Limites Fondamentales De Stockage Dans Les Réseaux Sans Fil

Asma Ghorbel

université PARIS-SACLAY

NNT : 2018SACLC031

CentraleSupélec

Limites Fondamentales De Stockage Dans Les Réseaux Sans Fil

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

École doctorale n°580
Sciences et technologies de l'information et de la communication (STIC)
Spécialité : Réseaux, Information et Communications

Thèse présentée et soutenue à Gif-sur-Yvette, le 13 Avril 2018, par

Asma Ghorbel

Composition du Jury :

Pierre DUHAMEL Directeur de recherches, CNRS	Président
Giuseppe CAIRE Professeur, Technical University Berlin	Rapporteur
Petros ELIA Professeur, Eurecom	Rapporteur
Michèle WIGGER Professeur, Télécom ParisTech	Examinateur
Abdellatif ZAIDI Professeur, Huawei Technologies	Examinateur
Apostolos DESTOUNIS Docteur ingénieur, Huawei Technologies	Examinateur
Mari KOBAYASHI Professeur, CentraleSupélec	Directeur de thèse
Sheng YANG Professeur, CentraleSupélec	Co-Directeur de thèse

Thèse de doctorat

Remerciement

Je remercie chaleureusement toutes les personnes qui m'ont aidé pendant l'élaboration de ma thèse et notamment ma directrice Professeur Mari Kobayashi, pour la rigueur avec laquelle elle m'a encadré ma thèse, sa grande disponibilité et ses nombreux conseils durant la rédaction de ma thèse. Je remercie également mon Co-Directeur Professeur Sheng Yang, pour son intérêt et son soutien.

Mes remerciements vont également aux membres de mon jury, Professeur Petros Elia, Professeur Giuseppe Caire, Directeur Pierre Duhamel, Professeur Michèle Wigger, Professeur Abdellatif Zaidi et Docteur Apostolos Destounis.

Huawei Technologies a rendu possible ce travail grâce au financement qu'il a apporté à CentraleSupélec.

Je tiens à remercier Georgios Paschos, Apostolos Destounis et Professeur Richard Combes pour les collaborations fructueuses que nous avons eues.

Je remercie mes collègues de CentraleSupélec avec qui j'ai eu des discussions constructives au cours de mon travail, notamment Hoang et Chao. J'ai partagé des moments enrichissants avec de nombreux autres, en particulier Jane, Salah, Maialen,Laura, Hafiz.

Un grand merci à tous ceux qui m'ont accompagné, à tous ceux qui étaient présents pour ma soutenance, notamment ma mère, mon mari, ma tente Souhaira, mon cousin Yassine, mes collègues et mes amies Marwa et Imen. Un grand merci à tous ceux qui ont pensé à moi à travers les nombreux messages que j'ai reçus, en particulier Houda, Islem, Tessnim, Wiem Samoud et Wiem Ouali, Abir, Sahar.

Un grand merci à ma famille qui m'a gratifié de son amour et fourni les motivations. Je leur adresse toute ma gratitude du fond du coeur.

Enfin, un très grand merci à ma grand mère qui m'a accompagné par ses prières. Que Dieu ait son âme dans sa sainte miséricorde.

Abstract

Caching, i.e. storing popular contents at caches available at end users, has received a significant interest as a technique to reduce the peak traffic in wireless networks. In particular, coded caching proposed by Maddah-Ali and Niesen has been considered as a promising approach to achieve a constant delivery time as the dimension grows, yielding a scalable system. Albeit conceptually appealing, several limitations prevent its straightforward applications in practical wireless systems. Throughout the thesis, we address the limitations of classical coded caching in various wireless channels. Then, we propose novel delivery schemes that exploit opportunistically the underlying wireless channels while preserving partly the promising gain of coded caching.

In the first part of the thesis, we study the achievable rate region of the erasure broadcast channel (EBC) with cache and state feedback. Based on Wang and Gatzianas scheme, we propose an achievable scheme that exploits multicasting opportunities created by receiver side information both from local cache and overhearing. We prove that our proposed delivery scheme achieves the optimal rate region for special cases of interest. Using the interesting duality between the EBC and the multi-antenna broadcast channel, these results are generalized to the multi-antenna broadcast channel with state feedback.

In the second part, we study the content delivery over asymmetric block-fading broadcast channels, where the channel quality varies across users and time. Assuming that user requests arrive dynamically, we design an online scheme based on queuing structure to deal jointly with admission control, files combinations, as well as scheduling. In the short-term, we allow transmissions to subsets of users with good channel quality, avoiding users with fades, while in the long-term we ensure fairness among users. We prove that our online delivery scheme maximizes the alpha-fair utility among all schemes restricted to decentralized cache placement. The performance analysis built on the Lyapunov theory.

In the last part, we study opportunistic scheduling over the asymmetric fading broadcast channel. Under this setting, we aim to design a scalable delivery scheme while ensuring fairness among users. To capture these two contrasted measures, we formulate our objective function by an alpha-fairness family of concave utility functions and we use the Gradient descent scheduler (GDS). We propose a simple threshold-based scheduling policy of linear complexity that does not require the exact channel state information but only a one-bit feedback from each user. We prove that the proposed threshold-based scheduling policy is asymptotically optimal for a large number of users.

Résumé

Le stockage de contenu populaire dans des caches disponibles aux utilisateurs, est une technique émergente qui permet de réduire le trafic dans les réseaux sans fil. En particulier, le coded caching proposée par Maddah-Ali et Niesen a été considérée comme une approche prometteuse pour atteindre un temps de livraison constant au fur et à mesure que la dimension augmente, ce qui donne un système évolutif. Bien que conceptuellement attrayant, plusieurs limitations empêchent ses applications. Nous avons adressé les limitations de coded caching dans les réseaux sans fil et avons proposé des schémas de livraison qui exploitent le gain de coded caching.

Dans la première partie de la thèse, nous étudions la région de capacité pour un canal à effacement avec cache et retour d'information. En se basant sur l'algorithme de Wang et de Gatzianas, nous proposons un schéma qui exploite les occasions de multidiffusion créées par les sous-fichiers stockés dans la cache et les sous-fichiers reçus au cours de la transmission. Nous prouvons que notre schéma de livraison est optimal pour des cas particuliers. En utilisant la dualité observée entre le canal à effacement et et le canal à antennes multiples, ces résultats sont généralisés pour le canal à diffusion avec des antennes multiples et retour d'information.

Dans la deuxième partie, nous étudions la livraison de contenu sur un canal d'atténuation asymétrique, où la qualité du canal varie à travers les utilisateurs et le temps. En supposant que les demandes des utilisateurs arrivent de manière dynamique, nous concevons un schéma dynamique basé sur une structure de queues pour assurer un contrôle d'admission, un contrôle de combinaisons de fichiers, aussi bien que la planification de la transmission. À court terme, nous permettons les transmissions aux sous-ensembles d'utilisateurs avec un bon canal, évitant les utilisateurs avec mauvais canal, tandis que à long terme nous assurons la justice entre les utilisateurs. Nous prouvons que notre schéma de livraison dynamique maximise la fonction d'utilité par rapport à tous les schémas limités au cache décentralisé. L'analyse de performance se base sur la théorie de Lyapunov.

Dans la dernière partie, nous étudions la planification opportuniste pour un canal d'atténuation asymétrique, en assurant une métrique de justice entre des utilisateurs. Pour capturer ces deux mesures contradictoire, nous formulons notre fonction objective par une famille de fonctions concaves de alpha-fairness et nous utilisons le planificateur de descente de Gradient. Nous proposons une politique de planification simple à base de seuil avec une complexité linéaire et qui n'exige pas la connaissance instantanée de l'état du canal, mais seulement un bit de retour de chaque utilisateur. Nous prouvons que la politique de planification à base de seuil proposée est asymptotiquement optimale pour un grand nombre d'utilisateurs.

Contents

Notations

K	number of users
$[K]$	set of users $\{1, \ldots, K\}$
N	number of files in data base
F	average size of the files
Z_k	cache memory of user k
M_k	$\in [0, N]$ size of cache memory of user k in files
M	average memory size in files
W_i	i-th file
$\|.\|$	cardinal of sub-file in packets or bits
m_k	$= \frac{M_k}{NF} \in [0, 1]$ normalized cache size of user k
m	$= \frac{M}{NF} \in [0, 1]$ normalized cache size for equal cache capacities
$\oplus$	the bit-wise XOR operation
d_k	demand of user k
$\mathbf{d}$	demands vector $\mathbf{d} = (d_1, .., d_K)$
$\mathcal{L}_{\mathcal{J}}(W_i)$	the sub-file of W_i stored exclusively by the users in $\mathcal{J}$
$T(m, k)$	number of bits to be transmitted when using coded caching [1, 2]
ϵ_x	a constant which vanishes as $x \to \infty$, i.e. $\lim_{x\to\infty} \epsilon_x = 0$
p_k	power allocated to serve user k
F_i	size of W_i
$\mathcal{N}$	set of integers

$\binom{K}{i}$		the number of i-combinations from the set $\{1, \ldots, K\}$
L		number of bits per packet
$\mathbb{F}_q$		input alphabet of size $L = \log_2(q)$
X		channel input
Y_k		channel output of receiver k
E		erasure output
δ_k		erasure probability of user k
S_t		state of the channel in slot t
S^t		states of the channel up to slot t
$log(x)$		common logarithm of x (base 10)
$log_a(x)$		logarithm of x (base a)
$t_{\mathcal{J}}$		length of sub-phase $\mathcal{J}$
$t_{\mathcal{J}}^{\{k\}}$		length needed by user k for sub-phase $\mathcal{J}$
$V_{\mathcal{J}}$		packets intended to users in $\mathcal{J}$
$\mathcal{L}_{\mathcal{J}}(V_{\mathcal{K}})$		part of packets $V_{\mathcal{K}}$ received by users in $\mathcal{J}$ and erased at users in $[K] \setminus \mathcal{J}$
$N_{\mathcal{I}\to\mathcal{J}}^{\{k\}}$		number of packets useful for user k generated in sub-phase $\mathcal{I}$ and to be sent in sub-phase $\mathcal{J}$
$W_{\mathcal{J}}$		message intended to users in $\mathcal{J}$
$R_{\mathcal{J}}$		rate of $W_{\mathcal{J}}$
$R^j(K)$		sum rate of order-j messages
$t_j = t_j^1$		length of any sub-phase in phase j
t_j^i		length of any sub-phase j when starting from phase i
$N_{i\to j}$ $N_{i\to j}^1$	=	number of packets created in sub-phase $\mathcal{I}$ and to be sent in sub-phase $\mathcal{J}$ for any $\mathcal{I} \subset \mathcal{J}$ of cardinality $i < j$
$N_{i\to j}^{i'}$		$N_{i\to j}$ when starting from phase i' for $i \leq i' \leq j$

$Q_{\mathcal{I}}(t)$	codeword queue storing XOR-packets intended users in $\mathcal{I}$.
$S_k(t)$	user queue storing admitted files for user k.
$U_k(t)$	virtual queue for the admission control.
$\sigma_{\mathcal{J}}(t)$	decision variable of number of combined requests for users $\mathcal{J}$ in $[0, \sigma_{\max}]$.
$\mu_{\mathcal{I}}(t)$	decision variable for multicast transmission rate to users $\mathcal{I}$.
$a_k(t)$	decision variable of the number of admitted files for user k in $[0, \gamma_{\max}]$.
$\gamma_k(t)$	the arrival process to the virtual queue in $[0, \gamma_{\max}]$.
$\bar{r}_k$	time average delivery rate equal to $\limsup_{t\to\infty} \frac{D_k(t)}{t}$ in files/slot.
λ_k	mean of the arrival process.
$b_{\mathcal{J},\mathcal{I}}$	length of codeword intended to users $\mathcal{I}$ from applying coded caching for user in $\mathcal{J}$.
$\Gamma(\mathbf{h})$	the capacity region for a fixed channel state $\mathbf{h}$.
$\mathcal{H}$	the set of all possible channel states.
$\phi_{\mathbf{h}}$	the probability that the channel state at slot t is $\mathbf{h} \in \mathcal{H}$.
$D_k(t)$	number of successfully decoded files by user k up to t.
$A_k(t)$	number of accumulated requested files by user k up to slot t.

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Overview

In the past decades, cellular networks have been evolving significantly from the first generation (1G) which only provides voice services with data speed of up to 2.4 kbps, to the current fourth generation (4G) offering a wide range of data services (such as mobile Internet access, multimedia applications, TV streaming, videoconferencing and more) with much faster data speed of about 100 Mbps while a user moves at high speeds and 1 Gbps data rate in a fixed position [3]. In order to support the exponentially growing mobile data traffic, essentially driven by smart-phones, laptops, and tablets, the network was brought closer to the users by denser node deployment including macro base station (BS), micro-BS, pico-BS, femto-BS and relays. Such heterogeneous networks reduces the distance between BSs/relays and users, and thus increases the spectral efficiency, yielding the increase of network capacity [4].

However, the ever-increasing number of mobile devices with exploding data demand will eventually exhaust the bandwidth of existing backhaul connecting small BSs. Moreover, nearly 75% of the mobile data traffic is expected to be due to video by 2020 (e.g. content-based video streams) [5]. Such video traffic has an interesting feature characterized by its skew nature. Namely, a few very popular files are requested over and over. The skewness of the video traffic together with the ever-growing cheap on-board storage memory suggests that the network performance can be boosted by caching popular contents at (or close to) end users in wireless networks. Therefore, if one can proactively prefetch the contents at (or close to) end users during off-peak hours prior to the actual demands, the traffic during peak hours can be substantially off-loaded.

In the past years, this concept of caching has received a great deal of interest from researchers of both the industry and the academia. Caching has been traditionally done in the core of the networks. Recent trends focus on caching at the edges near the user [6]. A large amount of works have demonstrated considerable performance improvement by considering one of three network architectures:

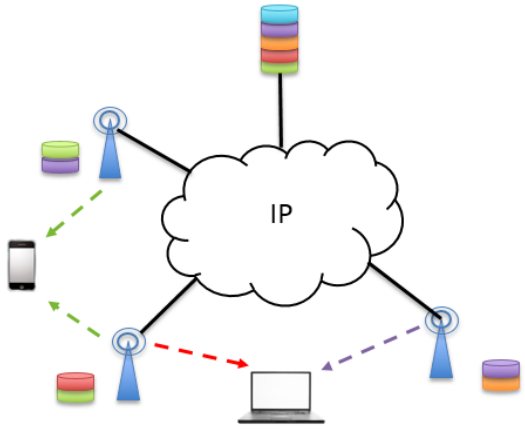


Figure 1.1: Femto-caching.

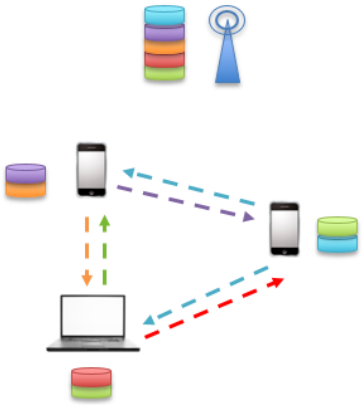


Figure 1.2: Caching with D2D communication.

- Femto-caching: As depicted in Fig. 1.1, users are connected to several small BSs (femto-cells), each equipped with a memory of finite size, and connected through a bottleneck network to a content-providing server. The performance of femto-caching was studied in [7–12]. [7] considers cellular network where helpers (small cell access points) with caching ability are installed in the cell. Each user has access to multiple helpers which can cooperate in order to minimize the access delay of the users. The work [12] considers heterogeneous network where the BSs, relays and cache-enabled users cooperate to satisfy the users demand.

- Caching with device-to-device (D2D) communication: Users request files from nearby users equipped with a memory of finite size as shown in Fig. 1.2. Such technologie greatly improves the network throughput and was studied by several works [13–21] under different setting. [19] proposed D2D caching scheme that achieves the information theoretic outer bound within a constant factor in some regimes. [16] has studied the joint optimization of cache placement and scheduling. An opportunistic cooperation strategy was proposed in [17] for D2D transmission to control the interference among D2D links. [20] has characterized the D2D caching capacity scaling law under more realistic physical channel.

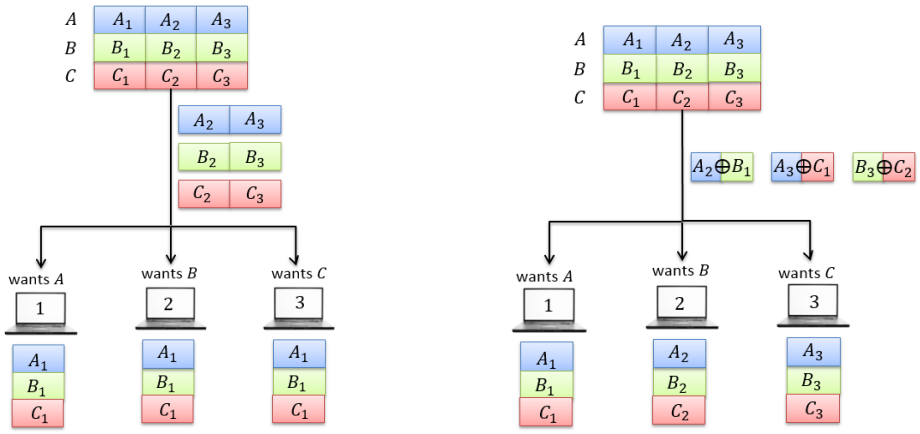
- Coded caching: users, equipped with a memory of finite size, request files from the content-providing server which sends coded messages to satisfy all users. Coded caching was introduced by Maddah Ali and Niesen [1,2], where sub-files are strategically placed in users' caches and linear combinations are delivered to simultaneously satisfy multiple users' requests. As we focus on coded caching, we provide the details in the following Section. Such a scheme significantly outperforms the conventional uncoded caching [22–26] where the caching gain is obtained by sending the remaining uncached requested files during peak hours.

1.2 Maddah-Ali and Niesen's Coded Caching

In our thesis, we mainly focus on coded caching. In [1, 2], Maddah-Ali and Niesen characterized the memory-rate trade-off when user demands are arbitrary and the bottleneck link is error-free. The network consists of a server with a database of N files, each F bits long, and K users. Each user k is equipped with a cache memory Z_k of size M files, with $M < N$. We often use the normalized size memory denoted by $m = \frac{M}{N}$. The server is connected to users through a shared link assumed to be error-free and perfect. The authors propose a two-phase *coded caching* scheme:

(i) placement phase: each file is divided in sub-files and each user pre-stores sub-files up to its memory constraint prior to the actual demands. The cost of this phase is negligible, provided that it could be performed in off-peak hours.

(ii) delivery phase: when the users make a particular request for certain files, the server transmits codewords such that each user, based on the received signal and the contents of its cache, is able to decode the requested file.



(a) Conventional caching

(b) Coded caching [1]

Figure 1.3: Example of $K = 3$ users with distinct user demands, data base of size $N = 3$ and cache memory size $M = 1$.

We illustrate coded caching and its gain through a toy example with $N = 3$ files, denoted by $\{A, B, C\}$, $K = 3$ users with a memory of $M = 1$ file, each. Assuming the worst case demand (i.e. different file requests) such that user 1, 2 and 3 requests A, B and C, respectively. We calculate the total number of transmissions measured in files, for uncoded caching and coded caching of [1]. Without user memory caches, the number of transmissions is three. In both cases, we first split each file into 3 sub-files of equal size, i.e. $A = (A_1, A_2, A_3)$, $B = (B_1, B_2, B_3)$, $C = (C_1, C_2, C_3)$.

- Uncoded caching: All users cache (A_1, B_1, C_1). To satisfy the users' demands, the server needs to send the remaining parts A_2, A_3 to user 1, B_2, B_3 to user 2, C_2, C_3

to user 3, as shown in Fig. 1.3. The number of transmissions is equal to $\frac{2}{3} \times 3 = 2$ files, decreased by one with respect to the case without caches.

- Coded caching: User k caches (A_k, B_k, C_k). User 1, 2, 3 needs respectively (A_2, A_3), (B_1, B_3), (C_1, C_2). By carefully choosing the placement, we can create three order-2 multicast symbols simultaneously useful to two users. As shown in Fig. 1.3, the server sends $A_2 \oplus B_1$ for users $\{1, 2\}$, $A_3 \oplus C_1$ for users $\{1, 3\}$, and $B_3 \oplus C_2$ for users $\{2, 3\}$, where $\oplus$ denotes the bit-wise XOR operation. The number of transmissions is equal to $\frac{1}{3} \times 3 = 1$ file.

This toy example shows that a careful design of sub-packetization and cache placement enables to perform *opportunistic multicasting* and thus decrease the total transmission time.

In the placement phase of [1], each file is split into multiple sub-files and cached in different users under the coordination of a central controller. Thus we refer to the scheme by *centralized coded caching*. An uncoordinated placement called *decentralized coded caching* has been proposed in [2], where each user independently caches sub-files uniformly at random without central coordination. We provide a general description of the two schemes in the following subsections.

Throughout the thesis, we use the following notational conventions. We let $[k] = \{1, \ldots, k\}$; ϵ_n denote a constant which vanishes as $n \to \infty$, i.e. $\lim_{n\to\infty} \epsilon_n = 0$; $|.|$ denote the length of sub-files in bits or in packets and $\binom{K}{i}$ denote the number of i-combinations from the set $\{1, \ldots, K\}$.

1.2.1 Decentralized Content Placement

Placement phase Under the memory constraint of MF bits, each user k independently caches a subset of mF bits of each file, chosen uniformly at random. By letting $\mathcal{L}_{\mathcal{J}}(W_i)$ denote the sub-file of W_i stored exclusively by the users in $\mathcal{J}$, the cache memory of user k after the decentralized placement is given by

$$Z_k = \{\mathcal{L}_{\mathcal{J}}(W_i) : \mathcal{J} \subseteq [K],\ \mathcal{J} \ni k,\ i = 1, \ldots, N\}. \tag{1.1}$$

The size of each sub-file is given by

$$|\mathcal{L}_{\mathcal{J}}(W_i)| = m^{|\mathcal{J}|}(1-m)^{K-|\mathcal{J}|}F + \epsilon_F \tag{1.2}$$

as $F \to \infty$.

To illustrate the placement strategy, let us consider an example of $K = 3$ users. After the placement phase, each file will be partitioned into 8 sub-files:

$$\begin{aligned} W_i = \{&\mathcal{L}_{\emptyset}(W_i), \mathcal{L}_1(W_i), \mathcal{L}_2(W_i), \mathcal{L}_3(W_i), \mathcal{L}_{12}(W_i), \mathcal{L}_{13}(W_i), \\ &\mathcal{L}_{23}(W_i), \mathcal{L}_{123}(W_i)\}. \end{aligned} \tag{1.3}$$

Delivery phase Once the requests of all users are revealed, the offline scheme proceeds to the delivery of the requested files. Assuming that user k requests file k, i.e. $d_k = k$, the server generates and conveys the following codeword simultaneously useful to the subset of users $\mathcal{J}$:

$$V_{\mathcal{J}} = \oplus_{k\in\mathcal{J}} \mathcal{L}_{\mathcal{J}\setminus\{k\}}\left(W_k\right). \tag{1.4}$$

The main idea here is to create a codeword useful to a subset of users by exploiting the receiver side information established during the placement phase. It is worth noticing that the *coded* delivery with XORs significantly reduces the number of transmissions. Obviously, the sub-files cached by the destination, e.g. $\mathcal{L}_1(W_1), \mathcal{L}_{12}(W_1), \mathcal{L}_{13}(W_1), \mathcal{L}_{123}(W_1)$ for user 1 requesting W_1, need not be transmitted in the delivery phase. Compared to uncoded delivery, where the server sends the remaining uncached sub-files with transmissions number equal to $|\mathcal{J}| \times |W_{k|\mathcal{J}\setminus\{k\}}|$, the coded delivery requires the transmission of $|W_{k|\mathcal{J}\setminus\{k\}}|$, yielding a reduction of a factor $|\mathcal{J}|$. In a practical case of $N > K$, it has been proved that decentralized coded caching achieves the total number of transmissions, measured in the number of files, given by [2]

$$T(m, K) = (1-m)\,\frac{1-(1-m)^K}{m}. \tag{1.5}$$

On the other hand, in uncoded delivery, the number of transmissions is given by $K(1-m)$ since it exploits only *local* caching gain at each user.

1.2.2 Centralized Content Placement

Placement phase We suppose that $M \in \{0, N/K, 2N/K, \ldots, N\}$ so that the parameter $b = \frac{MK}{N}$ is an integer. Each file is split into $\binom{K}{b}$ disjoint equal size sub-files. Each sub-file is cached at a subset of users $\mathcal{J}$, $\forall\, \mathcal{J} \subseteq [K]$ with cardinality $|\mathcal{J}| = b$. Namely, the size of any sub-file of file i is given by

$$|\mathcal{L}_{\mathcal{J}}(W_i)| = \frac{1}{\binom{K}{b}} F, \tag{1.6}$$

Delivery phase Once the requests of all users are revealed, the offline scheme proceeds to the delivery of the requested files. Assuming that user k requests file k, i.e. $d_k = k$, the server generates and conveys the following codeword simultaneously useful to the subset of users $\mathcal{J} \subseteq [K]$ for $|\mathcal{J}| = b + 1$:

$$V_{\mathcal{J}} = \oplus_{k\in\mathcal{J}} \mathcal{L}_{\mathcal{J}\setminus\{k\}}\left(W_k\right), \tag{1.7}$$

In a practical case of $N > K$, it has been proved that centralized coded caching achieves the total number of transmissions, measured in the number of files, given by [1]

$$T(m, K) = (1-m)\frac{1}{\frac{1}{K} + m}. \tag{1.8}$$

Throughout the thesis we use three facts concerning the mapping T that hold for both (1.5) and (1.8)

Property 1. *$T(m,k)$ converges to $T(m,\infty) \triangleq \frac{1-m}{m}$ when $k \to \infty$. The larger m is, the faster it converges.*

Property 2. *$T(m,k)$ is an increasing function of k and so $T(m,1) \leq T(m,k) \leq T(m,\infty)$.*

Property 3. *$\frac{k}{T(m,k)}$ is an increasing function of k.*

1.2.3 Assumptions

Compared to *uncoded caching*, where the server sends the remaining uncached parts of the requested files sequentially without any coding, the gain of coded caching is *phenomenal*. More specifically, it has been proved that the delivery time, given by (1.5) or (1.8), to satisfy K distinct requests converges to a constant in the regime of a large number of users K (see Property 1). Albeit conceptually and theoretically appealing, the promised gain of coded caching relies on some unrealistic assumptions (see. e.g. [6]). Namely, the most sensitive assumptions that the works [1] and [2] rely on include:

1. The files popularity profile is uniform so that each of N files is requested with probability $\frac{1}{N}$.
2. The file size is arbitrarily large.
3. The content placement and delivery are performed in an offline manner for a fixed set of user requests.
4. The shared bottleneck link is perfect and error-free.

1.2.4 Extensions

We summarize below recent progress to relax each of the above assumptions one by one.

Non-uniform files popularity
Existing works have relaxed the uniform demand assumption, and studied the performance of coded caching under heterogeneous popularity profile was studied in [27–30]. The proposed schemes in these works have modified the placement phase such that the most popular files are likely to be cached with higher probability than the less popular files. In [28, 30], the gap between the achievable bound and the lower bound was shown to be a constant.

Finite file size
In [31], it has been shown that the promised gain of coded caching holds only if the file size grows exponentially in the number of users when decentralized cache placement [2] and a class of clique cover delivery schemes are considered. In practical setting with finite

file size, the promised multicasting gain disappears. To overcome this limitation, several works [32–35] proposed different coded caching schemes. In [32] the shorter subfiles borrow bits from other subfiles to make their sizes equal in the linear combination. [34] divides the users into relatively small groups having the same cache contents. The work [33] proposed a polynomial-time algorithm based on greedy local graph-coloring to recover a part of the multicasting gain. In [35], the reduced coded caching gain in the finite file size case was improved by the use of multiple transmitting antennas. Furthermore, for sufficiently large number of antennas, the original coded caching gain under large file size assumption, can be recovered.

Online users requests
In practical scenarios, users request files asynchronously and randomly whenever they wish. Therefore, the time-varying user requests and corresponding delivery can be modeled as a random arrival and departure process, respectively. Recent works [36–38], addressed partly such issue of the online nature of the cache placement and/or delivery. On one hand, [36] studies the cache eviction strategies by assuming that the set of popular files evolves in the same time scale as the content delivery. On the other hand, [38] has focused on delay sensitive applications (such as video streaming) and studied the tradeoff between the performance gain of coded caching and the delivery delay.

Index coding
It is noted that the delivery phase of coded caching can be seen as index coding problem [39, 40]. For index coding, a server with database of files is connected to users. Each user has local access to a subset of files of the database and wishes to recover one file not locally available. The goal is to satisfy all user demands with minimum number of transmissions. Therefore, index coding (equivalently network coding, graph theory) was used for designing new delivery schemes [30,33,41] under different setting and assumptions (finite file length, random demands ..). The work in [41] considers the case of multiple requests per user. The proposed scheme is based on multiple groupcast index coding and showen to be approximately optimal within a constant. In [33], a graph coloring scheme was proposed for finite file length. The work [30] proposes a caching scheme based on chromatic number for the case of random demands.

Wireless broadcast channels
Further, recent works have attempted to relax the unrealistic assumption of a perfect shared link by replacing it with wireless channels. If wireless channels are used only to multicast a common signal, naturally the performance of coded caching (delivery phase) is limited by the user in the worst condition of fading channels as observed in [42]. This is due to the information theoretic limit, that is, the multicasting rate is determined by the worst user [43, Chapter 7.2]. If the underlying wireless channels enjoy some degrees of freedom to convey simultaneously both private messages and common messages, the delivery phase of coded caching can be further enhanced. These observations have inspired a number of recent works to overcome these drawbacks [44–56]. The works [44–48, 50, 51] have considered the use of multiple antennas, while [52–56] have proposed several interference management techniques. The works in [52, 53] consider the packet erasure broadcast channel (EBC) with two different class of users: weak users, in terms of channel quality,

equipped with equal cache memory and strong users with no cache, and provide an achievable scheme of joint cache-channel encoding based on Slepian–Wolf coding introduced in [57]. It was then generalized in [55, 58] for Gaussian broadcast channels (BC) with unequal cache sizes where memory assignment is allowed. It has shown that larger rates can be achieved by carefully assigning more cache memory to weaker users. For multiple input single output (MISO) broadcast channel, the interplay between coded caching and channel state information at the transmitter (CSIT) feedback was studied in [46, 51]. It has been shown that the multicasting gain of coded caching can reduce the CSIT feedback providing broadcast gains.

We investigate in the first part, the cache-aided erasure broadcast channel with state feedback. Note that the capacity region of the channel at hand was studied for the case without caches in [59, 60]. Thus we address the following questions in Chapter 2:

- What is the potential gain of coded caching for the EBC with feedback?
- What is the appropriate scheme to maximize such gain?

For coded caching in fading broadcast channel, recent works have studied opportunistic scheduling [47] by exploiting the fading peaks. However, when the users experience asymmetric fading statistics, opportunistic scheduling may lead to ignore some users in the system. In the literature of wireless scheduling without caches at receivers, this problem has been resolved by the use of fairness among user throughputs [61]. By allowing poorly located users to receive less throughput than others, precious air time is saved and the overall system performance is greatly increased. However, the classical coded caching scheme is designed to provide video files at equal data rates to all users. Furthermore, when user requests files asynchronously in a random process, the problem becomes more complicated. Thus we relax two assumptions (3 and 4) and address the following questions in Chapter 3:

- How shall we ensure fairness in coded caching scheme in fading broadcast channel, while adapting to the asynchronous and random arrival of the users requests?
- How shall we schedule a set of users to achieve our fairness objective while adapting to time-varying channel quality?

In Chapter 4, we address the fairness problem in coded caching over fading broadcast channel by relaxing assumption 4. We shed light into the following questions:

- How can we achieve a scalable content delivery over asymmetric fading channels?
- Is there a simple scheduling policy that requires only statistical channel knowledge?

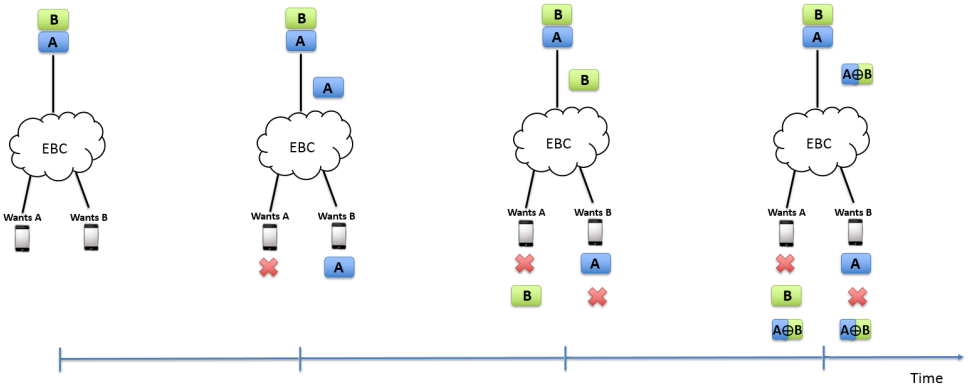


Figure 1.4: 2-user erasure broadcast channel during 3 time slots.

1.3 Contributions

Throughout the thesis, we have focused on the design of efficient content delivery schemes in wireless channels when the placement phase is restricted to centralized [1] and/or decentralized [2] placement strategy.

In Chapter 2, we relax assumtion 4 and consider erasure broadcast channels where receiver feedback is sent to the transmitter in the form of ACK/NACK messages. The capacity of the channel at hand has been characterized by Wang [60] and by Gatzianas [59]. The achievable scheme of [59, 60] relies on multicasting opportunities created by the overheard packets under instantaneous feedback assumption. Namely, when packets do not reach the destination because of the erasure event, and are received by other users, then instead of discarding them, these users can keep the received packets as side information that can be useful for future transmissions. We provide in Fig. 1.4 a 2-user example requesting different packets. After three transmissions time, each user is able to decode the requested packet where the overheard packets obtained in slots 1 and 2 are used in the third transmission to create a packet useful simultaneously to both users. Such multicasting opportunities are very similar to the one created by the placement phase in coded caching. We propose an achievable scheme exploiting receiver side information both from local caches and overhearing. We characterize the upper bound on the achievable region of the the cache-enabled EBC with state feedback for the decentralized placement. We provide an intuitive interpretation of the the algorithms proposed by Wang [60] and by Gatzianas [59] for the EBC with state feedback and then extend them to the case with receiver side information (acquired after cache placement phase). We prove that our proposed multi-phase delivery scheme achieves the optimal rate region for special cases of interest. These results are generalized to the centralized content placement [1] as well as the multi-antenna broadcast channel (BC) with state feedback.

In Chapters 3 and 4, we study the content delivery over asymmetric block-fading broadcast channel, where the channel quality varies across users and time. We address

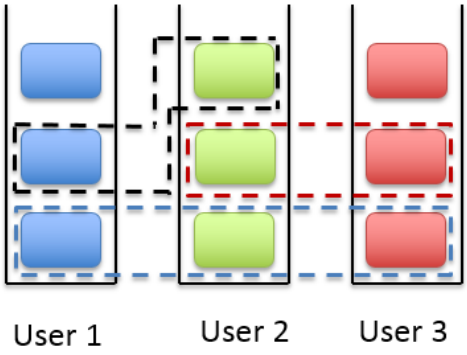


Figure 1.5: Files combinations decisions for 3 users example.

the fairness problem in the presence of caches by formulating our objective function by an alpha-fairness family of concave utility functions [62]. Unlike the classical coded caching which combines all the requested files, we exploit the fading peaks by deciding on the subset of users to linearly combine their requested files. We show in Fig. 1.5 an example on files combination decisions. We suppose that each user have several demands, stored in users queues. Depending on the channel and/or queues states we decide on the subset of files to combine. For this example the server decides on combing files requested by all users (dashed blue line), files requested by user 2 and 3 (dashed red line), and files requested by user 1 and 2 (dashed black line).

In Chapter 3, we propose an online scheme jointly dealing with the three main decisions (admission control, file combinations and scheduling) through queuing structure in order to ensure fairness between users with asymmetric channel statistics while dealing with the dynamic arriving requests and also exploiting opportunistically the time-varying fading channels. We prove in this chapter that our online delivery scheme maximizes the alpha-fair utility among all schemes restricted to decentralized placement [2].

In Chapter 4, we focus on the scheduling part and provide a rigorous analysis on the long-term average per-user rate in the regime of a large number of users. We study opportunistic scheduling, based on Gradient descent scheduling (GDS) [63], in order to achieve a scalable sum content delivery while ensuring some fairness among users. We propose a simple threshold-based scheduling policy and determine the threshold as a function of the fading statistics for each fairness parameter α. Such a threshold-based scheme exhibits two interesting features. On one hand, the complexity is linear in K. On the other hand, such a scheme does not require the exact channel state information but only a one-bit feedback from each user. Namely, each user indicates whether its measured signal to noise ratio (SNR) is above the threshold set before the communication. We prove that the proposed threshold-based scheduling policy is asymptotically optimal in Theorem 3. Namely, the utility achieved by our proposed policy converges to the optimal value as the number of users grows.

1.4 Preliminaries

1.4.1 Capacity region of Gaussian broadcast channels

Consider the additive white Gaussian noise BC. The channel output of user k at slot t is given by

$$\mathbf{y}_k = \sqrt{h_k}\mathbf{x} + \mathbf{z}_k, \tag{1.9}$$

where the channel input $\mathbf{x} \in \mathbb{C}^n$ is subject to the power constraint $\mathbb{E}[\|\mathbf{x}\|^2] \leq Pn$; $\mathbf{z}_k \sim \mathcal{N}_{\mathbb{C}}(0, \mathbf{I}_n)$ are additive white Gaussian noises with covariance matrix identity of size n, assumed independent of each other; $\{h_k \in \mathbb{C}\}$ are channel fading coefficients independently distributed across users.

