we have

$$\frac{t|\mathcal{Z}_t|}{\mathsf{K}_0} = \sum_{n=1}^{\mathsf{r}} \binom{\mathsf{H}}{n} \frac{\mathsf{K}_n}{\mathsf{K}_0} \binom{\mathsf{K}_n - 1}{t-1} (-1)^{n-1} \tag{5.10a}$$

$$= \sum_{n=1}^{\mathsf{r}} \binom{\mathsf{r}}{n} \binom{\mathsf{K}_n - 1}{t-1} (-1)^{n-1}. \tag{5.10b}$$

For the network in Fig. 1.4, we have

$$\mathcal{U}_1 = \{1,2,3\},\ \mathcal{U}_2 = \{1,4,5\},\ \mathcal{U}_3 = \{2,4,6\},\ \mathcal{U}_4 = \{3,5,6\}.$$

and thus, for instance, $\mathcal{H}_1 = \{1,2\}$, $\mathcal{R}_{\{1,2\}} = \{1\}$, $\mathcal{H}_{\{1,2\}} = \{1,2,3\}$, $\mathcal{U}_{\{2,3\}} = \mathcal{U}_2 \cap \mathcal{U}_3 = \{4\}$, $\mathcal{G}_{\{1,2,3\}} = \{1,2,4\}$, $\mathcal{K}_{\{1,2,3\}} = \{1,2,4\}$, $\mathsf{S} = [6]$, and $\mathcal{Z}_1$ contains all the 1-subsets of $[6]$, while $\mathcal{Z}_2$ contains all the 2-subsets of $[6]$ with the exception of $\{1,6\}, \{2,5\}, \{3,4\}$.

In the placement phase, user $k \in [\mathsf{K}]$ stores information about the N files in its cache of size $\mathsf{M}\mathsf{B}$ bits, where $\mathsf{M} \in [0, \mathsf{N}]$. We denote the content in the cache of user $k \in [\mathsf{K}]$ by Z_k and let $\mathbf{Z} := (Z_1, \ldots, Z_{\mathsf{K}})$. Centralized systems allow for coordination among users in the placement phase, while decentralized systems do not. During the delivery phase, user $k \in [\mathsf{K}]$ demands file $d_k \in [\mathsf{N}]$; the demand vector $\mathbf{d} := (d_1, \ldots, d_{\mathsf{K}})$ is revealed to all nodes. Given $(\mathbf{d}, \mathbf{Z})$, the server sends a message X_h of $\mathsf{B}\,\mathsf{R}_h(\mathbf{d}, \mathbf{Z}, \mathsf{M})$ bits to relay $h \in [\mathsf{H}]$. Then, relay $h \in [\mathsf{H}]$ transmits a message $X_{h\to k}$ of $\mathsf{B}\,\mathsf{R}_{h\to k}(\mathbf{d}, \mathbf{Z}, \mathsf{M})$ bits to user $k \in \mathcal{U}_h$. User $k \in [\mathsf{K}]$ must recover its desired file F_{d_k} from Z_k and $(X_{h\to k} : h \in \mathcal{H}_k)$ with high probability for some file size B. The objective is to determine optimal the max-link load in centralized systems defined as

$$\mathsf{R}_{\mathsf{c}}^{\star} := \min_{\mathbf{Z}} \max_{\mathbf{d} \in [\mathsf{N}]^{\mathsf{K}}} \max\{\mathsf{R}^{\mathsf{s}\to\mathsf{r}}, \mathsf{R}^{\mathsf{r}\to\mathsf{u}}\}, \tag{5.11}$$

$$\mathsf{R}^{\mathsf{s}\to\mathsf{r}} = \max_{h \in [\mathsf{H}]} \mathsf{R}_h(\mathbf{d}, \mathbf{Z}, \mathsf{M}), \tag{5.12}$$

$$\mathsf{R}^{\mathsf{r}\to\mathsf{u}} = \max_{h\in[\mathsf{H}],k\in\mathcal{U}_h} \mathsf{R}_{h\to k}(\mathbf{d},\mathbf{Z},\mathsf{M}), \tag{5.13}$$

where $\mathsf{R}^{\mathsf{s}\to\mathsf{r}}$ denotes the max link-load from the server to relays, and $\mathsf{R}^{\mathsf{r}\to\mathsf{u}}$ denotes the max link-load from the relays to users. Since the max-link load of the uncoded routing scheme in [12] is $\mathsf{R}_{\text{routing}} = \mathsf{K}(1-\mathsf{M}/\mathsf{N})/\mathsf{H}$, we define the *coded caching gain* g of a scheme with max-link load R as

$$g := \frac{\mathsf{R}_{\text{routing}}}{\mathsf{R}} = \frac{\mathsf{K}(1-\mathsf{M}/\mathsf{N})/\mathsf{H}}{\mathsf{R}}. \tag{5.14}$$

By the cut-set bound in [12], $g \leq \mathsf{K}_1$. It is obvious that when $\mathsf{M}=0$, and $\mathsf{M}=\mathsf{N}$, the coded caching gain is 1.

5.2 Systems with Uncoded Cache Placement

The max-link load under the constraint of uncoded cache placement is denoted by $\mathsf{R}^{\star}_{\mathsf{c},\mathsf{u}}$. In general, $\mathsf{R}^{\star}_{\mathsf{c},\mathsf{u}} \geq \mathsf{R}^{\star}$.

After the uncoded placement phase is concluded, each file can be effectively divided into non overlapping sub-files depending on which user stores which bit. Let $\mathcal{T}_{\mathbf{Z}} := \{F_{i,\mathcal{W}} : i \in [\mathsf{N}],\ \mathcal{W} \subseteq [\mathsf{K}]\}$, where $F_{i,\mathcal{W}}$ is the set of bits of the file F_i stored solely by the users in $\mathcal{W}$. The set of requested sub-files according to the demand vector $\mathbf{d} \in [\mathsf{N}]^{\mathsf{K}}$ is denoted by

$$\mathcal{T}_{\mathbf{d},\mathbf{Z}} := \{F_{d_k,\mathcal{W}} : k \in [\mathsf{K}],\ \mathcal{W} \subseteq [\mathsf{K}],\ k \notin \mathcal{W}\} \subset \mathcal{T}_{\mathbf{Z}}. \tag{5.15}$$

After the uncoded cache placement and the demand vector are revealed, the delivery phase for the sub-files in $\mathcal{T}_{\mathbf{d},\mathbf{Z}}$ is an index coding problem and can be represented by a directed graph $G_{\mathcal{T}_{\mathbf{d},\mathbf{Z}}}$ (i.e., known as side information graph): each node in the graph represents one sub-file demanded by one user only (if the same file is demanded by multiple users, only one such user is considered at a time) and a directed edge from node i to node i' exists if the sub-file represented by node i is in the cache of the user who requests the sub-file represented by node i'. If $\mathcal{J}$ is a subset of vertices in the graph $G_{\mathcal{T}_{\mathbf{d},\mathbf{Z}}}$, such that the subgraph of $G_{\mathcal{T}_{\mathbf{d},\mathbf{Z}}}$ over $\mathcal{J}$ does not contain a directed cycle, then the "acyclic index coding converse bound" in Theorem 2 for the shared-link broadcast networks can be used to lower bound for the max-link load as a function of the total number of bits of the sub-files in $\mathcal{J}$ (see Proposition 1).

5.3 Caching Scheme in [50, Theorem 1]

We state here the state-of-the-art scheme in [50] for the case of no cache at the relays; the scheme uses MDS-based coded placement so as the delivery from each relay is equivalent to that of a shared-link network serving K_1 virtual users and where the operations of the H virtual shared-link networks are not coordinated. In particular, each file is divided into r non-overlapping and equal-length pieces that are encoded by an (H,r) MDS code. The h-th MDS-coded symbol is denoted by s_i^h and must be delivered by relay $h \in [\mathsf{H}]$ to the users in $\mathcal{U}_h$ following the MAN scheme [4]. This is done as follows.

Placement Fix $g \in [1:\mathsf{K}_1]$. The MDS-coded symbol s_i^h is partitioned into $\binom{\mathsf{K}_1}{g-1}$ non-overlapping and equal-length *subfiles* as $s_i^h = \{s_{i,\mathcal{W}}^h : \mathcal{W} \subseteq \mathcal{U}_h,\ |\mathcal{W}| = g-1\}$ (recall $|\mathcal{U}_h| = \mathsf{K}_1$ for all $h \in [\mathsf{H}]$). There are in total

$$\mathsf{n} = \mathsf{H}\binom{\mathsf{K}_1}{g-1} \text{ [subfiles per file].} \tag{5.16}$$

User $k \in [\mathsf{K}]$ caches $s_{i,\mathcal{W}}^h$ if $k \in \mathcal{W}$ from all $h \in \mathcal{H}_k$ (recall $|\mathcal{H}_k| = \mathsf{r}$ for all users), for a total of

$$\mathsf{k}_1 = \mathsf{r}\binom{\mathsf{K}_1 - 1}{g-2} \text{ [subfiles per file].} \tag{5.17}$$

Delivery The MAN-like multicast coded message

$$w_{\mathcal{J}}^h = \underset{k\in\mathcal{J}}{\oplus} s_{d_k,\mathcal{J}\backslash\{k\}}^h,\ \forall \mathcal{J} \subseteq \mathcal{U}_h : |\mathcal{J}| = g,\ h \in [\mathsf{H}], \tag{5.18}$$

is delivered from the server to relay h, who then forwards it to the users in $\mathcal{J}$. User $k \in [\mathsf{K}]$, thanks to its cache content and the received multicast coded messages from the relays in $\mathcal{H}_k$, recovers

$$\mathsf{k}_2 = \mathsf{r}\binom{\mathsf{K}_1 - 1}{g-1} \text{ [subfiles per file].} \tag{5.19}$$

Performance Each user eventually knows $\mathsf{k}_1 + \mathsf{k}_2 = \mathsf{r}\binom{\mathsf{K}_1}{g-1}$ subfiles of its desired file (either cached or delivered), which suffices to recover all the $\mathsf{n} = \mathsf{H}\binom{\mathsf{K}_1}{g-1}$ subfiles of its desired file because of the (H,r) MDS encoding before placement, where k_1, k_2 and n are defined in (5.17), (5.19) and (5.16), respectively. Since each multicast coded message in (5.18) transmitted from the server is simultaneously useful for g users, the max link-load from the server to relays is $\mathsf{K}(1-\mathsf{M}_{\mathrm{ZY}}/\mathsf{N})/(\mathsf{H}g)$. In addition, since $|s_{d_{k_1},\mathcal{J}\backslash\{k_1\}}^h| = |s_{d_{k_2},\mathcal{J}\backslash\{k_2\}}^h|$ for each relay h, each set $\mathcal{J} \in \mathcal{U}_h$ and any $k_1, k_2 \in \mathcal{J}$, and each user recovers $\mathsf{B}(1-\mathsf{M}_{\mathrm{ZY}}/\mathsf{N})/\mathsf{r}$ bits from r relays, the load on each link from relays to users is $(1-\mathsf{M}_{\mathrm{ZY}}/Nsf)/\mathsf{r}$. With $\mathsf{K}/(\mathsf{H}g) \geq \mathsf{r}$, the max link-load is $\mathsf{K}(1-\mathsf{M}_{\mathrm{ZY}}/\mathsf{N})/(\mathsf{H}g)$. To achieve coded caching gain g, the required memory size per file is

$$\frac{\mathsf{M}_{\mathrm{ZY}}}{\mathsf{N}} = \frac{\mathsf{k}_1}{\mathsf{k}_1+\mathsf{k}_2} = \frac{g-1}{\mathsf{K}_1} \tag{5.20}$$

where in (5.20) the factor $\frac{\mathsf{H}}{\mathsf{r}}$ is the inverse of the rate of the MDS code used before placement.

Limitation In [50], the operations at the H relays are uncoordinated. Indeed, consider the network in Fig. 1.4 for $g = 2$. The scheme in [50] uses an $(\mathsf{H},\mathsf{r}) = (4,2)$ MDS code, and the MDS-coded symbols $s_{i,\mathcal{W}}^{h_1}$ and $s_{i,\mathcal{W}}^{h_2}$ are treated as two "independent" subfiles if $h_1 \neq h_2$. For example, among the MDS subfiles $s_{i,\{1\}}^1$, $s_{i,\{2\}}^1$, $s_{i,\{3\}}^1$, $s_{i,\{1\}}^2$, $s_{i,\{4\}}^2$ and $s_{i,\{5\}}^2$, each of length is $\mathsf{B}/6$, user 1 caches $s_{i,\{1\}}^1$ and $s_{i,\{1\}}^2$, which requires $\mathsf{M}/\mathsf{N} = 2/6$. However, $s_{i,\{1\}}^1$ and $s_{i,\{1\}}^2$ can be treated as a single subfile known / cached by user 1. This observation is key for the design of the novel proposed schemes.

5.4 Bit-Borrowing

To conclude this section, we introduce the *bit-borrowing idea* proposed in [63], [69], which we will also use in our novel delivery scheme. For decentralized shared-link caching problems with non-uniform demands or finite file size B, the subfiles in (2.14) may have different lengths. If this is the case, instead of zero-padding the sub-files to meet the length of the longest one, which leads to inefficient transmissions, we can borrow bits from some subfiles to 'lengthen' short sub-files in such a way that the borrowed bits need not to be transmitted at a later stage. More precisely, if $|F_{d_\ell,\mathcal{J}\backslash\{\ell\}}| < \max_{k\in\mathcal{J}} |F_{d_k,\mathcal{J}\backslash\{k\}}|$, we take bits from some $F_{d_\ell,\mathcal{W}}$, where $\ell \notin \mathcal{W}$ and $\mathcal{J}\backslash\{\ell\} \subset \mathcal{W}$ (because $F_{d_\ell,\mathcal{W}}$ is also demanded by user ℓ and known by the users in $\mathcal{J}\backslash\{\ell\}$) and add those bits to $F_{d_\ell,\mathcal{J}\backslash\{\ell\}}$.

Chapter 6

Novel Converse Bounds for Combination Networks with End-user-caches

6.1 Chapter Overview and Related Publications

This chapter provides novel converse for combination networks with end-user-caches. The main idea is to adapt the converse bounds in shared-link models and the acyclic index coding converse bound with the network topology. We first use a cut-set strategy to directly extend the converse bounds for shared-link networks to combination networks. We then propose two ways to tighten the acyclic index coding converse bound in combination networks.

Related Publications

1. Kai Wan, Mingyue Ji, Pablo Piantanida and Daniela Tuninetti, "Novel Outer Bounds and Inner Bounds with Uncoded Cache Placement for Combination Networks with End-User- Caches", *in preparation*, to IEEE Trans. on Information Theory. [55]

2. Kai Wan, Mingyue Ji, Pablo Piantanida and Daniela Tuninetti, "Novel Outer Bounds for Combination Networks with End-User-Caches", in Proceedings of the IEEE Information Theory Workshop (ITW), Nov. 2017. [60]

6.2 Main Results on Novel Converse Bounds for Combination Networks with End-user-caches

To derive bounds on the max link-load for combination networks with end-user-caches, we can use the cut-set strategy proposed in [12] to extend converse bounds for cache-aided shared-link models. More precisely, each time we consider the cut of $x \in [\mathsf{r}:\mathsf{H}]$ relays. The total link load transmitted from the server should be outer bounded by the load in shared-link model including N files, $\binom{x}{\mathsf{r}}$ users with memory of MB bits. So we can use this strategy to extend any converse bounds for shared-link models to combination networks. Based on this strategy, in the following we extend the enhance cut-set converse bound in (1.4) and the converse bound under the constraint of uncoded cache placement in Theorem 4.

Theorem 7. *For an* $(\mathsf{H},\mathsf{r},\mathsf{M},\mathsf{N})$ *combination network, the max link-load should satisfy*

$$\mathsf{R}_{\mathsf{c}}^{\star} \geq \mathsf{R}_{\mathsf{e}}^{\star} := \frac{s-1+\alpha-\frac{s(s-1)-l(l-1)+2\alpha s}{2(\mathsf{N}-l+1)}\mathsf{M}}{x},\ x\in[\mathsf{r}:\mathsf{H}]. \tag{6.1}$$

for any $\alpha \in \{1,\ldots,\min\{\mathsf{N},\binom{x}{\mathsf{r}}\}\}$, $\alpha\in[0,1]$, *where* $l\in[1:s]$ *is the minimum value such that* $\frac{s(s-1)-l(l-1)}{2}+\alpha s \leq (\mathsf{N}-l+1)l^2$.

Theorem 8. *Consider a combination network with end-user-caches, under the constraint of uncoded cache placement, the max link-load should satisfy*

$$\mathsf{R}_{\mathsf{c,u}}^{\star}\left(\mathsf{M}=t\frac{\mathsf{N}}{\binom{x}{\mathsf{r}}}\right) \geq \frac{1}{x}\,\frac{\binom{x}{\mathsf{r}}_{t+1}-\binom{\max\left\{\binom{x}{\mathsf{r}}-\mathsf{N},0\right\}}{t+1}}{\binom{\binom{x}{\mathsf{r}}}{t}},\ x\in[\mathsf{r}:\mathsf{H}]. \tag{6.2}$$

In the rest of the thesis, for a set of sub-files $\mathcal{S}\subseteq\mathcal{T}_{\mathbf{d},\mathbf{Z}}$ where $\mathcal{T}_{\mathbf{d},\mathbf{Z}}$ is given in (5.15), we denote by $H(\mathcal{S})$ the joint entropy of the sub-files in $\mathcal{S}$, and by $H(Y|\mathcal{S}^c)$ the entropy of a random variable Y conditioned on the sub-files in $\mathcal{S}^c := \mathcal{T}_{\mathbf{d},\mathbf{Z}}\setminus\mathcal{S}$. Other preliminary results for the converse bounds can be found in Section 6.3.

Recall that $\mathbf{p}(\mathcal{J}) := \big(p_1(\mathcal{J}),\ldots,p_{|\mathbf{p}(\mathcal{J})|}(\mathcal{J})\big)$ representing a permutation of elements of the set $\mathcal{J}$. We then propose the following lower bound which is tighter than the cut-set converse bound in (6.2). Since it follows quite straightforwardly from the work we did for shared-link broadcast networks [20], in this thesis we consider it as the 'baseline' bound. The proof is in Section 6.4.

Theorem 9. *Consider a combination network with uncoded cache placement and* $(\mathbf{d},\mathbf{Z})$ *such that the demands in* $\mathbf{d}$ *are distinct. For each subset* $\mathcal{Q}\subseteq[\mathsf{H}]$ *such that* $|\mathcal{Q}|\in[\mathsf{r}:\mathsf{H}]$, *and each permutation* $\mathbf{p}(\mathcal{K}_{\mathcal{Q}})$, *with* $B\gg 1$, *we have*

$$|\mathcal{Q}|\mathsf{R}_{\mathsf{c,u}}^{\star} \geq \sum_{i\in[|\mathcal{K}_{\mathcal{Q}}|]}\ \sum_{\mathcal{W}\subseteq[\mathsf{K}]\setminus\{p_1(\mathcal{K}_{\mathcal{Q}}),\ldots,p_i(\mathcal{K}_{\mathcal{Q}})\}} x_{\mathcal{W}}, \tag{6.3a}$$

$$x_{\mathcal{W}} := \frac{1}{\mathsf{NB}}\sum_{i\in[\mathsf{N}]}|F_{i,\mathcal{W}}|,\ \forall\mathcal{W}\subseteq[\mathsf{K}]:\ \sum_{\mathcal{W}\subseteq[\mathsf{K}]}x_{\mathcal{W}}=1, \tag{6.3b}$$

$$\sum_{\mathcal{W}\subseteq[\mathsf{K}]:i\in\mathcal{W}} x_{\mathcal{W}} \leq \frac{\mathsf{M}}{\mathsf{N}},\ \forall i\in[\mathsf{K}]. \tag{6.3c}$$

Remark 5. *The lower bound in Theorem 9 can be numerically computed by means of a linear program with variables* $(\mathsf{R}_{\mathsf{c,u}}^{\star}, x_{\mathcal{W}}:\mathcal{W}\subseteq[\mathsf{K}])$ *and constraints in* (6.3a)-(6.3c).

As we shall see, this 'baseline' bound can be improved by means of Propositions 2 and 3. We use the idea highlighted in Example 3 to get the following lower bound, whose proof can be found in Theorem 6.5.

Theorem 10. *Consider a combination network with uncoded cache placement and* $(\mathbf{d},\mathbf{Z})$ *such that the demands in* $\mathbf{d}$ *are distinct. For each set of relays* $\mathcal{Q}\subseteq[\mathsf{H}]$, *each integer* $a\in[\lfloor|\mathcal{Q}|/\mathsf{r}\rfloor]$, *each disjoint partition* $\mathcal{Q}=\mathcal{Q}_1\cup\ldots\cup\mathcal{Q}_a$ *where* $|\mathcal{Q}_i|\geq\mathsf{r}$ *and* $i\in[a]$, *and each combination of permutations*

$\mathbf{p}(\mathcal{K}_{\mathcal{Q}_1}),\ldots,\mathbf{p}(\mathcal{K}_{\mathcal{Q}_a}),\mathbf{p}\big(\mathcal{K}_{\mathcal{Q}}\setminus(\mathcal{K}_{\mathcal{Q}_1}\cup\cdots\cup\mathcal{K}_{\mathcal{Q}_a})\big)$*, the following must hold for* $B\gg 1$

$$|\mathcal{Q}|\mathsf{R}^{\star}_{\mathsf{c,u}}\geq\sum_{i\in[a]}\sum_{j\in[|\mathcal{K}_{\mathcal{Q}_i}|]}\sum_{\mathcal{W}\subseteq[\mathsf{K}]\setminus\cup_{k\in[j]}\{p_k(\mathcal{K}_{\mathcal{Q}_i})\}}x_{\mathcal{W}}+\sum_{j\in\left[\left|\mathcal{K}_{\mathcal{Q}}\setminus\mathcal{V}\right|\right]}\sum_{\mathcal{W}\subseteq([\mathsf{K}]\setminus\mathcal{V})\setminus\cup_{k\in[j]}\{p_k(\mathcal{K}_{\mathcal{Q}}\setminus\mathcal{V})\}}x_{\mathcal{W}},\tag{6.4}$$

where $\mathcal{V}:=\cup_{i\in[a]}\mathcal{K}_{\mathcal{Q}_i}$.

Remark 6. *The lower bound in Theorem 10 can be computed by means of a linear program with variables* $(\mathsf{R}^{\star}_{\mathsf{c,u}},x_{\mathcal{W}}:\mathcal{W}\subseteq[\mathsf{K}])$ *and constraints in* (6.4)*,* (6.3b) *and* (6.3c)*.*

We use the idea of Example 4 to get the following converse bound and the detailed proof is in Theorem 6.6.

Theorem 11. *Consider a combination network with uncoded cache placement and* $(\mathbf{d},\mathbf{Z})$ *such that the demands in* $\mathbf{d}$ *are distinct. For each integer* $b\in[\mathsf{r}:\mathsf{H}]$*, each set of relays* $\mathcal{Q}\subseteq[\mathsf{H}]$ *with* $|\mathcal{Q}|=b$*, each permutation* $\mathbf{p}(\mathcal{K}_{\mathcal{Q}})$*, with* $B\gg 1$*, the bound in* (6.39) *holds, satisfying* (6.3b)-(6.3c) *and for each permutation* $\mathbf{p}([\mathsf{K}])$*, the following must hold*

$$\sum_{\mathcal{Q}:|\mathcal{Q}|=b}y_{\mathcal{Q}}\geq\sum_{i\in[\mathsf{K}]}\sum_{\mathcal{W}\subseteq[\mathsf{K}]\setminus\cup_{j\in[i]}\{p_j([\mathsf{K}])\}}c\big(\{p_i([\mathsf{K}])\}\cup\mathcal{W},b\big)x_{\mathcal{W}},\tag{6.5a}$$

$$c(\mathcal{W}_1,l):=\max\left\{\binom{\mathsf{H}-1}{l-1}-|\{\mathcal{Q}\subseteq[\mathsf{H}]:|\mathcal{Q}|=l,\mathcal{K}_{\mathcal{Q}}\nsubseteq[\mathsf{K}]\setminus\mathcal{W}_1\}|,0\right\}.\tag{6.5b}$$

Remark 7. *The lower bound in Theorem 11 can be computed by means of a linear program with variables* $(\mathsf{R}^{\star}_{\mathsf{c,u}},x_{\mathcal{W}}:\mathcal{W}\subseteq[\mathsf{K}],y_{\mathcal{Q}}:\mathcal{Q}\subseteq[\mathsf{H}])$ *and constraints in* (6.39)*,* (6.5a)*,* (6.3b) *and* (6.3c)*.*

By combining the converse bound ideas in Theorem 10 and 11, our final converse bound is as follows (the proof can be found in Section 6.7).

Theorem 12. *Consider a combination network with uncoded cache placement and* $(\mathbf{d},\mathbf{Z})$ *such that the demands are distinct. For each integer* $b\in[\mathsf{r}:\mathsf{H}]$*, each set of relays* $\mathcal{Q}\subseteq[\mathsf{H}]$ *with* $|\mathcal{Q}|=b$*, each integer* $a\in[\lfloor b/\mathsf{r}\rfloor]$*, each disjoint partition* $\mathcal{Q}=\mathcal{Q}_1\cup\ldots\cup\mathcal{Q}_a$ *where* $|\mathcal{Q}_i|\geq\mathsf{r}$ *and* $i\in[a]$*, each combination of permutations* $\mathbf{p}(\mathcal{K}_{\mathcal{Q}_1}),\ldots,\mathbf{p}(\mathcal{K}_{\mathcal{Q}_a}),\mathbf{p}\big(\mathcal{K}_{\mathcal{Q}}\setminus(\mathcal{K}_{\mathcal{Q}_1}\cup\cdots\cup\mathcal{K}_{\mathcal{Q}_a})\big)$*, with* $B\gg 1$*, the following must hold*

$$|\mathcal{Q}|\mathsf{R}^{\star}_{\mathsf{c,u}}\geq\sum_{i\in[a]}\sum_{j\in[|\mathcal{K}_{\mathcal{Q}_i}|]}\sum_{\mathcal{W}\subseteq[\mathsf{K}]\setminus\cup_{k\in[j]}\{p_k(\mathcal{K}_{\mathcal{Q}_i})\}}x_{\mathcal{W}}+\sum_{j\in\left[\left|\mathcal{K}_{\mathcal{Q}}\setminus\mathcal{V}\right|\right]}\sum_{\mathcal{W}\subseteq([\mathsf{K}]\setminus\mathcal{V})\setminus\cup_{k\in[j]}\{p_k(\mathcal{K}_{\mathcal{Q}}\setminus\mathcal{V})\}}x_{\mathcal{W}}+y_{\mathcal{Q}},\tag{6.6}$$

satisfying (6.3b)*,* (6.3c) *and* (6.5a)*, where* $\mathcal{V}:=\cup_{i\in[a]}\mathcal{K}_{\mathcal{Q}_i}$.

Remark 8. *The lower bound in Theorem 12 can be numerically computed by means of a linear program with variables* $(\mathsf{R}^{\star}_{\mathsf{c,u}},x_{\mathcal{W}}:\mathcal{W}\subseteq[\mathsf{K}],y_{\mathcal{Q}}:\mathcal{Q}\subseteq[\mathsf{H}])$ *and constraints in* (6.6)*,* (6.5a)*,* (6.3b) *and* (6.3c)*. Notice that the computation complexity orders of all theorems reported in this thesis are the same with* $\mathcal{O}(2^{\mathsf{K}})$ *variables and* $\mathcal{O}(\mathsf{HK}!)$ *constraints.*

6.3 Preliminaries

We first extend the shared-link broadcast networks "acyclic index coding converse bound" from [20] to combinations networks.

Proposition 1. *Consider a combination network with uncoded cache placement and* $(\mathbf{d}, \mathbf{Z})$ *such that the demands in* $\mathbf{d}$ *are distinct. For a set of relays* $\mathcal{J} \subseteq [\mathsf{H}]$*, and for an acyclic set of sub-files* $\mathcal{S} \subseteq \mathcal{T}_{\mathbf{d},\mathbf{Z}}$ *in the directed graph* $G_{\mathcal{T}_{\mathbf{d},\mathbf{Z}}}$ *that are demanded by the users in* $\mathcal{K}_{\mathcal{J}}$*, the following must hold*

$$H(\mathcal{S}) \leq H(X_{\mathcal{J}}|\mathcal{S}^c) + \mathsf{B}\varepsilon_{\mathsf{B}} \tag{6.7a}$$
$$\leq |\mathcal{J}|\mathsf{B}\,\mathsf{R}^{\star}_{\mathsf{c},\mathrm{u}} + \mathsf{B}\varepsilon_{\mathsf{B}}. \tag{6.7b}$$

Proof. The entropy of the sub-files in $\mathcal{S}$ is bounded as

$$H(\mathcal{S}) = H(\mathcal{S}|\mathcal{S}^c) = H(X_{\mathcal{J}}, \mathcal{S}|\mathcal{S}^c) \tag{6.8a}$$
$$= H(X_{\mathcal{J}}|\mathcal{S}^c) + H(\mathcal{S}|X_{\mathcal{J}}, \mathcal{S}^c) \tag{6.8b}$$
$$\leq H(X_{\mathcal{J}}|\mathcal{S}^c) + \mathsf{B}\varepsilon_{\mathsf{B}} \tag{6.8c}$$
$$\leq H(X_{\mathcal{J}}) + \mathsf{B}\varepsilon_{\mathsf{B}} \tag{6.8d}$$
$$\leq |\mathcal{J}|\mathsf{R}^{\star}_{\mathsf{c},\mathrm{u}}\mathsf{B} + \mathsf{B}\varepsilon_{\mathsf{B}}, \tag{6.8e}$$

where in (6.8a) we used the independence of the sub-files and the fact that $X_{\mathcal{J}}$ is function of $\mathcal{T}_{\mathbf{d},\mathbf{Z}}$, in (6.8c) we use the fact that $\mathcal{S}$ is acyclic and Fano's inequality (where $\lim_{\mathsf{B}\to\infty} \varepsilon_{\mathsf{B}} = 0$), and in (6.8e) we used the definition of $\mathsf{R}^{\star}_{\mathsf{c},\mathrm{u}}$. □

Proposition 1 may not be tight when $|\mathcal{J}|\mathsf{R}^{\star}_{\mathsf{c},\mathrm{u}}$ in (6.8e) is strictly larger than $H(X_{\mathcal{J}}|\mathcal{S}^c)$ in (6.8c). In the following, we tighten the bound in Proposition 1.

Proposition 2. *Consider a combination network with uncoded cache placement and* $(\mathbf{d}, \mathbf{Z})$ *where the demands in* $\mathbf{d}$ *are distinct. For a set of relays* $\mathcal{J} \subseteq [\mathsf{H}]$*, and for two sets of sub-files* $\mathcal{S}_1, \mathcal{S}_2 \subseteq \mathcal{T}_{\mathbf{d},\mathbf{Z}}$ *that are acyclic in the graph* $G_{\mathcal{T}_{\mathbf{d},\mathbf{Z}}}$*, where* $\mathcal{S}_1$ *includes some sub-files demanded by the users in* $\mathcal{K}_{\mathcal{J}}$ *and* $\mathcal{S}_2$ *includes some sub-files demanded by the users in* $[\mathsf{K}] \setminus \mathcal{K}_{\mathcal{J}}$ *but not cached by the users in* $\mathcal{K}_{\mathcal{J}}$*, we have*

$$H(\mathcal{S}_1) + H(X_{\mathcal{J}}|\mathcal{S}_2^c) \leq |\mathcal{J}|\mathsf{B}\,\mathsf{R}^{\star}_{\mathsf{c},\mathrm{u}} + 2\mathsf{B}\varepsilon_{\mathsf{B}}. \tag{6.9}$$

Proof. The entropy of the sub-files can be bounded as

$$H(\mathcal{S}_1, \mathcal{S}_2) \leq H\big(X_{[\mathsf{H}]}|(\mathcal{S}_1 \cup \mathcal{S}_2)^c\big) + \mathsf{B}\varepsilon_{\mathsf{B}} \tag{6.10a}$$
$$= H\big(X_{\mathcal{J}}|(\mathcal{S}_1 \cup \mathcal{S}_2)^c\big) + H\big(X_{[\mathsf{H}]\setminus\mathcal{J}}|X_{\mathcal{J}}, \mathcal{S}_2^c\big) + I\big(\mathcal{S}_1; X_{[\mathsf{H}]\setminus\mathcal{J}}|X_{\mathcal{J}}, (\mathcal{S}_1 \cup \mathcal{S}_2)^c\big) + \mathsf{B}\varepsilon_{\mathsf{B}} \tag{6.10b}$$
$$\leq |\mathcal{J}|\mathsf{B}\,\mathsf{R}^{\star}_{\mathsf{c},\mathrm{u}} + H\big(X_{[\mathsf{H}]\setminus\mathcal{J}}|X_{\mathcal{J}}, \mathcal{S}_2^c\big) + H\big(\mathcal{S}_1|X_{\mathcal{J}}, (\mathcal{S}_1 \cup \mathcal{S}_2)^c\big) + \mathsf{B}\varepsilon_{\mathsf{B}} \tag{6.10c}$$
$$\leq |\mathcal{J}|\mathsf{B}\,\mathsf{R}^{\star}_{\mathsf{c},\mathrm{u}} + H\big(X_{[\mathsf{H}]\setminus\mathcal{J}}|X_{\mathcal{J}}, \mathcal{S}_2^c\big) + 2\mathsf{B}\varepsilon_{\mathsf{B}} \tag{6.10d}$$
$$\leq |\mathcal{J}|\mathsf{B}\,\mathsf{R}^{\star}_{\mathsf{c},\mathrm{u}} - H\big(X_{\mathcal{J}}|\mathcal{S}_2^c\big) + H(\mathcal{S}_2) + 2\mathsf{B}\varepsilon_{\mathsf{B}}, \tag{6.10e}$$

where (6.10a) is from (6.7a), where (6.10d) is from Fano's inequality (where $\lim_{\mathsf{B}\to\infty} \varepsilon_{\mathsf{B}} = 0$) and the fact $\mathcal{S}_1$ is acyclic and $\mathcal{S}_2$ does not include the side information of the user requesting $\mathcal{S}_1$, and where (6.10e) is because

$$H(\mathcal{S}_2) = H(\mathcal{S}_2|\mathcal{S}_2^c) \tag{6.11a}$$

$$= H\big(\mathcal{S}_2, X_{[\mathsf{H}]}|\mathcal{S}_2^c\big) \tag{6.11b}$$
$$\geq H\big(X_{[\mathsf{H}]}|\mathcal{S}_2^c\big) \tag{6.11c}$$
$$= H\big(X_{\mathcal{J}}|\mathcal{S}_2^c\big) + H\big(X_{[\mathsf{H}]\setminus\mathcal{J}}|X_{\mathcal{J}}, \mathcal{S}_2^c\big). \tag{6.11d}$$

This concludes the proof. □

Finally, we generalize the well-known sub-modularity of entropy.

Proposition 3. *Let $\mathcal{Y}$ be a set of random variables, and $\mathcal{M}$ be a set of mutually independent random variables (but not necessary independent of $\mathcal{Y}$). If $\mathcal{Y}_1, \mathcal{Y}_2 \subseteq \mathcal{Y}$ and $\mathcal{M}_1, \mathcal{M}_2 \subseteq \mathcal{M}$, then the following must hold*

$$H(\mathcal{Y}_1|\mathcal{M}_1) + H(\mathcal{Y}_2|\mathcal{M}_2) \geq H(\mathcal{Y}_1 \cup \mathcal{Y}_2|\mathcal{M}_1 \cup \mathcal{M}_2) + H(\mathcal{Y}_1 \cap \mathcal{Y}_2|\mathcal{M}_1 \cap \mathcal{M}_2). \tag{6.12}$$

Remark 9. *If either $\mathcal{Y}_1 = \mathcal{Y}_2$ or $\mathcal{M}_1 = \mathcal{M}_2$, Proposition 3 reduces to the well-known submodularity of entropy.*

Proof. Without loss of generality, assume $\mathcal{M}_1 = \{M_0, M_1\}$ and $\mathcal{M}_2 = \{M_0, M_2\}$, where M_0, M_1, M_2 are independent random variables. We have

$$H(\mathcal{Y}_1|M_0, M_1) + H(\mathcal{Y}_2|M_0, M_2) \tag{6.13a}$$
$$= H(\mathcal{Y}_1|M_0, M_1, M_2) + I(\mathcal{Y}_1; M_2|M_0, M_1) + H(\mathcal{Y}_2|M_0, M_1, M_2) + I(\mathcal{Y}_2; M_1|M_0, M_2) \tag{6.13b}$$
$$= H(\mathcal{Y}_1 \cup \mathcal{Y}_2|M_0, M_1, M_2) + I(\mathcal{Y}_1; \mathcal{Y}_2|M_0, M_1, M_2) + I(\mathcal{Y}_1; M_2|M_0, M_1) + I(\mathcal{Y}_2; M_1|M_0, M_2) \tag{6.13c}$$
$$\geq H(\mathcal{Y}_1 \cup \mathcal{Y}_2|M_0, M_1, M_2) + H(\mathcal{Y}_1 \cap \mathcal{Y}_2|M_0), \tag{6.13d}$$

where the last inequality follows from

$$I(\mathcal{Y}_1; \mathcal{Y}_2|M_0, M_1, M_2) + I(\mathcal{Y}_1; M_2|M_0, M_1) + I(\mathcal{Y}_2; M_1|M_0, M_2) \tag{6.14a}$$
$$= I(\mathcal{Y}_1; \mathcal{Y}_2|M_0, M_1, M_2) + I(\mathcal{Y}_1, M_1; M_2|M_0) + I(\mathcal{Y}_2; M_1|M_0, M_2) \tag{6.14b}$$
$$= I(\mathcal{Y}_1; \mathcal{Y}_2|M_0, M_1, M_2) + I(\mathcal{Y}_1, M_1; \mathcal{Y}_2, M_2|M_0) - I(\mathcal{Y}_1, M_1; \mathcal{Y}_2|M_0, M_2) + I(\mathcal{Y}_2; M_1|M_0, M_2) \tag{6.14c}$$
$$= I(\mathcal{Y}_1; \mathcal{Y}_2|M_0, M_1, M_2) + I(\mathcal{Y}_1, M_1; \mathcal{Y}_2, M_2|M_0) - I(\mathcal{Y}_1; \mathcal{Y}_2|M_0, M_1, M_2) \tag{6.14d}$$
$$= I(\mathcal{Y}_1, M_1; \mathcal{Y}_2, M_2|M_0) \geq I(\mathcal{Y}_1; \mathcal{Y}_2|M_0) \tag{6.14e}$$
$$\geq H(\mathcal{Y}_1 \cap \mathcal{Y}_2|M_0). \tag{6.14f}$$

□

6.4 Proof of Theorem 9

In (6.3b), $\mathsf{NB}x_{\mathcal{W}}$ represents the number of bits only cached by the users in $\mathcal{W} \subseteq [\mathsf{K}]$. For a demand vector $\mathbf{d} \in [\mathsf{N}]$ whose elements are distinct, a set $\mathcal{S}' \subseteq [\mathsf{K}]$ and a vector $\mathbf{v} = (v_1, \ldots, v_{|\mathbf{v}|})$ where $v_i \in \mathcal{S}'$, $\forall i \in$

$[|\mathbf{v}|]$, we define

$$f(\mathbf{d},\mathcal{S}',\mathbf{v}):=\bigcup_{i\in[|\mathbf{v}|]}\big\{F_{d_{v_i},\mathcal{W}}:\mathcal{W}\subseteq\mathcal{S}'\setminus\{v_1,\ldots,v_i\}\big\};\tag{6.15}$$

by [20, Lemma 1] the set $f(\mathbf{d},\mathcal{S}',\mathbf{v})$ forms an acyclic set in the directed graph $G_{\mathcal{T}_{\mathbf{d},\mathbf{z}}}$. For each $\mathcal{Q}\subseteq[\mathsf{H}]$ with $|\mathcal{Q}|\in[\mathsf{r}:\mathsf{H}]$, each permutation $\mathbf{p}(\mathcal{K}_\mathcal{Q})$, and each demand vector $\mathbf{d}$ with distinct demands, Proposition 1 with $\mathcal{S}=f\big(\mathbf{d},[\mathsf{K}],\mathbf{p}(\mathcal{K}_\mathcal{Q})\big)$ provides a lower bound on $\mathsf{R}^\star_{\mathsf{c,u}}$. In the limit for $B\gg 1$, by summing all the so obtained bounds for a fixed $\mathcal{Q}\subseteq[\mathsf{H}]$ we arrive at (6.3a).

6.5 Proof of Theorem 10

Our first improvement to Theorem 9 is explained by way of an example.

Example 3. Consider the combination network in Fig. 1.4 with $\mathsf{N}=6$ and $\mathsf{M}=2$. Consider the demand vector $\mathbf{d}=(1,\ldots,6)$. Choose a set of relays $\mathcal{Q}$ and divide $\mathcal{Q}$ into several disjoint subsets, each of which has a length not less than $\mathsf{r}=2$. In this example, we let $\mathcal{Q}=[\mathsf{H}]=[4]$ and divide $\mathcal{Q}$ into $\mathcal{Q}_1=\{1,2\}$ and $\mathcal{Q}_2=\{3,4\}$; so $\mathcal{K}_{\mathcal{Q}_1}=\{1\}$ and $\mathcal{K}_{\mathcal{Q}_2}=\{6\}$. We then consider the three permutations $\mathbf{p}(\mathcal{K}_{\mathcal{Q}_1})=(1)$, $\mathbf{p}(\mathcal{K}_{\mathcal{Q}_2})=(6)$ and $\mathbf{p}\big(\mathcal{K}_\mathcal{Q}\setminus(\mathcal{K}_{\mathcal{Q}_1}\cup\mathcal{K}_{\mathcal{Q}_2})\big)=(2,3,4,5)$. Recall the definition of f given in (6.15) and let

$$\mathcal{B}_1=f\big(\mathbf{d},[\mathsf{K}],\mathbf{p}(\mathcal{K}_{\mathcal{Q}_1})\big)=\{F_{1,\mathcal{W}}:\mathcal{W}\subseteq[2:6]\},\tag{6.16}$$
$$\mathcal{B}_2=f\big(\mathbf{d},[\mathsf{K}],\mathbf{p}(\mathcal{K}_{\mathcal{Q}_2})\big)=\{F_{6,\mathcal{W}}:\mathcal{W}\subseteq[1:5]\},\tag{6.17}$$
$$\mathcal{B}_3=f\Big(\mathbf{d},[\mathsf{K}]\setminus(\mathcal{K}_{\mathcal{Q}_1}\cup\mathcal{K}_{\mathcal{Q}_2}),\mathbf{p}\big(\mathcal{K}_\mathcal{Q}\setminus(\mathcal{K}_{\mathcal{Q}_1}\cup\mathcal{K}_{\mathcal{Q}_2})\big)\Big)=\{F_{i,\mathcal{W}}:i\in[2:5],\mathcal{W}\subseteq[i+1:5]\}.\tag{6.18}$$

By using Proposition 2 with $(\mathcal{J},\mathcal{S}_1,\mathcal{S}_2)=(\mathcal{Q}_1,\mathcal{B}_1,\mathcal{B}_3)$ we get

$$H(\mathcal{B}_1)\le|\mathcal{Q}_1|\mathsf{R}^\star_{\mathsf{c,u}}\mathsf{B}-H\big(X_{\mathcal{Q}_1}|\mathcal{B}_3^c\big)+2\mathsf{B}\varepsilon_\mathsf{B}.\tag{6.19}$$

and with $(\mathcal{J},\mathcal{S}_1,\mathcal{S}_2)=(\mathcal{Q}_2,\mathcal{B}_2,\mathcal{B}_3)$ we get

$$H(\mathcal{B}_2)\le|\mathcal{Q}_2|\mathsf{R}^\star_{\mathsf{c,u}}\mathsf{B}-H\big(X_{\mathcal{Q}_2}|\mathcal{B}_3^c\big)+2\mathsf{B}\varepsilon_\mathsf{B}.\tag{6.20}$$

We sum (6.19) and (6.20) to obtain

$$H(\mathcal{B}_1,\mathcal{B}_2)\le|\mathcal{Q}|\mathsf{R}^\star_{\mathsf{c,u}}\mathsf{B}-\Big[H\big(X_{\mathcal{Q}_1}|\mathcal{B}_3^c\big)+H\big(X_{\mathcal{Q}_2}|\mathcal{B}_3^c\big)\Big]+4\mathsf{B}\varepsilon_\mathsf{B}\tag{6.21a}$$
$$\le|\mathcal{Q}|\mathsf{R}^\star_{\mathsf{c,u}}\mathsf{B}-H(X_\mathcal{Q}|\mathcal{B}_3^c)+4\mathsf{B}\varepsilon_\mathsf{B}\tag{6.21b}$$
$$\le|\mathcal{Q}|\mathsf{R}^\star_{\mathsf{c,u}}\mathsf{B}-H(\mathcal{B}_3)+4\mathsf{B}\varepsilon_\mathsf{B},\tag{6.21c}$$

where (6.21c) follows from (6.7a). With the above mentioned choice of permutations and $B\gg 1$, the bound in (6.21c) becomes

$$4\mathsf{B}\mathsf{R}^\star_{\mathsf{c,u}}\ge\sum_{\mathcal{W}\subseteq[6]\setminus\{1\}}|F_{1,\mathcal{W}}|+\sum_{\mathcal{W}\subseteq[6]\setminus\{6\}}|F_{6,\mathcal{W}}|+\sum_{i\in[2:5]}\sum_{\mathcal{W}\subseteq[i+1:5]}|F_{i,\mathcal{W}}|.\tag{6.22}$$

If we list all the inequalities in the form of (6.22) for all the possible demands where users demand distinct files, and we sum them all together, we obtain(the definition of $x_{\mathcal{W}}$ is in (6.3b))

$$4\mathsf{R}^{\star}_{\mathsf{c,u}} \geq \sum_{\mathcal{W}\subseteq[2:6]} x_{\mathcal{W}} + \sum_{\mathcal{W}\subseteq[1:5]} x_{\mathcal{W}} + \sum_{i\in[2:5]} \sum_{\mathcal{W}\subseteq[i+1:5]} x_{\mathcal{W}}. \tag{6.23}$$

We then consider all the possible disjoint partitions of $\mathcal{Q}$, and for each partition we consider all the possible combinations of permutations to write bounds as in the form of (6.23). For $\mathcal{Q}$ with $|\mathcal{Q}| \leq 3$, since $\mathcal{Q}$ can not be divided into two sets each of which has length not less than $\mathsf{r} = 2$, we directly use the bound in (6.3a). With the file length and memory size constrains in (6.3b)-(6.3c), we can compute the converse bound by a linear program with the above mentioned constraints and with variables $(\mathsf{R}^{\star}_{\mathsf{c,u}}, x_{\mathcal{W}} : \mathcal{W} \subseteq [\mathsf{K}] = [6])$.

By solving the linear program numerically, the lower bound on $\mathsf{R}^{\star}_{\mathsf{c,u}}$ given by the above method is $7/17 \approx 0.411$, while Theorem 9 gives $9/23 \approx 0.391$. □

Remark 10. *Notice that in* (6.21c)*,* $|\mathcal{Q}|\mathsf{R}^{\star}_{\mathsf{c,u}}\mathsf{B} \geq H(\mathcal{B}_1, \mathcal{B}_2, \mathcal{B}_3)$ *where* $\mathcal{B}_1 \cup \mathcal{B}_2$ *forms a directed circle. The techniques in this example provides a tighter converse bound compared to Theorem 9 because it allows to deal with cycles in the directed graph that represents the equivalent index coding problem.*

We are now ready to prove Theorem 10.

At first, choose a demand vector $\mathbf{d}$ where users demand different files. For a set of relays $\mathcal{Q}$, consider one division $\mathcal{Q} = \mathcal{Q}_1 \cup \ldots \cup \mathcal{Q}_a$ where $a \in [\lfloor |\mathcal{Q}|/\mathsf{r} \rfloor]$, and one combination of permutations $\mathbf{p}(\mathcal{K}_{\mathcal{Q}_1}), \ldots, \mathbf{p}(\mathcal{K}_{\mathcal{Q}_a}), \mathbf{p}\big(\mathcal{K}_{\mathcal{Q}} \setminus (\mathcal{K}_{\mathcal{Q}_1} \cup \cdots \cup \mathcal{K}_{\mathcal{Q}_a})\big)$. We let

$$\mathcal{B}_i = f\big(\mathbf{d}, [\mathsf{K}], \mathbf{p}(\mathcal{K}_{\mathcal{Q}_i})\big), \text{ where } i \in [a], \tag{6.24}$$

$$\mathcal{B}_{a+1} = f\Big(\mathbf{d}, [\mathsf{K}] \setminus (\mathcal{K}_{\mathcal{Q}_1} \cup \cdots \cup \mathcal{K}_{\mathcal{Q}_a}), \mathbf{p}\big(\mathcal{K}_{\mathcal{Q}} \setminus (\mathcal{K}_{\mathcal{Q}_1} \cup \cdots \cup \mathcal{K}_{\mathcal{Q}_a})\big)\Big). \tag{6.25}$$

For each $i \in [a]$, we use Proposition 2 with $\mathcal{J} = \mathcal{Q}_i$, $\mathcal{S}_1 = \mathcal{B}_i$ and $\mathcal{S}_2 = \mathcal{B}_{a+1}$ to obtain

$$H(\mathcal{B}_i) \leq |\mathcal{Q}_i|\mathsf{R}^{\star}_{\mathsf{c,u}}\mathsf{B} - \mathsf{H}\big(X_{\mathcal{Q}_i}|\mathcal{B}^{c}_{a+1}\big) + 2\mathsf{B}\varepsilon_{\mathsf{B}}. \tag{6.26}$$

We sum all the inequalities in the form of (6.26) for $i \in [a]$ and sum them to obtain

$$\sum_{i\in[a]} H(\mathcal{B}_i) \leq |\mathcal{Q}|\mathsf{R}^{\star}_{\mathsf{c,u}}\mathsf{B} - \sum_{i\in[a]} H\big(X_{\mathcal{Q}_i}|\mathcal{B}^{c}_{a+1}\big) + 2a\mathsf{B}\varepsilon_{\mathsf{B}} \tag{6.27a}$$

$$\leq |\mathcal{Q}|\mathsf{R}^{\star}_{\mathsf{c,u}}\mathsf{B} - H\big(X_{\mathcal{Q}}|\mathcal{B}^{c}_{a+1}\big) + 2a\mathsf{B}\varepsilon_{\mathsf{B}} \tag{6.27b}$$

$$\leq |\mathcal{Q}|\mathsf{R}^{\star}_{\mathsf{c,u}}\mathsf{B} - H(\mathcal{B}_{a+1}) + 2a\mathsf{B}\varepsilon_{\mathsf{B}}, \tag{6.27c}$$

where from (6.27a) to (6.27b) we use the submodularity of entropy, and from (6.27b) to (6.27c) we use (6.7a). We list all the inequalities in the form of (6.27c) for all the possible demands where users demand different files, and sum them to obtain (6.4).

6.6 Proof of Theorem 11

For a set $\mathcal{S}$ and a vector $\mathbf{p}$, where each element of $\mathcal{S}$ is also an element in $\mathbf{p}$, we define $g(\mathcal{S},\mathbf{p})$ as the vector obtained by removing the elements not in $\mathcal{S}$ from $\mathbf{p}$, e.g., $g(\{1,2,3\},(2,4,1,3))=(2,1,3)$. Our second improvement to Theorem 9 is explained by way of an example first.

Example 4. Consider the combination network in Fig. 1.4 with $\mathsf{N}=6$ and $\mathsf{M}=1/2$. Consider the demand vector $\mathbf{d}=(1,\ldots,6)$. For an integer $b\in[\mathsf{r}:\mathsf{H}]=[2:4]$, e.g., say $b=3$, consider each set of relays $\mathcal{Q}$ with cardinality b. Consider a permutation $\mathbf{p}_{\mathcal{K}_\mathcal{Q}}$ and apply Proposition 2 with $\mathcal{J}=\mathcal{Q}$ so as to obtain

$$|\mathcal{Q}|\mathsf{BR}^\star_{\mathsf{c,u}} \geq H\big(X_\mathcal{Q}\big) \geq H\big(\mathcal{S}_1\big)+H\big(X_\mathcal{Q}|\mathcal{S}_2^c\big)+2\mathsf{B}\varepsilon_\mathsf{B}, \tag{6.28}$$

$$\mathcal{S}_1 = f\big(\mathbf{d},[\mathsf{K}],\mathbf{p}(\mathcal{K}_\mathcal{Q})\big), \tag{6.29}$$

$$\mathcal{S}_2 = f\Big(\mathbf{d},[\mathsf{K}]\setminus\mathcal{K}_\mathcal{Q},g\big([\mathsf{K}]\setminus\mathcal{K}_\mathcal{Q},\mathbf{p}([\mathsf{K}])\big)\Big). \tag{6.30}$$

If $\mathcal{Q}=\{1,2,3\}$ and thus $\mathcal{K}_\mathcal{Q}=\{1,2,4\}$, we have

$$g\big([\mathsf{K}]\setminus\mathcal{K}_\mathcal{Q},\mathbf{p}([\mathsf{K}])\big)=g\big(\{3,5,6\},(1,\ldots,6)\big)=(3,5,6), \tag{6.31}$$

$$H\big(X_\mathcal{Q}|\mathcal{S}_2^c\big)=H\Big(X_{\{1,2,3\}}\big|f\big(\mathbf{d},\{3,5,6\},(3,5,6)\big)\Big). \tag{6.32}$$

We sum all the inequalities in the form of (6.28) for all the possible demands where the users request distinct files. With $\mathsf{B}\gg 1$, we have

$$|\mathcal{Q}|\mathsf{R}^\star_{\mathsf{c,u}} \geq \sum_{j\in[|\mathcal{K}_\mathcal{Q}|]}\ \sum_{\mathcal{W}\subseteq[\mathsf{K}]\setminus\cup_{k\in[j]}\{p_k(\mathcal{K}_\mathcal{Q})\}} x_\mathcal{W}+y_{\mathcal{Q},\mathbf{p}([\mathsf{K}])}, \tag{6.33}$$

$$y_{\mathcal{Q},\mathbf{p}([\mathsf{K}])} := \frac{1}{\mathsf{BK!}}\sum_{\mathbf{d}:d_i\neq d_j,i\neq j} H\big(X_\mathcal{Q}\big|\mathcal{S}_2^c\big) \text{ with } \mathcal{S}_2 \text{ in (6.30).} \tag{6.34}$$

For $\mathcal{Q}=\{1,2,3\}$ and $\mathcal{Q}=\{1,2,4\}$, we have

$$H\big(X_{\{1,2,3\}}\big|\mathcal{A}^c\big)+H\big(X_{\{1,2,4\}}\big|\mathcal{B}^c\big) \geq H\big(X_{\{1,2,3,4\}}|\{F_{6,\emptyset}\}^c\big)+H\big(X_{\{1,2\}}\big|\mathcal{A}^c\cap\mathcal{B}^c\big) \tag{6.35a}$$

$$\geq H(F_{6,\emptyset})+H\Big(X_{\{1,2\}}\big|\mathcal{A}^c\cap\mathcal{B}^c\Big), \tag{6.35b}$$

where

$$\mathcal{A}:=f\big(\mathbf{d},\{3,5,6\},(3,5,6)\big)=\big\{F_{3,\mathcal{W}}:\mathcal{W}\subseteq\{5,6\}\big\}\cup\{F_{5,\emptyset},F_{5,\{6\}},F_{6,\emptyset}\}, \tag{6.36}$$

$$\mathcal{B}:=f\big(\mathbf{d},\{2,4,6\},(2,4,6)\big)=\big\{F_{2,\mathcal{W}}:\mathcal{W}\subseteq\{4,6\}\big\}\cup\{F_{4,\emptyset},F_{4,\{6\}},F_{6,\emptyset}\}, \tag{6.37}$$

where to get (6.35a) we used Proposition 3 and the fact that $\mathcal{A}^c\cup\mathcal{B}^c=\{F_{6,\emptyset}\}^c$. Notice that without using Proposition 3 we cannot bound the sum of the two terms in the LHS (left hand side) of (6.35a). By using Proposition 3, we have the term $H\big(X_{\{1,2,3,4\}}|\{F_{6,\emptyset}\}^c\big)$ and all the relays connected to user 6 demanding

$F_{6,\emptyset}$ are in $\{1,2,3,4\}$ such that we can use Proposition 1 to bound this term by $H(F_{6,\emptyset})$. Similarly,

$$\sum_{\mathcal{Q}\subseteq[\mathsf{H}]:|\mathcal{Q}|=b} y_{\mathcal{Q},\mathbf{p}([\mathsf{K}])} \geq \frac{1}{\mathsf{K!B}} \sum_{\mathbf{d}:d_i\neq d_j, i\neq j} H(F_{1,\emptyset},\ldots,F_{6,\emptyset}) = 6x_\emptyset. \tag{6.38}$$

We then consider each permutation $\mathbf{p}(\mathcal{K}_\mathcal{Q})$ for $\mathcal{Q}\subseteq[\mathsf{K}]$ with $|\mathcal{Q}|=b=3$ to write inequalities in the form of (6.33). With the constraints in (6.3b), (6.3c) and (6.38) we can compute a lower bound on $\mathsf{R}^\star_{\mathsf{c,u}}$ by solving a linear program which gives $13/12$, while Theorem 9 gives $17/16$. Notice that in general we should consider each permutation $\mathbf{p}([\mathsf{K}])$ to write constraints in the form of (6.38), but in this example it is enough to consider one permutation. In order to reduce the number of variables, the constraint in (6.33) is equivalent to the following

$$|\mathcal{Q}|\mathsf{R}^\star_{\mathsf{c,u}} \geq \sum_{j\in[|\mathcal{K}_\mathcal{Q}|]} \sum_{\mathcal{W}\subseteq[\mathsf{K}]\backslash\cup_{k\in[j]}\{p_k(\mathcal{K}_\mathcal{Q})\}} x_\mathcal{W} + y_\mathcal{Q}, \tag{6.39}$$

$$y_\mathcal{Q} := \max_{\mathbf{p}([\mathsf{K}])} y_{\mathcal{Q},\mathbf{p}([\mathsf{K}])}, \tag{6.40}$$

satisfying for each permutation $\mathbf{p}([\mathsf{K}])$,

$$\sum_{\mathcal{Q}\subseteq[\mathsf{H}]:|\mathcal{Q}|=b} y_\mathcal{Q} \geq \sum_{\mathcal{Q}\subseteq[\mathsf{H}]:|\mathcal{Q}|=b} y_{\mathcal{Q},\mathbf{p}([\mathsf{K}])}, \tag{6.41}$$

where $y_{\mathcal{Q},\mathbf{p}([\mathsf{K}])}$ is defined in (6.34). □

Remark 11. *In Theorem 9, for each set $\mathcal{Q}$ we have the constraint in* (6.39) *but without $y_\mathcal{Q}$. The above example shows that the sum of all the $y_\mathcal{Q}$'s, where $|\mathcal{Q}|=b$, is positive, thus the lower bound in* (6.39) *is tighter than the one in Theorem 9.*

We are now ready to prove Theorem 10.