Note that the channel model in (1.9) is equivalent to the Gaussian BC by normalizing (1.9) with $\sqrt{h_k}$

$$\mathbf{y}_k = \mathbf{x} + \boldsymbol{\nu}_k, \tag{1.10}$$

where $\boldsymbol{\nu}_k \sim \mathcal{N}_{\mathbb{C}}(0, N_k\mathbf{I}_n)$ with $N_k \triangleq \frac{1}{h_k}$. Since the capacity region of BC depends only on the marginal distribution, the capacity region of (1.9) and that of (1.10) coincide.

Theorem 1. *[43] The capacity region of a K-user degraded Gaussian broadcast channel with noise variances $N_1 \leq \cdots \leq N_K$ and total power constraint P is given by*

$$R_1 \leq \log \frac{N_1 + p_1}{N_1} \tag{1.11}$$

$$R_k \leq \log \frac{N_k + \sum_{j=1}^{k} p_j}{N_k + \sum_{j=1}^{k-1} p_j} \quad k = 2, \ldots, K \tag{1.12}$$

for non-negative variables $\{p_k\}$ such that $\sum_{k=1}^{K} p_k \leq P$.

1.4.2 Maximum weighted sum rate

We consider the following maximization problem

$$\max \sum_{k=1}^{K} \theta_k R_k \tag{1.13}$$

$$R_1 \leq \log \frac{N_1 + p_1}{N_1} \tag{1.14}$$

$$R_k \leq \log \frac{N_k + \sum_{j=1}^{k} p_j}{N_k + \sum_{j=1}^{k-1} p_j} \quad k = 2, \ldots, K \tag{1.15}$$

over the variables $\{p_k\}$ such that $\sum_{k=1}^{K} p_k \leq P$.

We provide Algorithm 1 to solve this power allocation problem as a special case of the parallel Gaussian broadcast channel studied in [64, Theorem 3.2]. Following [64], we define the rate utility function for user k given by

$$v_k(z) = \frac{\theta_k}{1/h_k + z} - \lambda, \tag{1.16}$$

where λ is a Lagrangian multiplier. The optimal solution corresponds to selecting the user with the maximum rate utility at each z and the resulting power allocation for user k is

$$p_k^* = \left\{ z : [\max_j v_j(z)]_+ = v_k(z) \right\} \tag{1.17}$$

with λ satisfying

$$P = \left[\max_k \frac{\theta_k}{\lambda} - \frac{1}{h_k} \right]_+ . \tag{1.18}$$

Algorithm 1 Weighted sum rate maximization

$\lambda = \max_{k=1}^{K} \frac{\theta_k}{N_k+P}$
Find the users: $k_0 = \arg\max_k \frac{\theta_k}{N_k}$ and $k_P = \arg\max_k \frac{\theta_k}{N_k+P} - \lambda$
Let $\mathcal{K} \leftarrow k_0$ and $k_1 = k_0$.
If $k_0 = k_P$, then serve only this user with the full power $p_{k_0} = P$.
Else
For $j = \{2, \ldots, K\}$
Find the smallest intersection point with user k_{j-1}

$$z_j \triangleq \min_{i \in [K] \setminus \mathcal{K}} \frac{\theta_i N_{k_{j-1}} - \theta_{k_{j-1}} N_i}{\theta_{k_{j-1}} - \theta_i}$$

and its associated user k_j.
If $z_j < P$,
$p_{k_{j-1}} = z_j - z_{j-1}$
$\mathcal{K} \leftarrow \{\mathcal{K}, k_j\}$.
Else $p_{k_{j-1}} = P - z_{j-1}$, Stop.

1.4.3 Gradient Descent Scheduling

We consider a time slotted system where at each slot t we decide on the service rate $r_k(t)$ of user k. We suppose that each user always has data to be served, and we are interested

in optimizing the average service rates $\bar{\mathbf{r}} = (\bar{r}_1, .., \bar{r}_1)$ with $\bar{r}_k = \mathbb{E}(r_k(t))$. We aim to find a scheduling algorithm that provides $\bar{\mathbf{r}}^*$ solving the following maximization problem

$$\max G(\bar{\mathbf{r}}) \tag{1.19}$$
$$\bar{\mathbf{r}} \in \Gamma, \tag{1.20}$$

where $G(.)$ is a concave utility function, and set Γ is the system rate region.The gradient descent algorithm [63] chooses a (possibly nonunique) decision

$$\max \Delta(G(\mathbf{u}(t))).\mathbf{r}(t) \tag{1.21}$$

maximizing the scalar product with the gradient of $G(\mathbf{u}(t))$ where $u_k(t+1) \triangleq (1-\epsilon)u_k(t) + \epsilon r_k(t)$ denotes the empirical data rate up to time t for a given constant $\epsilon > 0$.

Theorem 2. *(Asymptotic Optimality of the Gradient Algorithm [63]). Let $\bar{\mathbf{r}}^\epsilon$ denote the vector of expected average service rates in a system with fixed parameter $\epsilon > 0$. Then, as $\epsilon \to 0$, $\bar{\mathbf{r}}^\epsilon \to \bar{\mathbf{r}}^*$.*

1.4.4 Queueing structure and Lyapunov optimization

Queue stability Let $Q(t)$ represent a discrete time process over integer time slots $t \in \mathbb{N}$ evolving as the following recursive equation as shown in Fig. 1.6

$$Q(t+1) = [Q(t) - b(t)]_+ + a(t), \tag{1.22}$$

where $\{a(t)\}_{t=0}^{\infty}$ and $\{b(t)\}_{t=0}^{\infty}$ are stochastic processes. The value of $a(t)$, $b(t)$ represents the amount of work that arrives, the amount of work the server of the queue Q can process, respectively on slot t and assumed to be non-negative. The value of $Q(t)$ called the backlog at slot t, assumed to have non-negative initial state $Q(0) \geq 0$.

Definition 1 (Stability [65])**.** *A queue $Q(t)$ is said to be (strongly) stable if*

$$\limsup_{T\to\infty} \frac{1}{T} \sum_{t=0}^{T-1} \mathbb{E}\left[Q(t)\right] < \infty.$$

A queueing system is said to be stable if all its queues are stable. Moreover, the stability region of a system is the set of all vectors of arrivals such that the system is stable.

Lyapunov Function Consider a network with L queues, and let $\mathbf{U}(t) = (U_1(t), ..., U_L(t))$ represents the vector of backlog in each queue at time slot t. We define the following quadratic Lyapunov function

$$L(\mathbf{U}) = \sum_{i=1}^{L} U_i^2(t). \tag{1.23}$$

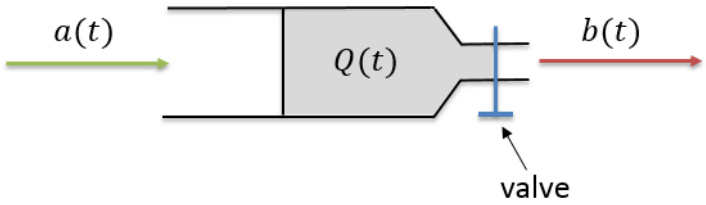


Figure 1.6: Illustration of the evolution of queue $Q(t)$.

which measures the total queue backlogs in the network. We define the Lyapunov drift given by

$$\Delta L(\mathbf{U}(t)) = \mathbb{E}\left\{L(\mathbf{U}(t+1) - L(\mathbf{U}(t) \,|\, \mathbf{U}(t)\right\} \tag{1.24}$$

which represent the expected change in the Lyapunov function (queue backlogs) from one slot to the next one.

Lemma 3 (Lyapunov stability [66]). *If there exist constants $B > 0$, $\epsilon > 0$, such that for all time slots t we have:*

$$\Delta L(\mathbf{U}(t)) \leq B - \epsilon \sum_{i=1}^{L} U_i(t), \tag{1.25}$$

then the network is strongly stable, and furthermore

$$\limsup_{T\to\infty} \frac{1}{T} \sum_{t=0}^{T-1} \sum_{i=1}^{L} \mathbb{E}\left\{U_i(t)\right\} \leq \frac{B}{\epsilon}. \tag{1.26}$$

Note that whenever $\sum_{i=1}^{L} U_i(t) \geq \frac{B+\delta}{\epsilon}$, the Lyapunov drift satisfies $\Delta L(\mathbf{U}(t)) \leq -\delta$. Namely, (1.25) ensures that the Lyapunov drift is negative whenever the sum of queue backlogs is relatively large, which ensures network stability.

Consider the following maximization problem

$$\overline{\boldsymbol{r}}^* = \arg\max_{\overline{\boldsymbol{r}}\in\Lambda} \sum_{k=1}^{K} g_k(\overline{r}_k) \tag{1.27}$$

for some capacity region Λ; utility function $g_k(.)$; $\overline{r}_k = \mathbb{E}\left\{r_k(t)\right\}$ and $\sum_{k=1}^{K} g_k(r_k(t)) \leq G_{\max}$. **The drift-plus-penalty algorithm**, corresponding to the maximization problem in (1.27), is such that it minimizes the following [66]

$$\Delta L(\mathbf{U}(t)) - V\mathbb{E}\left\{\sum_{k=1}^{K} g_k(\overline{r}_k) \,|\, \mathbf{U}(t)\right\} \tag{1.28}$$

Lemma 4 (Lyapunov optimization [61]). *If there are positive constants V, ϵ, B such that for all time slots t and all unfinished work $\mathbf{U}(t)$, the Lyapunov drift satisfies*

$$\Delta L(\mathbf{U}(t)) - V\mathbb{E}\left\{\sum_{k=1}^{K} g_k(\overline{r}_k) \,|\, \mathbf{U}(t)\right\} \leq B - \epsilon \sum_{i=1}^{L} U_i(t) - V \sum_{k=1}^{K} g_k(\overline{r}_k^*). \tag{1.29}$$

then the time average utility and congestion satisfy

$$\limsup_{T\to\infty} \frac{1}{T} \sum_{t=0}^{T-1} \sum_{i=1}^{L} \mathbb{E}\left\{U_i(t)\right\} \leq \frac{B + VG_{\max}}{\epsilon}, \tag{1.30}$$

$$\liminf_{T\to\infty} \frac{1}{T} \sum_{t=1}^{T} \sum_{k=1}^{K} g_k(r_k(t)) \geq \sum_{k=1}^{K} g_k(\overline{r}_k^*) - \frac{B}{V} \tag{1.31}$$

Notice that V is a parameter that controls the utility-delay tradeoff. Indeed, by tuning the constant V, the resulting utility can be arbitrarily close to the optimal one, where there is a tradeoff between the guaranteed optimality gap $\mathcal{O}(1/V)$ and the upper bound on the total queue backlogs $\mathcal{O}(V)$.

1.5 Publications

Journal

1. A. Ghorbel, M. Kobayashi and S. Yang, "Content delivery in erasure broadcast channels with cache and feedback", IEEE Transactions on Information Theory, vol. 62, no. 11, pp. 6407–6422, 2016.

Conferences

2. R. Combes, A. Ghorbel, M. Kobayashi, and S. Yang "Utility Optimal Scheduling for Coded Caching in General Topologies", submitted to the IEEE International Symposium on Information Theory (ISIT), 2018.

3. A. Ghorbel, K.-H. Ngo, R. Combes, M. Kobayashi, and S. Yang, "Opportunistic content delivery in fading broadcast channels", IEEE Global Communications Conference (GLOBECOM), Singapore, November 2017.

4. A. Destounis, M. Kobayashi, G. Paschos, and A. Ghorbel, "Alpha fair coded caching", 2017 15th IEEE International Symposium on Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks (WiOpt), Paris, France, May 2017.

5. A. Ghorbel, M. Kobayashi and S. Yang, "Content delivery in erasure broadcast channels with cache and feedback", IEEE International Symposium on Information Theory (ISIT), Barcelona, Spain, July 2016.

6. A. Ghorbel, M. Kobayashi and S. Yang, "Cache-enabled broadcast packet erasure channels with state feedback", 2015 53rd Annual Allerton Conference on Communication, Control, and Computing (Allerton), pp. 1446–1453, Chicago, USA, September 2015.

Workshop

7. A. Ghorbel, M. Kobayashi and S. Yang "Cache-enabled erasure broadcast channels with feedback: asymmetric user memory case", ACM Content Caching and Delivery in Wireless Networks (CCDWN) Workshop, Heidelberg, Germany, December 2016.

Patent

7. A. Destounis, G. Paschos, A. Ghorbel and M. Kobayashi "A method and an apparatus for efficient coded caching for wireless channels with fading", filed in November 2017.

Submitted papers

8. A. Destounis, A. Ghorbel, G. Paschos and M. Kobayashi, "Adaptive Coded Caching for Fair Delivery over Fading Channels", submitted to the IEEE Transactions on Information Theory, 2018, available on arXiv preprint arXiv:1802.02895, 2018.

9. R. Combes, A. Ghorbel, M. Kobayashi, and S. Yang "Utility Optimal Scheduling for Coded Caching in General Topologies", submitted to the IEEE Journal on Selected Areas in Communications (JSAC), 2017, available on arXiv preprint arXiv:1801.02594, 2018.

Chapter 2

Erasure Broadcast Channels with Feedback

We study the achievable rate region of the erasure broadcast channel (EBC) with cache and state feedback. Based on Wang and Gatzianas scheme, we propose an achievable scheme that exploits multicasting opportunities created by receiver side information both from local cache and overhearing. We prove that our proposed delivery scheme achieves the optimal rate region for special cases of interest. Using the interesting duality between the EBC and the multi-antenna broadcast channel, these results are generalized to the multi-antenna broadcast channel with state feedback.

2.1 Introduction

In this chapter, we model the bottleneck link between the server with N files and K users equipped with a cache of a finite memory as an erasure broadcast channel (EBC). The simple EBC captures the essential features of wireless channels such as random failure or disconnection of any server-user link that a packet transmission may experience during high-traffic hours, i.e. during the delivery phase.

We consider a memoryless EBC in which erasure is independent across users with probabilities $\{\delta_k\}$ and assume that each user k has a memory cache of M_k files. Moreover, the server is supposed to acquire the channel states causally via feedback sent by the users. Under this setting, we study the achievable rate region of the EBC with cache and state feedback. Our contribution is four-fold:

1. We characterize the upper bound on the achievable region of the the cache-enabled EBC with state feedback for the decentralized placement. The converse proof builds on a generalized form of the entropy inequalities (Lemma 6) as well as the reduced entropy of messages in the presence of receiver side information (Lemma 7). These lemmas can be easily adapted to other scenarios such as centralized placement as well as the multi-antenna broadcast channel.

2. We provide an intuitive interpretation of the the algorithms proposed by Wang [60] and by Gatzianas [59] for the EBC with state feedback and then extend them to the case with receiver side information (acquired after cache placement phase). We prove that our proposed multi-phase delivery scheme achieves the optimal rate region for special cases of interest.

3. These results are generalized to the centralized content placement [1] as well as the multi-antenna broadcast channel (BC) with state feedback. Here, a duality between the EBC and the multi-antenna BC in terms of the order-j multicast rate has been exploited.

4. Numerical examples are provided to quantify the benefit of state feedback, the relative merit of the centralized caching to the decentralized counterpart, as well as the gain due to the optimization of memory sizes, as a function of other system parameters.

Throughout the Chapter, we use the following notations. The superscript notation X^n represents a sequence $(X_1, \ldots, X_n)$ of variables. The entropy of X is denoted by $H(X)$. We say $f(x) = \mathcal{O}_x\left(g(x)\right)$ if $\lim_{x\to\infty} \frac{f(x)}{g(x)} < \infty$.

2.2 System Model and Definitions

We consider the cache model of Maddah Ali and Niesen (Section 1.2) and relax some assumptions (perfect shared link, equal file sizes and equal cache capacities). As depicted

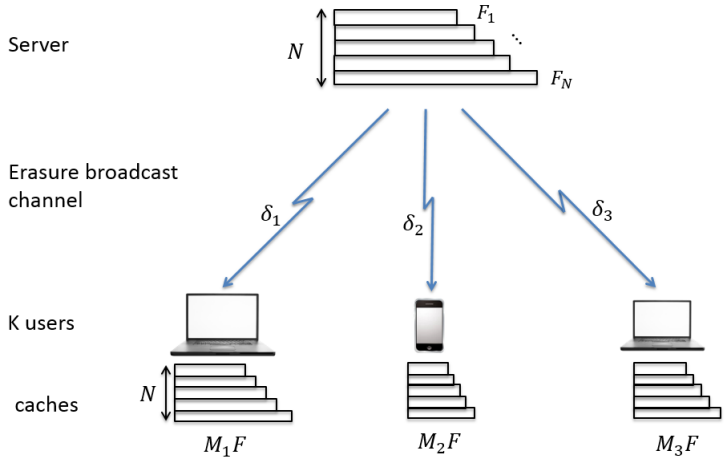


Figure 2.1: Cached-enabled EBC with $K = 3$.

in Fig. 2.1, the server, of data base with distinct file sizes, is connected to users with different cache capacities through an EBC. The data base files are denoted by $W_1, \ldots, W_N$ where the i-th file W_i consists of F_i packets, each of size $L \triangleq \log_2(q)$ bits. Each user k has a cache memory Z_k of M_kF packets for $M_k \in [0, N]$, where $F \triangleq \frac{1}{N}\sum_{i=1}^{N} F_i$ is the average size of the files. We often use the normalized cache size denoted by $m_k = \frac{M_k}{N}$.

Placement phase We mainly focus on the decentralized content placement recalled in Sub-Section 1.2.1. Under the memory constraint of M_kF packets, each user k independently caches a subset of m_kF_i packets of file i, chosen uniformly at random for $i = 1, \ldots, N$. We recall that $\mathcal{L}_{\mathcal{J}}(W_i)$ denotes the sub-file of W_i stored exclusively by the users in $\mathcal{J}$. The size of each sub-file is given by

$$|\mathcal{L}_{\mathcal{J}}(W_i)| = \prod_{j\in\mathcal{J}} m_j \prod_{j\in[K]\setminus\mathcal{J}} (1-m_j)F_i + \epsilon_{F_i} \tag{2.1}$$

as $F_i \to \infty$. It can be easily verified that the memory constraint of each user is fulfilled, namely,

$$\begin{aligned} |Z_k| &= \sum_{i=1}^{N} \sum_{\mathcal{J}:k\in\mathcal{J}} |\mathcal{L}_{\mathcal{J}}(W_i)| \\ &= \sum_{i=1}^{N} (F_i m_k + \epsilon_{F_i}) \\ &= M_kF + \sum_{i=1}^{N} \epsilon_{F_i} \end{aligned} \tag{2.2}$$

as $F_i \to \infty$ for all i. Throughout this Chapter, we assume that $F \to \infty$ and meanwhile $\frac{F_i}{F}$ converges to some constant $\tilde{F}_i > 0$. Thus, we identify all ϵ_{F_i} with a single ϵ_F. We provide a more formal definition below:

- N message files $W_1, \ldots, W_N$ independently and uniformly distributed over $\mathcal{W}_1 \times \cdots \times \mathcal{W}_N$ with $\mathcal{W}_i = \mathbb{F}_q^{F_i}$ for all i.

- K caching functions defined by $\phi_k : \mathbb{F}_q^{\sum_{i=1}^N F_i} \to \mathbb{F}_q^{FM_k}$ that map the files $W_1, \ldots, W_N$ into user k's cache contents

$$Z_k = \phi_k(W_1, \ldots, W_N), \quad k \in [K]. \tag{2.3}$$

Delivery phase Under such a setting, consider a discrete time communication system where a packet is sent in each slot over the K-user EBC. The channel input $X_k \in \mathbb{F}_q$ belongs to the input alphabet of size L bits. The erasure is assumed to be memoryless and independently distributed across users so that in a given slot we have

$$\Pr(Y_1, Y_2, \ldots, Y_K|X) = \prod_{k=1}^{K} \Pr(Y_k|X) \tag{2.4}$$

$$\Pr(Y_k|X) = \begin{cases} 1-\delta_k, & Y_k = X, \\ \delta_k, & Y_k = E \end{cases} \tag{2.5}$$

where Y_k denotes the channel output of receiver k, E stands for an erased output, δ_k denotes the erasure probability of user k. We let $S_t \in \mathcal{S} = 2^{\{1,\ldots,K\}}$ denote the state of the channel in slot t and indicate the set of users who received correctly the packet. We assume that all the receivers know instantaneously S_t, and that through feedback the transmitter only knows the past states S^{t-1} during slot t.

Once each user k makes a request d_k, the server sends the codewords so that each user can decode its requested file as a function of its cache contents and received signals during the delivery phase. A $(M_1, \ldots, M_K, F_{d_1}, \ldots, F_{d_K}, n)$ caching delivery scheme consists of the following components.

- A sequence of encoding functions defined by $f_t : \mathbb{F}_q^{\sum_{i=1}^N F_i} \times \mathcal{S}^{t-1} \to \mathbb{F}_q$ that map the requested files and the state feedback up to slot $t-1$ into a transmit symbol at slot t. Namely, the transmit symbol in slot t is given by

$$X_t = f_t(W_{d_1}, \ldots, W_{d_K}, S^{t-1}), \quad t = 1, \ldots, n \tag{2.6}$$

  where W_{d_k} denotes the message file requested by user k for $d_k \in \{1, \ldots, N\}$.

- K decoding functions defined by $\psi_k : \mathbb{F}_q^n \times \mathbb{F}_q^{FM_k} \times \mathcal{S}^n \to \mathbb{F}_q^{F_{d_k}}$, $k \in [K]$, that decode the file $\hat{W}_{d_k} = \psi_k(Y_k^n, Z_k, S^n)$ as a function of the received signals Y_k^n, the cache content Z_k, as well as the state information S^n.

A rate tuple $(R_1, \ldots, R_K)$ is said to be achievable if, for every $\epsilon > 0$, there exists a $(M_1, \ldots, M_K, F_{d_1}, \ldots, F_{d_K}, n)$ caching strategy that satisfies the reliability condition

$$\max_{(d_1,\ldots,d_K)\in\{1,\ldots,N\}^K} \max_k \Pr(\psi_k(Y_k^n, Z_k, S^n) \neq W_{d_k}) < \epsilon$$

as well as the rate condition

$$R_k < \frac{F_{d_k}}{n} \quad \forall k \in [K]. \tag{2.7}$$

Throughout this Chapter, we express for brevity the entropy and the rate in terms of packets in order to avoid the constant factor L.

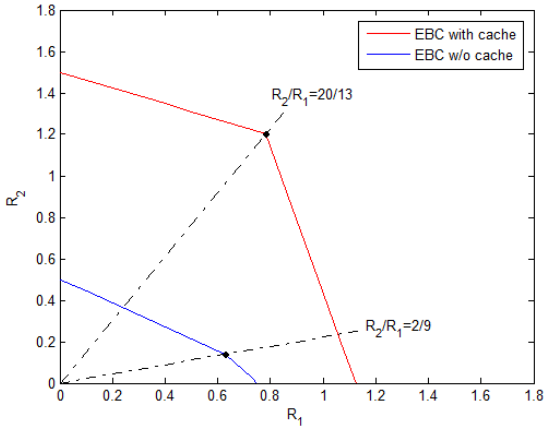


Figure 2.2: A two-user rate region with $(m_1, m_2) = (\frac{1}{3}, \frac{2}{3})$, $(\delta_1, \delta_2) = (\frac{1}{4}, \frac{1}{2})$.

2.3 Main Results

In order to present the main results, we specify two special cases.

Definition 2. *The cache-enabled EBC (or the network) is* symmetric *if the erasure probabilities as well as the memory sizes are the same for all users, i.e.* $\delta_1 = \cdots = \delta_K = \delta$, $m_1 = \cdots = m_K = m$.

Definition 3. *The rate vector is said to be* one-sided fair *in the cache-enabled EBC if* $\delta_k \geq \delta_j$ *implies*

$$\frac{R_k}{R_j} \geq \max\left\{\frac{\delta_j}{\delta_k}, \frac{(1-m_j)/m_j}{(1-m_k)/m_k}\right\}, \quad \forall k \neq j. \tag{2.8}$$

For the special case without cache memory ($m_1 = \cdots = m_K = 0$), the above definition reduces to the one-sided fairness originally defined in [60, Definition 5]: $\delta_k \geq \delta_j$ implies $\delta_k R_k \geq \delta_j R_j$ for $k \neq j$. For the particular scenario of symmetric rate ($R_k = R \ \ \forall k$), the rate vector $(R, \ldots, R)$ belongs to the one sided-rate region if and only if $\max\left\{\frac{\delta_j}{\delta_k}, \frac{m_k}{m_j}\right\} \leq 1$, $\forall j \neq k$, which means that the better user in terms of channel quality should have the larger cache memory.

Focusing on the case of most interest with $N \geq K$ and K distinct demands, we present the following main results of this Chapter.

Theorem 5. *For* $K \leq 3$*, or for the symmetric network with* $K > 3$*, or for the one-sided fair rate vector with* $K > 3$*, the achievable rate region of the cached-enabled EBC with the state feedback under the decentralized content placement is given by*

$$\sum_{k=1}^{K} \frac{\prod_{j=1}^{k}(1-m_{\pi_j})}{1-\prod_{j=1}^{k}\delta_{\pi_j}} R_{\pi_k} \leq 1 \tag{2.9}$$

for any permutation π *of* $\{1, \ldots, K\}$.

The above region has a polyhedron structure determined by $K!$ inequalities in general. For the symmetric network, the above region simplifies to the following

$$\sum_{k=1}^{K} \frac{(1-m)^k}{1-\delta^k} R_{\pi_k} \leq 1, \quad \forall \pi. \tag{2.10}$$

For the case without cache memory, i.e. $m_k = 0$ for all k, Theorem 5 boils down to the capacity region of the EBC with state feedback [59, 60] given by

$$\sum_{k=1}^{K} \frac{1}{1-\prod_{j=1}^{k} \delta_{\pi_j}} R_{\pi_k} \leq 1, \quad \forall \pi \tag{2.11}$$

which is achievable for $K \leq 3$ or the symmetric network or the one-sided fair rate vector where $\delta_k \geq \delta_j$ implies $\delta_k R_k \geq \delta_j R_j$ for any $k \neq j$. Comparing (2.9) and (2.11), we immediately see that the presence of cache memories decreases the weights in the weighted rate sum and thus enlarges the rate region. In order to gain some further insight, Fig. 2.2 illustrates a toy example of two users with $(m_1, m_2) = (\frac{1}{3}, \frac{2}{3})$ and $(\delta_1, \delta_2) = (\frac{1}{4}, \frac{1}{2})$. According to Theorem 5, the rate region is given by

$$\begin{aligned} \frac{8}{9} R_1 + \frac{16}{63} R_2 &\leq 1 \\ \frac{16}{63} R_1 + \frac{2}{3} R_2 &\leq 1 \end{aligned} \tag{2.12}$$

which is characterized by three vertices $(\frac{9}{8}, 0)$ $(0.78, 1.20)$, and $(0, \frac{63}{16})$. The vertex $(0.78, 1.20)$, achieving the sum rate of 1.98, corresponds to the case when the requested files satisfy the ratio $F_{d_2}/F_{d_1} = 20/13$. On the other hand, the region of the EBC without cache is given by

$$\begin{aligned} \frac{4}{3} R_1 + \frac{8}{7} R_2 &\leq 1 \\ \frac{8}{7} R_1 + 2 R_2 &\leq 1 \end{aligned} \tag{2.13}$$

which is characterized by three vertices $(\frac{3}{4}, 0)$, $(0.63, 0.14)$, $(0, \frac{1}{2})$. The sum capacity of 0.77 is achievable for the ratio $R_2/R_1 = 2/9$. The gain due to the cache is highlighted even in this toy example.

Theorem 5 yields the following corollary.

Corollary 1. *For $K \leq 3$, or for the symmetric network with $K > 3$, or for the one-sided fair rate vector with $K > 3$, the transmission length to deliver requested files to users in the cached-enabled EBC under the decentralized content placement is given by*

$$T_{\text{tot}} = \max_{\pi} \left\{ \sum_{k=1}^{K} \frac{\prod_{j=1}^{k} (1-m_{\pi_j})}{1-\prod_{j=1}^{k} \delta_{\pi_j}} F_{d_{\pi_k}} \right\} + \Theta(1), \tag{2.14}$$

as $F \to \infty$.

For the symmetric network with files of equal size ($F_i = F, \forall i$), the transmission length simplifies to

$$T_{\text{tot}} = \sum_{k=1}^{K} \frac{(1-m)^k}{1-\delta^k} F + \Theta(1), \tag{2.15}$$

as $F \to \infty$. The corollary 1 covers some existing results in the literature. For the case with files of equal size and without erasure, the transmission length in Corollary 1 normalized by F coincides with the "rate-memory tradeoff" [1] under the decentralized content placement for asymmetric memory sizes [67] given by

$$\frac{T_{\text{tot}}}{F} = \sum_{k=1}^{K} \left[\prod_{j=1}^{k} (1-m_j) \right], \tag{2.16}$$

where the maximum over all permutations is chosen to be identity by assuming $m_1 \geq \cdots \geq m_K$. If additionally we restrict ourselves to the case with caches of equal size, we recover the rate-memory tradeoff given in [2]

$$\frac{T_{\text{tot}}}{F} = \frac{N}{M} \left(1 - \frac{M}{N}\right) \left\{ 1 - \left(1 - \frac{M}{N}\right)^K \right\}. \tag{2.17}$$

In fact, the above expression readily follows by applying the geometric series to the RHS of (2.16).

2.4 Converse

In this section, we prove the converse of Theorem 5. First we provide two useful lemmas. The first one is a generalized form of the entropy inequality, while the second one is a simple relation of the message entropy in the presence of receiver side information. The former has been stated and proved in [68].

Lemma 6. *[68, Lemma 5] For the erasure broadcast channel, if U is such that $X_t \leftrightarrow UY_{\mathcal{I}}^{t-1}S^{t-1} \leftrightarrow (S_{t+1}, \ldots, S_n)$, $\forall \mathcal{I}$,*

$$\frac{1}{1-\prod_{j\in\mathcal{I}}\delta_j} H(Y_{\mathcal{I}}^n \,|\, U, S^n) \leq \frac{1}{1-\prod_{j\in\mathcal{J}}\delta_j} H(Y_{\mathcal{J}}^n \,|\, U, S^n) \tag{2.18}$$

for any sets $\mathcal{I}, \mathcal{J}$ such that $\mathcal{J} \subseteq \mathcal{I} \subseteq \{1, \ldots, K\}$.

Proof. We restate the proof in Appendix A.1. □

[1] In [2] and all follow-up works, the "rate" is defined as the number of files to deliver over the shared link, which corresponds to our T_{tot} here.

Lemma 7. *Under the decentralized content placement [2], the following inequality holds for any i and $\mathcal{I} \subseteq [K]$*

$$H(W_i \,|\, \{Z_k\}_{k\in\mathcal{I}}) \geq \prod_{k\in\mathcal{I}} (1-m_k)\, H(W_i).$$

Proof. Under the decentralized content placement, we have

$$\begin{aligned}
H(W_i \,|\, \{Z_k\}_{k\in\mathcal{I}}) &= H(W_i \,|\, \{\mathcal{L}_{\mathcal{J}}(W_l)\}_{\mathcal{J}\cap\mathcal{I}\neq\emptyset,\, l=1,\ldots,N}) && (2.19)\\
&= H(W_i \,|\, \{\mathcal{L}_{\mathcal{J}}(W_i)\}_{\mathcal{J}\cap\mathcal{I}\neq\emptyset}) && (2.20)\\
&= H(\{\mathcal{L}_{\mathcal{J}}(W_i)\}_{\mathcal{J}\cap\mathcal{I}=\emptyset}) && (2.21)\\
&= \sum_{\mathcal{J}:\mathcal{J}\subseteq[K]\setminus\mathcal{I}} H(\mathcal{L}_{\mathcal{J}}(W_i)) && (2.22)\\
&\geq \sum_{\mathcal{J}:\mathcal{J}\subseteq[K]\setminus\mathcal{I}} H(\mathcal{L}_{\mathcal{J}}(W_i) \,|\, \mathcal{L}_{\mathcal{J}}) && (2.23)
\end{aligned}$$

where the first equality follows from (1.1); the second equality follows due to the independence between messages $W_1, \cdots, W_N$; the third equality follows by identifying the unknown parts of W_i given the cache memories of $\mathcal{I}$ and using the independence of all sub-files; (2.22) is again from the independence of the sub-files. Note that $\mathcal{L}_{\mathcal{J}}$ is a random variable indicating which subset of packets of file W_i are shared by the users in $\mathcal{J}$. The size of the random subset $|\mathcal{L}_{\mathcal{J}}|$ follows thus the binomial distribution $B\Big(H(W_i),\, \prod_{j\in\mathcal{J}} m_j \prod_{k\in[K]\setminus\mathcal{J}}(1-m_k)\Big)$. It is readily shown that $H(\mathcal{L}_{\mathcal{J}}(W_i) \,|\, \mathcal{L}_{\mathcal{J}}) = \mathbb{E}\{|\mathcal{L}_{\mathcal{J}}|\}$. This implies that

$$\begin{aligned}
&H(W_i \,|\, \{Z_k\}_{k\in\mathcal{I}})\\
&\quad\geq \sum_{\mathcal{J}:\mathcal{J}\subseteq[K]\setminus\mathcal{I}} \prod_{j\in\mathcal{J}} m_j \prod_{k\in[K]\setminus\mathcal{J}} (1-m_k) H(W_i) && (2.24)\\
&\quad= \prod_{k\in\mathcal{I}}(1-m_k) \sum_{\mathcal{J}:\mathcal{J}\subseteq[K]\setminus\mathcal{I}} \prod_{j\in\mathcal{J}} m_j \prod_{k\in[K]\setminus\mathcal{I}\setminus\mathcal{J}} (1-m_k) H(W_i) && (2.25)\\
&\quad= \prod_{k\in\mathcal{I}} (1-m_k)\, H(W_i) && (2.26)
\end{aligned}$$

where the last inequality is obtained from the basic property that we have

$$\sum_{\mathcal{J}\subseteq\mathcal{M}} \prod_{j\in\mathcal{J}} m_j \prod_{k\in\mathcal{M}\setminus\mathcal{J}} (1-m_k) = 1$$

for any subset $\mathcal{M} \subseteq [K]$, in particular for $\mathcal{M} = [K] \setminus \mathcal{I}$. □

We apply genie-aided bounds to create a degraded erasure broadcast channel by providing the messages, the channel outputs, as well as the receiver side information (contents

of cache memories) to the enhanced receivers. Without loss of generality, we focus on the case without permutation and the demand $(d_1, \ldots, d_K) = (1, \ldots, K)$.

$$
\begin{aligned}
n \prod_{j=1}^{k} (1 - m_j) R_k &= \prod_{j=1}^{k} (1 - m_j) H(W_k) && (2.27)\\
&\le H(W_k | Z^k S^n) && (2.28)\\
&\le I(W_k; Y_{[k]}^n \,|\, Z^k S^n) + n\epsilon'_{n,k} && (2.29)\\
&\le I(W_k; Y_{[k]}^n, W^{k-1} \,|\, Z^k S^n) + n\epsilon'_{n,k} && (2.30)\\
&= I(W_k; Y_{[k]}^n \,|\, W^{k-1} Z^k S^n) + n\epsilon'_{n,k} && (2.31)
\end{aligned}
$$

where the second inequality is by applying Lemma 7 and noting that S^n is independent of others; (2.29) is from Fano's inequality; the last equality is from $I(W_k; W^{k-1} \,|\, Z^k S^n) = 0$ since the caches Z^k only store disjoint pieces of individual files by the decentralized content placement [2]. Putting all the rate constraints together, and defining $\epsilon_{n,k} \triangleq \epsilon'_{n,k} / \prod_{j=1}^{k} (1 - m_j)$, we have

$$
\begin{aligned}
n(1 - m_1)(R_1 - \epsilon_{n,1}) &\le H(Y_1^n \,|\, Z_1 S^n) - H(Y_1^n \,|\, W_1 Z_1 S^n)\\
&\vdots\\
n \prod_{j=1}^{K} (1 - m_j)(R_K - \epsilon_{n,K}) &\le H(Y_{[K]}^n \,|\, W^{K-1} Z^K S^n)\\
&\quad - H(Y_{[K]}^n \,|\, W^K Z^K S^n). && (2.32)
\end{aligned}
$$

We now sum up the above inequalities with different weights, and apply $K - 1$ times Lemma 6, namely, for $k = 1, \ldots, K - 1$,

$$
\begin{aligned}
\frac{H(Y_{[k+1]}^n \,|\, W^k Z^{k+1} S^n)}{1 - \prod_{j \in [k+1]} \delta_j} &\le \frac{H(Y_{[k+1]}^n \,|\, W^k Z^k S^n)}{1 - \prod_{j \in [k+1]} \delta_j} && (2.33)\\
&\le \frac{H(Y_{[k]}^n \,|\, W^k Z^k S^n)}{1 - \prod_{j \in [k]} \delta_j}, && (2.34)
\end{aligned}
$$

where the first inequality follows because removing conditioning increases entropy. Finally, we have

$$
\begin{aligned}
&\sum_{k=1}^{K} \frac{\prod_{j \in [k]} (1 - m_j)}{1 - \prod_{j \in [k]} \delta_j} (R_k - \epsilon_n)\\
&\quad \le \frac{H(Y_1^n \,|\, Z_1 S^n)}{n(1 - \delta_1)} - \frac{H(Y_{[K]}^n \,|\, W^K Z^K S^n)}{n(1 - \prod_{j \in [k]} \delta_j)} && (2.35)\\
&\quad \le \frac{H(Y_1^n)}{n(1 - \delta_1)} \le 1 && (2.36)
\end{aligned}
$$

which establishes the converse proof.

2.5 Broadcasting Without Receiver Side Information

In this section, we first revisit the algorithm proposed in [59, 60] achieving the capacity region of the EBC with state feedback for some cases of interest, as an important building block of our proposed scheme. Then, we provide an alternative achievability proof for the symmetric channel with uniform erasure probabilities across users.

2.5.1 Revisiting the algorithm by Wang and Gatzianas et al.

We recall the capacity region of the EBC with state feedback as below.

Theorem 8. *[59, 60] For $K \leq 3$, or for the symmetric channel with $K > 3$, or for the one-sided fair rate vector[2] with $K > 3$, the capacity region of the erasure broadcast channel with state feedback is given by*

$$\sum_{k=1}^{K} \frac{1}{1 - \prod_{j=1}^{k} \delta_{\pi_j}} R_{\pi_k} \leq 1, \quad \forall \pi. \tag{2.37}$$

We provide a high-level description of the broadcasting scheme [59,60] which is optimal under the special cases as specified in the above theorem. We recall that the number of private packets $\{F_k\}$ is assumed to be arbitrarily large so that the length of each phase becomes deterministic. Thus, we drop the ϵ_F term wherever confusion is not probable. The broadcasting algorithm has two main roles: 1) broadcast new information packets and 2) multicast side information or overheard packets based on state feedback. Therefore, we can call phase 1 *broadcasting phase* and phases 2 to K *multicasting phase*. Phase j consists of $\binom{K}{j}$ sub-phases in each of which the transmitter sends packets intended to a subset of users $\mathcal{J}$ for $|\mathcal{J}| = j$. Similarly to the receiver side information obtained after the placement phase, we let $\mathcal{L}_{\mathcal{J}}(V_{\mathbb{K}})$ denote the part of packet $V_{\mathbb{K}}$ received by users in $\mathcal{J}$ and erased at users in $[K] \setminus \mathcal{J}$.

Here is a high-level description of the broadcasting algorithm:

1. Broadcasting phase (phase 1): send each message $V_k = W_k$ of F_k packets sequentially for $k = 1, \ldots, K$. This phase generates overheard symbols $\{\mathcal{L}_{\mathcal{J}}(V_k)\}$ to be transmitted via linear combination in multicasting phase, where $\mathcal{J} \subseteq [K] \setminus k$ for all k.

2. Multicasting phase (phases $2 - K$): for a subset $\mathcal{J}$ of users, generate $V_{\mathcal{J}}$ as a linear combination of overheard packets such that

$$V_{\mathcal{J}} = \mathcal{F}_{\mathcal{J}} \left(\{\mathcal{L}_{\mathcal{J} \setminus \mathcal{I} \cup \mathcal{J}'}(V_{\mathcal{I}})\}_{\mathcal{J}' \mathcal{I}: \mathcal{J}' \subset \mathcal{I} \subset \mathcal{J}} \right), \tag{2.38}$$

   where $\mathcal{F}_{\mathcal{J}}$ denotes a linear function. Send $V_{\mathcal{J}}$ sequentially for all $\mathcal{J} \subseteq [K]$ of the cardinality $|\mathcal{J}| = 2, \ldots, K$.

[2] $\delta_k \geq \delta_j$ implies $\delta_k R_k \geq \delta_j R_j$ for any $k \neq j$.

The achievability result of Theorem 8 implies the following corollary.

Corollary 2. *For $K \leq 3$, or for the symmetric channel with $K > 3$, or for the one-sided fair rate vector with $K > 3$, the total transmission length to convey $W_1, \ldots, W_K$ to users $1, \ldots, K$, respectively, is given by*

$$T_{\text{tot}} = \sum_{k=1}^{K} \frac{F_{\pi_k}}{1 - \prod_{j=1}^{k} \delta_{\pi_j}} + \Theta(1).$$

The proof is omitted because the proof in Section 2.6.2 covers the case without user memories.

In order to calculate the total transmission length of the algorithm, we need to introduce further some notations and parameters which are explained as follows.

- A packet intended to $\mathcal{J}$ is consumed *for a given user* $k \in \mathcal{J}$, if this user or at least one user in $[K]\backslash\mathcal{J}$ receives it. The probability of such event is equal to $1-\prod_{j\in[K]\backslash\mathcal{J}\cup\{k\}} \delta_j$.

- A packet intended to $\mathcal{I}$ becomes a packet intended to $\mathcal{J}$ and useful for user $k \in \mathcal{I} \subset \mathcal{J} \subseteq [K]$, if erased at user k and all users in $[K]\setminus\mathcal{J}$ but received by $\mathcal{J}\setminus\mathcal{I}$. The number of packets useful for user k generated in sub-phase $\mathcal{I}$ and to be sent in sub-phase $\mathcal{J}$, denoted by $N_{\mathcal{I}\to\mathcal{J}}^{\{k\}}$, is then given by

$$N_{\mathcal{I}\to\mathcal{J}}^{\{k\}} = t_{\mathcal{I}}^{\{k\}} \prod_{j'\in[K]\backslash\mathcal{J}\cup\{k\}} \delta_{j'} \prod_{j\in\mathcal{J}\backslash\mathcal{I}} (1-\delta_j) \tag{2.39}$$

  where $t_{\mathcal{I}}^{\{k\}}$ denotes the length of sub-phase $\mathcal{I}$ viewed by user k to be defined shortly. We can also express $N_{\mathcal{I}\to\mathcal{J}}^{\{k\}}$ as

$$N_{\mathcal{I}\to\mathcal{J}}^{\{k\}} = \sum_{\mathcal{I}'\subseteq\mathcal{I}\backslash k} |\mathcal{L}_{\mathcal{J}\backslash\mathcal{I}\cup\mathcal{I}'}(V_{\mathcal{I}}^{\{k\}})|, \tag{2.40}$$

  where we let $V_{\mathcal{I}}^{\{k\}}$ denotes the part of $V_{\mathcal{I}}$ required for user k.

- The duration $t_{\mathcal{J}}$ of sub-phase $\mathcal{J}$ is given by

$$t_{\mathcal{J}} = \max_{k\in\mathcal{J}} t_{\mathcal{J}}^{\{k\}}, \tag{2.41}$$

  where

$$t_{\mathcal{J}}^{\{k\}} = \frac{\sum_{k\in\mathcal{I}\subset\mathcal{J}} N_{\mathcal{I}\to\mathcal{J}}^{\{k\}}}{1 - \prod_{j\in[K]\backslash\mathcal{J}\cup\{k\}} \delta_j}. \tag{2.42}$$

The total transmission length is given by summing up all sub-phases, i.e. $T_{\text{tot}} = \sum_{\mathcal{J}\subseteq[K]} t_{\mathcal{J}}$.

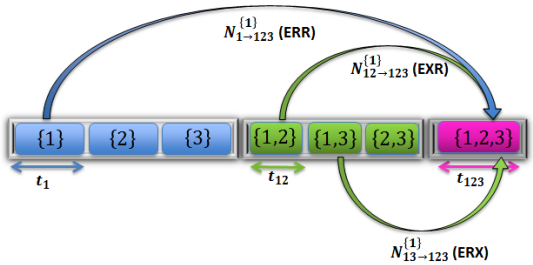


Figure 2.3: Phase organization for $K = 3$ and packet evolution viewed by user 1.

Fig. 2.3 illustrates the phase organization for $K = 3$ and the packet evolution viewed by user 1. The packets intended to $\{1, 2, 3\}$ are created from both phases 1 and 2. More precisely, sub-phase $\{1\}$ creates $\mathcal{L}_{23}(V_1)$ to be sent in phase 3 if erased at user 1 and received by others (ERR). The number of such packets is $N^{\{1\}}_{1\to 123}$. Sub-phase $\{1, 2\}$ creates $\mathcal{L}_3(V_{12}), \mathcal{L}_{23}(V_{12})$ if erased at user 1 but received by user 3 (EXR), while sub-phase $\{1, 3\}$ creates $\mathcal{L}_2(V_{13}), \mathcal{L}_{23}(V_{13})$ if erased at user 1 and received by user 2 (ERX). The total number of packets intended to $\{1, 2, 3\}$ generated in phase 2 and required by user 1 is $N^{\{1\}}_{12\to 123} + N^{\{1\}}_{13\to 123}$.

2.5.2 Achievability in the symmetric channel

We focus now on the special case of the symmetric channel with uniform erasure probabilities, i.e. $\delta_k = \delta$ for all k. In this case, the capacity region of the EBC with state feedback in (2.37) simplifies to

$$\sum_{k=1}^{K} \frac{1}{1-\delta^k} R_{\pi_k} \leq 1, \quad \forall \pi. \tag{2.43}$$

It readily follows that the capacity region yields the symmetric capacity, i.e. $R_1 = \cdots = R_K = R_{\text{sym}}(K)$, given by

$$R_{\text{sym}}(K) = \frac{1}{\sum_{k=1}^{K} \frac{1}{1-\delta^k}}. \tag{2.44}$$

In the following, we provide an alternative proof of the achievability of the symmetric capacity. Notice that other vertices of the capacity region can be characterized similarly as proved in subsection 2.6.3. Our proof follows the footsteps of [69] and uses the notion of order-j packets. Let us define message set $\{W_{\mathcal{J}}\}$ independently and uniformly distributed over $\{\mathcal{W}_{\mathcal{J}}\}$ for all $\mathcal{J} \subseteq [K]$. For $\mathcal{J}$ with the cardinality $j = |\mathcal{J}|$, the message set $\{W_{\mathcal{J}}\}$ are called order-j messages. We define $R_{\mathcal{J}}$ an achievable rate of the message $W_{\mathcal{J}}$ and define the sum rate of order-j messages as

$$R^j(K) \triangleq \sum_{\mathcal{J}:|\mathcal{J}|=j} R_{\mathcal{J}} = \binom{K}{j} R_{\mathcal{J}}. \tag{2.45}$$

The supremum of $R^j(K)$ is called the sum capacity of order-j messages. We characterize the sum capacity of order-j messages, in the erasure broadcast channel with state feedback in the following theorem.

Theorem 9. *In the K-user erasure broadcast channel with state feedback, the sum capacity of order-j packets is upper bounded by*

$$R^j(K) \leq \frac{\binom{K}{j}}{\sum_{k=1}^{K-j+1} \frac{\binom{K-k}{j-1}}{1-\delta^k}}, \quad j = 1, \ldots, K. \tag{2.46}$$

The algorithms in [59, 60] achieve the RHS with equality.

Proof. We first provide the converse proof. Similarly to section 2.4, we build on genie-aided bounds together with Lemma 6. Let us assume that the transmitter wishes to convey the message $W_{\mathcal{J}}$ to a subset of users $\mathcal{J} \subseteq \{1, \ldots, K\}$, and receiver k wishes to decode all messages $\tilde{W}_k \stackrel{\Delta}{=} \{W_{\mathcal{J}}\}_{\mathcal{J}:\mathcal{J}\ni k}$ for $j = 1, \ldots, K$. In order to create a degraded broadcast channel, we assume that receiver k provides the message set $\tilde{W}_k$ and the channel output Y_k^n to receivers $k+1$ to K for $k = 1, \ldots, K-1$, Under this setting and using Fano's inequality, we have for receiver 1 :

$$n\left(\sum_{1\in\mathcal{J}\subseteq[K]} R_{\mathcal{J}} - \epsilon_{n,1}\right) \leq H(Y_1^n|S^n) - H(Y_1^n|\tilde{W}_1 S^n). \tag{2.47}$$

For receiver $k = 2, \ldots, K$, we have:

$$\begin{aligned} n\left(\sum_{k\in\mathcal{J}\subseteq\{k,\ldots,K\}} R_{\mathcal{J}} - \epsilon_{n,k}\right) &\leq H(Y_1^n \ldots Y_k^n|\tilde{W}^{k-1} S^n) \\ &\quad - H(Y_1^n \ldots Y_k^n|\tilde{W}^k S^n), \end{aligned} \tag{2.48}$$

where we used $\tilde{W}_k \backslash \tilde{W}^{k-1} = \{W_{\mathcal{J}}\}_{\mathcal{J}:\mathcal{J}\backslash\{k,\ldots,K\}}$ in the LHS. Summing up the above inequalities and applying Lemma 6 $K-1$ times, we readily obtain:

$$\sum_{k=1}^{K} \frac{\sum_{k\in\mathcal{J}\subseteq\{k,\ldots,K\}}(R_{\mathcal{J}} - \epsilon_{n,k})}{1-\delta^k} \leq \frac{H(Y_1^n|S^n)}{n(1-\delta)} \tag{2.49}$$

$$\leq 1. \tag{2.50}$$

We further impose the symmetric rate condition such that $R_{\mathcal{J}} = R_{\mathcal{J}'}$ for any $\mathcal{J} \neq \mathcal{J}'$ with the same cardinality. By focusing on $\mathcal{J}$ of the same cardinality j in (2.49) and noticing that there are $\binom{K-k}{j-1}$ such subset, $R_{\mathcal{J}}$ is upper bounded by

$$R_{\mathcal{J}} \leq \frac{1}{\sum_{k=1}^{K-j+1} \frac{\binom{K-k}{j-1}}{1-\delta^k}}, \quad \forall\, \mathcal{J},\ |\mathcal{J}| = j. \tag{2.51}$$

This establishes the converse part.

In order to prove the achievability of $R^i(K)$ in Theorem 9, we apply the broadcasting algorithm of [59,60] from phase $i > 1$ by sending N_i packets to each subset $\mathcal{I} \subseteq [K]$ with $|\mathcal{I}| = i$. First, we redefine some parameters by taking into account the symmetry across users. Due to the symmetry, we drop the user index k in $t_{\mathcal{J}}^{\{k\}}$, $N_{\mathcal{I}\to\mathcal{J}}^{\{k\}}$ and replace them by t_j, $N_{i\to j}$, respectively for $\mathcal{I} \subset \mathcal{J} \subseteq [K]$ with $|\mathcal{I}| = i, |\mathcal{J}| = j$. Now, we introduce variants of these notations to reflect the fact that the algorithm starts from phase $i > 1$, rather than from phase 1. The length of any sub-phase in phase j when starting the algorithm from phase i, denoted by t_j^i, is given by

$$t_j^i = \frac{1}{1-\delta^{K-j+1}} \sum_{l=i}^{j-1} \binom{j-1}{l-1} N_{l\to j}^i, \quad j > i, \tag{2.52}$$

where

$$N_{l\to j}^i = t_l^i \delta^{K-j+1}(1-\delta)^{j-l} \tag{2.53}$$

denotes the number of order-j packets generated during a given sub-phase in phase i, again starting from phase i.

For $j = i$, we have

$$t_i^i = \frac{N_i}{1-\delta^{K-i+1}}. \tag{2.54}$$

By counting the total number of order-i packets and the transmission length from phase i to phase K, the sum rate of order-i messages achieved by the algorithm [59,60] is given by

$$\tilde{R}^i(K) = \frac{\binom{K}{i} N_i}{\sum_{j=i}^K \binom{K}{j} t_j^i}, \quad \forall\, i. \tag{2.55}$$

It remains to prove that $\tilde{R}^i(K)$ coincides with the RHS expression of (2.46). We notice that the transmission length from phase j to K can be expressed in the following different way, i.e.

$$\sum_{j=i}^K \binom{K}{j} t_j^i = \sum_{j=i}^K U_j^i, \tag{2.56}$$

where we let

$$U_j^i = \sum_{l=i}^j \binom{j-1}{l-1} t_l^i, \quad \forall\, j \geq i. \tag{2.57}$$

By following similar steps as [59, Appendix C], we obtain the recursive equations given by

$$U_j^i = \frac{1}{1-\delta^{K-j+1}} \sum_{l=1}^{j-i} \binom{j-1}{l} (-1)^{l+1} (1-\delta^{K-j+l+1}) U_{j-l}^i \tag{2.58}$$

for $j > i$. Since we have $U_i^i = t_i^i = \frac{N_i}{1-\delta^{K-i+1}}$ and using the equality $\binom{j-1}{c}\binom{j-c-1}{i-1} = \binom{j-1}{j-i}\binom{j-i}{c}$ and the binomial theorem $\sum_{k=0}^{n} \binom{n}{k} x^k y^{n-k} = (x+y)^n$, it readily follows that we have

$$U_j^i = \frac{N_i}{1-\delta^{K-j+1}} \binom{j-1}{j-i}, \quad j \geq i. \tag{2.59}$$

By plugging the last expression into (2.55) using (2.56), we have

$$\tilde{R}^i(K) = \frac{\binom{K}{i} N_i}{\sum_{j=i}^{K} \frac{N_i}{1-\delta^{K-j+1}} \binom{j-1}{j-i}} \tag{2.60}$$

$$= \frac{\binom{K}{i}}{\sum_{k=1}^{K-i+1} \frac{\binom{K-k}{i-1}}{1-\delta^k}} \tag{2.61}$$

which coincides the RHS of (2.46) for $i = 1, \ldots, K$. This establishes the achievability proof. □

As a corollary of Theorem 9, we provide an alternative expression for the sum capacity.

Corollary 3. *The sum capacity of the K-user symmetric broadcast erasure channel with state feedback can be expressed as a function of $R^2(K), \ldots, R^K(K)$ by*

$$R^1(K) = \frac{KN_1}{\frac{KN_1}{1-\delta^K} + \sum_{i=2}^{K} \frac{\binom{K}{i} N_{1\to i}}{R^i(K)}}, \tag{2.62}$$

where $\frac{KN_1}{1-\delta^K}$ is the duration of phase 1, $\binom{K}{j} N_{1\to j}$ corresponds to the total number of order-j packets generated in phase 1.

Proof. By letting f denote the RHS of (2.62), we wish to prove the equality $f = R^1(K) = \frac{K}{\sum_{k=1}^{K} \frac{1}{1-\delta^k}}$ by proving $f = \tilde{R}^1(K)$. If it is true, from the achievability proof of Theorem 9 that proves $\tilde{R}^i = R^i$ for all i, the proof is complete. In the RHS of (2.62), we replace R^i by the expression $\tilde{R}^i$ in (2.55) by letting $N_{1\to i} = N_i$ for $i \geq 2$. Then, we have

$$f = \frac{KN_1}{\frac{KN_1}{1-\delta^K} + \sum_{i=2}^{K} \sum_{j=i}^{K} \binom{K}{j} t_j^i} \tag{2.63}$$

$$= \frac{KN_1}{\frac{KN_1}{1-\delta^K} + \sum_{j=2}^{K} \binom{K}{j} \sum_{i=2}^{j} t_j^i}. \tag{2.64}$$

Comparing the desired equality $f = \tilde{R}^1(K) = \frac{KN_1}{\sum_{j=1}^{K}\binom{K}{j}t_j^1}$ with the above expression and noticing that $\frac{KN_1}{1-\delta^K} = Kt_1^1$, we immediately see that it remains to prove the following equality.

$$t_j^1 = \sum_{i=2}^{j} t_j^i \qquad \forall j \geq 2. \tag{2.65}$$

We prove this relation recursively. For $j = 2$, the above equality follows from (2.52) and (2.54).

$$t_2^1 = \frac{N_{1\to 2}}{1-\delta^{K-1}} = t_2^2. \tag{2.66}$$

Now suppose that (2.65) holds for $l = 2, \ldots, j-1$ and we prove it for j. From (2.52) we have

$$t_j^1 = \frac{1}{1-\delta^{K-j+1}} \sum_{l=1}^{j-1} \binom{j-1}{l-1} N_{l\to j}^1 \tag{2.67}$$

$$= \frac{1}{1-\delta^{K-j+1}} \left[N_{1\to j} + \sum_{l=2}^{j-1} \binom{j-1}{l-1} t_l^1 \delta^{K-j+1}(1-\delta)^{j-l} \right] \tag{2.68}$$

$$= \frac{1}{1-\delta^{K-j+1}} \left[N_{1\to j} + \sum_{l=2}^{j-1} \binom{j-1}{l-1} \sum_{i=2}^{l} t_l^i \delta^{K-j+1}(1-\delta)^{j-l} \right] \tag{2.69}$$

$$= \frac{1}{1-\delta^{K-j+1}} \left[N_{1\to j} + \sum_{l=2}^{j-1} \binom{j-1}{l-1} \sum_{i=2}^{l} N_{l\to j}^i \right] \tag{2.70}$$

$$= \frac{1}{1-\delta^{K-j+1}} \left[N_{1\to j} + \sum_{i=2}^{j-1} \sum_{l=i}^{j-1} \binom{j-1}{l-1} N_{l\to j}^i \right] \tag{2.71}$$

$$= t_j^j + \sum_{i=2}^{j-1} t_j^i, \tag{2.72}$$

where (2.68) follows from (2.53); (2.69) follows from our hypothesis (2.65); (2.70) follows from (2.53); (2.71) is due to the equality $\sum_{l=2}^{j-1}\sum_{i=2}^{l} = \sum_{i=2}^{j-1}\sum_{l=i}^{j-1}$; the last equality is due to (2.52). Therefore, the desired equality holds also for j. This completes the proof of Corollary 3. □

2.6 Achievability

We provide the achievability proof of Theorem 5 for the case of one-sided fair rate vector as well as the symmetric network. The proof for the case of $K = 3$ is omitted, since it is a straightforward extension of [60, Section V].

2.6.1 Proposed delivery scheme for $K > 3$

We describe the proposed delivery scheme for the case of $K > 3$ assuming that user k requests file W_k of size F_k packets for $k = 1, \ldots, K$ without loss of generality. Compared to the algorithm [59,60] revisited previously, our scheme must convey packets created during the placement phase as well as all previous phases in each phase. Here is a high-level description of our proposed delivery scheme.

1. Placement phase (phase 0) detailed in subsection 1.2.1: fill the caches $Z_1, \ldots, Z_K$ according to the decentralized content placement (see subsection 1.2.1). Let $\mathcal{L}_{\mathcal{J}}(W_i)$ denote the sub-file of W_i stored exclusively by the users in $\mathcal{J}$. This phase creates "overheard" packets $\{\mathcal{L}_{\mathcal{J}\backslash k}(W_k)\}$ for $\mathcal{J} \subset [K]$ and all k to be delivered during phases 1 to K.

2. Broadcasting phase (phase 1): the transmitter sends $V_1, \ldots, V_K$ sequentially until at least one user receives it, where $V_k = \mathcal{L}_{\emptyset}(W_k)$ corresponds to the order-1 packets.

3. Multicasting phase (phases 2-K): for a subset $\mathcal{J}$ of users, generate $V_{\mathcal{J}}$ as a linear combination of overheard packets during the placement phase as well as during phases 1 to $j-1$. Send $V_{\mathcal{J}}$ sequentially for $\mathcal{J} \subseteq [K]$,

$$V_{\mathcal{J}} = \mathcal{F}_{\mathcal{J}}\left(\{\mathcal{L}_{\mathcal{J}\backslash\mathcal{I}\cup\mathcal{J}'}(V_{\mathcal{I}})\}_{\mathcal{J}'\mathcal{I}:\mathcal{J}'\subset\mathcal{I}\subset\mathcal{J}}, \mathcal{L}_{\mathcal{J}\backslash\{k\}}(W_k)\right). \tag{2.73}$$

The proposed delivery scheme achieves the optimal rate region only in two special cases. We provide the proof separately in upcoming subsections.

2.6.2 Proof of Theorem 5 for the case of one-sided fair rate vector

We assume without loss of generality $\delta_1 \geq \cdots \geq \delta_K$, $\delta_1 R_1 \geq \cdots \geq \delta_K R_K$, and $\frac{1-m_1}{m_1} R_1 \geq \cdots \geq \frac{1-m_2}{m_2} R_K$. Under this setting, we wish to prove the achievability of the following equality.

$$\sum_{k=1}^{K} \frac{\prod_{j=1}^{k}(1-m_j)}{1-\prod_{j=1}^{k}\delta_j} R_k = 1. \tag{2.74}$$

By replacing $R_k = \frac{F_{d_k}}{T_{\text{tot}}}$ and further assuming $d_k = k$ for all k without loss of generality, the above equality is equivalent to

$$T_{\text{tot}} = \sum_{k=1}^{K} \frac{\prod_{j=1}^{k}(1-m_j)}{1-\prod_{j=1}^{k}\delta_j} F_k. \tag{2.75}$$

The rest of the subsection is dedicated to the proof of the total transmission length (2.75). We start by rewriting $t_{\mathcal{J}}^{\{k\}}$ in (2.42) by incorporating the packets generated during the placement phase. Namely we have for $k \in \mathcal{J} \subseteq [K]$

$$t_{\mathcal{J}}^{\{k\}} = \frac{\sum_{\mathcal{I}:k\in\mathcal{I}\subset\mathcal{J}} N_{\mathcal{I}\to\mathcal{J}}^{\{k\}} + |\mathcal{L}_{\mathcal{J}\setminus\{k\}}(W_k)|}{1-\prod_{j\in[K]\setminus\mathcal{J}\cup\{k\}}\delta_j}. \tag{2.76}$$

We recall that the length of sub-phase $\mathcal{J}$ is given by $t_{\mathcal{J}} = \max_{k\in\mathcal{J}} t_{\mathcal{J}}^{\{k\}}$. Our proof consists of four steps.

Step 1 We express $t_{\mathcal{J}}^{\{k\}}$ as a function of key parameters $\{\delta_k\}, \{m_k\}, \{F_k\}$ in two different ways. By following similar steps as in [59, Appendix C], the aggregate length of sub-phases $\mathcal{I} \subseteq \mathcal{J}$ required by user k for a fixed $\mathcal{J} \subseteq [K]$ is given by

$$\sum_{\mathcal{I}:k\in\mathcal{I}\subseteq\mathcal{J}} t_{\mathcal{I}}^{\{k\}} = \frac{\prod_{j\in[K]\setminus\mathcal{J}\cup\{k\}}(1-m_j)}{1-\prod_{j\in[K]\setminus\mathcal{J}\cup\{k\}}\delta_j} F_k. \tag{2.77}$$

We have an alternative expression for $t_{\mathcal{J}}^{\{k\}}$ which is useful as will be seen shortly. The length of sub-phase $\mathcal{J}$ needed by user k such that $k \in \mathcal{J} \subseteq [K]$ is equal to

$$t_{\mathcal{J}}^{\{k\}} = \sum_{\mathcal{H}:\mathcal{H}\subseteq\mathcal{J}\setminus\{k\}} (-1)^{|\mathcal{H}|} \frac{\prod_{j\in[K]\setminus\mathcal{J}\cup\{k\}\cup\mathcal{H}}(1-m_j)}{1-\prod_{j\in[K]\setminus\mathcal{J}\cup\{k\}\cup\mathcal{H}}\delta_j} F_k. \tag{2.78}$$

The proof is provided in Appendix A.2.

Step 2 The length of sub-phase $\mathcal{J}$ is determined by the worst user which requires the maximum length, i.e. $\arg\max_{k\in\mathcal{J}} t_{\mathcal{J}}^{\{k\}}$. For the special case of one-sided fair rate vector, by using (2.78) it is possible to prove that the worst user is given by

$$\arg\max_{k\in\mathcal{J}} t_{\mathcal{J}}^{\{k\}} = \min\{\mathcal{J}\} \quad , \forall\, \mathcal{J} \subseteq [K], \tag{2.79}$$

where $\min\{\mathcal{J}\}$ is the smallest index in the set of users $\mathcal{J}$ that corresponds to the user with the largest erasure probability. The proof is provided in Appendix A.3. This means that the user permutation (which determines the sub-phase length) is preserved in all sub-phases for the one-sided fair rate vector.

Step 3 By combining the two previous steps, the total transmission length can be derived as follows.

$$T_{\text{tot}} = \sum_{\mathcal{J}:\mathcal{J}\subseteq[K]} \max_{k\in\mathcal{J}} t_{\mathcal{J}}^{\{k\}} \tag{2.80}$$

$$= \sum_{\mathcal{J}:\mathcal{J}\subseteq[K]} t_{\mathcal{J}}^{\{\min \mathcal{J}\}} \tag{2.81}$$

$$= \sum_{k=1}^{K} \sum_{\mathcal{J}:k\in\mathcal{J}\subseteq\{k,\ldots,K\}} t_{\mathcal{J}}^{\{k\}} \tag{2.82}$$

$$= \sum_{k=1}^{K} F_k \frac{\prod_{j=1}^{k}(1-m_j)}{1-\prod_{j=1}^{k}\delta_j}, \tag{2.83}$$

where (2.81) is obtained from (2.79); the last equality follows from (2.77). Then, we obtain the desired equality (2.75).

Step 4 The final step is to prove that under the one-sided fair rate vector (2.74) implies all the other $K!-1$ inequalities of the rate region (2.9). This is proved in Appendix A.4. Hence, the achievability proof for the one-sided rate vector is completed.

2.6.3 Proof of Theorem 5 for the symmetric network

First we recall the rate region of the symmetric network with uniform channel statistics and memory sizes given in (2.10),

$$\sum_{k=1}^{K} \frac{(1-m)^k}{1-\delta^k} R_{\pi_k} \leq 1, \quad \forall \pi. \tag{2.84}$$

Exploiting the polyhedron structure and following the same footsteps as [69, Section V], we can prove that the vertices of the above rate region are characterized as:

$$R_k = \begin{cases} R_{\text{sym}}(|\mathcal{K}|), k \in \mathcal{K} \\ 0, k \notin \mathcal{K} \end{cases} \tag{2.85}$$

for $\mathcal{K} \subseteq [K]$, where the symmetric rate $R_{\text{sym}}(K)$ is given by

$$R_{\text{sym}}(K) = \frac{1}{\sum_{k=1}^{K} \frac{(1-m)^k}{1-\delta^k}}. \tag{2.86}$$

This means that when only $|\mathcal{K}|$ users are active in the system, each of these users achieves the same symmetric rate as the reduced system of dimension $|\mathcal{K}|$. Then, it suffices to prove the achievability of the symmetric rate for a given dimension K. As explained

in subsection 2.6.1, the placement phase generates "overheard packets" $\{\mathcal{L}_{\mathcal{J}\backslash k}(W_k)\}$ for $\mathcal{J} \subseteq [K]$ and all k. We let $N_{0\to j} = |\mathcal{L}_{\mathcal{J}\backslash k}(W_k)|$ denote the number of order-j packets created during the placement phase. Then, we can express the sum rate of the cached-enabled EBC by incorporating the packets generated from the placement phase into (2.62) as follows,

$$KR_{\text{sym}}(K) = \frac{KF}{\frac{KN_{0\to 1}}{\beta_1} + \sum_{j=2}^{K} \frac{\binom{K}{j}(N_{0\to j}+N_{1\to j})}{R^j(K)}}. \tag{2.87}$$

By repeating the same steps as the proof of Corollary 3, it readily follows that the above expression boils down to $\frac{K}{\sum_{k=1}^{K} \frac{(1-m)^k}{1-\delta^k}}$. This establishes the achievability proof for the symmetric network.

2.7 Extensions

In this section, we provide rather straightforward extensions of our previous results to other scenarios such as the centralized content placement and the multi-antenna broadcast channel with the state feedback.

2.7.1 Centralized content placement

So far, we have focused on the decentralized content placement. We shall show in this subsection that the rate region under the decentralized content placement can be easily modified to the case of the centralized content placement proposed in [1] and recalled in subsection 1.2.2. We restrict ourselves to the symmetric memory size $M_k = M$ such that $M \in \{0, N/K, 2N/K, \ldots, N\}$ so that the parameter $b = \frac{MK}{N}$ is an integer. In analogy to Lemma 7 for the decentralized content placement, we can characterize the message entropy given the receiver side information.

Lemma 10. *For the centralized content placement [1], the following equalities hold for any i and $\mathcal{I} \subseteq [K]$*

$$H(W_i \,|\, \{Z_k\}_{k\in\mathcal{I}}) = \frac{\binom{K-|\mathcal{I}|}{b}}{\binom{K}{b}} H(W_i).$$

Proof. Under the centralized content placement

$$H(W_i \,|\, \{Z_k\}_{k\in\mathcal{I}}) = \sum_{\mathcal{J}\subseteq[K]\setminus\mathcal{I}} H(\mathcal{L}_{\mathcal{J}}(W_i)) \tag{2.88}$$

$$= \sum_{\mathcal{J}\subseteq[K]\setminus\mathcal{I};|\mathcal{J}|=b} H(\mathcal{L}_{\mathcal{J}}(W_i)) \tag{2.89}$$

$$= \sum_{\mathcal{J}\subseteq[K]\setminus\mathcal{I};|\mathcal{J}|=b} \frac{1}{\binom{K}{b}} H(W_i) \tag{2.90}$$

$$= \frac{\binom{K-|\mathcal{I}|}{b}}{\binom{K}{b}} H(W_i), \tag{2.91}$$

where the first equality follows by repeating the same steps from (2.19) to (2.22); (2.89) and (2.90) follows from the definition of the centralized content placement (1.6). □

Then, we present the rate region of the cache-enabled EBC under the centralized content placement.

Theorem 11. *For the symmetric network, the rate region of the cached-enabled EBC with the state feedback under the centralized content placement is given by*

$$\sum_{k=1}^{K-b} \frac{\binom{K-k}{b}/\binom{K}{b}}{1-\delta^k} R_{\pi_k} \leq 1 \tag{2.92}$$

for any permutation π *of* $\{1,\ldots,K\}$.

Proof. Following the same steps as in section 2.4 and replacing Lemma 7 with Lemma 10, the converse proof follows immediately.

For achievability, as explained in subsection 2.6.3, it is sufficient to consider the case of symmetric rate for a given dimension. By focusing without loss of generality on the dimension K, we fix the number of packets per user to be F and prove that our proposed scheme can deliver requested files to users within the total transmission length given by

$$T_{\text{tot}} = F \sum_{k=1}^{K-b} \frac{\binom{K-k}{b}/\binom{K}{b}}{1-\delta^k} + \Theta(1), \tag{2.93}$$

as $F \to \infty$. We proceed our proposed delivery scheme from phase $b+1$ by sending packets of order $b+1$. More precisely, in phase $b+1$ we generate and send the packets intended to $\mathcal{J}$ by the following linear combination

$$V_{\mathcal{J}} = \mathcal{F}_{\mathcal{J}}\left(\mathcal{L}_{\mathcal{J}\setminus k}(W_k)\right), \tag{2.94}$$

for $\mathcal{J} \subseteq [K]$ with $|\mathcal{J}| = b+1$. In subsequent phases $b+2$ to K, we repeat

$$V_{\mathcal{J}} = \mathcal{F}_{\mathcal{J}}\left(\{\mathcal{L}_{\mathcal{J}\setminus\mathcal{I}\cup\mathcal{I}'}(V_{\mathcal{I}})\}_{\mathcal{I}'\mathcal{I}:\mathcal{I}'\subset\mathcal{I}\subset\mathcal{J}}\right) \tag{2.95}$$

for $\mathcal{J} \subseteq [K]$ with $|\mathcal{J}| = b+2, \ldots, K$. In order to calculate the total transmission length required by our delivery algorithm, we follow the same footsteps as in subsection 2.6.3 and exploit Theorem 9 on the sum capacity of order-i messages that we recall here for the sake of clarity.

$$R^i(K) = \frac{\binom{K}{i}}{\sum_{k=1}^{K-i+1} \frac{\binom{K-k}{i-1}}{1-\delta^k}}. \tag{2.96}$$

Noticing that there are $\binom{K}{b+1}$ sub-phases in phase $b+1$ and in each sub-phase we send a linear combination whose size is $\frac{F}{\binom{K}{b}}$, the total transmission length is given by

$$T_{\text{tot}} = \frac{\binom{K}{b+1}/\binom{K}{b}}{R^{b+1}} F \tag{2.97}$$

$$= \sum_{k=1}^{K-b} \frac{\binom{K-k}{b}/\binom{K}{b}}{1-\delta^k} F, \tag{2.98}$$

where the last equality follows by plugging the expression R^{b+1}. □

For the case without erasure, Theorem 11, in particular, the expression of the transmission length in (2.93), becomes the rate-memory tradeoff under the centralized content placement [1] given by

$$\frac{T_{\text{tot}}}{F} = K\left(1 - M/N\right) \frac{1}{1 + KM/N}. \tag{2.99}$$

2.7.2 MISO-BC

We consider the multi-input single-output broadcast channel (MISO-BC) between a N_t-antennas transmitter and K single-antenna receivers. The channel state S_l in slot l is given by the $N_t \times K$ matrix and we restrict ourselves to the i.i.d. channels across time and users. Here, we are interested in the capacity scaling in the high signal-to-noise ratio (SNR) regime and define the degree of freedom (DoF) of user k as

$$\mathsf{DoF}_k = \lim_{\text{SNR}\to\infty} \frac{R_k}{\log_2 \text{SNR}}.$$

We define the sum DoF of order-j messages given by

$$\mathsf{DoF}^j = \lim_{\text{SNR}\to\infty} \sum_{\mathcal{J}:|\mathcal{J}|=j} \frac{R_{\mathcal{J}}}{\log_2 \text{SNR}}. \tag{2.100}$$

First we recall the main results on the MISO-BC with state feedback by Maddah-Ali and Tse [69]. In [69, Theorem 3], the DoF region of the MISO-BC with state feedback has

been characterized as

$$\sum_{k=1}^{K} \frac{\mathsf{DoF}_{\pi_k}}{k} \leq 1, \quad \forall \pi. \tag{2.101}$$

The sum DoF of order-j messages has been characterized in [69, Theorem 2] for $N_t \geq K - j + 1$ and is given by

$$\mathsf{DoF}^j = \frac{\binom{K}{j}}{\sum_{k=1}^{K-j+1} \frac{\binom{K-k}{j-1}}{k}}. \tag{2.102}$$

It is worth comparing the DoF region of the MISO-BC in (2.101) and the capacity region of the EBC in (2.43). In fact, as remarked in [68], both regions have exactly the same structure and can be unified through a parameter $\alpha_k = k$ for the MISO-BC and $\alpha_k = 1-\delta^k$ for the EBC. The same holds for the sum DoF of order-j messages in the MISO-BC in (2.102) and the sum capacity of order-j packets in the EBC characterized in Theorem 9. By exploiting this duality and replacing $1 - \delta^k$ with k in the rate region of the symmetric EBC (2.10), we can easily characterize the DoF region of the cache-enabled MISO-BC with state feedback. Namely, under the decentralized content placement, the DoF region is given by

$$\sum_{k=1}^{K} \frac{(1-m)^k}{k} \mathsf{DoF}_{\pi_k} \leq 1, \ \ \forall \pi \tag{2.103}$$

for $N_t \geq K$, while under the centralized content placement, the DoF region is given by

$$\sum_{k=1}^{K-b} \frac{\binom{K-k}{b}/\binom{K}{b}}{k} \mathsf{DoF}_{\pi_k} \leq 1, \ \ \forall \pi \tag{2.104}$$

for $N_t \geq K - b$. The converse follows exactly in the same manner except that we use the entropy inequality for the MISO-BC given in [68, Lemma 4] by replacing the entropy by the differential entropy and again $1 - \delta^k$ by k. The achievability can be proved by modifying the scheme in [69] to the case of receiver side information along the line of [70].