Choose an integer $b\in[\mathsf{r}:\mathsf{H}]$, a permutation $\mathbf{p}([\mathsf{K}])$. As we claimed in Section 6.6, for each set of relays $\mathcal{Q}$ where $|\mathcal{Q}|=b$, we consider a permutation $\mathbf{p}(\mathcal{K}_\mathcal{Q})$ to obtain the following constraint (shown in (6.39))

$$|\mathcal{Q}|\mathsf{R}^\star_{\mathsf{c,u}} \geq \sum_{j\in[|\mathcal{K}_\mathcal{Q}|]} \sum_{\mathcal{W}\subseteq[\mathsf{K}]\backslash\cup_{k\in[j]}\{p_k(\mathcal{K}_\mathcal{Q})\}} x_\mathcal{W} + y_\mathcal{Q}, \tag{6.42}$$

satisfying for each permutation $\mathbf{p}([\mathsf{K}])$,

$$\sum_{\mathcal{Q}\subseteq[\mathsf{H}]:|\mathcal{Q}|=b} y_\mathcal{Q} \geq \sum_{\mathcal{Q}\subseteq[\mathsf{H}]:|\mathcal{Q}|=b} y_{\mathcal{Q},\mathbf{p}([\mathsf{K}])} = \frac{1}{\mathsf{K!B}} \sum_{\mathbf{d}:d_i\neq d_j \text{ for } i\neq j} \sum_{\mathcal{Q}\subseteq[\mathsf{H}]:|\mathcal{Q}|=b} H(X_\mathcal{Q}|S^c_\mathcal{Q}), \tag{6.43}$$

$$S_\mathcal{Q} := f\Big(\mathbf{d}, [\mathsf{K}]\backslash\mathcal{K}_\mathcal{Q}, g\big([\mathsf{K}]\backslash\mathcal{K}_\mathcal{Q}, \mathbf{p}([\mathsf{K}])\big)\Big). \tag{6.44}$$

Now for one permutation $\mathbf{p}([\mathsf{K}])$ we want to compute the converse bound of $\sum_{\mathcal{Q}\subseteq[\mathsf{H}]:|\mathcal{Q}|=b} H(X_\mathcal{Q}|S^c_\mathcal{Q})$.

Consider two sets of relays with cardinality b, $\mathcal{J}_1$ and $\mathcal{J}_2$. We can use Proposition 3 to bound

$$H\big(X_{\mathcal{J}_1}|\mathcal{S}^c_{\mathcal{J}_1}\big) + H\big(X_{\mathcal{J}_2}|\mathcal{S}^c_{\mathcal{J}_2}\big) \geq H\big(X_{\mathcal{J}_1\cup\mathcal{J}_2}|(\mathcal{S}_{\mathcal{J}_1}\cap\mathcal{S}_{\mathcal{J}_2})^c\big) + H\big(X_{\mathcal{J}_1\cap\mathcal{J}_2}|(\mathcal{S}_{\mathcal{J}_1}\cup\mathcal{S}_{\mathcal{J}_2})^c\big). \tag{6.45}$$

Each time we use Proposition 3, we call the first term in the RHS result as 'cup' term and the second term as 'cap' term. We then add $H\big(X_{\mathcal{J}_3}|\mathcal{S}^c_{\mathcal{J}_3}\big)$ to the RHS of (6.45), where $\mathcal{J}_3$ is the third term in the sum $\sum_{\mathcal{Q}\subseteq[\mathsf{H}]:|\mathcal{Q}|=b} H(X_{\mathcal{Q}}|S^c_{\mathcal{Q}})$. Firstly, we use Proposition 3 to bound the sum of the cup term of (6.45) and $H(X_{\mathcal{J}_3}|\mathcal{S}^c_{\mathcal{J}_3})$. We put the cup term in the new RHS result at the first position, and use Proposition 3 again to bound the sum of the cap term in the RHS result of (6.45) and the cap term in the new RHS result. The cup term of the latest one is put at the second position while the cap term is at the third position. So after considering three sets, we now have three terms. Similarly, each time we consider the j^{th} set with cardinality b, we use Proposition 3 to bound the sum of the term in the first position of the last iteration and $H\big(X_{\mathcal{J}_j}|\mathcal{S}^c_{\mathcal{J}_j}\big)$. The cup term of the result is put at the first position in this iteration. The cap term of the result should be added to the term at the second position in the last iteration. We do this procedure until the term in the last position in the last iteration. We describe this iterative procedure in Algorithm 1.

Notice that when we use Proposition 3 to bound a sum of two terms, the cap term of the result may be 0. When we use Proposition 3 to bound the sum of 0 and one term, the result is also the sum of this term (seen as the cup term) and 0 (seen as the cap term). We should also notice that after each iteration, by assuming the term at the i_1^{th} is $H\big(X_{\mathcal{G}_{i_1}}|\mathcal{I}^c_{i_1}\big)$ and the term at the i_2^{th} is $H(X_{\mathcal{G}_{i_2}}|\mathcal{I}^c_{i_2})$ where $i_1 < i_2$, we can see that $\mathcal{G}_{i_2} \subseteq \mathcal{G}_{i_1}$ and $\mathcal{I}^c_{i_2} \subseteq \mathcal{I}^c_{i_1}$.

Algorithm 1 Iterative Procedure by using Proposition 3

1. **input**: $H\big(X_{\mathcal{J}_i}|\mathcal{S}^c_{\mathcal{J}_i}\big)$ where $i \in [\binom{\mathsf{H}}{b}]$; (each $\mathcal{J}_i$ is a distinct set of relays with cardinality b;) **initialization**: $t = 2$;
2. use Proposition 3 to bound $H\big(X_{\mathcal{J}_1}|\mathcal{S}^c_{\mathcal{J}_1}\big) + H\big(X_{\mathcal{J}_2}|\mathcal{S}^c_{\mathcal{J}_2}\big)$; let $L_{t,1}$ be the cup term and $L_{t,2}$ be the cap term;
3. use Proposition 3 to bound $L_{t,1} + H\big(X_{\mathcal{J}_{t+1}}|\mathcal{S}^c_{\mathcal{J}_{t+1}}\big)$; let $L_{t+1,1}$ be the cup term and T_{cap} be the cap term.
4. **for** $i = 2, \ldots, t$, use Proposition 3 to bound $L_{t,i} + T_{\text{cap}}$ and let $L_{t+1,i}$ be the cup term and T_{cap} be the cap term.
5. let $L_{t+1,t+1} = T_{\text{cap}}$.
6. **if** $t < \binom{\mathsf{H}}{b} - 1$, **then** $t = t + 1$ and go to 3).
7. **output**: $\sum_{i\in[\binom{\mathsf{H}}{b}]} L_{\binom{\mathsf{H}}{b},i}$.

After considering all the sets of relays with cardinality b, we have a summation including $\binom{\mathsf{H}}{b}$ terms. In the end, for an acyclic set of sub-files $\mathcal{S}$, by using Proposition 1 we have

$$H\big(X_{[\mathsf{H}]}|\mathcal{S}^c\big) \geq H(\mathcal{S}). \tag{6.46}$$

Hence, we can bound this summation by a sum of the lengths of sub-files, then we obtain

$$\sum_{\mathcal{Q}\subseteq[\mathsf{H}]:|\mathcal{Q}|=b} H(X_{\mathcal{Q}}|S^c_{\mathcal{Q}}) \geq \sum_{i\in[\mathsf{K}]} \sum_{\mathcal{W}\subseteq[\mathsf{K}]\setminus\cup_{j\in[i]}\{p_j([\mathsf{K}])\}} c\big(\{p_i([\mathsf{K}])\}\cup\mathcal{W},b\big)|F_{d_{p_i([\mathsf{K}])},\mathcal{W}}|, \tag{6.47}$$

where $c(\mathcal{W}_1,l) := \binom{\mathsf{H}-1}{l-1} - |\{\mathcal{Q}\subseteq[\mathsf{H}]:|\mathcal{Q}|=l,\mathcal{K}_{\mathcal{Q}}\nsubseteq[\mathsf{K}]\setminus\mathcal{W}_1\}|$, which will be proved in the following. We focus on $|F_{d_{p_i([\mathsf{K}])},\mathcal{W}}|$, where $\mathcal{W}\subseteq[\mathsf{K}]\setminus\{p_1([\mathsf{K}]),\ldots,p_i([\mathsf{K}])\}$. For a set of relays $\mathcal{Q}$ with cardinality b, in the term $H(X_{\mathcal{Q}}|\mathcal{S}^c_{\mathcal{Q}})$, we can see that $F_{d_{p_i([\mathsf{K}])},\mathcal{W}}\in\mathcal{S}_{\mathcal{Q}}$ if and only if $\mathcal{K}_{\mathcal{Q}}\cap(\{p_i([\mathsf{K}])\}\cup\mathcal{W})=\emptyset$, in other words $\mathcal{K}_{\mathcal{Q}}\subseteq[\mathsf{K}]\setminus(\{p_i([\mathsf{K}])\}\cup\mathcal{W})$. Focus on one relay $h\in[\mathsf{H}]$. When we use Proposition 3 to bound the sum of two terms by the sum of two new terms, if among the two terms in the LHS of Proposition 3, one term includes X_h not knowing $F_{d_{p_i([\mathsf{K}])},\mathcal{W}}$ and the other does not include X_h knowing $F_{d_{p_i([\mathsf{K}])},\mathcal{W}}$, we can see that the number of terms in the RHS of Proposition 3 including X_h not knowing $F_{d_{p_i([\mathsf{K}])},\mathcal{W}}$ decreases by 1 compared to the LHS (the number of terms in the RHS not including X_h but knowing $F_{d_{p_i([\mathsf{K}])},\mathcal{W}}$ also decreases by 1 compared to the LHS); otherwise, the number of terms in the RHS including X_h not knowing $F_{d_{p_i([\mathsf{K}])},\mathcal{W}}$ does not change compared to the LHS. In addition, it can be checked that in any case when we use Proposition 3, the number of terms in the RHS not including X_h but knowing $F_{d_{p_i([\mathsf{K}])},\mathcal{W}}$ does not increase compared to the LHS. Hence, among all of the terms in the summation after the final iteration, the number of terms including X_h not knowing $F_{d_{p_i([\mathsf{K}])},\mathcal{W}}$ is not less than

$$\begin{aligned}
&\max\Big\{|\{\mathcal{Q}\subseteq[\mathsf{H}]:|\mathcal{Q}|=b,h\in\mathcal{Q},\mathcal{K}_{\mathcal{Q}}\subseteq[\mathsf{K}]\setminus(\{p_i([\mathsf{K}])\}\cup\mathcal{W})\}|\\
&-|\{\mathcal{Q}\subseteq[\mathsf{H}]:|\mathcal{Q}|=b,h\notin\mathcal{Q},\mathcal{K}_{\mathcal{Q}}\nsubseteq[\mathsf{K}]\setminus(\{p_i([\mathsf{K}])\}\cup\mathcal{W})\}|,0\Big\}\\
=&\max\left\{\binom{\mathsf{H}-1}{l-1}-|\{\mathcal{Q}\subseteq[\mathsf{H}]:|\mathcal{Q}|=l,\mathcal{K}_{\mathcal{Q}}\nsubseteq[\mathsf{K}]\setminus\mathcal{W}_1\}|,0\right\}.
\end{aligned} \tag{6.48}$$

In addition, after the final iteration, the term at the i_1^{th} position is $H(X_{\mathcal{G}_{i_1}}|\mathcal{I}^c_{i_1})$ and the term at the i_2^{th} position is $H(X_{\mathcal{G}_{i_2}}|\mathcal{I}^c_{i_2})$ where $i_1<i_2$, we can see that $\mathcal{G}_{i_2}\subseteq\mathcal{G}_{i_1}$ and $\mathcal{I}^c_{i_2}\subseteq\mathcal{I}^c_{i_1}$. So by (6.46) we have, $c(\mathcal{W}_1,l)=\binom{\mathsf{H}-1}{l-1}-|\{\mathcal{Q}\subseteq[\mathsf{H}]:|\mathcal{Q}|=l,\mathcal{K}_{\mathcal{Q}}\nsubseteq[\mathsf{K}]\setminus\mathcal{W}_1\}|$ as defined in (6.5b).

Finally, from (6.43), (6.47) and the value of $c(\mathcal{W}_1,l)$, we can obtain (6.5a) to finish the proof.

6.7 Proof of Theorem 12

Consider one demand vector where users demand different files and one division of $\mathcal{Q}=\mathcal{Q}_1\cup\cdots\cup\mathcal{Q}_a$. We let

$$\mathcal{B}_i = f\big(\mathbf{d},[\mathsf{K}],\mathbf{p}(\mathcal{K}_{\mathcal{Q}_i})\big),\ \text{where } i\in[a], \tag{6.49}$$

$$\mathcal{B}_{a+1} = f\Big(\mathbf{d},[\mathsf{K}]\setminus(\mathcal{K}_{\mathcal{Q}_1}\cup\cdots\cup\mathcal{K}_{\mathcal{Q}_a}),\mathbf{p}\big(\mathcal{K}_{\mathcal{Q}}\setminus(\mathcal{K}_{\mathcal{Q}_1}\cup\cdots\cup\mathcal{K}_{\mathcal{Q}_a})\big)\Big) \tag{6.50}$$

$$\mathcal{B}_{a+2} = f\Big(\mathbf{d},[\mathsf{K}]\setminus\mathcal{K}_{\mathcal{Q}},g\big([\mathsf{K}]\setminus\mathcal{K}_{\mathcal{Q}},\mathbf{p}([\mathsf{K}])\big)\Big). \tag{6.51}$$

For each $i \in [a]$, we write an inequality in the form of (6.9) by $\mathcal{J} = \mathcal{Q}_i$, $\mathcal{S}_1 = \mathcal{B}_i$, $\mathcal{S}_2 = \mathcal{B}_{a+1} \cup \mathcal{B}_{a+2}$. We then sum all of the a inequalities to obtain

$$\sum_{i\in[a]} H(\mathcal{B}_i) \le |\mathcal{Q}|\mathsf{R}^{\star}_{\mathsf{c,u}}\mathsf{B} - \sum_{i\in[a]} H\big(X_{\mathcal{Q}_i}|(\mathcal{B}_{a+1}\cup\mathcal{B}_{a+2})^c\big) + 2a\mathsf{B}\varepsilon_{\mathsf{B}} \tag{6.52a}$$

$$\le |\mathcal{Q}|\mathsf{R}^{\star}_{\mathsf{c,u}}\mathsf{B} - H\big(X_{\mathcal{Q}}|(\mathcal{B}_{a+1}\cup\mathcal{B}_{a+2})^c\big) + 2a\mathsf{B}\varepsilon_{\mathsf{B}} \tag{6.52b}$$

$$= |\mathcal{Q}|\mathsf{R}^{\star}_{\mathsf{c,u}}\mathsf{B} - H\big(X_{\mathcal{Q}}|\mathcal{B}^c_{a+2}\big) - I\big(X_{\mathcal{Q}};\mathcal{B}_{a+1}|(\mathcal{B}_{a+1}\cup\mathcal{B}_{a+2})^c\big) + 2a\mathsf{B}\varepsilon_{\mathsf{B}} \tag{6.52c}$$

$$= |\mathcal{Q}|\mathsf{R}^{\star}_{\mathsf{c,u}}\mathsf{B} - H\big(X_{\mathcal{Q}}|\mathcal{B}^c_{a+2}\big) - H(\mathcal{B}_{a+1}) + H\big(\mathcal{B}_{a+1}|(\mathcal{B}_{a+1}\cup\mathcal{B}_{a+2})^c, X_{\mathcal{Q}}\big) + 2a\mathsf{B}\varepsilon_{\mathsf{B}} \tag{6.52d}$$

$$\le |\mathcal{Q}|\mathsf{R}^{\star}_{\mathsf{c,u}}\mathsf{B} - H\big(X_{\mathcal{Q}}|\mathcal{B}^c_{a+2}\big) - H(\mathcal{B}_{a+1}) + (2a+1)\mathsf{B}\varepsilon_{\mathsf{B}}, \tag{6.52e}$$

where from (6.52a) to (6.52b) the submodularity of entropy is used, and from (6.52d) to (6.52e) we use Fano's inequality and the fact that $\mathcal{B}_{a+1}$ is acyclic and $\mathcal{B}_{a+2}$ does not include the side information of the user requiring $\mathcal{B}_{a+1}$. We then use the similar method in Theorem 11 to bound $H(X_{\mathcal{Q}}|\mathcal{B}^c_{a+2})$.

Remark 12. *If* $\mathsf{N} < \mathsf{K}$*, we can not find a demand vector where each user has distinct demand. In this case, it is possible to find demand vectors such that every user demands a different file and, in turns, it is relatively straightforward to apply the "acyclic index coding converse bound" (see Proposition 1). When* $\mathsf{N} < \mathsf{K}$ *one should consider many subsystems with only* $\min(\mathsf{N},\mathsf{K}) = \mathsf{N}$ *users with distinct demands (as we did in Section 4.2.1 for shared-link model), which is not conceptually more difficult but requires a somewhat heavier notation.*

Chapter 7

Novel Inner Bounds for Combination Networks with End-user-caches

7.1 Chapter Overview and Related Publications

This chapter provides novel inner bounds for combination networks with end-user-caches. The achievable schemes for combination networks could be divided into separation and non-separation approaches, dependent of whether we generate the multicast messages on the network topology. In Section 7.2, based on separation approach, we propose four schemes to deliver cMAN multicast messages. Based on non-separation approach, we first propose a delivery scheme in Section 7.3 by generating multicast messages on network topology. With this delivery scheme, In Section 7.4 we then propose an asymmetric coded placemen.

Related Publications

1. Kai Wan, Mingyue Ji, Pablo Piantanida and Daniela Tuninetti, "Caching in Combination Networks: Novel Asymmetric Coded Cache Placement and Multicast Message Generation by Leveraging Network Topology", *in preparation*, to IEEE Trans. on Information Theory. [54]

2. Kai Wan, Mingyue Ji, Pablo Piantanida and Daniela Tuninetti, "Novel Outer Bounds and Inner Bounds with Uncoded Cache Placement for Combination Networks with End-User- Caches", *in preparation*, to IEEE Trans. on Information Theory. [55]

3. Kai Wan, Mingyue Ji, Pablo Piantanida and Daniela Tuninetti, "On the Benefits of Asymmetric Coded Cache Placement in Combination Networks with End-User Caches", in IEEE Int. Symp. Inf. Theory (ISIT), Jun. 2018. [56]

4. Kai Wan, Mingyue Ji, Pablo Piantanida and Daniela Tuninetti, "Caching in Combination Networks: Novel Multicast Message Generation and Delivery by Leveraging the Network Topology", in IEEE Intern. Conf. Commun. (ICC), May 2018. [57]

5. Kai Wan, Daniela Tuninetti, Pablo Piantanida and Mingyue Ji, "A Novel Asymmetric Coded Placement in Combination Networks with end-user Caches", in Proceedings of IEEE Infor. Theory Application Workshop (ITA), Feb. 2018. [59]

6. Kai Wan, Daniela Tuninetti, Mingyue Ji, and Pablo Piantanida, "State-of-the-art in Cache-aided Combination Networks", in Proceedings of the IEEE Asilomar Conf., Nov 2017. [61]

7. Kai Wan, Mingyue Ji, Pablo Piantanida and Daniela Tuninetti, "Novel Inner Bounds for Combination Networks with End-User-Caches", in Proceedings of the 55th Allerton Conf. Commun., Control, Comp., Oct. 2017. [62]

8. Kai Wan, Mingyue Ji, Pablo Piantanida and Daniela Tuninetti, "Caching in Combination Networks: A Novel Delivery by Leveraging the Network Topology", *in preparation*, to IEEE Inf. Theory Workshop (ITW), available at arXiv:1802.10479, Apr. 2018. [65]

7.2 Novel Separation Based Achievable Schemes

Based on separation approach where cMAN placement in (2.13) and cMAN multicast message generation in (2.14) are used independent of the network topology, we proposed four delivery schemes to deliver cMAN multicast messages $W_{\mathcal{J}}=\oplus_{k\in\mathcal{J}}F_{d_k,\mathcal{J}\backslash\{k\}}$ to users in $\mathcal{J}$ where $\mathcal{J}\subseteq[\mathsf{K}]$, $|\mathcal{J}|=t+1$ and $t=\frac{\mathsf{KM}}{\mathsf{N}}\in[0:\mathsf{K}]$. Recall that $\mathcal{R}_{\mathcal{J}}$ is set of relays connected to all the users in $\mathcal{J}$ and we define that

$$\mathcal{V}_1:=\{\mathcal{J}\subseteq[\mathsf{K}]:|\mathcal{J}|=t+1,\mathcal{R}_{\mathcal{J}}\neq\emptyset\}, \tag{7.1}$$

$$\mathcal{V}_2:=\{\mathcal{J}\subseteq[\mathsf{K}]:|\mathcal{J}|=t+1,\mathcal{R}_{\mathcal{J}}=\emptyset\}. \tag{7.2}$$

Based on Separation approach, in Section 7.2.1 we first propose an improved scheme, namely Direct Independent delivery Scheme (DIS), compared to the second scheme in [12] by observing that the cMAN multicast message $W_{\mathcal{J}}$ is only useful to the users in $\mathcal{J}$ and thus we need not to let all users recover it. The achieved max link-load in the following theorem.

Theorem 13. *For a* $(\mathsf{H},\mathsf{r},\mathsf{M},\mathsf{N})$ *combination network with* $t=\mathsf{KM}/\mathsf{N}\in[0:\mathsf{K}]$*, the max link-load satisfies*

$$\mathsf{R}^{\star}_{\mathsf{c,u}}\leq\mathsf{R}_{\mathrm{DIS}}:=\frac{\sum_{\mathcal{J}\notin\mathcal{V}}1+\sum_{\mathcal{J}\in\mathcal{V}}|\{h\in[\mathsf{H}]:\mathcal{U}_h\cap\mathcal{J}\neq 0\}|}{\mathsf{H}\binom{\mathsf{K}}{t}}. \tag{7.3}$$

The tradeoff between memory size and max link-load is the lower convex envelope of the above points.

We then propose Interference Elimination delivery Scheme (IES) in Section 7.2.2 to improve DIS when $\mathsf{M}=\mathsf{N}/\mathsf{K}$, to achieve the max link-load in the following Theorem.

Theorem 14. *For a* $(\mathsf{H},\mathsf{r},\mathsf{M}=\mathsf{N}/\binom{\mathsf{H}}{\mathsf{r}},\mathsf{N})$ *combination network, the max link-load satisfies*

$$\mathsf{R}^{\star}_{\mathsf{c,u}}\leq\mathsf{R}_{\mathrm{IES}}:=\frac{1}{2\mathsf{H}}\left(\mathsf{K}-1-\binom{\mathsf{H}-\mathsf{r}}{\mathsf{r}}\right)+\frac{\binom{2\mathsf{r}-1}{\mathsf{r}-1}}{(2\mathsf{r}-1)\mathsf{K}}\binom{\mathsf{H}}{2\mathsf{r}}, \tag{7.4}$$

where $\mathsf{R}_{\mathrm{IES}}$ *achieved by IES if* $2\mathsf{r}-1=p^v$ *or* pq*, where* p,q *are different primes and* v *is a positive integer.*[1]

We then propose Concatenated Inner Code delivery Scheme (CICS) in Section 7.2.3, which includes two phases to deliver each $W_{\mathcal{J}}$ to the users in $\mathcal{J}$, to achieve the max link-load in the following theorem.

[1] The smallest value of r not satisfying the condition in Theorem 29 is 23. If $\mathsf{r}=23$ and $\mathsf{H}=2\mathsf{r}=46$, in the network there are more than 8.23×10^{12} users, which is not practical. Note that the proof in Appendix A.7 fails when $\mathsf{r}\geq23$ but this does not imply that the group division cannot be found for $\mathsf{r}\geq23$. We conjecture that the claim of the theorem is true in general.

Theorem 15. *For a* $(\mathsf{H},\mathsf{r},\mathsf{M},\mathsf{N})$ *combination network with* $t=\mathsf{KM}/\mathsf{N}\in[0:\mathsf{K}]$*, the max link-load is*

$$\mathsf{R}^{\star}_{\mathrm{c,u}}\le \mathsf{R}_{\mathrm{CICS}}:=\sum_{\mathcal{J}\subseteq[\mathsf{K}]:|\mathcal{J}|=t}\frac{1+\min_{h\in[\mathsf{H}]}|\mathcal{J}\setminus\mathcal{U}_h|/\mathsf{r}}{\mathsf{H}\binom{\mathsf{K}}{t}}, \tag{7.5}$$

where $\mathsf{R}_{\mathrm{CICS}}$ *is achieved by CICS. The tradeoff between memory size and max link-load is the lower convex envelope of the above points.*

By leveraging the ignored multicasting opportunities of CICS, in Section 7.2.4 we propose Improved Concatenated Inner Code delivery Scheme (ICICS).

7.2.1 Direct Independent delivery Scheme (DIS)

We propose a delivery scheme, namely DIS, based on the observation that each cMAN multicast message $W_{\mathcal{J}}$ is only useful to the users in $\mathcal{J}$; therefore, if $\mathcal{R}_{\mathcal{J}}\neq\emptyset$, i.e., there exists at least one relay that is connected to all the users in $\mathcal{J}$, it is enough to transmit $W_{\mathcal{J}}$ only to the relays in $\mathcal{R}_{\mathcal{J}}$; this observation motivates the next steps.

Step 1: For each $\mathcal{J}\in\mathcal{V}_1$, the source divides $W_{\mathcal{J}}$ into $|\mathcal{R}_{\mathcal{J}}|$ non-overlapping pieces with equal length and directly transmits each different piece to the relays in $\mathcal{R}_{\mathcal{J}}$.

Step 2: For each $\mathcal{J}\in\mathcal{V}_2$, the source node divide $W_{\mathcal{J}}$ into r non-overlapping equal-length pieces, which are then encoded by an $(|\{h\in[\mathsf{H}]:\mathcal{U}_h\cap\mathcal{J}\neq\emptyset\}|,\mathsf{r})$ MDS code. We then transmit one different MDS symbol to each relay $h'\in\{h\in[\mathsf{H}]:\mathcal{U}_h\cap\mathcal{J}\neq\emptyset\}$. Each user in $\mathcal{J}$ is connected to r relays, and thus can receive r MDS symbols of $W_{\mathcal{J}}$ to recover $W_{\mathcal{J}}$.

By Step 1 and Step 2, each user $k\in[\mathsf{K}]$ can recover all the multicast messages $W_{\mathcal{J}}$ where $k\in\mathcal{J}$ and then recover the sub-file $F_{d_k,\mathcal{J}\setminus\{k\}}$. The resulting load is in (7.3).

7.2.2 Interference Elimination delivery Scheme (IES) for the Case $t=1$

We propose a delivery scheme, namely IES, for $\mathsf{M}=\mathsf{N}/\mathsf{K}$ ($t=1$). Notice that when $\mathsf{H}<2\mathsf{r}$ and $\mathsf{M}=\mathsf{N}/\mathsf{K}$, $\mathcal{V}_2$ contains the sets $\mathcal{J}$ where no relay is connected to all the users in $\mathcal{J}$ with $|\mathcal{J}|=t+1=2$. So we have $\mathcal{V}_2=\emptyset$.

Delivery of multicast messages in $\mathcal{V}_1$ For each $\mathcal{J}\in\mathcal{V}_1$ with $\mathcal{V}_1$ in (7.1), we divide $W_{\mathcal{J}}$ into $|\mathcal{R}_{\mathcal{J}}|$ non-overlapping and equal-length pieces, i.e., $W_{\mathcal{J}}=\{W_{\mathcal{J},h}:h\in\mathcal{R}_{\mathcal{J}}\}$; we transmit $W_{\mathcal{J},h}$ to each relay $h\in\mathcal{R}_{\mathcal{J}}$. Note that each relay $h\in\mathcal{R}_{\mathcal{J}}$ is connected to all of the users in $\mathcal{J}$. Hence, if relay h forwards $W_{\mathcal{J},h}$ to the users in $\mathcal{J}$, each user in $\mathcal{J}$ can recover $W_{\mathcal{J}}$. The max-link load to deliver the coded multicast messages in $\mathcal{V}_1$ is $\frac{1}{2\mathsf{H}}\left(\binom{\mathsf{H}}{\mathsf{r}}-1-\binom{\mathsf{H}-\mathsf{r}}{\mathsf{r}}\right)$.

The key idea of IES is using an interference elimination scheme to transmit the messages $W_{\mathcal{J}}$ where $\mathcal{J}\in\mathcal{V}_2$. We examine three examples to highlight the key idea.

Example 5 ($\mathsf{H}=2\mathsf{r},\mathsf{r}=2$)**.** Consider the network in Fig. 1.4 with $\mathsf{N}=6$ and $\mathsf{M}=1$. Assume that $\mathbf{d}=(1:6)$, so that

$$\mathcal{V}_1=\big\{\{1,2\},\{1,3\},\{1,4\},\{1,5\},\{2,3\},\{2,4\},\{2,6\},\{3,5\},\{3,6\}\big\}$$

$$\mathcal{V}_2 = \{\{1,6\},\{2,5\},\{3,4\}\}.$$

For the messages in $\mathcal{V}_1$, the server transmits

$$\begin{aligned} &W_{\{1,2\}}, W_{\{1,3\}}, W_{\{2,3\}} \text{ to relay 1},\\ &W_{\{1,4\}}, W_{\{1,5\}}, W_{\{4,5\}} \text{ to relay 2},\\ &W_{\{2,4\}}, W_{\{2,6\}}, W_{\{4,6\}} \text{ to relay 3},\\ &W_{\{3,5\}}, W_{\{3,6\}}, W_{\{5,6\}} \text{ to relay 4}. \end{aligned}$$

We then use an interference elimination scheme to transmit the messages $W_{\mathcal{J}}$ where $\mathcal{J} \in \mathcal{V}_2$ and $\mathcal{V}_2 = \{\{1,6\},\{2,5\},\{3,4\}\}$. Consider user 1. User 1 is connected to relays 1 and 2, and only needs to recover $W_{\{1,6\}}$ (i.e., $W_{\{2,5\}}$ and $W_{\{3,4\}}$ are interference). In order to perform IES we transmit (with operations in some finite field[2])

$$\begin{aligned} X_1 &:= +W_{\{1,6\}} + W_{\{2,5\}} + W_{\{3,4\}} \text{ to relay 1},\\ X_2 &:= +W_{\{1,6\}} - W_{\{2,5\}} - W_{\{3,4\}} \text{ to relay 2},\\ X_3 &:= -W_{\{1,6\}} + W_{\{2,5\}} - W_{\{3,4\}} \text{ to relay 3},\\ X_4 &:= -W_{\{1,6\}} - W_{\{2,5\}} + W_{\{3,4\}} \text{ to relay 4}, \end{aligned}$$

and the relays then forward what they received to the users. Then, user 1 computes $X_1 + X_2 = +2W_{\{1,6\}}$ and recovers $W_{\{1,6\}}$. Similarly, user 2 computes $X_1 + X_3 = 2W_{\{2,5\}}$, and so on. With this, all users recover the missing sub-files for the demanded file. Since the length of $W_{\mathcal{J}}$, equal to $\mathsf{B}/6$, goes to infinity as $\mathsf{B} \to \infty$, we can divide each $W_{\mathcal{J}}$ where $\mathcal{J} \in \mathcal{V}_2$ into P sub-packets with length $\mathsf{B}/(6P)$ such that we can do operations among multicast messages on a finite field of size 3. Notice that each file is composed of $6P$ sub-packets.

The load to transmit $W_{\mathcal{J}}$ where $\mathcal{J} \in \mathcal{V}_2$ is $P/(6P) = 1/6$, while the load to transmit $W_{\mathcal{J}}$ where $\mathcal{J} \notin \mathcal{V}$ is $1/2$. Hence, the max link-load of this scheme is $2/3 = 1/2 + 1/6$, coinciding with the enhanced cut-set converse bound in (6.1), while that of CICS (ICICS) and the schemes in references [12], and [50] are 0.6875, $5/4$, and 1, respectively.

In the next example, we generalize IES to transmit $W_{\mathcal{J}}$ where $\mathcal{J} \in \mathcal{V}_2$ to the case $\mathsf{H} > 2\mathsf{r}$ and $\mathsf{r} = 2$. □

Example 6 ($\mathsf{H} > 2\mathsf{r}, \mathsf{r} = 2$)**.** Consider the combination network with $\mathsf{H} = 5$, $\mathsf{r} = 2$, $\mathsf{M} = 1$ and $\mathsf{K} = \mathsf{N} = 10$. Assume that $\mathbf{d} = (1:10)$, so that

$$\begin{aligned} &\mathcal{U}_1 = [1:4],\ \mathcal{U}_2 = \{1,5,6,7\},\ \mathcal{U}_3 = \{2,5,8,9\},\ \mathcal{U}_4 = \{3,6,8,10\},\ \mathcal{U}_5 = \{4,7,9,10\},\\ &\mathcal{V}_2 = \big\{\{1,8\},\{1,9\},\{1,10\},\{2,6\},\{2,7\},\{2,10\},\{3,5\},\end{aligned}$$

[2] We would like to perform the operations "+" and "−" on some finite field of sufficiently large size. In other words, with an abuse of notation, we use $W_{\mathcal{J}}$ to denote both the binary multicast messages as well as its representation on some higher field size. In this example we need a field size larger than 3; on GF(3) the "−1" becomes "+2" so that $X_1 = W_{\{1,6\}} + W_{\{2,5\}} + W_{\{3,4\}}$, $X_2 = W_{\{1,6\}} + 2W_{\{2,5\}} + 2W_{\{3,4\}}$, and thus $X_1 + X_2 = (2W_{\{1,6\}} + 3W_{\{2,5\}} + 3W_{\{3,4\}}) \bmod 3 = 2W_{\{1,6\}}$, etc.

$$\{3,7\},\{3,9\},\{4,5\},\{4,6\},\{4,8\},\{5,10\},\{6,9\},\{7,8\}\big\}.$$

The max link-load to transmit $W_{\mathcal{J}}$ where $\mathcal{J} \in \mathcal{V}_1$ is $3/5$. For $W_{\mathcal{J}}$ where $\mathcal{J} \in \mathcal{V}_2$, we expand on IES introduced in Example 5.

For each subset of relays $\mathcal{B} \subseteq [\mathsf{H}]$ with cardinality $|\mathcal{B}| = 2\mathsf{r} = 4$, the set of users connected to $\mathsf{r} = 2$ of the chosen relays is denoted as $\mathcal{P}_{\mathcal{B}}$ and we also let

$$\mathcal{T}_{\mathcal{B}} := \{\mathcal{J} : \mathcal{J} \in \mathcal{V}_2, \mathcal{J} \subseteq \mathcal{P}_{\mathcal{B}}\}, \tag{7.6}$$

to be the set of multicast messages in $\mathcal{V}_2$ which are useful to some users in $\mathcal{P}_{\mathcal{B}}$ and not useful to the users not in $\mathcal{P}_{\mathcal{B}}$. We then use the scheme in Example 5 to transmit the codewords for the multicast messages in $\mathcal{T}_{\mathcal{B}}$ through the relays in $\mathcal{B}$.

For example, for $\mathcal{B} = \{1,2,3,4\}$ we have $\mathcal{P}_{\{1,2,3,4\}} = \{1,2,3,5,6,8\}$ and $\mathcal{T}_{\{1,2,3,4\}} = \big\{\{1,8\},\{2,6\},\{3,5\}\big\}$; we transmit

$$\begin{aligned}
&+ W_{\{1,8\}} + W_{\{2,6\}} + W_{\{3,5\}} \text{ to relay } 1,\\
&+ W_{\{1,8\}} - W_{\{2,6\}} - W_{\{3,5\}} \text{ to relay } 2,\\
&- W_{\{1,8\}} + W_{\{2,6\}} - W_{\{3,5\}} \text{ to relay } 3,\\
&- W_{\{1,8\}} - W_{\{2,6\}} + W_{\{3,5\}} \text{ to relay } 4.
\end{aligned}$$

Hence, each multicast messages $W_{\mathcal{J}}$ where $\mathcal{J} \in \mathcal{T}_{\{1,2,3,4\}}$ can be recovered by the users in $\mathcal{J}$. We proceed similarly to transmit the codewords for the multicast messages in $\mathcal{T}_{\mathcal{B}}$ through the relays $\mathcal{B}$ such that each user in $\mathcal{J}$ can recover $W_{\mathcal{J}}$ where $\mathcal{J} \in \mathcal{T}_{\mathcal{B}}$.

The load to transmit the multicast messages in $\mathcal{V}_2$ is $2/5$ so that the max link-load of this scheme is $3/5 + 2/5 = 1$, while that of CICS (ICICS) and the schemes in references [12] and [50] are 1.05, 2.25, and 1.5, respectively. □

In the final example, we generalize IES to transmit $W_{\mathcal{J}}$ where $\mathcal{J} \in \mathcal{V}_2$ to any $\mathsf{r} \geq 2$.

Example 7 ($\mathsf{H} \geq 2\mathsf{r}, \mathsf{r} > 2$)**.** Consider the combination network with $\mathsf{H} = 6$, $\mathsf{r} = 3$, $\mathsf{K} = \mathsf{N} = 20$, $\mathsf{M} = 1$. Assume that $\mathbf{d} = (1:20)$ so that

$$\begin{aligned}
\mathcal{U}_1 &= \{1,2,3,4,5,6,7,8,9,10\},\\
\mathcal{U}_2 &= \{1,2,3,4,11,12,13,14,15,16\},\\
\mathcal{U}_3 &= \{1,5,6,7,11,12,13,17,18,19\},\\
\mathcal{U}_4 &= \{2,5,8,9,11,14,15,17,18,20\},\\
\mathcal{U}_5 &= \{3,6,8,10,12,14,16,17,19,20\},\\
\mathcal{U}_6 &= \{4,7,9,10,13,15,16,18,19,20\},\\
\mathcal{V}_2 &= \big\{\{1,20\},\{2,19\},\{3,18\},\{4,17\},\{5,16\},\{6,15\},\{7,14\},\{8,13\},\{9,12\},\{10,11\}\big\}.
\end{aligned}$$

We focus on the transmission of the multicast messages in $\mathcal{V}_2$. If $\mathsf{H} = 2\mathsf{r}$ (the case $\mathsf{H} > 2\mathsf{r}$ can be dealt as in Example 6), there are $\binom{2\mathsf{r}}{\mathsf{r}}/2 = \binom{2\mathsf{r}-1}{\mathsf{r}-1} = 10$ elements in $\mathcal{V}_2$. We partition $\mathcal{V}_2$ into $\binom{2\mathsf{r}-1}{\mathsf{r}-1}/(2\mathsf{r}-1) = 2$ groups where each group contains $2\mathsf{r}-1 = 5$ elements. The existence of such a partition is discussed in Appendix A.7. In this example the groups could be chosen as

$$\mathcal{G}_1 = \big\{\{1,20\},\{2,19\},\{3,18\},\{5,16\},\{7,14\}\big\}, \tag{7.7}$$

$$\mathcal{G}_2 = \big\{\{4,17\},\{6,15\},\{8,13\},\{9,12\},\{10,11\}\big\}. \tag{7.8}$$

We focus on group $\mathcal{G}_1$; the same applies to $\mathcal{G}_2$. The vector of multicast messages in $\mathcal{G}_1$ is indicated by $\mathbb{M}_1 = [W_{\{1,20\}}; W_{\{2,19\}}; W_{\{3,18\}}; W_{\{5,16\}}; W_{\{7,14\}}]$. We now design a linear code for $\mathbb{M}_1$ on some finite field; we denote the coding matrix by $\mathbb{A}_1$, the element on i^{th} row j^{th} column of $\mathbb{A}_1$ by $a_{1,i,j}$, and the i^{th} row of $\mathbb{A}_1$ by $\mathbb{A}_{1,i}$; $\mathbb{A}_{1,i} \times \mathbb{M}_1$ represents the codeword for $\mathbb{M}_1$ transmitted to relay $i \in [6]$.

Recall that each multicast messages $W_{\mathcal{J}} : \mathcal{J} \in \mathcal{G}_1$ is useful to the users in $\mathcal{J}$ and is interference to the users in $\cup_{\mathcal{J}_1 \in \mathcal{G}_1 : \mathcal{J}_1 \neq \mathcal{J}} \mathcal{J}_1$. Our objective is to eliminate the interference caused by $W_{\mathcal{J}}$ to those users who do not need it. To achieve our objective, we construct the linear code as follows.

We focus on $W_{\{1,20\}}$ (assumed to be the first element in the vector $\mathbb{M}_1$). Firstly, on some finite field, we let

$$\sum_{i \in [6]} a_{1,i,1} = 0. \tag{7.9a}$$

The reason for the choice in (7.9a) will become clear later. Then, for each $\mathcal{J}_1 \in \mathcal{G}_1 \backslash \{\{1,20\}\}$ we eliminate the interference caused be $W_{\{1,20\}}$ to users in $\mathcal{J}_1 = \{k_1, k_2\}$ (where k_1 is the user connected to relay 6) by letting [3]

$$\sum_{i \in \mathcal{H}_{k_1}} a_{1,i,1} = 0, \tag{7.9b}$$

It can be seen that each set $\mathcal{J} \in \mathcal{V}$ contains two users k_1 and k_2 for which $\mathcal{H}_{k_1} \cap \mathcal{H}_{k_2} = \emptyset$ and $\mathcal{H}_{k_1} \cup \mathcal{H}_{k_2} = [\mathsf{H}]$, therefore, from (7.9a) and (7.9b), we have

$$\sum_{i \in \mathcal{H}_{k_2}} a_{1,i,1} = 0. \tag{7.9c}$$

Hence, if user k_1 and k_2 sum their received codewords from their connected relays, they can eliminate the interference caused by $W_{\{1,20\}}$. This construction is repeated for all $\mathcal{J}_1 \in \mathcal{G}_1 \backslash \{\{1,20\}\}$. Lastly, since user 20 requires $W_{\{1,20\}}$, we let

$$\sum_{i \in \mathcal{H}_{20}} a_{1,i,1} = s : s \neq 0. \tag{7.9d}$$

[3] For each pair of users $\mathcal{J}_1 = \{k_1, k_2\}$, we can choose any one of them (denoted by k') and write the equation $\sum_{i \in \mathcal{H}_{k'}} a_{1,i,1} = 0$ which can also lead to $\sum_{i \notin \mathcal{H}_{k'}} a_{1,i,1} = 0$ from (7.9a). Thus, the interference of $W_{\{1,20\}}$ for these two users is eliminated. Here we choose the user connected to relay 6, i.e., $k' = k_1$.

Since $\mathcal{H}_1 \cap \mathcal{H}_{20} = \emptyset$ and $\mathcal{H}_1 \cup \mathcal{H}_{20} = [\mathsf{H}]$, from (7.9a) and (7.9d), it can be seen that

$$\sum_{i \in \mathcal{H}_1} a_{1,i,1} = -s \neq 0. \tag{7.9e}$$

Hence, if user 1 and 20 sum their received codewords from their connected relays, they can recover $W_{\{1,20\}}$.

To summarize, by collecting all the constraint in (7.9) we obtain the following system of equations to solve

$$\begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 \end{bmatrix} \times \begin{bmatrix} a_{1,1,1} \\ a_{1,2,1} \\ a_{1,3,1} \\ a_{1,4,1} \\ a_{1,5,1} \\ a_{1,6,1} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ s \end{bmatrix}. \tag{7.10}$$

If we let for example $s = -3$, we have $[a_{1,1,1}; a_{1,2,1}; a_{1,3,1}\ ; a_{1,4,1}; a_{1,5,1}; a_{1,6,1}] = [1; -2; -2; 1; 1; 1]$. Similarly, we can get all the elements in $\mathbb{A}_1$.

Hence, for all the multicast messages $W_{\mathcal{J}}$ where $\mathcal{J} \in \mathcal{G}_1$, we transmit (note that the following operations can be done on a finite field with size larger than 7)

$$\begin{bmatrix} X_{1,1} \\ X_{2,1} \\ X_{3,1} \\ X_{4,1} \\ X_{5,1} \\ X_{6,1} \end{bmatrix} = \begin{bmatrix} +1 & +1 & -2 & +1 & -2 \\ -2 & +1 & +1 & -2 & +1 \\ -2 & -2 & +1 & +1 & +1 \\ +1 & +1 & +1 & +1 & +1 \\ +1 & -2 & -2 & +1 & +1 \\ +1 & +1 & +1 & -2 & -2 \end{bmatrix} \times \begin{bmatrix} W_{\{1,20\}} \\ W_{\{2,19\}} \\ W_{\{3,18\}} \\ W_{\{5,16\}} \\ W_{\{7,14\}} \end{bmatrix}. \tag{7.11}$$

Similarly, for all the multicast messages $W_{\mathcal{J}}$ where $\mathcal{J} \in \mathcal{G}_2$, we transmit

$$\begin{bmatrix} X_{1,2} \\ X_{2,2} \\ X_{3,2} \\ X_{4,2} \\ X_{5,2} \\ X_{6,2} \end{bmatrix} = \begin{bmatrix} -2 & -2 & +1 & +1 & +1 \\ -2 & +1 & +1 & -2 & +1 \\ +1 & -2 & -2 & +1 & +1 \\ +1 & +1 & +1 & +1 & +1 \\ +1 & +1 & +1 & -2 & -2 \\ +1 & +1 & -2 & +1 & -2 \end{bmatrix} \times \begin{bmatrix} W_{\{4,17\}} \\ W_{\{6,15\}} \\ W_{\{8,13\}} \\ W_{\{9,12\}} \\ W_{\{10,11\}} \end{bmatrix}. \tag{7.12}$$

Each user $k \in \mathcal{J}$ can recover $W_{\mathcal{J}}$ where $\mathcal{J} \in \mathcal{G}_u, u \in \{1, 1\}$, by summing the received codewords (corresponding to $\mathcal{G}_u$) from the relays in $\mathcal{H}_k$.

We can now compute the max link-load. The load to transmit $W_{\mathcal{J}}$ where $\mathcal{J} \in \mathcal{V}_1$ is $3/2$. The load for the multicast messages in $\mathcal{V}_2$ is $1/10$. Thus, the max link-load of this scheme is $3/2 + 1/10 = 8/5$, while that of CICS (ICICS) and in references [12], and [50] are 1.6111, $19/6 \approx 3.1667$, and $29/12 \approx 2.41667$, respectively. □

We now generalize the scheme to transmit the messages $W_{\mathcal{J}}$ where $\mathcal{J} \in \mathcal{V}_2$, described in the above examples to the general case of $t = 1$ and $\mathsf{H} \geq 2\mathsf{r}$. We use cMAN cache placement and let $W_{\mathcal{J}} = \underset{j\in\mathcal{J}}{\oplus} F_{d_j,\mathcal{J}\setminus\{j\}}$ for each $\mathcal{J} \subseteq [\mathsf{K}]$ where $|\mathcal{J}| = 2$. Notice that if $t = 1$, $\mathcal{V}_2$ contains all the sets of two users k and k' where $\mathcal{H}_k \cap \mathcal{H}_{k'} = \emptyset$.

Transmission of $\mathcal{V}_2$ for $\mathsf{H} = 2\mathsf{r}$ If $\mathsf{H} = 2\mathsf{r}$, we have $|\mathcal{V}_2| = \binom{2\mathsf{r}}{\mathsf{r}}/2 = \binom{2\mathsf{r}-1}{\mathsf{r}-1}$. We partition $\mathcal{V}_2$ into $\binom{2\mathsf{r}-1}{\mathsf{r}-1}/(2\mathsf{r}-1) = \binom{2\mathsf{r}-2}{\mathsf{r}-1} - \binom{2\mathsf{r}-2}{\mathsf{r}}$ groups, each containing $2\mathsf{r}-1$ elements (the group division is explained in Appendix A.7). In Appendix A.6, we prove $\binom{2\mathsf{r}-1}{\mathsf{r}-1}/(2\mathsf{r}-1)$ is an integer.

For each group $\mathcal{G}_g,\ g \in \left[\binom{2\mathsf{r}-1}{\mathsf{r}-1}/(2\mathsf{r}-1)\right]$, the vector $\mathbb{M}_g$ of dimension $(2\mathsf{r}-1)\times 1$ lists the multicast messages $W_{\mathcal{J}} : \mathcal{J} \in \mathcal{G}_g$. Let $m_{g,j}$ denote the j^{th} element of $\mathbb{M}_g$. We design a linear code, on some finite field of sufficiently large size, to transmit $\mathbb{M}_g$, whose coding matrix is denoted by $\mathbb{A}_g$. Let the element on i^{th} row and j^{th} column of $\mathbb{A}_g$ be denoted by $a_{g,i,j}$, the i^{th} row by $\mathbb{A}_{g,i}$; the message $\mathbb{A}_{g,i} \times \mathbb{M}_g$ is transmitted to relay $i \in [\mathsf{H}]$.

We construct the j^{th} column of $\mathbb{A}_g$, whose elements are the coefficients for the multicast messages $m_{g,j}$, as follows. Firstly, we let

$$\sum_{i\in[\mathsf{H}]} a_{g,i,j} = 0. \tag{7.13a}$$

Then, for each $\mathcal{J}_1$ where $\mathcal{J}_1 \in \mathcal{G}_g \setminus \{\mathcal{J}\}$, we let

$$\sum_{i\in\mathcal{H}_{k_1}} a_{g,i,j} = 0, \tag{7.13b}$$

where k_1 is the user in $\mathcal{J}_1$ connected to relay H. Assume that k_2 is the other user in $\mathcal{J}_1$. Since $\mathcal{H}_{k_1} \cap \mathcal{H}_{k_2} = \emptyset$ and $\mathcal{H}_{k_1} \cup \mathcal{H}_{k_2} = [\mathsf{H}]$, from (7.13a) and (7.13b) it can be seen that

$$\sum_{i\in\mathcal{H}_{k_2}} a_{g,i,j} = 0. \tag{7.13c}$$

Users k_1 and k_2 can eliminate the interference caused by $W_{\mathcal{J}}$ by summing their received codewords for $\mathcal{G}_g$ from their connected relays. Lastly, we let

$$\sum_{i\in\mathcal{H}_{k_3}} a_{g,i,j} = s : s \neq 0, \tag{7.13d}$$

where k_3 is the user in $\mathcal{J}$ who is connected to relay H. Assume k_4 is the other user in $\mathcal{J}$, since $\mathcal{H}_{k_3} \cap \mathcal{H}_{k_4} = \emptyset$ and $\mathcal{H}_{k_3} \cup \mathcal{H}_{k_4} = [\mathsf{H}]$, from (7.13a) and (7.9d), it can be seen that

$$\sum_{i\in\mathcal{H}_{k_4}} a_{g,i,j} = -s. \tag{7.13e}$$

Hence, if the users in $\mathcal{J}$ sums their received codewords for $\mathcal{G}_g$ from their connected relays, they can recover $W_{\mathcal{J}}$.

By collecting all the equations in (7.13), we can write the following system of equations

$$\mathbb{C}_g \times \begin{bmatrix} a_{g,1,j} \\ \vdots \\ a_{g,\mathsf{H}-1,j} \\ a_{g,\mathsf{H},j} \end{bmatrix} = \begin{bmatrix} 0 \\ \vdots \\ 0 \\ s \end{bmatrix}, \ j \in [2\mathsf{r}-1]. \tag{7.14}$$

In Appendix A.7, we show how the proposed group division results in a matrix $\mathcal{C}_g$ that is full-rank. With this, we design the j^{th} column of $\mathbb{A}_g$, and then the whole matrix of $\mathbb{A}_g$.

Finally, each user $k \in \mathcal{J}$ can recover $W_{\mathcal{J}}$ for $\mathcal{J} \in \mathcal{G}_g$ by summing the received codewords (corresponding to $\mathcal{G}_g$) from the relays in $\mathcal{H}_k$. Having recovered the multicast messages $W_{\mathcal{J}}$ where $k \in \mathcal{J}$, user k then decodes $F_{d_k,\mathcal{J}\setminus\{k\}}$.

It can be easily see tat the load to transmit the multicast messages in $\mathcal{V}$ is $\frac{\binom{2\mathsf{r}-1}{\mathsf{r}-1}}{(2\mathsf{r}-1)K}$.

Transmission of $\mathcal{V}_2$ for $\mathsf{H} > 2\mathsf{r}$ For each set of relays $\mathcal{B}$ where $|\mathcal{B}| = 2\mathsf{r}$, we find the users who are connected to r of the chosen relays, and denote them by $\mathcal{P}_{\mathcal{B}}$. We can see that $|\mathcal{P}_{\mathcal{B}}| = \binom{2\mathsf{r}}{\mathsf{r}}$. It can also be seen that $|\mathcal{T}_{\mathcal{B}}| = \binom{2\mathsf{r}}{\mathsf{r}}/2 = \binom{2\mathsf{r}-1}{\mathsf{r}-1}$ where $\mathcal{T}_{\mathcal{B}}$ is defined in (7.6). So we can use the same scheme for the case $\mathsf{H} = 2\mathsf{r}$ to transmit $\mathcal{T}_{\mathcal{B}}$ through the relays in $\mathcal{B}$ with load $\frac{\binom{2\mathsf{r}-1}{\mathsf{r}-1}}{(2\mathsf{r}-1)K}$ such that each user $k \in \mathcal{P}_{\mathcal{B}}$ can recover $W_{\mathcal{J}}$ where $\mathcal{J} \in \mathcal{T}_{\mathcal{B}}$ and $k \in \mathcal{J}$. Notice that since $\mathcal{V}_2$ contains all the sets of two users k and k' where $\mathcal{H}_k \cap \mathcal{H}_{k'} = \emptyset$, we have $\bigcup_{\mathcal{B}\subseteq[\mathsf{H}]:|\mathcal{B}|=2\mathsf{r}} \mathcal{T}_{\mathcal{B}} = \mathcal{V}_2$. Hence, after considering all the sets of relay $\mathcal{B}$ where $|\mathcal{B}| = 2\mathsf{r}$, each user k can recover all the multicast messages $W_{\mathcal{J}}$ where $k \in \mathcal{J}$ and $\mathcal{J} \in \mathcal{V}_2$, and then decodes $F_{d_k,\mathcal{J}\setminus\{k\}}$.