As a final remark, for the case of the centralized content placement, our DoF region in (2.104) yields the following transmission length

$$T_{\text{tot}} = \sum_{k=1}^{K-b} \frac{\binom{K-k}{b}/\binom{K}{b}}{k} F, \tag{2.105}$$

which coincides with [46, Corollary 2b].

2.8 Numerical Examples

In this section, we provide some numerical examples to show the performance of our proposed delivery scheme. Fig. 2.4 illustrates the tradeoff between the erasure probability and the memory size for the symmetric network with $K = 3$ for the case of the

decentralized content placement Each curve corresponds to a different symmetric rate $R_{\text{sym}}(3) = \frac{1}{\sum_{k=1}^{3} \frac{(1-m)^k}{1-\delta^k}}$. The arrow shows the increasing symmetric rate from 1/3, corresponding to case with no memory and no erasure, to infinity. The memory size increases the rate performance even in the presence of erasure and the benefit of caching is significant for smaller erasure probabilities as expected from the analytical expression.

Fig. 2.5 compares the transmission length T_{tot}, normalized by the file size F, achieved by our delivery scheme with feedback and the scheme without feedback for the case of the decentralized content placement. We consider the system with $N = 100, K = 10$ and the erasure probabilities of $\delta = 0$ (perfect link), 0.2, and 0.6. We observe that state feedback can be useful especially when the memory size is small and the erasure probability is large. In fact, it can be easily shown that the rate region of the cached-enabled EBC without feedback under the decentralized content placement is given by

$$\sum_{k=1}^{K} \frac{\left(1 - \frac{M}{N}\right)^k}{1-\delta} R_{\pi_k} \leq 1 \tag{2.106}$$

where the denominator in the LHS reflects the fact that each packet must be received by all K users. This yields the transmission length given by

$$T_{\text{tot}-\text{noFB}} = \frac{\sum_{k=1}^{K} \left(1 - \frac{M}{N}\right)^k}{1-\delta} F + \Theta(1). \tag{2.107}$$

Under the centralized content placement, the rate region of the cached-enabled EBC without feedback is given by

$$\sum_{k=1}^{K-b} \frac{\binom{K-k}{b} / \binom{K}{b}}{1-\delta} R_{\pi_k} \leq 1 \tag{2.108}$$

yielding

$$T_{\text{tot}-\text{noFB}} = \frac{K\left(1 - M/N\right) \frac{1}{1+KM/N}}{1-\delta} F + \Theta(1). \tag{2.109}$$

Without state feedback, the transmission length in (2.107), (2.109) corresponds to the transmission length over the perfect link expanded by a factor $\frac{1}{1-\delta} > 1$, because each packet must be received by all users. The merit of feedback becomes significant if the packets of lower-order dominate the order-K packets. The case of small normalized memory $m = \frac{M}{N}$ and large erasure probability corresponds to such a situation.

Fig. 2.6 plots the normalized transmission length T_{tot}/F versus the memory size M in the symmetric network with $N = 100, K = 10$. We compare the performance with and without feedback under the decentralized and the centralized caching for $\delta = 0$ and $\delta = 0.6$. The relative merit of the centralized content placement compared to the decentralized the counterpart can be observed.

Table 2.1: Optimal memory allocation for the lower bound.

Average memory	user 1 $\delta_1 = 0.2$	user 2 $\delta_2 = 0.4$	user 3 $\delta_3 = 0.6$	user 4 $\delta_4 = 0.8$
$\frac{\sum_{k=1}^{4} M_k}{4} = 4$	0	0	5	11
$\frac{\sum_{k=1}^{4} M_k}{4} = 8$	1	7	10	14
$\frac{\sum_{k=1}^{4} M_k}{4} = 12$	8	11	13	16
$\frac{\sum_{k=1}^{4} M_k}{4} = 16$	14	15	17	18

Fig. 2.7 plots the normalized transmission length T_{tot}/F versus *average* memory size M in the asymmetric network with $N = 20$ and $K = 4$ under the decentralized content placement. We let erasure probabilities $\delta_k = \frac{k}{5}$ for $k = 1, \ldots, 4$ and consider files of equal size. We compare "symmetric memory" ($M_k = M, \forall k$), "asymmetric memory" obtained by optimizing over all possible sets of $\{M_k\}$ using our delivery scheme, as well as "lower bound" obtained by optimizing over all possible of $\{M_k\}$ based on (2.14). This result shows the advantage (in terms of delivery time) of optimally allocating cache sizes across users, whenever possible, according to the condition of the delivery channels. We provide in Table 2.1, the optimal memory allocation of the lower bound for different average memory size. We observe that more cache is provided to the user with worse channel quality. As we consider files of equal size, the rate vector $(R, \ldots, R)$ does not belongs to the one sided rate region in Definition 3 for the optimal memory allocation given by Table 2.1. Thus we obtain the gap between our delivery scheme and the lower bound in Fig. 2.7. Note that what we observe in Table 2.1, agrees with [55] which consider a noisy broadcast channel and optimizes the receivers' cache sizes subject to a total cache memory to maximize the rate. It have been shown that more cache is allocated to the weak users and for small cache memories, it is optimal to assign all the cache memory to the weakest receivers, which correspond to $M = 4$ in Table 2.1 where no cache is allocated to users 1 and 2 ($M_1 = M_2 = 0$).

2.9 Conclusions

In this Chapter, we investigate the content delivery problem in the EBC with state feedback, assuming that the content placement phase is performed with existing methods proposed in the literature. Our main contribution is the characterization of the optimal rate region of the channel under these conditions for some special cases, namely for $K \leq 3$, or for the symmetric network with $K \geq 3$, or for the one-sided fair rate vector with $K > 3$. This appears as a non-trivial extension of the work by Wang and Gatzianas et al. [59, 60] which have characterized the capacity region of the EBC with state feedback for some cases of interest. We provide an intuitive interpretation of the algorithm proposed in these works and revealed an explicit connection between the capacity in the symmetric EBC

and the DoF in the MISO-BC. More specifically, we showed that there exists a duality in terms of the order-j multicast capacity/DoF. Such a connection was fully exploited to generalize our results to the cache-enabled MISO-BC. This Chapter demonstrates the benefits of coded caching combined with state feedback in the presence of random erasure. Furthermore, if a memory allocation is allowed, numerical examples show that an optimal memory allocation is to provide more cache to the weakest users as confirmed in [55]. However, for such cache allocation, the symmetric rate vector does not belongs to the one-sided fair region and so our linear encoding scheme is not optimal. A non linear technique of joint cache-channel coding was provided in [71] over erasure broadcast channels, where only the weak users are equipped with cache memory. We believe that such encoding technique can improve the achievability region.

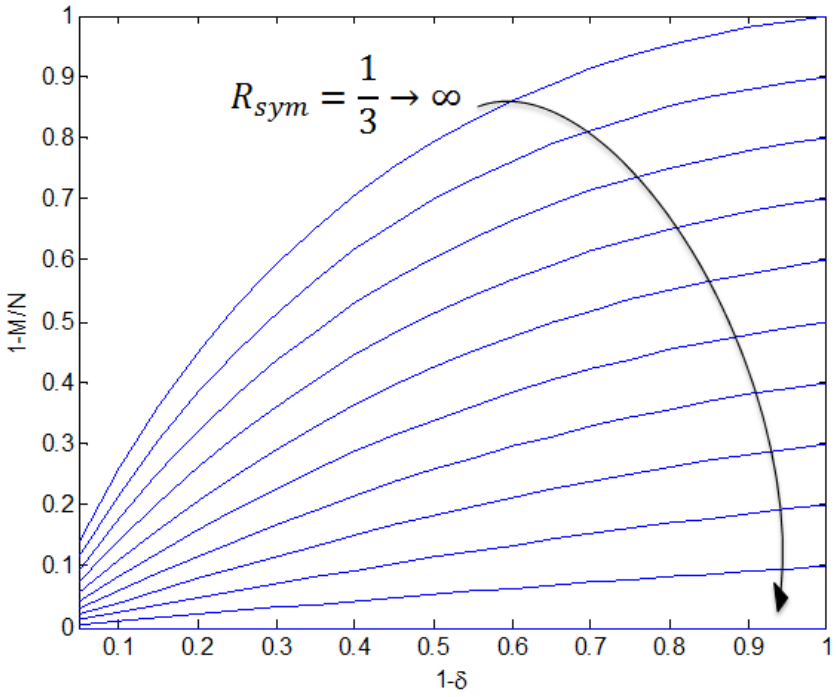


Figure 2.4: The tradeoff between the memory and the erasure for $K = 3$.

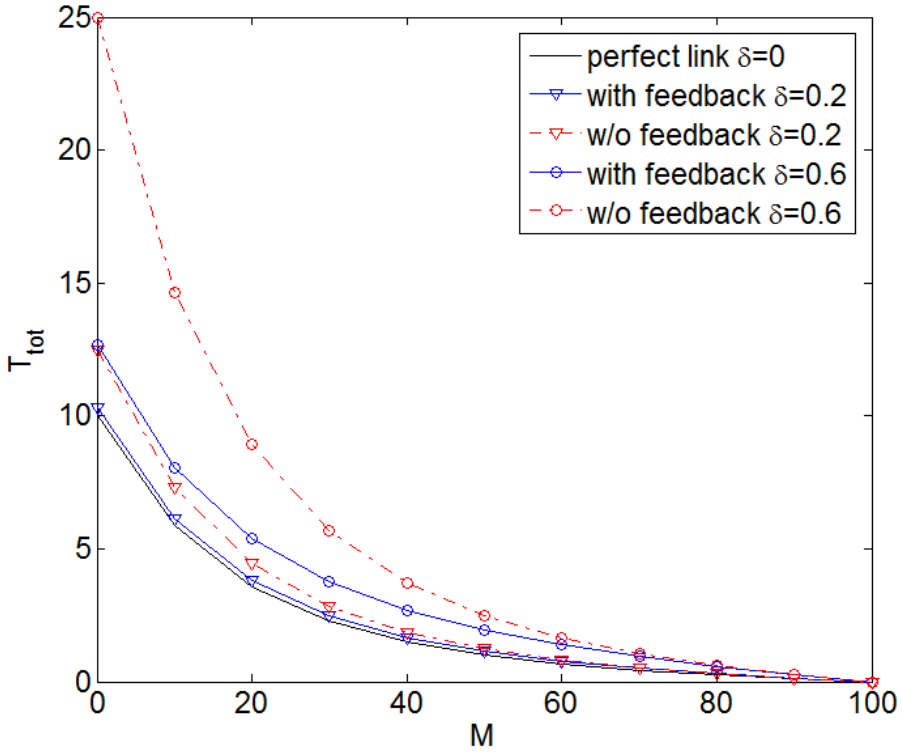


Figure 2.5: The transmission length T_{tot} as a function of memory size M for $N = 100, K = 10$.

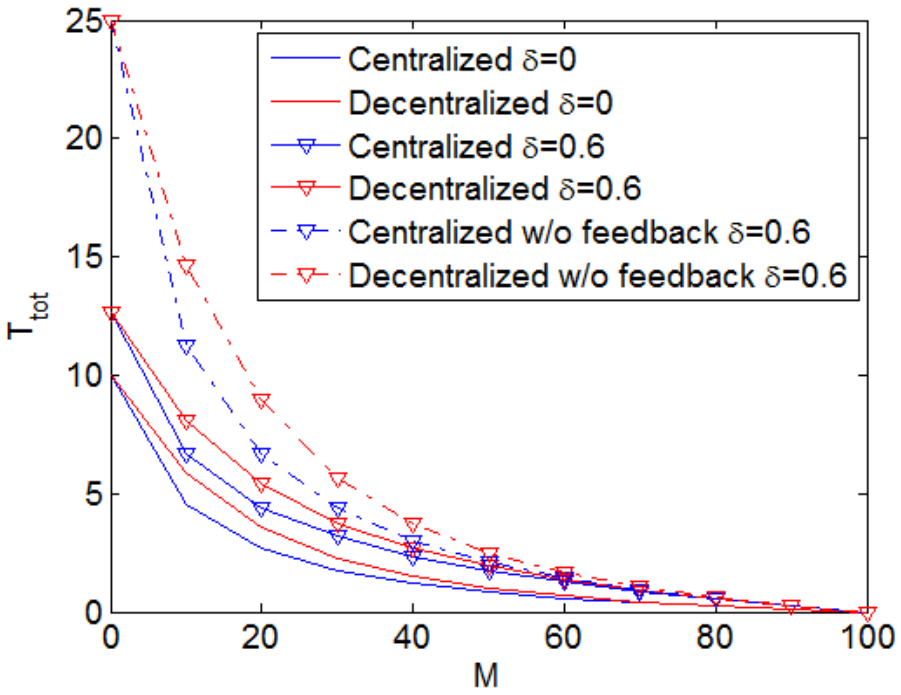


Figure 2.6: The transmission length T_{tot} as a function of memory size M for $N = 100$, $K = 10$.

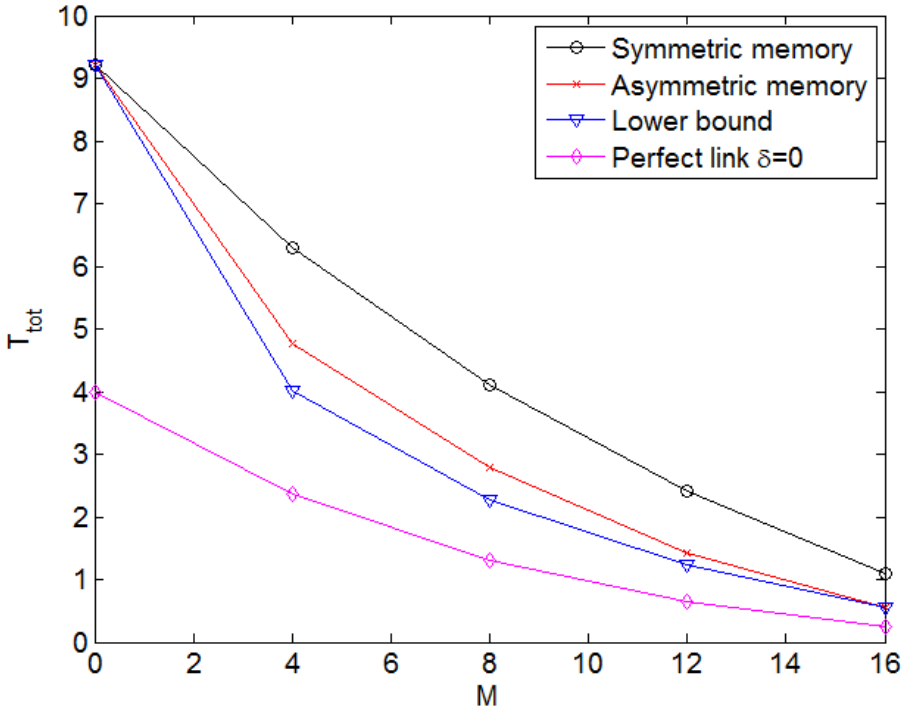


Figure 2.7: T_{tot} vs M for $\delta_i = \frac{i}{5}$, $N = 20$, $K = 4$ and $F_i = 1$.

Chapter 3

Fading Broadcast Channels with Dynamic User Requests

We study the content delivery over asymmetric block-fading broadcast channels, where the channel quality varies across users and time. Assuming that user requests arrive dynamically, we design an online scheme based on queuing structure to deal jointly with admission control, files combinations, as well as scheduling. In the short-term, we allow transmissions to subsets of users with good channel quality, avoiding users with fades, while in the long-term we ensure fairness among users. We prove that our online delivery scheme maximizes the alpha-fair utility among all schemes restricted to decentralized cache placement. The performance analysis built on the Lyapunov theory.

3.1 Introduction

As we mentioned before, in more realistic scenario, the performance of coded caching is limited by the user in the worst channel condition. In particular, for the case of the i.i.d. quasi-static Rayleigh fading channel, the works [45, 47] showed that the long-term sum content delivery rate does not grow with the system dimension if coded caching is naively applied to this channel. In fact, the long-term average multicast rate of the i.i.d. Rayleigh fading channel vanishes, as it scales as $\mathcal{O}(\frac{1}{K})$ as $K \to \infty$, [72].

In the literature of wireless scheduling without caches at receivers, standard downlink techniques are used to prevent such limitation such as *opportunistic scheduling* [61,63,73], which serve the user with the best instantaneous channel quality and fairness among user throughputs [61], which allows users with weak channel quality to receive less throughput.

Since serving the best user and equally satisfying all the users are typically the two extreme objectives, past works have proposed the use of alpha-fairness [62] which allows to select the coefficient α and drive the system to any desirable tradeoff point in between of the two extremes. Previously, the alpha-fair objectives have been studied in the context of (i) multiple user activations [63], (ii) multiple antennas [74] and (iii) broadcast channels [75]. However, in the presence of caches at user terminals, the fairness problem is further complicated by the interplay between scheduling and coding operations. For the case of asychnoneous demand, online transmission scheduling over wireless channels has been extensively studied in the context of opportunistic scheduling [63] and network utility maximization [65].

We study in this chapter, the content delivery over a realistic block-fading broadcast channel, where the channel quality varies across users and time. Furthermore, we deal with asynchronous demands where the time-varying user requests and corresponding delivery can be modeled as a random arrival and departure process, respectively. The new element in our study is the joint consideration of user scheduling with codeword construction for the coded caching delivery phase. More specifically, our approaches and contributions are summarized below:

- We design a novel queueing structure which decouples the channel scheduling from the codeword construction. Although it is clear that the codeword construction needs to be adaptive to channel variation, our scheme ensures this through our *backpressure* that connects the user queues and the codeword queues. Hence, we are able to show that this decomposition is without loss of optimality.

- We then provide an online policy consisting of (i) admission control of new files into the system; (ii) combination of files to perform coded caching; (iii) scheduling and power control of codeword transmissions to subset of users on the wireless channel. We prove that the long-term video delivery rate vector achieved by our scheme is a near optimal solution to the alpha-fair optimization problem under the restriction to policies that are based on the decentralized coded caching scheme [2].

- Through numerical examples, we demonstrate the superiority of our approach versus

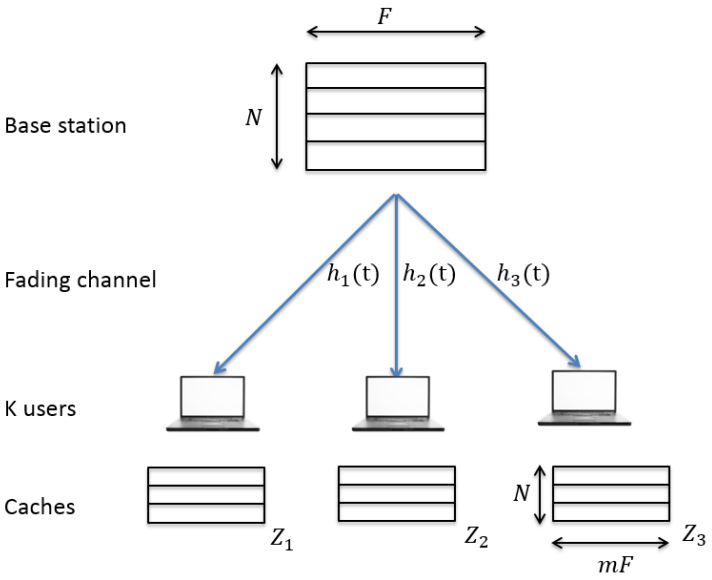


Figure 3.1: System model with $K = 3$.

(a) standard coded caching with multicast transmission limited by the worst channel condition yet exploiting the global caching gain, (b) opportunistic scheduling with unicast transmissions exploiting only the local caching gain (c) superposition and selection schemes exploiting both coded caching gain and channel fading peaks as detailed in Chapter 4. This shows that our proposed scheme is the best among online decentralized coded caching schemes.

3.2 System Model and Motivation

We present the system model considered in this chapter and provide a long-term analysis on the performance of Maddah Ali and Niesen coded caching, recalled in Section 1.2, in fading broadcast channel.

3.2.1 System model

We consider a content delivery system where a server with N files wishes to convey the requested files to K users over a wireless downlink channel. We assume that N files are of equal size of F bits and have equal popularity, while each user k has a cache memory Z_k of size MF bits, where $M \geq 1$ denotes the cache size measured in files. We often use the normalized cache size denoted by $m = M/N$. We restrict ourselves to decentralized cache placement [2]. As depicted in Fig. 3.1, we relax the perfect shared link assumption and consider a wireless channel modeled by a standard block-fading broadcast channel, such that the channel state remains constant over a slot and changes from one slot to another in an i.i.d. manner. Each slot is assumed to allow for n channel uses. The channel output

of user k in any channel use of slot t is given by

$$\mathbf{y}_k(t) = \sqrt{h_k(t)}\mathbf{x}(t) + \mathbf{z}_k(t), \tag{3.1}$$

where the channel input $\mathbf{x} \in \mathbb{C}^n$ is subject to the power constraint $\mathbb{E}[\|\mathbf{x}\|^2] \leq Pn$; $\mathbf{z}_k(t) \sim \mathcal{N}_{\mathbb{C}}(0, \mathbf{I}_n)$ are additive white Gaussian noises with covariance matrix identity of size n, assumed independent of each other; $\{h_k(t) \in \mathbb{C}\}$ are channel fading coefficients independently distributed across users. In addition, we are particularly interested in the long-term behavior (e.g., time span of hours or days) of the system. To simplify such analysis, we further assume that the channel coefficient $h_k(t)$ is i.i.d. over t for a given k. At each slot t, the channel state $\mathbf{h}(t) = (h_1(t), \ldots, h_K(t))$ is perfectly known to the base station and to all users.

Encoding At each time slot, the base station observes the channel state $\mathbf{h}(t)$ and the request vector up to t, $\mathbf{d}^t$, and constructs a transmit symbol using the *encoding function* $f_t : \mathbb{F}_2^{MFK} \times \{1, .., N\}^{Kt} \times \mathbb{C}^K \to \mathbb{C}^n$.

$$\mathbf{x}(t) = f_t\left(Z^K, \mathbf{d}^t, \mathbf{h}(t)\right), \tag{3.2}$$

Decoding At the end of the whole transmission as $t \to \infty$, each receiver decodes its sequence of requested files by applying a decoding function ξ_k to the sequence of the received signals $\mathbf{y}_k^t = (\mathbf{y}_k(1), \ldots, \mathbf{y}_k(t))$, that of the channel state $\mathbf{h}^t = (\mathbf{h}(1), \ldots, \mathbf{h}(t))$, its cache Z_k. Namely, the output of the k-th user's *decoding function* at slot t, $\xi_k(t) : \mathbb{F}_2^{MFK} \times \mathbb{C}^{nt} \times \mathbb{C}^{tK} \to \mathbb{F}_2^{F\hat{D}_k(t)}$, is given by

$$\xi_k(t) = \xi_k(Z_k, \mathbf{y}_k^t, \mathbf{h}^t) \in \mathbb{F}_2^{F\hat{D}_k(t)} \tag{3.3}$$

where $\hat{D}_k(t)$ denotes the number of decoded files by user k up to slot t.

3.2.2 Standard coded caching and motivation

A naive application of coded caching consists on performing coded caching content delivery on all users in the system and sending $T(m, K)$ files to satisfy all K demands. It is well-known that the multicast capacity of the channel at hand, or the common message rate, is given by

$$r_{\text{mc}}(\mathbf{h}) = \log\left(1 + P \min_{j \in [K]} h_j\right) \tag{3.4}$$

measured in [bits/channel use], and limited by the user in the worst channel condition. It has been proved in [45] that such limitation is detrimental for a scalable content delivery network. To see this, let us first define the sum content delivery rate when coded caching is applied directly to the fading broadcast channel. In order to satisfy the distinct demands from K users, the server sends $T(m, K)$ files over the wireless link at rate $r_{\text{mc}}(\mathbf{h})$. Therefore, to deliver KF bits, it takes $\frac{T(m,K)F}{r_{\text{mc}}(\mathbf{h})}$ channel use. As a result, the sum content

delivery rate of a naive application of coded caching for a given channel realization $\mathbf{h}$ is given by

$$\frac{K}{T(m,K)} r_{\mathrm{mc}}(\mathbf{h}) \tag{3.5}$$

measured in [bits/channel use] ($\times \frac{n}{F}$ in [files/slot]). For convenience, we call such a naive application as the "baseline" ("bl") scheme where the base station serves all K users with the multicast rate limited by the worst user as in (3.4). The corresponding (long-term) average sum content delivery rate is given by

$$\bar{r}_{\mathrm{bl,sum}}(K) = \frac{K}{T(m,K)} \mathbb{E}[r_{\mathrm{mc}}(\mathbf{h})]. \tag{3.6}$$

To gain an insight into the harmful effect, let us consider the case of symmetric fading statistics where the fading gains are exponentially distributed with mean 1. The average multicast capacity $\mathbb{E}[r_{\mathrm{mc}}(\mathbf{h})]$ vanishes as $\mathcal{O}(1/K)$ for $K \to \infty$ [72], the average sum content delivery rate converges to a constant, yielding a non-scalable system. More precisely, the performance analysis of this scheme is given below.

Proposition 1. *(i)* $\bar{r}_{\mathrm{bl,sum}}(K,P) = \frac{K}{T(m,K)} e^{\frac{K}{P}} E_1\left(\frac{K}{P}\right)$.

(ii) For all P*:* $\bar{r}_{\mathrm{bl,sum}}(K,P) \sim \frac{Pm}{1-m}$ *when* $K \to \infty$.

(iii) For all K*:* $\bar{r}_{\mathrm{bl,sum}}(K,P) \sim \frac{K}{T(m,K)} \log(P)$ *when* $P \to \infty$.

where we define the exponential integral function $E_1(x) = \int_1^{+\infty} \frac{e^{-xt}}{t} dt$.

Proof. Refer to Appendix B.1. □

This negative result motivates us to study some opportunistic scheduling strategy which benefits both from the coded caching gain and the diversity of the underlying wireless channel, while ensuring certain fairness among users.

3.3 Objectives

This section formulates the problem of alpha-fair file delivery. The performance metric is the *long-term average delivery rate of files* to user k, denoted by $\overline{r}_k$. Hence our objective is expressed with respect to the vector of delivery rates $\overline{\mathbf{r}}$. We define the feasible rate region as the set of the average number of successfully delivered files for K users. We let Λ denote the set of all feasible delivery rate vectors.

Definition 4 (Feasible rate)**.** *A rate vector* $\overline{\mathbf{r}} = (\overline{r}_1, \ldots, \overline{r}_K)$*, measured in file/slot, is said to be feasible* $\overline{\mathbf{r}} \in \Lambda$ *if there exist a file combining and transmission scheme such that*

$$\overline{r}_k = \limsup_{t \to \infty} \frac{D_k(t)}{t}. \tag{3.7}$$

where $D_k(t)$ *denotes the number of successfully delivered files to user* k *up to* t.

Under the assumption that n is arbitrarily large and $t \to \infty$, the number of decoded files $\hat{D}_k(t)$ shall coincide with the number of successfully delivered files $D_k(t)$ under the assumptions discussed previously. In contrast to the original framework [1, 2], our rate metric measures the ability of the system to continuously and reliably deliver requested files to the users. Since finding the optimal policy is very complex in general, we restrict our study to a specific class of policies given by the following mild assumptions:

Definition 5 (Admissible class policies Π^{CC})**.** *The admissible policies have the following characteristics:*

1. *The caching placement and delivery follow the decentralized scheme [2].*
2. *The users request distinct files, i.e., the ids of the requested files of any two users are different.*

Since we restrict our action space, the feasibility rate region, denoted by Λ^{CC}, under the class of policies Π^{CC} is smaller than the one for the original problem Λ. However, the joint design of caching and online delivery appears to be a very hard problem; note that the design of an optimal code for coded caching alone is an open problem and the proposed solutions are constant factor approximations. Restricting the caching strategy to the decentralized scheme proposed in [2] makes the problem amenable to analysis and extraction of conclusions for general cases such as the general setup where users may not have the symmetrical rates. Additionally, if two users request the same file simultaneously, it is efficient to handle exceptionally the transmissions as native broadcasting instead of using the decentralized coded caching scheme, yielding a small efficiency benefit but complicating further the problem. Note, however, the probability that two users simultaneously requesting the same parts of video is very low in practice, hence to simplify our model we exclude this consideration altogether.

Our objective is to solve the *fair file delivery* problem:

$$\overline{\boldsymbol{r}}^* = \arg \max_{\overline{\boldsymbol{r}} \in \Lambda^{CC}} \sum_{k=1}^{K} g_k(\overline{r}_k), \tag{3.8}$$

where the utility function corresponds to the *alpha fair* family of concave functions obtained by choosing:

$$g_\alpha(x) = \begin{cases} \frac{x^{1-\alpha}-1}{1-\alpha}, & \text{if } \alpha \neq 1, \\ \log(x), & \text{if } \alpha = 1. \end{cases} \tag{3.9}$$

Tuning the value of α changes the shape of the utility function and consequently drives the system performance $\overline{\boldsymbol{r}}^*$ to different operating points: (i) $\alpha = 0$ yields max sum delivery rate, (ii) $\alpha \to \infty$ yields max-min delivery rate [62], (iii) $\alpha = 1$ yields

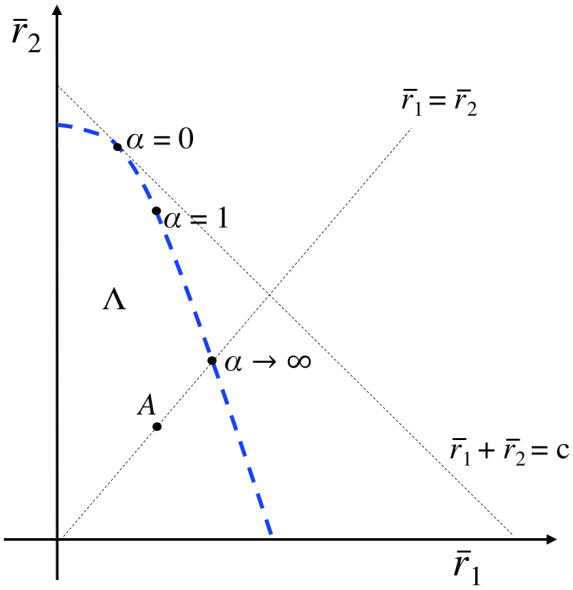


Figure 3.2: Illustration of the feasibility region and different performance operating points for $K = 2$ users. Point A corresponds to a naive adaptation of [2] on our channel model, while the rest points are solutions to our fair delivery problem.

proportionally fair delivery rate [76]. Choosing $\alpha \in (0, 1)$ leads to a tradeoff between max sum and proportionally fair delivery rates.

The optimization (3.8) is designed to allow us tweak the performance of the system; we highlight its importance by an example. Suppose that for a 2-user system Λ is given by the convex set shown on Fig. 3.2. Different boundary points are obtained as solutions to (3.8). If we choose $\alpha = 0$, the system is operated at the point that maximizes the sum $\bar{r}_1 + \bar{r}_2$. The choice $\alpha \to \infty$ leads to the maximum r such that $\bar{r}_1 = \bar{r}_2 = r$, while $\alpha = 1$ maximizes the sum of logarithms. The operation point A is obtained when we always broadcast to all users at the weakest user rate and use [2] for coded caching transmissions. Note that this results in a significant loss of efficiency due to the variations of the fading channel, and consequently A lies in the interior of Λ. To reach the boundary point that corresponds to $\alpha \to \infty$ we need to carefully group users together with good instantaneous channel quality but also serve users with poor average channel quality. This shows the necessity of our approach when using coded caching in realistic wireless channel conditions.

3.4 Proposed Online Delivery Scheme

This section presents the queued delivery network. At each time slot t, the controller *admits* $a_k(t)$ files to be delivered to user k, and hence $a_k(t)$ is a control variable. We equip the base station with the following types of queues:

1. **User queues** to store admitted files, one for each user. The buffer size of queue k

is denoted by $S_k(t)$ and expressed in number of files.

2. **Codeword queues** to store codewords to be multicast. There is one codeword queue for each subset of users $\mathcal{I} \subseteq \{1, \ldots, K\}$. The size of codeword queue $\mathcal{I}$ is denoted by $Q_{\mathcal{I}}(t)$ and expressed in bits.

A queueing policy π performs the following operations: (i) it decides how many files to admit into the user queues $S_k(t)$ in the form of variables $(a_k(t))$, (ii) it combines files destined to user subset $\mathcal{J}$ to create multiple codewords. We let the control variable $\sigma_{\mathcal{J}} \in \mathcal{N}$, denote the number of combinations among files requested from the user subset $\mathcal{J}$ according to the coded caching scheme in [2], (iii) it decides the encoding function for the wireless transmission. Namely, at slot t, it determines the number $\mu_{\mathcal{I}}(t)$ of bits per channel use to be transmitted for the users in subset $\mathcal{I}$.

The user queue S_k evolves as:

$$S_k(t+1) = \big[S_k(t) - \underbrace{\sum_{\mathcal{J}:k\in\mathcal{J}} \sigma_{\mathcal{J}}(t)}_{\substack{\text{number of files}\\ \text{combined into}\\ \text{codewords}}} \big]^+ + \underbrace{a_k(t)}_{\substack{\text{number of}\\ \text{admitted files}}} \tag{3.10}$$

The codeword queue $Q_{\mathcal{I}}$ evolves as

$$Q_{\mathcal{I}}(t+1) = \big[Q_{\mathcal{I}}(t) - \underbrace{n\mu_{\mathcal{I}}(t)}_{\substack{\text{number of bits}\\ \text{multicast to } \mathcal{I}}} \big]^+ + \underbrace{\sum_{\mathcal{J}:\mathcal{I}\subseteq\mathcal{J}} b_{\mathcal{J},\mathcal{I}}\sigma_{\mathcal{J}}(t)}_{\substack{\text{number of bits}\\ \text{created by}\\ \text{combining files}}} \tag{3.11}$$

where $b_{\mathcal{J},\mathcal{I}} = m^{|\mathcal{I}|-1}(1-m)^{|\mathcal{J}|-|\mathcal{I}|+1}$ denotes the number of bits generated for codeword queue $Q_{\mathcal{I}}$, $\mathcal{I} \subseteq \mathcal{J}$, when coded caching is performed to the users in $\mathcal{J}$ (see (1.4)).

3.4.1 Feasibility Region

The main idea here is to characterize the set of feasible file delivery rates via characterizing the stability performance of the queueing system. To this end, let $\overline{a}_k = \limsup_{T\to\infty} \frac{1}{T}\sum_{t=0}^{T-1} \mathbb{E}\left[a_k(t)\right]$, denote the time average number of admitted files for user k.

If the queueing system we have introduced is stable (Definition 1), the rate of admitted files (input rate) is equal to the rate of successfully decoded files (output rate), hence we can characterize the system performance by means of the stability region of our queueing system. We let $\Gamma(\mathbf{h})$ denote the capacity region for a fixed channel state $\mathbf{h}$, as defined in Theorem 14. Then we have the following:

Theorem 12 (Stability region)**.** *Let Γ^{CC} be a set to which a rate vector of admitted files $\overline{\mathbf{a}}$ belongs to, if and only if there exist $\overline{\boldsymbol{\mu}} \in \sum_{\mathbf{h}\in\mathcal{H}} \phi_{\mathbf{h}}\Gamma(\mathbf{h})$, $\overline{\sigma}_{\mathcal{I}} \in [0, \sigma_{\max}], \forall \mathcal{I} \subseteq \{1, \ldots, K\}$*

such that:

$$\sum_{\mathcal{J}:k\in\mathcal{J}} \overline{\sigma}_{\mathcal{J}} \geq \overline{a}_k, \forall k = 1, \ldots, K \tag{3.12}$$

$$n\overline{\mu}_{\mathcal{J}} \geq \sum_{\mathcal{J}:\mathcal{I}\subseteq\mathcal{J}} b_{\mathcal{J},\mathcal{I}}\overline{\sigma}_{\mathcal{J}}, \forall \mathcal{I} \subseteq \{1, 2, ..., K\}. \tag{3.13}$$

Then, the stability region of the system is the interior of Γ^{CC}, where the above inequalities are strict.

Constraint (3.12) says that the aggregate service rate is greater than the arrival rate, while (3.13) implies that the long-term average rate for the subset $\mathcal{J}$ is greater than the arrival rate of the codewords intended to this subset. In terms of the queueing system defined, these constraints impose that the service rates of each queue should be greater than their arrival rates, thus rendering them stable [1]. The proof of this theorem relies on existence of static policies, i.e. randomized policies whose decision distribution depends only on the realization of the channel state. See the Appendix, Section B.4 for a definition and results on these policies.

Since the channel process $\mathbf{h}(t)$ is a sequence of i.i.d. realizations of the channel states (the same results hold if, more generally, $\mathbf{h}(t)$ is an ergodic Markov chain), we can obtain any admitted file rate vector $\overline{\mathbf{a}}$ in the stability region by a Markovian policy, i.e. a policy that chooses $\{\mathbf{a}(t), \boldsymbol{\sigma}(t), \boldsymbol{\mu}(t)\}$ based only the state of the system at the beginning of time slot t, $\{\mathbf{h}(t), \mathbf{S}(t), \mathbf{Q}(t)\}$, and not the time index itself. This implies that $(\mathbf{S}(t), \mathbf{Q}(t))$ evolves as a Markov chain, therefore our stability definition is equivalent to that Markov chain being ergodic with every queue having finite mean under the stationary distribution. Therefore, if we develop a policy that keeps *user queues* $\mathbf{S}(t)$ stable, then all admitted files will, at some point, be combined into codewords. Additionally, if *codeword queues* $\mathbf{Q}(t)$ are stable, then all generated codewords will be successfully conveyed to their destinations. This in turn means that all receivers will be able to decode the admitted files that they requested:

Lemma 13. *The region of all feasible delivery rates Λ^{CC} is the same as the stability region of the system, i.e. $\Lambda^{CC} = Int(\Gamma^{CC})$.*

Proof. Please refer to Appendix B.5. □

Lemma 13 implies the following Corollary.

Corollary 4. *Solving* (3.8) *is equivalent to finding a policy π such that*

$$\overline{\boldsymbol{a}}^{\pi} = arg\max \sum_{k=1}^{K} g_{\alpha}(\overline{a}_k) \tag{3.14}$$

$$s.t. \quad \text{the system is stable.}$$

[1] We restrict vectors $\overline{\mathbf{a}}$ to the interior of Γ^{CC}, since arrival rates at the boundary are exceptional cases of no practical interest, and require special treatment.

This implies that the solution to the original problem (3.8) in terms of the long-term average rates is equivalent to the new problem in terms of the admission rates stabilizing the system.

3.4.2 Admission control and files combination

At the beginning of each slot, the controller decides how many requests $a_k(t)$ for user k, (input of $S_k(t)$), should be pulled into the system from the infinite reservoir. Moreover, it decides on the files to be combined (output of $S_k(t)$) by applying coded caching among specified files.

Input of the user queues $S_k(t)$: Our goal is to find a control policy that optimizes (3.14). To this aim, we need to introduce one more set of queues. These queues are virtual, in the sense that they do not hold actual file demands or bits, but are merely counters to drive the control policy. Each user k is associated with a queue $U_k(t)$ which evolves as follows:

$$U_k(t+1) = [U_k(t) - a_k(t)]^+ + \gamma_k(t) \tag{3.15}$$

where $\gamma_k(t)$ represents the arrival process to the virtual queue and is an additional control parameter. We require these queues to be stable: The actual mean file admission rates are greater than the virtual arrival rates and the control algorithm actually seeks to optimize the time average of the virtual arrivals $\gamma_k(t)$. However, since $U_k(t)$ is stable, its service rate, which is the actual admission rate, will be greater than the rate of the virtual arrivals, therefore giving the same optimizer. Stability of all other queues will guarantee that admitted files will be actually delivered to the users. With these considerations, $U_k(t)$ will be a control indicator such that when $U_k(t)$ is above $S_k(t)$ then we admit files into the system, else we set $a_k(t) = 0$. In particular, we will control the way $U_k(t)$ grows over time using the actual utility objective $g_\alpha(.)$ such that a user with rate x and rapidly increasing utility $g_\alpha(x)$ (steep derivative at x) will also enjoy a rapidly increasing $U_k(t)$ and hence admit more files into the system.

In our proposed policy, the arrival process to the virtual queues are given by

$$\gamma_k(t) = \arg \max_{0 \le x \le \gamma_{k,\max}} [V g_\alpha(x) - U_k(t)x] \tag{3.16}$$

In the above, $V > 0$ is a parameter that controls the utility-delay tradeoff achieved by the algorithm (see Theorem 16).

For every user k, admission control chooses $a_k(t)$ demands given by

$$a_k(t) = \gamma_{k,\max} \mathbf{1}\{U_k(t) \ge S_k(t)\} \tag{3.17}$$

Output of the user queues $S_k(t)$: At each slot, files from subsets of these queues are combined into codewords by means of performing coded caching content delivery scheme [2]. Specifically, the decision at slot t for a subset of users $\mathcal{J} \subseteq \{1, .., K\}$, denoted by

$\sigma_{\mathcal{J}}(t) \in \{0, 1, \ldots, \sigma_{\max}\}$, refers to the number of combined requests for this subset of users. [2] For every subset $\mathcal{J} \subseteq \{1, \ldots, K\}$, the server combines $\sigma_{\mathcal{J}}(t)$ demands of users in $\mathcal{J}$ given by

$$\sigma_{\mathcal{J}}(t) = \sigma_{\max} \mathbf{1} \left\{ \sum_{k \in \mathcal{J}} S_k(t) > \sum_{\mathcal{I}: \mathcal{I} \subseteq \mathcal{J}} \frac{b_{\mathcal{J},\mathcal{I}}}{F^2} Q_{\mathcal{I}}(t) \right\}. \tag{3.18}$$

If $\sigma_{\mathcal{J}}(t) > 0$, the server creates codewords by applying (1.4) for this subset of users as a function of the cache contents $\{Z_j : j \in \mathcal{J}\}$.

3.4.3 Scheduling and transmission

The codewords intended to the user subset $\mathcal{I}$ are stored in codeword queue whose size is given by $Q_{\mathcal{I}}(t)$ for $\mathcal{I} \subseteq \{1, \ldots, K\}$. Given the instantaneous channel realization $\mathbf{h}(t)$ and the queue state $\{Q_{\mathcal{I}}(t)\}$, the server performs multicast scheduling and rate allocation.

In the following section we propose the scheduling and resource allocation solving the following weighted sum rate maximization at each slot t where the weight of the subset $\mathcal{J}$ corresponds to the queue length of $Q_{\mathcal{J}}$

$$\boldsymbol{\mu}(t) = \arg \max_{\mathbf{r} \in \Gamma(\mathbf{h}(t))} \sum_{\mathcal{J} \subseteq \{1, \ldots, K\}} Q_{\mathcal{J}}(t) r_{\mathcal{J}}, \tag{3.19}$$

where $\Gamma(\mathbf{h}(t))$ is the capacity region of a K-user degraded Gaussian broadcast channel with $2^K - 1$ independent messages provided in the following Subsection 3.4.4. Algorithm 2 summarizes our online delivery scheme.

[2] It is worth noticing that standard coded caching lets $\sigma_{\mathcal{J}} = 1$ for $\mathcal{J} = \{1, \ldots, K\}$ and zero for all the other subsets. On the other hand, uncoded caching can be represented by $\sigma_{\mathcal{J}} = 1$ for $\mathcal{J} = k, k \in 1, ..., K$. Our scheme can, therefore be seen as a combination of both, which explains its better performance.

Algorithm 2 Proposed delivery scheme

PLACEMENT (same as [2]):

Fill the cache of each user k

$$Z_k = \{W_{i\,|\,\mathcal{J}} : \ \mathcal{J} \subseteq \{1,\dots,K\}, k \in \mathcal{J}, \forall i = 1,\dots,N\}.$$

DELIVERY:

for $t = 1,\dots,T$

Decide the arrival process to the virtual queues

$$\gamma_k(t) = \arg \max_{0 \le x \le \gamma_{k,\max}} [V g_\alpha(x) - U_k(t)x]$$

Decide the number of admitted files

$$a_k(t) = \gamma_{k,\max} \mathbf{1}\{U_k(t) \ge S_k(t)\} \ .$$

Update the virtual queues

$$U_k(t+1) = [U_k(t) - a_k(t)]^+ + \gamma_k(t)$$

Decide the number of files to be combined

$$\sigma_{\mathcal{J}}(t) = \sigma_{\max} \mathbf{1}\left\{ \sum_{k \in \mathcal{J}} S_k(t) > \sum_{\mathcal{I}:\mathcal{I} \subseteq \mathcal{J}} \frac{b_{\mathcal{J},\mathcal{I}}}{F^2} Q_{\mathcal{I}}(t) \right\}.$$

Scheduling decides the instantaneous rate

$$\boldsymbol{\mu}(t) = \arg \max_{\mathbf{r} \in \Gamma(\mathbf{h}(t))} \textstyle\sum_{\mathcal{J} \subseteq \{1,\dots,K\}} Q_{\mathcal{J}}(t) r_{\mathcal{J}}.$$

Update user queues and codeword queues:

$$S_k(t+1) = \left[S_k(t) - \textstyle\sum_{\mathcal{J}:k \in \mathcal{J}} \sigma_{\mathcal{J}}(t)\right]^+ + a_k(t),$$

$$Q_{\mathcal{I}}(t+1) = [Q_{\mathcal{I}}(t) - n\mu_{\mathcal{I}}(t)]^+ + \sum_{\mathcal{J}:\mathcal{I} \subseteq \mathcal{J}} b_{\mathcal{J},\mathcal{I}} \sigma_{\mathcal{J}}(t).$$

3.4.4 Degraded Broadcast Channel with Private and Common Messages

It readily follows that the channel in (3.1) for a given channel realization $\mathbf{h}$ is a degraded Gaussian broadcast channel. Without loss of generality, we assume $h_1 \ge \cdots \ge h_K$. Let us consider that the transmitter wishes to convey $2^K - 1$ mutually independent messages, denoted by $\{M_{\mathcal{J}}\}$, where $M_{\mathcal{J}}$ denotes the message intended to the users in subset $\mathcal{J} \subseteq [K]$. Each user k must decode all messages $\{M_{\mathcal{J}}\}$ for $\mathcal{J} \ni k$. By letting $R_{\mathcal{J}}$ denote the multicast rate of the message $M_{\mathcal{J}}$, we say that the rate-tuple $\mathbf{R} \in \mathbb{R}_+^{2^K-1}$ is achievable if there exists some encoding and decoding functions such that decoding error probability can be arbitrarily small with large codeword length n. The capacity region is defined as the set of all achievable rate-tuples and is given by the following theorem.

Theorem 14. *The capacity region $\Gamma(\mathbf{h})$ of a K-user degraded Gaussian broadcast channel*

with fading gains $h_1 \geq \cdots \geq h_K$ *and* $2^K - 1$ *independent messages* $\{M_\mathcal{J}\}$ *is given by*

$$R_1 \leq \log(1 + h_1 p_1) \tag{3.20}$$

$$\sum_{\mathcal{J} \subseteq \{1,\ldots,k\}: k \in \mathcal{J}} R_\mathcal{J} \leq \log \frac{1 + h_k \sum_{j=1}^{k} p_j}{1 + h_k \sum_{j=1}^{k-1} p_j} \quad k = 2, \ldots, K \tag{3.21}$$

for non-negative variables $\{p_k\}$ *such that* $\sum_{k=1}^{K} p_k \leq P$.

Proof. The proof is quite straightforward and is based on rate-splitting and the private-message region of degraded broadcast channel. For completeness, see details in Appendix B.2. □

In order to characterize the boundary of the capacity region $\Gamma(\mathbf{h})$, we consider the weighted sum rate maximization given as

$$\max_{\mathbf{R} \in \Gamma(\mathbf{h})} \sum_{\mathcal{J}: \mathcal{J} \subseteq [K]} \theta_\mathcal{J} R_\mathcal{J}. \tag{3.22}$$

By exploiting a simple property of the capacity region, the problem at hand can be cast into a simpler problem as summarized below.

Theorem 15. *The weighted sum rate maximization with* $2^K - 1$ *variables in* (3.22) *reduces to a simpler problem with* K *variables given by*

$$\max_{\mathbf{p}} \sum_{k=1}^{K} \tilde{\theta}_k \log \frac{1 + h_k \sum_{j=1}^{k} p_j}{1 + h_k \sum_{j=1}^{k-1} p_j}. \tag{3.23}$$

where $\mathbf{p} = (p_1, \ldots, p_K) \in \mathbb{R}_+^K$ *is a positive real vector satisfying the total power constraint, and* $\tilde{\theta}_k$ *denotes the largest weight for user* k

$$\tilde{\theta}_k = \max_{\mathcal{J}: k \in \mathcal{J} \subseteq \{1,\ldots,k\}} \theta_\mathcal{J}.$$

Proof. Refer to Appendix B.3. □

Note that (3.23) is similar to (3.19) by taking $\theta_\mathcal{J} = Q_\mathcal{J}(t)$.

We provide an efficient algorithm to solve this power allocation problem in Subsection 1.4.2.

3.4.5 Example

We conclude this section by providing an example of our proposed online delivery scheme for $K = 3$ users as illustrated in Fig. 3.3.

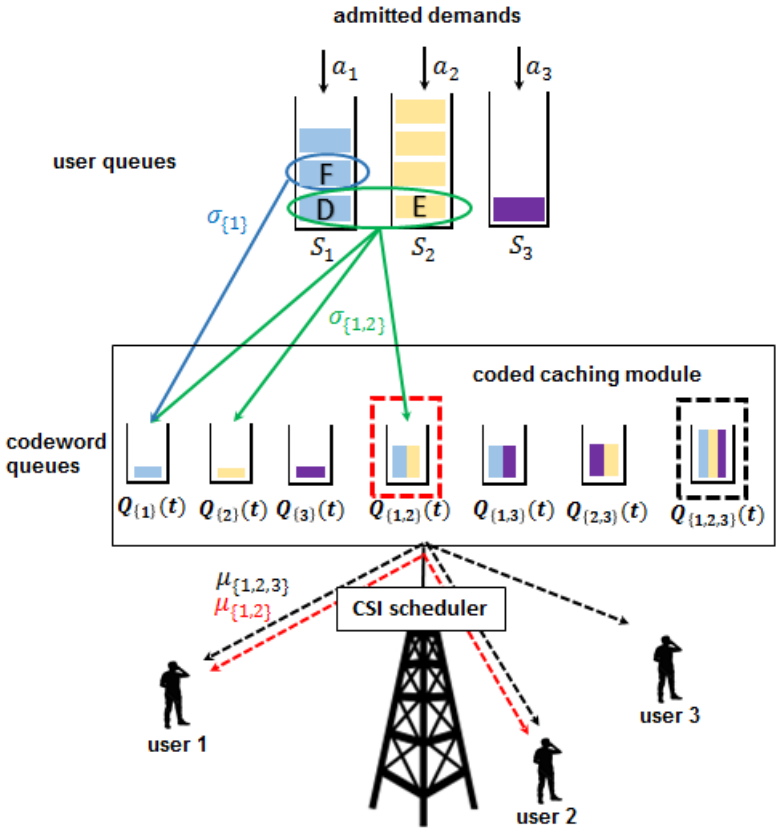


Figure 3.3: An example of the queueing model for a system with 3 users. Dashed lines represent wireless transmissions, solid circles files to be combined and solid arrows codewords generated.

We focus on the evolution of codeword queues between two slots, t and $t+1$. The exact backlog of codeword queues is shown in Table 3.1. Given the routing and scheduling decisions ($\sigma_{\mathcal{J}}(t)$ and $\mu_{\mathcal{J}}(t)$), we provide the new states of the queues at the next slot in the same Table.

We consider the set of files $\{A, B, C, D, E, F\}$ where user 1 requests $\{A, D, F\}$, user 2 requests $\{B, E\}$ and user 3 requests $\{C, F\}$. We suppose that files $\{A, B, C\}$ were linearly combined in previous slot(s) using (1.4) and the created codewords are stored in the corresponding codeword queues at slot t. We let $A_{\mathcal{J}}$ denote the sub-file A exclusively cached at users in $\mathcal{J} \subseteq [K]$. The same notation is used for the other files. We suppose that $h_1(t) > h_2(t) > h_3(t)$. The scheduler uses (3.19) to allocate positive rates to user set $\{1, 2\}$ and $\{1, 2, 3\}$ given by $\mu_{\{1,2\}}$ and $\mu_{\{1,2,3\}}$, and multicasts the superposed signal $x(t) = A_2 \oplus B_1 + A_{23} \oplus B_{13} \oplus C_{12}$. User 3 decodes only $A_{23} \oplus B_{13} \oplus C_{12}$. User 1 and 2 decode first $A_{23} \oplus B_{13} \oplus C_{12}$, then subtracts it and decode $A_2 \oplus B_1$. In the next slot, the received sub-files are evacuated from the codeword queues as shown in Table 3.1. Using successive interference cancellation (SIC) and XOR operation: user 1 decodes (A_2, A_{23}), user 2 decodes (B_1, B_{13}) and user 3 decodes C_{12}.

For the routing decision, the server decides at slot t to combine D requested by user 1 with E requested by user 2 and to process F requested by user 1 uncoded. Therefore, we

Table 3.1: Codeword queues evolution for $\mu_{\{1,2\}}(t),\ \mu_{\{1,2,3\}}(t) > 0$ and $\sigma_{\{1,2\}}(t) = \sigma_{\{1\}}(t) = 1$.

	$\mathcal{Q}_{\{1\}}$	$\mathcal{Q}_{\{2\}}$	$\mathcal{Q}_{\{3\}}$	$\mathcal{Q}_{\{1,2\}}$	$\mathcal{Q}_{\{1,3\}}$	$\mathcal{Q}_{\{2,3\}}$	$\mathcal{Q}_{\{1,2,3\}}$
$\mathcal{Q}_{\mathcal{J}}(t)$	$A_\emptyset$	$B_\emptyset$	$C_\emptyset$	$A_2 \oplus B_1$	$A_3 \oplus C_1$	$B_3 \oplus C_2$	$A_{23} \oplus B_{13} \oplus C_{12}$
Output $\mu_{\{1,2\}}(t) > 0$ $\mu_{\{1,2,3\}}(t) > 0$	-	-	-	$A_2 \oplus B_1$	-	-	$A_{23} \oplus B_{13} \oplus C_{12}$
Input $\sigma_{\{1,2\}}(t) = 1$ $\sigma_{\{1\}}(t) = 1$	$D_\emptyset;\ D_3$ $\{F_{\mathcal{J}}\}_{1 \notin \mathcal{J}}$	$E_\emptyset;\ E_3$	-	$E_1 \oplus D_2$ $E_{13} \oplus D_{23}$	-	-	-
$\mathcal{Q}_{\mathcal{J}}(t+1)$	$A_\emptyset$ $D_\emptyset;\ D_3$ $\{F_{\mathcal{J}}\}_{1 \notin \mathcal{J}}$	$B_\emptyset$ $E_\emptyset;\ E_3$	$C_\emptyset$	$E_1 \oplus D_2$ $E_{13} \oplus D_{23}$	$A_3 \oplus C_1$	$B_3 \oplus C_2$	-

have $\sigma_{\{1,2\}}(t) = \sigma_{\{1\}}(t) = 1$ and $\sigma_{\mathcal{J}}(t) = 0$ otherwise. Given this codeword construction, codeword queues have inputs that change its state in the next slot as described in Table 3.1.

3.5 Performance Analysis

In this section, we present the main result by proving that our proposed online algorithm achieves near-optimal performance for all policies within the class Π^{CC}:

Theorem 16. *Let $\bar{r}_k^\pi$ the long-term average delivery rate for user k achieved by the proposed policy. Then*

$$\sum_{k=1}^{K} g_\alpha(\bar{r}_k^\pi) \geq \max_{\bar{\mathbf{r}} \in \Lambda^{CC}} \sum_{k=1}^{K} g_k(\bar{r}_k) - \frac{B}{V}$$

$$\limsup_{T \to \infty} \frac{1}{T} \sum_{t=0}^{T-1} \mathbb{E}\left\{\hat{Q}(t)\right\} = \frac{B + V \sum_{k=1}^{K} g_\alpha(\gamma_{max,k})}{\epsilon_0},$$

where $\hat{Q}(t)$ is the sum of all queue lengths at the beginning of time slot t, thus a measure of the mean delay of file delivery. The quantities B and ϵ_0 are constants that depend on the statistics of the system.

Proof. Please refer to Appendix B.6. □

The above theorem states that, by tuning the constant V, the utility resulting from our online policy can be arbitrarily close to the optimal one, where there is a tradeoff between the guaranteed optimality gap $\mathcal{O}(1/V)$ and the upper bound on the total buffer length $\mathcal{O}(V)$.

3.6 Dynamic File Requests

In this Section, we extend our algorithm to the case where there is no infinite amount of demands for each user, rather each user requests a finite number of files at slot t. Let $A_k(t)$ be the number of files requested by user k at the beginning of slot t. We assume it is an i.i.d. random process with mean λ_k and such that $A_k(t) \leq A_{\max}$ almost surely. [3] In this case, the alpha fair delivery problem is to find a delivery rate $\overline{\mathbf{r}}$ that solves

$$\begin{aligned} \text{Maximize } & \sum_{k=1}^{K} g_\alpha(\overline{r}_k) \\ \text{s.t. } & \overline{\mathbf{r}} \in \Lambda^{CC} \\ & \overline{r}_k \leq \lambda_k, \forall k \in \{1, ..., K\}, \end{aligned}$$

where the additional constraints $\overline{r}_k \leq \lambda_k$ denote that a user cannot receive more files than the ones actually requested.

The fact that file demands are not infinite and come as a stochastic process is dealt with by introducing one "reservoir queue" per user, $L_k(t)$, which stores the file demands that have not been admitted, and an additional control decision on how many demands to reject permanently from the system, $d_k(t)$. At slot t, no more demands then the ones that arrived at the beginning of this slot and the ones waiting in the reservoir queues can be admitted, therefore the admission control must have the additional constraint

$$a_k(t) \leq A_k(t) + L_k(t), \forall k, t,$$

and a similar restriction holds for the number of rejected files from the system, $d_k(t)$. The reservoir queues then evolve as

$$L_k(t+1) = L_k(t) + A_k(t) - a_k(t) - d_k(t).$$

The above modification with the reservoir queues has only an impact that further constrains the admission control of files to the system. The queuing system remains the same as described in Section 3.4, with the user queues $\mathbf{S}(t)$, the codeword queues $\mathbf{Q}(t)$ and the virtual queues $\mathbf{U}(t)$. Similar to the case with infinite demands we can restrict ourselves to policies that are functions only of the system state at time slot t, $\{\mathbf{S}(t), \mathbf{Q}(t), \mathbf{L}(t), \mathbf{A}(t), \mathbf{h}(t), \mathbf{U}(t)\}$ without loss of optimality. Furthermore, we can show that the alpha fair optimization problem equivalent to the problem of controlling the admission rate. That is, we want to find a policy π such that

$$\begin{aligned} \overline{\mathbf{a}}^\pi = \arg\max & \sum_{k=1}^{K} g_\alpha(\overline{a}_k) \\ \text{s.t. } & \text{the queues } (\mathbf{S}(t), \mathbf{Q}(t), \mathbf{U}(t)) \text{ are strongly stable} \\ & a_k(t) \leq \min[a_{max,k}, L_k(t) + A_k(t)], \forall t \geq 0, \forall k \end{aligned}$$

[3] The assumptions can be relaxed to arrivals being ergodic Markov chains with finite second moment under the stationary distribution

The rules for scheduling, codeword generation, virtual queue arrivals and queue updating remain the same as in the case of infinite demands in Section 3.4. The only difference is that there are multiple possibilities for the admission control; see [61] and Chapter 5 of [66] for more details. Here we propose that at each slot t, any demand that is not admitted get rejected (i.e. the reservoir queues hold no demands), the admission rule is

$$a_k^{\pi}(t) = A_k(t)\mathbf{1}\{U_k(t) \geq S_k(t)\}, \tag{3.24}$$

and the constants are set as $\gamma_{k,\max}, \sigma_{\max} \geq A_{\max}$. Using the same ideas employed in the performance analysis of the case with infinite demands and the ones employed in [61], we can prove that the $O(1/V) - O(V)$ utility-queue length tradeoff of Theorem 16 holds for the case of dynamic arrivals as well.

3.7 Numerical Examples

In this section, we compare our online proposed delivery scheme Section 3.4 with the following other schemes, all building on decentralized cache placement in (1.1).

- **Unicast opportunistic scheduling**: for any request, the server sends the remaining $(1-m)F$ bits to the corresponding user without combining any files (we only exploit the local caching gain). In slot t the serve sends with full power to user

$$k^*(t) = \arg\max_k \frac{\log\left(1 + h_k(t)P\right)}{u_k(t)^{\alpha}},$$

  where $\mathbf{u}(t) = (u_1(t), ..., u_K(t))$ is the vector of empirical data rates up to time t, and obeys the recursive equation $u_i(t+1) = \frac{1}{t+1}\left[tu_i(t) + r_i^*(t)\right]$.

The following two schemes are more detailed in Chapter 4.

- **Superposition**: At each slot t, this scheme solves the weighted sum rate maximization problem in $\Gamma(\mathbf{h}(t)) \subseteq \mathbb{R}_+^{2^K-1}$, using Theorem 14:

$$\mathbf{R}_{\text{sp}}(\mathbf{h}(t), t) = \arg\max_{\mathbf{R}\in\Gamma(\mathbf{h}(t))} \sum_{\mathcal{J}:\mathcal{J}\subseteq[K]} \theta_{\mathcal{J}}(t) R_{\mathcal{J}} \text{ with } \theta_{\mathcal{J}}(t) = \frac{\sum_{i\in\mathcal{J}} \frac{1}{u_i^{\alpha}(t)}}{T(m, |\mathcal{J}|)},$$

  The average rate of user i is

$$\bar{r}_{\text{sp},i} = \lim_{t\to\infty} \mathbb{E}\left[\sum_{\mathcal{J}:i\in\mathcal{J}} \frac{1}{T(m,|\mathcal{J}|)} R_{\text{sp},\mathcal{J}}(\mathbf{h}(t), t)\right].$$

- **Selection with full CSIT**: At each slot t, this scheme selects the subset of users

$$\mathcal{J}_{\text{sc}}(\mathbf{h}(t), t) = \underset{\mathcal{J}\subseteq[K]}{\text{argmax}} \left\{ \frac{1}{T(m,|\mathcal{J}|)} \log(1 + \min_{j\in\mathcal{J}} h_j(t)) \sum_{i=1}^{K} \frac{\mathbf{1}\{i\in\mathcal{J}\}}{u_i(t)^{\alpha}} \right\}.$$

The average rate of user i is:

$$\bar{r}_{\mathrm{sc},i} = \lim_{t\to\infty} \mathbb{E}\left[\frac{1}{T(m, |\mathcal{J}_{\mathrm{sc}}(\mathbf{h}(t), t)|)} \log(1 + \min_{j\in\mathcal{J}_{\mathrm{sc}}(\mathbf{h}(t),t)} h_j(t))\mathbf{1}\{i \in \mathcal{J}_{\mathrm{sc}}(\mathbf{h}(t), t)\}\right].$$

We consider the system with normalized memory of $m = 0.6$, power constraint $P = 10dB$, file size $F = 10^3$ bits and number of channel uses per slot $n = 10^2$, channel coefficients exponentially distributed with mean ρ_k. We divide users into two classes of $K/2$ users each: strong users with $\rho_k = 1$ and weak users with $\rho_k = 0.2$. We compare all the algorithms for the cases where the objective of the system is sum rate maximization ($\alpha = 0$) and proportional fairness ($\alpha = 1$). The results are depicted in Fig. 3.4 and 3.5, respectively.

Regarding the sum rate objective and the proportional fair objective, baseline scheme performs very poorly, indicative of the adverse effect of users with bad channel quality. It is notable that our proposed scheme outperforms the unicast opportunistic scheme, which maximizes the utility if only private information packets are to be conveyed. Although the selection scheme and threshold-based scheme can simultaneously exploit the coded caching gain and use the channel opportunistically, our proposed scheme provides better performance since it is declared near-optimal. The relative merit of our scheme increases as the number of users grows. This can be attributed to the fact that our scheme can exploit any available multicast opportunities. *Our result here implies that, in realistic wireless systems, coded caching can indeed provide a significant throughput increase when an appropriate joint design of routing and opportunistic transmission is used.*

3.8 Conclusions

In this chapter, we studied coded caching over wireless fading channels in order to address its limitation governed by the user with the worst fading state. We designed an online scheme that deals with the asynchronous nature of the user demands and benefits both from the coded caching gain and the diversity of the underlying wireless channel. By formulating an alpha-fair optimization problem with respect to the long-term average delivery rates and using queueing structure, our proposed scheme allowed us to obtain an optimal algorithm for joint file admission control, codeword construction and wireless transmissions. The main conclusion is that, by appropriately combining the multicast opportunities and the opportunism due to channel fading, coded caching can lead to significant gains in wireless systems with fading. Furthermore, the queueing structure makes our scheme flexible to be adapted to different setting such us asymmetric finite file size and/or distinct memory capacity. In fact, the admitted files to the system can be partitioned into sub-files and stored in virtual queues depending on the users caches before deciding on the files combination. However the queueing structure makes our optimal solution complex for implementation. Thus, in the next chapter, we address the same limitation of coded caching in wireless channels by focusing on the scheduling

part. We propose a simple scheme of low-complexity, that achieves a scalable sum content delivery in fading broadcast channels.

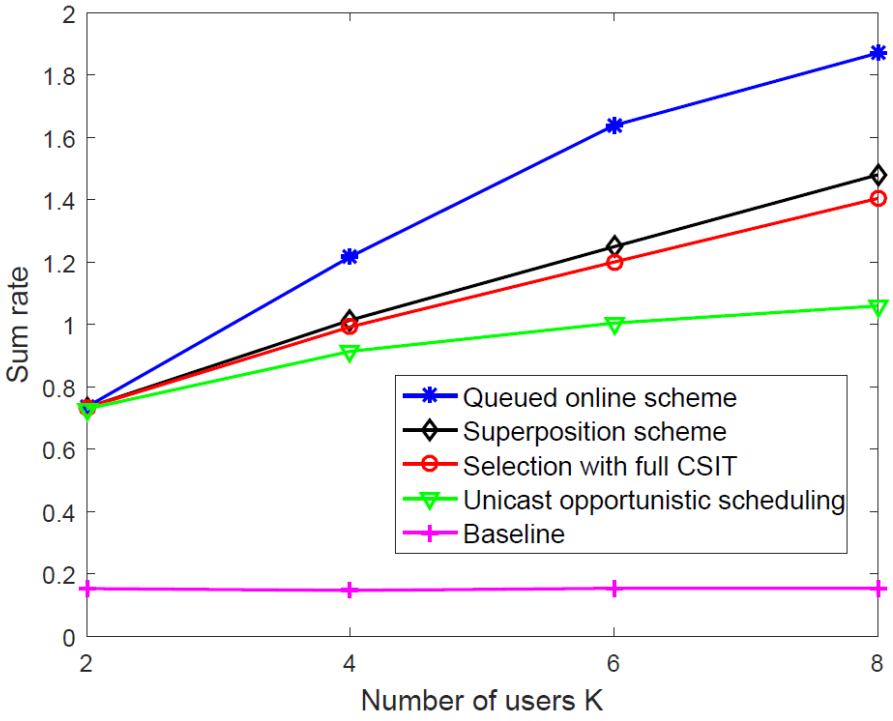


Figure 3.4: Sum rate ($\alpha = 0$)

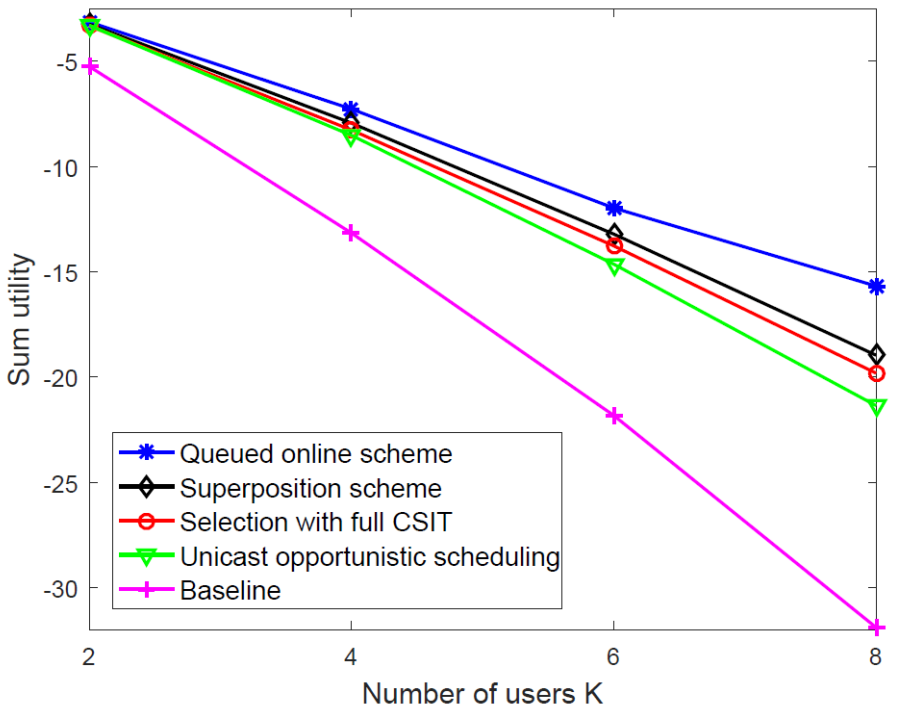


Figure 3.5: Proportional fair utility ($\alpha = 1$)

Chapter 4

Opportunistic Scheduling

We consider the same system model of Chapter 3, where the users, equipped with a memory of finite size, experience asymmetric fading statistics. We showed that a naive application of coded caching over the channel at hand performs poorly especially in the regime of a large number of users due to the vanishing multicast rate. To overcome this detrimental effect we focus on the scheduling part and propose opportunistic scheduling policies of low-complexity compared to the queued scheme in Chapter 3. In particular, we propose a threshold-based scheduling that requires only statistical channel state information and one-bit feedback from each user. More specifically, each user indicates via feedback whenever its SNR is above a threshold determined solely by the fading statistics and the fairness requirement. Surprisingly, we prove that this simple scheme achieves the optimal utility in the regime of a large number of users. Numerical examples show that our proposed scheme performs closely to the scheduling with full channel state information, but at a significantly reduced complexity.

4.1 Introduction

We consider the same system model as in chapter 3, where the users, equipped with a memory of finite size, experience asymmetric fading statistics. We provided a long-term analysis on the performance of Maddah Ali and Niesen coded caching fading broadcast channel and showed its limitation. Although the proposed online delivery scheme is able to exploit both coded caching gain and fading peaks while ensuring fairness among users, its complexity grows exponentially with the number of users. This makes the scheme of Chapter 3 difficult to implement in a system with a large number of users. In this chapter, focusing on scheduling, we propose a more practical scheme with reduced complexity. We provide a rigorous analysis on the long-term average per-user rate in the regime of a large number of users.

Our contribution in this Chapter is three-fold:

1. We propose a simple threshold-based scheduling policy and determine the threshold as a function of the fading statistics for each fairness parameter α. Such threshold-based scheme exhibits two interesting features. On the one hand, the complexity is linear in K and significantly reduced with respect to the original problem where the search is done over K^2 variables. On the other hand, a threshold-based policy does not require the exact channel state information but only a one-bit feedback from each user. Namely, each user indicates whether its measured SNR is above the threshold set before the communication.

2. We prove that the proposed threshold-based scheduling policy is asymptotically optimal in Theorem 3. Namely, the utility achieved by our proposed policy converges to the optimal value as the number of users grows. The proof of Theorem 3 involves essentially three steps. First, we characterize the lower and upper bounds on the long-term average rate of each user. Second, we prove that the size of the selected user set grows unbounded as the number of users grows. Finally, we prove the convergence of the utility value.

3. Our numerical experiments show that the proposed scheme indeed achieves a near-optimal performance. Namely, it converges to the selection scheme with full channel knowledge as the number of users and/or SNR increases. Such scheme is therefore appropriate for a large number of users. In addition, the multicast rate is less sensitive to the user in the worst fading condition in the large SNR regime. Furthermore, the speed of convergence increases with the memory size and/or α-fair parameter.

Throughout this Chapter, we use the notation $\xrightarrow{\mathbb{P}}$ to denote convergence in probability and $\xrightarrow{a.s.}$ to denote almost sure convergence.

4.2 System Model and Objectives

4.2.1 System model

We consider the same channel model as in the previous chapter. We recall the network model given by a content delivery system where a server with N files wishes to convey the requested files to K users over a wireless downlink channel. We assume that N files are of equal size of F bits and have equal popularity, while each user k has a cache memory Z_k of size MF bits, where $M \geq 1$ denotes the cache size measured in files. We often use the normalized cache size denoted by $m = M/N$. We also restrict the cache placement to be performed according to decentralized cache placement [2]. As depicted in Fig. 3.1, we relax the perfect shared link assumption and consider a wireless channel modeled by a standard block-fading broadcast channel (3.1).

In Sections/Subsections 4.4, 4.6.1 and 4.6, we provide our results for the case where the channel coefficient of user k is exponentially distributed with mean ρ_k. For simplicity, we use in the following sections, the notation $h_k(t)$ to denote $Ph_k(t)$.

Unlike Chapter 3, we do not consider any admission control and we focus only on the scheduling part to maximize some utility function described in the following section.

4.2.2 Objectives

Our objective is to design a scheduling policy maximizing a utility function of *alpha fair* family of concave functions. Such policy benefits from coded caching gain and at the same time exploits CSI opportunistically. We restrict our self to the scheduling policy region $\mathit{\Pi}$ such that:

- At each slot t, and channel realization $\mathbf{h}(t)$, we select a subset(s) of users to perform the content delivery scheme of Maddah-Ali and Niesen [2].
- The caching placement and delivery follow the decentralized scheme [2].
- The transmission is under time sharing strategy.
- The users request distinct files.

Under policy π, we denote the long-term average rate of user i by $\bar{r}_i^{\pi}$, which is the expectation of the instantaneous data rate over the channel realizations $\mathbf{h}$.