Performance The max link-load to transmit the multicast messages in $\mathcal{V}_2$ is $\frac{\binom{2\mathsf{r}-1}{\mathsf{r}-1}}{(2\mathsf{r}-1)\mathsf{K}}\binom{\mathsf{H}}{2\mathsf{r}}$. In Appendix A.7, we show that when $2\mathsf{r}-1 = p^v$ or pq, where p and q are as in the statement of Theorem 14, we can partition $\mathcal{V}_2$ in such a way that IES is doable. By summing the loads for the two classes of messages, the achieved max link-load of IES is given in Theorem 14.

7.2.3 Concatenated Inner Code delivery Scheme (CICS)

At a high level, the proposed Concatenated Inner Code delivery Scheme (CICS) works are follows. We directly transmit each MAN message to some relays in the first phase such that each $W_{\mathcal{J}}$ can be recovered by a subset of users in $\mathcal{J}$; these messages can be seen as side information for the second phase. In the second phase, we design linear combinations of the MAN messages such that the remaining users in $\mathcal{J}$ recover $W_{\mathcal{J}}$. We illustrate this idea by means of one example first.

Example 8. Consider the network in Fig. 1.4 with $\mathsf{N} = \mathsf{K} = 6$, $\mathsf{M} = t = 2$ and let $\mathbf{d} = (1:6)$. For each $\mathcal{J} \subseteq [6]$ where $|\mathcal{J}| = t+1 = 3$, the MAN multicast messages in (2.14) contain $\mathsf{B}/15$ bits, because each file was split into $\binom{\mathsf{K}}{t} = 15$ equal length parts. Let us now look at the two-phase delivery of MAN multicast messages.

First phase For each $\mathcal{J} \subseteq [6]$ of size $|\mathcal{J}| = 3$, we compute the set of relays each of which is connected to the largest number of users in $\mathcal{J}$, that is, $\mathcal{S}_{\mathcal{J}} := \arg\max_{h\in[\mathsf{H}]} |\mathcal{U}_h \cap \mathcal{J}|$. We then partition each $W_{\mathcal{J}}$ into $|\mathcal{S}_{\mathcal{J}}|$ equal-length parts, denoted as $W_{\mathcal{J}} = \{W^{|\mathcal{S}_{\mathcal{J}}|}_{\mathcal{J},h} : h \in \mathcal{S}_{\mathcal{J}}\}$, and transmit $W^{|\mathcal{S}_{\mathcal{J}}|}_{\mathcal{J},h}$ to relay $h \in \mathcal{S}_{\mathcal{J}}$.

For example, for $\mathcal{J} = \{1, 2, 3\}$, relay 1 is connected to three users in $\mathcal{J}$ (user 1, 2 and 3), relay 2 is connected to one user in $\mathcal{J}$ (user 1), relay 3 is connected to one user in $\mathcal{J}$ (user 2), and relay 4 is connected to one user in $\mathcal{J}$ (user 3). So we have $\mathcal{S}_{\{1,2,3\}} = \arg\max_{h\in[\mathsf{H}]} |\mathcal{U}_h \cap \mathcal{J}| = 1$. Therefore, we have $W_{\{1,2,3\}} = \{W^1_{\{1,2,3\},1}\}$.

As another example, for $\mathcal{J} = \{1, 2, 4\}$, relay 1 is connected to two users in $\mathcal{J}$ (user 1 and 2), relay 2 is connected to two users in $\mathcal{J}$ (user 1 and 4), relay 3 is connected to two users in $\mathcal{J}$ (user 2 and 4), and relay 4 is not connected to all users in $\mathcal{J}$. So we have $\mathcal{S}_{\mathcal{J}} = \arg\max_{h\in[\mathsf{H}]} |\mathcal{U}_h \cap \mathcal{J}| = 3$. Therefore, we have $W_{\{1,2,4\}} = \{W^3_{\{1,2,4\},1}, W^3_{\{1,2,4\},2}, W^3_{\{1,2,4\},3}\}$.

After considering all the sets $\mathcal{J}$, the server has sent to relay 1 (and similarly for all other relays)

$$W^1_{\{1,2,3\},1}, W^3_{\{1,2,4\},1}, W^2_{\{1,2,5\},1}, W^2_{\{1,2,6\},1}, W^2_{\{1,3,4\},1},$$
$$W^3_{\{1,3,5\},1}, W^2_{\{1,3,6\},1}, W^2_{\{2,3,4\},1}, W^2_{\{2,3,5\},1}, W^3_{\{2,3,6\},1},$$

for a total of $\mathsf{B}/15 + 6\mathsf{B}/30 + 3\mathsf{B}/45 = \mathsf{B}/3$ bits; these messages are then transmitted by relay 1 to the users in $\mathcal{U}_1$.

Second phase Let us first focus on $W_{\{1,2,3\}} = \{W^1_{\{1,2,3\},1}\}$. From the first phase, $W^1_{\{1,2,3\},1}$ can be recovered by users $1, 2, 3$ from the coded message transmitted by relay 1. Hence, $W^1_{\{1,2,3\},1}$ need not to be transmitted again.

For $W_{\{1,2,4\}} = \{W^3_{\{1,2,4\},1}, W^3_{\{1,2,4\},2}, W^3_{\{1,2,4\},3}\}$, from the first phase, $W^3_{\{1,2,4\},1}$ can be recovered by users $1, 2$ from the coded message transmitted by relay 1. Note that user 4 has not received $W^3_{\{1,2,4\},1}$ yet. In the second phase, we aim to transmit $W^3_{\{1,2,4\},1}$ to user 4. We divide $W^3_{\{1,2,4\},1}$ into $\mathsf{r} = 2$ non-overlapping and equal-length pieces, $W^3_{\{1,2,4\},1} = \{W^3_{\{1,2,4\},1,h} : h \in \mathcal{H}_4\}$. We then let user 4 recover $W^3_{\{1,2,4\},1,2}$ from relay 2, and $W^3_{\{1,2,4\},1,3}$ from relay 3. In order to do so, since user 1, who is also connected to relay 2, knows $W^3_{\{1,2,4\},1,2}$, we put $W^3_{\{1,2,4\},1,2}$ into $\mathcal{P}^2_{4,\{1\}}$ representing the set of bits needed to be recovered by user 4 (the first entry in the subscript) from relay 2 (the superscript) and already known by user 1 (the second entry in the subscript) who is also connected to relay 2. Similarly, we put $W^3_{\{1,2,4\},1,3}$ in $\mathcal{P}^3_{4,\{2\}}$. After considering all the pieces of the multicast messages $W_{\mathcal{J}}$, for $\mathcal{J} \subseteq [6]$ and $|\mathcal{J}| = 3$, in the second phase for relay 1 we have (similarly for all other relays)

$$\mathcal{P}^1_{1,\{2\}} = \{W^3_{\{1,2,4\},3,1}, W^2_{\{1,2,6\},3,1}, W^2_{\{1,4,6\},3,1}\}, \tag{7.15}$$
$$\mathcal{P}^1_{1,\{3\}} = \{W^3_{\{1,3,5\},4,1}, W^2_{\{1,3,6\},4,1}, W^2_{\{1,5,6\},4,1}\}, \tag{7.16}$$
$$\mathcal{P}^1_{2,\{1\}} = \{W^3_{\{1,2,4\},2,1}, W^2_{\{1,2,5\},2,1}, W^2_{\{2,4,5\},2,1}\}, \tag{7.17}$$
$$\mathcal{P}^1_{2,\{3\}} = \{W^2_{\{2,3,5\},4,1}, W^3_{\{2,3,6\},4,1}, W^2_{\{2,5,6\},4,1}\}, \tag{7.18}$$
$$\mathcal{P}^1_{3,\{1\}} = \{W^2_{\{1,3,4\},2,1}, W^3_{\{1,3,5\},2,1}, W^2_{\{3,4,5\},2,1}\}, \tag{7.19}$$
$$\mathcal{P}^1_{3,\{2\}} = \{W^2_{\{2,3,4\},3,1}, W^3_{\{2,3,6\},3,1}, W^2_{\{3,4,6\},3,1}\}; \tag{7.20}$$

each of such set $\mathcal{P}$ contains $\mathsf{B}/90 + 2\mathsf{B}/60 = 2\mathsf{B}/45$ bits. We finally transmit $\mathcal{P}^1_{1,\{2\}} \oplus \mathcal{P}^1_{2,\{1\}}$ to relay 1 such that user 1 knowing $\mathcal{P}^1_{2,\{1\}}$ recovers $\mathcal{P}^1_{1,\{2\}}$ and user 2 knowing $\mathcal{P}^1_{1,\{2\}}$ recovers $\mathcal{P}^1_{2,\{1\}}$. Similarly, the server transmits $\mathcal{P}^1_{1,\{3\}} \oplus \mathcal{P}^1_{3,\{1\}}$ and $\mathcal{P}^1_{2,\{3\}} \oplus \mathcal{P}^1_{3,\{2\}}$ to relay 1. The total number of transmitted bits from server to relay 1 in this phase is $2\mathsf{B}/15$. In the end, for each set $\mathcal{J} \subseteq \mathcal{U}_1$, relay 1 then forwards $\underset{k\in\mathcal{J}}{\oplus} \mathcal{P}^1_{k,\mathcal{J}\setminus\{k\}}$ to users $k \in \mathcal{U}_1$ if $k \in \mathcal{J}$.

In conclusion, the achieved max link-load is $1/3 + 2/45 = 7/15$, while the max link-loads of the schemes [12], and [50] are $2/3$ and $1/2$, respectively. □

First phase For each $W_{\mathcal{J}}$ in (2.14) where $\mathcal{J} \subseteq [\mathsf{K}]$ and $|\mathcal{J}| = t+1$, we find $\mathcal{S}_{\mathcal{J}} := \arg\max_{h\in[\mathsf{H}]} |\mathcal{U}_h \cap \mathcal{J}|$ (i.e., the set of relays each relay in which is connected to the largest number of users in $\mathcal{J}$). We then partition $W_{\mathcal{J}}$ into $|\mathcal{S}_{\mathcal{J}}|$ non-overlapping equal-length pieces and denote $W_{\mathcal{J}} = \{W^{|\mathcal{S}_{\mathcal{J}}|}_{\mathcal{J},h} : h \in \mathcal{S}_{\mathcal{J}}\}$. The server transmits $W^{|\mathcal{S}_{\mathcal{J}}|}_{\mathcal{J},h}$ to relay $h \in \mathcal{S}_{\mathcal{J}}$, and relay $h \in \mathcal{S}$ transmits $W^{|\mathcal{S}_{\mathcal{J}}|}_{\mathcal{J},h}$ to the users in $\mathcal{U}_h$.

Second phase For each $W_{\mathcal{J}}$ as in the first phase the users in $\mathcal{J} \cap \mathcal{U}_h, h \in \mathcal{S}_{\mathcal{J}}$, can recover $W^{|\mathcal{S}_{\mathcal{J}}|}_{\mathcal{J},h}$; thus the second phase aims to transmit $W^{|\mathcal{S}_{\mathcal{J}}|}_{\mathcal{J},h}$ to the users in $\mathcal{J}\setminus\mathcal{U}_h$. For each user $k \in \mathcal{J}\setminus\mathcal{U}_h$, $W^{|\mathcal{S}_{\mathcal{J}}|}_{\mathcal{J},h}$ is divided into r non-overlapping and equal-length pieces and denoted as $W^{|\mathcal{S}_{\mathcal{J}}|}_{\mathcal{J},h} = \{W^{|\mathcal{S}_{\mathcal{J}}|}_{\mathcal{J},h,h'} : h' \in \mathcal{H}_k\}$. We aim to let user $k \in \mathcal{J}\setminus\mathcal{U}_h$ recover $W^{|\mathcal{S}_{\mathcal{J}}|}_{\mathcal{J},h,h'}$ from relay $h' \in \mathcal{H}_k$. For relays h, h' and user k, where user k is connected to relay h' but not to relay h, we define

$$\mathcal{Q}^k_{h,h'} := \left\{j \in \mathcal{U}_h \cap \mathcal{U}_{h'} : \mathcal{H}_j \subseteq \mathcal{H}_k \cup \{h\}\right\} \tag{7.21}$$

and put $W^{|\mathcal{S}_{\mathcal{J}}|}_{\mathcal{J},h,h'}$ in $\mathcal{P}^{h'}_{k,\mathcal{Q}^k_{h,h'}}$ representing the set of bits known by the users in $\mathcal{Q}^k_{h,h'}$ and to be recovered by user k from relay h'. Note $|\mathcal{Q}^k_{h,h'}| = \mathsf{r}-1$, as explained in Remark 13. Finally, for each relay $h \in [\mathsf{H}]$ and each set $\mathcal{V} \subseteq \mathcal{U}_h$ where $|\mathcal{V}| = \mathsf{r}$, the server transmits $\oplus_{k\in\mathcal{V}}\mathcal{P}^h_{k,\mathcal{J}\setminus\{k\}}$ to relay h, which forwards it to the users in $\mathcal{V}$.

Remark 13. *The two partition steps for each $W_{\mathcal{J}}$ (e.g., $W^{|\mathcal{S}_{\mathcal{J}}|}_{\mathcal{J},h}$ and $W^{|\mathcal{S}_{\mathcal{J}}|}_{\mathcal{J},h,h'}$) ensure that the number of bits transmitted from the server to each relay is the same. So the achieved max link-load is proportional to $1/\mathsf{H}$.*

We put $W^{|\mathcal{S}_{\mathcal{J}}|}_{\mathcal{J},h,h'}$ in $\mathcal{P}^{h'}_{k,\mathcal{Q}^k_{h,h'}}$ where $\mathcal{Q}^k_{h,h'}$ is defined in (7.21); among the relays h, h' user k is only connected to relay h'. Since the users in $\mathcal{Q}^k_{h,h'}$ are connected to relays h and h' simultaneously and the connected relays of these users are in the set $\mathcal{H}_k \cup \{h\}$ including $\mathsf{r}+1$ relays, one has $|\mathcal{Q}^k_{h,h'}| = \binom{\mathsf{r}+1-2}{\mathsf{r}-2} = \mathsf{r}-1$. By the symmetry of combination networks, for each relay $a \in \mathcal{H}_k \setminus \{h'\}$, there must exist one set $\mathcal{J}'$ with $|\mathcal{J}'| = t$ where a is in the set $\arg\max_{b\in[\mathsf{H}]} |\mathcal{U}_b \cap \mathcal{J}'|$, which also includes $|\mathcal{S}_{\mathcal{J}}|$ elements, and the user (assumed to be k') connected to relays in $(\mathcal{H}_k \cup \{h\})\setminus\{a\}$ is also in $\mathcal{J}'$. Since user k' is connected to relays h and h', one has $k' \in \mathcal{Q}^k_{h,h'}$. In addition, user k' needs to recover $W^{|\mathcal{S}_{\mathcal{J}'}|}_{\mathcal{J}',a,h'}$ from relay h', whose length is equal to the length of $W^{|\mathcal{S}_{\mathcal{J}}|}_{\mathcal{J},h,h'}$. Notice that $W^{|\mathcal{S}_{\mathcal{J}'}|}_{\mathcal{J}',a}$ is directly transmitted to relay a, so $W^{|\mathcal{S}_{\mathcal{J}'}|}_{\mathcal{J}',a,h'}$ is known by the $\mathsf{r}-2$ users in $\mathcal{Q}^k_{h,h'}\setminus\{k'\}$ and by user k. So we can add $W^{|\mathcal{S}|}_{\mathcal{J},h,h'}$ and $W^{|\mathcal{S}_{\mathcal{J}'}|}_{\mathcal{J}',a,h'}$ such that user k and k' can recover their desired pieces. Similarly, there are $|\mathcal{H}_k \setminus \{h'\}| = \mathsf{r}-1$ relays as relay a. So $W^{|\mathcal{S}|}_{\mathcal{J},h,h'}$ can be added with the other $\mathsf{r}-1$ pieces with the same length (each of which is demanded by one user in $\mathcal{Q}^k_{h,h'}$) and then be transmitted to relay h'.

Performance For each $W_{\mathcal{J}}$ the server directly transmits $W_{\mathcal{J},h}^{|\mathcal{S}_{\mathcal{J}}|}$ to relay $h \in \mathcal{S}_{\mathcal{J}}$ in the first phase for a total of $|W_{\mathcal{J}}| = \mathsf{B}/\binom{\mathsf{K}}{t}$ bits. In the second phase, for relay $h \in \mathcal{S}_{\mathcal{J}}$, $|\mathcal{J} \setminus \mathcal{U}_h|$ users recover $W_{\mathcal{J},h}^{|\mathcal{S}_{\mathcal{J}}|}$. So the server transmits $W_{\mathcal{J},h,h'}^{|\mathcal{S}_{\mathcal{J}}|}$ to each user $k \in \mathcal{J} \setminus \mathcal{U}_h$ for $h' \in \mathcal{H}_k$) in one linear combination with other $\mathsf{r}-1$ pieces of the same length (equal to $\frac{\mathsf{B}}{\mathsf{r}\binom{\mathsf{K}}{t}|\mathcal{S}_{\mathcal{J}}|}$. Hence, the total link-load to transmit $W_{\mathcal{J}}$ is $\frac{1}{\binom{\mathsf{K}}{t}} + \frac{|\mathcal{S}_{\mathcal{J}}||\mathcal{J}\setminus\mathcal{U}_h|}{\mathsf{r}\binom{\mathsf{K}}{t}|\mathcal{S}_{\mathcal{J}}|} = \frac{1+|\mathcal{J}\setminus\mathcal{U}_h|/\mathsf{r}}{\binom{\mathsf{K}}{t}}$. By the symmetry of combination networks, the number of transmitted bits to each relay is the same as in Theorem 15.

7.2.4 Improved Concatenated Inner Code delivery Scheme (ICICS)

The main idea of Improved Concatenated Inner Code delivery Scheme (ICICS) is to leverage the multicasting opportunities which are ignored in CICS. We also examine one example to highlight the improvement given by ICICS compared to CICS. Recall that $\mathrm{RLC}(m, \mathcal{S})$ represents m random linear combinations of the equal-length packets indexed by $\mathcal{S}$; m random linear combinations of $|\mathcal{S}|$ packets are linearly independent with high probability if operations are done on a large enough finite field; the same can be obtained by using the parity-check matrix of an $(|\mathcal{S}|, |\mathcal{S}| - m)$ MDS (Maximum Distance Separable) code.

Example 9. Consider the network in Fig. 1.4 with $\mathsf{N} = \mathsf{K} = 6$, $\mathsf{M} = t = 3$ and let $\mathbf{d} = (1:6)$. For each $\mathcal{J} \subseteq [6]$ where $|\mathcal{J}| = t+1 = 4$, we have that the MAN multicast messages in (2.14) contain $\mathsf{B}/20$ bits. Let us know look at the two-phase delivery of CICS.

First phase of CICS for each $\mathcal{J} \subseteq [6]$ of size $|\mathcal{J}| = 4$, we compute the set of relays each of which is connected to the largest number of users in $\mathcal{J}$, $\mathcal{S}_{\mathcal{J}} := \arg\max_{h\in[\mathsf{H}]} |\mathcal{U}_h \cap \mathcal{J}|$. We then partition $W_{\mathcal{J}}$ into $|\mathcal{S}_{\mathcal{J}}|$ non-overlapping equal-length pieces, $W_{\mathcal{J}} = (W_{\mathcal{J},h}^{|\mathcal{S}_{\mathcal{J}}|} : h \in \mathcal{S}_{\mathcal{J}})$. We transmit $W_{\mathcal{J},h}^{|\mathcal{S}_{\mathcal{J}}|}$ to relay $h \in \mathcal{S}_{\mathcal{J}}$. After considering all the sets $\mathcal{J} \subseteq [\mathsf{K}]$ where $|\mathcal{J}| = 4$, we see that the server has sent to relay 1 (and similarly for all other relays) the following messages

$$W_{\{1,2,3,4\},1}^{1}, W_{\{1,2,3,5\},1}^{1}, W_{\{1,2,3,6\},1}^{1}, W_{\{1,2,5,6\},1}^{4}, W_{\{1,3,4,6\},1}^{4}, W_{\{2,3,4,5\},1}^{4},$$

for a total of $3\mathsf{B}/16$ bits; these messages are then transmitted by relay 1 to the users in $\mathcal{U}_1$.

Second phase of CICS Let us focus on $W_{\{1,2,5,6\},1}^{4}$, which is directly transmitted to relay 1 in the first phase and can be recovered by users $1, 2$. In the second phase, we aim to transmit $W_{\{1,2,5,6\},1}^{4}$ to users $5, 6$, while considering it as side information for users $1, 2$. For user 5, we partition $W_{\{1,2,5,6\},1}^{4}$ into $\mathsf{r} = 2$ equal-length parts and denote $W_{\{1,2,5,6\},1}^{4} = \{W_{\{1,2,5,6\},1,k}^{4} : k \in \mathcal{H}_5\}$. where $\mathcal{H}_5 = \{2, 4\}$; we then let user 5 recover the first part $W_{\{1,2,5,6\},1,2}^{4}$ from relay 2, and the second part $W_{\{1,2,5,6\},1,4}^{4}$ from relay 4. Since user 1, who is connected to relay 2, knows $W_{\{1,2,5,6\},1}^{4}$, we put $W_{\{1,2,5,6\},1,2}^{4}$ in $\mathcal{T}_{5,\{1\}}^{2}$. Since user 3 connected to relay 4 knows $W_{\{1,2,5,6\},1}^{4}$ we put $W_{\{1,2,5,6\},1,4}^{4}$ in $\mathcal{P}_{5,\{3\}}^{4}$.

We do the same procedure for user 6. We partition $W_{\{1,2,5,6\},1}^{4}$ into $\mathsf{r} = 2$ equal-length parts and denote $W_{\{1,2,5,6\},1}^{4} = \{W_{\{1,2,5,6\},1,k}^{4} : k \in \mathcal{H}_6\}$ and put $W_{\{1,2,5,6\},1,3}^{4}$ in $\mathcal{P}_{6,\{2\}}^{3}$, $W_{\{1,2,5,6\},1,4}^{4}$ in $\mathcal{P}_{6,\{3\}}^{4}$.

After considering all the pieces of the multicast messages $W_{\mathcal{J}}$, for relay 1 we have (but similarly for all other relays)

$$\mathcal{P}_{1,\{2\}}^{1} = \{W_{\{1,2,4,6\},1,1}^{1}, W_{\{1,2,5,6\},3,1}^{4}, W_{\{1,3,4,6\},3,1}^{4}\}, \tag{7.22}$$

$$\mathcal{P}^1_{1,\{3\}} = \{W^1_{\{1,3,5,6\},1,1}, W^4_{\{1,2,5,6\},4,1}, W^4_{\{1,3,4,6\},4,1}\}, \tag{7.23}$$
$$\mathcal{P}^1_{2,\{1\}} = \{W^1_{\{1,2,4,5\},1,1}, W^4_{\{1,2,5,6\},2,1}, W^4_{\{2,3,4,5\},2,1}\}, \tag{7.24}$$
$$\mathcal{P}^1_{2,\{3\}} = \{W^1_{\{2,3,5,6\},1,1}, W^4_{\{1,2,5,6\},4,1}, W^4_{\{2,3,4,5\},4,1}\}, \tag{7.25}$$
$$\mathcal{P}^1_{3,\{1\}} = \{W^1_{\{1,3,4,5\},1,1}, W^4_{\{1,3,4,6\},2,1}, W^4_{\{2,3,4,5\},2,1}\}, \tag{7.26}$$
$$\mathcal{P}^1_{3,\{2\}} = \{W^1_{\{2,3,4,6\},1,1}, W^4_{\{1,3,4,6\},3,1}, W^4_{\{2,3,4,5\},3,1}\}; \tag{7.27}$$

all such $\mathcal{P}$'s contains $3\mathsf{B}/80$ bits. Finally the server transmits to relay 1

$$\mathcal{P}^1_{1,\{2\}} \oplus \mathcal{P}^1_{2,\{1\}},\ \mathcal{P}^1_{1,\{3\}} \oplus \mathcal{P}^1_{3,\{1\}},\ \mathcal{P}^1_{2,\{3\}} \oplus \mathcal{P}^1_{3,\{2\}},$$

with totally $9\mathsf{B}/80$ bits.

In the end, for each set $\mathcal{J} \subseteq \mathcal{U}_h$, each relay $h \in [\mathsf{H}]$ transmits $\oplus_{j\in\mathcal{J}} \mathcal{P}_{d_j,\mathcal{J}\setminus\{j\}}$ to each user $j \in \mathcal{J}$. In conclusion, the max link-load of CICS is $\frac{15}{4\binom{6}{3}} + \frac{9}{4\binom{6}{3}} = 0.3$ while the max link-loads of the schemes in [12] and [50] are 0.375 and $1/3$, respectively.

Improved First phase of ICICS This step is the same as CICS with the exception that each coded messages $W_{\mathcal{J}}$ is divided into B/P packets for some large enough length P (possible since B can be taken arbitrary large).

Improved Second phase of ICICS It can be seen that CICS treats $W^4_{\{1,2,5,6\},1}$ demanded by user 1 and 2 as two independent pieces. It lets user 1 recover $|W^4_{\{1,2,5,6\},4}|/2$ bits from relay 1 and $|W^4_{\{1,2,5,6\},4}|/2$ bits from relay 2 in two linear combinations, and lets user 2 recover $|W^4_{\{1,2,5,6\},4}|/2$ bits from relay 1 and $|W^4_{\{1,2,5,6\},4}|/2$ bits from relay 3 in two other linear combinations.

Instead, we can leverage the following multicasting opportunity. We put $\mathrm{RLC}(|W^4_{\{1,2,5,6\},4}|/(2P), W^4_{\{1,2,5,6\},4})$ in $\mathcal{X}^1_{\{1,2\},\{3\}}$, where $\mathcal{X}^1_{\{1,2\},\{3\}}$ is the set of packets needed to be recovered by users in $\{1,2\}$ (first part of the subscript) from relay 1 (superscript) and already known by the users in $\{3\}$ (second part of the subscript) who are also connected to relay 1 (superscript). The number of packets in $\mathcal{X}^1_{\{1,2\},\{3\}}$ is $|\mathcal{X}^1_{\{1,2\},\{3\}}|/P$. We then encode the messages at relay 1 as

$$\mathcal{X}^1_{\{1\},\{2\}} \oplus \mathcal{X}^1_{\{2\},\{1\}},\ \mathcal{X}^1_{\{1\},\{3\}} \oplus \mathcal{X}^1_{\{3\},\{1\}},\ \mathcal{X}^1_{\{2\},\{3\}} \oplus \mathcal{X}^1_{\{3\},\{2\}},$$

where we used the same convention as before when it comes to 'summing' sets. We also send $\mathrm{RLC}(2|\mathcal{X}^1_{\{2,3\},\{1\}}|/P, \mathcal{X}^1_{\{1,2\},\{3\}} \cup \mathcal{X}^1_{\{1,3\},\{2\}} \cup \mathcal{X}^1_{\{2,3\},\{1\}})$ to relay 1. Note that the users in $\{1,2,3\}$ know $|\mathcal{X}^1_{\{2,3\},\{1\}}|/P$ packets of $\mathcal{X}^1_{\{1,2\},\{3\}} \cup \mathcal{X}^1_{\{1,3\},\{2\}} \cup \mathcal{X}^1_{\{2,3\},\{1\}}$. So if the server transmits $2|\mathcal{X}^1_{\{2,3\},\{1\}}|/P$ random linear combinations of those packets to relay 1, which will then forward them to its connected users, each user can recover all of the packets of $\mathcal{X}^1_{\{1,2\},\{3\}} \cup \mathcal{X}^1_{\{1,3\},\{2\}} \cup \mathcal{X}^1_{\{2,3\},\{1\}}$ with high probability provided that $\mathsf{B} \to \infty$.

The max link-load of ICICS is $\frac{15}{4\binom{K}{t}} + \frac{17}{8\binom{K}{t}}\Big|_{\mathsf{K}=6,t=3} = 0.29375$, which is less than the max link-load of CICS (equal to 0.3); for the same set of parameters, the max link-loads of the schemes in [12] and [50] are 0.375 and $1/3$, respectively. □

The main idea of ICICS is to leverage the multicasting opportunities that were ignored in CICS, as illustrated in the above example. The pseudo-code for this improved delivery can be found in Algorithm 3 in Appendix A.17.

First phase This step is the same as in Section 7.2.3 with the exception that each coded messages $W_{\mathcal{J}}$ is divided into B/P packets for some large enough length P

Second phase For each $W_{\mathcal{J}}$ where $\mathcal{J} \subseteq [\mathsf{K}]$ and $|\mathcal{J}| = t+1$, and each $h \in \mathcal{S}_{\mathcal{J}}$, the second phase is used to transmit $W_{\mathcal{J},h}^{|\mathcal{S}_{\mathcal{J}}|}$ to the users in $\mathcal{J} \setminus \mathcal{U}_h$. In this paragraph, to simplify the notation, we let $\mathcal{A} := \mathcal{U}_{h'} \cap (\mathcal{J} \setminus \mathcal{U}_h) \neq \emptyset$ and $\mathcal{B} := \left\{ j \in \mathcal{U}_h \cap \mathcal{U}_{h'} : \mathcal{H}_j \subseteq \mathcal{H}_{\mathcal{A}} \cup \{h\} \right\}$. For each $h' \in [\mathsf{H}] \setminus \{h\}$ where $\mathcal{A} \neq \emptyset$, we add $|W_{\mathcal{J},h}^{|\mathcal{S}_{\mathcal{J}}|}|/(\mathsf{r}P)$ random linear combinations of packets of $W_{\mathcal{J},h}^{|\mathcal{S}_{\mathcal{J}}|}$ in $\mathcal{X}_{\mathcal{A},\mathcal{B}}^{h'}$ representing the packets to be recovered by users in $\mathcal{A}$ from relay h' and already known by the users in $\mathcal{B}$ who are also connected to relay h'.

We aim to let each user $k \in \mathcal{J} \setminus \mathcal{U}_h$ recover all the sets of packets $\mathcal{X}_{\mathcal{W}_1,\mathcal{W}_2}^{h'} \neq \emptyset$ where $h' \in \mathcal{U}_k$ and $k \in \mathcal{W}_1$, such that he can recover $|W_{\mathcal{J},h}^{|\mathcal{S}_{\mathcal{J}}|}|/P$ random linear combinations of packets in $W_{\mathcal{J},h}^{|\mathcal{S}_{\mathcal{J}}|}$ and then recover $W_{\mathcal{J},h}^{|\mathcal{S}_{\mathcal{J}}|}$ with high probability provided that $\mathsf{B} \to \infty$. We use a two-stage coding.

Stage 1: For each relay $h \in [\mathsf{H}]$ and each set $\mathcal{V} \subseteq \mathcal{U}_h$, we encode all the sets of packets $\mathcal{X}_{\mathcal{W}_1,\mathcal{W}_2}^{h} \neq \emptyset$ where $\mathcal{W}_1 \cup \mathcal{W}_2 = \mathcal{V}$, by $\mathcal{L}_{\mathcal{V}}^h = \text{RLC}(c/P, \mathcal{C})$ where

$$\mathcal{C} = \bigcup_{\mathcal{W}_1,\mathcal{W}_2:\mathcal{W}_1\cup\mathcal{W}_2=\mathcal{V}} \mathcal{X}_{\mathcal{W}_1,\mathcal{W}_2}^{h}, \tag{7.28}$$

$$c = \max_{k\in\mathcal{V}} \sum_{\mathcal{W}_1,\mathcal{W}_2:\mathcal{W}_1\cup\mathcal{W}_2=\mathcal{V},k\notin\mathcal{W}_2} |\mathcal{X}_{\mathcal{W}_1,\mathcal{W}_2}^{h}|, \tag{7.29}$$

where c/P is the maximal number of packets in $\mathcal{C}$ not known by the users in $\mathcal{V}$. So from $\mathcal{L}_{\mathcal{V}}^h$, each user $k \in \mathcal{V}$ can recover all the sets $\mathcal{X}_{\mathcal{W}_1,\mathcal{W}_2}^{h} \neq \emptyset$ where $k \in \mathcal{W}_1$.

Stage 2: If for each set $\mathcal{X}_{\mathcal{W}_1,\mathcal{W}_2}^{h}$, where $\mathcal{W}_1 \cup \mathcal{W}_2 = \mathcal{V}$ and $|\mathcal{X}_{\mathcal{W}_1,\mathcal{W}_2}^{h}| > 0$, we have $k \in \mathcal{W}_2$, it can be seen that user k already knows $\mathcal{L}_{\mathcal{V}}^h$ from the first stage. Hence, in one relay $h \in [\mathsf{H}]$, we can encode $\mathcal{L}_{\mathcal{V}}^h$ for all $\mathcal{V} \subseteq \mathcal{U}_h$ where $\mathcal{L}_{\mathcal{V}}^h \neq \emptyset$ by $\text{RLC}(c'/P, \mathcal{C}')$ where

$$\mathcal{C}' = \bigcup_{\mathcal{V}\subseteq\mathcal{U}_h:\mathcal{L}_{\mathcal{V}}^h\neq\emptyset} \mathcal{L}_{\mathcal{V}}^h, \tag{7.30}$$

$$c' = \max_{k\in\mathcal{U}_h} \sum_{\mathcal{V}\subseteq\mathcal{U}_h:\mathcal{L}_{\mathcal{V}}^h \text{is unknown to } k} |\mathcal{L}_{\mathcal{V}}^h|. \tag{7.31}$$

We transmit $\text{RLC}(c'/P, \mathcal{C}')$ to relay h and relay h then forwards $\text{RLC}(c'/P, \mathcal{C}')$ to users in $\mathcal{U}_h$.

Remark 14. *We now analyse the actual file split level for our proposed schemes. In IES, we need not divide $W_{\mathcal{J}}$. In DIS, we divide each $W_{\mathcal{J}}$ into r non-overlapping equal-length pieces. In CICS, we first divide $W_{\mathcal{J}}$ into $|\mathcal{S}_{\mathcal{J}}|$ non-overlapping equal-length pieces and then divide each obtained piece into r non-overlapping equal-length pieces. Hence, $W_{\mathcal{J}}$ is divided into $|\mathcal{S}_{\mathcal{J}}|\mathsf{r}$ pieces. in ICICS, we first divide $W_{\mathcal{J}}$ into $|\mathcal{S}_{\mathcal{J}}|$ non-overlapping equal-length pieces and then divide each obtained piece into at most $\mathsf{H}-1$ non-overlapping equal-length pieces. Hence, $W_{\mathcal{J}}$ is divided into at most $|\mathcal{S}_{\mathcal{J}}|(\mathsf{H}-1)$ pieces.*

7.3 Novel Non-separation Based Delivery Scheme

With any uncoded cache placement, we introduce a novel delivery scheme by leveraging the symmetry in the topology of combination networks, referred to as Separate Relay Decoding delivery Scheme (SRDS). Each multicast message sent to relay $h \in [\mathsf{H}]$ is aimed to be useful for the largest possible subset of $\mathcal{H}_h$ (i.e, users connected to relay h). This delivery phase will also be used with our proposed asymmetric coded placement in the next section. In addition, this delivery scheme can also used with the decentralized placements in [5] and in [70], while the caching scheme in [50] cannot be extended to decentralized systems (because its placement is designed with the knowledge of the position (the connected relays) of each user in the delivery phase).

7.3.1 Example for $\mathsf{H}=4$, $\mathsf{r}=2$, $\mathsf{N}=\mathsf{K}=6$ and $\mathsf{M}=2$

Consider the network in Fig. 1.4 with $\mathsf{N}=\mathsf{K}=6$ and $\mathsf{M}=2$. We use MAN placement where each file F_i is partitioned into $\binom{\mathsf{K}}{t}=15$ non-overlapping sub-files of length $\mathsf{B}/15$ bits, where $t=\mathsf{KM}/\mathsf{N}$. Each subfile denoted by $F_{i,\mathcal{W}}$ where $\mathcal{W}\subseteq[\mathsf{K}]$ and $|\mathcal{W}|=t$, is cached by users in $\mathcal{W}$. In the delivery phase, assume that $\mathbf{d}=(1:6)$. There are mainly three steps.

Step 1 [Subfile partition] For each $F_{d_k,\mathcal{W}}, k\in[\mathsf{K}]$ where $k\notin\mathcal{W}$, we seek to find the set of relays in $\mathcal{H}_k$ each of which is connected to the largest number of users in $\mathcal{W}$. Consider the following examples.

For sub-file $F_{1,\{2,3\}}$, which is demanded by user $k=1$ and cached by the users in $\mathcal{W}=\{2,3\}$, we have that $\mathcal{U}_1=\{1,2,3\}$ (relay 1 is connected to two users in $\mathcal{W}$) and $\mathcal{U}_2=\{1,4,5\}$ (relay 2 is not connected to any user in $\mathcal{W}$). So we solve $\mathcal{S}_{k,\mathcal{W}}=\arg\max_{h\in\mathcal{H}_k}|\mathcal{U}_h\cap\mathcal{W}|=\{1\}$ (i.e., relay $h=1$). Since $|\mathcal{S}_{1,\{2,3\}}|=1$ we divide $F_{1,\{2,3\}}$ into $|\mathcal{S}_{1,\{2,3\}}|=1$ equal-length pieces and denote $F_{1,\{2,3\}}=(F^{|\mathcal{S}|}_{1,\{2,3\},h}: h\in\mathcal{S}_{1,\{2,3\}})$. We add $F^1_{1,\{2,3\},1}$ into $\mathcal{T}^h_{k,\mathcal{W}\cap\mathcal{U}_h}=\mathcal{T}^1_{1,\{2,3\}}$ representing the set of bits needed to be recovered by user $k=1$ from relay $h\in\mathcal{S}_{1,\{2,3\}}$ and already known by the users in $\mathcal{W}\cap\mathcal{U}_h=\{2,3\}$ who are also connected to relay $h=1$.

Consider now $F_{1,\{2,5\}}$ where $k=1$ and $\mathcal{W}=\{2,5\}$. Sub-file $F_{1,\{2,5\}}$ is also demanded by user 1, who is connected to relays $\mathcal{H}_1=\{1,2\}$. Relay 1 is connected to users $\mathcal{W}\cap\mathcal{U}_1=\{2,5\}\cap\{1,2,3\}=\{2\}$, while relay 2 is connected to users $\mathcal{W}\cap\mathcal{U}_2=\{2,5\}\cap\{1,4,5\}=\{5\}$, and thus we have $\mathcal{S}_{k,\mathcal{W}}=\arg\max_{h\in\mathcal{H}_k}|\mathcal{U}_h\cap\mathcal{W}|=\{1,2\}$. Since now $|\mathcal{S}_{1,\{2,5\}}|=2$, we divide $F_{1,\{2,5\}}$ into $|\mathcal{S}|=2$ equal-length pieces and denote $F_{1,\{2,5\}}=(F^{|\mathcal{S}|}_{1,\{2,5\},h}: h\in\mathcal{S}_{1,\{2,5\}})$. We add $F^2_{1,\{2,5\},1}$ (i.e., $(k,\mathcal{W},h)=(1,\{2,3\},1)$) to the set $\mathcal{T}^h_{k,\mathcal{W}\cap\mathcal{U}_h}=\mathcal{T}^1_{1,\{2\}}$, and $F^2_{1,\{2,5\},2}$ (i.e., $(k,\mathcal{W},h)=(1,\{2,3\},2)$) in the set $\mathcal{T}^h_{k,\mathcal{W}\cap\mathcal{U}_h}=\mathcal{T}^2_{1,\{5\}}$.

After considering all the sub-files demanded by all the users, for relay $h=1$ (and similarly for all other relays) we have

$$\mathcal{T}^1_{1,\{2,3\}}=\{F^1_{1,\{2,3\},1}\},\ |\mathcal{T}^1_{1,\{2,3\}}|=\mathsf{B}/15, \tag{7.32}$$

$$\mathcal{T}^1_{1,\{2\}}=\{F^2_{1,\{2,4\},1},F^2_{1,\{2,5\},1},F^1_{1,\{2,6\},1}\},\ |\mathcal{T}^1_{1,\{2\}}|=2\mathsf{B}/15, \tag{7.33}$$

$$\mathcal{T}^1_{1,\{3\}}=\{F^2_{1,\{3,4\},1},F^2_{1,\{3,5\},1},F^1_{1,\{3,6\},1}\},\ |\mathcal{T}^1_{1,\{3\}}|=2\mathsf{B}/15, \tag{7.34}$$

$$\mathcal{T}^1_{2,\{1,3\}}=\{F^1_{2,\{1,3\},1}\},\ |\mathcal{T}^1_{1,\{1,3\}}|=\mathsf{B}/15, \tag{7.35}$$

$$\mathcal{T}^1_{2,\{1\}}=\{F^2_{2,\{1,4\},1},F^1_{2,\{1,5\},1},F^2_{2,\{1,6\},1}\},\ |\mathcal{T}^1_{2,\{1\}}|=2\mathsf{B}/15, \tag{7.36}$$

$$\mathcal{T}^1_{2,\{3\}} = \{F^2_{2,\{3,4\},1}, F^1_{2,\{3,5\},1}, F^2_{2,\{3,6\},1}\},\ |\mathcal{T}^1_{2,\{3\}}| = 2\mathsf{B}/15, \tag{7.37}$$

$$\mathcal{T}^1_{3,\{1,2\}} = \{F^1_{3,\{1,2\},1}\},\ |\mathcal{T}^1_{1,\{1,2\}}| = \mathsf{B}/15, \tag{7.38}$$

$$\mathcal{T}^1_{3,\{1\}} = \{F^1_{3,\{1,4\},1}, F^2_{3,\{1,5\},1}, F^2_{3,\{1,6\},1}\},\ |\mathcal{T}^1_{3,\{1\}}| = 2\mathsf{B}/15, \tag{7.39}$$

$$\mathcal{T}^1_{3,\{2\}} = \{F^1_{3,\{2,4\},1}, F^2_{3,\{2,5\},1}, F^2_{3,\{2,6\},1}\},\ |\mathcal{T}^1_{3,\{2\}}| = 2\mathsf{B}/15. \tag{7.40}$$

Step 2 [Multicast Message Generation] In this example, for each relay $h \in [\mathsf{H}]$ and each $\mathcal{J} \subseteq \mathcal{U}_h$, we have $\min_{k\in\mathcal{J}} |\mathcal{T}^h_{k,\mathcal{J}\backslash\{k\}}| = \max_{k\in\mathcal{J}} |\mathcal{T}^h_{k,\mathcal{J}\backslash\{k\}}|$, i.e., only a function of $|\mathcal{J}|$. We thus create the multicast messages $W^h_{\mathcal{J}} := \underset{k\in\mathcal{J}}{\oplus} \mathcal{T}^h_{k,\mathcal{J}\backslash\{k\}}$. For example, the server transmits to relay 1 the following messages

$$W^1_{\{1,2,3\}} = \mathcal{T}^1_{1,\{2,3\}} \oplus \mathcal{T}^1_{2,\{1,3\}} \oplus \mathcal{T}^1_{3,\{1,2\}} \text{of length } \mathsf{B}/15 \text{ bits}, \tag{7.41}$$

$$W^1_{\{1,2\}} = \mathcal{T}^1_{1,\{2\}} \oplus \mathcal{T}^1_{2,\{1\}} \text{of length } 2\mathsf{B}/15 \text{ bits}, \tag{7.42}$$

$$W^1_{\{1,3\}} = \mathcal{T}^1_{1,\{3\}} \oplus \mathcal{T}^1_{3,\{1\}} \text{of length } 2\mathsf{B}/15 \text{ bits, and} \tag{7.43}$$

$$W^1_{\{2,3\}} = \mathcal{T}^1_{2,\{3\}} \oplus \mathcal{T}^1_{3,\{2\}} \text{of length } 2\mathsf{B}/15 \text{ bits}. \tag{7.44}$$

Step 3 [Multicast Message Delivery] Finally, for each relay $h \in [\mathsf{H}]$ and each set $\mathcal{J} \subseteq \mathcal{U}_h$ where $W^h_{\mathcal{J}} \neq \emptyset$, relay h forwards $W^h_{\mathcal{J}}$ to the users in $k \in \mathcal{J}$.

The normalized (by the file length) number of bits sent from the server to each relay is the same, thus the achieved max link-load is $7/15 = 14/30$. The max link-loads of the schemes in [12], [50] are $20/30$, and $15/30$, respectively.

7.3.2 General Scheme of SRDS

The key in above example is that for each relay $h \in [\mathsf{H}]$ and each set $\mathcal{J} \subseteq \mathcal{U}_h$, the length of the message $\mathcal{T}^h_{k,\mathcal{J}\backslash\{k\}}$ only depends on $|\mathcal{J}|$. However, in other cases (e.g., $\mathsf{r} > 2$ and MAN placement is used), we may have $\min_{k\in\mathcal{J}} |\mathcal{T}^h_{k,\mathcal{J}\backslash\{k\}}| < \max_{k\in\mathcal{J}} |\mathcal{T}^h_{k,\mathcal{J}\backslash\{k\}}|$. In order to 'equalize' the lengths of the various parts involved in the linear combinations for the multicast messages, we propose to use the bit-borrowing idea described in Section 5.4. The pseudo code of the proposed SRDS delivery scheme is in Algorithm 4 in Appendix A.17.

For any uncoded cache placement, each file F_i where $i \in [\mathsf{N}]$ can be divided into non-overlapping subfiles and $F_{i,\mathcal{W}}$ represents the bits known by users in $\mathcal{W}$.

Step 1 [Subfile partition] For each user $k \in [\mathsf{K}]$ and each set $\mathcal{W} \subseteq [\mathsf{K}] \setminus \{k\}$ where $|F_{d_k,\mathcal{W}}| > 0$, we search for the set of relays $\mathcal{S}_{k,\mathcal{W}} \subseteq \mathcal{H}_k$, each relay in which is connected to the largest number of users in $\mathcal{W}$, i.e., $\max_{h\in\mathcal{H}_k} |\mathcal{U}_h \cap \mathcal{W}|$ users. We partition $F_{d_k,\mathcal{W}}$ into $|\mathcal{S}_{k,\mathcal{W}}|$ equal-length pieces $F_{d_k,\mathcal{W}} = (F^{|\mathcal{S}_{k,\mathcal{W}}|}_{d_k,\mathcal{W},h} : h \in \mathcal{S}_{k,\mathcal{W}})$. For each relay $h \in \mathcal{S}_{k,\mathcal{W}}$, we add $F^{|\mathcal{S}_{k,\mathcal{W}}|}_{d_k,\mathcal{W},h}$ to $\mathcal{T}^h_{k,\mathcal{W}\cap\mathcal{U}_h}$, where $\mathcal{T}^h_{k,\mathcal{W}\cap\mathcal{U}_h}$ represents the set of bits needed to be recovered by user k from relay h and already known by the users in $\mathcal{W} \cap \mathcal{U}_h$ who are also connected to relay h.

Step 2 [Multicast Message Generation] Focus on each relay $h \in [\mathsf{H}]$ and each set $\mathcal{J} \subseteq \mathcal{U}_h$ where $W^h_{\mathcal{J}} \neq \emptyset$. For each user $k \in \mathcal{J}$, if $|\mathcal{T}^h_{k,\mathcal{J}\backslash\{k\}}| < \max_{k_1\in\mathcal{J}} |\mathcal{T}^h_{k_1,\mathcal{J}\backslash\{k_1\}}|$, we use the bit-borrowing idea

described in Section 5.4: we take bits from $\mathcal{T}^h_{k,\mathcal{W}}$ where $\mathcal{J}\setminus\{k\}\subset\mathcal{W}$ and $k\notin\mathcal{W}$ and add them to $\mathcal{T}^h_{k,\mathcal{J}\setminus\{k\}}$ so that these borrowed bits need not to be transmitted to k later. Since $\mathcal{J}\setminus\{k\}\subset\mathcal{W}$, the users in $\mathcal{J}\setminus\{k\}$ also knows $\mathcal{T}^h_{k,\mathcal{W}}$. After considering all the users in $\mathcal{J}$, the server forms the multicast messages

$$W^h_{\mathcal{J}} := \underset{k\in\mathcal{J}}{\oplus}\mathcal{T}^h_{k,\mathcal{J}\setminus\{k\}}. \tag{7.45}$$

Step 3 [Multicast Message Delivery] For each relay $h\in[\mathsf{H}]$ and each set $\mathcal{J}\subseteq\mathcal{U}_h$ where $W^h_{\mathcal{J}}\neq\emptyset$, the server sends $W^h_{\mathcal{J}}$ to relay h, who then forwards it to each user $k\in\mathcal{J}$.

Performance In the following we show that, when $\mathsf{r}=2$ and MAN placement is used, the bit-borrowing step is not needed (as example in Section 7.3.1). In this case the achieved max link-load is given in the following theorem.

Theorem 16. *For a* $(\mathsf{H},\mathsf{r}=2,\mathsf{M},\mathsf{N})$ *combination network with* $t=\mathsf{KM}/\mathsf{N}\in[0:\mathsf{K}]$*, the max link-load is*

$$\mathsf{R}^{\star}_{\mathrm{c,u}}\le\mathsf{R}_{\mathrm{SRDS}}:=\frac{\mathsf{K}X_{\mathsf{K},\mathsf{H}}}{\mathsf{H}\binom{\mathsf{K}}{t}}, \tag{7.46a}$$

$$X_{\mathsf{K},\mathsf{H}}:=\sum_{b_1=0}^{\min\{t,\mathsf{H}-2\}}\frac{1}{b_1+1}\binom{\mathsf{H}-2}{b_1}^2\binom{\binom{\mathsf{H}-2}{2}}{t-2b_1}+\sum_{b_1=0}^{\min\{t,\mathsf{H}-2\}}\sum_{b_2=0}^{b_1-1}\frac{2}{b_1+1}\binom{\mathsf{H}-2}{b_1}\binom{\mathsf{H}-2}{b_2}\binom{\binom{\mathsf{H}-2}{2}}{t-b_1-b_2}, \tag{7.46b}$$

where $\mathsf{R}_{\mathrm{SRDS}}$ *achieved by SRDS with MAN placement. The tradeoff between memory size and max link-load is the lower convex envelope of the above points.*

Proof. When $\mathsf{r}=2$, each user k is connected to 2 relays, which are assumed to be h and h'. It can be seen that $\mathcal{U}_h\cap\mathcal{U}_{h'}=\{k\}$. For one relay $h\in[\mathsf{H}]$ and each set $\mathcal{J}\subseteq\mathcal{U}_h$, we can compute the length of $\mathcal{T}^h_{k,\mathcal{J}\setminus\{k\}}$ for each user $k\in\mathcal{J}$. Assume that the other relay connected to k is h'. The number of $\mathcal{W}$ where $|\mathcal{W}|=t$ and $k\notin\mathcal{W}$ such that $|\mathcal{U}_{h'}\cap\mathcal{W}|=|\mathcal{U}_h\cap\mathcal{W}|=|\mathcal{J}|-1$, is $\binom{\mathsf{Kr}/\mathsf{H}-1}{|\mathcal{J}|-1}\binom{\binom{\mathsf{H}-\mathsf{r}}{\mathsf{r}}}{t-2(|\mathcal{J}|-1)}$, where $\mathsf{Kr}/\mathsf{H}-1$ is the number of users connected to relay h' besides k and $\binom{\mathsf{H}-\mathsf{r}}{\mathsf{r}}$ is the number of users which are not connected to relay h nor h'. For each of this kind $\mathcal{W}$, we divide $F_{d_k,\mathcal{W}}$ into 2 non-overlapping equal-length parts and put one part in $\mathcal{T}^h_{k,\mathcal{J}\setminus\{k\}}$. Similarly, The number of $\mathcal{W}$ where $|\mathcal{W}|=t$ and $k\notin\mathcal{W}$ such that $|\mathcal{U}_{h'}\cap\mathcal{W}|<|\mathcal{U}_h\cap\mathcal{W}|=|\mathcal{J}|-1$, is $\sum_{m=0}^{|\mathcal{J}|-2}\binom{\mathsf{Kr}/\mathsf{H}-1}{m}\binom{\binom{\mathsf{H}-\mathsf{r}}{\mathsf{r}}}{t-(|\mathcal{J}|-1)-m}$. For this kind of $\mathcal{W}$, we put $F_{d_k,\mathcal{W}}$ in $\mathcal{T}^h_{k,\mathcal{J}\setminus\{k\}}$. Hence, we can find that for each user $k\in\mathcal{J}$, $|\mathcal{T}^h_{k,\mathcal{J}\setminus\{k\}}|$ is identical and we need not to use the borrowing bits procedure. So we encode each $F_{d_k,\mathcal{W}}$ by a sum including $\max_{h\in[\mathsf{H}]}|\mathcal{U}_h\cap\mathcal{W}|$ sub-files. It is equivalent to say the total link-load to transmit $F_{d_k,\mathcal{W}}$ is $|F_{d_k,\mathcal{W}}|/\max_{h\in[\mathsf{H}]}\mathsf{B}|\mathcal{U}_h\cap\mathcal{W}|$. By considering each integer $b_1\in[0:\min\{t,\mathsf{Kr}/\mathsf{H}-1\}]$ representing the number of sub-files in one sum, we can compute the max link-load achieved by SRDS shown in (7.46a). □

7.4 Novel Non-separation Based Asymmetric Coded Placement

In Section 7.3, we propose a delivery phase by generating multicasting messages on network topology. However, with symmetric MAN placement where each $t-$subfile exists, the SRDS delivery phase is not

symmetric; the subfiles in one multicasting messages may have different length. In this Chapter, we introduce an asymmetric coded cache placement by leveraging the relay connectivity. The proposed asymmetric coded placement can lead the symmetry of SRDS in the delivery phase.

We fix one coded caching gain $g \in \left[1 : \binom{\mathsf{H}-1}{\mathsf{r}-1}\right]$ compared to uncoded routing scheme and want to minimize the needed memory size to achieve this coded caching gain. The achieved max link-load of the proposed scheme is stated as follows.

Theorem 17. *For an* $(\mathsf{H}, \mathsf{r}, \mathsf{M}, \mathsf{N})$ *combination network, the lower convex envelop of the following points*

$$(\mathsf{M}_b, \mathsf{R}_b) = \left(\frac{\mathsf{k}_{1,b}\mathsf{N}}{\mathsf{k}_{1,b}+\mathsf{k}_{2,b}}, \frac{\mathsf{K}(1-\mathsf{M}_b/\mathsf{N})}{\mathsf{H}g}\right) \tag{7.47a}$$

for (coded caching gain) $g \in [1 : \mathsf{K}_1]$ *and the point* $(\mathsf{M}_b, \mathsf{R}_b) = (\mathsf{N}, 0)$ *is achievable, where*

$$\mathsf{k}_{1,b} := \begin{cases} \sum_{a=1}^{\mathsf{r}} \binom{\mathsf{r}}{a}\binom{\mathsf{K}_a-1}{g-2}(-1)^{a-1}, \text{ if } g \le \mathsf{K}_2+1, \\ \sum_{m=1}^{\mathsf{H}-\mathsf{r}+1} N_m \frac{\mathsf{K}-\binom{\mathsf{H}-m}{\mathsf{r}}-m\mathsf{K}_1+m(g-1)}{\mathsf{K}}, \text{ if } g > \mathsf{K}_2+1, \end{cases} \tag{7.47b}$$

$$\mathsf{k}_{2,b} := g|\mathcal{Z}_g|/\mathsf{K} = \sum_{a=1}^{\mathsf{r}} \binom{\mathsf{r}}{a}\binom{\mathsf{K}_a-1}{g-1}(-1)^{a-1},, \tag{7.47c}$$

$$N_m = \binom{\mathsf{H}}{m}\left(\binom{\binom{\mathsf{H}-m}{\mathsf{r}-1}}{y_{g-1,m}} - \sum_{a=1}^{\mathsf{H}-m}\binom{\mathsf{H}-m}{a}\binom{\binom{\mathsf{H}-m}{\mathsf{r}-1} - \sum_{b=1}^{a}\binom{a}{b}\binom{\mathsf{H}-m-b}{\mathsf{r}-b-1}(-1)^{b-1}}{y_{g-1,m} - \sum_{b=1}^{a}\binom{a}{b}\binom{\mathsf{H}-m-b}{\mathsf{r}-b-1}(-1)^{b-1}}(-1)^{a-1}\right)$$
$$\text{for } m \in [2 : \mathsf{H}-r+1], \tag{7.47d}$$

$$N_1 := |\mathcal{Z}_{g-1}| - \sum_{m=2}^{\mathsf{H}-\mathsf{r}+1} mN_m, \tag{7.47e}$$

$$y_{t,m} := t - \mathsf{K}_1 + \binom{\mathsf{H}-m}{\mathsf{r}-1}. \tag{7.47f}$$

7.4.1 Proposed Asymmetric Coded Placement for $g \in [2 : \mathsf{K}_2+1]$

We aim to achieve coded caching gain $g \in [2 : \mathsf{K}_2+1]$. In other words, every multicast coded message send through the network is simultaneously useful for g users and each subfile is cached by at least $g-1$ other users.