We are interested in utility-optimal scheduling, where the goal is to maximize some utility function of the long-term rates. We restrict our attention to α-fair allocations (3.9), namely,

$$\pi^{\star} \in \underset{\pi \in \mathit{\Pi}}{\operatorname{argmax}} \left\{ \frac{1}{K} \sum_{i=1}^{K} g_{\alpha}(\bar{r}_i^{\pi}) \right\}. \tag{4.1}$$

The solution of the maximization problem in (4.1) is obtained by applying GDS schedular recalled in Subsection 1.4.3. At each slot t and for each channel realizations $\mathbf{h}$, the GDS rule is given by (1.21)

$$r_k^*(t) = \underset{\mathbf{r}}{\operatorname{argmax}} \sum_{k=1}^{K} \frac{r_k}{u_k(t)^\alpha}, \tag{4.2}$$

where r_k is the service rate of user k and $\mathbf{u}(t) = (u_1(t), ..., u_K(t))$ is the vector of empirical data rates up to time t, and obeys the recursive equation:

$$u_i(t+1) = \frac{1}{t+1}\left[tu_i(t) + r_i^*(t)\right]. \tag{4.3}$$

In the following section we provide the optimal scheduling policy for (4.1). Moreover, we propose a simple threshold-based scheme and prove that it is asymptotically optimal for a large number of users.

Note that when relaxing the time sharing strategy condition we prove in 4.5 that superposition scheme is optimal. However, such policy is much complex for implementation. In fact, additionally to its channel sate, each user needs to know the channel state of the weaker users in order to apply successive cancellation technique to decode its message.

4.3 Selection Scheme

By performing coded caching to the user subset $\mathcal{J}$, the total number of bits to be multicast to satisfy $|\mathcal{J}|$ distinct demands is equal to $T(m, |\mathcal{J}|)F$ bits. By letting $R_{\mathcal{J}}$ denote the multicast rate of the codewords intended to user subset $\mathcal{J}$, the per-user rate after applying coded caching to subset $\mathcal{J}$ is given by $\frac{1}{T(m,|\mathcal{J}|)}R_{\mathcal{J}}$ for any user in $\mathcal{J}$. Under time sharing strategy, we allocate a fraction of time $\tau_{\mathcal{J}}$ to the subset of users $\mathcal{J}$, with $\sum_{\mathcal{J}\subseteq[K]} \tau_{\mathcal{J}} = 1$. The base station performs coded caching content delivery on users in $\mathcal{J}$ and transmits codewords at rate $\log(1 + \min_{k\in\mathcal{J}} h_k)$ during $\tau_{\mathcal{J}}$ fraction of time slot.

4.3.1 Scheduling rule

Under selection scheme, the per-user data rate is given by

$$r_k = \sum_{\mathcal{J}: k\in\mathcal{J}\subseteq[K]} \frac{\tau_{\mathcal{J}}}{T(m, |\mathcal{J}|)} \log(1 + \min_{k\in\mathcal{J}} h_k), \tag{4.4}$$

By plugging (4.4) into (4.2) we obtain

$$\sum_{k=1}^{K} \frac{r_k}{u_k(t)} = \sum_{k=1}^{K} \frac{\sum_{\mathcal{J}:k\in\mathcal{J}\subseteq[K]} \frac{\tau_{\mathcal{J}}}{T(m,|\mathcal{J}|)}}{u_k(t)} \log(1+\min_{k\in\mathcal{J}} h_k) \tag{4.5}$$

$$= \sum_{\mathcal{J}:\mathcal{J}\subseteq[K]} \sum_{k:k\in\mathcal{J}} \frac{\frac{\tau_{\mathcal{J}}}{T(m,|\mathcal{J}|)}}{u_k(t)} \log(1+\min_{k\in\mathcal{J}} h_k) \tag{4.6}$$

$$= \sum_{\mathcal{J}:\mathcal{J}\subseteq[K]} \tau_{\mathcal{J}} \frac{\sum_{k:k\in\mathcal{J}} \frac{1}{u_k(t)^\alpha}}{T(m,|\mathcal{J}|)} \log(1+\min_{k\in\mathcal{J}} h_k), \tag{4.7}$$

where (4.6) holds because $\sum_{k=1}^{K}\sum_{\mathcal{J}:\mathcal{J}\subseteq[K]} = \sum_{\mathcal{J}:\mathcal{J}\subseteq[K]}\sum_{k:k\in\mathcal{J}}$. Thus, using (4.7), the maximization problem in (4.2) is equivalent to

$$\max_{\boldsymbol{\tau}:\sum_{\mathcal{J}}\tau_{\mathcal{J}}=1} \sum_{\mathcal{J}:\mathcal{J}\subseteq[K]} \tau_{\mathcal{J}} \frac{\sum_{k:k\in\mathcal{J}} \frac{1}{u_k(t)^\alpha}}{T(m,|\mathcal{J}|)} \log(1+\min_{k\in\mathcal{J}} h_k). \tag{4.8}$$

The optimal solution is readily given by

$$\tau_{\mathcal{J}} = \begin{cases} 1, & \text{if } \mathcal{J} = \operatorname{argmax}_{\mathcal{J}} \frac{\sum_{k:k\in\mathcal{J}} \frac{1}{u_k(t)^\alpha}}{T(m,|\mathcal{J}|)} \log(1+\min_{k\in\mathcal{J}} h_k), \\ 0, & \text{otherwise.} \end{cases} \tag{4.9}$$

Proposition 2. *The selection selecting at each time slot t the set of users:*

$$\mathcal{J}(\boldsymbol{h}(t),t) \in \operatorname*{argmax}_{\mathcal{J}\subseteq[K]} \frac{1}{T(m,|\mathcal{J}|)} \log(1+\min_{j\in\mathcal{J}} h_j) \sum_{i=1}^{K} \frac{\mathbf{1}\{i\in\mathcal{J}\}}{u_i(t)^\alpha}, \tag{4.10}$$

converges almost surely to a utility optimal scheduling in Π:

$$\frac{1}{K}\sum_{i=1}^{K} g_\alpha(u_i(t)) \to^{a.s.}_{t\to\infty} \max_{\pi\in\Pi} \left\{ \frac{1}{K}\sum_{i=1}^{K} g_\alpha(\bar{r}_i^{\pi}) \right\}.$$

The corresponding long-term average rate of user i is given by:

$$\bar{r}_i = \mathbb{E}\left(\frac{\mathbf{1}\{i\in\mathcal{J}(\boldsymbol{h}(t),t)\}}{T(m,|\mathcal{J}(\boldsymbol{h}(t),t)|)} \log(1+\min_{j\in\mathcal{J}(\boldsymbol{h}(t),t)} h_j) \right). \tag{4.11}$$

Proof. The result readily follows from (4.2) and (4.4)-(4.9). □

Therefore, utility-optimal scheduling can be achieved simply by applying the above scheme during a large number of time slots. By corollary, we deduce an alternative characterization of the optimal policy which is essential to prove our main result.

Corollary 5. *The following scheme yields a utility optimal scheduling:*

$$\mathcal{J}^\star(\boldsymbol{h}) \in \underset{\mathcal{J}}{\operatorname{argmax}} \left\{ \frac{1}{T(m,|\mathcal{J}|)} \log(1+\min_{j\in\mathcal{J}} h_j) \sum_{i=1}^{K} \frac{\mathbf{1}\{k\in\mathcal{J}\}}{(\bar{r}_i^{\pi^\star})^\alpha} \right\}.$$

Proof. The result holds as a consequence of proposition 2, by letting $t \to \infty$ in (4.10). Equation (4.10) indeed defines which group is selected by the above iterative scheme as $t \to \infty$. □

4.3.2 Complexity

Assume that $h_1(t) \geq ... \geq h_K(t)$, i.e. $\boldsymbol{h}(t)$ has been previously sorted. Define $k = \max \mathcal{J}(\mathbf{h}(t),t)$ the index of the worst user and the set size $s = |\mathcal{J}(\mathbf{h}(t),t)|$. Let ν_k be a permutation on $\{1,...,k\}$ such that $u_{\nu_k(1)}(t) \leq ... \leq u_{\nu_k(k)}(t)$. Since $\mathcal{J}(\mathbf{h}(t),t)$ is a maximizer of (4.10):

$$\begin{aligned}\frac{\log(1+h_k(t))}{T(m,s)} \sum_{i=1}^{K} \frac{\mathbf{1}\{i\in\mathcal{J}(\mathbf{h}(t),t)\}}{u_i(t)^\alpha} &= \frac{\log(1+h_k(t))}{T(m,s)} \max_{\mathcal{J}\subseteq[K]:|\mathcal{J}|=s,\max\mathcal{J}=k} \sum_{i=1}^{K} \frac{\mathbf{1}\{i\in\mathcal{J}\}}{u_i(t)^\alpha} \\ &= \frac{\log(1+h_k(t))}{T(m,s)} \sum_{i=1}^{s} \frac{1}{(u_{\nu_k(i)}(t))^\alpha}.\end{aligned}$$

This implies:

$$\mathcal{J}(\mathbf{h}(t),t) = \{\nu_k(1),...,\nu_k(s)\}.$$

Hence $\mathcal{J}(\mathbf{h}(t),t)$ can be computed by sorting $\boldsymbol{h}(t)$ and $\boldsymbol{u}(t)$, (with complexity $\mathcal{O}(K\log(K))$ using quick sort), and searching over the possible values of $k = 1,...,K$ and $s = 1,...,K$ (with complexity $\mathcal{O}(K^2)$). Thus, finding $\mathcal{J}(\mathbf{h}(t),t)$ takes time $\mathcal{O}(K^2)$. For the encoding operation, channel state information at the transmitter (CSIT) is necessary. For the decoding functions, only local channel state information (CSIR) at the receiver is needed, i.e. each user knows its channel state.

4.4 Threshold-Based Scheduling Scheme

We also introduce a sub-class of policies called threshold policies. We say that policy $\pi \in \mathit{\Pi}$ is a threshold policy with threshold c if, for any channel realization $\boldsymbol{h}$ it selects all users with a channel gain larger than c, that is:

$$\mathcal{J}^\pi(\boldsymbol{h}) = \{i = 1,\ldots,K : h_i \geq c\}. \tag{4.12}$$

We prove in the following that a well designed threshold policy in fact becomes optimal in $\mathit{\Pi}$, when the number of users K grows large.

4.4.1 Fair scheduling for large number of users

In this section, we consider utility optimal scheduling when the number of users K grows large. We show that threshold policies become optimal in this regime. Our result is general and applies to any value of $\alpha \geq 0$ as well as heterogeneous users where the channel gains statistics $\rho_1, ..., \rho_K$ are arbitrary as long as they are bounded. We denote by $\underline{\rho} = \min_i \rho_i$ and $\overline{\rho} = \max_i \rho_i$. As a corollary, we compute the optimal threshold policy in closed form as a function of $\rho_1, ..., \rho_K$, so that the system is indeed tractable.

We first state Theorem 17, the main technical contribution of this Chapter. That is, as the number of users grows large ($K \to \infty$), a well designed threshold policy become utility optimal, and that the optimal threshold may be derived explicitly as a function of the channel gains statistics $\rho_1, ..., \rho_K$.

Theorem 17. *Consider the solution of the optimization problem:*

$$c^\star \in \underset{c \geq 0}{\operatorname{argmax}} \left\{ \frac{1}{K} \sum_{i=1}^{K} g_\alpha \left(\log(1+c) e^{-\frac{c}{\rho_i}} \right) \right\}, \tag{4.13}$$

and $\pi_{c^\star}$ the threshold policy with threshold $c^\star$. Then the long term data rates under $\pi_{c^\star}$ are:

$$\bar{r}_i^{\pi_{c^\star}} = \frac{1}{T(m,\infty)} \log(1+c^\star) e^{-\frac{c^\star}{\rho_i}} + o(1) \ , \ K \to \infty. \tag{4.14}$$

Furthermore, $\pi_{c^\star}$ is asymptotically optimal in Π, in the sense that:

$$\frac{1}{K} \sum_{i=1}^{K} g_\alpha(\bar{r}_i^{\pi_{c^\star}}) = \max_{\pi \in \Pi} \left\{ \frac{1}{K} \sum_{i=1}^{K} g_\alpha(\bar{r}_i^{\pi}) \right\} + o(1) \ , \ K \to \infty.$$

Proof. Refer to Appendix C.1 □

We now show that, for $\alpha \geq 1$ the optimal threshold defined in (4.13) reduces to the maximization of a concave function, so that it can be computed efficiently using a local search method such as Newton's method.

Proposition 3. *Consider $c^\star$ the optimal threshold as defined in* (4.13)*. For $\alpha = 1$, the optimal threshold is given by:*

$$c^\star = e^{W_0\left(K(\sum_{i=1}^{K}(1/\rho_i))^{-1}\right)} - 1,$$

with W_0 the Lambert W function. For $\alpha \geq 1$, the optimal threshold is the unique solution to the equation:

$$(1+c)\log(1+c) = \frac{\sum_{i=1}^{K} e^{-\frac{c(1-\alpha)}{\rho_i}}}{\sum_{i=1}^{K} \frac{1}{\rho_i} e^{-\frac{c(1-\alpha)}{\rho_i}}}.$$

Proof. Refer to Appendix C.2. □

4.4.2 Complexity

While threshold policies are in general sub-optimal, they can be implemented with minimal complexity. Indeed, computing the solution of (4.10) can be done in time $\mathcal{O}(K^2)$. Whereas, computing a threshold policy requires $\mathcal{O}(K)$ time. Furthermore, while computing (4.10) requires all users to report the value of their channel gain $h_1(t), ..., h_K(t)$ up to a given accuracy, implementing a threshold policy simply requires user to report 1 bit of information which is $\mathbf{1}\{h_i(t) \geq c\}$.

Surprisingly, as stated in Theorem 17, a well designed threshold policy is asymptotically optimal when the number of users K grows large, so that utility optimal scheduling can be achieved with both linear complexity $\mathcal{O}(K)$ and 1-bit feedback.

4.5 Superposition Scheme

We relax the time sharing assumption given in the policies region Π and provide a superposition scheme that solves the utility maximization problem (4.1) in Π' ($\Pi \subseteq \Pi'$).

4.5.1 Scheduling rule

By simultaneously applying coded caching over different subset of users, the per-user data rate is given by

$$r_k = \sum_{\mathcal{J}:k\in\mathcal{J}\subseteq[K]} \frac{1}{T(m,|\mathcal{J}|)} R_{\mathcal{J}}. \tag{4.15}$$

Plugging (4.15) into (4.2) and following (4.5)-(4.7), we obtain

$$\sum_{k=1}^{K} \frac{r_k}{u_k(t)^\alpha} = \sum_{\mathcal{J}:\mathcal{J}\subseteq[K]} \frac{\sum_{k:k\in\mathcal{J}} \frac{1}{u_k(t)^\alpha}}{T(m,|\mathcal{J}|)} R_{\mathcal{J}}, \tag{4.16}$$

Thus, using (4.16), the maximization problem of (4.2) in Π' is equivalent to

$$\mathbf{R}^*(t) = \underset{\mathbf{R}\in\Gamma(\mathbf{h})}{\operatorname{argmax}} \sum_{\mathcal{J}:\mathcal{J}\subseteq[K]} \theta_{\mathcal{J}} R_{\mathcal{J}}, \quad with \quad \theta_{\mathcal{J}} = \frac{\sum_{k:k\in\mathcal{J}} \frac{1}{u_k(t)^\alpha}}{T(m,|\mathcal{J}|)}, \tag{4.17}$$

where $\Gamma(\mathbf{h}) \subseteq \mathbb{R}_+^{2^K-1}$ is the capacity region of K-user degraded Gaussian broadcast channel with $2^K - 1$ independent messages characterized in Theorem 14 in Subsection 3.4.4. We solve the maximization problem using Theorem 15 in Subsection 3.4.4 and Algorithm 1 in Subsection 1.4.2. To summarize the proposed scheme: at each slot t, 1) given the weights $\theta_{\mathcal{J}}$, we calculate $\mathbf{R}^*(t) \in \mathbb{R}_+^{2^K-1}$, 2) using superposition scheme we obtain at most K parallel channels of capacity given by $\mathbf{R}^*(t)$, which enable the base station to perform coded caching content delivery simultaneously to different subset of users (i.e. sends $T(m,|\mathcal{J}|)F$ bits to user subset $\mathcal{J}$ at rate $R_{\mathcal{J}}^*$).

Proposition 4. *Under the above scheme* $\mathbf{u}(t)$ *converges almost surely to a utility optimal allocation:*

$$\frac{1}{K}\sum_{i=1}^{K} g_\alpha(u_i(t)) \to^{a.s.}_{t\to\infty} \max_{\pi\in\Pi'}\left\{\frac{1}{K}\sum_{i=1}^{K} g_\alpha(\bar{r}_i^{\pi})\right\}.$$

Proof. The result readily follows from (4.2) and (4.15)-(4.17). □

4.5.2 Complexity

As shown in Subsection 3.4.4 of Chapter 3, the maximization problem of (4.17) with 2^K-1 variables, can be simplified to K variables. Without loss of generality we suppose $h_1 \geq .. \geq h_K$. Using Theorem 15 in Subsection 3.4.4, maximizing (4.17) is equivalent to

$$\max_{\mathbf{R}\in\Gamma(\mathbf{h})} \sum_{k=1}^{K} \theta_{\mathcal{J}_k} R_{\mathcal{J}_k}, \tag{4.18}$$

where $\mathcal{J}_k = \text{argmax}_{\mathcal{J}:k\in\mathcal{J}\subseteq\{1,..,k\}}\,\theta_{\mathcal{J}}$ and $R_{\mathcal{J}} = 0$ for $\mathcal{I} \notin \{\mathcal{J}_k : 1 \leq k \leq K\}$. Let ν_k be a permutation on $\{1,...,k-1\}$ such that $u_{\nu_k(1)}(t) \leq ... \leq u_{\nu_k(k-1)}(t)$. For a given k we have

$$\max_{\mathcal{J}:k\in\mathcal{J}\subseteq\{1,..,k\}} \theta_{\mathcal{J}} = \max_{\mathcal{J}:k\in\mathcal{J}\subseteq\{1,..,k\}} \frac{\sum_{k:k\in\mathcal{J}} \frac{1}{u_k(t)^\alpha}}{T(m,|\mathcal{J}|)} \tag{4.19}$$

$$= \max_{j:1\leq j\leq k} \frac{1}{T(m,j)}\left(\frac{1}{u_k(t)^\alpha} + \sum_{i=1}^{j-1}\frac{1}{u_{\nu_k(i)}(t)^\alpha}\right). \tag{4.20}$$

Thus, the complexity of searching for all $\{\mathcal{J}_k : 1 \leq k \leq K\}$ is $\mathcal{O}(K^2)$. Note that superposition encoding requires full CSI everywhere (transmitter and receivers) so that each user can decode its message by performing successive interference cancellation (SIC).

4.6 Special-Cases and Numerical Examples

We provide a rigorous analysis on the long-term average per-user rate for special cases and then we provide numerical examples on the proposed schemes.

4.6.1 Special cases

We consider provide a close form of the utility of selection scheme with full CSIT and of the threshold-based scheme for some special cases.

Selection scheme for symmetric channel statistics and $\alpha = 0$ in large P regime: We suppose that $\rho_i = P\ \forall i \in [K]$. In this case all users have the same long-term average rate since the channel statistics are equals. When no fairness is required ($\alpha = 0$), the

utility boils down to the average per user rate. Let $\bar{r}_{sc}$ denote the long-term average rate of any user in the system under the selection scheme given by (4.10).

Proposition 5. *For all K: the selection scheme coincides with baseline scheme in large P regime:*

$$\bar{r}_{sc} \sim \frac{\log(P)}{T(m,K)} \quad when \quad P \to \infty. \tag{4.21}$$

Proof. refere to Appendix C.3 □which coincides with the baseline scheme in large P regime in Proposition 1, which means that in such regime, the optimal scheduling rule is to apply coded caching of Maddah Ali and Niesen on all the users. **Threshold-based scheme in large K regime**

- For the symmetric channel case ($\rho_i = P$), we have from Theorem 17

$$c^\star = \underset{c\geq 0}{\operatorname{argmax}} \log(1+c)e^{-\frac{c}{P}} \tag{4.22}$$

$$= \underset{c\geq 0}{\operatorname{argmax}} \log\left(\log(1+c)\right) - \frac{c}{P}. \tag{4.23}$$

  Since $c \mapsto \log\log(1+c)$ is strictly concave, there exist a unique solution given by $c^\star = e^{W_0(P)} - 1 = \frac{P}{W_0(P)} - 1$.

  Thus, the long term data rate of user i is given by

$$\bar{r}_i^{\pi_{c^\star}} = \frac{1}{T(m,\infty)} W_0\left(P\right) e^{-\frac{e^{W_0(P)}-1}{P}} \tag{4.24}$$

$$= \frac{1}{T(m,\infty)} W_0\left(P\right) e^{\frac{1}{P} - \frac{1}{W_0(P)}}. \tag{4.25}$$

  Furthermore, when $P \to \infty$ we have $W_0(P) \sim \log(P)$, and so

$$\bar{r}_i^{\pi_{c^\star}} \sim \frac{\log(P)}{T(m,\infty)} \tag{4.26}$$

  which coincides with selection scheme in (4.21) for large K.

- For $P \to 0$ (equivalent to $\rho_i \to 0$) and $\alpha > 1$, the optimal threshold is the unique solution of $(1+c^*)\ln(1+c^*) = \min_{1\leq i\leq K} \rho_i$. Thus,

$$c^* = e^{W_0\left(\min_{1\leq i\leq K} \rho_i\right)} - 1. \tag{4.27}$$

  Note that the optimal threshold for $\alpha = 1$ depends on the channel statistics of all users. However, when $\alpha > 1$ and $P \to 0$, the optimal threshold depends only on the worst user in terms of channel statistics and does not depend on α. Thus, in low P regime, and $\alpha > 1$, the threshold scheme boils down to the max min scheduler as confirmed in the following.

- For $\alpha \to \infty$: the optimal threshold is the unique solution of

$$(1+c^*)\ln(1+c^*) = \frac{\sum_{i=1}^{K} e^{-c^*(1-\alpha)/\rho_i}}{\sum_{i=1}^{K} \frac{e^{-c^*(1-\alpha)/\rho_i}}{\rho_i}} \tag{4.28}$$

$$\simeq \frac{e^{c^*\alpha/\min_{1\le i\le K}\rho_i}}{\frac{e^{c^*\alpha/\min_{1\le i\le K}\rho_i}}{\min_{1\le i\le K}\rho_i}} \tag{4.29}$$

$$= \min_{1\le i\le K} \rho_i. \tag{4.30}$$

Thus,

$$c^*(\alpha\to\infty) = e^{W_0\left(\min_{1\le i\le K}\rho_i\right)} - 1. \tag{4.31}$$

Note that $\alpha \to \infty$ corresponds to the max min per user rate, and so the threshold depends only on the worst user channel statistics, which coincides with (4.27) for $P \to 0$ and $\alpha > 1$.

4.6.2 Numerical Examples

In this section, we illustrate the performance of the various schemes defined in the previous sections through numerical experiments. For each scheme, we compute the long term average data rates of each user $\bar{r}_1, ..., \bar{r}_K$, and the corresponding utility $\frac{1}{K}\sum_{i=1}^{K} g_\alpha(\bar{r}_i)$, which is our objective function. The considered schemes are recalled below.

- Superposition: At each slot t, this scheme solves the weighted sum rate maximization problem in $\mathit{\Gamma}(\mathbf{h}(t)) \subseteq \mathbb{R}_+^{2^K-1}$, using Theorem 14 Chapter 3:

$$\mathbf{R}_{\text{sp}}(\mathbf{h}(t),t) = \arg\max_{\mathbf{R}\in\mathit{\Gamma}(\mathbf{h}(t))} \sum_{\mathcal{J}:\mathcal{J}\subseteq[K]} \theta_{\mathcal{J}}(t) R_{\mathcal{J}} \text{ with } \theta_{\mathcal{J}}(t) = \frac{\sum_{i\in\mathcal{J}} \frac{1}{u_i^\alpha(t)}}{T(m,|\mathcal{J}|)},$$

The average rate of user i is

$$\bar{r}_{\text{sp},i} = \lim_{t\to\infty} \mathbb{E}\left[\sum_{\mathcal{J}:i\in\mathcal{J}} \frac{1}{T(m,|\mathcal{J}|)} R_{\text{sp},\mathcal{J}}(\mathbf{h}(t),t)\right].$$

- Selection with full CSIT: At each slot t, this scheme selects the subset of users

$$\mathcal{J}_{\text{sc}}(\mathbf{h}(t),t) = \underset{\mathcal{J}\subseteq[K]}{\text{argmax}} \left\{ \frac{1}{T(m,|\mathcal{J}|)} \log(1+\min_{j\in\mathcal{J}} h_j(t)) \sum_{i=1}^{K} \frac{\mathbf{1}\{i\in\mathcal{J}\}}{u_i(t)^\alpha} \right\}.$$

The average rate of user i is:

$$\bar{r}_{\text{sc},i} = \lim_{t\to\infty} \mathbb{E}\left[\frac{1}{T(m,|\mathcal{J}_{\text{sc}}(\mathbf{h}(t),t)|)} \log(1+\min_{j\in\mathcal{J}_{\text{sc}}(\mathbf{h}(t),t)} h_j(t)) \mathbf{1}\{i\in\mathcal{J}_{\text{sc}}(\mathbf{h}(t),t)\}\right].$$

- Threshold-based selection: At each slot t, this scheme selects the subset of users $\mathcal{J}_{\text{th}}(\mathbf{h}(t)) = \{i : h_i(t) \geq c^\star\}$, where $c^\star$ is the threshold given by (4.13), and depends only on the channel statistics $\rho_1, ..., \rho_K$. The average rate of user i is:

$$\bar{r}_{\text{th},i} = \mathbb{E}\left[\frac{1}{T(m, |\mathcal{J}_{\text{th}}(\mathbf{h}(t))|)} \log(1 + \min_{j \in \mathcal{J}_{\text{th}}(\mathbf{h}(t))} h_j(t)) \mathbf{1}\{i \in \mathcal{J}_{\text{th}}(\mathbf{h}(t))\}\right].$$

- Baseline: At each slot t, this scheme selects the subset of users $\{1, ..., K\}$, and the average rate of user i is:

$$\bar{r}_{\text{bl},i} = \frac{1}{T(m, K)} \mathbb{E}\left[\log(1 + \min_{1 \leq j \leq K} h_j(t))\right].$$

- Unicast opportunistic scheduling: At each slot t, the server sends the remaining $(1-m)F$ bits to the corresponding user (exploits only the local caching gain). At slot t the server sends with full power to user

$$k^\star(t) = \arg \max_{1 \leq k \leq K} \frac{\log(1 + h_k(t))}{u_k(t)^\alpha}.$$

In all scenarios, we divide users into two classes of $K/2$ users each: strong users with $\rho_k = P$ and weak users with $\rho_k = 0.2P$. For each figure we consider a normalized cache size of $m = [0.1, 0.6]$. In Figs. 4.1, 4.3 and 4.5 we plot the utility versus K for $\alpha = 0$, $\alpha = 1$ and $\alpha = 2$ respectively at $P = 10$ dB. In Figs. 4.2, 4.4 and 4.6 we plot the utility versus P for $\alpha = 0$, $\alpha = 1$ and $\alpha = 2$ respectively with $K = 20$ users. We draw the following conclusions:

Complexity: As seen in Figs. 4.1-4.6, superposition encoding outperforms all the others schemes at the price of a larger complexity of coding/decoding $\mathcal{O}(K)$ compared to the other schemes whose complexity is $\mathcal{O}(1)$.

Number of users K: From Figs. 4.1, 4.3 and 4.5, the performance of the threshold-based scheme is as good as full CSIT selection scheme for a sufficiently large K, as predicted by Theorem 17. In Fig. 4.1, corresponding to $\alpha = 0$, the average per user rate of the baseline scheme vanishes as the number of users increases for both small and large cache size as predicted by Proposition 1 in Chapter 3. For $\alpha = 1$ and $\alpha = 2$, the utility of the baseline scheme decreases with the number of users. On the contrary, the utility of all the other schemes converges to a constant as K grows for all α.

Power constraint P: We observe in Figs. 4.2, 4.4 and 4.6 that the performance of full CSIT selection, threshold-based selection and baseline schemes becomes identical for large P, which is expected since in that case the multicast rate is not limited by users with small channel gains. Therefore, all users are selected. Note that Proposition 5 proves that the full CSIT selection scheme coincides with the baseline scheme in the large P regime for $\alpha = 0$.

Memory size m: Figs. 4.1-4.6 show that the gap between the threshold-based scheme and the full CSIT scheme decreases with the memory size. Such a behavior is justified by

Property 1 stating that the function $k \to T(m,k)$ converges to $\frac{1-m}{m}$ faster as the memory size m increases.

Alpha-fairness α: We now consider the performance as a function of the fairness parameter α. We notice that the gap between the selection with full CSIT and the threshold-based selection decreases as the parameter α increases. This is because both schemes tend to coincide with the baseline scheme, or max-min scheduler as $\alpha \to \infty$.

In summary, remarkably, even for a relatively reasonable number of users, say $K \geq 50$, the threshold-based selection scheme ensures near optimal performance, with both 1-bit feedback and linear complexity $\mathcal{O}(K)$, which makes this scheme appealing for practical implementation.

4.7 Conclusions

In order to overcome the limitation of coded caching in wireless channel, we have studied opportunistic scheduling schemes for coded caching over the asymmetric fading broadcast channel. Although, the same limitation was addressed in the previous Chapter, the objectives and approach are different. The proposed solution in Chapter 3 deals with asynchronous user demands through an optimal algorithm for joint file admission control, codeword construction and wireless transmissions. However, the queueing structure of the proposed scheme increases its implementation complexity. In this Chapter we have focused on the scheduling part and have proposed a simple threshold-based scheduling policy, which requires only statistical channel knowledge and can be implemented by a simple one-bit feedback from each user. Our striking result, through rigorous and rather involved analysis, demonstrates that such threshold-based policy is asymptotically optimal as the number of users grows. Additionally, the numerical examples show that our proposed policy incurs a negligible loss with respect to the optimal scheduling scheme (requiring full channel knowledge) for a reasonable number of users, i.e., between 20 to 100 users depending on the fairness parameter and the memory size. Albeit simple and appealing, the threshold-based scheme does not capture the asynchronous nature of user requests. In fact the scheduling rule is based on GDS scheduler, which supposes that each user has enough data to be served. Thus, adapting the threshold-based scheduler to the online scenario remains an open problem .

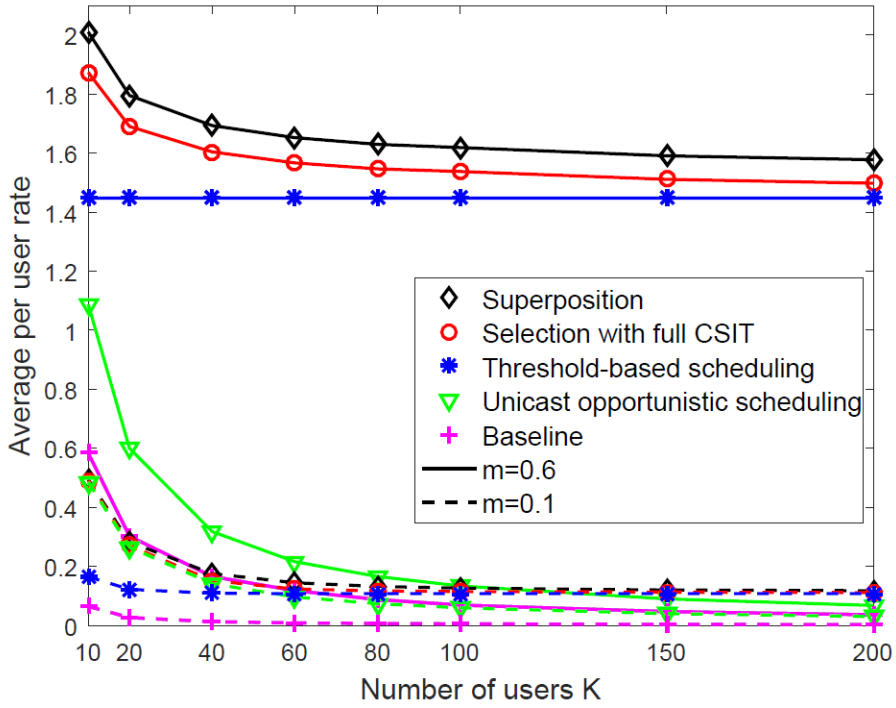


Figure 4.1: Average per user rate vs K for $\alpha = 0$, $P = 10$dB and $m = [0.1, 0.6]$.

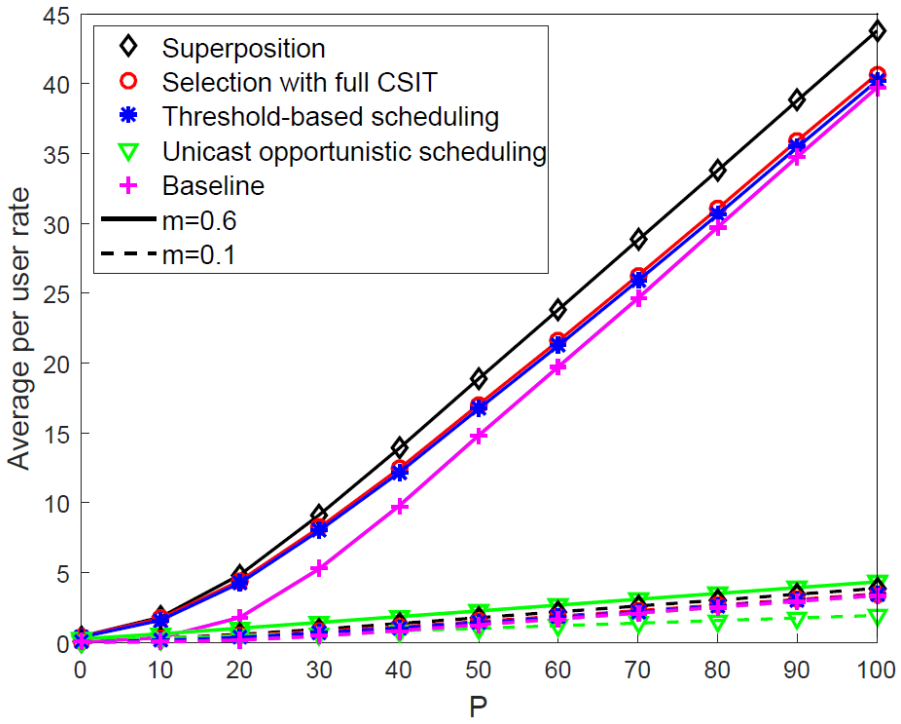


Figure 4.2: Average per user rate vs P for $\alpha = 0$, $K = 20$ and $m = [0.1, 0.6]$.

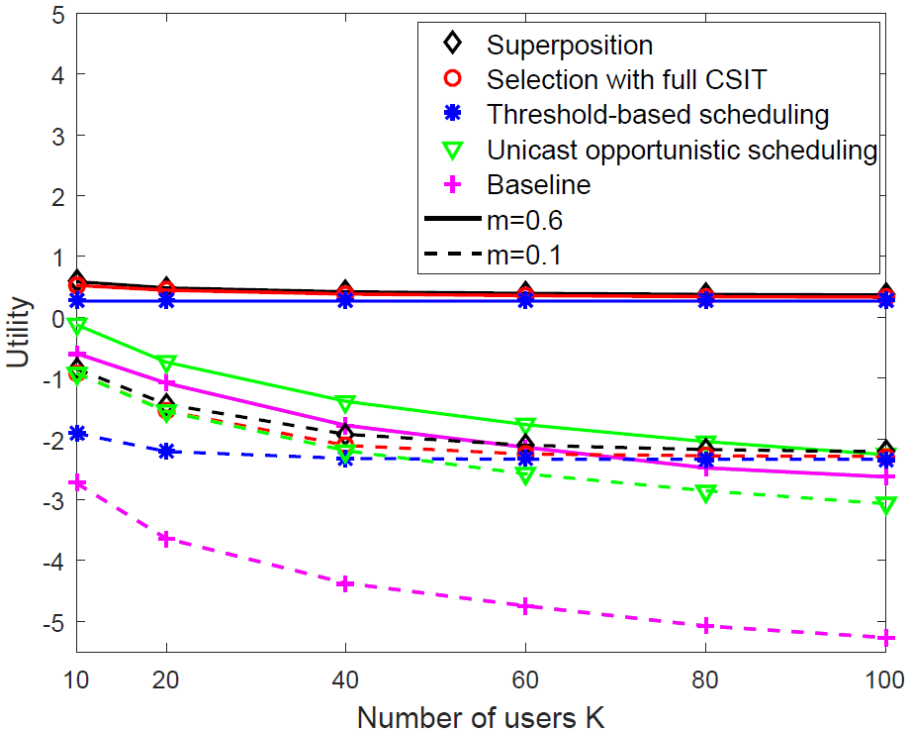


Figure 4.3: Utility vs K for $\alpha = 1$, $P = 10$dB and $m = [0.1, 0.6]$.