Placement We consider the elements of $\mathcal{Z}_{g-1}$ defined in (5.8), that is, those subsets of users with cardinality $g-1$ (from a ground set of cardinality K_1) for which there exists at least one relay connected to all of them. We aim to partition each *coded file* (the parameters of the MDS code will be specified later) into

$$\mathsf{n_b} := |\mathcal{Z}_{g-1}| \tag{7.48}$$

equal-length subfiles, i.e.,

$$f_i = (f_{i,\mathcal{W}} : \mathcal{W} \in \mathcal{Z}_{g-1}),\ i \in [\mathsf{N}], \tag{7.49}$$

where subfile $f_{i,\mathcal{W}}$ is cached by the users in $\mathcal{W}$. Therefore, each user caches

$$\mathsf{k}_{1,\mathsf{b}} = \frac{g-1}{\mathsf{K}}|\mathcal{Z}_{g-1}| \text{ [subfiles per file]}, \tag{7.50}$$

since each subfile is cached by $g-1$ users and all users cache the same amount of subfiles. This placement is *asymmetric* because not all subfiles $f_{i,\mathcal{W}}$ for $\mathcal{W} \subseteq [\mathsf{K}] : |\mathcal{W}| = g-1$ are present.

Delivery We should create a multicast coded message similarly to (5.18) for each subset of users $\mathcal{J}$ of the form

$$\mathcal{J} = \mathcal{W} \cup \{k\} : \mathcal{W} \in \mathcal{Z}_{g-1},\ k \in [\mathsf{K}],\ k \notin \mathcal{W}; \tag{7.51}$$

however, only those $\mathcal{J} \in \mathcal{Z}_g$ are such that all users in $\mathcal{J}$ have at least one common connected relay; in order to have a symmetric delivery scheme from the relays to the users, we aim to deliver only those multicast coded messages for $\mathcal{J} \in \mathcal{Z}_g$ and consider those for $\mathcal{J} \notin \mathcal{Z}_g$ as "erased". Therefore, each user eventually receives

$$\mathsf{k}_{2,\mathsf{b}} = \frac{g|\mathcal{Z}_g|}{\mathsf{K}} \text{ [subfiles per file]}, \tag{7.52}$$

More precisely, for each set $\mathcal{J} \in \mathcal{Z}_g$, we generate the MAN-like multicast message

$$W_{\mathcal{J},\mathsf{b}} := \underset{k\in\mathcal{J}}{\oplus} f_{d_k,\mathcal{J}\setminus\{k\}} \tag{7.53}$$

and divide $W_{\mathcal{J},\mathsf{b}}$ into $|\mathcal{R}_{\mathcal{J}}|$ non-overlapping and equal-length pieces, $W_{\mathcal{J},\mathsf{b}} = \{W^h_{\mathcal{J},\mathsf{b}} : h \in \mathcal{R}_{\mathcal{J}}\}$. For each relay $h \in \mathcal{R}_{\mathcal{J}}$, the server transmits $W^h_{\mathcal{J},\mathsf{b}}$ to relay h which then forwards it to users in $\mathcal{J}$. Each user must be able to recover its required file by recovering all the $\mathsf{k}_{1,\mathsf{b}} + \mathsf{k}_{2,\mathsf{b}}$ subfiles that were either cached or received of its desired file; this is possible if we divide each file into $\mathsf{k}_{1,\mathsf{b}} + \mathsf{k}_{2,\mathsf{b}}$ non-overlapping and equal-length pieces and use $(\mathsf{n}_{\mathsf{b}}, \mathsf{k}_{1,\mathsf{b}} + \mathsf{k}_{2,\mathsf{b}})$ MDS code to generate the subfiles before placement, where $\mathsf{k}_{1,\mathsf{b}}$, $\mathsf{k}_{2,\mathsf{b}}$ and n_{b} are given in (7.50), (7.52) and (7.48), respectively.

Performance By construction each multicast coded message transmitted from the server is simultaneously useful for g users, the max link-load from the server to relays is $\mathsf{K}(1-\mathsf{M}_{\mathsf{b}}/\mathsf{N})/(\mathsf{H}g)$. In addition, since $|f_{d_{k_1},\mathcal{J}\setminus\{k_1\}}| = |f_{d_{k_2},\mathcal{J}\setminus\{k_2\}}|$ for each set $\mathcal{J} \in \mathcal{Z}_g$ and any $k_1, k_2 \in \mathcal{J}$, and each user recovers $\mathsf{B}(1-\mathsf{M}_{\mathsf{b}}/\mathsf{N})/\mathsf{r}$ bits from r relays, the load on each link from relays to users is $(1-\mathsf{M}_{\mathsf{b}}/Nsf)/\mathsf{r}$. With $\mathsf{K}/(\mathsf{H}g) \geq \mathsf{r}$, the max link-load is $\mathsf{K}(1-\mathsf{M}_{\mathsf{b}}/\mathsf{N})/(\mathsf{H}g)$. Hence, a coded caching gain of g is achieved with cache size per file

$$\frac{\mathsf{M}_{\mathsf{b}}}{\mathsf{N}} = \frac{\mathsf{k}_1}{\mathsf{k}_1+\mathsf{k}_2} = \frac{\sum_{a=1}^{\mathsf{r}} \binom{\mathsf{r}}{a}\binom{\mathsf{K}_a-1}{g-2}(-1)^{a-1}}{\sum_{a=1}^{\mathsf{r}} \binom{\mathsf{r}}{a}\binom{\mathsf{K}_a}{g-1}(-1)^{a-1}}, \tag{7.54}$$

coinciding with the needed memory size in Theorem 17.

7.4.2 Proposed Asymmetric Coded Placement for $g \in [\mathsf{K}_2 + 2 : \mathsf{K}_1]$

Since the number of users simultaneously connected to two relays is K_2, when $g - 1 > \mathsf{K}_2$, we have $|\mathcal{R}_{\mathcal{W}}| = 1$ for each set $\mathcal{W} \in \mathcal{Z}_{g-1}$. hence, the proposed scheme in Section 7.4.1 does not leverage the relay connectivity–which in this case is equivalent to the scheme in [50]. In this part, for $g \in [\mathsf{K}_2 + 2 : \mathsf{K}_1]$ we further create multicasting opportunities across the relays which are not available in the scheme in Section 7.4.1. We start by two examples to highlight the key idea. In the first example, we present one way to create multicasting opportunities between two relays.

Example 10 ($\mathsf{H} = 6$, $\mathsf{r} = 3$, $\mathsf{N} = 20$, $g = 6$)**.** In this example, we have $\mathsf{N} = \mathsf{K} = 20$, $\mathsf{K}_1 = 10$ and

$$\begin{aligned}
\mathcal{U}_1 &= \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10\},\\
\mathcal{U}_2 &= \{1, 2, 3, 4, 11, 12, 13, 14, 15, 16\},\\
\mathcal{U}_3 &= \{1, 5, 6, 7, 11, 12, 13, 17, 18, 19\},\\
\mathcal{U}_4 &= \{2, 5, 8, 9, 11, 14, 15, 17, 18, 20\},\\
\mathcal{U}_5 &= \{3, 6, 8, 10, 12, 14, 16, 17, 19, 20\},\\
\mathcal{U}_6 &= \{4, 7, 9, 10, 13, 15, 16, 18, 19, 20\}.
\end{aligned}$$

High level description of the proposed scheme In this example, we can see $g = 6 > \mathsf{K}_2 + 1 = 5$ and so there is only one relay simultaneously connected to the users in $\mathcal{W} \in \mathcal{Z}_{g-1}$ (i.e., $|\mathcal{R}_{\mathcal{W}}| = 1$). For each $\mathcal{W} \in \mathcal{Z}_{g-1}$ and each file $i \in [\mathsf{N}]$, we aim to create one MDS subfile $f_{i,\mathcal{W}'}$ such that $\mathcal{W}' \supseteq \mathcal{W}$. We define the collection of the common users of each two relays as

$$\mathsf{S} := \{\mathcal{U}_{\mathcal{J}} : \mathcal{J} \subseteq [\mathsf{H}], |\mathcal{J}| = 2\}. \tag{7.55}$$

Let us focus on the set of two relays, e.g., $\{1, 2\}$, whose set of common connected users is $\mathcal{U}_{\{1,2\}} = \{1, 2, 3, 4\}$. We choose $(g - 1) - |\mathcal{U}_{\{1,2\}}| = 1$ element in $\mathcal{U}_1 \setminus \mathcal{U}_{\{1,2\}}$, say 5, and in $\mathcal{U}_2 \setminus \mathcal{U}_{\{1,2\}}$, say 11; we let $\mathcal{W}' = \mathcal{U}_{\{1,2\}} \cup \{5\} \cup \{11\}$, which has cardinality 6, and generate one MDS subfile $f_{i,\mathcal{W}'}$ to be cached by the users in $\mathcal{W}'$ for each file $i \in [\mathsf{N}]$. By construction, we can see that $\mathcal{W}' \cap \mathcal{U}_1 = \{1, 2, 3, 4, 5\} \in \mathcal{Z}_{g-1}$ and $\mathcal{W}' \cap \mathcal{U}_2 = \{1, 2, 3, 4, 11\} \in \mathcal{Z}_{g-1}$. Hence, in the delivery phase, each user $k \in \mathcal{U}_1 \setminus \{1, 2, 3, 4, 5\} = [6 : 10]$ will recover $f_{d_k,\mathcal{W}'}$ from relay 1 and each user $k \in \mathcal{U}_2 \setminus \{1, 2, 3, 4, 11\} = [12 : 16]$ will recover $f_{d_k,\mathcal{W}'}$ from relay 2. Similarly, each time we choose one set

$$\mathcal{C}' \in \mathcal{C}_{\{1,2\},1} := \big\{\mathcal{C} \subseteq \mathcal{U}_1 \setminus \mathcal{U}_{\{1,2\}} : |\mathcal{C}| = (g - 1) - |\mathcal{U}_{\{1,2\}}| = 1\big\} \tag{7.56}$$

and one set

$$\mathcal{C}'' \in \mathcal{C}_{\{1,2\},2} := \big\{\mathcal{C} \subseteq \mathcal{U}_2 \setminus \mathcal{U}_{\{1,2\}} : |\mathcal{C}| = (g - 1) - |\mathcal{U}_{\{1,2\}}| = 1\big\} \tag{7.57}$$

and create the MDS subfile $f_{i,\mathcal{U}_{\{1,2\}} \cup \mathcal{C}' \cup \mathcal{C}''}$ cached by $\mathcal{U}_{\{1,2\}} \cup \mathcal{C}' \cup \mathcal{C}''$ for each file $i \in [\mathsf{N}]$. Notice that each element in $\mathcal{C}_{\{1,2\},1}$ and $\mathcal{C}_{\{1,2\},2}$ is only chosen one time. We then repeat the construction for all subsets of two relays. We define $\mathcal{H}_2$ as the collection of the sets $\mathcal{W} \in \mathcal{Z}_{g-1}$ where $|\{\mathcal{V} \in \mathsf{S} : \mathcal{V} \subseteq \mathcal{W}\}| = 2 - 1$, i.e., there exists one pair of relays $\{h_1, h_2\}$ such that $\mathcal{W} \supseteq \mathcal{U}_{\{h_1,h_2\}}$. Hence, considering all the set of two

relays, we create the MDS subfiles for each set $\mathcal{W} \in \mathcal{H}_2$. Since we create one MDS subfile corresponding to two sets in $\mathcal{H}_2$ for each file $i \in [\mathsf{N}]$ (we say such MDS subfiles are in Hierarchy 2), the number of created MDS subfiles per file in Hierarchy 2 is denoted by $N_2 := |\mathcal{H}_2|/2 = 90$.

The collection of remaining sets which should be considered is defined as $\mathcal{H}_1 = \mathcal{Z}_{g-1} \setminus \mathcal{H}_2$, i.e., $\mathcal{H}_1$ is the collection of the sets $\mathcal{W} \in \mathcal{Z}_{g-1}$ where $|\{\mathcal{V} \in \mathsf{S} : \mathcal{V} \subseteq \mathcal{W}\}| = 1 - 1 = 0$. For each set $\mathcal{W} \in \mathcal{H}_1$ and each file $i \in [\mathsf{N}]$, we directly create one MDS subfile $f_{i,\mathcal{W}}$ cached by users in $\mathcal{W}$ for each file $i \in [\mathsf{N}]$. We say such created MDS subfiles are in Hierarchy 1. The number of created MDS subfiles per file in Hierarchy 1 is denoted by $N_1 := |\mathcal{H}_1| = 1332$.

Hence, we totally create $\mathsf{n_b} = N_1 + N_2 = 1422$ MDS subfiles per file. Each MDS subfile in Hierarchy 1 and Hierarchy 2 is known by 5 and 6 users, respectively. Hence, each user caches $\mathsf{k}_{1,\mathsf{b}} = 5N_1/\mathsf{K} + 6N_2/\mathsf{K} = 360$ MDS subfiles per file.

Notice that for each generated MDS subfile in Hierarchy 2 (assumed to be $f_{i,\mathcal{W}'}$ from relay h_1 and h_2), since $\mathcal{W}' \supseteq \mathcal{U}_{\{h_1,h_2\}}$, we have $(\mathcal{U}_{h_1} \setminus \mathcal{W}') \cap (\mathcal{U}_{h_2} \setminus \mathcal{W}') = \emptyset$. Hence, if we focus on one user k, for any sets $\mathcal{J}_1, \mathcal{J}_2 \in \mathcal{Z}_g$ where $k \in \mathcal{J}_1$ and $k \in \mathcal{J}_2$, the created MDS subfiles $f_{d_k,\mathcal{W}'}$ where $\mathcal{W}' \supseteq (\mathcal{J}_1 \setminus \{k\})$ and $f_{d_k,\mathcal{W}''}$ where $\mathcal{W}'' \supseteq (\mathcal{J}_2 \setminus \{k\})$, are different. So in the delivery phase, for each user $k \in [\mathsf{K}]$, we let user k recover the created MDS subfile $f_{d_k,\mathcal{W}'}$ where $\mathcal{W}' \supseteq (\mathcal{J} \setminus \{k\})$ for each $\mathcal{J} \in \mathcal{Z}_g$ where $k \in \mathcal{J}$. Hence, the number of recovered MDS subfiles by each user in the delivery phase is $\mathsf{k}_{2,\mathsf{b}} = g|\mathcal{Z}_g|/\mathsf{K} = 378$.

Placement We divide each file F_i into $\mathsf{k}_{1,\mathsf{b}} + \mathsf{k}_{2,\mathsf{b}}$ non-overlapping and equal-length pieces, which are then encoded by an $(\mathsf{n_b}, \mathsf{k}_{1,\mathsf{b}} + \mathsf{k}_{2,\mathsf{b}})$ MDS code. Each MDS symbol represents one MDS subfile in either Hierarchy 1 or 2 denoted by $f_{i,\mathcal{W}'}$ and cached by users in $\mathcal{W}'$.

Delivery For each set $\mathcal{J} \in \mathcal{Z}_g$, we generate the MAN-like multicast message

$$W'_{\mathcal{J},\mathsf{b}} := \underset{k\in\mathcal{J}}{\oplus} g_{d_k,\mathcal{J}\setminus\{k\}}, \tag{7.58}$$

and $g_{i,\mathcal{W}}$ represents the created MDS subfile $f_{i,\mathcal{W}'}$ where $\mathcal{W}' \supseteq \mathcal{W}$. Since $g > \mathsf{K}_2 + 1$, we have for each $\mathcal{J} \in \mathcal{Z}_g$, $|\mathcal{R}_\mathcal{J}| = 1$. Hence, the server transmits $W'_{\mathcal{J},\mathsf{b}}$ to relay h where $h \in \mathcal{R}_\mathcal{J}$, and relay h forwards it to users in $\mathcal{J}$.

In this example, the needed memory to achieve $g = 6$ is $\mathsf{M_b} = \mathsf{N}\mathsf{k}_{1,\mathsf{b}}/(\mathsf{k}_{1,\mathsf{b}} + \mathsf{k}_{2,\mathsf{b}}) = 9.75$ while the one of [50] is $\mathsf{M}_{([50]} = 10$. □

In the following example, we extend the scheme in Example 10 to create multicasting opportunities among more than two relays.

Example 11 ($\mathsf{H} = 6$, $\mathsf{r} = 3$, $\mathsf{N} = 20$, $g = 8$)**.** We consider the same network as Example 10.

High level description of the proposed scheme We aim to achieve coded caching gain $g = 8$. The main idea is also to create one MDS subfile $f_{i,\mathcal{W}'}$ such that $\mathcal{W}' \supseteq \mathcal{W}$ for each $\mathcal{W} \in \mathcal{Z}_{g-1}$ and each file $i \in [\mathsf{N}]$.

We focus on the set of relays $\{1, 2, 3\}$. Recall that $\mathcal{G}_\mathcal{Y}$ defined in (5.5) represents the set of users connected to at least two relays in $\mathcal{Y}$. In this example, we have

$$\mathcal{G}_{\{1,2,3\}} = \{1, 2, 3, 4, 5, 6, 7, 11, 12, 13\}, \tag{7.59}$$

$$\mathcal{G}_{\{1,2,3\}} \cap \mathcal{U}_1 = \mathcal{U}_{\{1,2\}} \cup \mathcal{U}_{\{1,3\}} = \{1,2,3,4,5,6,7\} \in \mathcal{Z}_{g-1}, \tag{7.60}$$

$$\mathcal{G}_{\{1,2,3\}} \cap \mathcal{U}_2 = \mathcal{U}_{\{1,2\}} \cup \mathcal{U}_{\{2,3\}} = \{1,5,6,7,11,12,13\} \in \mathcal{Z}_{g-1}, \tag{7.61}$$

$$\mathcal{G}_{\{1,2,3\}} \cap \mathcal{U}_3 = \mathcal{U}_{\{1,3\}} \cup \mathcal{U}_{\{2,3\}} = \{1,5,6,7,11,12,13\} \in \mathcal{Z}_{g-1}. \tag{7.62}$$

Hence, we can create one MDS subfile $f_{i,\mathcal{G}_{\{1,2,3\}}}$ cached by users in $\mathcal{G}_{\{1,2,3\}}$ which can cover three sets in $\mathcal{Z}_{g-1}$ for each file $i \in [\mathsf{N}]$. In the delivery phase, each user k in $\mathcal{U}_1 \setminus \mathcal{G}_{\{1,2,3\}} = [8:10]$, in $\mathcal{U}_2 \setminus \mathcal{G}_{\{1,2,3\}} = [14:16]$, in $\mathcal{U}_3 \setminus \mathcal{G}_{\{1,2,3\}} = [17:19]$ will recover $f_{d_k,\mathcal{W}'}$ from relay 1, 2 and 3, respectively. We then repeat the construction for all subsets of three relays. We define $\mathcal{H}_3$ as the collection of the sets $\mathcal{W} \in \mathcal{Z}_{g-1}$ where $\{\mathcal{V} \in \mathsf{S} : \mathcal{V} \subseteq \mathcal{W}\} = 3-1$, i.e., there exist two pairs of relays $\{h_1, h_2\}$ such that $\mathcal{W} \supseteq \mathcal{U}_{\{h_1,h_2\}}$. Hence, considering all the set of three relays, we create the MDS subfiles for each set $\mathcal{W} \in \mathcal{H}_3$. Since we create one MDS subfile corresponding to three sets in $\mathcal{H}_3$ for each file $i \in [\mathsf{N}]$ (we say such MDS subfiles are in Hierarchy 3), the number of created MDS subfiles per file in Hierarchy 3 is denoted by $N_3 := |\mathcal{H}_3|/3 = 20$.

In the next step, we create MDS subfiles for each of two relays, e.g., $\{1,2\}$ whose set of common connected users is $\mathcal{U}_{\{1,2\}} = \{1,2,3,4\}$. Since the set $\mathcal{U}_{\{1,2\}} \cup \mathcal{U}_{\{1,3\}} = \{1,2,3,4,5,6,7\}$ which is in the collection $\mathcal{Z}_{g-1}$, has already been covered by Hierarchy 3, we need not to create MDS subfile for this set in this step. Similarly, we need not to create MDS subfile for $\mathcal{U}_{\{1,2\}} \cup \mathcal{U}_{\{1,h\}}$ where $h \in [\mathsf{H}] \setminus \{1,2\}$ in this step. Hence, each time we choose one set

$$\mathcal{C}' \in \mathcal{C}_{\{1,2\},1} = \big\{\mathcal{C} \subseteq \mathcal{U}_1 \setminus \mathcal{U}_{\{1,2\}} : |\mathcal{C}| = (g-1) - |\mathcal{U}_{\{1,2\}}| = 3, (\mathcal{U}_{\{1,2\}} \cup \mathcal{C}) \not\supseteq \mathcal{U}_{1,h}, h \in [\mathsf{H}] \setminus \{1,2\}\big\}, \tag{7.63}$$

and one set

$$\mathcal{C}'' \in \mathcal{C}_{\{1,2\},2} = \big\{\mathcal{C} \subseteq \mathcal{U}_2 \setminus \mathcal{U}_{\{1,2\}} : |\mathcal{C}| = (g-1) - |\mathcal{U}_{\{1,2\}}| = 3, (\mathcal{U}_{\{1,2\}} \cup \mathcal{C}) \not\supseteq \mathcal{U}_{\{2,h\}}, h \in [\mathsf{H}] \setminus \{1,2\}\big\}, \tag{7.64}$$

and create the MDS subfile $f_{i,\mathcal{U}_{\{1,2\}} \cup \mathcal{C}' \cup \mathcal{C}''}$ cached by $\mathcal{U}_{\{1,2\}} \cup \mathcal{C}' \cup \mathcal{C}''$ for each file $i \in [\mathsf{N}]$. Notice that each element in $\mathcal{C}_{\{1,2\},1}$ and $\mathcal{C}_{\{1,2\},2}$ is only chosen one time. We then repeat the construction for all subsets of two relays. We recall that $\mathcal{H}_2$ is the collection of the sets $\mathcal{W} \in \mathcal{Z}_{g-1}$ where $|\{\mathcal{V} \in \mathsf{S} : \mathcal{V} \subseteq \mathcal{W}\}| = 2-1$. Hence, considering all the set of two relays, we create the MDS subfiles for each set $\mathcal{W} \in \mathcal{H}_2$. Since we create one MDS subfile corresponding to two sets in $\mathcal{H}_2$ for each file $i \in [\mathsf{N}]$ (we say such MDS subfiles are in Hierarchy 2), the number of created MDS subfiles per file in Hierarchy 2 is denoted by $N_2 := |\mathcal{H}_2|/2 = 240$.

In the last step, for each set $\mathcal{W} \in \mathcal{H}_1$, we directly create one MDS subfile $f_{i,\mathcal{W}}$ cached by users in $\mathcal{W}$ for each file $i \in [\mathsf{N}]$ (we say such MDS subfiles are in Hierarchy 1). The number of created MDS subfiles per file in Hierarchy 1 is denoted by $N_1 := |\mathcal{H}_1| = 180$.

Hence, we totally create $\mathsf{n_b} = N_1 + N_2 + N_3 = 440$ MDS subfiles per file. Each MDS subfile in Hierarchy 1, 2 and 3 is known by 7, 10 and 10 users, respectively. Hence, each user caches $\mathsf{k_{1,b}} = (7N_1 + 10N_2 + 10N_3)/\mathsf{K} = 193$ MDS subfiles per file.

If we focus on one user k, for any sets $\mathcal{J}_1, \mathcal{J}_2 \in \mathcal{Z}_g$ where $k \in \mathcal{J}_1$ and $k \in \mathcal{J}_2$, we can see that the created MDS subfiles $f_{d_k,\mathcal{W}'}$ where $\mathcal{W}' \supseteq (\mathcal{J}_1 \setminus \{k\})$ and $f_{d_k,\mathcal{W}''}$ where $\mathcal{W}'' \supseteq (\mathcal{J}_2 \setminus \{k\})$, are different. So in the delivery phase, for each user $k \in [\mathsf{K}]$, we let user k recover the created MDS subfile $f_{d_k,\mathcal{W}'}$ where $\mathcal{W}' \supseteq (\mathcal{J} \setminus \{k\})$ for each $\mathcal{J} \in \mathcal{Z}_g$ where $k \in \mathcal{J}$. Hence, the number of recovered MDS subfiles by each

user in the delivery phase is $\mathsf{k}_{2,\mathsf{b}} = g|\mathcal{Z}_g|/\mathsf{K} = 108$.

Placement We divide each file F_i into $\mathsf{k}_{1,\mathsf{b}} + \mathsf{k}_{2,\mathsf{b}}$ non-overlapping and equal-length pieces, which are then encoded by an $(\mathsf{n}_\mathsf{b}, \mathsf{k}_{1,\mathsf{b}} + \mathsf{k}_{2,\mathsf{b}})$ MDS code. Each MDS symbol represents one MDS subfile in either Hierarchy 1, 2 or 3 denoted by $f_{i,\mathcal{W}'}$ and cached by users in $\mathcal{W}'$ for each file $i \in [\mathsf{N}]$.

Delivery For each set $\mathcal{J} \in \mathcal{Z}_g$, we generate the MAN-like multicast message

$$W'_{\mathcal{J},\mathsf{b}} := \underset{k\in\mathcal{J}}{\oplus} g_{d_k,\mathcal{J}\setminus\{k\}}, \tag{7.65}$$

and $g_{i,\mathcal{W}}$ represents the created MDS subfile $f_{i,\mathcal{W}'}$ where $\mathcal{W}' \supseteq \mathcal{W}$. Since $g > \mathsf{K}_2 + 1$, we have for each $\mathcal{J} \in \mathcal{Z}_g$, $|\mathcal{R}_\mathcal{J}| = 1$. Hence, the server transmits $W'_{\mathcal{J},\mathsf{b}}$ to relay h where $h \in \mathcal{R}_\mathcal{J}$, and relay h forwards it to users in $\mathcal{J}$.

In this example, the needed memory to achieve $g = 8$ is $\mathsf{M}_\mathsf{b} = \mathsf{N}\mathsf{k}_{1,\mathsf{b}}/(\mathsf{k}_{1,\mathsf{b}} + \mathsf{k}_{2,\mathsf{b}}) = 12.82$ while the one of [50] is $\mathsf{M}_{([50]} = 14$. □

Now we are ready to present our proposed scheme for $g \in [\mathsf{K}_2 + 2 : \mathsf{K}_1]$.

High level description of the proposed scheme The main idea is to create one MDS subfile $f_{i,\mathcal{W}'}$ such that $\mathcal{W}' \supseteq \mathcal{W}$ for each $\mathcal{W} \in \mathcal{Z}_{g-1}$ and each file $i \in [\mathsf{N}]$. We divide $\mathcal{Z}_{g-1}$ into $\mathsf{H} - \mathsf{r} + 1$ collections and define that

$$\mathcal{H}_m := \big\{\mathcal{W} \in \mathcal{Z}_{g-1} : \{\mathcal{V} \in \mathsf{S} : \mathcal{V} \subseteq \mathcal{W}\} = m - 1\big\} \text{ for } m \in [\mathsf{H} - \mathsf{r} + 1]. \tag{7.66}$$

The created MDS subfiles for the sets in $\mathcal{H}_m$ are defined in Hierarchy m. For each MDS subfile $f_{i,\mathcal{W}'}$ in Hierarchy m, we have $|\{\mathcal{W} \in \mathcal{Z}_{g-1} : \mathcal{W} \subseteq \mathcal{W}'\}| = m$.

We now focus on $\mathcal{H}_m$. For any set of relays $\mathcal{Y} \subseteq [\mathsf{H}]$ where $|\mathcal{Y}| = m$ and any relay $h \in \mathcal{Y}$, we can compute that

$$|\mathcal{G}_\mathcal{Y} \cap \mathcal{U}_h| = |\underset{h_1\in\mathcal{Y}\setminus\{h\}}{\cup} \mathcal{U}_{\{h,h_1\}}| = \mathsf{K}_2 + \binom{\mathsf{H}-3}{\mathsf{r}-2} + \ldots + \binom{\mathsf{H}-m-1}{\mathsf{r}-2} = \mathsf{K}_1 - \binom{\mathsf{H}-m}{\mathsf{r}-1}. \tag{7.67}$$

If $\mathsf{K}_1 - \binom{\mathsf{H}-m}{\mathsf{r}-1} > g - 1$, we have $|\mathcal{H}_m| = 0$ and there is no created MDS subfile in Hierarchy m; otherwise, for each file $i \in [\mathsf{N}]$ we use the following way to create MDS subfiles for $\mathcal{H}_m$ as described in Example 11. We focus on one set of relays $\mathcal{Y} \subseteq [\mathsf{H}]$ where $|\mathcal{Y}| = m$. For each relay $h \in \mathcal{Y}$, we define that

$$\mathcal{C}_{h,\mathcal{Y}} := \{\mathcal{C} \subseteq \mathcal{U}_h \setminus \mathcal{G}_\mathcal{Y} : |\mathcal{C}| = (g-1) - |\mathcal{G}_\mathcal{Y} \cap \mathcal{U}_h|, (\mathcal{G}_\mathcal{Y} \cup \mathcal{C}) \not\supseteq \mathcal{U}_{\{h,h_1\}}, h_1 \in [\mathsf{H}] \setminus \mathcal{Y}\}. \tag{7.68}$$

Each time we choose one element (denoted by $\mathcal{C}^h$) in $\mathcal{C}_{h,\mathcal{Y}}$ for each $h \in \mathcal{Y}$. We create one MDS subfile $f_{i,\left(\underset{h\in\mathcal{Y}}{\cup}\mathcal{C}^h\right)\cup\mathcal{G}_\mathcal{Y}}$ cached by users in $\left(\underset{h\in\mathcal{Y}}{\cup}\mathcal{C}^h\right) \cup \mathcal{G}_\mathcal{Y}$ for each file $i \in [\mathsf{N}]$. Notice that each element in $\mathcal{C}_{h,\mathcal{Y}}$ where $h \in \mathcal{Y}$ is only chosen one time. Since $\mathcal{G}_\mathcal{Y}$ is the set of users connected to at least two relays in $\mathcal{Y}$, We

can compute that

$$|\mathcal{G}_{\mathcal{Y}}| = \mathsf{K} - \binom{\mathsf{H}-m}{\mathsf{r}} - m\binom{\mathsf{H}-m}{\mathsf{r}-1} \tag{7.69}$$

From (7.67) and (7.69), we can see that each MDS file in Hierarchy m is known by $\mathsf{K} - \binom{\mathsf{H}-m}{\mathsf{r}} - m\binom{\mathsf{H}-m}{\mathsf{r}-1} + m(g-1-\mathsf{K}_1+\binom{\mathsf{H}-m}{\mathsf{r}-1}) = \mathsf{K} - \binom{\mathsf{H}-m}{\mathsf{r}} - m\mathsf{K}_1 + m(g-1)$ users. We then repeat the construction for all subsets of m relays. The number of created MDS subfiles per file in Hierarchy m is denoted by

$$N_m := |\mathcal{H}_m|/m. \tag{7.70}$$

In Appendix A.14, we prove that N_m is given in (7.47d) for $m \in [2 : \mathsf{H}-\mathsf{r}+1]$ and (7.47e) for $m = 1$.

Hence, we totally create $\mathsf{n}_{\mathrm{b}} = \sum_{m=1}^{\mathsf{H}-\mathsf{r}+1} N_m$ MDS subfiles per file. Each MDS subfile in Hierarchy m is known by $\mathsf{K} - \binom{\mathsf{H}-m}{\mathsf{r}} - m\mathsf{K}_1 + m(g-1)$ users. Hence, each user caches $\mathsf{k}_{1,\mathrm{b}} = \sum_{m=1}^{\mathsf{H}-\mathsf{r}+1} N_m \frac{\mathsf{K}-\binom{\mathsf{H}-m}{\mathsf{r}}-m\mathsf{K}_1+m(g-1)}{\mathsf{K}}$ MDS subfiles per file.

Notice that for each generated MDS subfile in Hierarchy m (assumed to be $f_{i,\mathcal{W}'}$ from relay $h \in \mathcal{Y}$ where $|\mathcal{Y}| = m$), since $\mathcal{W}' \supseteq \mathcal{G}_{\mathcal{Y}}$, we have $(\mathcal{U}_{h_1} \setminus \mathcal{W}') \cap (\mathcal{U}_{h_2} \setminus \mathcal{W}') = \emptyset$ for any two relays $h_1, h_2 \in \mathcal{Y}$. Hence, for each user $k \in [\mathsf{K}]$ and for any sets $\mathcal{J}_1, \mathcal{J}_2 \in \mathcal{Z}_g$ where $k \in \mathcal{J}_1$ and $k \in \mathcal{J}_2$, we can see that the created MDS subfiles $f_{d_k,\mathcal{W}'}$ where $\mathcal{W}' \supseteq (\mathcal{J}_1 \setminus \{k\})$ and $f_{d_k,\mathcal{W}''}$ where $\mathcal{W}'' \supseteq (\mathcal{J}_2 \setminus \{k\})$, are different. So in the delivery phase, for each user $k \in [\mathsf{K}]$, we let user k recover the created MDS subfile $f_{d_k,\mathcal{W}'}$ where $\mathcal{W}' \supseteq (\mathcal{J} \setminus \{k\})$ for each $\mathcal{J} \in \mathcal{Z}_g$ where $k \in \mathcal{J}$. Hence, the number of recovered MDS subfiles by each user in the delivery phase is $\mathsf{k}_{2,\mathrm{b}} = g|\mathcal{Z}_g|/\mathsf{K}$.

Placement We divide each file F_i into $\mathsf{k}_{1,\mathrm{b}} + \mathsf{k}_{2,\mathrm{b}}$ non-overlapping and equal-length pieces, which are then encoded by an $(\mathsf{n}_{\mathrm{b}}, \mathsf{k}_{1,\mathrm{b}} + \mathsf{k}_{2,\mathrm{b}})$ MDS code. Each MDS symbol represents one MDS subfile in Hierarchy $m \in [\mathsf{H}-\mathsf{r}+1]$ denoted by $f_{i,\mathcal{W}'}$ and cached by users in $\mathcal{W}'$ for each file $i \in [\mathsf{N}]$. The needed memory size per file to achieve the coded caching gain g is $\frac{\mathsf{M}_{\mathrm{b}}}{\mathsf{N}} = \mathsf{k}_{1,\mathrm{b}}/(\mathsf{k}_{1,\mathrm{b}} + \mathsf{k}_{2,\mathrm{b}})$.

Delivery For each set $\mathcal{J} \in \mathcal{Z}_g$, we generate the MAN-like multicast message

$$W'_{\mathcal{J},\mathrm{b}} := \underset{k\in\mathcal{J}}{\oplus} g_{d_k,\mathcal{J}\setminus\{k\}}, \tag{7.71}$$

and $g_{i,\mathcal{W}}$ represents the created MDS subfile $f_{i,\mathcal{W}'}$ where $\mathcal{W}' \supseteq \mathcal{W}$. Since $g > \mathsf{K}_2 + 1$, we have for each $\mathcal{J} \in \mathcal{Z}_g$, $|\mathcal{R}_{\mathcal{J}}| = 1$. Hence, the server transmits $W'_{\mathcal{J},\mathrm{b}}$ to relay h where $h \in \mathcal{R}_{\mathcal{J}}$, and relay h forwards it to users in $\mathcal{J}$.

Performance Hence, each demanded MDS subfile is multicasted from the server with other $g-1$ demanded MDS subfiles and thus the max link-load from the server to relays is $\mathsf{K}(1-\mathsf{M}_{\mathrm{b}}/\mathsf{N})/(\mathsf{H}g)$. In addition, since $|g_{d_{k_1},\mathcal{J}\setminus\{k_1\}}| = |g_{d_{k_2},\mathcal{J}\setminus\{k_2\}}|$ for each set $\mathcal{J} \in \mathcal{Z}_g$ and any $k_1, k_2 \in \mathcal{J}$, and each user recovers $\mathsf{B}(1-\mathsf{M}_{\mathrm{b}}/\mathsf{N})/\mathsf{r}$ bits from r relays, the load on each link from relays to users is $(1-\mathsf{M}_{\mathrm{b}}/Nsf)/\mathsf{r}$. With $\mathsf{K}/(\mathsf{H}g) \geq \mathsf{r}$, the max link-load is $\mathsf{K}(1-\mathsf{M}_{\mathrm{b}}/\mathsf{N})/(\mathsf{H}g)$. To achieve coded caching gain g, the needed memory size is given in (7.47a).

Remark 15. *It can be checked that the delivery phases in Section 7.4.1 and Section 7.4.2 are equivalent to SRDS; in other word, for each set $\mathcal{J} \in \mathcal{Z}_g$ and each user $k \in \mathcal{J}$, assuming $f_{i,\mathcal{W}'} = g_{i,\mathcal{J}\setminus\{k\}}$, we have $\mathcal{R}_{\mathcal{J}} = arg\max_{h\in\mathcal{H}_k} |\mathcal{U}_h \cap \mathcal{W}'|$. Hence, with the proposed asymmetric coded placement, the SRDS delivery phase becomes symmetric.*

Chapter 8

Performance Analysis and Numerical Evaluations for Combination Networks with End-user-caches

8.1 Performance Analysis for Combination Networks with End-user-caches

8.1.1 Optimality Results

We first give the following information theoretical optimality results of the proposed asymmetric coded placement scheme, whose proof is in Appendix A.12.

Theorem 18. *For an* $(\mathsf{H},\mathsf{r},\mathsf{M},\mathsf{N})$ *combination network, when* $\mathsf{M}\in\left[\frac{(\mathsf{K}-\mathsf{H}+\mathsf{r}-1)\mathsf{N}}{\mathsf{K}},\mathsf{N}\right]$*, we have*

$$\mathsf{R}_{\mathsf{c}}^{\star}=\mathsf{R}_{b}=\frac{1-\mathsf{M}/\mathsf{N}}{\mathsf{r}}. \tag{8.1}$$

Remark 16. *As claimed in Appendix A.12, when* $\mathsf{M}=\frac{(\mathsf{K}-\mathsf{H}+\mathsf{r}-1)\mathsf{N}}{\mathsf{K}}$ *corresponding to* $g=\mathsf{K}_1$*, the proposed non-separation cache placement is uncoded. The description of the placement can be simplified as follows. We divide each file* $i\in[\mathsf{N}]$ *into* $\binom{\mathsf{H}}{\mathsf{r}-1}$ *non-overlapping and equal-length subfiles. For each subset of relays* $\mathcal{Y}\subseteq[\mathsf{H}]$ *where* $|\mathcal{Y}|=\mathsf{H}-\mathsf{r}+1$*, there is a subfile of each file* $i\in[\mathsf{N}]$ *known by users in* $\mathcal{G}_{\mathcal{Y}}$*.*

In the following, we introduce the optimality results under the constraint of uncoded cache placement. We first compare the cut-set converse bound in (6.2), the converse bound in Theorem 11 and the aforementioned proposed schemes, in order to derive the following optimality results under the constraint of uncoded cache placement, whose detailed proof can be found in Appendix A.10.

Theorem 19. *For combination networks with end-user-caches, under the constraint of uncoded cache placement and* $\mathsf{N}\geq\mathsf{K}=\binom{\mathsf{H}}{\mathsf{r}}$*, we have*

1. $\mathsf{H}<\mathsf{r}\frac{t+1}{t}$ *and* $t=\mathsf{KM}/\mathsf{N}\in[0:\mathsf{K}]$*. The optimal max link-load*

$$\mathsf{R}_{\mathsf{c},\mathrm{u}}^{\star}=\frac{\mathsf{K}-t}{\mathsf{H}(t+1)}, \tag{8.2a}$$

which is achieved by DIS, CICS and ICICS.

2. $\mathsf{r} = \mathsf{H} - 1$. *The optimal max link-load*

$$\mathsf{R}^{\star}_{\mathrm{c,u}} = \frac{\mathsf{K} - t}{\mathsf{H}(t+1)} \text{ when } t \in [0 : \mathsf{K} - 2]; \tag{8.2b}$$

$$\mathsf{R}^{\star}_{\mathrm{c,u}} = \frac{\mathsf{K} - t}{\mathsf{H}(\mathsf{K}-1)} \text{ when } t \in [\mathsf{K} - 2 : \mathsf{K}]; \tag{8.2c}$$

which is achieved by DIS, CICS and ICICS.

3. $\mathsf{M} \leq \frac{\mathsf{N}}{\mathsf{K}}$. *The optimal max link-load*

$$\mathsf{R}^{\star}_{\mathrm{c,u}} = \frac{\mathsf{K}}{\mathsf{H}} - \frac{\mathsf{K}+1}{2\mathsf{H}} \frac{\mathsf{KM}}{\mathsf{N}}, \text{ when } \mathsf{H} < 2\mathsf{r}; \tag{8.2d}$$

$$\mathsf{R}^{\star}_{\mathrm{c,u}} = \frac{\mathsf{K}(\mathsf{H}-1) - (\frac{\mathsf{KH}+\mathsf{H}-\mathsf{K}}{2} - 1)\frac{\mathsf{KM}}{\mathsf{N}}}{\mathsf{H}(\mathsf{H}-1)}, \text{ when } \mathsf{H} = 2\mathsf{r}, \tag{8.2e}$$

which is achieved by IES.

By using the techniques in [27], in the above regimes the mentioned schemes are optimal to within a factor 2.

We then compare the cut-set converse bound and the achieved max link-loads of CICS and IES to derive the following order optimality results, whose detailed proof is in Appendix A.11.

Theorem 20. *For combination networks with end-user-caches, under the constraint of uncoded cache placement and* $\mathsf{N} \geq \mathsf{K} = \binom{\mathsf{H}}{\mathsf{r}}$*, we have*

1. *DIS and the scheme in [12] is order optimal within a factor of* $\min\{\mathsf{H}/\mathsf{r}, t+1\}$.
2. $\mathsf{M} \leq \frac{t\mathsf{N}}{\mathsf{K}}$, *CICS is order optimal within a factor of* $1 + t/\mathsf{r}$.
3. $\mathsf{M} \leq \frac{\mathsf{r}\mathsf{N}}{\mathsf{K}}$, *CICS is order optimal within a factor of* 2.
4. $\mathsf{M} \leq \frac{t\mathsf{N}}{\mathsf{K}}$, *and* $t/\mathsf{r} \to 0$, *CICS is order optimal within a factor closed to* 1.
5. $\mathsf{H} > 2\mathsf{r}$ *and* $\mathsf{M} \leq \frac{\mathsf{N}}{\mathsf{K}}$, *IES is order optimal within a factor of* $\frac{2\mathsf{r}}{2\mathsf{r}-1} \leq \frac{4}{3}$.

By using the techniques in [27], in each of the above regimes the mentioned scheme is optimal to within the previous factor multiplied by 2.

Notice that in [12], where it was proved that the caching schemes in [12] is order optimal within factor $\max\{6\sqrt{3}, 6\log(\mathsf{N}/\mathsf{M})\}$ and thus when M is small the factor becomes large. In this thesis, we give the order optimality results for small memory size M.

8.1.2 Comparison to Existing Schemes

Comparing the achieved loads in (13) and (1.7), we have the following theorem.

Theorem 21. *For a* $(\mathsf{H}, \mathsf{r}, \mathsf{M}, \mathsf{N})$ *combination network, it holds that*

$$\mathsf{R}_{\mathrm{DIS}} \leq \mathsf{R}_{MJ}, \tag{8.3}$$

where R_{MJ} *defined in* (1.7) *is achieved by the caching schemes in [12].*

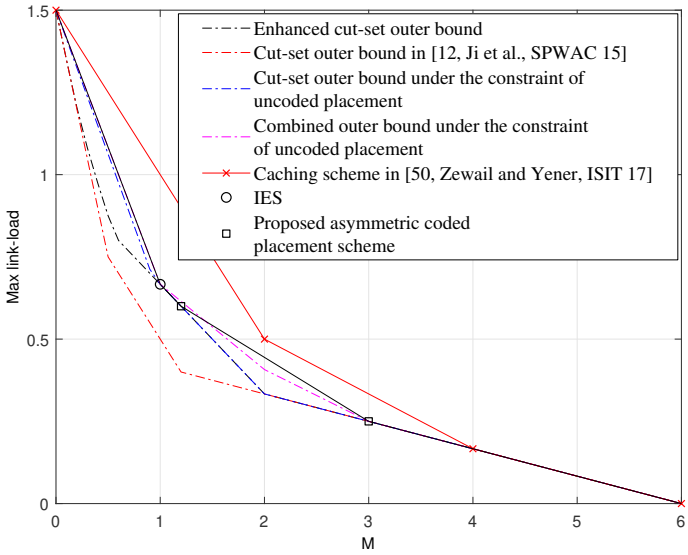

FIGURE 8.1: A combination network with end-user-caches in centralized caching systems, with $\mathsf{H} = 4$ relays, $\mathsf{K} = 6$ users and $\mathsf{N} = 6$ files, i.e., $\mathsf{r} = 2$.

Proof. It is obvious to prove that DIS is better than the second scheme in [12] which transmit $|\mathcal{W}_{\mathcal{J}}|/\mathsf{r}$ bits to each relay $h \in [\mathsf{H}]$. The first scheme in [12] is equivalent to use $|\mathcal{J}||\mathcal{W}_{\mathcal{J}}|$ bits to transmit $\mathcal{W}_{\mathcal{J}}$. However, in DIS, we use at most $|\mathcal{J}||\mathcal{W}_{\mathcal{J}}|$ bits to transmit $\mathcal{W}_{\mathcal{J}}$; only when each two users $k_1, k_2 \in Jc$, we have $\mathcal{H}_{k_1} \cap \mathcal{H}_{k_2} = \emptyset$, we use $|\mathcal{J}||\mathcal{W}_{\mathcal{J}}|$ bits. Hence, the max link-load of DIS is lower than the one in [12]. □

In the following we show that our proposed asymmetric coded placement with SRDS is strictly better than the scheme in [50]. For example, it can be checked from Theorem 17 that for the combination network with $\mathsf{H} = 4$, $\mathsf{r} = 2$, $\mathsf{N} = \mathsf{K} = 6$ in Fig. 1.4, to achieve $g = 2$, the needed memory size of the proposed scheme is $6/5$ while the one of [50] is 2. In addition, when $\mathsf{M} = 6/5$, the achieved max-link load in Theorem 17 is $3/5$, which coincides with the converse bound in Theorem 7, while the max-link load with the scheme from [50] is $9/10$. Notice that the converse bound under the constraint of uncoded placement in Theorem 12 is $157/255 \approx 0.616$, that is, in this example using uncoded cache placement is strictly suboptimal. In general we have the following corollary whose proof is in Appendix A.13.

Theorem 22. *For an* $(\mathsf{H}, \mathsf{r}, \mathsf{M}, \mathsf{N})$ *combination network achieving each coded caching gain* $g \in [2 : \mathsf{K}_1]$*, it holds that the minimum needed memory size of the proposed scheme is strictly less than the one of [50], i.e.,* $\mathsf{M}_b < \mathsf{M}_{ZY}$.

8.2 Numerical Evaluations for Combination Networks with End-user-caches

In this part, we will show the numerical results of our proposed converse bounds and novel delivery schemes for combination networks with end-user-caches.

8.2.1 Example for $\mathsf{H} = 4$, $\mathsf{r} = 2$, $\mathsf{N} = \mathsf{K} = 6$

Firstly, we give an example where $\mathsf{H} = 4$, $\mathsf{r} = 2$ and $\mathsf{N} = \mathsf{K} = \binom{\mathsf{H}}{\mathsf{r}} = 6$, to illustrate the proposed achievable bounds and converse bounds compared to the state-of-the-art. In Fig. 8.1, we plot the cut-set converse bound in [12] (described in (1.9) of this thesis), the proposed enhanced cut-set converse bound in Theorem 7, the

cut-set converse bound under the constraint of uncoded placement in Theorem 8 and the converse bound under the constraint of uncoded placement based on the improved acyclic index coding converse bound in Theorem 12. It can be seen in Fig. 8.1 that the enhanced cut-set converse bound tightens the converse bound by directly extending the cut-set converse bound for shared-link networks to combination networks, and the converse bound under the constraint of uncoded placement based on the improved acyclic index coding converse bound tightens the converse bound by directly extending the converse bound under the constraint of uncoded placement for shared-link models to combination networks.

We also plot the lowest load achieved by all the proposed schemes in this thesis. More precisely, it contains five corner points: 1. $(\mathsf{M},\mathsf{R}) = (0, 3/2)$ for zero-cache case; 2. $(\mathsf{M},\mathsf{R}) = (1, 2/3)$ achieved by IES in Section 7.2.2; 3. $(\mathsf{M},\mathsf{R}) = (6/5, 3/5)$ achieved by the asymmetric coded placement scheme in Section 7.4.1; 4. $(\mathsf{M},\mathsf{R}) = (3, 1/4)$ achieved by the asymmetric coded placement scheme in Section 7.4.1; 5. $(\mathsf{M},\mathsf{R}) = (6, 0)$ where each user caches the full information of all the N files.

It can be seen that when $\mathsf{M} = 1$, IES coincides with the enhanced cut-set converse bound in Theorem 7 and with the converse bound under the constraint of uncoded placement in Theorem 12. So it corresponds to the optimality results in (8.2e). When $\mathsf{M} = 6/5$, the asymmetric coded placement coincides with the enhanced cut-set converse bound in Theorem 7, which is strictly lower than the converse bound under the constraint of uncoded placement in Theorem 12. So in this case, uncoded cache placement is sub-optimal. When $\mathsf{M} = 3$, the asymmetric coded placement coincides with the enhanced cut-set converse bound in Theorem 7, which corresponds to the optimality results in Theorem 18.

8.2.2 Examples for $\mathsf{H} = 6$, $\mathsf{r} = 2$, $\mathsf{N} = \mathsf{K} = 15$ and $\mathsf{H} = 6$, $\mathsf{r} = 3$, $\mathsf{N} = \mathsf{K} = 20$

We then compare the performance of the proposed caching schemes to the existing caching schemes in [12], [50]. In these examples, the caching scheme in [50] outperforms the one in [12]. So we only plot the caching scheme in [50]. As a converse bound, we use the enhanced cut-set converse bound in Theorem 7. We also plot the ratios $\mathsf{R}_{\text{IES}}/\mathsf{R}_{\text{ZY}}$ of IES, $\min(\mathsf{R}_{\text{ICICS}}, \mathsf{R}_{\text{SRDS}})/\mathsf{R}_{\text{ZY}}$ of ICICS and SRDS with cMAN placement, $\mathsf{R}_{\text{b}}/\mathsf{R}_{\text{ZY}}$ of the asymmetric coded cache placement scheme, and $\mathsf{R}_{\text{c}}^{\star}/\mathsf{R}_{\text{ZY}}$ of the converse bound in Theorem 7. We plot the ratio of max-link loads as otherwise their difference would not be clearly visible on a small figure.

In Fig. 8.2 and 8.3, we consider the combination network with $\mathsf{H} = 6$, $\mathsf{r} = 2$ and $\mathsf{H} = 6$, $\mathsf{r} = 3$, respectively. It can be noted from the right-hand side figures of Fig. 8.2 and 8.3, that the blue curves, which represent the asymmetric coded placement scheme, are never above 1, that is, it is never inferior in performance to the caching scheme in [50]. Moreover, the asymmetric coded placement scheme coincides with the enhanced cut-set converse bound when $\mathsf{M} \geq 10$ if $\mathsf{r} = 2$ and when $\mathsf{M} \geq 16$ if $\mathsf{r} = 3$. In addition, the proposed delivery schemes with cMAN placement outperform the asymmetric coded placement scheme when $\mathsf{M} \leq 1.6$ if $\mathsf{r} = 2$ and when $\mathsf{M} \leq 10$ if $\mathsf{r} = 3$.

8.2.3 Numerical Evaluations for Decentralized Combination Networks with End-user-caches

Following similar steps to [5], we can extend our proposed delivery schemes (DIS, IES, CICS, ICICS and SRDS), to decentralized systems in combination networks. The cache placement phase is the same as dMAN in [5], i.e., each user $k \in [\mathsf{K}]$ stores MB/N bits of each file independently at random. Given the

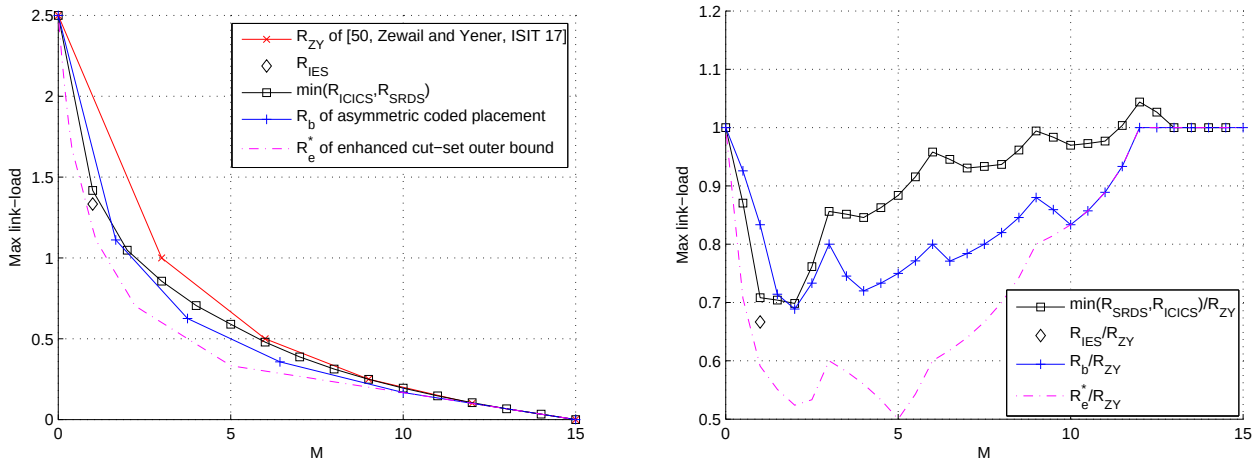

FIGURE 8.2: A combination network with end-user-caches, where $H = 6$, $N = K = \binom{H}{r}$ and $r = 2$.

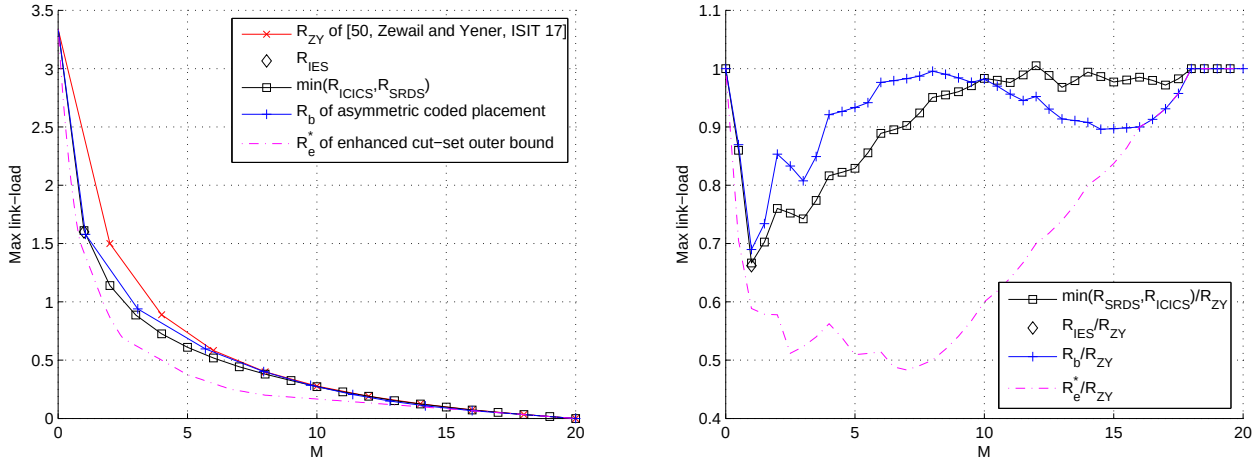

FIGURE 8.3: A combination network with end-user-caches, where $H = 6$, $N = K = \binom{H}{r}$ and $r = 3$.

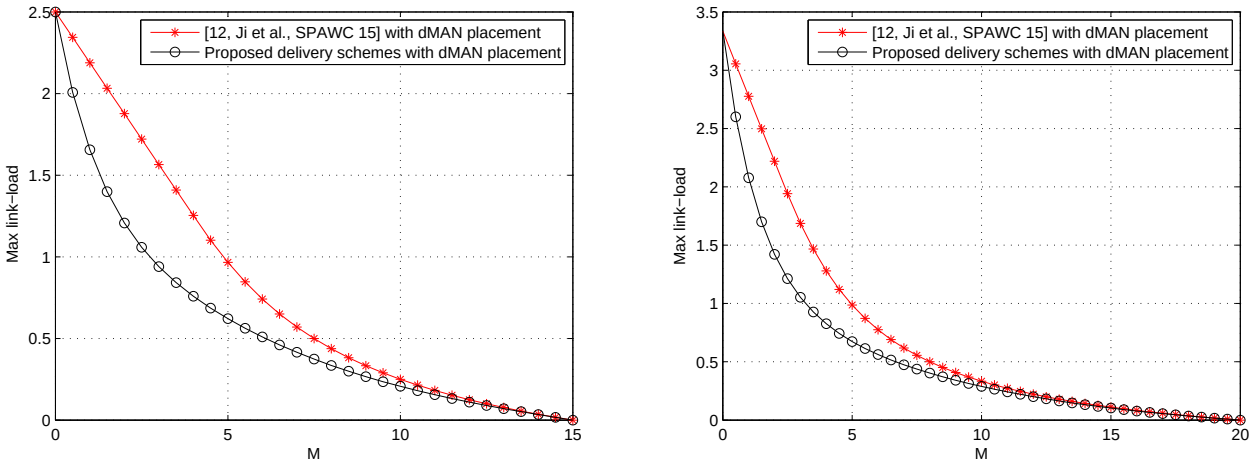

FIGURE 8.4: A combination network in decentralized caching systems with end-user-caches, with $\mathsf{H} = 6$ relays, $\mathsf{N} = \mathsf{K} = \binom{\mathsf{H}}{\mathsf{r}}$ and $\mathsf{r} = 2$ and 3.

cache content of all the users, each file can be divided into sub-files. $F_{i,\mathcal{W}}$ represents the bits of file F_i which are only known by the users in $\mathcal{W} \subseteq [1 : K]$. By the law of the large number, the length of each sub-file only depends on the number of user knowing it. In the delivery phase, an iteration is run from $\mathsf{K}-1$ to 0. In each loop $t \in [0 : \mathsf{K}-1]$, all the $\binom{\mathsf{K}}{t+1}$ sub-files $(F_{i,\mathcal{W}} : |\mathcal{W}| = t,\ i \in [\mathsf{N}])$ are gathered together to be transmitted by the chosen proposed scheme with the lowest load among all the proposed delivery phases for the centralized case where $\mathsf{M} = \mathsf{N}t/\mathsf{K}$.

Notice that the proposed asymmetric coded placement and the placements of the scheme in [50] and in [49] are designed based on the knowledge of the position (the connected relays) of each user in the delivery phase. Hence, it is hard to extend these two schemes to the decentralized systems due to the mobility of the users. Fig. 8.4 shows that our proposed scheme outperforms the state-of-the-art schemes.

Chapter 9

Cache-aided Extended Models

9.1 Chapter Overview and Related Publications

In this chapter, we extend our results for combination networks with end-user-caches to more general models, including combination networks with cache-aided relays and users, and combination networks with more general relay networks. Different to the existing scheme in [50] where the packets transmitted from the server are independent of the cached contents of relays, in Section 9.2 we propose a caching scheme where the cached contents in relays can also help users to decode the coded messages transmitted from the server and thus can further reduce the transmitted load from the server to relays. We prove that for some memory size regime, the achieved first layer load of the proposed scheme is information theoretically optimal. We also show that the proposed scheme reduces the first layer load of the state-of-the-art scheme. For general relay networks, due to the asymmetry, the loads transmitted to relays may be different. Based on the proposed SRDS, in Section 9.3, we propose a further step to 'balance' the loads transmitted to relays.

Related Publications

1. Kai Wan, Mingyue Ji, Pablo Piantanida and Daniela Tuninetti, "Caching in Combination Networks: Novel Asymmetric Coded Cache Placement and Multicast Message Generation by Leveraging Network Topology", *in preparation*, to IEEE Trans. on Information Theory. [54]

2. Kai Wan, Mingyue Ji, Pablo Piantanida and Daniela Tuninetti, "Caching in Combination Networks: Novel Multicast Message Generation and Delivery by Leveraging the Network Topology", in IEEE Intern. Conf. Commun. (ICC), May 2018. [57]

3. Kai Wan, Daniela Tuninetti, Pablo Piantanida and Mingyue Ji, "On Combination Networks with Cache-aided Relays and Users", in Proceedings of Workshop on Smart Antennas (WSA), Mar. 2018. [58]

9.2 Combination Networks with Cache-Aided Relays and Cache-Aided Users

Combination networks with both cache-aided relays and users were considered in [50], where each relay can store $\mathsf{M}^{\text{relay}}\mathsf{B}$ bits and each user can store $\mathsf{M}^{\text{user}}\mathsf{B}$ bits, for $\mathsf{M}^{\text{relay}} \in [0, \mathsf{N}/\mathsf{r}]$ and $\mathsf{M}^{\text{user}} \in [0, \mathsf{N}]$. Notice that if $\mathsf{M}^{\text{relay}} = \mathsf{N}/\mathsf{r}$, each user can get all the N files from its connected relays. For each $(\mathsf{H}, \mathsf{r}, \mathsf{M}^{\text{relay}}, \mathsf{M}^{\text{user}}, \mathsf{N})$

combination network, the objective is to determine the lower convex envelop of the load (number of transmitted bits in the delivery phase) pairs

$$(\mathsf{R}^{\mathsf{s}\to\mathsf{r}}, \mathsf{R}^{\mathsf{r}\to\mathsf{u}}) = \big(\max_{h\in[\mathsf{H}]} \mathsf{R}_h(\mathbf{d},\mathbf{Z}), \max_{h\in[\mathsf{H}],k\in\mathcal{U}_h} \mathsf{R}_{h\to k}(\mathbf{d},\mathbf{Z})\big)$$

for the worst case demands $\mathbf{d}$ for a given placement $\mathbf{Z}$.

The authors in [50] extended the cut-set converse bound in (1.9) to the combination networks with cache-aided relays and users.

Theorem 23. *In a* $(\mathsf{H},\mathsf{r},\mathsf{M}^{relay},\mathsf{M}^{user},\mathsf{N})$ *combination network, the optimal loads are lower bounded by*

$$\mathsf{R}^{\mathsf{s}\to\mathsf{r}\star} \geq \begin{cases} \max_{l\in[\mathsf{r}:\mathsf{H}]} \max_{s\in\left[\min\left\{\mathsf{N},\binom{l}{\mathsf{r}}\right\}\right]} \frac{1}{l}\left(s - \frac{s\mathsf{M}^{user}+l\mathsf{M}^{relay}}{\lfloor \mathsf{N}/s\rfloor}\right), \text{ if } \mathsf{r}\mathsf{M}^{relay}+\mathsf{M}^{user}\leq \mathsf{N}, \\ 0, \text{ if } \mathsf{r}\mathsf{M}^{relay}+\mathsf{M}^{user}\geq \mathsf{N}, \end{cases} \tag{9.1a}$$

$$\mathsf{R}^{\mathsf{r}\to\mathsf{u}\star} \geq \frac{1}{\mathsf{r}}\left(1-\frac{\mathsf{M}^{user}}{\mathsf{N}}\right). \tag{9.1b}$$

In practice, the throughput of transmission from the server to relays may be much lower than the throughput from the relays to their local connected users. For example, in wireless networks where the throughput from small cell base stations to users are much higher than that from the macro base stations to small base stations if all use sub-6GHz wireless communications. In this thesis, for combination networks with cache-aided relays and users, we mainly want to minimize the max-link load from the server to relays, i.e., $\mathsf{R}^{\mathsf{s}\to\mathsf{r}} = \max_{h\in[\mathsf{H}]} \mathsf{R}_h(\mathbf{d},\mathbf{Z})$. For a caching scheme with max-link load among all the links from the server to relays $\mathsf{R}^{\mathsf{s}\to\mathsf{r}}$, we say it attains a *coded caching gain* of g if

$$\mathsf{R}^{\mathsf{s}\to\mathsf{r}} = \frac{\mathsf{R}^{\mathsf{s}\to\mathsf{r}}_{\text{routing}}}{g}, \text{ for } \mathsf{R}^{\mathsf{s}\to\mathsf{r}}_{\text{routing}} := \frac{\mathsf{K}\max\left\{1-(\mathsf{r}\mathsf{M}^{\text{relay}}+\mathsf{M}^{\text{user}})/\mathsf{N},0\right\}}{\mathsf{H}} \tag{9.2}$$

where $\mathsf{R}^{\mathsf{s}\to\mathsf{r}}_{\text{routing}}$ is achieved by routing. It can be seen that when $\mathsf{M}^{\text{relay}}=0$, since the max link-load from relays to users can not be larger than the max link-load from the server to relays, the coded caching gain on the the max link-load from the server to relays is equivalent to the one on the max link-load among all links. Thus the definition of coded caching gain in (9.2) is equivalent to the one given in (5.14) for combination networks with end-user-caches.

We start by give our main results on combination networks with both cache relays and users.

Theorem 24. *In a* $(\mathsf{H},\mathsf{r},\mathsf{M}^{relay},\mathsf{M}^{user},\mathsf{N})$ *combination network, the lower convex envelop of the following four groups of corner points is achievable,*

1. *Group 1:* $(\mathsf{M}^{relay},\mathsf{M}^{user}) = \left(0, \frac{\mathsf{N}\mathsf{k}_{1,b}}{\mathsf{k}_{1,b}+\mathsf{k}_{2,b}}\right)$ *for each* $g\in[1:\mathsf{K}_1]$. *The achieved loads are*

$$(\mathsf{R}^{\mathsf{s}\to\mathsf{r}},\mathsf{R}^{\mathsf{r}\to\mathsf{u}}) = \big(\mathsf{K}(1-\mathsf{M}^{user}/\mathsf{N})/(\mathsf{H}g), (1-\mathsf{M}^{user}/\mathsf{N})/\mathsf{r}\big). \tag{9.3a}$$

2. *Group 2:* $(\mathsf{M}^{relay}, \mathsf{M}^{user}) = \left(\frac{\mathsf{N}\binom{\mathsf{K}_1}{g-1}}{(\mathsf{k}_{1,b}+\mathsf{k}_{2,b}) \max\{y\in[\mathsf{r}]:\mathsf{K}_y \geq g-1\}}, \mathsf{N}\frac{\mathsf{k}_{1,b}}{(\mathsf{k}_{1,b}+\mathsf{k}_{2,b})} - \frac{\mathsf{N}\binom{\mathsf{K}_1-1}{g-2}\mathsf{r}}{(\mathsf{k}_{1,b}+\mathsf{k}_{2,b}) \max\{y\in[\mathsf{r}]:\mathsf{K}_y \geq g-1\}} \right)$ *for each* $g \in [2:\mathsf{K}_2+1]$. *The achieved loads are*

$$\mathsf{R}^{\mathsf{s}\to\mathsf{r}} = \frac{\mathsf{K}\big(1-(\mathsf{r}\mathsf{M}^{relay}+\mathsf{M}^{user})/\mathsf{N}\big)}{\mathsf{H}g}, \tag{9.3b}$$

$$\mathsf{R}^{\mathsf{r}\to\mathsf{u}} = \frac{(1-\mathsf{M}^{user}/\mathsf{N})}{\mathsf{r}} + \frac{\sum_{a=2}^{\mathsf{r}} L_{g-1,a}\big(\mathsf{K}-\binom{\mathsf{H}-a}{\mathsf{r}}-\mathsf{K}_1\big)}{(\mathsf{k}_{1,b}+\mathsf{k}_{2,b}) \max\{y\in[\mathsf{r}]:\mathsf{K}_y\geq g-1\}}, \tag{9.3c}$$

$$L_{t,a} := |\{\mathcal{W}\subseteq\mathcal{U}_h : |\mathcal{W}|=t, k\in\mathcal{W}, |\mathcal{R}_{\mathcal{W}}|=a\}|, \text{ for any relay } h \text{ and any user } k\in\mathcal{U}_h. \tag{9.3d}$$

3. *Group 3:* $(\mathsf{M}^{relay}, \mathsf{M}^{user}) = \left(\frac{\mathsf{N}\binom{\mathsf{K}_1}{g-1}}{(\mathsf{k}_{1,b}+\mathsf{k}_{2,b})\mathsf{m}^g_{\max}}, \mathsf{N}\frac{\mathsf{k}_{1,b}}{(\mathsf{k}_{1,b}+\mathsf{k}_{2,b})} - \frac{\mathsf{N}\binom{\mathsf{K}_1-1}{g-2}\mathsf{r}}{(\mathsf{k}_{1,b}+\mathsf{k}_{2,b})\mathsf{m}^g_{\max}} \right)$ *for* $g\in[\mathsf{K}_2+2:\mathsf{K}_1]$ *and* $\mathsf{m}^g_{\max} := \max\left\{m\in[\mathsf{H}-\mathsf{r}+1] : \mathsf{K}_1 - \binom{\mathsf{H}-m}{\mathsf{r}-1} \leq g-1\right\}$. *The achieved loads are*

$$\mathsf{R}^{\mathsf{s}\to\mathsf{r}} = \frac{\mathsf{K}\big(1-(\mathsf{r}\mathsf{M}^{relay}+\mathsf{M}^{user})/\mathsf{N}\big)}{\mathsf{H}g}, \tag{9.3e}$$

$$\mathsf{R}^{\mathsf{r}\to\mathsf{u}} = \frac{(1-\mathsf{M}^{user}/\mathsf{N})}{\mathsf{r}} + \frac{\sum_{a=2}^{\mathsf{r}} L'_{g-1,a}(a-1)(\mathsf{K}_1-g+1)}{(\mathsf{k}_{1,b}+\mathsf{k}_{2,b})\mathsf{m}^g_{\max}}, \tag{9.3f}$$

$$L'_{t,a} := |\{\mathcal{W}\subseteq\mathcal{U}_h : |\mathcal{W}|=t, k\in\mathcal{W}, |\{h_1\in\mathcal{H}_k\setminus\{h\} : \mathcal{U}_{\{h,h_1\}}\subseteq\mathcal{W}\}|=a-1\}|,$$
$$\text{for any relay } h \text{ and any user } k\in\mathcal{U}_h. \tag{9.3g}$$

4. *Group 4:* $(\mathsf{M}^{relay}, \mathsf{M}^{user}) = (\mathsf{N}/\mathsf{r}, \mathsf{N}t_1/\mathsf{K}_1)$ *for* $t_1 \in \{0, \mathsf{K}_1\}$ *and* $(\mathsf{M}^{relay}, \mathsf{M}^{user}) = (0, \mathsf{N})$. *If* $t_1 = 0$, $(\mathsf{R}^{\mathsf{s}\to\mathsf{r}}, \mathsf{R}^{\mathsf{r}\to\mathsf{u}}) = (0, 1/\mathsf{r})$. *If* $t_1 = \mathsf{K}_1$, $(\mathsf{R}^{\mathsf{s}\to\mathsf{r}}, \mathsf{R}^{\mathsf{r}\to\mathsf{u}}) = (0,0)$. *If* $(\mathsf{M}^{relay}, \mathsf{M}^{user}) = (0, \mathsf{N})$, $(\mathsf{R}^{\mathsf{s}\to\mathsf{r}}, \mathsf{R}^{\mathsf{r}\to\mathsf{u}}) = (0,0)$.

Compared to the caching scheme in [50], we have the following corollary.

Corollary 2. *The achievable memory-load tradeoff region for* $(\mathsf{H}, \mathsf{r}, \mathsf{N})$ *combination network from the proposed scheme in Theorem 24 is strictly larger than the one of [50].*

Compared to the cut-set converse bound in Theorem 23, we can derive the following optimality result on the max link-load from the server to relays whose proof is in Appendix A.16.

Theorem 25. *In a* $(\mathsf{H}, \mathsf{r}, \mathsf{M}^{relay}, \mathsf{M}^{user}, \mathsf{N})$ *combination network, if the memory pair* $(\mathsf{M}^{relay}, \mathsf{M}^{user})$ *can be obtained by memory-sharing among the following four memory pairs,* $\left(0, \frac{(\mathsf{K}-\mathsf{H}+\mathsf{r}-1)\mathsf{N}}{\mathsf{K}}\right)$, $\left(\frac{\mathsf{N}}{\mathsf{H}}, \frac{(\mathsf{K}-\mathsf{H}+\mathsf{r}-1)\mathsf{N}}{\mathsf{K}} - \frac{\mathsf{N}(\mathsf{K}_1-1)\mathsf{r}}{(\mathsf{H}-\mathsf{r}+1)\binom{\mathsf{H}}{\mathsf{r}-1}}\right)$, $(\mathsf{N}/\mathsf{r}, 0)$ *and* $(0, \mathsf{N})$, *the optimal max link-load from the server to relays is*

$$\mathsf{R}^{\mathsf{s}\to\mathsf{r}\star} = \frac{1-(\mathsf{r}\mathsf{M}^{relay}+\mathsf{M}^{user})/\mathsf{N}}{\mathsf{r}}, \tag{9.4}$$

achieved by the proposed scheme in Theorem 24.

From a similar proof of Theorem 25, we can derive the following optimality result on the load pair $(\mathsf{R}^{\mathsf{s}\to\mathsf{r}}, \mathsf{R}^{\mathsf{r}\to\mathsf{u}})$.

Theorem 26. *In a* $(\mathsf{H},\mathsf{r},\mathsf{M}^{relay},\mathsf{M}^{user},\mathsf{N})$ *combination network, if the memory pair* $(\mathsf{M}^{relay},\mathsf{M}^{user})$ *can be obtained by memory-sharing among the following three memory pairs,* $\left(0,\frac{(\mathsf{K}-\mathsf{H}+\mathsf{r}-1)\mathsf{N}}{\mathsf{K}}\right)$, $(\mathsf{N}/\mathsf{r},0)$ *and* $(0,\mathsf{N})$*, the optimal max link-load pair is*

$$\mathsf{R}^{\mathsf{s}\to\mathsf{r}\star}=\frac{1-(\mathsf{r}\mathsf{M}^{relay}+\mathsf{M}^{user})/\mathsf{N}}{\mathsf{r}}, \tag{9.5a}$$

$$\mathsf{R}^{\mathsf{r}\to\mathsf{u}\star}=\frac{1-\mathsf{M}^{user}/\mathsf{N}}{\mathsf{r}}. \tag{9.5b}$$

achieved by the proposed scheme in Theorem 24.

For combination networks with cache-aided relays and users, we first revise the caching scheme in [50] which divides each file into two parts and the packets transmitted from the server are independent of the cached contents of relays. The memories-loads tradeoff of the scheme in [50] is the lower convex envelope of the two groups of points.

1. $(\mathsf{M}^{\text{relay}},\mathsf{M}^{\text{user}})=(0,\mathsf{N}(g-1)/\mathsf{K}_1)$ where $g\in[1:\mathsf{K}_1]$. For each point in this group, we can see that relays do not have memory and the scheme is equivalent to the one for combination networks with end-user-caches described in Chapter 5.3. So $(\mathsf{R}^{\mathsf{s}\to\mathsf{r}},\mathsf{R}^{\mathsf{r}\to\mathsf{u}})=\left(\frac{\mathsf{K}(1-\mathsf{M}^{\text{user}})/\mathsf{N}}{\mathsf{H}g},\frac{1-\mathsf{M}^{\text{user}}/\mathsf{N}}{\mathsf{r}}\right)$.

2. $(\mathsf{M}^{\text{relay}},\mathsf{M}^{\text{user}})=(\mathsf{N}/\mathsf{r},\mathsf{N}t_1/\mathsf{K}_1)$ where $t_1\in\{0,\mathsf{K}_1\}$ and $(\mathsf{M}^{\text{relay}},\mathsf{M}^{\text{user}})=(0,\mathsf{N})$. In this case, each relay directly caches s_i^h such that the server needs not to transmit any packets to relays.

   If $t_1=0$, each user does not cache any bits. In the delivery phase, each relay $h\in[\mathsf{H}]$ transmits $s_{d_k}^h$ to each user $k\in\mathcal{U}_h$. So we have $(\mathsf{R}^{\mathsf{s}\to\mathsf{r}},\mathsf{R}^{\mathsf{r}\to\mathsf{u}})=(0,1/\mathsf{r})$.

   If $t_1=\mathsf{K}_1$, in the placement phase, each user k caches s_i^h for $h\in[\mathsf{H}]$ and $i\in[\mathsf{N}]$. So it caches all the N files and $(\mathsf{R}^{\mathsf{s}\to\mathsf{r}},\mathsf{R}^{\mathsf{r}\to\mathsf{u}})=(0,0)$.

   If $(\mathsf{M}^{\text{relay}},\mathsf{M}^{\text{user}})=(0,\mathsf{N})$, we also have $(\mathsf{R}^{\mathsf{s}\to\mathsf{r}},\mathsf{R}^{\mathsf{r}\to\mathsf{u}})=(0,0)$.

Compared to the converse bound in Theorem 24, it is obvious that the max link-load from relays to users achieved by the above scheme is optimal. To reduce the max link-load from the server to relays, we can directly replace the used caching scheme for combination networks with cache-end-users in [50] by our proposed asymmetric coded placement scheme in Chapter 7.4, which corresponds to Group 1 in Theorem 24. So we can reduce $\mathsf{R}^{\mathsf{s}\to\mathsf{r}}$ without increasing $\mathsf{R}^{\mathsf{r}\to\mathsf{u}}$. Hence we can prove Corollary 2.

Moreover, we propose a novel caching scheme, in which the users can leverage the cached contents of the connected relays to decode the coded messages transmitted from the server, to further reduce $\mathsf{R}^{\mathsf{s}\to\mathsf{r}}$. In the rest of this subsection, we present our proposed scheme to achieve Theorem 24. We start by an example to illustrate the main idea of the proposed scheme.

Example 12 ($\mathsf{H}=5$, $\mathsf{r}=3$, $\mathsf{N}=10$, $\mathsf{M}^{\text{relay}}=25/12$ and $\mathsf{M}^{\text{user}}=5/12$, $g=3$)**.** In this example, we have

$$\mathcal{U}_1=[6],\ \mathcal{U}_2=\{1,2,3,7,8,9\},\ \mathcal{U}_3=\{1,4,5,7,8,10\},\ \mathcal{U}_4=\{2,4,6,7,9,10\},\ \mathcal{U}_5=\{3,5,6,8,9,10\}.$$

To achieve $g=3$, we impose that each demanded subfile of each user which is neither stored in its memory nor in the memories of its connected relays, is transmitted from the server in one linear combination including other $g-1=2$ subfiles.

Placement phase As the asymmetric coded placement scheme in Chapter 7.4, we divide each F_i into $\mathsf{k}_{1,\mathrm{b}} + \mathsf{k}_{2,\mathrm{b}} = 36$ non-overlapping and equal-length pieces, which are then encoded by $(\mathsf{n}_\mathrm{b}, \mathsf{k}_{1,\mathrm{b}} + \mathsf{k}_{2,\mathrm{b}}) = (45, 36)$ MDS code. The length is each MDS symbol is $\mathsf{B}/36$. For each $\mathcal{W} \in \mathcal{Z}_g$, there is one MDS symbol denoted by $f_{i,\mathcal{W}}$. However, different to the asymmetric coded placement scheme, not the whole symbol $f_{i,\mathcal{W}}$ is stored in the cache of each user in $\mathcal{W}$. Instead, we divide $f_{i,\mathcal{W}}$ into $|\mathcal{R}_\mathcal{W}| + 1$ non-overlapping parts (but not necessary with identical length), $f_{i,\mathcal{W}} = \{f_{i,\mathcal{W},h} : h \in \mathcal{R}_\mathcal{W}\} \cup \{f'_{i,\mathcal{W}}\}$. For each $h \in \mathcal{R}_\mathcal{W}$, $f_{i,\mathcal{W},h}$ is cached by relay h where $|f_{i,\mathcal{W},h}| = \frac{\mathsf{M}^{\text{relay}}\mathsf{B}}{\mathsf{N}\binom{\mathsf{K}_1}{g-1}}$. In addition, $f'_{i,\mathcal{W}}$ is cached by each user in $\mathcal{W}$ where $|f'_{i,\mathcal{W}}| = \mathsf{B}/36 - \frac{|\mathcal{R}_\mathcal{W}|\mathsf{M}^{\text{relay}}\mathsf{B}}{\mathsf{N}\binom{\mathsf{K}_1}{g-1}}$. Hence, each relay $h \in [\mathsf{H}]$ caches $f_{i,\mathcal{W},h}$ for each $\mathcal{W} \subseteq \mathcal{U}_h$ and each $i \in [\mathsf{N}]$ where $|\mathcal{W}| = g - 1$. Thus the number of cached bits of relay h is

$$\mathsf{N}\binom{\mathsf{K}_1}{g-1}\frac{\mathsf{M}^{\text{relay}}\mathsf{B}}{\mathsf{N}\binom{\mathsf{K}_1}{g-1}} = \mathsf{M}^{\text{relay}}\mathsf{B}. \tag{9.6}$$

For example, consider $f_{i,\{1,2\}}$ which is divided into $|\mathcal{R}_{\{1,2\}}| + 1 = 3$ non-overlapping pieces. Each relay $h \in \mathcal{R}_{\{1,2\}} = \{1,2\}$ caches $f_{i,\{1,2\},h}$ with $\frac{\mathsf{M}^{\text{relay}}\mathsf{B}}{\mathsf{N}\binom{\mathsf{K}_1}{g-1}} = \mathsf{B}/72$ bits. So for the last piece, we have $|f'_{i,\{1,2\}}| = \mathsf{B}/36 - 2\mathsf{B}/72 = 0$ and thus no user caches any bits of $f_{i,\{1,2\}}$.

Consider now $f_{i,\{1,6\}}$ which is divided into $|\mathcal{R}_{\{1,6\}}| + 1 = 2$ non-overlapping pieces. Each relay $h \in \mathcal{R}_{\{1,6\}} = \{1\}$ caches $f_{i,\{1,6\},h}$ with $\frac{\mathsf{M}^{\text{relay}}\mathsf{B}}{\mathsf{N}\binom{\mathsf{K}_1}{g-1}} = \mathsf{B}/72$ bits. So $|f'_{i,\{1,6\}}| = \mathsf{B}/36 - \mathsf{B}/72 = \mathsf{B}/72$. Thus each user in $\{1,6\}$ caches $f'_{i,\{1,6\}}$ with $\mathsf{B}/72$ bits.

We then focus on user 1. For each set $\mathcal{W} \in \big\{\{1,2\},\{1,3\},\{1,4\},\{1,5\},\{1,7\},\{1,8\}\big\}$, we have $|\mathcal{R}_\mathcal{W}| = 2$ and $|f'_{i,\mathcal{W}}| = 0$. For each set $\mathcal{W} \in \big\{\{1,6\},\{1,9\},\{1,10\}\big\}$, we have $|\mathcal{R}_\mathcal{W}| = 1$ and $|f'_{i,\mathcal{W}}| = \mathsf{B}/72$. So user 1 caches $3\mathsf{NB}/72 = \mathsf{M}^{\text{user}}\mathsf{B}$ bits.

Delivery phase We let each user k recover $f_{d_k,\mathcal{W}}$ where $\mathcal{W} \in \mathcal{Z}_{g-1}$ and $\mathcal{R}_{\{k\}\cup\mathcal{W}} \neq \emptyset$. There are three steps in delivery phase:

1. In the first step, for each relay $h \in [\mathsf{H}]$ and each user $k \in \mathcal{U}_h$, relay h delivers all the cached bits of F_{d_k} to user k. More precisely, for each set $\mathcal{W} \subseteq \mathcal{U}_h$ where $|\mathcal{W}| = g - 1$, relay h delivers $f_{d_k,\mathcal{W},h}$ to user k. So by this step and the placement phase, each user $k \in [\mathsf{K}]$ can recover $f_{d_k,\mathcal{W}}$ where $\mathcal{W} \in \mathcal{Z}_{g-1}$ and $k \in \mathcal{W}$. User k can also recover $f_{d_k,\mathcal{W},h}$ where $\mathcal{W} \in \mathcal{Z}_{g-1}$, $k \notin \mathcal{W}$, $\mathcal{R}_{\{k\}\cup\mathcal{W}} \neq \emptyset$ and $h \in (\mathcal{R}_\mathcal{W} \cap \mathcal{H}_k)$.

2. In the second step, we also focus on each relay $h \in [\mathsf{H}]$ and each user $k \in \mathcal{U}_h$. For each set $\mathcal{W}' \subseteq \mathcal{U}_h$ and each $k' \subseteq [\mathsf{K}] \setminus \mathcal{U}_h$ where $|\mathcal{W}'| = g - 1$, $k \in \mathcal{W}'$ and $\mathcal{R}_{\{k'\}\cup\mathcal{W}'} \neq \emptyset$, relay h delivers $f_{d_{k'},\mathcal{W}',h}$ to user k. These additional side information of user k will help him decode the multicast messages transmitted from the server in the second step.

3. In the last step, as Example 12, we let each user k recover $f_{d_k,\mathcal{W}} \setminus \{f_{d_k,\mathcal{W},h} : h \in (\mathcal{R}_\mathcal{W} \cap \mathcal{H}_k)\}$ where $\mathcal{W} \in \mathcal{Z}_3$, $k \notin \mathcal{W}$ and $\mathcal{R}_{\{k\}\cup\mathcal{W}} \neq \emptyset$. More precisely, we let

$$\mathcal{T}_{k,\mathcal{W}} := f_{d_k,\mathcal{W}} \setminus \{f_{d_k,\mathcal{W},h} : h \in (\mathcal{R}_\mathcal{W} \cap \mathcal{H}_k)\} \tag{9.7}$$

representing the unknown bits in $f_{d_k,\mathcal{W}}$ of user k. We divide $\mathcal{T}_{k,\mathcal{W}}$ into $|\mathcal{R}_{\{k\}\cup\mathcal{W}}|$ non-overlapping and equal-length pieces, $\mathcal{T}_{k,\mathcal{W}} = \{\mathcal{T}_{k,\mathcal{W},h} : h \in \mathcal{R}_{\{k\}\cup\mathcal{W}}\}$. After considering all the MDS coded symbols demanded by all the users, for each relay $h \in [\mathsf{H}]$ and each set $\mathcal{J} \subseteq \mathcal{U}_h$ where $|\mathcal{J}| = g$, we create the multicast message

$$V^h_{\mathcal{J}} := \underset{k\in\mathcal{J}}{\oplus} \mathcal{T}_{k,\mathcal{J}\setminus\{k\},h} \tag{9.8}$$

to be sent to relay h and then forwarded to the users in $\mathcal{J}$.

For example, consider relay 1 and set $\mathcal{J}_1 = \{1,2,3\}$. It can be seen that $\mathcal{T}_{1,\{2,3\}} = f_{d_1,\{2,3\}} \setminus (f_{d_1,\{2,3\},1} \cup f_{d_1,\{2,3\},2}) = \emptyset$. Similarly $\mathcal{T}_{2,\{1,3\}} = \mathcal{G}_{3,\{1,2\}} = \emptyset$. So $V^1_{\{1,2,3\}} = \emptyset$.

Consider now relay 1 and set $\mathcal{J}_2 = \{1,2,4\}$. It can be seen that $\mathcal{T}_{1,\{2,4\}} = f_{d_1,\{2,4\}} \setminus f_{d_1,\{2,4\},1} = f_{d_1,\{2,4\},4}$. Since $\mathcal{R}_{\{1,2,4\}} = \{1\}$, we don't further partition $\mathcal{T}_{1,\{2,4\}}$ which includes $\mathsf{B}/72$ bits. Similarly, each of $\mathcal{T}_{2,\{1,4\}}$ and $\mathcal{T}_{4,\{1,2\}}$ has $\mathsf{B}/72$ bits. Hence, $V^1_{\{1,2,4\}} = \mathcal{T}_{1,\{2,4\}} \oplus \mathcal{T}_{2,\{1,4\}} \oplus \mathcal{T}_{4,\{1,2\}}$ including $\mathsf{B}/72$ is transmitted from the server to relay 1, which then forwards it to users in $\{1,2,4\}$.

Hence, we achieve $g = 3$ and $(\mathsf{R}^{\mathsf{s}\to\mathsf{r}}, \mathsf{R}^{\mathsf{r}\to\mathsf{u}}) = (\frac{2}{9}, \frac{41}{72}) \approx (0.22, 0.57)$ while the scheme in [50] described in Chapter 5.3 gives $(\mathsf{R}^{\mathsf{s}\to\mathsf{r}}, \mathsf{R}^{\mathsf{r}\to\mathsf{u}}) = (\frac{11}{24}, \frac{23}{72}) \approx (0.46, 0.32)$. It can be seen the max link-load from the server to relays achieved by the proposed method is less than the half of the one achieved by the scheme in [50].

Comparing the proposed scheme and the scheme in [50], there are main two advantages. On one hand, we can see that the cached contents of relays help users to decode the packets transmitted from the server which can lead an additional coded caching gain. For example, $f_{d_7,\{1,2\},1}$ is cached by relay 1 and $f_{d_2,\{1,7\},3}$ is cached by relay 3. In the second step of delivery, $f_{d_7,\{1,2\},1}$ is transmitted from relay 1 to user 1 and $f_{d_2,\{1,7\},3}$ is transmitted from relay 3 to user 1 such that user 1 knows them. In the third step of delivery, the server transmit $f_{d_1,\{2,7\},4} \oplus f_{d_2,\{1,7\},3} \oplus f_{d_7,\{1,2\},1}$ to relay 2 and user 1 can use $f_{d_7,\{1,2\},1}$ and $f_{d_2,\{1,7\},3}$ to decode $f_{d_1,\{2,7\},4}$. On the other hand, our proposed scheme is based on the asymmetric coded placement scheme in Chapter 7.4 which is proved to be better than the scheme in [50]. □

We now generalize the proposed scheme in Example 12. Notice that in this example, $f_{i,\{1,2\}}$ with $\mathsf{B}/36$ bits is divided into $|\mathcal{R}_{\{1,2\}}|+1 = 3$ non-overlapping pieces where $|f_{i,\{1,2\},1}| = |f_{i,\{1,2\},2}| = \frac{\mathsf{M}^{\text{relay}}\mathsf{B}}{\mathsf{N}\binom{\mathsf{K}_1}{g-1}} = \mathsf{B}/72$ bits and $|f'_{i,\{1,2\}}| = 0$. It can be seen that if we increase $\mathsf{M}^{\text{relay}}$ by a small value and we still desire to achieve $g = 3$, we have $|f_{i,\{1,2\},1}| + |f_{i,\{1,2\},2}| > f_{i,\{1,2\}}$ and thus these two pieces are overlapped which leads to redundancy. In other words, not all the bits of F_{d_1} cached in relays 1 and 2 are useful to user 1, from the proposed scheme in Example 12, we cannot achieve the coded caching gain g compared to the routing scheme in which all the bits of F_{d_k} cached from $h \in \mathcal{H}_k$ is useful to user k. So in this thesis, we only consider the case

$$\mathsf{M}^{\text{relay}} \le \frac{\mathsf{N}\binom{\mathsf{K}_1}{g-1}}{\max_{\mathcal{W}\in\mathcal{Z}_g}|\mathcal{R}_{\mathcal{W}}|} = \frac{\mathsf{N}\binom{\mathsf{K}_1}{g-1}}{(\mathsf{k}_{1,\mathrm{b}}+\mathsf{k}_{2,\mathrm{b}})\max\{y\in[\mathsf{r}]:\mathsf{K}_y\ge g-1\}}, \tag{9.9}$$

where $\mathsf{B}/(\mathsf{k}_{1,\mathrm{b}}+\mathsf{k}_{2,\mathrm{b}})$ is the length of each MDS symbol generated by the asymmetric coded placement scheme in Chapter 7.4. The memories-loads tradeoff of the proposed scheme is the lower convex envelope of the four groups of corner points.

1. Group 1: $(\mathsf{M}^{\text{relay}}, \mathsf{M}^{\text{user}}) = \left(0, \frac{\mathsf{N}\mathsf{k}_{1,\text{b}}}{\mathsf{k}_{1,\text{b}}+\mathsf{k}_{2,\text{b}}}\right)$ for each $g \in [1 : \mathsf{K}_1]$. For each point in this group, we can see that relays do not have memory and the scheme is equivalent to the one for combination networks with end-user-caches. We directly use the asymmetric coded placement scheme in Chapter 7.4 which leads $(\mathsf{R}^{\text{s}\to\text{r}}, \mathsf{R}^{\text{r}\to\text{u}}) = \big(\mathsf{K}(1-\mathsf{M}^{\text{user}}/\mathsf{N})/(\mathsf{H}g), (1-\mathsf{M}^{\text{user}}/\mathsf{N})/\mathsf{r}\big)$.

2. Group 2: $(\mathsf{M}^{\text{relay}}, \mathsf{M}^{\text{user}}) = \left(\frac{\mathsf{N}\binom{\mathsf{K}_1}{g-1}}{(\mathsf{k}_{1,\text{b}}+\mathsf{k}_{2,\text{b}})\max\{y\in[\mathsf{r}]:\mathsf{K}_y\geq g-1\}}, \mathsf{N}\frac{\mathsf{k}_{1,\text{b}}}{(\mathsf{k}_{1,\text{b}}+\mathsf{k}_{2,\text{b}})} - \frac{\mathsf{N}\binom{\mathsf{K}_1-1}{g-2}\mathsf{r}}{(\mathsf{k}_{1,\text{b}}+\mathsf{k}_{2,\text{b}})\max\{y\in[\mathsf{r}]:\mathsf{K}_y\geq g-1\}}\right)$ where the coded caching gain $g \in [2 : \mathsf{K}_2 + 1]$ and the proposed scheme is based on the asymmetric coded placement scheme in Chapter 7.4.1.

Placement phase We divide each file F_i where $i \in [\mathsf{N}]$ into $\mathsf{k}_{1,\text{b}} + \mathsf{k}_{2,\text{b}}$ non-overlapping and equal-length pieces, which are then encoded by $(|\mathcal{Z}_{g-1}|, \mathsf{k}_{1,\text{b}} + \mathsf{k}_{2,\text{b}})$ MDS code. The length of each MDS symbol is $\mathsf{B}/(\mathsf{k}_{1,\text{b}} + \mathsf{k}_{2,\text{b}})$. For each $\mathcal{W} \in \mathcal{Z}_{g-1}$ and each file $i \in [\mathsf{N}]$, there is one MDS symbol denoted by $f_{i,\mathcal{W}}$ and we divide $f_{i,\mathcal{W}}$ into $|\mathcal{R}_{\mathcal{W}}| + 1$ non-overlapping parts, $f_{i,\mathcal{W}} = \{f_{i,\mathcal{W},h} : h \in \mathcal{R}_{\mathcal{W}}\} \cup \{f'_{i,\mathcal{W}}\}$. For each $h \in \mathcal{R}_{\mathcal{W}}$, $f_{i,\mathcal{W},h}$ is cached by relay h where

$$|f_{i,\mathcal{W},h}| = \frac{\mathsf{M}^{\text{relay}}\mathsf{B}}{\mathsf{N}\binom{\mathsf{K}_1}{g-1}} \tag{9.10a}$$

$$= \frac{\mathsf{B}}{\max\{y \in [\mathsf{r}] : \mathsf{K}_y \geq g-1\}(\mathsf{k}_{1,\text{b}} + \mathsf{k}_{2,\text{b}})}. \tag{9.10b}$$

In addition, $f'_{i,\mathcal{W}}$ is cached by each user in $\mathcal{W}$ where

$$|f'_{i,\mathcal{W}}| = \mathsf{B}/(\mathsf{k}_{1,\text{b}} + \mathsf{k}_{2,\text{b}}) - \frac{|\mathcal{R}_{\mathcal{W}}|\mathsf{M}^{\text{relay}}\mathsf{B}}{\mathsf{N}\binom{\mathsf{K}_1}{g-1}} \tag{9.11a}$$

$$= \mathsf{B}/(\mathsf{k}_{1,\text{b}} + \mathsf{k}_{2,\text{b}}) - \frac{|\mathcal{R}_{\mathcal{W}}|\mathsf{B}}{(\mathsf{k}_{1,\text{b}} + \mathsf{k}_{2,\text{b}})\max\{y \in [\mathsf{r}] : \mathsf{K}_y \geq g-1\}}. \tag{9.11b}$$

Hence, each user $k \in [\mathsf{K}]$ totally caches $\sum_{i\in[\mathsf{N}]}\sum_{\mathcal{W}\in\mathcal{Z}_{g-1}:k\in\mathcal{W}} |f'_{i,\mathcal{W}}| = \mathsf{M}^{\text{user}}\mathsf{B}$ bits. Notice that for each $\mathcal{W} \in \mathcal{Z}_{g-1}$, if $\mathcal{W} \in \arg\max_{\mathcal{W}'\in\mathcal{Z}_g} |\mathcal{R}_{\mathcal{W}'}|$, we have $|f'_{i,\mathcal{W}}| = 0$; otherwise, $|f'_{i,\mathcal{W}}| > 0$.

Delivery phase We let each user k recover $f_{d_k,\mathcal{J}\setminus\{k\}}$ where $\mathcal{J} \in \mathcal{Z}_g$ and $k \in \mathcal{J}$. There are three steps in delivery phase:

(a) For each relay $h \in [\mathsf{H}]$ and each user $k \in \mathcal{U}_h$, relay h delivers all the cached bits of F_{d_k} to user k. More precisely, for each set $\mathcal{W} \subseteq \mathcal{U}_h$ where $|\mathcal{W}| = g - 1$, relay h delivers $f_{d_k,\mathcal{W},h}$ to user k.

(b) For each set $\mathcal{W} \subseteq \mathcal{U}_h$ and each user $k \in \mathcal{W}$, where $|\mathcal{W}| = g-1$ and $|\mathcal{R}_{\mathcal{W}}| > 1$, relay h delivers $f_{d_{k'},\mathcal{W},h}$ to user k where $k' \subseteq ([\mathsf{K}] \setminus \mathcal{U}_h)$ and $\mathcal{R}_{\{k'\}\cup\mathcal{W}} \neq \emptyset$.

(c) In the last step, we let each user k recover $f_{d_k,\mathcal{W}}\setminus\{f_{d_k,\mathcal{W},h} : h \in (\mathcal{R}_{\mathcal{W}}\cap\mathcal{H}_k)\}$ where $\mathcal{W} \in \mathcal{Z}_{g-1}$, $k \notin \mathcal{W}$ and $\mathcal{R}_{\{k\}\cup\mathcal{W}} \neq \emptyset$. More precisely, we let

$$\mathcal{T}_{k,\mathcal{W}} := f_{d_k,\mathcal{W}} \setminus \{f_{d_k,\mathcal{W},h} : h \in (\mathcal{R}_{\mathcal{W}} \cap \mathcal{H}_k)\} \tag{9.12}$$

representing the unknown bits in $f_{d_k,\mathcal{W}}$ of user k. We divide $\mathcal{T}_{k,\mathcal{W}}$ into $|\mathcal{R}_{\{k\}\cup\mathcal{W}}|$ non-overlapping and equal-length pieces, $\mathcal{T}_{k,\mathcal{W}} = \{\mathcal{T}_{k,\mathcal{W},h} : h \in \mathcal{R}_{\{k\}\cup\mathcal{W}}\}$. After considering all the MDS coded symbols demanded by all the users, for each relay $h \in [\mathsf{H}]$ and each set $\mathcal{J} \subseteq \mathcal{U}_h$ where $|\mathcal{J}| = g$, we create the multicast message in (9.8), which is to be sent to relay h and then forwarded to the users in $\mathcal{J}$. It is also easily to check that each subfile in the multicast message in (9.8) has the same length.

In Appendix A.15, we compute the achieved loads are given in (9.3c).

3. Group 3: $(\mathsf{M}^{\text{relay}}, \mathsf{M}^{\text{user}}) = \left(\frac{\mathsf{N}\binom{\mathsf{K}_1}{g-1}}{(\mathsf{k}_{1,\mathrm{b}}+\mathsf{k}_{2,\mathrm{b}})\mathsf{m}^g_{\max}}, \mathsf{N}\frac{\mathsf{k}_{1,\mathrm{b}}}{(\mathsf{k}_{1,\mathrm{b}}+\mathsf{k}_{2,\mathrm{b}})} - \frac{\mathsf{N}\binom{\mathsf{K}_1-1}{g-2}\mathsf{r}}{(\mathsf{k}_{1,\mathrm{b}}+\mathsf{k}_{2,\mathrm{b}})\mathsf{m}^g_{\max}}\right)$ where the coded caching gain $g \in [\mathsf{K}_2+2 : \mathsf{K}_1]$ and $\mathsf{m}^g_{\max} := \max\left\{m \in [\mathsf{H}-\mathsf{r}+1] : \mathsf{K}_1 - \binom{\mathsf{H}-m}{\mathsf{r}-1} \le g-1\right\}$ representing the max value of $m \in [\mathsf{H}-\mathsf{r}+1]$ such that there exist some created MDS subfiles in Hierarchy m (i.e., $N_m > 0$). We use the similar way as the above case to extend the asymmetric coded placement scheme in Chapter 7.4.2.

Placement phase We divide each file F_i where $i \in [\mathsf{N}]$ into $\mathsf{k}_{1,\mathrm{b}} + \mathsf{k}_{2,\mathrm{b}}$ non-overlapping and equal-length pieces, which are then encoded by $(\mathsf{n}_{\mathrm{b}}, \mathsf{k}_{1,\mathrm{b}}+\mathsf{k}_{2,\mathrm{b}})$ MDS code. The length of each MDS symbol is $\mathsf{B}/(\mathsf{k}_{1,\mathrm{b}} + \mathsf{k}_{2,\mathrm{b}})$. Each MDS symbol represents one MDS subfile in Hierarchy $m \in [\mathsf{H}-\mathsf{r}+1]$. For each MDS subfile $f_{i,\mathcal{W}'}$ in Hierarchy m which is created from the multicasting opportunities among m relays (assumed to be $\mathcal{Y}$), we divide it into $m+1$ non-overlapping pieces, $f_{i,\mathcal{W}'} = \{f_{i,\mathcal{W}',h} : h \in \mathcal{Y}\} \cup \{f'_{i,\mathcal{W}'}\}$. For each $h \in \mathcal{Y}$, $f_{i,\mathcal{W}',h}$ is cached by relay h where

$$|f_{i,\mathcal{W}',h}| = \frac{\mathsf{M}^{\text{relay}}\mathsf{B}}{\mathsf{N}\binom{\mathsf{K}_1}{g-1}} \tag{9.13a}$$

$$= \frac{\mathsf{B}}{\mathsf{m}^g_{\max}(\mathsf{k}_{1,\mathrm{b}} + \mathsf{k}_{2,\mathrm{b}})}. \tag{9.13b}$$

In addition, $f'_{i,\mathcal{W}'}$ is cached by each user in $\mathcal{W}'$ where

$$|f'_{i,\mathcal{W}'}| = \mathsf{B}/(\mathsf{k}_{1,\mathrm{b}} + \mathsf{k}_{2,\mathrm{b}}) - \frac{m\mathsf{M}^{\text{relay}}\mathsf{B}}{\mathsf{N}\binom{\mathsf{K}_1}{g-1}} \tag{9.14a}$$

$$= \mathsf{B}/(\mathsf{k}_{1,\mathrm{b}} + \mathsf{k}_{2,\mathrm{b}}) - \frac{m\mathsf{B}}{(\mathsf{k}_{1,\mathrm{b}} + \mathsf{k}_{2,\mathrm{b}})\mathsf{m}^g_{\max}}. \tag{9.14b}$$

Hence, each user $k \in [\mathsf{K}]$ totally caches $\frac{\mathsf{N}\mathsf{k}_{1,\mathrm{b}}}{(\mathsf{k}_{1,\mathrm{b}}+\mathsf{k}_{2,\mathrm{b}})} - \frac{\mathsf{N}\binom{\mathsf{K}_1-1}{g-2}\mathsf{r}}{(\mathsf{k}_{1,\mathrm{b}}+\mathsf{k}_{2,\mathrm{b}})\mathsf{m}^g_{\max}} = \mathsf{M}^{\text{user}}\mathsf{B}$ bits. Notice that if $m = \mathsf{m}^g_{\max}$, we have $|f'_{i,\mathcal{W}'}| = 0$; otherwise, $|f'_{i,\mathcal{W}'}| > 0$.

Delivery phase We let each user $k \in [\mathsf{K}]$ recover the created MDS subfile $f_{d_k,\mathcal{W}'}$ where $\mathcal{W}' \supseteq (\mathcal{J} \setminus \{k\})$ for each $\mathcal{J} \in \mathcal{Z}_g$ where $k \in \mathcal{J}$. There are three steps in delivery phase:

(a) For each relay $h \in [\mathsf{H}]$ and each user $k \in \mathcal{U}_h$, relay h delivers all the cached bits of F_{d_k} to user k. More precisely, for each set $\mathcal{W} \subseteq \mathcal{U}_h$ where $|\mathcal{W}| = g-1$, relay h delivers $f_{d_k,\mathcal{W}',h}$ to user k where $\mathcal{W}' \supseteq \mathcal{W}$.

(b) For each set $\mathcal{W} \subseteq \mathcal{U}_h$ and each user $k \in \mathcal{W}$ where $|\mathcal{W}| = g-1$, assuming the created subfile for each $i \in [\mathsf{N}]$ is $f_{i,\mathcal{W}'}$ by leveraging multicasting opportunities among relays in $\mathcal{Y}$ where $\mathcal{W}' \supseteq \mathcal{W}$, relay h delivers $f_{d_{k'},\mathcal{W}',h}$ to user k where $k' \in \big(\bigcup_{h_1 \in (\mathcal{Y}\setminus\{h\})} \mathcal{U}_{h_1}\big) \setminus \mathcal{W}'$.

(c) In the last step, we let user $k \in [\mathsf{K}]$ recover

$$\mathcal{P}_{k,\mathcal{J}\setminus\{k\}} := f_{d_k,\mathcal{W}'} \setminus \{f_{d_k,\mathcal{W}',h} : h \in (\mathcal{Y} \cap \mathcal{H}_k)\} \tag{9.15}$$

for each set $\mathcal{J} \in \mathcal{Z}_g$ where $k \in \mathcal{J}$ and we assume the created MDS subfile for $\mathcal{J}\setminus\{k\}$ is $f_{d_k,\mathcal{W}'}$ by leveraging multicasting opportunities among relays in $\mathcal{Y}$. More precisely, for each relay $h \in [\mathsf{H}]$ and each set $\mathcal{J} \subseteq \mathcal{U}_h$ where $|\mathcal{J}| = g$, we create the MAN-like multicast message $\underset{k\in\mathcal{J}}{\oplus} \mathcal{P}_{k,\mathcal{J}\setminus\{k\}}$ which is to be sent to relay h and then forwarded to the users in $\mathcal{J}$. It is also easily to check that each subfile in the multicast message in (9.8) has the same length.

In Appendix A.15, we compute the achieved loads are given in (9.3f).

4. Group 4: $(\mathsf{M}^{\text{relay}}, \mathsf{M}^{\text{user}}) = (\mathsf{N}/\mathsf{r}, \mathsf{N}t_1/\mathsf{K}_1)$ where $t_1 \in \{0, \mathsf{K}_1\}$ and $(\mathsf{M}^{\text{relay}}, \mathsf{M}^{\text{user}}) = (0, \mathsf{N})$. This group is equivalent to the second group of the caching scheme in [50] described at beginning of this subsection. So if $t_1 = 0$, $(\mathsf{R}^{\mathsf{s}\to\mathsf{r}}, \mathsf{R}^{\mathsf{r}\to\mathsf{u}}) = (0, 1/\mathsf{r})$. If $t_1 = \mathsf{K}_1$, $(\mathsf{R}^{\mathsf{s}\to\mathsf{r}}, \mathsf{R}^{\mathsf{r}\to\mathsf{u}}) = (0, 0)$. If $(\mathsf{M}^{\text{relay}}, \mathsf{M}^{\text{user}}) = (0, \mathsf{N})$, $(\mathsf{R}^{\mathsf{s}\to\mathsf{r}}, \mathsf{R}^{\mathsf{r}\to\mathsf{u}}) = (0, 0)$.

In the corner points of Groups 2 and 3, each user can leverage the cached contents of the connected relays to decode the coded messages transmitted from the server to reduce the load $\mathsf{R}^{\mathsf{s}\to\mathsf{r}}$. However, the tradeoff is that in the second step of delivery phase, each relay $h \in [\mathsf{H}]$ transmits some bits of the files which are not demanded by user $k \in \mathcal{U}_h$ to user k and thus we increase the load $\mathsf{R}^{\mathsf{r}\to\mathsf{u}}$. So if we do not want to increase $\mathsf{R}^{\mathsf{r}\to\mathsf{u}}$, we can directly take the corner points in Group 1 and 4.

Numerical Evaluations We compare the performance of the proposed scheme in Theorem 24 and the caching scheme in [50] for the combination network with cache-aided relays and users, where $\mathsf{H} = 6$, $\mathsf{r} = 2$ and $\mathsf{N} = \mathsf{K} = \binom{\mathsf{H}}{\mathsf{r}}$. We fix $\mathsf{M}^{\text{relay}} = 1$ and plot the tradeoffs between M^{user} and $(\mathsf{R}^{\mathsf{s}\to\mathsf{r}}, \mathsf{R}^{\mathsf{r}\to\mathsf{u}})$. We have two strategies:

1. Strategy 1: we take the memory-sharing of the corner points in Groups 1, 2, 3 and 4 in Theorem 24 in order to minimize $\mathsf{R}^{\mathsf{s}\to\mathsf{r}}$. The achieved load pair is $(\mathsf{R}^{\mathsf{s}\to\mathsf{r}}_{\text{Strategy 1}}, \mathsf{R}^{\mathsf{r}\to\mathsf{u}}_{\text{Strategy 1}})$.

2. Strategy 2: we take memory-sharing of the corner points in Groups 1 and 4 in Theorem 24 to minimize $\mathsf{R}^{\mathsf{s}\to\mathsf{r}}$, which can simultaneously lead to the optimal max link-load from relays to users $\mathsf{R}^{\mathsf{r}\to\mathsf{u}\star}$. The achieved load pair is $(\mathsf{R}^{\mathsf{s}\to\mathsf{r}}_{\text{Strategy 2}}, \mathsf{R}^{\mathsf{r}\to\mathsf{u}}_{\text{Strategy 2}})$.

The achieved load pair of [50] is $(\mathsf{R}^{\mathsf{s}\to\mathsf{r}}_{\text{ZY}}, \mathsf{R}^{\mathsf{r}\to\mathsf{u}}_{\text{ZY}})$. As a converse bound, we use the cut-set converse bound $(\mathsf{R}^{\mathsf{s}\to\mathsf{r}\star}, \mathsf{R}^{\mathsf{r}\to\mathsf{u}\star})$ in Theorem 23. We also plot the ratios of the max link-loads from relays to user as otherwise their difference would not be clearly visible on a small figure. It can be seen in Fig. 9.1 that both Strategies 1 and 2 can reduce $\mathsf{R}^{\mathsf{s}\to\mathsf{r}}_{\text{ZY}}$. In addition, Strategy 1 leads to the optimal $\mathsf{R}^{\mathsf{s}\to\mathsf{r}\star}$ when $\mathsf{M}^{\text{user}} \geq 4.5$, while the minimum user memory sizes to achieve the optimal $\mathsf{R}^{\mathsf{s}\to\mathsf{r}\star}$ of Strategy 2 and [50] are 5 and 6, respectively. In addition, when $\mathsf{M}^{\text{user}} \geq 5$, the proposed scheme can achieve the optimal load pair $(\mathsf{R}^{\mathsf{s}\to\mathsf{r}\star}, \mathsf{R}^{\mathsf{r}\to\mathsf{u}\star})$.

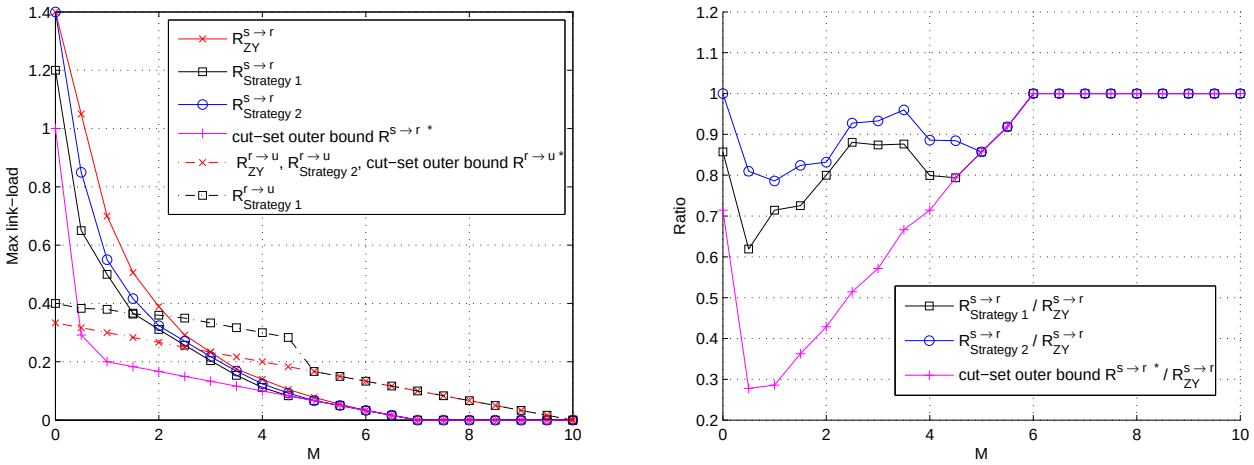

FIGURE 9.1: A combination network with cache-aided relays and users, where $\mathsf{H} = 6$, $\mathsf{N} = \mathsf{K} = \binom{\mathsf{H}}{\mathsf{r}}$, $\mathsf{r} = 2$ and $\mathsf{M}^{\text{relay}} = 1$.