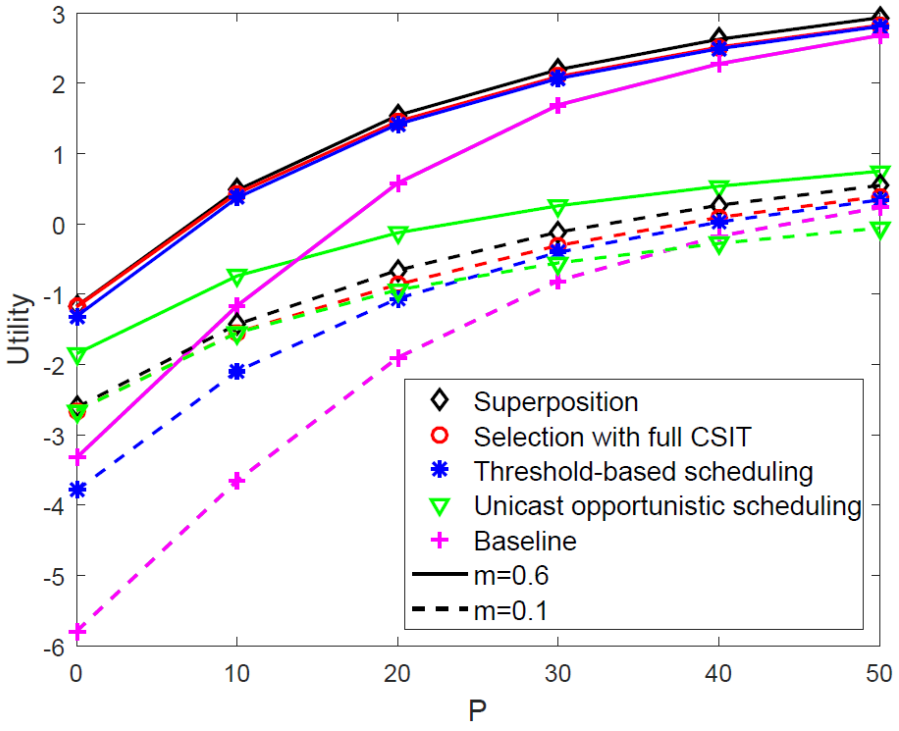


Figure 4.4: Utility vs P for $\alpha = 1$, $K = 20$ and $m = [0.1, 0.6]$.

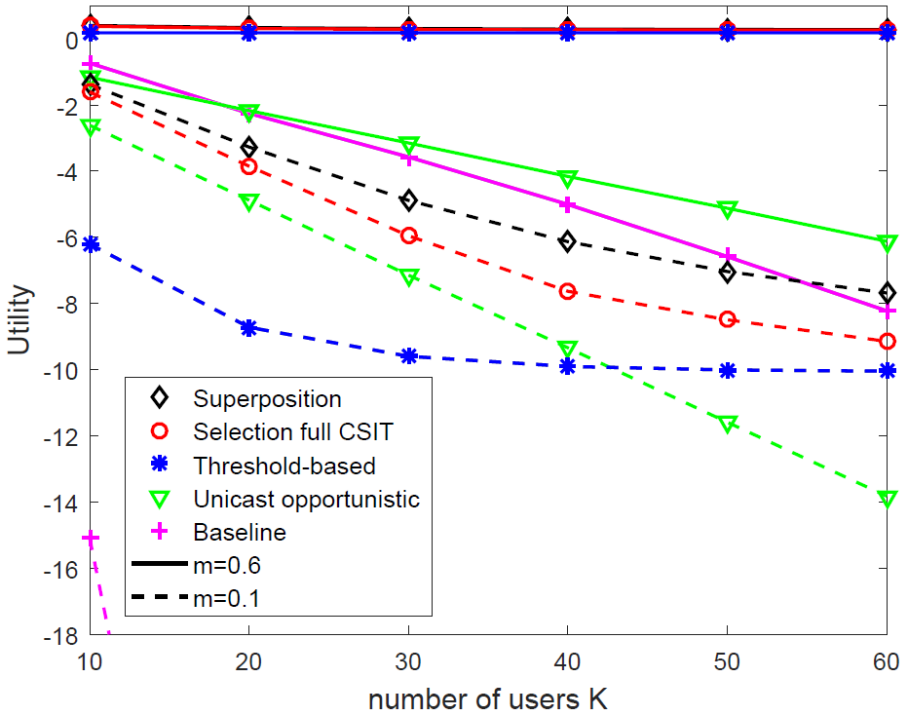


Figure 4.5: Utility vs K for $\alpha = 2$, $P = 10$dB and $m = [0.1, 0.6]$.

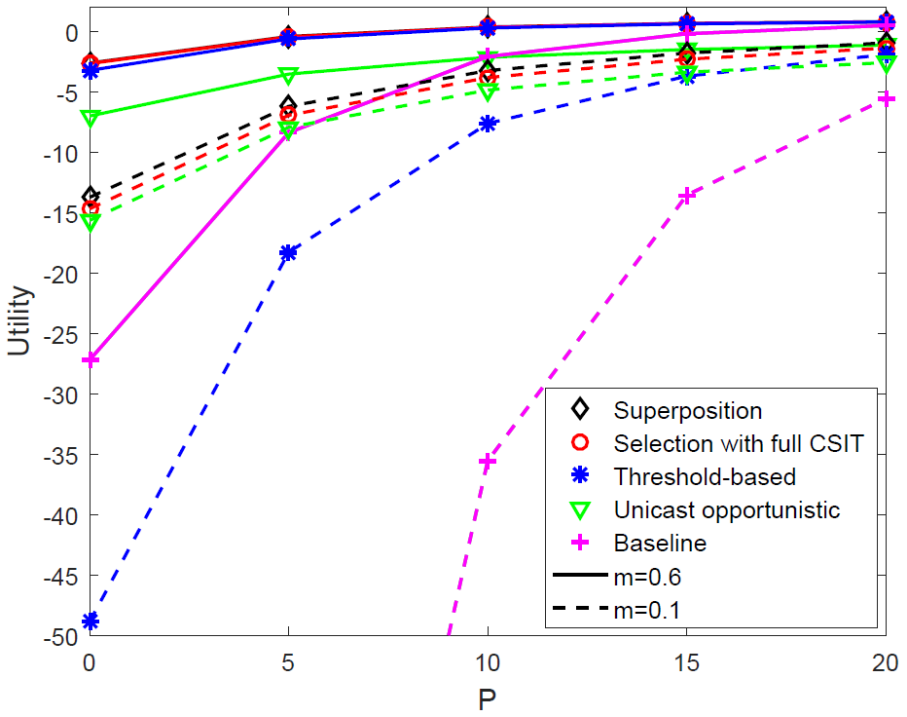


Figure 4.6: Utility vs P for $\alpha = 2$, $K = 20$ and $m = [0.1, 0.6]$.

Conclusions

We have studied coded caching [1, 2] in more realistic scenarios by relaxing the perfect shared link and considering wireless channel. Recent works have revealed that the theoretical gain of coded caching is sensitive to the behavior of the multicast rate of the underlying channel and might vanish in the regime of a large number of users. Thus we have focused on the design of efficient content delivery schemes in wireless channels when the placement phase is restricted to centralized [1] and/or decentralized [2] placement strategy.

We considered in Chapter 2, erasure broadcast channel with state feedback for asymmetric file sizes and distinct cache capacities. We demonstrated the benefits of coded caching combined with state feedback in the presence of random erasure and characterized the optimal rate region of the channel for some special cases, namely for $K \leq 3$, or for the symmetric network with $K \geq 3$, or for the one-sided fair rate vector with $K > 3$. The proposed scheme was based on the works by Wang and Gatzianas [59, 60] of which, we provided an intuitive interpretation and revealed an explicit connection between the capacity in the symmetric EBC and the DoF in the MISO-BC. More specifically, we showed that there exists a duality in terms of the order-j multicast capacity/DoF. Such a connection was fully exploited to generalize our results to the cache-enabled MISO-BC. Note that our proposed scheme uses only linear combinations (XOR) and so, can be further improved by using the joint source-channel coding [57].

In Chapter 3 and 4, we address the fairness problem in the presence of caches. In particular we studied the content delivery over asymmetric block-fading broadcast channel, where the channel quality varies across users and time. Unlike the classical coded caching which combines all the requested files, we exploit the fading peaks by deciding on the subset of users of which the requested files are linearly combined. The main conclusion of these chapters is that, by appropriately combining the multicast opportunities and the opportunism due to channel fading, coded caching can lead to significant gains in wireless systems with fading. Furthermore, in Chapter 3 we dealt with the dynamic arriving user requests through a queued scheme that combines file admission control, codeword construction and wireless transmissions. Moreover the above queueing structure cuts both ways, it makes our scheme flexible to be adapted to different setting such us asymmetric finite file size and/or distinct memory capacity, but it makes our optimal solution complex for implementation. Thus, in Chapter 4, we addressed the same limitation of coded caching in wireless channel in offline scenario by focusing on the scheduling part

and proposed a low-complex scheme that achieves a scalable sum content delivery in fading broadcast channel. In fact the proposed scheduling policy, named threshold-based scheduling, requires only statistical channel knowledge and can be implemented by a simple one-bit feedback from each user. We demonstrated that such threshold-based policy is asymptotically optimal as the number of users grows.

To see the tradeoff between complexity and performance of the proposed schemes for the fading BC, we compare their complexity in Table 4.1. Although the queued scheme provides the best performance as shown in Fig. 3.4, it has an exponentially growing complexity with the system dimension. However, the threshold-based scheme has linear complexity which makes it appealing even though it provides the lowest performance. Moreover, the threshold-based scheme requires the less channel state information among all the proposed schemes.

In all the above content delivery scheme over wireless channels, we did not consider any delivery delay constraints which can be important in several applications. For example, each file can have a delay tolerance that should be respected, otherwise the request is dropped. In that case, additionally to the channel state we need to consider the delay to decide on the scheduling policy, which remains an open problem for future work.

Table 4.1: Comparison of the proposed schemes for the fading BC.

Sec.	delivery scheme	Required CSI	Encoding	Decoding	Complexity
3.4	Queued online scheme	Full CSI at everyone	• Superposition encoding to perform multiple coded caching • $2^K - 1$ codeword queues	SIC to decode multiple sub-files and remove sub-files of weaker users	$\mathcal{O}(2^K)$
4.5	Superposition scheme	Full CSI at everyone	Select multiple subset of users to perform multiple coded caching	SIC to decode multiple sub-files and remove sub-files of weaker users	$\mathcal{O}(K^2)$
4.3	Selection scheme	Full CSIT local CSIR	Select the best user set based on GDS	XOR decoding	$\mathcal{O}(K^2)$
4.4	Threshold scheme	Statistical CSIT and local CSIR	Perform coded caching to the subset of users whose SNR is above a threshold	XOR decoding	$\mathcal{O}(K)$

Appendix A

Erasure Broadcast Channels with Feedback

In the appendix, we repeatedly use the following weight expression.

$$w_{\mathcal{J}} = \frac{\prod_{j\in\mathcal{J}}(1-m_j)}{1-\prod_{j\in\mathcal{J}}\delta_j} = \frac{\overline{m}_{\mathcal{J}}}{1-\delta_{\mathcal{J}}} \tag{A.1}$$

where we let $\overline{m}_j = 1-m_j$ and use a short-hand notation $\delta_{\mathcal{J}} = \prod_{j\in\mathcal{J}}\delta_j$ and $\overline{m}_{\mathcal{J}} = \prod_{j\in\mathcal{J}}\overline{m}_j$.

A.1 Proof of Lemma 6

We have, for $\mathcal{J}\subseteq\mathcal{I}$,

$$\begin{aligned}
&H(Y_{\mathcal{J}}^n \,|\, U, S^n) && \text{(A.2)}\\
&\quad= \sum_{t=1}^{n} H(Y_{\mathcal{I},t} \,|\, Y_{\mathcal{J}}^{t-1}, U, S^n) && \text{(A.3)}\\
&\quad= \sum_{t=1}^{n} H(Y_{\mathcal{I},t} \,|\, Y_{\mathcal{J}}^{t-1}, U, S^{t-1}, S_t) && \text{(A.4)}\\
&\quad= \sum_{t=1}^{n} \Pr\{S_t\cap\mathcal{I}\neq\emptyset\}\, H(X_t \,|\, Y_{\mathcal{J}}^{t-1}, U, S^{t-1}, S_t\cap\mathcal{I}\neq\emptyset) && \text{(A.5)}\\
&\quad= \sum_{t=1}^{n} \big(1-\prod_{i\in\mathcal{I}}\delta_i\big) H(X_t \,|\, Y_{\mathcal{J}}^{t-1}, U, S^{t-1}) && \text{(A.6)}\\
&\quad\le \big(1-\prod_{i\in\mathcal{I}}\delta_i\big) \sum_{t=1}^{n} H(X_t \,|\, Y_{\mathcal{J}}^{t-1}, U, S^{t-1}) && \text{(A.7)}
\end{aligned}$$

where the first equality is from the chain rule; the second equality is because the current input does not depend on future states conditioned on the past outputs/states and U;

the third one holds since $Y_{\mathcal{I},t}$ is deterministic and has entropy 0 when all outputs in $\mathcal{I}$ are erased ($S_t \cap \mathcal{I} = \emptyset$); the fourth equality is from the independence between X_t and S_t; and we get the last inequality by removing the terms $Y_{\mathcal{J}\setminus\mathcal{G}}^{t-1}$ in the condition of the entropy. Following the same steps, we have

$$H(Y_{\mathcal{G}}^n \,|\, U, S^n) = \big(1 - \prod_{i\in\mathcal{G}} \delta_i\big) \sum_{t=1}^{n} H(X_t \,|\, Y_{\mathcal{G}}^{t-1}, U, S^{t-1}), \tag{A.8}$$

from which and (A.7), we obtain (2.18).

A.2 Length of sub-phase

In this section, we prove (2.78) given by

$$t_{\mathcal{G}}^{\{k\}} = \sum_{\mathcal{H}:\mathcal{H}\subseteq\mathcal{G}\setminus\{k\}} (-1)^{|\mathcal{H}|} \frac{\prod_{j\in[K]\setminus\mathcal{G}\cup\{k\}\cup\mathcal{H}}(1-m_j)}{1-\prod_{j\in[K]\setminus\mathcal{G}\cup\{k\}\cup\mathcal{H}}\delta_j} F_k. \tag{A.9}$$

To this end, we first introduce a new variable $g_{\mathcal{G}}^{\{k\}} = \frac{t_{\mathcal{G}}^{\{k\}}}{F_k}$ for $k \in \mathcal{G} \subseteq [K]$. Using (2.77) we obtain

$$\sum_{\mathcal{I}:k\in\mathcal{I}\subseteq\mathcal{G}} g_{\mathcal{I}}^{\{k\}} = w_{[K]\setminus\mathcal{G}\cup\{k\}}. \tag{A.10}$$

We first need to prove the following lemma.

Lemma 18. *For any nonempty set $[K]$ and $\mathcal{G} \subseteq [K]$. It holds*

$$\sum_{\mathcal{I}:\mathcal{I}\subseteq\mathcal{G}} \sum_{\mathcal{H}:\mathcal{H}\subseteq\mathcal{I}} (-1)^{|\mathcal{H}|} w_{[K]\setminus\mathcal{I}\cup\mathcal{H}} = w_{[K]\setminus\mathcal{G}} \tag{A.11}$$

Proof.

$$\begin{aligned}
\sum_{\mathcal{I}:\mathcal{I}\subseteq\mathcal{G}} \sum_{\mathcal{H}:\mathcal{H}\subseteq\mathcal{I}} (-1)^{|\mathcal{H}|} w_{[K]\setminus\mathcal{I}\cup\mathcal{H}} &= \sum_{\mathcal{I}:\mathcal{I}\subseteq\mathcal{G}} \sum_{\mathcal{H}:\mathcal{H}\subseteq\mathcal{I}} (-1)^{|\mathcal{H}|} w_{[K]\setminus(\mathcal{I}\setminus\mathcal{H})} && \text{(A.12)}\\
&= \sum_{\mathcal{I}:\mathcal{I}\subseteq\mathcal{G}} \sum_{\mathcal{H}':\mathcal{H}'\subseteq\mathcal{I}} (-1)^{|\mathcal{I}\setminus\mathcal{H}'|} w_{[K]\setminus\mathcal{H}'} && \text{(A.13)}\\
&= \sum_{\mathcal{H}':\mathcal{H}'\subseteq\mathcal{G}} \sum_{\mathcal{I}:\mathcal{H}'\subseteq\mathcal{I}\subseteq\mathcal{G}} (-1)^{|\mathcal{I}\setminus\mathcal{H}'|} w_{[K]\setminus\mathcal{H}'} && \text{(A.14)}\\
&= \sum_{\mathcal{H}':\mathcal{H}'\subseteq\mathcal{G}} w_{[K]\setminus\mathcal{H}'} \sum_{\mathcal{I}:\mathcal{H}'\subseteq\mathcal{I}\subseteq\mathcal{G}} (-1)^{|\mathcal{I}\setminus\mathcal{H}'|} && \text{(A.15)}\\
&= \sum_{\mathcal{H}':\mathcal{H}'\subseteq\mathcal{G}} w_{[K]\setminus\mathcal{H}'} \sum_{\mathcal{I}':\mathcal{I}'\subseteq\mathcal{G}\setminus\mathcal{H}'} (-1)^{|\mathcal{I}'|} && \text{(A.16)}\\
&= w_{[K]\setminus\mathcal{G}} + \sum_{\mathcal{H}':\mathcal{H}'\subset\mathcal{G}} w_{[K]\setminus\mathcal{H}'} \sum_{\mathcal{I}':\mathcal{I}'\subseteq\mathcal{G}\setminus\mathcal{H}'} (-1)^{|\mathcal{I}'|} && \text{(A.17)}\\
&= w_{[K]\setminus\mathcal{G}}. && \text{(A.18)}
\end{aligned}$$

We set $\mathcal{H}' = \mathcal{I} \setminus \mathcal{H}$ and $\mathcal{I}' = \mathcal{I} \setminus \mathcal{H}'$ to obtain (A.13) and (A.16), respectively. The last equality follows from $\sum_{\mathcal{I}:\mathcal{I}\subseteq\mathcal{J}}(-1)^{|\mathcal{I}|} = 0$ for all $\mathcal{J} \neq \emptyset$. □

We prove (2.78) by induction on $|\mathcal{J}|$. For $\mathcal{J} = \{i\}$ we have $\sum_{\mathcal{I}:i\in\mathcal{I}\subseteq\mathcal{J}} g_{\mathcal{I}}^{\{i\}} = g_{\mathcal{J}}^{\{i\}}$ and $\sum_{\mathcal{H}:\mathcal{H}\subseteq\mathcal{J}\setminus\{i\}}(-1)^{|\mathcal{H}|}w_{[K]\setminus\mathcal{J}\cup\{i\}\cup\mathcal{H}} = w_{[K]\setminus\mathcal{J}\cup\{i\}}$. By applying (A.10) for $\mathcal{J} = \{i\}$, we obtain the proof for $|\mathcal{J}| = 1$.

Now suppose (2.78) holds for any $\mathcal{I} \subseteq [K]$ such that $|\mathcal{I}| < |\mathcal{J}|$ and we prove in the following that it holds for $\mathcal{J}$ too. We have

$$\sum_{\mathcal{I}:i\in\mathcal{I}\subseteq\mathcal{J}} g_{\mathcal{I}}^{\{i\}} = w_{[K]\setminus\mathcal{J}\cup\{i\}} \tag{A.19}$$

$$= g_{\mathcal{J}}^{\{i\}} + \sum_{\mathcal{I}:i\in\mathcal{I}\subset\mathcal{J}} g_{\mathcal{I}}^{\{i\}}. \tag{A.20}$$

Thus, we obtain

$$g_{\mathcal{J}}^{\{i\}} = w_{[K]\setminus\mathcal{J}\cup\{i\}} - \sum_{\mathcal{I}:i\in\mathcal{I}\subset\mathcal{J}} g_{\mathcal{I}}^{\{i\}} \tag{A.21}$$

$$= w_{[K]\setminus\mathcal{J}\cup\{i\}} - \sum_{\mathcal{I}:i\in\mathcal{I}\subset\mathcal{J}} \sum_{\mathcal{H}:\mathcal{H}\subseteq\mathcal{I}\setminus\{i\}} (-1)^{|\mathcal{H}|} w_{[K]\setminus\mathcal{I}\cup\{i\}\cup\mathcal{H}} \tag{A.22}$$

$$= w_{[K]\setminus\mathcal{J}\cup\{i\}} - \sum_{\mathcal{I}:i\in\mathcal{I}\subseteq\mathcal{J}} \sum_{\mathcal{H}:\mathcal{H}\subseteq\mathcal{I}\setminus\{i\}} (-1)^{|\mathcal{H}|} w_{[K]\setminus\mathcal{I}\cup\{i\}\cup\mathcal{H}} + \sum_{\mathcal{H}:\mathcal{H}\subseteq\mathcal{J}\setminus\{i\}} (-1)^{|\mathcal{H}|} w_{[K]\setminus\mathcal{J}\cup\{i\}\cup\mathcal{H}} \tag{A.23}$$

$$= w_{[K]\setminus\mathcal{J}\cup\{i\}} - \sum_{\mathcal{I}:\mathcal{I}\subseteq\mathcal{J}\setminus\{i\}} \sum_{\mathcal{H}:\mathcal{H}\subseteq\mathcal{I}} (-1)^{|\mathcal{H}|} w_{[K]\setminus\mathcal{I}\cup\mathcal{H}} + \sum_{\mathcal{H}:\mathcal{H}\subseteq\mathcal{J}\setminus\{i\}} (-1)^{|\mathcal{H}|} w_{[K]\setminus\mathcal{J}\cup\{i\}\cup\mathcal{H}} \tag{A.24}$$

$$= w_{[K]\setminus\mathcal{J}\cup\{i\}} - w_{[K]\setminus(\mathcal{J}\setminus\{i\})} + \sum_{\mathcal{H}:\mathcal{H}\subseteq\mathcal{J}\setminus\{i\}} (-1)^{|\mathcal{H}|} w_{[K]\setminus\mathcal{J}\cup\{i\}\cup\mathcal{H}} \tag{A.25}$$

$$= \sum_{\mathcal{H}:\mathcal{H}\subseteq\mathcal{J}\setminus\{i\}} (-1)^{|\mathcal{H}|} w_{[K]\setminus\mathcal{J}\cup\{i\}\cup\mathcal{H}}, \tag{A.26}$$

where (A.22) is obtained from the hypothesis of induction; (A.24) is obtained from a simple manipulation of the summation, i.e. $\sum_{\mathcal{I}:i\in\mathcal{I}\subseteq\mathcal{J}} a_{\mathcal{I}\setminus\{i\}} = \sum_{\mathcal{I}:\mathcal{I}\subseteq\mathcal{J}\setminus\{i\}} a_{\mathcal{I}}$; (A.25) is from Lemma 18.

A.3 Existence of the permutation

In this section, we prove that the worst user under the one-sided fair rate vector is determined by (2.79), namely

$$\arg\max_{k\in\mathcal{J}} t_{\mathcal{J}}^{\{k\}} = \min\{\mathcal{J}\} \quad , \forall\, \mathcal{J} \subseteq [K]. \tag{A.27}$$

We set $l = \min(\mathcal{J})$ for any subset $\mathcal{J} \subseteq [K]$ such that $|\mathcal{J}| \geq 2$. Proving (2.79) is equivalent to prove

$$R_l g_{\mathcal{J}}^{\{l\}} \geq R_i g_{\mathcal{J}}^{\{i\}} \quad \forall i \in \mathcal{J}. \tag{A.28}$$

Recall that from our one-sided rate vector assumption we have for $i \in \mathcal{J}$, $\delta_l \geq \delta_i$; $\delta_l R_l \geq \delta_i R_i$ and $\frac{\bar{m}_l}{m_l} R_l \geq \frac{\bar{m}_i}{m_i} R_i$. Plugging (2.1) and (2.39) into (2.76), we obtain

$$g_{\mathcal{J}}^{\{i\}} = \frac{1}{1 - \delta_{[K]\setminus\mathcal{J}\cup\{i\}}} \left[\sum_{\mathcal{I}: i \in \mathcal{I} \subset \mathcal{J}} g_{\mathcal{I}}^{\{i\}} \bar{\delta}_{\mathcal{J}\setminus\mathcal{I}} \delta_{[K]\setminus\mathcal{J}\cup\{i\}} + m_{\mathcal{J}\setminus\{i\}} \bar{m}_{[K]\setminus\mathcal{J}\cup\{i\}} \right], \tag{A.29}$$

and

$$g_{\mathcal{J}}^{\{l\}} = \frac{1}{1 - \delta_{[K]\setminus\mathcal{J}\cup\{l\}}} \left[\sum_{\mathcal{I}: l \in \mathcal{I} \subset \mathcal{J}} g_{\mathcal{I}}^{\{l\}} \bar{\delta}_{\mathcal{J}\setminus\mathcal{I}} \delta_{[K]\setminus\mathcal{J}\cup\{l\}} + m_{\mathcal{J}\setminus\{l\}} \bar{m}_{[K]\setminus\mathcal{J}\cup\{l\}} \right]. \tag{A.30}$$

We prove by induction on $|\mathcal{J}|$ that $R_l g_{\mathcal{J}}^{\{l\}} \geq R_i g_{\mathcal{J}}^{\{i\}}$: For $|\mathcal{J}| = 2$, $\mathcal{J} = \{l, i\}$ hence (A.29) and (A.30) imply the following

$$g_{\mathcal{J}}^{\{i\}} = \frac{1}{1 - \delta_{[K]\setminus\mathcal{J}\cup\{i\}}} \left[g_i^{\{i\}} \bar{\delta}_l \delta_{[K]\setminus\mathcal{J}\cup\{i\}} + m_l \bar{m}_{[K]\setminus\mathcal{J}\cup\{i\}} \right], \tag{A.31}$$

and

$$g_{\mathcal{J}}^{\{l\}} = \frac{1}{1 - \delta_{[K]\setminus\mathcal{J}\cup\{l\}}} \left[g_l^{\{l\}} \bar{\delta}_i \delta_{[K]\setminus\mathcal{J}\cup\{l\}} + m_i \bar{m}_{[K]\setminus\mathcal{J}\cup\{l\}} \right]. \tag{A.32}$$

Since $\delta_l \geq \delta_i$, it holds $\frac{1}{1-\delta_{[K]\setminus\mathcal{J}\cup\{l\}}} \geq \frac{1}{1-\delta_{[K]\setminus\mathcal{J}\cup\{i\}}}$ and $\bar{\delta}_i \geq \bar{\delta}_l$. Since $\frac{\bar{m}_l}{m_l} R_l \geq \frac{\bar{m}_i}{m_i} R_i$, then it holds $m_i \bar{m}_{[K]\setminus\mathcal{J}\cup\{l\}} R_l \geq m_l \bar{m}_{[K]\setminus\mathcal{J}\cup\{i\}} R_i$. In addition we have from (2.78) : $g_l^{\{l\}} = g_i^{\{i\}} = \frac{\bar{m}_{[K]}}{1-\delta_{[K]}}$ and $\delta_l R_l \geq \delta_i R_i$, thus we obtain $R_l g_{\mathcal{J}}^{\{l\}} \geq R_i g_{\mathcal{J}}^{\{i\}}$ for $|\mathcal{J}| = 2$.

Suppose that (A.28) holds for any $\mathcal{I} \subseteq [K]$ such that $|\mathcal{I}| < |\mathcal{J}|$ and we prove that it holds also for $\mathcal{J}$ in the following.

Since $\delta_l \geq \delta_i$, it holds $\frac{1}{1-\delta_{[K]\setminus\mathcal{J}\cup\{l\}}} \geq \frac{1}{1-\delta_{[K]\setminus\mathcal{J}\cup\{i\}}}$. Since $\frac{\bar{m}_l}{m_l} R_l \geq \frac{\bar{m}_i}{m_i} R_i$, it holds $m_{\mathcal{J}\setminus\{l\}} \bar{m}_{[K]\setminus\mathcal{J}\cup\{l\}} R_l \geq m_{\mathcal{J}\setminus\{i\}} \bar{m}_{[K]\setminus\mathcal{J}\cup\{i\}} R_i$. By observing (A.30) and (A.29), it remains to prove that

$$R_l \sum_{\mathcal{I}: l \in \mathcal{I} \subset \mathcal{J}} g_{\mathcal{I}}^{\{l\}} \bar{\delta}_{\mathcal{J}\setminus\mathcal{I}} \delta_{[K]\setminus\mathcal{J}\cup\{l\}} \geq R_i \sum_{\mathcal{I}: i \in \mathcal{I} \subset \mathcal{J}} g_{\mathcal{I}}^{\{i\}} \bar{\delta}_{\mathcal{J}\setminus\mathcal{I}} \delta_{[K]\setminus\mathcal{J}\cup\{i\}}. \tag{A.33}$$

We have for user l

$$\sum_{\mathcal{I}: l \in \mathcal{I} \subset \mathcal{J}} g_{\mathcal{I}}^{\{l\}} \bar{\delta}_{\mathcal{J}\setminus\mathcal{I}} \delta_{[K]\setminus\mathcal{J}\cup\{l\}} = \sum_{\mathcal{I}: \{l,i\} \subseteq \mathcal{I} \subset \mathcal{J}} g_{\mathcal{I}}^{\{l\}} \bar{\delta}_{\mathcal{J}\setminus\mathcal{I}} \delta_{[K]\setminus\mathcal{J}\cup\{l\}} + \sum_{l \in \mathcal{I} \subset \mathcal{J}\setminus\{i\}} g_{\mathcal{I}}^{\{l\}} \bar{\delta}_{\mathcal{J}\setminus\mathcal{I}} \delta_{[K]\setminus\mathcal{J}\cup\{l\}} \tag{A.34}$$

$$= \sum_{\mathcal{I}: \{l,i\} \subseteq \mathcal{I} \subset \mathcal{J}} g_{\mathcal{I}}^{\{l\}} \bar{\delta}_{\mathcal{J}\setminus\mathcal{I}} \delta_{[K]\setminus\mathcal{J}\cup\{l\}} + \sum_{\mathcal{I}: \mathcal{I} \subset \mathcal{J}\setminus\{i,l\}} g_{\mathcal{I}\cup\{l\}}^{\{l\}} \bar{\delta}_{\mathcal{J}\setminus\mathcal{I}\setminus\{l\}} \delta_{[K]\setminus\mathcal{J}\cup\{l\}}, \tag{A.35}$$

and similarly for user i

$$\sum_{\mathcal{I}:i\in\mathcal{I}\subset\mathcal{J}} g_{\mathcal{I}}^{\{i\}}\bar{\delta}_{\mathcal{J}\setminus\mathcal{I}}\delta_{[K]\setminus\mathcal{J}\cup\{i\}} = \sum_{\mathcal{I}:\{l,i\}\subseteq\mathcal{I}\subset\mathcal{J}} g_{\mathcal{I}}^{\{i\}}\bar{\delta}_{\mathcal{J}\setminus\mathcal{I}}\delta_{[K]\setminus\mathcal{J}\cup\{i\}} + \sum_{\mathcal{I}:i\in\mathcal{I}\subset\mathcal{J}\setminus\{l\}} g_{\mathcal{I}}^{\{i\}}\bar{\delta}_{\mathcal{J}\setminus\mathcal{I}}\delta_{[K]\setminus\mathcal{J}\cup\{i\}} \tag{A.36}$$

$$= \sum_{\mathcal{I}:\{l,i\}\subseteq\mathcal{I}\subset\mathcal{J}} g_{\mathcal{I}}^{\{i\}}\bar{\delta}_{\mathcal{J}\setminus\mathcal{I}}\delta_{[K]\setminus\mathcal{J}\cup\{i\}} + \sum_{\mathcal{I}:\mathcal{I}\subset\mathcal{J}\setminus\{l,i\}} g_{\mathcal{I}\cup\{i\}}^{\{i\}}\bar{\delta}_{\mathcal{J}\setminus\mathcal{I}\setminus\{i\}}\delta_{[K]\setminus\mathcal{J}\cup\{i\}}. \tag{A.37}$$

For any $\mathcal{I}$ satisfying $\{l,i\}\subseteq\mathcal{I}\subset\mathcal{J}$ we have $|\mathcal{I}|<|\mathcal{J}|$, $\min(\mathcal{I})=l$ and $i\in\mathcal{I}$ so by the hypothesis we have $g_{\mathcal{I}}^{\{l\}}R_l \geq g_{\mathcal{I}}^{\{i\}}R_i$. In addition we have $\delta_l\geq\delta_i$ thus

$$\sum_{\mathcal{I}:\{l,i\}\subseteq\mathcal{I}\subset\mathcal{J}} g_{\mathcal{I}}^{\{l\}}\bar{\delta}_{\mathcal{J}\setminus\mathcal{I}}\delta_{[K]\setminus\mathcal{J}\cup\{l\}}R_l \geq \sum_{\mathcal{I}:\{l,i\}\subseteq\mathcal{I}\subset\mathcal{J}} g_{\mathcal{I}}^{\{i\}}\bar{\delta}_{\mathcal{J}\setminus\mathcal{I}}\delta_{[K]\setminus\mathcal{J}\cup\{i\}}R_i. \tag{A.38}$$

For any $\mathcal{I}$ satisfying $\mathcal{I}\subset\mathcal{J}\setminus\{l,i\}$ we have from (2.78) $g_{\mathcal{I}\cup\{l\}}^{\{l\}} = g_{\mathcal{I}\cup\{i\}}^{\{i\}}$. In addition we have $\bar{\delta}_i\geq\bar{\delta}_l$ and $R_l\delta_l\geq R_i\delta_i$, then $\bar{\delta}_{\mathcal{J}\setminus\mathcal{I}\setminus\{l\}}\delta_{[K]\setminus\mathcal{J}\cup\{l\}}R_l \geq \bar{\delta}_{\mathcal{J}\setminus\mathcal{I}\setminus\{i\}}\delta_{[K]\setminus\mathcal{J}\cup\{i\}}R_i$. As a result we obtain

$$R_l\sum_{\mathcal{I}\subset\mathcal{J}\setminus\{i,l\}} g_{\mathcal{I}\cup\{l\}}^{\{l\}}\bar{\delta}_{\mathcal{J}\setminus\mathcal{I}\setminus\{l\}}\delta_{[K]\setminus\mathcal{J}\cup\{l\}} \geq R_i\sum_{\mathcal{I}\subset\mathcal{J}\setminus\{l,i\}} g_{\mathcal{I}\cup\{i\}}^{\{i\}}\bar{\delta}_{\mathcal{J}\setminus\mathcal{I}\setminus\{i\}}\delta_{[K]\setminus\mathcal{J}\cup\{i\}}. \tag{A.39}$$

Hence the proof is completed.

A.4 The outer-bound under the one-sided fair rate vector

Suppose that there exists π_1 such that $\sum_{j=1}^{K} R_{\pi_1(j)}w_{\pi_1(1)..\pi_1(j)}\leq 1$ and that $\pi_1(i)\leq\pi_1(i+1)$ holds for some $i\in[K-1]$. We prove that for any permutation π_2 that satisfies $\pi_2(i+1)=\pi_1(i)=k$, $\pi_2(i)=\pi_1(i+1)=k'$ and $\pi_1(j)=\pi_2(j)\ \forall\ j\in[K]\setminus\{i,i+1\}$, it holds $\sum_{j=1}^{K} R_{\pi_2(j)}w_{\pi_2(1)..\pi_2(j)}\leq 1$. It suffices to show that

$$\begin{aligned} &w_{\pi_1(1)..\pi_1(i)}R_{\pi_1(i)} + w_{\pi_1(1)..\pi_1(i+1)}R_{\pi_1(i+1)} \\ &\quad\geq w_{\pi_2(1)..\pi_2(i)}R_{\pi_2(i)} + w_{\pi_2(1)..\pi_2(i+1)}R_{\pi_2(i+1)} \end{aligned}$$

equivalent to

$$\begin{aligned} &(w_{\pi_1(1)..\pi_1(i)} - w_{\pi_2(1)..\pi_2(i+1)})R_{\pi_1(i)} \\ &\quad\geq (w_{\pi_2(1)..\pi_2(i)} - w_{\pi_1(1)..\pi_1(i+1)})R_{\pi_1(i+1)} \end{aligned}$$

equivalent to

$$(w_{\mathcal{I}k} - w_{\mathcal{I}kk'})R_k \geq (w_{\mathcal{I}k'} - w_{\mathcal{I}kk'})R_{k'}, \tag{A.40}$$

where $\mathcal{I} = \pi_1(1)..\pi_1(i-1)$. By replacing the weight by its expression (A.1) we obtain

$$w_{\mathcal{I}k} - w_{\mathcal{I}kk'} = \frac{\bar{m}_{\mathcal{I}k}}{1-\delta_{\mathcal{I}k}} - \frac{\bar{m}_{\mathcal{I}kk'}}{1-\delta_{\mathcal{I}kk'}} \tag{A.41}$$

$$= \bar{m}_{\mathcal{I}k}\left[\frac{1}{1-\delta_{\mathcal{I}k}} - \frac{1}{1-\delta_{\mathcal{I}kk'}} + \frac{m_{k'}}{1-\delta_{\mathcal{I}kk'}}\right] \tag{A.42}$$

$$= \bar{m}_{\mathcal{I}k}\left[\frac{(1-\delta_{\mathcal{I}kk'}) - (1-\delta_{\mathcal{I}k})}{(1-\delta_{\mathcal{I}k})(1-\delta_{\mathcal{I}kk'})} + \frac{m_{k'}}{1-\delta_{\mathcal{I}kk'}}\right] \tag{A.43}$$

$$= \frac{\bar{m}_{\mathcal{I}k}}{1-\delta_{\mathcal{I}kk'}}\left[\frac{\delta_{\mathcal{I}k}(1-\delta_{k'})}{(1-\delta_{\mathcal{I}k})} + m_{k'}\right], \tag{A.44}$$

where we obtain: (A.42) by replacing $\bar{m}_{k'}$ with $1 - m_{k'}$; (A.43) and (A.44) by trivial calculations. Similarly

$$w_{\mathcal{I}k'} - w_{\mathcal{I}kk'} = \frac{\bar{m}_{\mathcal{I}k'}}{1-\delta_{\mathcal{I}kk'}}\left[\frac{\delta_{\mathcal{I}k'}(1-\delta_k)}{(1-\delta_{\mathcal{I}k'})} + m_k\right]. \tag{A.45}$$

Thus, (A.40) is equivalent to

$$\frac{\delta_{\mathcal{I}}(1-\delta_{k'})}{(1-\delta_{\mathcal{I}k})}\bar{m}_k\delta_k R_k - \frac{\delta_{\mathcal{I}}(1-\delta_k)}{(1-\delta_{\mathcal{I}k'})}\bar{m}_{k'}\delta_{k'}R_{k'} + (\bar{m}_k m_{k'}R_k - \bar{m}_{k'}m_k R_{k'}) \geq 0. \tag{A.46}$$

Since $k \leq k'$ then $\delta_k \geq \delta_{k'}$, so it is sufficient to prove that

$$\frac{\delta_{\mathcal{I}}(1-\delta_k)}{(1-\delta_{\mathcal{I}k'})}\underbrace{[\bar{m}_k\delta_k R_k - \bar{m}_{k'}\delta_{k'}R_{k'}]}_{A} + \underbrace{(\bar{m}_k m_{k'}R_k - \bar{m}_{k'}m_k R_{k'})}_{B} \geq 0. \tag{A.47}$$

This is satisfied if $A \geq 0$ and $B \geq 0$. The condition B holds thanks to the definition of one-sided fair rate vector, and it is equivalent to

$$\frac{R_{k'}}{R_k} \leq \frac{\bar{m}_k m_{k'}}{\bar{m}_{k'}m_k} \triangleq \theta. \tag{A.48}$$

We will examine condition A by considering the case $m_{k'} \geq m_k$ and $m_k \geq m_{k'}$ separately.