9.3 Cache-aided More General Relay Networks

In this part, we consider more general relay networks. For the existing schemes with coded cache placement (the asymmetric coded placement scheme in Section 7.4 and the scheme in [50]), the schemes are achievable if each user k, the number of cached subfiles per file is the same and the number of the subfiles to be recovered is also the same. In Example 13, we can directly use the proposed coded placement scheme and the scheme in [50].

Example 13. Consider a relay network with end-user-caches where $\mathsf{N} = \mathsf{K} = 5$, $\mathsf{H} = 5$, $g = 3$ and

$$\mathcal{U}_1 = \{1,2,3\},\ \mathcal{U}_2 = \{1,3,4\},\ \mathcal{U}_3 = \{1,4,5\},\ \mathcal{U}_4 = \{2,4,5\},\ \mathcal{U}_5 = \{2,3,5\}.$$

Each user is connected to three relays and each relay is connected to three users. We want to use our asymmetric coded placement scheme in Section 7.4.1. In this example, we can see $|\mathcal{Z}_{g-1}| = 10$. Each user should cache the subfile $f_{i,\mathcal{W}}$ if $\mathcal{R}_{\mathcal{W}} \neq \emptyset$ for each file $i \in [\mathsf{N}]$. Hence, each user caches $\mathsf{k}_{1,\mathrm{b}} = 4$ subfiles per file. In the delivery phase, each user k should recover $f_{d_k,\mathcal{J}\backslash\{k\}}$ for each set $\mathcal{J} \in \mathcal{Z}_g$ and $k \in \mathcal{J}$. Hence each user should recover $\mathsf{k}_{2,\mathrm{b}} = 3$ subfiles. So the needed memory size to achieve $g = 3$ is $\mathsf{M}_{\mathrm{b}} = \mathsf{N}\mathsf{k}_{1,\mathrm{b}}/(\mathsf{k}_{1,\mathrm{b}} + \mathsf{k}_{2,\mathrm{b}}) = 20/7$. For the scheme in [50], each user caches $\mathsf{k}_1 = 6$ subfiles per file and recovers $\mathsf{k}_2 = 3$ subfiles. Hence, the needed memory size to achieve $g = 3$ is $\mathsf{M}_{\mathrm{ZY}} = \mathsf{N}\mathsf{k}_1/(\mathsf{k}_1+\mathsf{k}_2) = 10/3 > 20/7$.

When $\mathsf{M} = 20/7$, the achieved max link-load of the asymmetric coded placement scheme in Section 7.4.1 is $1/7 \approx 0.142$, which coincides with the cut-set converse bound in (1.9) extended to this network. The achieved max link-load of the schemes in [50] and in [12], and the scheme SRDS with MAN placement are $11/63 \approx 0.174$, $4/21 \approx 0.190$ and $23/140 \approx 0.164$, respectively. □

Since SRDS does not rely on the symmetric topology of combination networks, it can be used in general relay networks with end-user-caches. In addition, SRDS is designed to minimize the total link-loads to all the relays. For combination networks, minimizing the total link-loads to all the relays is equivalent to minimize the max link-load. However, for asymmetric networks, we may need to further 'balance' the link-load to each relay, which will be shown in the following example.

Example 14. Consider a relay network with end-user-caches where $\mathsf{N} = \mathsf{K} = 5$, $\mathsf{H} = 5$, $\mathsf{M} = 2$ and

$$\mathcal{U}_1 = \{1,2,3\},\ \mathcal{U}_2 = \{1,3,4\},\ \mathcal{U}_3 = \{1,4,5\},\ \mathcal{U}_4 = \{3,4,5\},\ \mathcal{U}_5 = \{2,3,5\},$$

i.e., we changed $\mathcal{U}_4$ from $\{2,4,5\}$ in Example 13 to $\{3,4,5\}$ so that the number of connected relays to each user is not the same. Let $\mathbf{d} = (1:5)$. We use SRDS with MAN placement and indicate the multicast messages as $W^h_{\mathcal{J},L} = \underset{k\in\mathcal{J}}{\oplus} \mathcal{T}^h_{k,\mathcal{J}\backslash\{k\}}$, where in this example we added the subscript L to indicate that the message has length L bits. Then, we have the server transmit

$$\begin{array}{llll}
W^1_{\{1,2,3\},\mathsf{B}/10}, & W^1_{\{1,2\},3\mathsf{B}/20}, & W^1_{\{1,3\},\mathsf{B}/30}, & W^1_{\{2,3\},\mathsf{B}/20} \quad \text{to relay } 1,\\
W^2_{\{1,3,4\},\mathsf{B}/10}, & W^2_{\{1,3\},\mathsf{B}/30}, & W^2_{\{1,4\},\mathsf{B}/20}, & W^2_{\{3,4\},\mathsf{B}/10} \quad \text{to relay } 2,\\
W^3_{\{1,4,5\},\mathsf{B}/10}, & W^3_{\{1,4\},\mathsf{B}/20}, & W^3_{\{1,5\},\mathsf{B}/12}, & W^3_{\{4,5\},\mathsf{B}/20} \quad \text{to relay } 3,\\
W^4_{\{3,4,5\},\mathsf{B}/10}, & W^4_{\{3,4\},\mathsf{B}/20}, & W^4_{\{3,5\},\mathsf{B}/30}, & W^4_{\{4,5\},\mathsf{B}/20} \quad \text{to relay } 4,\\
W^5_{\{2,3,5\},\mathsf{B}/10}, & W^5_{\{2,3\},\mathsf{B}/20}, & W^5_{\{2,5\},3\mathsf{B}/20}, & W^5_{\{3,5\},\mathsf{B}/30} \quad \text{to relay } 5.
\end{array}$$

It can be seen that the link-loads to relay 1 to 5 are $1/3$, $7/30$, $17/60$, $7/30$ and $1/3$, respectively. So the achieved max link-load is $1/3$; while the achieved max link-load by the schemes in [12] is $1/2$.

One can further improve on SRDS by observing that the link-load to relay 1 (or relay 5) is the largest. Thus, instead of transmitting $W^1_{\{1,3\},\mathsf{B}/30}$ to relay 1, we can transmit it to relay 2 which is also connected to users in $\{1,3\}$. Similarly, instead of transmitting $W^5_{\{3,5\},\mathsf{B}/30}$ to relay 5, we can transmit it to relay 4 which is also connected to users in $\{3,5\}$. With this modification, the achieved max link-load is reduced to $3/10$, which is equal to the cut-set converse bound in (1.9) (by considering the cut of relays $\{1,5\}$). □

The observation at the end of Example 14 can be translated in an improvement of the Algorithm 4 as described in Algorithm 5.

Algorithm 5 New Step 5) for Algorithm 4

1. **initialization:** $i = 1$; Define that $L_h = \sum_{\mathcal{J}\subseteq\mathcal{U}_h} |W^h_{\mathcal{J}}|$ for each $h \in [\mathsf{H}]$;
2. sort all the relays $h \in [\mathsf{H}]$ by L_h, i.e., $C(1)$ represents the relay whose L_h is maximal and $C(\mathsf{H})$ represents the relay whose L_h is minimal;
3. **if** there exists a set $\mathcal{J} \subseteq \mathcal{U}_{C(i)}$ such that $|W^{C(i)}_{\mathcal{J}}| > 0$ and there exists a set of relays (denoted by $\mathcal{Q}$) where $\mathcal{Q} \subseteq \{C(i+1),\ldots,C(\mathsf{H})\}$ and each relay $h \in \mathcal{Q}$ is connected to all the users in $\mathcal{J}$, **then,**
   (a) choose one relay $h \in \mathcal{Q}$ where $L(h) = \max_{h\in\mathcal{Q}} L(h)$, and move $\min\{|W^{C(i)}_{\mathcal{J}}|, (L_{C(i)} - L_h)/2\}$ bits from $|W^{C(i)}_{\mathcal{J}}|$ to $W^h_{\mathcal{J}}$;
   (b) update $L(h)$ for relay h and update $L(C(i))$ for relay $C(i)$;

   else then, $i = i + 1$;
4. **if** $i < \mathsf{H}$, go to Step 2) of Algorithm 4;
5. **for** each relay h and each $\mathcal{J} \subseteq \mathcal{U}_h$ where $W^h_{\mathcal{J}} \neq \emptyset$, transmit $W^h_{\mathcal{J}}$ to relay h and relay h transmits $W^h_{\mathcal{J}}$ to each user in $\mathcal{J}$;

Part 3

Summary and Prospectives

Chapter 10

Summary of Thesis and Prospectives on Future Work

In this chapter, we first revisit some results, summarize our contributions in this thesis and then present some prospectives on the future work.

10.1 Conclusions

In this thesis, we investigated the coded caching problem by building the connection between coded caching with uncoded placement and index coding, and leveraging the index coding results to characterize the fundamental limits of coded caching problem. We mainly analysed the caching problem in shared-link broadcast model and in combination networks.

10.1.1 Cache-aided Shared-link Broadcast Networks

In the first part of this thesis, for cache-aided shared-link broadcast networks, we considered the constraint that content is placed uncoded within the caches. When the cache contents are uncoded and the user demands revealed, the caching problem can be connected to an index coding problem. We derived fundamental limits for the caching problem by using tools that are either known or newly developed for the index coding problem.

A novel index coding achievable scheme was first derived based on distributed source coding. This achievable bound was proved to be strictly better than the widely used "composite (index) coding" achievable bound by leveraging the ignored correlation among composites and the non-unique decoding. For the centralized caching problem, a converse bound under the constraint of uncoded cache placement is proposed based on the "acyclic index coding converse bound". This converse bound is proved to be achieved by the cMAN scheme when the number of files is not less than the number of users, and by the proposed novel index coding achievable scheme otherwise. For the decentralized caching problem, this thesis proposes a converse bound under the constraint that each user stores bits uniformly and independently at random. This converse bound is achieved by dMAN when the number of files is not less than the number of users, and by our proposed novel index coding achievable bound otherwise.

Comment 1: Further advancement on the coded caching problem in shared-link problem beyond the results presented in this thesis are thus only possible by considering strategies where the cache placement phase is coded.

Comment 2: Index coding problem is generally an open problem. For a very limited number of graphs, we can characterize the rate region. However, in this thesis, we showed the caching problem with uncoded placement is solved. The main reason is that we can design the placement to lead the maximum multicasting opportunities.

During my Ph.D. study, we also analysed a more practical setting, the cache-aided shared-link model with finite file size (see results in [63]). But in this thesis, we focused on the information theoretical model where the file size is infinite and thus we did not go into detail on the results on finite file size regime.

10.1.2 Cache-aided Combination Networks

In this second part of this thesis, we considered the centralized caching problem in two-hop relay networks, where the server communicates with cache-aided users through some intermediate relays. Because of the hardness of analysis on the general networks, we mainly considered a well-known symmetric relay networks, combination networks, including H relays and $\binom{\mathsf{H}}{\mathsf{r}}$ users where each user is connected to a different $\mathsf{r}-$subset of relays. We aimed to minimize the max link-load for the worst cases. We derived converse and achievable bounds in this thesis.

For the converse bound, the straightforward way is that each time we consider a cut of x relays and the total load transmitted to these x relays could be converse bounded by the converse bound for the shared-link model including $\binom{x}{\mathsf{r}}$ users. We used this strategy to extend the converse bounds for the shared-link model and the acyclic index coding converse bound to combination networks. In this thesis, we also tightened the extended acyclic index coding converse bound in combination networks by further leveraging the network topology and joint entropy of the various random variables. As a result of independent interest, an inequality that generalizes the well-known sub-modularity of entropy is derived, which may find applications in other network information theory problems.

For the achievable schemes, there are two approaches, separation and non-separation. In the separation approach, we use cMAN cache placement and multicast message generation independent of the network topology. We then deliver cMAN multicast messages based on the network topology. In the non-separation approach, we design the placement and/or the multicast messages on the network topology. In this thesis, we proposed four delivery schemes on separation approach. On non-separation approach, firstly for any uncoded cache placement, we proposed a delivery scheme by generating multicast messages on network topology. With this novel delivery scheme, we then proposed an asymmetric coded cache placement, where 'asymmetric' means that it is not necessary to generate one subfile known by each $(g-1)-$subset of users and 'coded' means that we use an MDS-precoded placement such that each user needs only to decode the subfiles with highest multicast opportunities. Optimality results were also given for certain system parameters.

Comment 3: The main advantage of separation approach is the low design complexity and its extensibility to more general networks such as decentralized systems, more general relay networks. However, if we focus on some specific networks, non-separation approach provides better performance. For example, as we showed in Section 8.2.1, for combination network with $\mathsf{H}=4$, $\mathsf{r}=2$ *and* $\mathsf{M}=6/5$, *our proposed asymmetric coded placement scheme on non-separation approach is strictly better than all the scheme with uncoded cache placement and thus strictly better than all the separation approach based schemes.*

Moreover, we also extended our results to more general models, such as combination networks with cache-aided relays and users, and caching systems in more general relay networks. For combination networks with cache-aided relays and users, different to the existing scheme in which the cached contents of relays are independent of the transmitted packets from the server, we proposed a novel strategy which leads to an additional coded caching gain on the load transmitted from the server because of the cached contents of relays. Optimality results were given under some constraints and numerical evaluations showed that our proposed schemes outperform the state-of-the-art.

Comment 4: The proposed converse bounds does not lie on the symmetry of the combination networks and thus can be directly extended to more general relay networks. Our proposed achievable schemes were designed to minimize the total load transmitted from the server. Because of the symmetry of combination networks, it is equivalent to minimize the max link-load. However, if we consider the more general relay networks which are not symmetric, some further steps are needed to 'balance' the link-loads (e.g., the additional steps in Algorithm 5 described in Section 9.3).

10.2 Prospectives on Future Work

Owing to the limitation upon the duration of a Ph.D. program, it remains some possible problems to be investigated. Besides searching tighter converse bounds and achievable bounds for combination networks with end-user-caches, we allude in the sequel some other potential directions for future work.

10.2.1 Cache-aided Combination Networks with More Users than Files

It can be seen that the existing caching schemes and the proposed caching schemes in this thesis treat each sub-file demanded by each user as an independent sub-file. However, when $\mathsf{N} < \mathsf{K}$, one file may be demanded by several users such that there exist some multicasting opportunities from which we can profit. Similar to the scheme in [12] where each cMAN multicast messages is transmitted to all users by an (H, r) MDS code, one straightforward way is to extend the caching scheme for shared-link models in [19] with the load in (1.3) to combination networks. In other words, for each demanded file, we choose arbitrarily a demanding user as a leader and thus we can form a leader set $\mathcal{U}$. We let all users recover cMAN multicast messages in (2.14) $W_{\mathcal{J}}$ where $\mathcal{J} \subseteq [\mathsf{K}]$, $|\mathcal{J}| = t+1$ and $\mathcal{J} \cap \mathcal{U} \neq \emptyset$. The max link-load is achieved in the following theorem.

Theorem 27. *For a $(\mathsf{H}, \mathsf{r}, \mathsf{M}, \mathsf{N})$ combination network with $t = \mathsf{KM}/\mathsf{N} \in [0 : \mathsf{K}]$, the max link-load satisfies*

$$\mathsf{R}^{\star}_{\mathrm{c,u}} \leq \mathsf{R}_1 := \frac{\binom{\mathsf{K}}{t+1} - \binom{\mathsf{K}-\min(\mathsf{K},\mathsf{N})}{t+1}}{\mathsf{r}\binom{\mathsf{K}}{t}}. \tag{10.1}$$

The tradeoff between memory size and max link-load is the lower convex envelope of the above points.

Comparing the load in (10.1) and the cut-set converse bound under the constraint of uncoded placement in (6.2) with $x = \mathsf{H}$, we can prove that the above scheme is order optimal within factor H/r under the constraint of uncoded placement.

In the following example, we propose an improved scheme based on SRDS which can lead to a lower max link-load than the previous scheme.

Example 15. Consider the network in Fig. 1.4 with $\mathsf{N} = 2$, $\mathsf{M} = 1$ and $\mathbf{d} = (1, 1, 1, 2, 2, 2)$. We use cMAN placement phase and each sub-file $F_{i,\mathcal{W}}$ where $i \in [6]$, $\mathcal{W} \subseteq [6]$ and $|\mathcal{W}| = 1$, has length equal to $\mathsf{B}/6$. In the delivery phase, we firstly use a similar idea of SRDS to transmit $F_{d_k,\mathcal{W}}$ where $k \notin \mathcal{W}$ and there exist at least one relay connected to the users in $\{k\} \cup \mathcal{W}$, i.e., $\{k\} \cup \mathcal{W} \neq \{1, 6\}, \{2, 5\}, \{3, 4\}$. Different to SRDS, we observe that $F_{1,\{2\}}$, which will be transmitted to relay 1, is demanded by both of user 1 and 3. So we put $F_{1,\{2\}}$ in $\mathcal{T}^1_{\{1,3\},\{2\}}$ the set of bits which need to be recovered by user 1 and 3 from relay 1 and are known by user 2. Hence, in this step we have

$$\begin{aligned}
&\mathcal{T}^1_{\{1,3\},\{2\}} = \{F_{1,\{2\}}\},\ \mathcal{T}^1_{\{1,2\},\{3\}} = \{F_{1,\{3\}}\},\ \mathcal{T}^1_{\{2,3\},\{2\}} = \{F_{1,\{1\}}\} \text{ in relay } 1,\\
&\mathcal{T}^2_{\{1\},\{4\}} = \{F_{1,\{4\}}\},\ \mathcal{T}^2_{\{1\},\{5\}} = \{F_{1,\{5\}}\},\ \mathcal{T}^2_{\{4,5\},\{1\}} = \{F_{2,\{1\}}\},\\
&\mathcal{T}^2_{\{4\},\{5\}} = \{F_{2,\{5\}}\},\ \mathcal{T}^2_{\{5\},\{4\}} = \{F_{2,\{4\}}\} \text{ in relay } 2,\\
&\mathcal{T}^3_{\{2\},\{4\}} = \{F_{1,\{4\}}\},\ \mathcal{T}^3_{\{2\},\{6\}} = \{F_{1,\{6\}}\},\ \mathcal{T}^3_{\{4,6\},\{2\}} = \{F_{2,\{2\}}\},\\
&\mathcal{T}^3_{\{4\},\{6\}} = \{F_{2,\{6\}}\},\ \mathcal{T}^3_{\{6\},\{4\}} = \{F_{2,\{4\}}\} \text{ in relay } 3,\\
&\mathcal{T}^4_{\{3\},\{5\}} = \{F_{1,\{5\}}\},\ \mathcal{T}^4_{\{3\},\{6\}} = \{F_{1,\{6\}}\},\ \mathcal{T}^4_{\{5,6\},\{3\}} = \{F_{2,\{3\}}\},\\
&\mathcal{T}^4_{\{5\},\{6\}} = \{F_{2,\{6\}}\},\ \mathcal{T}^4_{\{6\},\{5\}} = \{F_{2,\{5\}}\},\ \text{in relay } 4.
\end{aligned}$$

So we transmit

$$\begin{aligned}
&RLC(\mathsf{B}/3, \mathcal{T}^1_{\{1,3\},\{2\}} \cup \mathcal{T}^1_{\{1,2\},\{3\}} \cup \mathcal{T}^1_{\{2,3\},\{2\}}) \text{ to relay } 1,\\
&RLC(\mathsf{B}/3, \mathcal{T}^2_{\{4,5\},\{1\}} \cup \mathcal{T}^2_{\{4\},\{5\}} \cup \mathcal{T}^2_{\{5\},\{4\}}), \mathcal{T}^2_{\{4\},\{5\}} \oplus \mathcal{T}^2_{\{5\},\{4\}} \text{ to relay } 2,\\
&RLC(\mathsf{B}/3, \mathcal{T}^3_{\{2\},\{4\}} \cup \mathcal{T}^3_{\{2\},\{6\}} \cup \mathcal{T}^3_{\{4,6\},\{2\}}), \mathcal{T}^3_{\{4\},\{6\}} \oplus \mathcal{T}^3_{\{6\},\{4\}} \text{ to relay } 3,\\
&RLC(\mathsf{B}/3, \mathcal{T}^4_{\{3\},\{5\}} \cup \mathcal{T}^4_{\{3\},\{6\}} \cup \mathcal{T}^4_{\{5,6\},\{3\}}), \mathcal{T}^4_{\{5\},\{6\}} \oplus \mathcal{T}^4_{\{6\},\{5\}} \text{ to relay } 4.
\end{aligned}$$

It remains to let user 1 recover $F_{1,\{6\}}$, user 2 recover $F_{1,\{5\}}$, user 3 recover $F_{1,\{4\}}$, user 4 recover $F_{2,\{3\}}$, user 5 recover $F_{2,\{2\}}$, user 6 recover $F_{2,\{1\}}$. Observe that $F_{1,\{6\}}$ has already been recovered by user 2 and 3, so in the next step $F_{1,\{6\}}$ can be treated by the side information of user 2 and 3. Hence, in the next step we transmit

$$\begin{aligned}
&RLC(\mathsf{B}/6, F_{1,\{6\}} \cup F_{1,\{5\}} \cup F_{1,\{4\}}) \text{ to relay } 1,\ \ RLC(\mathsf{B}/12, F_{2,\{1\}} \cup F_{2,\{2\}} \cup F_{2,\{3\}}) \text{ to relay } 2,\\
&RLC(\mathsf{B}/12, F_{2,\{1\}} \cup F_{2,\{2\}} \cup F_{2,\{3\}}) \text{ to relay } 3,\ \ RLC(\mathsf{B}/12, F_{2,\{1\}} \cup F_{2,\{2\}} \cup F_{2,\{3\}}) \text{ to relay } 4,
\end{aligned}$$

such that each user can recover its remaining sub-file. The link-load of relay 1 is $1/2$ and the link-loads of relay 2, 3 and 4 are all $7/12$. Hence, the achieved max link-load is $7/12$ while the minimum achieved max link-load among existing methods without considering the multicasting opportunities is $2/3$ and R_1 in (10.1) is $3/4$. □

The above example inspires us to profit from the multicasting opportunities arised from one file demanded by several users, and the future work aims to search a general scheme leveraging these opportunities.

10.2.2 Linear Programming of Converse Bound in Theorem 12

By improving the extended acyclic index coding converse bound to combination networks, we proposed a converse bound under the constraint of uncoded cache placement which was numerically showed to outperform the other converse bounds under the constraint of uncoded cache placement. However, we need to solve a linear programming including $\mathcal{O}(\mathsf{HK!})$ constraints, where for each permutation of $[\mathsf{K}]$ there is one constraint. It can be seen that for the proposed converse bound under the constraint of uncoded cache placement for shared-link models, we also have one constraint for each permutation of $[\mathsf{K}]$. Moreover, in shared-link mode, subfiles known by the same number of users are equivalent and thus we can sum all the constraints without loosing the converse bound. In contrast to the shared-link models, due to the topology of combination networks, subfiles known by the same number of users are all equivalent and thus if we sum all the constraints, the converse bound is loosen. However, there is still a symmetry in the networks. We think that we can divide the constraints into groups and sum the constraints in each group without loosing the converse bound. The future work includes solving or simplifying the linear programming.

10.2.3 Extension to Distributed Computing

Distributed computing is a hot compute science problem attracting a large number of people. Two stages ('Map' and 'Reduce') are included in a distributed computing framework. In the Map stage, distributed computing nodes process compute some intermediate values using local input values according to the designed Map functions. The main limit in the Map stage is the computation load. In the Reduce stage, the computing nodes exchange the local computed values among each other in order to compute the final output results distributedly according to the designed Reduce functions. The main limit in the Reduce stage is the communication load. The objective is to minimize the communication load for each fix computation load. Recently a coded distributed computing scheme was proposed in [71] which showed that increasing the computation load in the Map stage by a factor of r can create coded multicasting opportunities to reduce the communication load in the Reduce stage by the same factor. This scheme was designed inspired by the caching scheme for Device-to-Device systems proposed in [13]. Indeed, the Map-Reduce distributed computing problem is quite similar to the Placement-Delivery caching problem. As stated in the end of [71], the results in [71] only consider a singe-layer structure, where each node multicast the computed data to all the other nodes. However, in practical data center networks, nodes should be connected through multiple switches at different layers. Hence, one of our future direction is to extend our results in cache-aided relay networks (especially combination networks) to topological distributed computing problem.

In addition, we can also think about how to leverage the acyclic index coding bound in the Device-to-Device systems instead of broadcasting systems. We believe it can help us derive the tighter converse bound in the cache-aided Device-to-Device problem and the distributed computing problem.

Appendix A

Appendices

A.1 Proof of Theorem 3

To clarify the notations, we use different symbols for transmitted messages or known messages (nothing above), uniquely decoded ones (hat above) and non-uniquely decoded ones (check above).

Codebook Generation Fix a probability mass function

$$p_{U_1,\ldots,U_{\mathsf{N}'}}(u_1,\ldots,u_{\mathsf{N}'}) = p_{U_1}(u_1) \times \cdots \times p_{U_{\mathsf{N}'}}(u_{\mathsf{N}'}), \tag{A.1}$$

where each random variable U_i is defined on the finite alphabet $\mathcal{U}_i$ for $i \in [\mathsf{N}']$, and functions

$$f_{\mathcal{P}} : \prod_{i\in\mathcal{P}} \mathcal{U}_i \to \mathcal{X}_{\mathcal{P}}, \quad \forall \mathcal{P} \subseteq [\mathsf{N}'], \tag{A.2}$$

for some finite alphabets $\mathcal{X}_{\mathcal{P}}$ for $\mathcal{P} \subseteq [\mathsf{N}']$.

For each $i \in [\mathsf{N}']$, randomly and independently generate 2^{nR_i} sequences $u_i^n(m_i)$ indexed by $m_i \in [2^{nR_i}]$, each according to $\prod_{t=1}^n p_{U_i}(u_{i,t})$. For each $\mathcal{P} \subseteq [\mathsf{N}']$, let $x_{\mathcal{P}}^n := (x_{\mathcal{P},1},\ldots,x_{\mathcal{P},n})$ and $x_{\mathcal{P},t}(u_i : i \in \mathcal{P}) = f_{\mathcal{P}}\big((u_{i,t} : i \in \mathcal{P})\big) \in \mathcal{X}_{\mathcal{P}}$ where $t \in [n]$.

Randomly and independently assign an index $g \in [|\mathcal{X}|^n]$ to each collection of sequences $(x_{\mathcal{P}}^n : \mathcal{P} \subseteq [\mathsf{N}'])$ according to a uniform probability mass function over $[|\mathcal{X}|^n]$. The sequences with the same index g are said to form bin $\mathcal{B}(g)$. We also indicate $g = \mathsf{bin}(x_{\mathcal{P}}^n : \mathcal{P} \subseteq [\mathsf{N}'])$, the index of the bin of $x_{\mathcal{P}}^n$.

The codebook so generated is revealed to all the decoders.

Encoding Given messages $(m_1,\ldots,m_{\mathsf{N}'})$, the encoder produces $(u_1^n(m_1),\ldots,u_{\mathsf{N}'}^n(m_{\mathsf{N}'}))$ based on which it computes $(x_{\mathcal{P}}^n : \mathcal{P} \subseteq [\mathsf{N}'])$ and eventually transmits $g = \mathsf{bin}(x_{\mathcal{P}}^n : \mathcal{P} \subseteq [\mathsf{N}'])$ to all the decoders.

Decoding Fix $\mathcal{K}_j$ where $\mathcal{D}_j \subseteq \mathcal{K}_j$ and $\mathcal{K}_j \cap \mathcal{A}_j = \emptyset$ for each receiver $j \in [\mathsf{K}']$. Decoding proceeds in two steps.

Step 1: Since receiver $j \in [\mathsf{K}']$ has messages $(m_i : i \in \mathcal{A}_j)$ as side information, it also knows $(u_i^n : i \in \mathcal{A}_j)$ and $(x_{\mathcal{P}}^n : \mathcal{P} \subseteq \mathcal{A}_j)$. Upon receiving $g = \mathsf{bin}(x_{\mathcal{P}}^n : \mathcal{P} \subseteq [\mathsf{N}'])$, receiver $j \in [\mathsf{K}']$ estimates the sequences $(\widehat{x}_{\mathcal{P}}^n : \mathcal{P} \subseteq [\mathsf{N}'],\ \mathcal{P} \nsubseteq \mathcal{A}_j)$ as the unique

$$\Big(\big(\widehat{x}_{\mathcal{P}}^n : \mathcal{P} \subseteq [\mathsf{N}'],\ \mathcal{P} \nsubseteq \mathcal{A}_j\big), \big(x_{\mathcal{P}}^n : \mathcal{P} \subseteq \mathcal{A}_j\big)\Big) \in \mathcal{B}(g); \tag{A.3}$$

if none or more than one are found, it picks one uniformly at random within $\mathcal{B}(g)$.

Step 2: Receiver $j \in [\mathsf{K}']$ then uses the found $(\widehat{x}^n_{\mathcal{P}} : \mathcal{P} \subseteq [\mathsf{N}'],\ \mathcal{P} \nsubseteq \mathcal{A}_j)$, and the side information to decode all messages in $\mathcal{K}_j$, but only those in $\mathcal{D}_j$ uniquely, that is, it finds a unique tuple $(\widehat{m}_i : i \in \mathcal{D}_j)$ and some tuple $(\check{m}_i : i \in \mathcal{K}_j \backslash \mathcal{D}_j)$ such that

$$\Big(\big(u^n_i(m_i) : i \in \mathcal{A}_j\big),\ \big(u^n_i(\widehat{m}_i) : i \in \mathcal{D}_j\big),\ \big(u^n_i(\check{m}_i) : i \in \mathcal{K}_j \backslash \mathcal{D}_j\big), \\ \big(\widehat{x}^n_{\mathcal{P}} : \mathcal{P} \subseteq [\mathsf{N}'],\ \mathcal{P} \nsubseteq \mathcal{A}_j\big),\ \big(x^n_{\mathcal{P}} : \mathcal{P} \subseteq \mathcal{A}_j\big)\Big) \in T^{(n)}_\varepsilon \Big(\big(U_i : i \in \mathcal{A}_j \cup \mathcal{K}_j\big), \big(X_{\mathcal{P}} : \mathcal{P} \subseteq [\mathsf{N}']\big)\Big); \quad \text{(A.4)}$$

if none or more than one $(\widehat{m}_i : i \in \mathcal{D}_j)$ are found, it picks one uniformly at random.

Error Analysis For each decoder $j \in [\mathsf{K}']$ and $\mathcal{J} \subseteq \mathcal{K}_j$ where $\mathcal{J} \cap \mathcal{D}_j \neq \emptyset$, we define the following error events

$$\mathcal{E}_1 = \big\{ \big((U^n_i(M_i) : i \in [\mathsf{N}']), (X^n_{\mathcal{P}} : \mathcal{P} \subseteq [\mathsf{N}']) \big) \notin T^{(n)}_\varepsilon \big((U_i : i \in [\mathsf{N}']), (X^n_{\mathcal{P}} : \mathcal{P} \subseteq [\mathsf{N}']) \big) \big\}, \quad \text{(A.5)}$$

$$\mathcal{E}_{2,j} = \big\{ \text{there exists } (\widehat{x}^n_{\mathcal{P}} : \mathcal{P} \subseteq [\mathsf{N}'] \text{ and } \mathcal{P} \nsubseteq \mathcal{A}_j) \in T^{(n)}_\varepsilon \big((X_{\mathcal{P}} : \mathcal{P} \subseteq [\mathsf{N}'] \text{ and } \mathcal{P} \nsubseteq \mathcal{A}_j) | (U^n_i : i \in \mathcal{A}_j) \big) \\ \text{such that } \big((\widehat{x}^n_{\mathcal{P}} : \mathcal{P} \subseteq [\mathsf{N}'] \text{ and } \mathcal{P} \nsubseteq \mathcal{A}_j), (X^n_{\mathcal{P}} : \mathcal{P} \subseteq \mathcal{A}_j) \big) \in \mathcal{B}(G) \text{ and } (\widehat{x}^n_{\mathcal{P}} : \mathcal{P} \subseteq [\mathsf{N}'] \\ \text{and } \mathcal{P} \nsubseteq \mathcal{A}_j) \neq (X^n_{\mathcal{P}} : \mathcal{P} \subseteq [\mathsf{N}'] \text{ and } \mathcal{P} \nsubseteq \mathcal{A}_j) \big\}, \text{ where } G \text{ is the random index of } g, \quad \text{(A.6)}$$

$$\mathcal{E}_{j,\mathcal{J}} = \big\{ \text{there exists } \widehat{m}_i \neq M_i \text{ where } i \in \mathcal{J} \text{ such that } \big((U^n_i(M_i) : i \in \mathcal{K}_j \cup \mathcal{A}_j \setminus \mathcal{J}), (U^n_i(\widehat{m}_i) : i \in \mathcal{J}), \\ (X^n_{\mathcal{P}} : \mathcal{P} \subseteq [\mathsf{N}']) \big) \in T^{(n)}_\varepsilon \big((U_i : i \in \mathcal{K}_j \cup \mathcal{A}_j), (X_{\mathcal{P}} : \mathcal{P} \subseteq [\mathsf{N}']) \big) \big\}. \quad \text{(A.7)}$$

For decoder j, the probability of error at decoder j denoted by $\Pr(\mathcal{E}(j))$ can be upper bounded by

$$\Pr\big(\mathcal{E}(j)\big) \leq \Pr(\mathcal{E}_1) + \Pr\big(\mathcal{E}_1^c \cap \mathcal{E}_{2,j} | \mathcal{B}(1)\big) + \sum_{\mathcal{J} \subseteq \mathcal{K}_j : \mathcal{J} \cap \mathcal{D}_j \neq \emptyset} \Pr\big(\mathcal{E}_{j,\mathcal{J}} \cap \mathcal{E}_1^c \cap \mathcal{E}_{1,j}^c\big). \quad \text{(A.8)}$$

We now bound each term of the above expression. By LLN, $\Pr(\mathcal{E}_1) \to 0$ as $n \to \infty$. Next consider the second term in (A.8)

$$\Pr\big(\mathcal{E}_1^c \cap \mathcal{E}_{2,j} | \mathcal{B}(1)\big) \leq \sum_{(u^n_i : i \in [\mathsf{N}'])} \Pr\Big\{ U^n_i = u^n_i, i \in [\mathsf{N}'] \\ \Big| \Big(f_{\mathcal{P}}\big((U^n_i : i \in \mathcal{P})\big) : \mathcal{P} \subseteq [\mathsf{N}'] \Big) \in \mathcal{B}(1) \Big\} q_{(u^n_i : i \in [\mathsf{N}'])} \quad \text{(A.9)}$$

where

$$q_{(u^n_i : i \in [\mathsf{N}'])} := \Pr\Big\{ \big((\widehat{x}^n_{\mathcal{P}} : \mathcal{P} \subseteq [\mathsf{N}'] \text{ and } \mathcal{P} \nsubseteq \mathcal{A}_j), \Big(f_{\mathcal{P}}\big((u^n_i : i \in \mathcal{P})\big) : \mathcal{P} \subseteq \mathcal{A}_j \Big) \Big) \in \mathcal{B}(1) \text{ for some} \\ (\widehat{x}^n_{\mathcal{P}} : \mathcal{P} \subseteq [\mathsf{N}'] \text{ and } \mathcal{P} \nsubseteq \mathcal{A}_j) \in \mathcal{G}_{(u^n_i : i \in [\mathsf{N}'])} \Big| \Big(f_{\mathcal{P}}\big((U^n_i : i \in \mathcal{P})\big) : \mathcal{P} \subseteq [\mathsf{N}'] \Big) \\ \in \mathcal{B}(1), U^n_i = u^n_i \text{ where } i \in [\mathsf{N}'] \Big\}; \quad \text{(A.10)}$$

$$\mathcal{G}_{(u^n_i : i \in [\mathsf{N}'])} := \Big\{ (\widehat{x}^n_{\mathcal{P}} : \mathcal{P} \subseteq [\mathsf{N}'] \text{ and } \mathcal{P} \nsubseteq \mathcal{A}_j) \neq \Big(f_{\mathcal{P}}\big((u^n_i : i \in \mathcal{P})\big) : \mathcal{P} \subseteq [\mathsf{N}'] \text{ and } \mathcal{P} \nsubseteq \mathcal{A}_j \Big) : \\ (\widehat{x}^n_{\mathcal{P}} : \mathcal{P} \subseteq [\mathsf{N}'] \text{ and } \mathcal{P} \nsubseteq \mathcal{A}_j) \in T^{(n)}_\varepsilon \big((X_{\mathcal{P}} : \mathcal{P} \subseteq [\mathsf{N}'] \text{ and } \mathcal{P} \nsubseteq \mathcal{A}_j) | (u^n_i : i \in \mathcal{A}_j) \big) \Big\}. \quad \text{(A.11)}$$

We then focus on $q_{(u_i^n:i\in[\mathsf{N}'])}$ to obtain

$$q_{(u_i^n:i\in[\mathsf{N}'])} \leq \sum_{\substack{(\widehat{x}^n_{\mathcal{P}}:\mathcal{P}\subseteq[\mathsf{N}'] \text{ and } \mathcal{P}\nsubseteq\mathcal{A}_j)\in \\ T_\varepsilon^{(n)}\left((X_{\mathcal{P}}:\mathcal{P}\subseteq[\mathsf{N}'] \text{ and } \mathcal{P}\nsubseteq\mathcal{A}_j)|(u_i^n:i\in\mathcal{A}_j)\right)}} \Pr\left\{\left((\widehat{x}^n_{\mathcal{P}}:\mathcal{P}\subseteq[\mathsf{N}'] \text{ and } \mathcal{P}\nsubseteq\mathcal{A}_j), \left(f_{\mathcal{P}}\big((u_i^n:i\in\mathcal{P})\big):\mathcal{P}\subseteq\mathcal{A}_j\right)\right)\in\mathcal{B}(1)\right\} \tag{A.12a}$$

$$\leq 2^{n\left[H\left((X_{\mathcal{P}}:\mathcal{P}\subseteq[\mathsf{N}'] \text{ and } \mathcal{P}\nsubseteq\mathcal{A}_j)|(U_i:i\in\mathcal{A}_j)\right)+\delta\right]}|\mathcal{X}|^{-n}. \tag{A.12b}$$

From (A.9) and (A.12b) we can see that $\Pr(\mathcal{E}_1^c\cap\mathcal{E}_{2,j}|\mathcal{B}(1))$ vanishes provided that

$$H\left((X_{\mathcal{P}}:\mathcal{P}\subseteq[\mathsf{N}'] \text{ and } \mathcal{P}\nsubseteq\mathcal{A}_j)|(U_i:i\in\mathcal{A}_j)\right)\leq\log_2(|\mathcal{X}|). \tag{A.13}$$

Finally for the third term on the RHS of (A.8), by packing lemma [72, Lemma 3.1], $\Pr\left(\mathcal{E}_{j,\mathcal{J}}\cap\mathcal{E}_1^c\cap\mathcal{E}_{1,j}^c\right)\to 0$ provided that

$$\sum_{i\in\mathcal{J}}R_i \leq I\left((U_i:i\in\mathcal{J});(X_{\mathcal{P}}:\mathcal{P}\subseteq[\mathsf{N}'] \text{ and } \mathcal{P}\nsubseteq\mathcal{K}_j\cup\mathcal{A}_j\setminus\mathcal{J}),(U_i:i\in\mathcal{K}_j\cup\mathcal{A}_j\setminus\mathcal{J})\right) \tag{A.14a}$$

$$= I\left((U_i:\in\mathcal{J});(X_{\mathcal{P}}:\mathcal{P}\subseteq[\mathsf{N}'] \text{ and } \mathcal{P}\nsubseteq\mathcal{K}_j\cup\mathcal{A}_j\setminus\mathcal{J})|(U_i:i\in\mathcal{K}_j\cup\mathcal{A}_j\setminus\mathcal{J})\right). \tag{A.14b}$$

A.2 Proof of Expression (4.1a)

To avoid heavy notations, in the rest we let $c_q := \frac{\binom{\mathsf{K}}{q+1}-\binom{\mathsf{K}-\min(\mathsf{K},\mathsf{N})}{q+1}}{\binom{\mathsf{K}}{q}}$ such that $s_q = c_q - c_{q-1}$, where s_q is defined in (4.1b). From (4.9) (i.e., the fact that $(x_0,\dots,x_{\mathsf{K}})$ as defined in (4.7) can be interpreted as a probability mass function), we have

$$(c_q - qs_q)(x_q + x_{q-1}) = (c_q - qs_q)\left(1 - \sum_{i\in[0:\mathsf{K}]\setminus\{q-1,q\}} x_i\right), \tag{A.15}$$

where s_q is given in (4.1b). By the Lemma proved in Appendix A.3, for any $q\in[\mathsf{K}-1]$, $s_{q+1}\geq s_q$. Since $s_{\mathsf{K}}\leq 0$, we have $s_q\leq 0$ for all $q\in[\mathsf{K}]$. From (4.10), we have

$$s_q\big((q-1)x_{q-1}+qx_q\big) \geq s_q\left(\frac{\mathsf{KM}}{\mathsf{N}} - \sum_{i\in[0:\mathsf{K}]\setminus\{q-1,q\}} ix_i\right). \tag{A.16}$$

By summing (A.15) and (A.16) we get

$$c_{q-1}x_{q-1} + c_q x_q \geq s_q\,\frac{\mathsf{KM}}{\mathsf{N}} + c_q - s_q\,q + \sum_{i\in[0:\mathsf{K}]:i\neq q-1,q} (-c_q + (q-i)s_q)x_i. \tag{A.17}$$

Next, we substitute (A.17) into (4.16b) and get

$$\mathsf{R}^\star_{\text{c,u}} \geq \frac{s_q\mathsf{KM}}{\mathsf{N}} + c_q - s_q\,q + \sum_{i\in[0:\mathsf{K}]} w_{q,i}x_i, \tag{A.18}$$

$$w_{q,i} := c_i - c_q + (q-i)s_q. \tag{A.19}$$

Note that when $i \in \{q, q-1\}$ we have $w_{q,i} = 0$. It remains to prove for each $i \in [0:\mathsf{K}]$ we have $w_{q,i} \geq 0$. For any $q \in [\mathsf{K}]$ and $i \in [0:\mathsf{K}-1]$ we have

$$w_{q,i+1} - w_{q,i} = \frac{s_{i+1}}{\mathsf{N}} - \frac{s_q}{\mathsf{N}}. \tag{A.20}$$

From Lemma 1 and (A.20), it can be seen that for any $q \in [\mathsf{K}]$ and $i \in [0:\mathsf{K}-1]$, if $i \leq q-1$, $w_{q,i+1} \leq w_{q,i}$ and if $i \geq q-1$, $w_{q,i+1} \geq w_{q,i}$. Furthermore, $w_{q,i} = 0$ for $i \in \{q, q-1\}$. Hence, for each $i \in [0:\mathsf{K}]$, $w_{q,i} \geq 0$. As a result we have

$$\mathsf{R}^{\star}_{\mathrm{c,u}} \geq \frac{\binom{\mathsf{K}}{q+1} - \binom{\mathsf{K}-\min(\mathsf{K},\mathsf{N})}{q+1}}{\binom{\mathsf{K}}{q}} + s_q\left(\frac{\mathsf{KM}}{\mathsf{N}} - q\right), \tag{A.21a}$$

which proves bound given in (4.1a).

A.3 Lemma 1

Lemma 1. *Let* K, N *be positive integers where* $\mathsf{K} > \mathsf{N}$. *For any* $q \in [\mathsf{K}-1]$, $s_{q+1} \geq s_q$, *where* s_q *is defined in* (4.1b).

Proof. Recall that $s_q = \dfrac{\binom{\mathsf{K}}{q+1} - \binom{\mathsf{K}-\min(\mathsf{K},\mathsf{N})}{q+1}}{\binom{\mathsf{K}}{q}} - \dfrac{\binom{\mathsf{K}}{q} - \binom{\mathsf{K}-\min(\mathsf{K},\mathsf{N})}{q}}{\binom{\mathsf{K}}{q-1}}$. Focus on the first term of s_q

$$\frac{\binom{\mathsf{K}}{q+1} - \binom{\mathsf{K}-\min(\mathsf{K},\mathsf{N})}{q+1}}{\binom{\mathsf{K}}{q}} = \frac{\binom{\mathsf{K}-1}{q} + \cdots + \binom{\mathsf{K}-\min(\mathsf{K},\mathsf{N})}{q}}{\binom{\mathsf{K}}{q}} \tag{A.22a}$$

$$= \frac{\mathsf{K}-q}{\mathsf{K}} + \cdots + \frac{(\mathsf{K}-q)\times\ldots\times(\mathsf{K}-q-\min(\mathsf{K},\mathsf{N})+1)}{\mathsf{K}\times\ldots\times(\mathsf{K}-\min(\mathsf{K},\mathsf{N})+1)}. \tag{A.22b}$$

For the second term of s_q

$$\frac{\binom{\mathsf{K}}{q} - \binom{\mathsf{K}-\min(\mathsf{K},\mathsf{N})}{q}}{\binom{\mathsf{K}}{q-1}} = \frac{\binom{\mathsf{K}-1}{q-1} + \cdots + \binom{\mathsf{K}-\min(\mathsf{K},\mathsf{N})}{q-1}}{\binom{\mathsf{K}}{q-1}} \tag{A.23a}$$

$$= \frac{\mathsf{K}-q+1}{\mathsf{K}} + \cdots + \frac{(\mathsf{K}-q+1)\times\cdots\times(\mathsf{K}-q-\min(\mathsf{K},\mathsf{N})+2)}{\mathsf{K}\times\cdots\times(\mathsf{K}-\min(\mathsf{K},\mathsf{N})+1)}. \tag{A.23b}$$

Taking (A.22b) and (A.23b) into s_q, we finally obtain

$$s_q = -\frac{1}{\mathsf{K}} - \frac{2(\mathsf{K}-q-1)}{\mathsf{K}(\mathsf{K}-1)} - \cdots - \frac{\min(\mathsf{K},\mathsf{N})(\mathsf{K}-q)\times\cdots\times(\mathsf{K}-q-\min(\mathsf{K},\mathsf{N})+2)}{\mathsf{K}\times\cdots\times(\mathsf{K}-\min(\mathsf{K},\mathsf{N})+1)}. \tag{A.24}$$

Hence, it is easy to check that for any $q \in [\mathsf{K}-1]$, $s_{q+1} \geq s_q$. □

A.4 Lemma 2

Lemma 2. *Let* $\mathbf{u} = (u_1, u_2, \ldots, u_{\min(\mathsf{K},\mathsf{N})})$ *be a permutation of* $\mathcal{C}$*, where* $\mathcal{C}$ *is the chosen user set with different demands. A set of nodes not containing a directed cycle in the directed graph of the corresponding IC problem can be composed of sub-files* $\left(F_{d_{u_i},\mathcal{W}_i} : \mathcal{W}_i \subseteq [\mathsf{K}]\backslash\{u_1,\ldots,u_i\},\ i \in [\min(\mathsf{K},\mathsf{N})]\right)$.

Proof. For a $\mathbf{u} = (u_1, u_2, \ldots, u_{\min(\mathsf{K},\mathsf{N})})$, we say that sub-files/nodes $F_{d_{u_i},\mathcal{W}_i}$, for all $\mathcal{W}_i \subseteq [\mathsf{K}]\backslash\{u_1,\ldots,u_i\}$, are in level i. It is easy to see each node in level i only knows the sub-files $F_{j,\mathcal{W}}$ where $u_i \in \mathcal{W}$. So each node in level i knows neither the sub-files in the same level, nor the sub-files in the higher levels. As a result, in the proposed set there is no sub-set containing a directed cycle. □

A.5 Lemma 3

Lemma 3. *For decentralized caching system with* N *files and* K *users, if each user randomly and uniformly stores* MB/N *bits of each file, we have*

$$\Pr\left(\left|\frac{1}{\mathsf{B}}|F_{j,\mathcal{W}}| - \mathbb{E}\left[\frac{1}{\mathsf{B}}|F_{j,\mathcal{W}}|\right]\right| \geq \varepsilon\right) < \frac{1}{\mathsf{B}\varepsilon^2}. \tag{A.25}$$

Proof. Recall that $X_{j,b,u} = 1$ if user $u \in [\mathsf{K}]$ cached bit $b \in [\mathsf{B}]$ of file $j \in [\mathsf{N}]$, and zero otherwise. Note that X_{j_1,b_1,u_1} is independent of X_{j_2,b_2,u_2} when $u_1 \neq u_2$ or $b_1 \neq b_2$. For user u, there are $\binom{\mathsf{NB}}{\mathsf{MB}}$ choices with identical probability to store MB/N bits and there are $\binom{\mathsf{NB}-1}{\mathsf{MB}-1}$ choices that the b^{th} bit of F_j is known by u. So $X_{j,b,u}$ is with *Bernoulli* distribution with parameter $\binom{\mathsf{NB}-1}{\mathsf{MB}-1}/\binom{\mathsf{NB}}{\mathsf{MB}} = \mathsf{M}/\mathsf{N}$.

Let

$$|F_{j,\mathcal{W}}| = \sum_{b\in[\mathsf{B}]}\left(\prod_{u\in\mathcal{W}} X_{j,b,u}\right)\left(\prod_{k\notin\mathcal{W}}(1 - X_{j,b,k})\right) \tag{A.26a}$$

$$= \sum_{m\in[\mathsf{B}]} Y_{j,b,\mathcal{W}}, \tag{A.26b}$$

where $Y_{j,b,\mathcal{W}} = \left(\prod_{u\in\mathcal{W}} X_{j,b,u}\right)\left(\prod_{k\notin\mathcal{W}}(1 - X_{j,b,k})\right)$.

For fixed (j, b), it can be seen that $X_{j,b,u}$ are independent for all $u \in [\mathsf{K}]$. So $Y_{j,b,\mathcal{W}}$ is with Bern $\left((\frac{\mathsf{M}}{\mathsf{N}})^{|\mathcal{W}|}\left(1 - \frac{\mathsf{M}}{\mathsf{N}}\right)^{\mathsf{K}-|\mathcal{W}|}\right)$. We need to compute $\mathbb{E}[Y_{j,b_1,\mathcal{W}}Y_{j,b_2,\mathcal{W}}]$. For $b_1 = b_2$, it is not difficult to check that

$$\mathbb{E}[Y_{j,b_1,\mathcal{W}}Y_{j,b_2,\mathcal{W}}] = \mathbb{E}[Y_{j,b_1,\mathcal{W}}] \tag{A.27a}$$

$$= \left(\frac{\mathsf{M}}{\mathsf{N}}\right)^{|\mathcal{W}|}\left(1 - \frac{\mathsf{M}}{\mathsf{N}}\right)^{\mathsf{K}-|\mathcal{W}|}. \tag{A.27b}$$

When $b_1 \neq b_2$, we have

$$\mathbb{E}[Y_{j,b_1,\mathcal{W}}Y_{j,b_2,\mathcal{W}}] = \Pr(Y_{j,b_1,\mathcal{W}} = 1, Y_{j,b_2,\mathcal{W}} = 1). \tag{A.28}$$

So it remains to compute $\Pr(Y_{j,b_1,\mathcal{W}} = 1, Y_{j,b_2,\mathcal{W}} = 1)$. For user u, the probability that he knows simultaneously the b_1^{th} bit and the b_2^{th} bit of F_j is

$$\binom{\mathsf{NB}-2}{\mathsf{MB}-2}\Big/\binom{\mathsf{NB}}{\mathsf{MB}} = \frac{\mathsf{M}}{\mathsf{N}}\left(\frac{\mathsf{M}}{\mathsf{N}} - \frac{1-\frac{\mathsf{M}}{\mathsf{N}}}{\mathsf{NB}-1}\right). \tag{A.29}$$

Similarly, for user u, the probability that he knows neither the b_1^{th} bit nor the b_2^{th} bit of F_j is

$$\binom{\mathsf{NB}-2}{\mathsf{MB}}\Big/\binom{\mathsf{NB}}{\mathsf{MB}} = \left(1-\frac{\mathsf{M}}{\mathsf{N}}\right)\left(1 - \frac{\mathsf{M}}{\mathsf{N}} - \frac{\frac{\mathsf{M}}{\mathsf{N}}}{\mathsf{NB}-1}\right). \tag{A.30}$$

Hence, from (A.27b) and (A.28)

$$\begin{aligned}
\mathbb{E}[Y_{j,b_1,\mathcal{W}}Y_{j,b_2,\mathcal{W}}] &= \Pr(Y_{j,b_1,\mathcal{W}} = 1, Y_{j,b_2,\mathcal{W}} = 1) && \text{(A.31a)}\\
&= \left[\frac{\mathsf{M}}{\mathsf{N}}\left(\frac{\mathsf{M}}{\mathsf{N}} - \frac{1-\frac{\mathsf{M}}{\mathsf{N}}}{\mathsf{NB}-1}\right)\right]^{|\mathcal{W}|}\left[\left(1-\frac{\mathsf{M}}{\mathsf{N}}\right)\left(1 - \frac{\mathsf{M}}{\mathsf{N}} - \frac{\frac{\mathsf{M}}{\mathsf{N}}}{\mathsf{NB}-1}\right)\right]^{\mathsf{K}-|\mathcal{W}|} && \text{(A.31b)}\\
&< \left(\frac{\mathsf{M}}{\mathsf{N}}\right)^{2|\mathcal{W}|}\left(1-\frac{\mathsf{M}}{\mathsf{N}}\right)^{2\mathsf{K}-2|\mathcal{W}|} && \text{(A.31c)}\\
&= \mathbb{E}[Y_{j,b_1,\mathcal{W}}]^2. && \text{(A.31d)}
\end{aligned}$$

So by the proprieties of the expected value and the variance of a sum of dependent variables, it follows that

$$\begin{aligned}
\mathbb{E}\left[\frac{1}{\mathsf{B}}|F_{j,\mathcal{W}}|\right] &= \mathbb{E}\left[\sum_{b\in[\mathsf{B}]}\frac{Y_{j,b,\mathcal{W}}}{\mathsf{B}}\right] && \text{(A.32a)}\\
&= \mathbb{E}[Y_{j,b,\mathcal{W}}] && \text{(A.32b)}\\
&= \left(\frac{\mathsf{M}}{\mathsf{N}}\right)^{|\mathcal{W}|}\left(1-\frac{\mathsf{M}}{\mathsf{N}}\right)^{\mathsf{K}-|\mathcal{W}|}, && \text{(A.32c)}\\
\operatorname{Var}\left(\frac{1}{\mathsf{B}}|F_{j,\mathcal{W}}|\right) &= \frac{1}{\mathsf{B}^2}\operatorname{Var}\left(\sum_{b\in[\mathsf{B}]}Y_{j,b,\mathcal{W}}\right) && \text{(A.32d)}\\
&= \frac{1}{\mathsf{B}^2}\sum_{b\in[\mathsf{B}]}\operatorname{Var}(Y_{j,b,\mathcal{W}}) + \frac{2}{\mathsf{B}^2}\sum_{1\le b_1<b_2\le\mathsf{B}}\operatorname{Cov}(Y_{j,b_1,\mathcal{W}}, Y_{j,b_2,\mathcal{W}}) && \text{(A.32e)}\\
&= \frac{1}{\mathsf{B}}\operatorname{Var}(Y_{j,b,\mathcal{W}}) + \frac{2}{\mathsf{B}^2}\sum_{1\le b_1<b_2\le\mathsf{B}}\mathbb{E}[Y_{j,b_1,\mathcal{W}}Y_{j,b_2,\mathcal{W}}] - \mathbb{E}[Y_{j,b_1,\mathcal{W}}]^2 && \text{(A.32f)}\\
&< \frac{1}{\mathsf{B}}\operatorname{Var}(Y_{j,b,\mathcal{W}}), && \text{(A.32g)}
\end{aligned}$$

where $\operatorname{Var}(Y_{j,b,\mathcal{W}}) = (\frac{\mathsf{M}}{\mathsf{N}})^{|\mathcal{W}|}(1-\frac{\mathsf{M}}{\mathsf{N}})^{\mathsf{K}-|\mathcal{W}|}(1-(\frac{\mathsf{M}}{\mathsf{N}})^{|\mathcal{W}|}(1-\frac{\mathsf{M}}{\mathsf{N}})^{\mathsf{K}-|\mathcal{W}|}) < 1$. Finally, by Chebyshev's inequality, we have

$$\Pr\left(\left|\frac{1}{\mathsf{B}}|F_{j,\mathcal{W}}| - \mathbb{E}\left[\frac{1}{\mathsf{B}}|F_{j,\mathcal{W}}|\right]\right| \ge \varepsilon\right) < \frac{\operatorname{Var}(\frac{1}{\mathsf{B}}|F_{j,\mathcal{W}}|)}{\varepsilon^2} < \frac{1}{\mathsf{B}\varepsilon^2}, \tag{A.33}$$

which coincides with (A.25). □

A.6 Proof: $\binom{2k+1}{k}/(2k+1)$ is an integer.

If k is a positive integer, we have that

$$\frac{1}{2k+1}\binom{2k+1}{k} = \frac{1}{k+1}\binom{2k}{k} \tag{A.34a}$$

$$= \frac{k+1-k}{k+1}\binom{2k}{k} \tag{A.34b}$$

$$= \binom{2k}{k} - \frac{k}{k+1}\binom{2k}{k} \tag{A.34c}$$

$$= \binom{2k}{k} - \binom{2k}{k+1} \text{ is an integer.} \tag{A.34d}$$

A.7 Discussion of the Group Division of the Interference Elimination Scheme

To get the coefficients $[a_{g,1,j}; \ldots; a_{g,\mathsf{H},j}]$ in equation (7.14), $\mathbb{C}_g$ should be full-rank for each group $\mathcal{G}_g$. We should solve the following problem to ensure the feasibility where we introduce an integer $k = \mathsf{r} - 1$ such that $2k + 2 = 2\mathsf{r}$.

Problem 1: Let k be a positive integer. We focus on all the $\binom{2k+1}{k}$ subsets of $[2k+2]$ with cardinality $k+1$ and $2k+2$ is in each subset (because in (7.13b) and (7.13d) we only focus on the users connected to relay H). We want to divide these subsets into $\binom{2k+1}{k}/(2k+1)$ groups such that each group has $2k+1$ subsets. For each group $\mathcal{P}_i$, we create a $(2k+2)\times(2k+2)$ matrix. The first row is all 1. For each subset in this group, we have one row of 0 and 1, where the j^{th} element is 1 if and only if j is in this subset. The condition that the solution exsits for this problem is that each such matrix is full-rank.

In Appendix A.6, we prove that $\binom{2k+1}{k}/(2k+1)$ is an integer if k is a positive integer. We provide the following algorithm to construct such groups for Problem 1, which is shown by numerical evaluation to find such groups when $k \leq 12$. When $k > 12$, this numerical simulation might be infeasible due to the complexity.

Algorithm 2 Group division method for Problem 1

---

1. **input**: k, $\mathcal{P} = \{\mathcal{J} \subseteq [2k+2] : |\mathcal{J}| = k+1, (2k+2) \in \mathcal{J}\}$; **initialization**: $t_1 = 0$; $times = 10$; $\mathcal{P}_i = \emptyset$ for $i \in [\binom{2k+1}{k}/(2k+1)]$;
2. **for** $i \in [\binom{2k+1}{k}/(2k+1)]$,
   (a) $Test = 0$; randomly choose $2k+1$ subsets in $\mathcal{P}$; create a $(2k+2)\times(2k+2)$ matrix denoted by $\mathbb{C}$; (the first row of $\mathbb{C}$ is all 1 and for each chosen subset, there is one row of 0 and 1, where j^{th} element is 1 if and only if $j \in [2k+2]$ is in this subset;)
   (b) **if** $\mathbb{C}$ is full-rank, **then** $Test = 1$ and put the chosen subsets in $\mathcal{P}_i$;
   (c) **if** $Test = 0$ and $t_1 \leq times$, **then** $t_1 = t_1 + 1$ and go to Step 2-a);

3. **if** $\mathcal{P}_i \neq \emptyset$ for all $i \in [\binom{2k+1}{k}/(2k+1)]$, **then Output** $\mathcal{P}_i$ for all $i \in [\binom{2k+1}{k}/(2k+1)]$; **else, then** go to Step 1);

We can use Algorithm 2 to construct the groups. However, it is hard to prove the existence of the group division satisfying the full-rank condition for the general case. Instead of proving the existence of solution for Problem 1, we introduce Problem 2, the existence of whose solution is easier to analyse. Since the number $2k+2$ appears in each subsets of Problem 1, we do not consider the number $2k+2$ in Problem 2. In Appendix A.8 we prove that the we can add $2k+2$ into each subset in the solution of Problem 2 to get one solution of Problem 1.

Problem 2: Let k be a positive integer. We focus on all the $\binom{2k+1}{k}$ subsets of $[2k+1]$ with cardinality k. We want to divide these subsets into $\binom{2k+1}{k}/(2k+1)$ groups such that each group has $2k+1$ subsets. In each group, the number of subsets containing each number in $[2k+1]$ is the same (equal to k). We create a $(2k+1) \times (2k+1)$ matrix, called *incident matrix*. There is one row of 0 and 1 in the incident matrix corresponding to each subset in this group, where the j^{th} element in the row is 1 if and only if j is in this subset. The condition is that each incident matrix is full-rank.

Compared to Problem 1, Problem 2 has an additional constraint, which is that in each group, the number of subsets containing each number in $[2k+1]$ is the same. In Example 7, we have a group division satisfying Problem 1. In addition, if we take out the number $2k+2=6$ in each subset, it is a solution for Problem 2. To analyse the existence, we firstly recall the following theorem given in [73, Theorem 1.1].

Theorem 28 ([73])**.** *Let k' and n' be positive integers, and let λ be the smallest non-trivial divisor of n'. Then all the $\binom{n'}{k'}$ subsets of $[n']$ with cardinality k' could be divided into $\binom{n'}{k'}/n'$ non-overlapping groups, where each group includes n' subsets and its incident matrix is circulant, if and only if n' is relatively prime to k', $\lambda k' > n'$ and n' divides $\binom{n'}{k'}$.*

A circulant $n' \times n'$ matrix is uniquely determined by its first row $[c_0, c_1, \ldots, c_{n'-1}]$ and its i^{th} row is obtained by shifting the first row rightwards by $i-1$ where $i \in [2 : n']$. In Problem 2, $n' = 2k+1$ and $k' = k$. It is easy to see that $2k+1$ and k are relatively prime and that $\lambda \geq 3$ leading $\lambda k > 2k+1$. In addition, in Appendix A.6 we prove that $2k+1$ divides $\binom{2k+1}{k}$. Hence, if we choose $n' = 2k+1$ and $k' = k$, the conditions in Theorem 28 are satisfied. Since the incident matrix of each group is circulant, we can see that in each group, the number of sets containing each number in $[2k+1]$ is the same. Hence, it remains to analyse the rank of each incident matrix. In Appendix A.9, we prove the following theorem (We are grateful to Mr. Zhangchi Chen from Math department of University Paris-sud who proves the following theorem for us).

Theorem 29. *Let k be a positive integer. A $(2k+1) \times (2k+1)$ circulant matrix, where the number of 1 in the first row is k and the number of 0 in the first row is $k+1$, is always invertible if $2k+1 = p^v$ or pq, where p, q are different primes and v is a positive integer.*

A.8 Proof: A solution of Problem 2 is a solution of Problem 1

For a solution of Problem 2, we focus on any group g and assume that the $(2k+1) \times (2k+1)$ incident matrix is $\mathbb{B}$. We prove that the matrix $\mathbb{C}_g$ is also full-rank, where $\mathbb{C}_g = \begin{bmatrix} \mathbb{E}^T & 1 \\ \mathbb{B} & \mathbb{E} \end{bmatrix}$ and $\mathbb{E} = [1; \ldots; 1]$ with dimension $(2k+1) \times 1$. $\mathbb{E}^T$ is the transpose of $\mathbb{E}$. It can be seen that if $\mathbb{C}_g$ with dimension $(2k+2) \times (2k+2)$ is full-rank for each group g, we can add $2k+2$ into each subset in the solution of Problem 2 to get one solution of Problem 1.

We prove it by contradiction, i.e., assume that $\mathbb{C}_g$ is not full-rank. There must exist a sequence of real numbers $(a_1, \ldots, a_{2k+1})$ such that $\sum_{i \in [2k+1]} a_i \mathbb{B}_i = \mathbb{E}^T$, where $\mathbb{B}_i$ represents the i^{th} row of $\mathbb{B}$. In other words, we have

$$\mathbb{B}^T \times \begin{bmatrix} a_1 \\ \cdots \\ a_{2k+1} \end{bmatrix} = \mathbb{E}. \tag{A.35}$$

Since $\mathbb{B}$ is full-rank, $\mathbb{B}^T$ is also full-rank. Hence, there is only one sequence $(a_1, \ldots, a_{2k+1})$ where $\sum_{i \in [2k+1]} a_i \mathbb{B}_i = \mathbb{E}^T$. For a solution of Problem 2, in each group, the number of sets containing each number in $[2k+1]$ is the same. Hence, $(a_1, \ldots, a_{2k+1}) = (1/k, \ldots, 1/k)$. However, in this case, $\sum_{i \in [2k+1]} a_i \neq 1$, not satisfying the last column of $\mathbb{C}_g$. Hence, we prove that $\mathbb{C}_g$ is full-rank.

A.9 Proof of Theorem 29

In this section, we let i be the imaginary unit. Recall that the following equation

$$X^{n'} = 1, \tag{A.36}$$

has n' distinct roots in $\mathbb{C}$ (complex field), called $(n')^{\text{th}}$ roots of unity. Write $\zeta := e^{\frac{2\pi i}{n'}}$, then the group of n' roots can be expressed as

$$\mu_{n'} := \{1, \zeta, \ldots, \zeta^{n'-1}\}. \tag{A.37}$$

A circulant $n' \times n'$ matrix is uniquely determined by its first row $[c_0, c_1, \ldots, c_{n'-1}]$ and its i^{th} row is obtained by shifting the first row rightwards by $i-1$ where $i \in [2 : n']$. The eigenvalues of this matrix are given by

$$\lambda_j := \sum_{s=0}^{n'-1} c_s \omega_j^s, \;\; j = 0, \ldots, n'-1 \tag{A.38}$$

where $\omega_j := e^{\frac{2\pi i j}{n'}}$. In our case $n' = 2k+1$ where k is a positive integer and in the first row the number of 1 is k while the number of 0 is $k+1$, i.e., $|\{s \in [0 : n'-1] : c_s = 1\}| = k$ and $|\{s \in [0 : n'-1] : c_s = 0\}| = k+1$. In this thesis we will study the following matrix.

Definition 2. *A $(2k+1)\times(2k+1)$ matrix is called a* k-matrix *if it is circulant, binomial and the first row contains exactly k elements of 1 and $k+1$ elements of 0.*

Note that when $\gcd(j,n)=1$, the eigenvalues of a k-matrix is always a sum of k distinct $(2k+1)$-th roots of unity. But when $\gcd(j,n)\geq 1$, the sum may be taken over repeated roots. In this thesis, we want to discuss whether a k-matrix is invertible. We have the following relation:

$$\begin{gathered}\text{There exists } k \text{ distinct } (2k+1)\text{-th roots of unity with sum } 0.\\ \Downarrow\\ \text{There exists some non-invertible } k\text{-matrices.}\\ \Downarrow\\ \text{There exists } k \ (2k+1)\text{-th roots of unity with sum } 0.\end{gathered}$$

And equivalently

$$\begin{gathered}\text{Any sum of } k \text{ distinct } (2k+1)\text{-th roots of unity is not } 0.\\ \Uparrow\\ \text{Any } k\text{-matrix is invertible.}\\ \Uparrow\\ \text{Any sum of } k \ (2k+1)\text{-th roots of unity is not } 0.\end{gathered}$$

In the following we prove the following lemmas.

Lemma 4. *If $2k+1=p^e$, where p is a prime number and $e\in\mathbb{Z}_+$, then any k distinct $(2k+1)$-th roots will* NOT *have sum 0.*

Lemma 5. *If $2k+1=pq$, where p,q are 2 distinct prime number, then any k distinct $(2k+1)$-th roots will* NOT *have sum 0.*

Lemma 6. *If $2k+1=p^e$ is a power of a prime, then any k-matrix is invertible.*

Lemma 7. *If $2k+1=pq$ is a product of two distinct primes p,q, then any k-matrix is invertible.*

Proof of Lemma 4. Recall that $\mathbb{Q}$ is the field of rational numbers and that $\mathbb{Q}[\zeta]$ is the cyclotomic field obtained by adjoining a primitive root of unit ζ to the rational numbers. We have $dim_{\mathbb{Q}}\mathbb{Q}[\zeta]=\phi(2k+1)=p^e-p^{e-1}$ where ϕ is the *Euler phi function* and $dim_{\mathbb{Q}}$ represents the degree of the field extension over $\mathbb{Q}$. As a $\mathbb{Q}$-vector space, $\mathbb{Q}[\zeta]$ is generated by $\{1,\zeta,\zeta^2,\ldots,\zeta^{2k}\}$ with the linear relations

$$\tau_\beta:\zeta^\beta(1+\zeta^{p^{e-1}}+\zeta^{2p^{e-1}}+\cdots+\zeta^{(p-1)p^{e-1}})=0, \tag{A.39}$$

where $\beta\in[0:p^{e-1}-1]$. Recall $\mathbb{Q}^a$ represents the $a-$dimension vector space of $\mathbb{Q}$. We define an injective linear map d_1 and a surjective linear map d_0 with the following following sequence of $\mathbb{Q}$-linear maps

$$\mathbb{Q}^{p^{e-1}}\xrightarrow{d_1}\mathbb{Q}^{p^e}\xrightarrow{d_0}\mathbb{Q}[\zeta] \tag{A.40}$$

Define $\{e_\alpha:\alpha\in[0:p^e-1]\}$ is a base of $\mathbb{Q}^{p^e}$ and $\{\tau_\beta:\beta\in[0:p^{e-1}-1]\}$ is a base of $\mathbb{Q}^{p^{e-1}}$. We have $d_0(e_\alpha)=\zeta^\alpha$ and $d_1(\tau_\beta)=\sum_{s=0}^{p-1}e_{sp^{e-1}+\beta}$. Recall $\ker(L)$ is the kernel of a linear map L and $\mathrm{im}(L)$ is

the image of a linear map L. In addition, we have $d_0\Big(d_1(\tau_\beta)\Big) = 0$. Thus $\mathrm{im}(d_1) \subset \ker(d_0)$. Moreover, $dim_{\mathbb{Q}} \ker(d_0) = dim_{\mathbb{Q}}\mathbb{Q}^{p^e} - dim_{\mathbb{Q}}\mathrm{im}(d_0) = p^e - \phi(p^e) = p^{e-1} = dim_{\mathbb{Q}}\mathrm{im}(d_1)$. So $\mathrm{im}(d_1) = \ker(d_0)$. We conclude that the sequence above is exact.

Suppose that there exists some $\alpha_1, \ldots, \alpha_k \in \mathbb{Z}_+$ such that $0 \leqslant \alpha_1 < \cdots < \alpha_k \leqslant p^e - 1$ and $\sum_{j=1}^{k} \zeta^{\alpha_j} = 0$. Then $\sum_{j=1}^{k} e_{\alpha_j} \in \ker(d_0) = \mathrm{im}(d_1)$. There exists (uniquely) some $\lambda_0, \ldots, \lambda_{p^{e-1}-1} \in \mathbb{Q}$ such that $d_1(\sum_{\beta=0}^{p^{e-1}-1} \lambda_\beta \tau_\beta) = \sum_{j=1}^{k} e_{\alpha_j}$, i.e.

$$\sum_{j=1}^{k} e_{\alpha_j} = \sum_{s=0}^{p-1} \sum_{\beta=0}^{p^{e-1}-1} \lambda_\beta e_{sp^{e-1}+\beta}. \tag{A.41}$$

Thus $\lambda_0, \ldots, \lambda_{p^{e-1}-1} \in \mathbb{Z}$. We have $p \sum_{\beta=0}^{p^{e-1}-1} \lambda_\beta = k$. So $p|k$. This contradicts with the fact that $p|(2k+1)$ and $\gcd(k, 2k+1) = 1$. □

Proof of Lemma 6. In fact, even if we allow repetitions, any sum of k $(2k+1)$-th roots of unity is nonzero. Suppose not, i.e., we have some $\alpha_1, \ldots, \alpha_t \in \mathbb{Z}$ and $r_1, \ldots, r_t \in \mathbb{Z}_+$ such that $0 \leqslant \alpha_1 < \alpha_2 < \cdots < \alpha_t \leqslant 2k$, $\sum_{j=1}^{t} r_j = k$ and $\sum_{j=1}^{t} r_j \zeta^{\alpha_j} = 0$. Then, there exists (uniquely) some $\lambda_0, \ldots, \lambda_{p^{e-1}-1} \in \mathbb{Q}$ such that

$$\sum_{j=1}^{t} r_j e_{\alpha_j} = \sum_{s=0}^{p-1} \sum_{\beta=0}^{p^{e-1}-1} \lambda_\beta e_{sp^{e-1}+\beta}. \tag{A.42}$$

Thus $\lambda_0, \ldots, \lambda_{p^{e-1}-1} \in \mathbb{Z}$ and we raise the same contradiction as above. □

In the rest of the thesis we consider the case $2k + 1 = pq$, where p, q are two distinct primes, and prove Lemma 5 and Lemma 7. The idea is similar to the proves of Lemma 4 and Lemma 6, where we write down generators of $\mathbb{Q}$-linear relations among $\{1, \zeta, \ldots, \zeta^{2k}\}$. In this case there could be two kinds of linear relations:

$$\tau_{p,\beta_p} : \zeta^{\beta_p}(1 + \zeta^q + \zeta^{2q} + \cdots + \zeta^{(p-1)q}) = 0, \quad \beta_p \in [0 : q-1]. \tag{A.43}$$

$$\tau_{q,\beta_q} : \zeta^{\beta_q}(1 + \zeta^p + \zeta^{2p} + \cdots + \zeta^{(q-1)p}) = 0, \quad \beta_p \in [0 : q-1]. \tag{A.44}$$

Definition 3. *For any prime $p|2k+1$, a* p-orbit *is a set*

$$O_{p,\beta_p} := \{\zeta^{\beta_p}, \zeta^{\frac{2k+1}{p}+\beta_p}, \zeta^{\frac{2(2k+1)}{p}+\beta_p}, \ldots, \zeta^{\frac{(p-1)(2k+1)}{p}+\beta_p}\} \tag{A.45}$$

where $\beta_p \in \left[0 : \frac{2k+1}{p} - 1\right]$. It contains exactly p distinct $(2k+1)$-th roots of unity and has sum 0.

Note that when $2k + 1 = pq$, there are in total q p-orbits (summands in τ_{p,β_p}) and p q-orbits (summands in τ_{q,β_q}).

Conjecture 1. *Any subset of μ_{2k+1} with sum 0 is a disjoint union of $\cup_j O_{p_j,\beta_{p_j}}$, where $O_{p_j,\beta_{p_j}}$ is a p_j-orbit with p_j a prime factor of $2k+1$.*

We have proved Conjecture 1 when $2k+1=p^e$. Now we prove Conjecture 1 when $2k+1=pq$.

Proof of Conjecture 1 when $2k+1=pq$. Without loss of generality we suppose that $p<q$. In this case we have $dim_{\mathbb{Q}}(\mathbb{Q}[\zeta])=\phi(pq)=(p-1)(q-1)=pq-p-q+1$. As a $\mathbb{Q}$-vector space, $\mathbb{Q}[\zeta]$ is generated by $\{1,\zeta,\ldots,\zeta^{2k}\}$ with relations $\{\tau_{p,\beta_p}:\beta_p\in[0:q-1]\}$ and $\{\tau_{q,\beta_q}:\beta_q\in[0:p-1]\}$. However, unlike the case where $2k+1=p^e$, these relations are not independent. There is a second relation since

$$\sum_{\beta_p=0}^{q-1}\sum_{s_p=0}^{p-1}e_{s_pq+\beta_p}=\sum_{\beta_q=0}^{p-1}\sum_{s_q=0}^{q-1}e_{s_qp+\beta_q}. \tag{A.46}$$

In other words, defining linear maps d_0, d_1 and d_2 where d_2 is injective and d_0 is surjective, we have the following sequence of $\mathbb{Q}$-linear maps

$$\mathbb{Q}\xrightarrow{d_2}\mathbb{Q}^{p+q}\xrightarrow{d_1}\mathbb{Q}^{pq}\xrightarrow{d_0}\mathbb{Q}[\zeta] \tag{A.47}$$

Define $\{e_\alpha:\alpha\in[0:pq-1]\}$ is a base of $\mathbb{Q}^{pq}$ and $\{\tau_{p,\beta_p}:\beta_p\in[0:q-1]\}\cup\{\tau_{q,\beta_q}:\beta_q\in[0:p-1]\}$ is a base of $\mathbb{Q}^{p+q}$. We have $d_0(e_\alpha)=\zeta^\alpha$, $d_1(\tau_{p,\beta_p})=\sum_{\beta_p=0}^{q-1}\sum_{s_p=0}^{p-1}e_{s_pq+\beta_p}$, $d_1(\tau_{q,\beta_q})=\sum_{\beta_q=0}^{p-1}\sum_{s_q=0}^{q-1}e_{s_qp+\beta_q}$ and $d_2(1)=\sum_{\beta_p=0}^{q-1}\tau_{p,\beta_p}-\sum_{\beta_q=0}^{p-1}\tau_{q,\beta_q}$. In addition, we have $d_0\big(d_1(\tau_{p,\beta_p})\big)=0$, $d_0\big(d_1(\tau_{q,\beta_q})\big)=0$ and $d_1\big(d_2(1)\big)=0$. Thus $\mathrm{im}(d_2)\subset\ker(d_1)$, $\mathrm{im}(d_1)\subset\ker(d_2)$. Now we prove $\mathrm{im}(d_2)=\ker(d_1)$. For any $\Lambda_1:=\sum_{\beta_p=0}^{q-1}\lambda_{p,\beta_p}\tau_{p,\beta_p}+\sum_{\beta_q=0}^{p-1}\lambda_{q,\beta_q}\tau_{q,\beta_q}\in\ker(d_1)$, where $\lambda_{p,0},\ldots,\lambda_{p,q-1},\lambda_{q,0},\ldots,\lambda_{q,p-1}\in\mathbb{Q}$, we have

$$d_1(\Lambda_1)=\sum_{\beta_p=0}^{q-1}\sum_{\beta_q=0}^{p-1}(\lambda_{p,\beta_p}+\lambda_{q,\beta_q})e_{f(\beta_p,\beta_q)}=0, \tag{A.48}$$

where $f(\beta_p,\beta_q)$ is the unique solution in $\{0,\ldots,pq-1\}$, guaranteed by Chinese Reminder Theorem [74], of the equations

$$\begin{cases} f(\beta_p,\beta_q)\equiv\beta_p \pmod q\\ f(\beta_p,\beta_q)\equiv\beta_q \pmod p\end{cases} \tag{A.49}$$

where $\equiv$ is the congruence modulo. Thus we get $\lambda_{p,\beta_p}+\lambda_{q,\beta_q}=0,\forall\beta_p,\beta_q$. So $\lambda_{p,\beta_p}=\lambda_{p,0}=-\lambda_{q,\beta_q},\forall\beta_p,\beta_q$, i.e. $\Lambda_1=d_2(\lambda_{p,0})\in\mathrm{im}(d_2)$.

To prove $\mathrm{im}(d_1)=\ker(d_0)$ we calculate dimensions. We have $dim_{\mathbb{Q}}\ker(d_0)=dim_{\mathbb{Q}}\mathbb{Q}^{pq}-dim_{\mathbb{Q}}\mathbb{Q}[\zeta]=p+q-1$, while $dim_{\mathbb{Q}}\mathrm{im}(d_1)=dim_{\mathbb{Q}}\mathbb{Q}^{p+q}-dim_{\mathbb{Q}}\ker(d_1)=p+q-dim_{\mathbb{Q}}\mathrm{im}(d_2)=p+q-1$ as well. Thus $\mathrm{im}(d_1)=\ker(d_0)$. We conclude that the sequence above is exact. Define μ_{2k+1} as the Abelian group composed by all the $(2k+1)$-th roots of unity. Let $E\subset\mu_{2k+1}$ be a set of $(2k+1)$-th roots with sum 0. Define $\lambda_\alpha=1$ if $\zeta^\alpha\in E$ and 0 if not. Thus we have

$$\sum_{\alpha=0}^{pq-1}\lambda_\alpha\zeta^\alpha=0, \tag{A.50}$$

$$\text{i.e., } \Lambda_0 := \sum_{\alpha=0}^{pq-1} \lambda_\alpha e_\alpha \in \ker(d_0) = \text{im}(d_1). \tag{A.51}$$

So there exists $\lambda_{p,0}, \ldots, \lambda_{p,q-1}, \lambda_{q,0}, \ldots, \lambda_{q,p-1} \in \mathbb{Q}$ such that

$$\Lambda_0 = d_1\Big(\sum_{\beta_p=0}^{q-1} \lambda_{p,\beta_p}\tau_{p,\beta_p} + \sum_{\beta_q=0}^{p-1} \lambda_{q,\beta_q}\tau_{q,\beta_q}\Big), \tag{A.52}$$

$$\text{i.e., } \lambda_{p,\beta_p} + \lambda_{q,\beta_q} = \lambda_{f(\beta_p,\beta_q)} \;\; \beta_p \in [0:q-1], \beta_q \in [0:p-1]. \tag{A.53}$$

The choice of an element in $d_1^{-1}(\Lambda_0)$ is not unique. By subtracting $d_2(\lambda_{p,0})$ we may assume that $\lambda_{p,0} = 0$. Since $\lambda_f(\beta_p, \beta_q) \in \{0, 1\}$ we conclude that $\lambda_{q,\beta_q} \in \{0, 1\}, \forall \beta_q$ and $\lambda_{p,\beta_p} \in \{-1, 0, 1\}, \forall \beta_p \geqslant 1$.

Easy case: if none of λ_{p,β_p} takes value of -1, then $A = (\bigcup_{\beta_p,\lambda_{p,\beta_p}=1} O_{p,\beta_p}) \cup (\bigcup_{\beta_q,\lambda_{q,\beta_q}=1} O_{q,\beta_q})$ is a union of orbits. It is indeed a disjoint union since $\lambda_{f(\beta_p,\beta_q)} \leqslant 1$.

Special case: if there exists some $\beta_p' \in \{1, \ldots, q-1\}$ such that $\lambda_{p,\beta_p} = -1$, then $\lambda_{q,\beta_q} = 1, \forall \beta_q$ since $\lambda_f(\beta_p, \beta_q) \geqslant 0$. Moreover $\lambda_{p,\beta_p} \neq -1$ since $\lambda_f(\beta_p, \beta_q) \leqslant 1$. By adding $d_2(1)$ we get an element in $d_1^{-1}(\Lambda_0)$ with coefficient in $\{0, 1\}$. We are back in the easy case then. □

Proof of Lemma 5. Note that any p-orbit O_{p,β_p} intersects with any q-orbit O_{q,β_q} at exactly one element $\zeta^{f(\beta_p,\beta_q)}$, by using Chinese Remainder Theorem [74]. Define $\#(A)$ as the number of orbits in the union A. Hence a disjoint union of orbits must be a union of orbits with the same length. So either $p|\#(A)$ or $q|\#(A)$. Thus $\#(A) \neq k$ since $\gcd(k, pq) = \gcd(k, 2k+1) = 1$. □

Proof of Lemma 7. Now we study the eigenvalues of k-matrices. Given a k-matrix

$$\begin{bmatrix} c_0 & c_1 & c_2 & \ldots & c_{n-1} \\ c_{n-1} & c_0 & c_1 & \ldots & c_{n-2} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ c_1 & c_2 & c_3 & \ldots & c_0 \end{bmatrix}$$

where the only nonzero terms are $c_{\alpha_j} = 1$, with $0 \leqslant \alpha_1 < \alpha_2 < \cdots < \alpha_k \leqslant 2k$. The eigenvalues are given by

$$\lambda_j = \sum_{s=0}^{k} \zeta^{j\alpha_s}, \;\; j \in [0:2k]. \tag{A.54}$$

Recall that $2k+1 = pq$. If $\gcd(j, pq) = 1$, then λ_j is a sum of k distinct pq-th roots of unity, which is nonzero by the argument above. If $\gcd(j, pq) = p$, then λ_j is a sum of k q-th roots of unity, which is nonzero by the previous section and the fact that $\gcd(k, q) = 1$. For the same reason if $\gcd(j, pq) = q$, $\lambda_j \neq 0$. Thus all eigenvalues of this arbitrarily chosen k-matrix are nonzero.

A.10 Proof of Theorem 19

A.10.1 Proof of Theorem 19-1)

Achievability. We focus on one set of users $\mathcal{J}$ where $|\mathcal{J}| > t$. If each relay is connected to at most t users of $\mathcal{J}$, $\mathcal{J}$ at most contains $\mathsf{H}t/\mathsf{r}$ users. So if $\mathsf{H}t/\mathsf{r} < t+1$, there must be one relay connected to $t+1$ users of $\mathcal{J}$. So if $\mathsf{H}t/\mathsf{r} < t+1$, we have $\mathcal{V}_2 = \emptyset$, i.e., for any set $\mathcal{J}$ of $t+1$ users, there exists at least one relay connected to all of these users. So the max link-load of DIS is $\frac{\mathsf{K}(1-\mathsf{M}/\mathsf{N})}{\mathsf{H}(1+\mathsf{KM}/\mathsf{N})}$.

Then we focus on CICS under this case. For each set $\mathcal{J}$ of $t+1$ users, we have $\max_{h\in[\mathsf{H}]} |\mathcal{U}_h \cap \mathcal{J}| = t+1$. So the total link-load to transmit is $\frac{|W_{\mathcal{J}}|}{\mathsf{HB}}$; thus the link-load to transmit all the demanded sub-files of users is also $\frac{\mathsf{K}(1-\mathsf{M}/\mathsf{N})}{\mathsf{H}(1+\mathsf{KM}/\mathsf{N})}$.

Converse. The converse bound in (6.2) coincides with the above load by taking with $x = \mathsf{r}$.

A.10.2 Proof of Theorem 19-2)

Achievability. By setting $\mathsf{r} = \mathsf{H}-1$ in (7.2), we have $\mathcal{V}_2 = \emptyset$ when $t \le \mathsf{K}-2$, i.e., for any set $\mathcal{J}$ of $t+1$ users, there exists at least one relay connected to all of these users. So similar to the previous case, for each $t \in [0:\mathsf{K}-2]$, the max link-load achieved by DIS or CICS is $\frac{\mathsf{K}-t}{(t+1)\mathsf{H}}$. For $t = \mathsf{K}$, the link-load is 0. For $t \in [\mathsf{K}-2:\mathsf{K}]$, we use memory-sharing between the points $t = \mathsf{K}-2$ and $t = 0$.

Converse. The converse bound in (6.2) coincides with the lower convex-hull of the above loads by taking $x = \mathsf{H}$ when $\mathsf{M} \le \mathsf{N}(\mathsf{K}-2)/\mathsf{K}$, and $x = \mathsf{r}$ when $\mathsf{M} \ge \mathsf{N}(\mathsf{K}-2)/\mathsf{K}$.

A.10.3 Proof of Theorem 19-3)

Achievability for $\mathsf{H} < 2\mathsf{r}$. When $t = \mathsf{KM}/\mathsf{N} = 1$, we have $\mathcal{V}_2 = \emptyset$ so the load from the source to all the relays is $(\mathsf{K}-t)/(t+1) = (\mathsf{K}-1)/2$; due to the symmetry, the load from the source to each relay is the same, and thus the link-load is $\frac{\mathsf{K}-1}{2\mathsf{H}}$. When $\mathsf{M} = 0$, the load is K/H. By the memory-sharing between $\mathsf{M} = 0$ and $\mathsf{M} = \mathsf{KM}/\mathsf{N}$, we can achieve the load in (8.2d).

Converse for $\mathsf{H} < 2\mathsf{r}$. The converse bound in (6.2) coincides with the achievable bound by taking $x = \mathsf{H}$ when $0 \le \mathsf{M} \le \mathsf{N}/\mathsf{K}$.

Achievability for $\mathsf{H} = 2\mathsf{r}$. When $0 \le \mathsf{M} \le \mathsf{N}/\mathsf{K}$, from Theorem 14 and by memory-sharing between $\mathsf{M} = 0$ and $\mathsf{M} = \mathsf{N}/\mathsf{K}$, the load achieved by IES is $\frac{\mathsf{K}(\mathsf{H}-1)-(\frac{\mathsf{KH}+\mathsf{H}-\mathsf{K}}{2}-1)\frac{\mathsf{KM}}{\mathsf{N}}}{\mathsf{H}(\mathsf{H}-1)}$.

Converse for $\mathsf{H} = 2\mathsf{r}$. We use the converse bound in Theorem 11. In Appendix A.10.4 we prove that

$$\mathsf{H}(\mathsf{H}-1)\mathsf{R}^{\star}_{\mathsf{c,u}} \ge \mathsf{K}(\mathsf{H}-1) - \left(\frac{\mathsf{KH}+\mathsf{H}-\mathsf{K}}{2}-1\right)\frac{\mathsf{KM}}{\mathsf{N}} + \sum_{\mathcal{W}\subseteq[\mathsf{K}]:|\mathcal{W}|>1} z_{\mathcal{W}} x_{\mathcal{W}}, \tag{A.55}$$

where $z_{\mathcal{W}} \ge 0$ is the coefficient of $x_{\mathcal{W}}$ in (6.3b). Hence, the bound in (A.55) coincides with the achievable bound when $0 \le \mathsf{M} \le \mathsf{N}/\mathsf{K}$.

A.10.4 Proof of (A.55)

When $\mathsf{H} = 2\mathsf{r}$, we compute the converse bound in Theorem 11. We sum all the inequalities as (6.39) for all the sets of relays $\mathcal{Q}$ where $|\mathcal{Q}| = \mathsf{H}-1$ and for all the permutations $\mathbf{p}(\mathcal{K}_{\mathcal{Q}})$. In (6.39), there are $|\mathcal{K}_{\mathcal{Q}}| = \mathsf{K}/2$

terms of $x_\emptyset$ and $(\mathsf{K}-1)+\ldots+(\mathsf{K}-|\mathcal{K}_\mathcal{Q}|)=(3\mathsf{K}/2-1)\mathsf{K}/4$ terms of $x_\mathcal{W}$ where $|\mathcal{W}|=1$. Because of the symmetry, from the sum we obtain

$$\mathsf{H}(\mathsf{H}-1)\mathsf{R}^\star_{\mathsf{c},\mathsf{u}} \geq \frac{\mathsf{K}}{2}\mathsf{H}x_\emptyset + \frac{\mathsf{H}}{4}(\frac{3}{2}\mathsf{K}-1)x_1 + \sum_{\mathcal{W}\subseteq[\mathsf{K}]:|\mathcal{W}|>1} v_\mathcal{W} x_\mathcal{W} + \sum_{\mathcal{Q}:|\mathcal{Q}|=b} y_\mathcal{Q}, \tag{A.56}$$

where $x_1=\sum_{\mathcal{W}\subseteq[\mathsf{K}]:|\mathcal{W}|=1} x_\mathcal{W}$ and $v_\mathcal{W}\geq 0$ represents the coefficient of $x_\mathcal{W}$. Then, we focus on (6.5a). It is easy to see that $c_{\mathcal{W}_1,\mathsf{H}-1}=\binom{\mathsf{H}-1}{\mathsf{H}-2}-\mathsf{r}=\mathsf{r}-1$ where $|\mathcal{W}_1|=1$. So in (6.5a), the total coefficient of $x_\emptyset$ is $\mathsf{K}(\mathsf{r}-1)$. Then we focus on a set of user $\mathcal{W}_1\subseteq[\mathsf{K}]$ where $|\mathcal{W}_1|=2$ under the assumption that the two users in $\mathcal{W}_1$ are k_1 and k_2. In (6.5a), there is only one term with coefficient $c_{\mathcal{W}_1,\mathsf{H}-1}$ for each $\mathcal{W}_1\subseteq[\mathsf{K}]$ where $|\mathcal{W}_1|=2$. Now we want to compute $c_{\mathcal{W}_1,\mathsf{H}-1}$ for each $\mathcal{W}_1\subseteq[\mathsf{K}]$ where $|\mathcal{W}_1|=2$. If $\mathcal{H}_{k_1}\cap\mathcal{H}_{k_2}=\emptyset$, we have $c_{\mathcal{W}_1,\mathsf{H}-1}=\max\{\mathsf{H}-1-\mathsf{H},0\}=0$. In addition, there are $\frac{\mathsf{K}}{2}\binom{\mathsf{r}}{0}\binom{\mathsf{r}}{\mathsf{r}}$ such sets. If $|\mathcal{H}_{k_1}\cap\mathcal{H}_{k_2}|=1$, we have $c_{\mathcal{W}_1,\mathsf{H}-1}=\max\{\mathsf{H}-1-\mathsf{H}-1,0\}=0$. There are $\frac{\mathsf{K}}{2}\binom{\mathsf{r}}{1}\binom{\mathsf{r}}{\mathsf{r}-1}$ such sets. If $|\mathcal{H}_{k_1}\cap\mathcal{H}_{k_2}|=i\in[2:\mathsf{r}-1]$, we have $c_{\mathcal{W}_1,\mathsf{H}-1}=\max\{\mathsf{H}-1-(\mathsf{H}-i),0\}=i-1$. There are $\frac{\mathsf{K}}{2}\binom{\mathsf{r}}{i}\binom{\mathsf{r}}{\mathsf{r}-i}$ such sets. Hence, we have

$$\begin{aligned}
\sum_{\mathcal{W}_1\subseteq[\mathsf{K}]:|\mathcal{W}_1|=2} c_{\mathcal{W}_1,\mathsf{H}-1} &= \sum_{i\in[2:\mathsf{r}-1]}(i-1)\frac{\mathsf{K}}{2}\binom{\mathsf{r}}{i}\binom{\mathsf{r}}{\mathsf{r}-i} && \text{(A.57a)}\\
&= \frac{\mathsf{K}}{2}\Big\{\sum_{i\in[0:\mathsf{r}]}(i-1)\binom{\mathsf{r}}{i}\binom{\mathsf{r}}{\mathsf{r}-i}+1-(\mathsf{r}-1)\Big\} && \text{(A.57b)}\\
&= \frac{\mathsf{K}}{2}\Big\{\sum_{i\in[0:\mathsf{r}]} i\binom{\mathsf{r}}{i}\binom{\mathsf{r}}{\mathsf{r}-i}-\binom{2\mathsf{r}}{\mathsf{r}}-\mathsf{r}+2\Big\} && \text{(A.57c)}\\
&= \frac{\mathsf{K}}{2}\Big\{\mathsf{r}\sum_{i\in[0:\mathsf{r}]}\frac{i}{\mathsf{r}}\binom{\mathsf{r}}{i}\binom{\mathsf{r}}{\mathsf{r}-i}-\mathsf{K}-\mathsf{r}+2\Big\} && \text{(A.57d)}\\
&= \frac{\mathsf{K}}{2}\Big\{\mathsf{r}\sum_{i\in[1:\mathsf{r}]}\binom{\mathsf{r}-1}{i-1}\binom{\mathsf{r}}{\mathsf{r}-i}-\mathsf{K}-\mathsf{r}+2\Big\} && \text{(A.57e)}\\
&= \frac{\mathsf{K}}{2}\Big\{\mathsf{r}\binom{2\mathsf{r}-1}{\mathsf{r}}-\mathsf{K}-\mathsf{r}+2\Big\} && \text{(A.57f)}\\
&= \frac{\mathsf{K}}{2}(\mathsf{r}-2)\Big(\frac{\mathsf{K}}{2}-1\Big). && \text{(A.57g)}
\end{aligned}$$

Hence, we can sum all the inequalities as (6.5a) for all the permutations $\mathbf{p}([\mathsf{K}])$ to obtain,

$$\sum_{\mathcal{Q}:|\mathcal{Q}|=b} y_\mathcal{Q} \geq \mathsf{K}(\mathsf{r}-1)x_\emptyset + \frac{1}{2}(\mathsf{r}-2)\Big(\frac{\mathsf{K}}{2}-1\Big)x_1. \tag{A.58}$$

From (A.56) and (A.58), we have

$$\begin{aligned}
&\mathsf{H}(\mathsf{H}-1)\mathsf{R}^\star_{\mathsf{c},\mathsf{u}} && \text{(A.59a)}\\
&\geq \frac{\mathsf{K}}{2}\mathsf{H}x_\emptyset + \frac{\mathsf{H}}{4}(\frac{3}{2}\mathsf{K}-1)x_1 + \sum_{\mathcal{W}\subseteq[\mathsf{K}]:|\mathcal{W}|>1} v_\mathcal{W}x_\mathcal{W} + \mathsf{K}(\mathsf{r}-1)x_\emptyset + \frac{1}{2}(\mathsf{r}-2)(\frac{\mathsf{K}}{2}-1)x_1 && \text{(A.59b)}\\
&= \Big(\frac{\mathsf{K}}{2}\mathsf{H}+\mathsf{K}(\mathsf{r}-1)\Big)x_\emptyset + \frac{\mathsf{H}(\mathsf{K}-1)-(\mathsf{K}-2)}{2}x_1 + \sum_{\mathcal{W}\subseteq[\mathsf{K}]:|\mathcal{W}|>1} v_\mathcal{W}x_\mathcal{W}. && \text{(A.59c)}
\end{aligned}$$

Then, we want to eliminate $x_\emptyset$ and x_1 with the help of (6.3b) and (6.3c). From (6.3b), we have

$$\Big(\frac{\mathsf{K}}{2}\mathsf{H}+\mathsf{K}(\mathsf{r}-1)\Big)(x_\emptyset+x_1)=\Big(\frac{\mathsf{K}}{2}\mathsf{H}+\mathsf{K}(\mathsf{r}-1)\Big)(1-\sum_{\mathcal{W}\subseteq[\mathsf{K}]:|\mathcal{W}|>1} x_{\mathcal{W}}). \tag{A.60}$$

From (6.3c), we have

$$\Big(\frac{\mathsf{K}-\mathsf{H}-\mathsf{K}\mathsf{H}}{2}+1\Big)x_1=\Big(\frac{\mathsf{K}-\mathsf{H}-\mathsf{K}\mathsf{H}}{2}+1\Big)\big(\frac{\mathsf{K}\mathsf{M}}{\mathsf{N}}-\sum_{\mathcal{W}\subseteq[\mathsf{K}]:|\mathcal{W}|>1} x_{\mathcal{W}}\big). \tag{A.61}$$

We take (A.60) and (A.61) into (A.59c) to obtain,

$$\mathsf{H}(\mathsf{H}-1)\mathsf{R}^\star_{\mathsf{c,u}} \geq \mathsf{K}(\mathsf{H}-1)-\Big(\frac{-\mathsf{K}+\mathsf{H}+\mathsf{K}\mathsf{H}}{2}-1\Big)\frac{\mathsf{K}\mathsf{M}}{\mathsf{N}}+\sum_{\mathcal{W}\subseteq[\mathsf{K}]:|\mathcal{W}|>1} z_{\mathcal{W}}x_{\mathcal{W}}, \tag{A.62}$$

where $z_{\mathcal{W}}=v_{\mathcal{W}}+\big(\frac{-\mathsf{K}+\mathsf{H}+\mathsf{K}\mathsf{H}}{2}-1\big)\geq 0$. So we prove (A.55).

A.11 Proof of Theorem 20

By taking $x=\mathsf{H}$ and comparing the cut-set bound under the constraint of uncoded placement and the achievable bounds in (1.7), we can straightforward obtain Theorem 20-1).

A.11.1 Proof of Theorem 20-2), 3), 4)

It can be seen that for each $\mathcal{J}\subseteq[\mathsf{K}]$ where $|\mathcal{J}|=t+1$, $\min_{h\in[\mathsf{H}]}|\mathcal{J}\backslash\mathcal{U}_h|\leq|\mathcal{J}|-1=t$. Hence, from (7.5),

$$\mathsf{R}_{\mathrm{CICS}}\leq\binom{\mathsf{K}}{t+1}\frac{1+t/\mathsf{r}}{\mathsf{H}\binom{\mathsf{K}}{t}}. \tag{A.63}$$

By the cut-set converse bound in (6.2), $\mathsf{R}^\star_{\mathsf{c,u}}$ is lower bounded by the convex hull of $\Big(\frac{\mathsf{N}t}{\mathsf{K}},\frac{\binom{\mathsf{K}}{t+1}}{\mathsf{H}\binom{\mathsf{K}}{t}}\Big)$ for $t\in[0:\mathsf{K}]$. Hence, the scheme in Section 7.2.3 is order optimal within a factor of $1+t/\mathsf{r}$ under the constraint of uncoded cache placement. So we can sequentially prove Theorem 20-2) and 3).

A.11.2 Proof of Theorem 20-5)

To encode each message $W_{\mathcal{J}}$ where $\mathcal{J}\in\mathcal{V}_1$, the total link-load of IES is $|W_{\mathcal{J}}|/\mathsf{B}$. To encode each message $W_{\mathcal{J}}$ where $\mathcal{J}\in\mathcal{V}_2$, we transmit one linear combination including $W_{\mathcal{J}}$ and other $2\mathsf{r}-2$ messages to $2\mathsf{r}$ relays. So the total link-load to transmit $W_{\mathcal{J}}$ is $2\mathsf{r}/(2\mathsf{r}-1)$. Compared to The converse bound in (6.2) with $x=\mathsf{H}$ when $0\leq\mathsf{M}\leq\mathsf{N}/\mathsf{K}$, IES is order optimal within factor $\frac{2\mathsf{r}}{2\mathsf{r}-1}\leq\frac{4}{3}$ under the constraint of uncoded cache placement.

A.12 Proof of Theorem 18

Converse From the cut-set converse bound in (1.9) with $\alpha = 1$ and $l = 1$, we have $\mathsf{R}_\mathsf{c}^\star \geq (1 - \mathsf{M}/\mathsf{N})/\mathsf{r}$. Between $\mathsf{M} \in \left[\frac{(\mathsf{K}-\mathsf{H}+\mathsf{r}-1)\mathsf{N}}{\mathsf{K}}, \mathsf{N}\right]$, the converse bound is a straight line. When $\mathsf{M} = \frac{(\mathsf{K}-\mathsf{H}+\mathsf{r}-1)\mathsf{N}}{\mathsf{K}}$, we have

$$\mathsf{R}_\mathsf{c}^\star \geq \frac{1 - (\mathsf{K} - \mathsf{H} + \mathsf{r} - 1)/\mathsf{K}}{\mathsf{r}} = \frac{\mathsf{H} - \mathsf{r} + 1}{\mathsf{r}\mathsf{K}}. \tag{A.64}$$

When $\mathsf{M} = \mathsf{N}$, we have $\mathsf{R}_\mathsf{c}^\star \geq 0$.

Achievability It is obvious that when $\mathsf{M} = \mathsf{N}$, we can achieve $\mathsf{R} = 0$. So it remains to prove that we can achieve $\mathsf{R} = \frac{\mathsf{H}-\mathsf{r}+1}{\mathsf{r}\mathsf{K}}$ when $\mathsf{M} = \frac{(\mathsf{K}-\mathsf{H}+\mathsf{r}-1)\mathsf{N}}{\mathsf{K}}$. The memory-sharing of these two points can coincide with the converse bound.

We focus on the coded caching gain $g = \mathsf{K}_1$. We consider two cases, $\mathsf{H} = \mathsf{r} + 1$ and $\mathsf{H} > \mathsf{r} + 1$.

When $\mathsf{H} = \mathsf{r}+1$, we can see that $\mathsf{K}_2 + 1 = \binom{\mathsf{r}-1}{\mathsf{r}-2} + 1 = \mathsf{r}$. In addition, $\mathsf{K}_1 = \binom{\mathsf{r}}{\mathsf{r}-1} = \mathsf{r}$. Hence, we have in this case $\mathsf{K}_2 + 1 = \mathsf{K}_1$, which means that we can use the scheme in Section 7.4.1 to achieve the maximum coded caching gain. In this case, $\mathsf{K} = \binom{\mathsf{H}}{\mathsf{r}} = \binom{\mathsf{r}+1}{\mathsf{r}} = \mathsf{r} + 1$ and $\mathsf{K}_1 = \binom{\mathsf{H}-1}{\mathsf{r}-1} = \mathsf{r}$. In addition, we have

$$\mathsf{k}_{1,\mathrm{b}} = \sum_{a=1}^{\mathsf{r}} \binom{\mathsf{r}}{a} \binom{\mathsf{K}_a - 1}{g - 2} (-1)^{a-1} \tag{A.65a}$$

$$= \binom{\mathsf{r}}{1}(\mathsf{K}_1 - 1) - \binom{\mathsf{r}}{2} \tag{A.65b}$$

$$= \binom{\mathsf{r}}{2}, \tag{A.65c}$$

$$\mathsf{k}_{2,\mathrm{b}} = \sum_{a=1}^{\mathsf{r}} \binom{\mathsf{r}}{a} \binom{\mathsf{K}_a - 1}{g - 1} (-1)^{a-1} = \mathsf{r}. \tag{A.65d}$$

Hence, the needed memory size is

$$\mathsf{M}_\mathrm{b} = \mathsf{k}_{1,\mathrm{b}}\mathsf{N}/(\mathsf{k}_{1,\mathrm{b}} + \mathsf{k}_{2,\mathrm{b}}) \tag{A.66a}$$

$$= \frac{\mathsf{N}(\mathsf{K} - 2)}{\mathsf{K}} \tag{A.66b}$$

$$= \frac{(\mathsf{K} - \mathsf{H} + \mathsf{r} - 1)\mathsf{N}}{\mathsf{K}}. \tag{A.66c}$$

The achieved max link-load is

$$\mathsf{R}_\mathrm{b} = \frac{\mathsf{K}(1 - \mathsf{M}_\mathrm{b}/\mathsf{N})}{\mathsf{H}g} \tag{A.67a}$$

$$= \frac{\mathsf{H} - \mathsf{r} + 1}{\mathsf{r}\mathsf{K}}. \tag{A.67b}$$

In addition, we have

$$\mathsf{n}_\mathrm{b} = |\mathcal{Z}_{g-1}| \tag{A.68a}$$

$$= \binom{\mathsf{r}}{1}\mathsf{K}_1 - \binom{\mathsf{r}}{2} \tag{A.68b}$$

$$= \binom{\mathsf{r}}{2} + \mathsf{r}. \tag{A.68c}$$

Hence, in this case we have $\mathsf{n_b} = \mathsf{k}_{1,\mathsf{b}} + \mathsf{k}_{2,\mathsf{b}}$.

When $\mathsf{H} > \mathsf{r}+1$, we have $\mathsf{K}_1 - 1 - \mathsf{K}_2 = \binom{\mathsf{H}-2}{\mathsf{r}-1} - 1 > 0$ and thus $g = \mathsf{K}_1 > \mathsf{K}_2 + 1$. So we should use the asymmetric coded placement scheme in Section 7.4.2. Since $y_{g-1,m} = g - 1 - \mathsf{K}_1 + \binom{\mathsf{H}-m}{\mathsf{r}-1}$, we have when $m = \mathsf{H} - \mathsf{r} + 1$, $y_{g-1,m} = 0$. In addition, when $m = \mathsf{H} - \mathsf{r} + 1$, for any $a \in [1 : \mathsf{H} - m]$, we have $\sum_{b=1}^{a} \binom{a}{b}\binom{\mathsf{H}-m-b}{\mathsf{r}-b-1}(-1)^{b-1} = \sum_{b=1}^{a} \binom{a}{b}(-1)^{b-1} > 0$. So we have when $m = \mathsf{H} - \mathsf{r} + 1$,

$$N_m = \binom{\mathsf{H}}{m}\left(\binom{\binom{\mathsf{H}-m}{\mathsf{r}-1}}{y_{g-1,m}} - \sum_{a=1}^{\mathsf{H}-m}\binom{\mathsf{H}-m}{a}\binom{\binom{\mathsf{H}-m}{\mathsf{r}-1} - \sum_{b=1}^{a}\binom{a}{b}\binom{\mathsf{H}-m-b}{\mathsf{r}-b-1}(-1)^{b-1}}{y_{g-1,m} - \sum_{b=1}^{a}\binom{a}{b}\binom{\mathsf{H}-m-b}{\mathsf{r}-b-1}(-1)^{b-1}}(-1)^{a-1}\right) \tag{A.69a}$$

$$= \binom{\mathsf{H}}{m}\binom{\binom{\mathsf{H}-m}{\mathsf{r}-1}}{y_{g-1,m}} = \binom{\mathsf{H}}{\mathsf{r}-1} \tag{A.69b}$$

In addition, with $\mathsf{H} > \mathsf{r}+1$ we have $\mathsf{K}_1 - 1 > \mathsf{K}_2$. So we have

$$|\mathcal{Z}_{g-1}| = |\mathcal{Z}_{\mathsf{K}_1-1}| \tag{A.70a}$$

$$= \sum_{n=1}^{\mathsf{r}}\binom{\mathsf{H}}{n}\binom{\mathsf{K}_n}{\mathsf{K}_1-1}(-1)^{n-1} \tag{A.70b}$$

$$= \mathsf{H}\mathsf{K}_1. \tag{A.70c}$$

From (A.70c) and (A.69b), we can see that $(\mathsf{H}-\mathsf{r}+1)N_{\mathsf{H}-\mathsf{r}+1} = \mathsf{H}\binom{\mathsf{H}-1}{\mathsf{r}-1} = |\mathcal{Z}_{g-1}|$. Furthermore, we have $\sum_{m=1}^{\mathsf{H}-\mathsf{r}+1} mN_m = |\mathcal{Z}_{g-1}|$. Hence, $N_m = 0$ for $m \in [\mathsf{H}-\mathsf{r}]$. So

$$\mathsf{k}_{1,\mathsf{b}} = \sum_{m=1}^{\mathsf{H}-\mathsf{r}+1} N_m \frac{\mathsf{K} - \binom{\mathsf{H}-m}{\mathsf{r}} - m\mathsf{K}_1 + m(g-1)}{\mathsf{K}} \tag{A.71a}$$

$$= N_{\mathsf{H}-\mathsf{r}+1}\frac{\mathsf{K} - \binom{\mathsf{r}-1}{\mathsf{r}} - (\mathsf{H}-\mathsf{r}+1)\mathsf{K}_1 + (\mathsf{H}-\mathsf{r}+1)(\mathsf{K}_1-1)}{\mathsf{K}} \tag{A.71b}$$

$$= \binom{\mathsf{H}}{\mathsf{r}-1} - \binom{\mathsf{H}}{\mathsf{r}-1}\frac{\mathsf{H}-\mathsf{r}+1}{\mathsf{K}} \tag{A.71c}$$

$$= \binom{\mathsf{H}}{\mathsf{r}-1} - \mathsf{r}, \tag{A.71d}$$

$$\mathsf{k}_{2,\mathsf{b}} = \sum_{a=1}^{\mathsf{r}}\binom{\mathsf{r}}{a}\binom{\mathsf{K}_a - 1}{g-1}(-1)^{a-1} \tag{A.71e}$$

$$= \mathsf{r}. \tag{A.71f}$$

Hence, the needed memory size is

$$\mathsf{M_b} = \mathsf{k}_{1,\mathsf{b}}\mathsf{N}/(\mathsf{k}_{1,\mathsf{b}} + \mathsf{k}_{2,\mathsf{b}}) \tag{A.72a}$$

$$= \frac{\binom{\mathsf{H}}{\mathsf{r}-1} - \mathsf{r}}{\binom{\mathsf{H}}{\mathsf{r}-1}} \tag{A.72b}$$

$$= \frac{(\mathsf{K}-\mathsf{H}+\mathsf{r}-1)\mathsf{N}}{\mathsf{K}}. \tag{A.72c}$$

The achieved max link-load is

$$\mathsf{R}_\mathsf{b} = \frac{\mathsf{K}(1-\mathsf{M}_\mathsf{b}/\mathsf{N})}{\mathsf{H}g} \tag{A.73a}$$

$$= \frac{\mathsf{H}-\mathsf{r}+1}{\mathsf{r}\mathsf{K}}. \tag{A.73b}$$

In addition, in this case,

$$\mathsf{n}_\mathsf{b} = \sum_{m=1}^{\mathsf{H}-\mathsf{r}+1} N_m \tag{A.74a}$$

$$= N_{\mathsf{H}-\mathsf{r}+1} \tag{A.74b}$$

$$= \binom{\mathsf{H}}{\mathsf{r}-1}. \tag{A.74c}$$

Hence, in this case we have $\mathsf{n}_\mathsf{b} = \mathsf{k}_{1,\mathsf{b}} + \mathsf{k}_{2,\mathsf{b}}$.

In conclusion, for both cases we prove Theorem 18.

In addition, for both cases, we have $\mathsf{n}_\mathsf{b} = \mathsf{k}_{1,\mathsf{b}} + \mathsf{k}_{2,\mathsf{b}}$ and thus we need not to use MDS precoding. Hence, when $g = \mathsf{K}_1$, the proposed cache placement is uncoded.

A.13 Proof of Theorem 22

We consider two cases, where $g \in [2 : \mathsf{K}_2 + 1]$ and $g \in [\mathsf{K}_2 + 2, \mathsf{K}_1]$.

If $g \in [2 : \mathsf{K}_2 + 1]$, the asymmetric coded placement scheme in Section 7.4.1 is used. The needed memory size $\mathsf{M}_\mathsf{b} = \frac{(g-1)|\mathcal{Z}_{g-1}|\mathsf{N}}{(g-1)|\mathcal{Z}_{g-1}|+g|\mathcal{Z}_g|}$. It can be seen that $\mathsf{M}_\mathsf{b} < \mathsf{M}_{\mathrm{ZY}} = \frac{(g-1)\mathsf{N}}{\mathsf{K}_1}$ if $\frac{|\mathcal{Z}_g|}{|\mathcal{Z}_{g-1}|} < \frac{\mathsf{K}_1-g+1}{g}$, which is always true because

$$\frac{|\mathcal{Z}_g|}{|\mathcal{Z}_{g-1}|} = \frac{\sum_{n=1}^{\mathsf{r}} \binom{\mathsf{H}}{n}\binom{\mathsf{K}_n}{g}(-1)^{n-1}}{\sum_{n=1}^{\mathsf{r}} \binom{\mathsf{H}}{n}\binom{\mathsf{K}_n}{g-1}(-1)^{n-1}} \tag{A.75a}$$

$$= \frac{\sum_{n=1}^{\mathsf{r}} \binom{\mathsf{H}}{n}\frac{\mathsf{K}_n-g+1}{g}\binom{\mathsf{K}_n}{g-1}(-1)^{n-1}}{\sum_{n=1}^{\mathsf{r}} \binom{\mathsf{H}}{n}\binom{\mathsf{K}_n}{g-1}(-1)^{n-1}} \tag{A.75b}$$

$$< \frac{\sum_{n=1}^{\mathsf{r}} \binom{\mathsf{H}}{n}\frac{\mathsf{K}_1-g+1}{g}\binom{\mathsf{K}_n}{g-1}(-1)^{n-1}}{\sum_{n=1}^{\mathsf{r}} \binom{\mathsf{H}}{n}\binom{\mathsf{K}_n}{g-1}(-1)^{n-1}} \tag{A.75c}$$

$$= \frac{\mathsf{K}_1-g+1}{g}. \tag{A.75d}$$

(A.75d) holds because $\mathsf{K}_2 \geq g-1$ (i.e., $\binom{\mathsf{K}_2}{g-1} \geq 0$).

If $g \in [\mathsf{K}_2+2, \mathsf{K}_1]$, we should use the asymmetric coded placement scheme in Section 7.4.2. In this case, from a similar derivation as (A.75d), we can see that $\mathsf{M}_{\mathrm{ZY}} = \frac{(g-1)|\mathcal{Z}_{g-1}|/\mathsf{K}}{(g-1)|\mathcal{Z}_{g-1}|/\mathsf{K}+g|\mathcal{Z}_g|/\mathsf{K}}$. Hence, it is sufficient to prove that $\mathsf{k}_{1,\mathsf{b}} < (g-1)|\mathcal{Z}_{g-1}|/\mathsf{K}$. From the definition of $\mathsf{k}_{1,\mathsf{b}}$ in (7.47b), it is equivalent to prove that

$$\sum_{m=1}^{\mathsf{H}-\mathsf{r}+1} N_m \frac{\mathsf{K} - \binom{\mathsf{H}-m}{\mathsf{r}} - m\mathsf{K}_1 + m(g-1)}{g-1} < |\mathcal{Z}_{g-1}|. \tag{A.76}$$

From the construction of the asymmetric coded placement scheme in Section 7.4.2, it can be seen that when $g > \mathsf{K}_2 + 1$, there exists at least one $m \in [2 : \mathsf{H}-\mathsf{r}+1]$ such that $N_m > 0$. In addition, $\sum_{m=1} mN_m = |\mathcal{Z}_{g-1}|$. In the following we prove the statement that for $m \in [\mathsf{H}-\mathsf{r}+1]$, we have $\frac{\mathsf{K}-\binom{\mathsf{H}-m}{\mathsf{r}}-m\mathsf{K}_1+m(g-1)}{g-1} \leq m$ and the equality holds only when $m = 1$. With this statement, we can obtain (A.76) and thus we prove Theorem 22. To prove this statement, it is equivalent to prove $\mathsf{K} - \binom{\mathsf{H}-m}{\mathsf{r}} - m\mathsf{K}_1 \leq 0$. In addition,

$$\mathsf{K} - \binom{\mathsf{H}-m}{\mathsf{r}} - m\mathsf{K}_1 = \binom{\mathsf{H}-1}{\mathsf{r}-1} + \cdots + \binom{\mathsf{H}-m}{\mathsf{r}-1} - m\binom{\mathsf{H}-1}{\mathsf{r}-1} \tag{A.77a}$$

$$\leq 0, \tag{A.77b}$$

where the equality holds only when $m = 1$.

A.14 Computation of N_a for $a \in [\mathsf{H}-\mathsf{r}+1]$ in (7.47d) and (7.47e)

Since each created MDS subfile in Hierarchy m where $m \in [\mathsf{H}-\mathsf{r}+1]$ covers m sets in $\mathcal{Z}_{g-1}$, it is obvious to obtain N_1 is given in (7.47e).

In the rest of this proof, we focus on Hierarchy m where $m \in [2 : \mathsf{H}-\mathsf{r}+1]$ and prove that N_m is given in (7.47d). We choose a set of relays $\mathcal{Y}$ where $|\mathcal{Y}| = m$ and we focus on one relay $h \in \mathcal{Y}$. To generate MDS subfiles in Hierarchy, each time we choose one set in

$$\mathcal{C}_{h,\mathcal{Y}} := \{\mathcal{C} \subseteq \mathcal{U}_h \setminus \mathcal{G}_{\mathcal{Y}} : |\mathcal{C}| = (g-1) - |\mathcal{G}_{\mathcal{Y}} \cap \mathcal{U}_h|, (\mathcal{G}_{\mathcal{Y}} \cup \mathcal{C}) \not\supseteq \mathcal{U}_{\{h,h_1\}}, h_1 \in [\mathsf{H}] \setminus \mathcal{Y}\}. \tag{A.78}$$

Hence, we can see that

$$N_m = \binom{\mathsf{H}}{m} |\mathcal{C}_{h,\mathcal{Y}}|, \tag{A.79}$$

and it remains to compute $|\mathcal{C}_{h,\mathcal{Y}}|$. In (7.67), we compute that $|\mathcal{G}_{\mathcal{Y}} \cap \mathcal{U}_h| = \mathsf{K}_1 - \binom{\mathsf{H}-m}{\mathsf{r}-1}$. Hence the length of each element in $\mathcal{C}_{h,\mathcal{Y}}$ is $(g-1) - |\mathcal{G}_{\mathcal{Y}} \cap \mathcal{U}_h| = y_{g-1,m}$ where $y_{t,m}$ is defined in (7.47f). In addition, it can be computed that $|\mathcal{U}_h \setminus \mathcal{G}_{\mathcal{Y}}| = \binom{\mathsf{H}-m}{\mathsf{r}-1}$. So we have that

$$|\mathcal{C}_{h,\mathcal{Y}}| = \binom{\binom{\mathsf{H}-m}{\mathsf{r}-1}}{y_{g-1,m}} - |\mathcal{C}'_{h,\mathcal{Y}}|, \tag{A.80}$$

where

$$\mathcal{C}'_{h,\mathcal{Y}} := \{\mathcal{C} \subseteq \mathcal{U}_h \setminus \mathcal{G}_{\mathcal{Y}} : |\mathcal{C}| = (g-1) - |\mathcal{G}_{\mathcal{Y}} \cap \mathcal{U}_h|, \\ \text{and there exist some relay(s)} h_1 \in [\mathsf{H}] \setminus \mathcal{Y} \text{ such that } (\mathcal{G}_{\mathcal{Y}} \cup \mathcal{C}) \supseteq \mathcal{U}_{\{h,h_1\}}\}. \tag{A.81}$$

It is sufficient to compute $|\mathcal{C}'_{h,\mathcal{Y}}|$. In addition, we define that

$$\mathcal{C}'_{h,\mathcal{Y},a} := \{\mathcal{C} \subseteq \mathcal{U}_h \setminus \mathcal{G}_{\mathcal{Y}} : |\mathcal{C}| = (g-1) - |\mathcal{G}_{\mathcal{Y}} \cap \mathcal{U}_h|, \\ \text{and there exists exactly } a \text{ relays } h_1 \in [\mathsf{H}] \setminus \mathcal{Y} \text{ such that } (\mathcal{G}_{\mathcal{Y}} \cup \mathcal{C}) \supseteq \mathcal{U}_{\{h,h_1\}}\}. \tag{A.82}$$

Hence, by the inclusion-exclusion principle [68, Theorem 10.1], we have

$$|\mathcal{C}'_{h,\mathcal{Y}}| = \sum_{a=1}^{\mathsf{H}-m} \binom{\mathsf{H}-m}{a} |\mathcal{C}'_{h,\mathcal{Y},a}|(-1)^a. \tag{A.83}$$

By the inclusion-exclusion principle [68, Theorem 10.1], we can also compute that for any set $\mathcal{V} \subseteq ([\mathsf{H}] \setminus \mathcal{Y})$ where $|\mathcal{V}| = a$, we have

$$|\bigcup_{h_1 \in \mathcal{V}} \mathcal{U}_{\{h,h_1\}}| = \sum_{b=1}^{a} \binom{a}{b} \binom{\mathsf{H}-m-b}{\mathsf{r}-b-1} (-1)^{b-1}. \tag{A.84}$$

Hence, from (A.84), we have

$$|\mathcal{C}'_{h,\mathcal{Y},a}| = \begin{pmatrix} \binom{\mathsf{H}-m}{\mathsf{r}-1} - \sum_{b=1}^{a} \binom{a}{b} \binom{\mathsf{H}-m-b}{\mathsf{r}-b-1} (-1)^{b-1} \\ y_{g-1,m} - \sum_{b=1}^{a} \binom{a}{b} \binom{\mathsf{H}-m-b}{\mathsf{r}-b-1} (-1)^{b-1} \end{pmatrix}. \tag{A.85}$$

In the end, from (A.85), (A.83), (A.80) and (A.79), we can prove (7.47d).

A.15 Proof of (9.3c) and (9.3f)

We focus on the corner points in Groups 2 and 3. For each coded caching gain $g \in [2 : \mathsf{K}_1]$, it is straightforward to obtain the max link-load from the server to relays is $\frac{\mathsf{K} \max\left\{1-(\mathsf{r}\mathsf{M}^{\text{relay}}+\mathsf{M}^{\text{user}})/\mathsf{N}, 0\right\}}{\mathsf{H}g}$. So it remains to compute the max link-load from relays to users. We consider two cases, when $g \in [2 : \mathsf{K}_2 + 1]$ and $g \in [\mathsf{K}_2 + 2, \mathsf{K}_1]$.

When $g \in [2 : \mathsf{K}_2 + 1]$, in Steps 1 and 3 of delivery phase, for each user $k \in [\mathsf{K}]$, relays in $\mathcal{H}_k$ transmit the uncached bits if F_{d_k} of user k to user k. Hence, the number of bits transmitted from relay relay in $\mathcal{H}_k$ to user k is $\frac{\mathsf{B}(1-\mathsf{M}^{\text{user}}/\mathsf{N})}{\mathsf{r}}$. In Step 2 of delivery phase, we focus on one relay $h \in [\mathsf{H}]$. For each set $\mathcal{W} \subseteq \mathcal{U}_h$ and each user $k \in \mathcal{W}$, where $|\mathcal{W}| = g - 1$ and $|\mathcal{R}_{\mathcal{W}}| > 1$, relay h delivers $f_{d_{k'},\mathcal{W},h}$ to user k where $k' \subseteq ([\mathsf{K}] \setminus \mathcal{U}_h)$ and $\mathcal{R}_{\{k'\}\cup\mathcal{W}} \neq \emptyset$. The length of each $f_{d_{k'},\mathcal{W},h}$ is $\frac{\mathsf{B}}{(\mathsf{k}_{1,\mathsf{b}}+\mathsf{k}_{2,\mathsf{b}}) \max\{y \in [\mathsf{r}] : \mathsf{K}_y \geq g-1\}}$. Moreover, for each such $\mathcal{W}$ and k, the number of users k' where $k' \subseteq ([\mathsf{K}] \setminus \mathcal{U}_h)$ and $\mathcal{R}_{\{k'\}\cup\mathcal{W}} \neq \emptyset$ is $\mathsf{K} - |\{k_1 \in [\mathsf{K}] : \mathcal{H}_{k_1} \cap \mathcal{R}_{\mathcal{W}} = \emptyset\}| - |\mathcal{U}_h| = \mathsf{K} - \binom{\mathsf{H}-|\mathcal{R}_{\mathcal{W}}|}{\mathsf{r}} - \mathsf{K}_1$. Hence, the number of bits transmitted from relay h to each user $k \in \mathcal{U}_h$ is given by the second term of the RHS of (9.3c) and thus we can prove that the max link-load from relays to users is given in (9.3c).

Similarly, when $g \in [\mathsf{K}_2 + 2, \mathsf{K}_1]$, in Steps 1 and 3 of delivery phase, the number of bits transmitted from relay relay in $\mathcal{H}_k$ to each user $k \in [\mathsf{K}]$ is $\frac{\mathsf{B}(1-\mathsf{M}^{\text{user}}/\mathsf{N})}{\mathsf{r}}$. In Step 2 of delivery phase, we focus on one relay $h \in [\mathsf{H}]$. For each set $\mathcal{W} \subseteq \mathcal{U}_h$ and each user $k \in \mathcal{W}$ where $|\mathcal{W}| = g - 1$, assuming the created subfile for each $i \in [\mathsf{N}]$ is $f_{i,\mathcal{W}'}$ by leveraging multicasting opportunities among relays in $\mathcal{Y}$ where $\mathcal{W}' \supseteq \mathcal{W}$, relay h delivers $f_{d_{k'},\mathcal{W}',h}$ to user k where $k' \in \left(\bigcup_{h_1 \in (\mathcal{Y}\setminus\{h\})} \mathcal{U}_{h_1}\right) \setminus \mathcal{W}'$. The length of each $f_{d_{k'},\mathcal{W}',h}$ is $\frac{\mathsf{B}}{(\mathsf{k}_{1,\mathsf{b}}+\mathsf{k}_{2,\mathsf{b}})\mathsf{m}^g_{\max}}$. Moreover, for each such $\mathcal{W}$ and k, the number of users k' where $k' \in \left(\bigcup_{h_1 \in (\mathcal{Y}\setminus\{h\})} \mathcal{U}_{h_1}\right) \setminus \mathcal{W}'$ is $\sum_{h_1 \in (\mathcal{Y}\setminus\{h\})} |\mathcal{U}_{h_1} \setminus \mathcal{W}'| = (|\mathcal{Y}| - 1)(\mathsf{K}_1 - g + 1)$. Hence, the number of bits transmitted from relay h to each user $k \in \mathcal{U}_h$ is given by the second term of the RHS of (9.3f) and thus we can prove that the max link-load from relays to users is given in (9.3f).

A.16 Proof of Theorem 25

If $\mathsf{H} = \mathsf{r}+1$, we focus on Group 2 with $g = \mathsf{K}_2 + 1 = \mathsf{K}_1$. So we have $\max\{y \in [\mathsf{r}] : \mathsf{K}_y \geq g-1\} = 2$. It is computed in Appendix A.12 that in this case $\mathsf{k}_{1,\mathrm{b}} = \binom{\mathsf{r}}{2}$ and $\mathsf{k}_{2,\mathrm{b}} = \mathsf{r}$. Hence, $\mathsf{M}^{\text{relay}} = \frac{\mathsf{N}\mathsf{K}_1}{\mathsf{r}(\mathsf{r}+1)} = \frac{\mathsf{N}}{\mathsf{H}}$ and $\mathsf{M}^{\text{user}} = \frac{(\mathsf{K}-\mathsf{H}+\mathsf{r}-1)\mathsf{N}}{\mathsf{K}} - \frac{\mathsf{N}(\mathsf{K}_1-1)\mathsf{r}}{(\mathsf{H}-\mathsf{r}+1)\binom{\mathsf{H}}{\mathsf{r}-1}}$.

If $\mathsf{H} > \mathsf{r}+1$, we focus on Group 3 with $g = \mathsf{K}_1$. So we have $\mathsf{m}^{\mathsf{K}_1}_{\max} := \max\{m \in [\mathsf{H}-\mathsf{r}+1] : \mathsf{K}_1 - \binom{\mathsf{H}-m}{\mathsf{r}-1} \leq \mathsf{K}_1 - 1\} = \mathsf{H}-\mathsf{r}+1$. It is computed in Appendix A.12 that in this case $\mathsf{k}_{1,\mathrm{b}} = \binom{\mathsf{H}}{\mathsf{r}-1} - \mathsf{r}$ and $\mathsf{k}_{2,\mathrm{b}} = \mathsf{r}$. Hence, $\mathsf{M}^{\text{relay}} = \frac{\mathsf{N}\mathsf{K}_1}{(\mathsf{H}-\mathsf{r}+1)\binom{\mathsf{H}}{\mathsf{r}-1}} = \frac{\mathsf{N}}{\mathsf{H}}$ and $\mathsf{M}^{\text{user}} = \frac{(\mathsf{K}-\mathsf{H}+\mathsf{r}-1)\mathsf{N}}{\mathsf{K}} - \frac{\mathsf{N}(\mathsf{K}_1-1)\mathsf{r}}{(\mathsf{H}-\mathsf{r}+1)\binom{\mathsf{H}}{\mathsf{r}-1}}$.

So we focus on the corner point $(\mathsf{M}^{\text{relay}}, \mathsf{M}^{\text{user}}) = \left(\frac{\mathsf{N}}{\mathsf{H}}, \frac{(\mathsf{K}-\mathsf{H}+\mathsf{r}-1)\mathsf{N}}{\mathsf{K}} - \frac{\mathsf{N}(\mathsf{K}_1-1)\mathsf{r}}{(\mathsf{H}-\mathsf{r}+1)\binom{\mathsf{H}}{\mathsf{r}-1}}\right)$, and the achieved max link-load from the server to relays is $\frac{\mathsf{K}\left(1-(\mathsf{r}\mathsf{M}^{\text{relay}}+\mathsf{M}^{\text{user}})/\mathsf{N}\right)}{\mathsf{H}\mathsf{K}_1} = \frac{\left(1-(\mathsf{r}\mathsf{M}^{\text{relay}}+\mathsf{M}^{\text{user}})/\mathsf{N}\right)}{\mathsf{r}}$, which coincides the converse bound $\mathsf{R}^{\mathsf{s}\to\mathsf{r}\star} \geq \frac{\left(1-(\mathsf{r}\mathsf{M}^{\text{relay}}+\mathsf{M}^{\text{user}})/\mathsf{N}\right)}{\mathsf{r}}$.

In addition, for each memory pair of $\left(0, \frac{(\mathsf{K}-\mathsf{H}+\mathsf{r}-1)\mathsf{N}}{\mathsf{K}}\right)$, $(\mathsf{N}/\mathsf{r}, 0)$ and $(0, \mathsf{N})$, we can also achieve the converse bound $\mathsf{R}^{\mathsf{s}\to\mathsf{r}\star} \geq \frac{\left(1-(\mathsf{r}\mathsf{M}^{\text{relay}}+\mathsf{M}^{\text{user}})/\mathsf{N}\right)}{\mathsf{r}}$. Hence, if a memory pair can be obtained by the memory-sharing of the four memory pairs, $\left(0, \frac{(\mathsf{K}-\mathsf{H}+\mathsf{r}-1)\mathsf{N}}{\mathsf{K}}\right)$, $\left(\frac{\mathsf{N}}{\mathsf{H}}, \frac{(\mathsf{K}-\mathsf{H}+\mathsf{r}-1)\mathsf{N}}{\mathsf{K}} - \frac{\mathsf{N}(\mathsf{K}_1-1)\mathsf{r}}{(\mathsf{H}-\mathsf{r}+1)\binom{\mathsf{H}}{\mathsf{r}-1}}\right)$, $(\mathsf{N}/\mathsf{r}, 0)$ and $(0, \mathsf{N})$, we can achieve the converse bound $\mathsf{R}^{\mathsf{s}\to\mathsf{r}\star} \geq \frac{\left(1-(\mathsf{r}\mathsf{M}^{\text{relay}}+\mathsf{M}^{\text{user}})/\mathsf{N}\right)}{\mathsf{r}}$.

A.17 Pseudo Codes

Algorithm 3 Improved Concatenated Inner Code delivery Scheme (ICICS)

1. **input:** $F_{i,\mathcal{W}}$ where $i \in [\mathsf{N}]$, $\mathcal{W} \subseteq [\mathsf{K}]$ and $|\mathcal{W}| = t$; **initialization:** $\mathcal{X}^h_{\mathcal{W}_1,\mathcal{W}_2} = \emptyset$ for each $h \in [\mathsf{H}]$, $\mathcal{W}_1 \subseteq \mathcal{U}_h$ and $\mathcal{W}_2 \subseteq \mathcal{U}_h \setminus \mathcal{W}_1$;
2. **for** each $\mathcal{J} \subseteq [K]$ where $|\mathcal{J}| = t+1$,
   (a) let $W_{\mathcal{J}} = \oplus_{j\in\mathcal{J}} F_{d_j,\mathcal{J}\setminus\{j\}}$;
   (b) $\mathcal{S}_{\mathcal{J}} = \arg\max_{h\in[\mathsf{H}]} |\mathcal{U}_h \cap \mathcal{J}|$; divide $W_{\mathcal{J}}$ into $|\mathcal{S}_{\mathcal{J}}|$ non-overlapping parts with equal length, $W_{\mathcal{J}} = (W^{|\mathcal{S}_{\mathcal{J}}|}_{\mathcal{J},h} : h \in [\mathsf{H}]$;
   (c) **for** each $h \in \mathcal{S}_{\mathcal{J}}$,
      i. transmit $W^{|\mathcal{S}_{\mathcal{J}}|}_{\mathcal{J},h}$ to relay h;
      ii. **for** each $h' \in [\mathsf{H}]$ where $\mathcal{U}_{h'} \cap (\mathcal{J} \setminus \mathcal{U}_h) \neq \emptyset$, add $|W^{|\mathcal{S}_{\mathcal{J}}|}_{\mathcal{J},h}|/\mathsf{r}$ random linear combinations of $W^{|\mathcal{S}_{\mathcal{J}}|}_{\mathcal{J},h}$ in $\mathcal{X}^{h'}_{\mathcal{A},\mathcal{B}}$ where $\mathcal{A} := \mathcal{U}_{h'} \cap (\mathcal{J}\setminus\mathcal{U}_h) \neq \emptyset$ and $\mathcal{B} := \left\{j \in \mathcal{U}_h \cap \mathcal{U}_{h'} : \mathcal{H}_j \subseteq \mathcal{H}_{\mathcal{A}} \cup \{h\}\right\}$;
3. **for** each $h \in [\mathsf{H}]$,
   (a) for each $\mathcal{J}' \subseteq \mathcal{U}_h$, let $\mathcal{L}^h_{\mathcal{J}'} = \mathrm{RLC}(c, \mathcal{C})$ where $c = \max_{k\in\mathcal{J}'} \sum_{\mathcal{W}_1,\mathcal{W}_2:\mathcal{W}_1\cup\mathcal{W}_2=\mathcal{J}',k\notin\mathcal{W}_2} |\mathcal{X}^h_{\mathcal{W}_1,\mathcal{W}_2}|$ and $\mathcal{C} = \cup_{\mathcal{W}_1,\mathcal{W}_2:\mathcal{W}_1\cup\mathcal{W}_2=\mathcal{J}'} \mathcal{X}^h_{\mathcal{W}_1,\mathcal{W}_2}$;

(b) we transmit RLC$(c', \mathcal{C}')$ to relay h and relay h then forwards RLC$(c', \mathcal{C}')$ to users in $\mathcal{U}_h$, where $c' = \max_{k \in \mathcal{U}_h} \sum_{\mathcal{J}' \subseteq \mathcal{U}_h : \mathcal{L}^h_{\mathcal{J}'} \text{is unknown to } k} |\mathcal{L}^h_{\mathcal{J}'}|$ and $\mathcal{C}' = \cup_{\mathcal{J}' \subseteq \mathcal{U}_h : \mathcal{L}^h_{\mathcal{J}'} \neq \emptyset} \mathcal{L}^h_{\mathcal{J}'}$;

Algorithm 4 Separate Relay Decoding delivery Scheme (SRDS)

---

1. **input:** $F_{i,\mathcal{W}}$ where $i \in [\mathsf{N}]$, $\mathcal{W} \subseteq [\mathsf{K}]$ and $|F_{i,\mathcal{W}}| > 0$; **initialization:** $t_1 = 1$; $\mathcal{T}^h_{k,\mathcal{J}} = \emptyset$ for each $h \in [\mathsf{H}]$, $k \in \mathcal{U}_h$ and $\mathcal{J} \subseteq \mathcal{U}_h \setminus \{k\}$;

2. **for** each $k \in [\mathsf{K}]$ and each $\mathcal{W} \subseteq [\mathsf{K}]$ where $k \notin \mathcal{W}$ and $|F_{d_k,\mathcal{W}}| > 0$,

   (a) $\mathcal{S}_{k,\mathcal{W}} = \arg\max_{h \in \mathcal{H}_k} |\mathcal{W} \cap \mathcal{U}_h|$; divide $F_{d_k,\mathcal{W}}$ into $|\mathcal{S}_{k,\mathcal{W}}|$ non-overlapping parts with equal length, $F_{d_k,\mathcal{W}} = (F^{|\mathcal{S}_{k,\mathcal{W}}|}_{d_k,\mathcal{W},h} : h \in \mathcal{S}_{k,\mathcal{W}})$;

   (b) **for** each $h \in \mathcal{S}_{k,\mathcal{W}}$, pad $F^{|\mathcal{S}_{k,\mathcal{W}}|}_{d_k,\mathcal{W},h}$ at the end of $\mathcal{T}^h_{k,\mathcal{W} \cap \mathcal{U}_h}$;

3. **for** each $h \in [\mathsf{H}]$ and each $\mathcal{J} \subseteq \mathcal{U}_h$ where $|\mathcal{J}| = t_1$,

   (a) $m_1 = \max_{k \in \mathcal{J}} |\mathcal{T}^h_{k,\mathcal{J}\setminus\{k\}}|$;

   (b) **for** each $k \in \mathcal{J}$, **if** $|\mathcal{T}^h_{k,\mathcal{J}\setminus\{k\}}| < m_1$, **then**

      i. $R_e = m_1 - |\mathcal{T}^h_{k,\mathcal{J}\setminus\{k\}}|$; $t_2 = |\mathcal{J}|$; (R_e represents the number of bits to be borrowed.)

      ii. $\mathcal{D} = \left\{\mathcal{W} \subseteq \mathcal{U}_h : k \notin \mathcal{W}, \mathcal{J} \setminus \{k\} \subset \mathcal{W}, |\mathcal{W}| = t_2, \mathcal{T}^h_{k,\mathcal{W}} \neq \emptyset\right\}$; ($\mathcal{D}$ represents the set of bits which can be borrowed.)

      iii. **if** $R_e \geq \sum_{\mathcal{W} \in \mathcal{D}} |\mathcal{T}^h_{k,\mathcal{W}}|$, **then** $\mathcal{C} = \underset{\mathcal{W} \in \mathcal{D}}{\cup} \mathcal{T}^h_{k,\mathcal{W}}$; **else then**

         A. $\mathcal{C} = \emptyset$; sort all the sets $\mathcal{W} \in \mathcal{D}$ by the length of $\mathcal{T}^h_{k,\mathcal{W}}$ such that we let $\mathcal{D}(1)$ represents the set where $|\mathcal{T}^h_{k,\mathcal{D}(1)}| = \max_{\mathcal{W} \in \mathcal{D}} |\mathcal{T}^h_{k,\mathcal{W}}|$ while $\mathcal{D}(|\mathcal{D}|)$ represents the set where $|\mathcal{T}^h_{k,\mathcal{D}(|\mathcal{D}|)}| = \min_{\mathcal{W} \in \mathcal{D}} |\mathcal{T}^h_{k,\mathcal{W}}|$; assume $\mathcal{T}^h_{k,\mathcal{D}(|\mathcal{D}|+1)} = \emptyset$;

         B. a is the minimum number in $\{i \in [2 : |\mathcal{D}|] : \sum_{j \in [1:i-1]} |\mathcal{T}^h_{k,\mathcal{D}(j)}| - (i-1)|\mathcal{T}^h_{k,\mathcal{D}(i)}| \geq R_e\} \cup \{|\mathcal{D}| + 1\}$;

         C. **for** each $i \in [a-1]$, pad the first $|\mathcal{T}^h_{k,\mathcal{D}(i)}| - \frac{\sum_{i \in [a-1]} |\mathcal{T}^h_{k,\mathcal{D}(i)}| - R_e}{a-1}$ bits of $\mathcal{T}^h_{k,\mathcal{D}(i)}$ at the end of $\mathcal{C}$;

      iv. pad the bits in $\mathcal{C}$ at the end of $\mathcal{T}^h_{k,\mathcal{J}\setminus\{k\}}$;

      v. **for** each $\mathcal{W} \in \mathcal{D}$, update $\mathcal{T}^h_{k,\mathcal{W}} = \mathcal{T}^h_{k,\mathcal{W}} \setminus \mathcal{C}$;

      vi. $R_e = R_e - |\mathcal{C}|$; **if** $R_e > 0$ and $t_2 < |\mathcal{U}_h| - 1$, **then** $t_2 = t_2 + 1$ and go to step 3-b-ii);

   (c) let $W^h_{\mathcal{J}} = \underset{k \in \mathcal{J}}{\oplus} \mathcal{T}^h_{k,\mathcal{J}\setminus\{k\}}$;

4. **if** $t_1 < |\mathcal{U}_h|$, $t_1 = t_1 + 1$ and go to step 3);

5. **for** each relay h and each $\mathcal{J} \subseteq \mathcal{U}_h$ where $W^h_{\mathcal{J}} \neq \emptyset$, transmit $W^h_{\mathcal{J}}$ to relay h and relay h transmits $W^h_{\mathcal{J}}$ to each user in $\mathcal{J}$;

Appendix B

Resumé en Français

Au cours de la dernière décennie, il y a eu une augmentation massive des appareils Internet connectés via l'accès sans fil, et une forte augmentation du trafic mobile en raison du streaming multimédia, des applications de navigation Web et réseaux socialement interconnectés. Dans le même temps, l'évolution des appareils mobiles (dans le calcul puissance, taille et vitesse de RAM et de ROM, etc.) a la possibilité ouverte pour plus avancé et sophistiqué techniques de transmission pour réduire le trafic réseau. De plus, les demandes des utilisateurs peuvent se concentrer sur un nombre relativement limité de fichiers (par exemple, les derniers films et musiques). De plus, dans systèmes de communication réels, la forte variabilité temporelle du trafic sur le réseau entraîne des congestions les heures de pointe et la sous-utilisation du réseau pendant les heures creuses. Tout ce qui précède motive l'utilisation de mise en cache. Le cache est un composant réseau qui exploite la mémoire de l'appareil pour stocker les données de façon transparente afin que les futurs les demandes pour ces données peuvent être servies plus rapidement. La mise en cache réduit le trafic de pointe en tirant parti des souvenirs distribué sur le réseau pour dupliquer le contenu pendant les heures creuses. Ce faisant, mettre en cache efficacement permet de déplacer le trafic des heures de pointe aux heures de pointe, ce qui atténue la variabilité du trafic et réduit congestion. Deux phases sont incluses dans un système de mise en cache: i) phase de placement: chaque utilisateur stocke des bits dans son cache sans connaissance des demandes ultérieures; ii) phase de livraison: après que chaque utilisateur a fait sa demande et selon le contenu de la mémoire cache, le serveur transmet les paquets afin de satisfaire les demandes de l'utilisateur. L'objectif est de minimiser le nombre de bits transmis (charger) de sorte que les demandes des utilisateurs puissent être satisfaites.

Dans cette thèse, nous avons étudié le problème de cache codée en construisant la connexion entre le problème de cache codée avec placement non-codé et codage d'index, et en tirant parti des résultats de codage d'index pour caractériser les limites fondamentales du problème de cache codée. Nous avons principalement analysé le problème de cache codée dans le modèle de diffusion à liaison partagée et dans les réseaux combinés. Dans la première partie de cette thèse, pour les réseaux de diffusion de liens partagés, nous avons considéré la contrainte que le contenu placé dans les caches est non-codé. Lorsque le contenu du cache est non-codé et que les demandes de l'utilisateur sont révélées, le problème de cache peut être lié à un problème de codage d'index. Nous avons dérivé des limites fondamentales pour le problème de cache en utilisant des outils pour le problème de codage d'index. Nous avons dérivé un nouveau schéma réalisable de codage d'index en base d'un codage de source distribué. Cette borne interne est strictement meilleure que la borne interne du codage composite largement utilisée. Pour le problème de cache centralisée, une borne externe sous la contrainte de placement de cache non-codé est proposée en base de une borne externe "acyclic" de codage d'index. Il est prouvé que cette borne externe est atteinte par le schéma cMAN lorsque

le nombre de fichiers n'est pas inférieur au nombre d'utilisateurs, et par le nouveau schéma proposé pour le codage d'index, sinon. Pour le problème de cache décentralisée, cette thÃšse propose une borne externe sous la contrainte que chaque utilisateur stocke des bits uniformément et indépendamment au hasard. Cette borne externe est atteinte par le schéma dMAN lorsque le nombre de fichiers n'est pas inférieur au nombre d'utilisateurs, et par notre codage d'index proposé autrement. Dans la deuxiÃšme partie de cette thèse, nous avons considéré le problème de cache dans les rèseaux de relais, où le serveur communique avec les utilisateurs aidés par le cache via certains relais intermédiaires. En raison de la dureté de l'analyse sur les réseaux généraux, nous avons principalement considéré un réseau de relais symétrique bien connu, 'réseaux de combinaison', y compris H relais et $\binom{H}{r}$ utilisateurs où chaque utilisateur est connecté à un r-sous-ensemble de relais différent. Nous avons cherché à minimiser la charge de liaison maximale pour les cas les plus défavorables. Nous avons dérivé des bornes externes et internes dans cette thèse. Pour la borne externes, la méthode directe est que chaque fois que nous considérons une coupure de x relais et que la charge totale transmise à ces x relais peut être limitée à l'extérieur par la borne externes du modèle de lien partagé, y compris $\binom{x}{r}$ utilisateurs. Nous avons utilisé cette stratégie pour étendre les bornes externes du modèle de lien partagé et la borne externe "acyclic" aux réseaux de combinaison. Dans cette thèse, nous avons également resserré la borne externe "acyclic" dans les réseaux de combinaison en exploitant davantage la topologie du réseau et l'entropie conjointe des diverses variables aléatoires. Pour les schémas réalisables, il existe deux approches, la séparation et la non-séparation. De plus, nous avons étendu nos résultats à des modàles plus généraux, tels que des réseaux combinés où tous les relais et utilisateurs sont équipés par cache, et des systèmes de cache dans des réseaux relais plus généraux. Les résultats d'optimisation ont été donnés sous certaines contraintes et les évaluations numériques ont montré que nos schémas proposés surpassent l'état de l'art.

Bibliography

[1] Cisco, "The zettabyte era-trends and analysis", 2013.

[2] ——, "Cisco visual networking index: Global mobile data traffic forecast update, 2013-2018", *White Paper, [Online] http://goo.gl/l77HAJ*, 2014.

[3] E. Bastug, M. Bennis, and M. Debbah, "Living on the edge: The role of proactive caching in 5g wireless networks", *IEEE Communications Magazine*, vol. 52, pp. 82–89, 8 Aug. 2014.

[4] M. A. Maddah-Ali and U. Niesen, "Fundamental limits of caching", *IEEE Trans. Infor. Theory*, vol. 60, no. 5, pp. 2856–2867, May 2014.

[5] ——, "Decentralized coded caching attains order-optimal memory-rate tradeoff", *arXiv:1301.5848v3*, Mar. 2014.

[6] Y. Birk and T. Kol, "Informed source coding on demand (iscod) over broadcast channels", *in Proc. IEEE Conf. Comput. Commun.*, pp. 1257–1264, 1998.

[7] N. Golrezaei, K. Shanmugam, A. G. Dimakis, A. F. Molisch, and G. Caire, "Femtocaching: Wireless video content delivery through distributed caching helpers", *IEEE Trans. Inf. Theory*, vol. 59, no. 12, pp. 8402–8413, Dec. 2013.

[8] S. P. Shariatpanahi, S. A. Motahari, and B. H. Khalaj, "Multi-server coded caching", *IEEE Trans. Infor. Theory*, vol. 62, pp. 7253 –7271, 12 Dec. 2016.

[9] N. Mital, D. Gunduz, and C. Ling, "Coded caching in a multi-server system with random topology", *arXiv:1712.00649*, Dec. 2017.

[10] N. Naderializadeh, M. A. Maddah-Ali, and S. Avestimehr, "On the optimality of separation between caching and delivery in general cache networks", *in IEEE Int. Symp. Inf. Theory (ISIT)*, Jun. 2017.

[11] N. Karamchandani, U. Niesen, M. A. Maddah-Ali, and S. Diggavi, "Hierarchical coded caching", *in IEEE Int. Symp. Inf. Theory (ISIT)*, pp. 2142–2146, Jun. 2014.

[12] M. Ji, M. F. Wong, A. M. Tulino, J. Llorca, G. Caire, M. Effros, and M. Langberg, "On the fundamental limits of caching in combination networks", *IEEE 16th Int. Workshop on Sig. Processing Advances in Wireless Commun.*, pp. 695–699, 2015.

[13] M. Ji, G. Caire, and A. Molisch, "Fundamental limits of caching in wireless d2d networks", *IEEE Trans. Inf. Theory*, vol. 62, no. 1, pp. 849–869, 2016.

[14] E. Bastug, M. Bennis, M. Kountouris, and M. Debbah, "Cache-enabled small cell networks: Modeling and tradeoffs", *EURASIP Journal on Wireless Communications and Networking*, vol. 2015, no. 1, pp. 1–11, Feb. 2015.

[15] M. A. Maddah-Ali and U. Niesen, "Cache-aided interference channels", *in IEEE Int. Symp. Inf. Theory (ISIT)*, June. 2015.

[16] C. K. Ngai and R. W. Yeung, "Network coding gain of combination networks", *IEEE Inf. Theory Workshop*, pp. 283–287, Oct. 2004.

[17] C.-Y. Wang, S. H. Lim, and M. Gastpar, "A new converse bound for coded caching", *available at arXiv:1601.05690*, Jan. 2016.

[18] M. M. Amiri, Q. Yang, and D. Gunduz, "Coded caching for a large number of users", *arXiv:1605.01993*, May 2016.

[19] Q. Yu, M. A. Maddah-Ali, and S. Avestimehr, "The exact rate-memory tradeoff for caching with uncoded prefetching", *in IEEE Int. Symp. Inf. Theory (ISIT)*, Jun. 2017.

[20] K. Wan, D. Tuninetti, and P. Piantanida, "On the optimality of uncoded cache placement", *in Proceedings of the IEEE Information Theory Workshop (ITW)*, pp. 161–165, Sep. 2016.

[21] ——, "On caching with more users than files", *in IEEE Int. Symp. Inf. Theory (ISIT)*, Jul. 2016.

[22] H. Ghasemi and A. Ramamoorthy, "Further results on lower bounds for coded caching", *in IEEE Int. Symp. Inf. Theory (ISIT)*, pp. 2319 –2323, Jul. 2016.

[23] N. Ajaykrishnan, N. S. Prem, V. M. Prabhakaran, and R. Vaze, "Critical database size for effective caching", *arXiv:1501.02549*, Jan. 2015.

[24] A. Sengupta, R. Tandon, and T. C. Clancy, "Improved approximation of storage-rate tradeoff for caching via new outer bounds", *in IEEE Int. Symp. Inf. Theory (ISIT)*, June. 2015.

[25] C. Tian, "Symmetry, demand types and outer bounds in caching systems", *in IEEE Int. Symp. Inf. Theory (ISIT)*, pp. 825–829, Jul. 2016.

[26] T. M. Cover and J. A. Thomas, "Elements of information theory", *2nd ed. New York, NY, USA: Wiley-interscience*, 2006.

[27] Q. Yu, M. A. Maddah-Ali, and S. Avestimehr, "Characterizing the rate-memory tradeoff in cache networks within a factor of 2", *in IEEE Int. Symp. Inf. Theory (ISIT)*, Jun. 2017.

[28] Z. Chen, P. Fan, and K. B. Letaief, "Fundamental limits of caching: Improved bounds for small buffer users", *IET Commun.*, vol. 10, no. 17, pp. 2315–2318, Nov. 2016.

[29] C. Tian and J. Chen, "Caching and delivery via interference elimination", *in IEEE Int. Symp. Inf. Theory (ISIT)*, pp. 830–834, Jul. 2016.

[30] S. Sahraei and M. Gastpar, "K users caching two files: An improved achievable rate", *arXiv:1512.06682*, Dec. 2015.

[31] M. M. Amiri and D. Gunduz, "Fundamental limits of coded caching: Improved delivery rate-cache capacity tradeoff", *IEEE Trans. Communications*, vol. 65, no. 2, pp. 806–815, Feb. 2017.

[32] J. Gomez-Vilardebo, "Fundamental limits of caching: Improved bounds with coded prefetching", *available at arXiv:1612.09071*, Jan 2017.

[33] K. Wan, "On caching with uncoded cache placement", *Ph.D. midterm defence report*, Jun. 2016, preliminary defense held during ISIT 2016.

[34] A. Blasiak, R. Kleinberg, and E. Lubetzky, "Lexicographic products and the power of non-linear network coding", *in IEEE 52nd Annu. Symp. Found. Comput. Sci.*, pp. 609–618, Oct. 2011.

[35] F. Arbabjolfaei, B. Bandemer, Y.-H. Kim, E. Sasoglu, and L. Wang, "On the capacity region for index coding", *in IEEE Int. Symp. Inf. Theory (ISIT)*, Jul. 2013.

[36] Z. Bar-Yossef, Y. Birk, T. S. Jayram, and T. Kol, "Index coding with side information", *IEEE Trans. Infor. Theory*, vol. 57, no. 7, pp. 1479–1494, Mar. 2011.

[37] H. Maleki, V. Cadambe, and S. Jafar, "Index coding an interference alignment perspective", *IEEE Trans. Infor. Theory*, vol. 60, no. 9, pp. 5402–5432, Sep. 2014.

[38] S. A. Jafar, "Topological interference management through index coding", *IEEE Trans. Infor. Theory*, vol. 60, no. 1, pp. 529–568, Jan. 2014.

[39] H. Witsenhausen, "The zero-error side information problem and chromatic numbers", *IEEE Trans. Infor. Theory*, vol. 22, pp. 592–593, 5 Sep. 1976.

[40] K. Shanmugam, A. G. Dimakis, and M. Langberg, "Local graph coloring and index coding", *in IEEE Int. Symp. Inf. Theory (ISIT)*, Jul. 2013.

[41] A. Agarwal and A. Mazumdar, "Local partial clique covers for index coding", *in Proceedings of the IEEE Globecom Workshops (GCW)*, Dec. 2016.

[42] C. Thapa, L. Ong, and S. J. Johnson, "Interlinked cycles for index coding: Generalizing cycles and cliques", *IEEE Trans. Infor. Theory*, vol. 63, no. 6, pp. 3692–3711, Jun. 2017.

[43] S. Unal and A. B. Wagner, "A rate-distortion approach to index coding", *in presented in IEEE Infor. Theory Application Workshop (ITA)*, Feb. 2014.

[44] S. Miyake and J. Muramatsu, "Index coding over correlated sources", *in Proc. IEEE Int. Symp. Net. Coding*, Jun. 2015.

[45] S. H. Lim, C.-Y. Wang, and M. Gastpar, "Information theoretic caching: The multi-user case", *in IEEE Int. Symp. Inf. Theory (ISIT)*, Jul. 2016.

[46] C. Heegard and T. Berger, "Rate distortion when side information may be absent", *IEEE Trans. Infor. Theory*, vol. 31, pp. 727–734, 6 Nov. 1985.

[47] D. Slepian and J. K. Wolf, "Noiseless coding of correlated information sources", *IEEE Trans. Infor. Theory*, vol. 19, no. 4, pp. 471–480, Jul. 1973.

[48] T. S. Han, "The capacity region of general multiple-access channel with certain correlated sources", *Inf. Contr.*, vol. 40, no. 1, pp. 37–60, 1979.

[49] L. Tang and A. Ramamoorthy, "Coded caching for networks with the resolvability property", *in IEEE Int. Symp. Inf. Theory (ISIT)*, Jul. 2016.

[50] A. A. Zewail and A. Yener, "Coded caching for combination networks with cache-aided relays", *in IEEE Int. Symp. Inf. Theory (ISIT)*, pp. 2438–2442, June 2017.

[51] Q. Yan, M. Cheng, X. Tang, and Q. Chen, "On the placement delivery array design in centralized coded caching scheme", *IEEE Trans. Infor. Theory*, vol. 63, no. 9, pp. 5821–5833, Sep. 2017.

[52] Q. Yan, M. Wigger, and S. Yang, "Placement delivery array design for combination networks with edge caching", *arXiv:1801.03048*, Jan. 2018.

[53] K. Wan, D. Tuninetti, and P. Piantanida, "An index coding approach to caching with uncoded cache placement", *submitted to IEEE Trans. Infor. Theory*, 2017.

[54] K. Wan, M. Ji, P. Piantanida, and D. Tuninetti, "Caching in combination networks: Novel asymmetric coded cache placement and multicast message generation by leveraging network topology", *submitted to IEEE Trans. Infor. Theory*, 2018.

[55] ——, "Novel outer bounds and inner bounds with uncoded cache placement for combination networks with end-user-caches", *submitted to IEEE Trans. Infor. Theory*, 2018.

[56] ——, "On the benefits of asymmetric coded cache placement in combination networks with end-user caches", *in IEEE Int. Symp. Inf. Theory (ISIT)*, Jun. 2018.

[57] ——, "Caching in combination networks: Novel multicast message generation and delivery by leveraging the network topology", *in IEEE Intern. Conf. Commun. (ICC)*, May 2018.

[58] K. Wan, D. Tuninetti, P. Piantanida, and M. Ji, "On combination networks with cache-aided relays and users", *in Proceedings of Workshop on Smart Antennas (WSA)*, Mar. 2018.

[59] ——, "A novel asymmetric coded placement in combination networks with end-user caches", *in Proceedings of IEEE Infor. Theory Application Workshop (ITA)*, Feb. 2018.

[60] K. Wan, M. Ji, P. Piantanida, and D. Tuninetti, "Novel outer bounds for combination networks with end-user-caches", *in Proceedings of the IEEE Information Theory Workshop (ITW)*, Nov. 2017.

[61] K. Wan, D. Tuninetti, M. Ji, and P. Piantanida, "State-of-the-art in cache-aided combination networks", *in Proceedings of the 2017 IEEE Asilomar Conf.*, Nov. 2017.

[62] K. Wan, M. Ji, P. Piantanida, and D. Tuninetti, "Novel inner bounds for combination networks with end-user-caches", *in Proceedings of the 55th Allerton Conf. Commun., Control, Comp.*, Oct. 2017.

[63] K. Wan, D. Tuninetti, and P. Piantanida, "Novel delivery schemes for decentralized coded caching in the finite file size regime", *in Proceedings of IEEE Intern. Conf. Commun. Workshops (ICCW)*, May. 2017.

[64] K. Wan, M. Ji, P. Piantanida, and D. Tuninetti, "A novel index coding scheme and its application to coded caching", *in Proceedings of IEEE Infor. Theory Application Workshop (ITA)*, Feb. 2017.

[65] ——, "Caching in combination networks: A novel delivery by leveraging the network topology", *submitted to IEEE Inf. Theory Workshop (ITW), available at arXiv:1802.10479*, Feb. 2018.

[66] B. Bandemer, A. E. Gamal, and Y. H. Kim, "Simultaneous nonunique decoding is rate-optimal", *in Proc. 50th Allerton Conf.*, pp. 9–16, Oct. 2012.

[67] Y. Liu, P. Sadeghi, F. Arbabjolfaei, and Y.-H. Kim, "On the capacity for distributed index coding", *in IEEE Int. Symp. Inf. Theory (ISIT)*, Jun. 2017.

[68] J. H. V. Lint and R. M. Wilson, "A course in combinatorics (second edition)", *Cambridge University Press, ISBN 9780521803403*, 2001.

[69] A. Ramakrishnan, C. Westphal, and A. Markopoulou, "An efficient delivery scheme for coded caching", *in 27th Int. Tel. Cong. (ITC 27)*, pp. 46–54, Sept. 2015.

[70] S. Mohajer and M. A. Maddah-Ali, "Erasure coding for decentralized caching", *in presented in IEEE Infor. Theory Application Workshop (ITA)*, Feb. 2017.

[71] S. Li, M. A. Maddah-Ali, Q. Yu, and A. S. Avestimehr, "A fundamental tradeoff between computation and communication in distributed computing", *IEEE Trans. Inf. Theory*, vol. PP (99), Sep. 2017.

[72] A. E. Gamal and Y. H. Kim, "Network information theory", *Cambridge University Press*, 2011.

[73] P. Petecki, "On cyclic hamiltonian decompositions of complete k-uniform hypergraphs", *Discrete Math.*, 325:74–76, 2014.

[74] C. F. Gauss, "Disquisitiones arithmeticae (English edition)", *New York: Springer, ISBN 978-0-387-96254-2*, 1986.

université PARIS-SACLAY
ÉCOLE DOCTORALE
Sciences et technologies de l'information et de la communication (STIC)

Titre : Limites Fondamentales de Stockage pour les réseaux de diffusion de liens partagés et les réseaux de combinaison

Mots clés : Cache, Codage d'index, Théorie d'information, Réseaux de combinaison

Résumé : Dans cette thèse, nous avons étudié le problème de cache codée en construisant la connexion entre le problème de cache codée avec placement non-codé et codage d'index, et en tirant parti des résultats de codage d'index pour caractériser les limites fondamentales du problème de cache codée. Nous avons principalement analysé le problème de cache codée dans le modèle de diffusion à liaison partagée et dans les réseaux combinés.
Dans la première partie de cette thèse, pour les réseaux de diffusion de liens partagés, nous avons considéré la contrainte que le contenu placé dans les caches est non-codé. Lorsque le contenu du cache est non-codé et que les demandes de l'utilisateur sont révélées, le problème de cache peut être lié à un problème de codage d'index. Nous avons dérivé des limites fondamentales pour le problème de cache en utilisant des outils pour le problème de codage d'index. Nous avons dérivé un nouveau schéma réalisable de codage d'index en base d'un codage de source distribué. Cette borne interne est strictement meilleure que la borne interne du codage composite largement utilisée. Pour le problème de cache centralisée, une borne externe sous la contrainte de placement de cache non-codé est proposée en base de une borne externe "acyclic" de codage d'index. Il est prouvé que cette borne externe est atteinte par le schéma cMAN lorsque le nombre de fichiers n'est pas inférieur au nombre d'utilisateurs, et par le nouveau schéma proposé pour le codage d'index, sinon. Pour le problème de cache décentralisée, cette thèse propose une borne externe sous la contrainte que chaque utilisateur stocke des bits uniformément et indépendamment au hasard. Cette borne externe est atteinte par le schéma dMAN lorsque le nombre de fichiers n'est pas inférieur au nombre d'utilisateurs, et par notre codage d'index proposé autrement.
Dans la deuxième partie de cette thèse, nous avons considéré le problème de cache dans les réseaux de relais, où le serveur communique avec les utilisateurs aidés par le cache via certains relais intermédiaires. En raison de la dureté de l'analyse sur les réseaux généraux, nous avons principalement considéré un réseau de relais symétrique bien connu, `réseaux de combinaison', y compris H relais et binom {H} {r} utilisateurs où chaque utilisateur est connecté à un r-sous-ensemble de relais différent. Nous avons cherché à minimiser la charge de liaison maximale pour les cas les plus défavorables. Nous avons dérivé des bornes externes et internes dans cette thèse. Pour la borne externes, la méthode directe est que chaque fois que nous considérons une coupure de x relais et que la charge totale transmise à ces x relais peut être limitée à l'extérieur par la borne externes du modèle de lien partagé, y compris binom {x} {r} utilisateurs. Nous avons utilisé cette stratégie pour étendre les bornes externes du modèle de lien partagé et la borne externe "acyclic" aux réseaux de combinaison. Dans cette thèse, nous avons également resserré la borne externe "acyclic" dans les réseaux de combinaison en exploitant davantage la topologie du réseau et l'entropie conjointe des diverses variables aléatoires. Pour les schémas réalisables, il existe deux approches, la séparation et la non-séparation. De plus, nous avons étendu nos résultats à des modèles plus généraux, tels que des réseaux combinés où tous les relais et utilisateurs sont équipés par cache, et des systèmes de cache dans des réseaux relais plus généraux. Les résultats d'optimisation ont été donnés sous certaines contraintes et les évaluations numériques ont montré que nos schémas proposés surpassent l'état de l'art.

Université Paris-Saclay
Espace Technologique / Immeuble Discovery
Route de l'Orme aux Merisiers RD 128 / 91190 Saint-Aubin, France

Title : Fundamental Limits of Cache-aided Shared-link Broadcast Networks and Combination Networks

Keywords : Caching, Index Coding, Information Theory, Combination Networks

Abstract : In this thesis, we investigated the coded caching problem by building the connection between coded caching with uncoded placement and index coding, and leveraging the index coding results to characterize the fundamental limits of coded caching problem. We mainly analysed the caching problem in shared-link broadcast model and in combination networks.

In the first part of this thesis, for cache-aided shared-link broadcast networks, we considered the constraint that content is placed uncoded within the caches. When the cache contents are uncoded and the user demands are revealed, the caching problem can be connected to an index coding problem. We derived fundamental limits for the caching problem by using tools for the index coding problem. A novel index coding achievable scheme was first derived based on distributed source coding. This inner bound was proved to be strictly better than the widely used "composite (index) coding" inner bound by leveraging the ignored correlation among composites and the non-unique decoding. For the centralized caching problem, an outer bound under the constraint of uncoded cache placement is proposed based on the " acyclic index coding outer bound". This outer bound is proved to be achieved by the cMAN scheme when the number of files is not less than the number of users, and by the proposed novel index coding achievable scheme otherwise. For the decentralized caching problem, this thesis proposes an outer bound under the constraint that each user stores bits uniformly and independently at random. This outer bound is achieved by dMAN when the number of files is not less than the number of users, and by our proposed novel index coding inner bound otherwise.

In the second part of this thesis, we considered the centralized caching problem in two-hop relay networks, where the server communicates with cache-aided users through some intermediate relays. Because of the hardness of analysis on the general networks, we mainly considered a well-known symmetric relay networks, combination networks, including H relays and binom{H}{r} users where each user is connected to a different r-subset of relays. We aimed to minimize the max link-load for the worst cases. We derived outer and inner bounds in this thesis. For the outer bound, the straightforward way is that each time we consider a cut of x relays and the total load transmitted to these x relays could be outer bounded by the outer bound for the shared-link model including binom{x}{r} users. We used this strategy to extend the outer bounds for the shared-link model and the acyclic index coding outer bound to combination networks. In this thesis, we also tightened the extended acyclic index coding outer bound in combination networks by further leveraging the network topology and joint entropy of the various random variables. For the achievable schemes, there are two approaches, separation and non-separation. In the separation approach, we use cMAN cache placement and multicast message generation independent of the network topology. We then deliver cMAN multicast messages based on the network topology. In the non-separation approach, we design the placement and/or the multicast messages on the network topology. We proposed four delivery schemes on separation approach. On non-separation approach, firstly for any uncoded cache placement, we proposed a delivery scheme by generating multicast messages on network topology. Moreover, we also extended our results to more general models, such as combination networks with cache-aided relays and users, and caching systems in more general relay networks. Optimality results were given under some constraints and numerical evaluations showed that our proposed schemes outperform the state-of-the-art.

Université Paris-Saclay
Espace Technologique / Immeuble Discovery
Route de l'Orme aux Merisiers RD 128 / 91190 Saint-Aubin, France