- Case $\theta > 1$
  In this case we have $m_k < m_{k'}$, or $\bar{m}_k > \bar{m}_{k'}$. Condition A reduces to:

  $$\delta_k R_k - \delta_{k'}R_{k'} \geq 0.$$

- Case $\theta < 1$
  In this case we have $m_k > m_{k'}$ or $\bar{m}_k < \bar{m}_{k'}$. Then we have

  $$\frac{R_{k'}}{R_k} \leq \frac{\bar{m}_k m_{k'}}{\bar{m}_{k'}m_k} \leq \frac{\bar{m}_k\delta_k}{\bar{m}_{k'}\delta_{k'}} \leq \frac{\delta_k}{\delta_{k'}}.$$

This means that B implies A so that the desired inequality holds once B holds. Since A is inactive, we can then consider a looser bounds

$$\delta_k R_k - \delta_{k'} R_{k'} \geq 0,$$

which holds by the definition of one-sided fair rate vector.

Thus we obtain the result. Starting by π_1 as the identity we can obtain all the remaining $K! - 1$ permutations.

Appendix B

Fading Broadcast Channels with Dynamic User Requests

B.1 Proof of Proposition 1

The content delivery rate is:

$$\bar{r}_{\text{bl,sum}}(K,P) = \frac{K}{T(m,K)}\mathbb{E}\left[\log\left(1+Ph_{\min}\right)\right],$$

where $h_{\min} \triangleq \min_{k=1,\dots,K} h_k$. Since $(h_k)_{k=1,\dots,K}$ are i.i.d. with distribution Exp(1), $h_{\min}$ has distribution Exp(K). Hence:

$$\begin{aligned}\mathbb{E}\left[\log\left(1+Ph_{\min}\right)\right] &= \int_0^{+\infty} e^{-x}\log\left(1+\frac{P}{K}x\right)dx \\ &= e^{\frac{K}{P}}E_1\left(\frac{K}{P}\right),\end{aligned}$$

which yields statement (i).

When $K \to \infty$ we have $\frac{K}{T(m,K)} \sim \frac{Km}{1-m}$ and

$$\int_0^{+\infty} e^{-x}\log\left(1+\frac{P}{K}x\right)dx \sim \frac{P}{K}\int_0^{+\infty} xe^{-x}dx = \frac{P}{K},$$

Replacing yields statement (ii).

When $P \to \infty$, $\frac{K}{P} \to 0$. Since $E_1(x) \sim \log(1/x)$ for $x \to 0$ we obtain statement (iii).

B.2 Proof of Theorem 14

Let $M_{\mathcal{J}}$ be the message for all the users in $\mathcal{J} \subseteq [K]$ and of size $2^{nR_{\mathcal{J}}}$. We first show the converse. It follows that the set of $2^K - 1$ independent messages $\{M_{\mathcal{J}} :\ \mathcal{J} \subseteq [K], \mathcal{J} \neq \emptyset\}$

can be partitioned as

$$\bigcup_{k=1}^{K}\{M_{\mathcal{J}}:\ k\in\mathcal{J}\subseteq[k]\}. \tag{B.1}$$

We can now define K independent mega-messages $\tilde{M}_k := \{M_{\mathcal{J}}:\ k\in\mathcal{J}\subseteq[k]\}$ with rate $\tilde{R}_k := \sum_{\mathcal{J}:\,k\in\mathcal{J}\subseteq[k]} R_{\mathcal{J}}$. Note that each mega-message k must be decoded at least by user k reliably. Thus, the K-tuple $(\tilde{R}_1,\ldots,\tilde{R}_K)$ must lie inside the private-message capacity region of the K-user BC. Since it is a degraded BC, the capacity region is known, see Sub-section 1.4.1, then we obtain

$$\tilde{R}_k \le \log\frac{1+h_k\sum_{j=1}^{k}p_j}{1+h_k\sum_{j=1}^{k-1}p_j},\quad k=2,\ldots,K, \tag{B.2}$$

for some $p_j\ge 0$ such that $\sum_{j=1}^{K}p_j\le P$. This establishes the converse.

To show the achievability, it is enough to use rate-splitting. Specifically, the transmitter first assembles the original messages into K mega-messages, and then applied the standard K-level superposition coding [43] putting the $(k-1)$-th signal on top of the k-th signal. The k-th signal has average power p_k, $k\in[K]$. At the receivers' side, if the rate of the mega-messages are inside the private-message capacity region of the K-user BC, i.e., the K-tuple $(\tilde{R}_1,\ldots,\tilde{R}_K)$ satisfies (B.2), then each user k can decode the mega-message k. Since the channel is degraded, the users 1 to $k-1$ can also decode the mega-message k and extract its own message. Specifically, each user j can obtain $M_{\mathcal{J}}$ (if $\mathcal{J}\ni j$), from the mega-message k when $k\in\mathcal{J}\subseteq[k]$. This completes the achievability proof.

B.3 Proof of Theorem 15

The proof builds on the simple structure of the capacity region. We remark that for a given power allocation of users 1 to $k-1$, user k sees 2^{k-1} messages $\{M_{\mathcal{J}}\}$ for all $\mathcal{J}$ such that $k\in\mathcal{J}\subseteq\{1,\ldots,k\}$ with the equal channel gain. For a given set of $\{p_j\}_{j=1}^{k-1}$, the capacity region of these messages is a simple hyperplane characterized by 2^{k-1} vertices $R_{k,\text{sum}}\mathbf{e}_i$ for $i=1,\ldots,2^{k-1}$, where $R_{k,\text{sum}}$ is the sum rate of user k in the RHS of (3.21) and $\mathbf{e}_i$ is a vector with one for the i-th entry and zero for the others. Therefore, the weighted sum rate is maximized for user k by selecting the vertex corresponding to the largest weight, denoted by $\tilde{\theta}$. This holds for any k.

B.4 Static Policies

An important concept for characterizing the feasibility region and proving optimality of our proposed policy is the one we will refer to here as "static policies". The concept is that decisions taken according to these policies depend only on the channel state realization (i.e. the uncontrollable part of the system) as per the following definition:

Definition 6 (Static Policy)**.** *Any policy that selects the control variables* $\{\mathbf{a}(t), \boldsymbol{\sigma}(t), \boldsymbol{\mu}(t)\}$ *according to a probability distribution that depends only on the channel state* $\mathbf{h}(t)$ *will be called a static policy.*

It is clear from the definition that all static policies belong to the set of admissible policies for our setting. An important case is where actually admission control $\mathbf{a}(t)$ and codeword routing $\boldsymbol{\sigma}(t)$ are decided at random and independently of everything and transmissions $\boldsymbol{\mu}(t)$ are decided at by a distribution that depends only on the channel state realization of the slot: It can be shown using standard arguments in stochastic network optimization (see for example [61,63,65,66]) that the optimal long term file delivery vector and any file delivery vector in the stability region of the queueing system can be achieved by such static policies, as formalized by the following Lemmas:

Lemma 19 (Static Optimal Policy)**.** *Define a policy* $\pi^* \in \Pi^{CC}$ *that in each slot where the channel states are* $\mathbf{h}$ *works as follows: (i) it pulls random user demands with mean* $\bar{a}_k^*$*, and it gives the virtual queues arrivals with mean* $\overline{\gamma}_k = \overline{a}_k^*$ *as well (ii) the number of combinations for subset* $\mathcal{J}$ *is a random variable with mean* $\overline{\sigma}_{\mathcal{J}}^*$ *and uniformly bounded by* $\sigma_{\max}$*, (iii) selects one out of* $K+1$ *suitably defined rate vectors* $\boldsymbol{\mu}^{\mathbf{l}} \in \Gamma(\mathbf{h}), l = 1, .., K+1$ *with probability* $\psi_{l,\mathbf{h}}$*. The parameters above are selected such that they solve the following problem:*

$$\begin{aligned} \max_{\overline{\mathbf{a}}} & \sum_{k=1}^{K} g_k(\overline{a}_k^*) \\ s.t. \quad & \sum_{\mathcal{J}:k\in\mathcal{J}} \overline{\sigma}_{\mathcal{J}}^* \geq \overline{a}_k^*, \forall k \in \{1, .., K\} \\ & \sum_{\mathcal{J}:\mathcal{I}\subseteq\mathcal{J}} b_{\mathcal{J},\mathcal{I}} \overline{\sigma}_{\mathcal{J}}^* \leq T_{\text{slot}} \sum_{\mathbf{h}} \phi_{\mathbf{h}} \sum_{l=1}^{K+1} \psi_{l,\mathbf{h}} \mu_{\mathcal{I}}^l(\mathbf{h}), \forall \mathcal{I} \subseteq \{1, 2, ..., K\} \end{aligned}$$

Then, π^* *results in the optimal delivery rate vector (when all possible policies are restricted to set* Π^{CC}*).*

Lemma 20 (Static Policy for the $\delta-$ interior of Γ^{CC})**.** *Define a policy* $\pi^\delta \in \Pi^{CC}$ *that in each slot where the channel states are* $\mathbf{h}$ *works as follows: (i) it pulls random user demands with mean* $\overline{a}_k^\delta$ *such that* $(\overline{\mathbf{a}} + \boldsymbol{\delta}) \in \Gamma^{CC}$*, and gives the virtual queues random arrivals with mean* $\overline{\gamma}_k \leq \overline{a}_k + \epsilon'$ *for some* $\epsilon' > 0$ *(ii) the number of combinations for subset* $\mathcal{J}$ *is a random variable with mean* $\overline{\sigma}_{\mathcal{J}}^\delta$ *and uniformly bounded by* $\sigma_{\max}$*, (iii) selects one out of* $K+1$ *suitably defined rate vectors* $\boldsymbol{\mu}^{\mathbf{l}} \in \Gamma(\mathbf{h}), l = 1, .., K+1$ *with probability* $\psi_{l,\mathbf{h}}^\delta$*. The parameters above are selected such that:*

$$\begin{aligned} & \sum_{\mathcal{J}:k\in\mathcal{J}} \overline{\sigma^\delta}_{\mathcal{J}} \geq \epsilon + \overline{a}_k^\delta, \forall k \in \{1, .., K\} \\ & \sum_{\mathcal{J}:\mathcal{I}\subseteq\mathcal{J}} b_{\mathcal{J},\mathcal{I}} \overline{\sigma}_{\mathcal{J}}^\delta \leq \epsilon + T_{\text{slot}} \sum_{\mathbf{h}} \phi_{\mathbf{h}} \sum_{l=1}^{K+1} \psi_{l,\mathbf{h}}^\delta \mu_{\mathcal{I}}^l(\mathbf{h}), \forall \mathcal{I} \in 2^{\mathcal{K}} \end{aligned}$$

for some appropriate $\epsilon < \delta$*. Then, the system under* π^δ *has mean incoming rates of* $\overline{\mathbf{a}}^\delta$ *and is strongly stable.*

B.5 Proof of Lemma 13

We prove the Lemma in two parts: (i) $\mathit{\Gamma}^{CC} \subseteq \mathit{\Lambda}^{CC}$ and (ii) $\left(\mathit{\Gamma}^{CC}\right)^c \subseteq \left(\mathit{\Lambda}^{CC}\right)^c$.

For the first part, we show that if $\overline{\mathbf{a}} \in Int(\mathit{\Gamma}^{CC})$ then also $\overline{\mathbf{a}} \in \mathit{\Lambda}^{CC}$, that is the long term file delivery rate vector observed by the users as per (3.7) is $\overline{\mathbf{r}} = \overline{\mathbf{a}}$. Denote $\overline{A}_k(t)$ the number of files that have been admitted to the system for user k up to slot t. Also, note that due to our restriction on the class of policies $\mathit{\Pi}^{CC}$ and our assumption about long enough blocklengths, there are no errors in decoding the files, therefore the number of files correctly decoded for user k till slot t is $\overline{D}_k(t)$. From Lemma 20 it follows that there exists a static policy π^{RAND}, the probabilities of which depending only on the channel state realization at each slot, for which the system is strongly stable. Since the channels are i.i.d. random with a finite state space and queues are measured in files and bits, the system now evolves as a discrete time Markov chain $(\mathbf{S}(t), \mathbf{Q}(t), \mathbf{H}(t))$, which can be checked that is aperiodic, irreducible and with a single communicating class. In that case, strong stability means that the Markov chain is ergodic with finite mean.

Further, this means that the system reaches to the set of states where all queues are zero infinitely often. Let $T[n]$ be the number of timeslots between the $n-$th and $(n+1)-$th visit to this set (we make the convention that $T[0]$ is the time slot that this state is reached for the first time). In addition, let $\tilde{A}_k[n], \tilde{D}_k[n]$ be the number of demands that arrived and were delivered in this frame, respectively. Then, since within this frame the queues start and end empty, we have

$$\tilde{A}_k[n] = \tilde{D}_k[n], \forall n, \forall k.$$

In addition since the Markov chain is ergodic,

$$\overline{a}_k = \lim_{t\to\infty} \frac{A(t)}{t} = \lim_{N\to\infty} \frac{\sum_{n=0}^{N} \tilde{A}_k[n]}{\sum_{n=0}^{N} T[n]}$$

and

$$\overline{r}_k = \lim_{t\to\infty} \frac{D(t)}{t} = \lim_{N\to\infty} \frac{\sum_{n=0}^{N} \tilde{D}_k[n]}{\sum_{n=0}^{N} T[n]}$$

Combining the three expressions, $\overline{\mathbf{r}} = \overline{\mathbf{a}}$ thus the result follows.

We now proceed to show the second part, that is given any arrival rate vector $\overline{\mathbf{a}}$ that is not in the stability region of the queueuing system we cannot have a long term file delivery rate vector $\overline{\mathbf{r}} = \overline{\mathbf{a}}$. Indeed, since $\overline{\mathbf{a}} \notin \mathit{\Gamma}^{CC}$, for any possible $\overline{\boldsymbol{\sigma}}$ satisfying (3.12), for every $\overline{\boldsymbol{\mu}} \in \sum_{\mathbf{h}\in\mathcal{H}} \phi_{\mathbf{h}} \mathit{\Gamma}(\mathbf{h})$ there will be some subset(s) of users for which the corresponding inequality (3.13) is violated. Since codeword generation decisions are assumed to be irrevocable and $\sum_{\mathbf{h}\in\mathcal{H}} \phi_{\mathbf{h}} \mathit{\Gamma}(\mathbf{h})$ is the capacity region of the wireless channel, the above implies that there is not enough wireless capacity to satisfy a long term file delivery rate vector of $\overline{\mathbf{a}}$. Therefore, $\overline{\mathbf{a}} \notin \mathit{\Lambda}^{CC}$, finishing the proof. [1]

[1] We would also need to check the boundary of $\mathit{\Gamma}^{CC}$. Note, however, that by similar arguments we

B.6 Proof of Theorem 16

We first look at static policies, which take random decisions based only on the channel realizations. We focus on two such policies: (i) one that achieves the optimal utility, as described in Lemma 19 and (ii) one that achieves (i.e. admits and stabilizes the system for that) a rate vector in the $\delta-$ interior of Λ^{CC} (for any $\delta > 0$), as described in Lemma 20. Then, we show that our proposed policy minimizes a bound on the drift of the quadratic Lyapunov function and compare with the two aforementioned policies: Comparison with the second policy proves strong stability of the system under our proposed policy, while comparison with the first one proves almost optimality.

From Lemma 13 and Corollary 4, it suffices to prove that under the online policy the queues are strongly stable and the resulting time average admission rates maximize the desired utility function subject to minimum rate constraints.

The proof of the performance of our proposed policy is based on applying Lyapunov optimization theory [65] with the following as Lyapunov function (where we have defined $\mathbf{Z}(t) = (\mathbf{S}(t), \mathbf{Q}(t), \mathbf{U}(t))$ to shorten the notation)

$$L(\mathbf{Z}) = L(\mathbf{S}, \mathbf{Q}, \mathbf{U}) = \frac{1}{2}\left(\sum_{k=1}^{K} U_k^2(t) + S_k^2(t) + \sum_{\mathcal{I}\in 2^{\mathcal{K}}} \frac{Q_{\mathcal{I}}^2(t)}{F^2}\right).$$

We then define the drift of the aforementioned Lyapunov function as

$$\Delta L(\mathbf{Z}) = \mathbb{E}\left\{L(\mathbf{Z}(t+1)) - L(\mathbf{Z}(t))|\mathbf{Z}(t) = \mathbf{Z}\right\},$$

where the expectation is over the channel distribution and possible randomizations of the control policy. Using the queue evolution equations (3.10), (3.11), (3.15) and the fact that $([x]^+)^2 \leq x^2$, we have

$$\begin{aligned}\Delta L(\mathbf{Z}(t)) \leq & B \\ & + \sum_{\mathcal{I}\in 2^{\mathcal{K}}} \frac{Q_{\mathcal{I}}(t)}{F^2}\mathbb{E}\left\{\sum_{\mathcal{J}:\mathcal{I}\subseteq\mathcal{J}} b_{\mathcal{I},\mathcal{J}}\sigma_{\mathcal{J}}(t) - T_{\text{slot}}\mu_{\mathcal{I}}(t)\middle|\mathbf{Z}(t)\right\} \\ & + \sum_{k=1}^{K} S_k(t)\mathbb{E}\left\{a_k(t) - \sum_{\mathcal{I}:k\in\mathcal{I}}\sigma_{\mathcal{I}}(t)\middle|\mathbf{Z}(t)\right\} \\ & + \sum_{k=1^K} U_k(t)\mathbb{E}\left\{\gamma_k(t) - a_k(t)|\mathbf{Z}(t)\right\}, \end{aligned} \tag{B.3}$$

can show that for each vector on $\partial\Gamma^{CC}$ we need to achieve a rate vector on the boundary of the capacity region of the wireless channel. Since, as mentioned in the main text, we do not consider boundaries in this work, we can discard these points.

where

$$B = \sum_{k=1}^{K} \left(\gamma_{k,\max}^2 + \frac{1}{2} \left(\sum_{\mathcal{I}:k\in\mathcal{I}} \sigma_{\max} \right)^2 \right) + \frac{1}{2F^2} \sum_{\mathcal{I}\in 2^{\mathcal{K}}} \sum_{\mathcal{J}:\mathcal{I}\subseteq\mathcal{J}} \left(\sigma_{\max} b_{\mathcal{I},\mathcal{J}}\right)^2 + \frac{T_{\text{slot}}^2}{2F^2} \sum_{\mathcal{I}\in 2^{\mathcal{K}}} \sum_{k\in\mathcal{I}} \mathbb{E}\left\{ \left(\log_2(1+Ph_k(t))\right)^2 \right\}. \tag{B.4}$$

Note that B is a finite constant that depends only on the parameters of the system. Adding the quantity $-V\sum_{k=1}^{K}\mathbb{E}\left\{g_k(\gamma_k(t))|\mathbf{Z}(t)\right\}$ to both hands of (B.3) and rearranging the right hand side, we have the drift-plus-penalty expression

$$\begin{aligned} &\Delta L(\mathbf{Z}(t)) - V\sum_{k=1}^{K}\mathbb{E}\left\{g_k(\gamma_k(t))|\mathbf{Z}(t)\right\} \leq \\ &B + \sum_{k=1}^{K}\mathbb{E}\left\{-Vg_k(\gamma_k(t)) + \gamma_k(t)U_k(t)|\mathbf{Z}(t)\right\} \\ &+ \sum_{\mathcal{J}\in 2^{\mathcal{K}}}\mathbb{E}\left\{\sigma_{\mathcal{J}}(t)|\mathbf{Z}(t)\right\}\left(\sum_{\mathcal{I}:\mathcal{I}\subseteq\mathcal{J}}\frac{Q_{\mathcal{I}}(t)}{F^2}b_{\mathcal{I},\mathcal{J}} - \sum_{k:k\in\mathcal{J}}S_k(t)\right) \\ &+ \sum_{k=1}^{K}\left(S_k(t)-U_k(t)\right)\mathbb{E}\left\{a_k(t)|\mathbf{Z}(t)\right\} \\ &- \sum_{\mathcal{J}\in 2^{\mathcal{K}}}\frac{Q_{\mathcal{J}}(t)}{F^2}T_{\text{slot}}\mathbb{E}\left\{\mu_{\mathcal{J}}(t)|\mathbf{Z}(t)\right\} \end{aligned} \tag{B.5}$$

Now observe that the proposed scheme π minimizes the right hand side of (B.5) given any channel state $\mathbf{h}(t)$ (and hence in expectation over the channel state distributions). Therefore, for every vectors $\overline{\mathbf{a}} \in [1,\gamma_{\max}]^K, \overline{\mathbf{fl}} \in [1,\gamma_{\max}]^K, \overline{\boldsymbol{\sigma}} \in Conv(\{0,..,\sigma_{\max}\}^M), \overline{\boldsymbol{\mu}} \in \sum_{\mathbf{h}\in\mathcal{H}}\phi_{\mathbf{h}}\Gamma(\mathbf{h})$ that denote time averages of the control variables achievable by any static (i.e. depending only on the channel state realizations) randomized policies it holds that

$$\begin{aligned} &\Delta L^{\pi}(\mathbf{Z}(t)) - V\sum_{k=1}^{K}\mathbb{E}\left\{g_k(\gamma_k^{\pi}(t))\right\} \leq \\ &B - V\sum_{k=1}^{K}g_k(\overline{\gamma}_k) + \sum_{k=1}^{K}U_k(t)\left(\overline{\gamma}_k - \overline{a}_k\right) \\ &+ \sum_{k=1}^{K}S_k(t)\left(\overline{a}_k - \sum_{\mathcal{J}:k\in\mathcal{J}}\overline{\sigma}_{\mathcal{J}}\right) \\ &+ \sum_{\mathcal{J}}\frac{Q_{\mathcal{J}}(t)}{F^2}\left(\sum_{\mathcal{I}:\mathcal{J}\subseteq\mathcal{I}}b_{\mathcal{J},\mathcal{I}}\overline{\sigma}_{\mathcal{I}} - T_{\text{slot}}\overline{\mu}_{\mathcal{J}}\right) \end{aligned} \tag{B.6}$$

We will use (B.6) to compare our policy with the specific static policies defined in Lemmas 19, 20.

Proof of strong stability: Replacing the time averages we get from the static stabilizing policy π^{δ} of Lemma 20 for some $\delta > 0$, we get that there exist $\epsilon, \epsilon' > 0$ such that (the superscript π denotes the quantities under our proposed policy)

$$\begin{aligned}\Delta L^{\pi}(\mathbf{Z}(t)) \leq B + V\sum_{k=1}^{K}\mathbb{E}\left\{g_k(a_k^{\pi}(t))\right\} - V\sum_{k=1}^{K} g_k(\overline{a}_k^{\delta}) \\ - \epsilon\left(\sum_{k=1}^{K} S_k(t) + \sum_{\mathcal{J}\in 2^{\mathcal{K}}}\frac{Q_{\mathcal{J}}(t)}{F^2}\right) \\ - \epsilon'\sum_{k=1}^{K} U_k(t) \end{aligned} \tag{B.7}$$

Since $a_k(t) \leq \gamma_{\max,k}\forall t$, it follows that $g_k(\overline{a}_k^{\pi}) < g_k(\gamma_{\max,k})$. In addition, $g_k(x) \geq 0, \forall x \geq 0$ therefore

$$\begin{aligned}\Delta L^{\pi}(\mathbf{Z}(t)) \leq B + V\sum_{k=1}^{K}\mathbb{E}\left\{g_k(\gamma_{\max,k})\right\} \\ - \epsilon\left(\sum_{k=1}^{K} S_k(t) + \sum_{\mathcal{J}\in 2^{\mathcal{K}}}\frac{Q_{\mathcal{J}}(t)}{F^2}\right) \\ - \epsilon'\sum_{k=1}^{K} U_k(t) \end{aligned} \tag{B.8}$$

Using Lemma 3, we obtain

$$\begin{aligned}\limsup_{T\to\infty}\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}\left\{\sum_{\mathcal{J}\in 2^{\mathcal{K}}}\frac{Q_{\mathcal{J}}(t)}{F^2} + \sum_{k=1}^{K}(S_k(t) + U_k(t))\right\} \\ \leq \frac{B + V\sum_{k=1}^{K} g_k(\gamma_{\max,k})}{\epsilon}. \end{aligned} \tag{B.9}$$

Therefore the queues are strongly stable under our proposed policy. In order to prove the part of Theorem 16 regarding the guaranteed bound on the average queue lengths, we first note that the above inequality holds for every $\epsilon > 0$ and define ϵ_0 as

$$\epsilon_0 = \operatorname*{argmax}_{\epsilon>0}\epsilon \tag{B.10}$$

$$\text{s.t. } \epsilon\mathbf{1} \in \Lambda^{CC}. \tag{B.11}$$

Following the same arguments as in Section IV of [61], we can show that the Right Hand Side of (B.9) is bounded from below by

$$\frac{B + V\sum_{k=1}^{K} g_k(\gamma_{\max,k})}{\epsilon_0},$$

therefore proving the requested bound on the long-term average queue lengths.

We now proceed to proving the near-optimality of our proposed policy.

Proof of near optimal utility: Here we compare π with the static optimal policy π^* from Lemma 19. Since π^* takes decisions irrespectively of the queue lengths, we can replace quantities $\overline{\mathbf{a}}, \overline{\boldsymbol{\sigma}}, \overline{\boldsymbol{\mu}}$ on (B.6) with the time averages corresponding to π^*, i.e. $\overline{\mathbf{a}}^*, \overline{\boldsymbol{\sigma}}^*, \overline{\boldsymbol{\mu}}^*$. From the inequalities in Lemma 19 we have

$$V\sum_{k=1}^{K}\mathbb{E}\left\{g_k(\gamma_k^{\pi}(t))\right\} \geq V\sum_{k=1}^{K}g_k(\overline{a}_k^*) - B + \Delta L^{\pi}(\mathbf{Z}(t))$$

Taking expectations over $\mathbf{Z}(t)$ for both sides and summing the inequalities for $t = 0, 1, .., T-1$ and dividing by VT we get

$$\frac{1}{T}\sum_{t=1}^{T-1}\sum_{k=1}^{K}\mathbb{E}\left\{g_k(\gamma_k^{\pi}(t))\right\} \geq \sum_{k=1}^{K}g_k(\overline{a}_k^*) - \frac{B}{V} - \frac{\mathbb{E}\left\{L^{\pi}(\mathbf{Z}(0))\right\}}{VT} + \frac{\mathbb{E}\left\{L^{\pi}(\mathbf{Z}(T))\right\}}{VT}$$

Assuming $\mathbb{E}\left\{L^{\pi}(\mathbf{Z}(0))\right\} < \infty$ (this assumption is standard, for example it holds if the system starts empty), since $\mathbb{E}\{L^{\pi}(\mathbf{Z}(T))\} > 0, \forall T > 0$, taking the limit as T goes to infinity gives

$$\lim_{T\to\infty}\frac{1}{T}\sum_{t=1}^{T-1}\sum_{k=1}^{K}\mathbb{E}\left\{g_k(\gamma_k^{\pi}(t))\right\} \geq \sum_{k=1}^{K}g_k(\overline{a}_k^*) - \frac{B}{V}$$

In addition, since $g_k(x)$ are concave, Jensen's inequality implies

$$\begin{aligned}\sum_{k=1}^{K}g_k(\overline{\gamma}_k^{\pi}) &= \sum_{k=1}^{K}g_k\left(\lim_{T\to\infty}\frac{1}{T}\sum_{t=0}^{T}\mathbb{E}\{\gamma_k^{\pi}(t)\}\right)\\ &\geq \lim_{T\to\infty}\frac{1}{T}\sum_{t=1}^{T-1}\sum_{k=1}^{K}\mathbb{E}\left\{g_k(\gamma_k^{\pi}(t))\right\}\\ &\geq \sum_{k=1}^{K}g_k(\overline{a}_k^*) - \frac{B}{V}.\end{aligned}$$

Finally, since the virtual queues $U_k(t)$ are strongly stable, it holds $\overline{a}_k^{\pi} > \overline{\gamma}_k^{\pi}$. We then have

$$\sum_{k=1}^{K}g_k(\overline{a}_k^{\pi}) > \sum_{k=1}^{K}g_k(\overline{\gamma}_k^{\pi}) \geq \sum_{k=1}^{K}g_k(\overline{a}_k^*) - \frac{B}{V},$$

which proves the near optimality of our proposed policy π.

Appendix C

Opportunistic Scheduling

C.1 Proof of Theorem 17

C.1.1 Proof element 1: lower bound on the rates

The first step towards proving Theorem 17 is to show that the rates allocated by α-fair scheduling are upper and lower bounded by two constants, so that $\min_i 1/(\bar{r}_i^{\pi^\star})^\alpha$ and $\max_i 1/(\bar{r}_i^{\pi^\star})^\alpha$ are of the same order even as $K \to \infty$. This is in fact the step of the proof which is the most involved.

Proposition 6. *There exists* $0 < C_1(\underline{\rho}, \overline{\rho}) < C_2(\underline{\rho}, \overline{\rho}) < \infty$ *such that for all* $K \geq 0$ *and all* $i = 1, \ldots, K$*:*

$$C_1(\underline{\rho}, \overline{\rho}) \leq \bar{r}_i^{\pi^\star} \leq C_2(\underline{\rho}, \overline{\rho}).$$

Proof. Without loss of generality, we may order users to ensure $\bar{r}_1^{\pi^\star} \leq ... \leq \bar{r}_K^{\pi^\star}$. Throughout the proof we consider the optimal policy $\pi^\star$ and, to ease notation, we denote $\bar{r}_i^{\pi^\star}$ by $\bar{r}_i$. We define the function:

$$f(\mathcal{J}, \boldsymbol{h}) = \frac{1}{T(m, |\mathcal{J}|)} \log(1 + \min_{j \in \mathcal{J}} h_j) \sum_{i=1}^{K} \frac{\mathbf{1}\{i \in \mathcal{J}\}}{(\bar{r}_i)^\alpha}. \tag{C.1}$$

As shown in corollary 5, under the optimal policy $\pi^\star$, the chosen group is

$$\mathcal{J}^\star(\boldsymbol{h}) \in \arg \max_{\mathcal{J} \subset \{1,\ldots,K\}} f(\mathcal{J}, \boldsymbol{h}).$$

As a first step, we control the chosen group $\mathcal{J}^\star$, in an alternative system when user 1 is ignored. We define $\mathcal{J}_1^\star(\boldsymbol{h}) \in \arg\max_{\mathcal{J} \subset \{2,\ldots,K\}} f(\mathcal{J}, \boldsymbol{h})$, the maximizer of f if user 1 is ignored. Denote by $\bar{r} = \sum_{i=2}^{K} \frac{1}{(\bar{r}_i)^\alpha}$, the sum of weights of all users except user 1. Define $z = \frac{\bar{r}}{2T(m,\infty)} \log(1 + \underline{\rho} \log 2)$. We now prove the following inequality:

$$\mathbb{P}\left(f(\mathcal{J}_1^\star(\boldsymbol{h}), \boldsymbol{h}) \leq z\right) \leq \frac{1}{2}.$$

Define the group: $\mathcal{J}_1(\boldsymbol{h}) = \{i \geq 2 : h_i \geq \rho_i \log 2\}$. Let us lower bound $f(\mathcal{J}_1(\boldsymbol{h}), \boldsymbol{h})$. By definition, $j \in \mathcal{J}_1(\boldsymbol{h})$ implies $h_j \geq \rho_j \log 2 \geq \underline{\rho} \log 2$, hence:

$$\log(1 + \underline{\rho} \log 2) \leq \log(1 + \min_{j \in \mathcal{J}_1(\boldsymbol{h})} h_j),$$

and further using Property 2 implying $T(m, \infty) > T(m, \mathcal{J}_1(\boldsymbol{h}))$, we obtain the lower bound:

$$\frac{1}{T(m, \infty)} \log(1 + \underline{\rho} \log 2) \sum_{i=2}^{K} \frac{\mathbf{1}\{h_i \geq \rho_i \log 2\}}{(\bar{r}_i)^\alpha} \leq f(\mathcal{J}_1(\boldsymbol{h}), \boldsymbol{h}).$$

Define the random variable:

$$Z = \frac{1}{T(m, \infty)} \log(1 + \underline{\rho} \log 2) \sum_{i=2}^{K} \frac{\mathbf{1}\{h_i \geq \rho_i \log 2\}}{(\bar{r}_i)^\alpha}.$$

By definition of $\mathcal{J}_1^\star(\boldsymbol{h})$, we have $f(\mathcal{J}_1^\star(\boldsymbol{h}), \boldsymbol{h}) \geq f(\mathcal{J}_1(\boldsymbol{h}), \boldsymbol{h})$, so that:

$$Z \leq f(\mathcal{J}_1(\boldsymbol{h}), \boldsymbol{h}) \leq f(\mathcal{J}_1^\star(\boldsymbol{h}), \boldsymbol{h}).$$

Since h_i follows an exponential distribution with mean ρ_i, we have $\mathbb{P}(h_i \geq \rho_i \log 2) = \frac{1}{2}$ and since the channel realizations are independent across users, the random variables $\mathbf{1}\{h_i \geq \rho_i \log 2\}$ and $\mathbf{1}\{h_{i'} \geq \rho_{i'} \log 2\}$ are independent whenever $i \neq i'$. Therefore:

$$\mathbb{E}(Z) = \frac{1}{T(m, \infty)} \log(1 + \underline{\rho} \log 2) \sum_{i=2}^{K} \frac{\mathbb{P}(h_i \geq \rho_i \log 2)}{(\bar{r}_i)^\alpha} = z,$$

and Z is a weighted sum of Bernoulli independent random variables with mean $\frac{1}{2}$ so that Z is symmetrical, i.e. $Z - z$ has the same distribution as $z - Z$. Therefore: $\mathbb{P}(Z \leq z) = \mathbb{P}(Z \geq z) = \frac{1}{2}$ and:

$$\mathbb{P}\left(f(\mathcal{J}_1^\star(\boldsymbol{h}), \boldsymbol{h}) \leq z\right) \leq \mathbb{P}(Z \leq z) = \frac{1}{2}.$$

We now control the value of $\min_{i \in \mathcal{J}_1^\star(\boldsymbol{h})} h_i$. Choose any c_1, c_2 such that both of the conditions below are satisfied:

$$\text{(i)} \;\; \log(1 + c_1) < \frac{T(m, 1)}{2T(m, \infty)} \log(1 + \underline{\rho} \log 2); \text{ and}$$
$$\text{(ii)} \;\; \frac{2T(m, \infty) \int_{c_2}^{\infty} (\log(1 + y)/\overline{\rho}) e^{-y/\overline{\rho}} dy}{T(m, 1) \log(1 + \underline{\rho} \log 2)} \leq \frac{1}{4}.$$

It is noted that we may indeed choose c_1, c_2 in that way since $c \mapsto \log(1 + c)$ is increasing and vanishes for $c = 0$, and since $c \mapsto \int_c^{\infty} (\log(1 + y)/\overline{\rho}) e^{-y/\overline{\rho}} dy$ is decreasing and vanishes for $c \to \infty$. It is also noted that c_1 c_2 may be chosen only based on the value of $\underline{\rho}$ and $\overline{\rho}$ and m.

Assume that $\min_{i\in\mathcal{J}_1^\star(\boldsymbol{h})} h_i \leq c_1$ and that $f(\mathcal{J}_1^\star(\boldsymbol{h}),\boldsymbol{h}) \geq z$. If this event occurs, using the facts that (a) $\log(1+\min_{i\in\mathcal{J}_1^\star(\boldsymbol{h})} h_i) \leq \log(1+c_1)$, and (b) $T(m,|\mathcal{J}_1^\star(\boldsymbol{h})|) \geq T(m,1)$ since Property 2, and (c) that $\sum_{i=1}^K \frac{\mathbf{1}\{i\in\mathcal{J}_1^\star(\boldsymbol{h})\}}{(\bar{r}_i)^\alpha} \leq \bar{r}$, we obtain the upper bound:

$$f(\mathcal{J}_1^\star(\boldsymbol{h}),\boldsymbol{h}) \leq \frac{\bar{r}}{T(m,1)}\log(1+c_1).$$

In summary, if $\min_{i\in\mathcal{J}_1^\star(\boldsymbol{h})} h_i \leq c_1$ and $f(\mathcal{J}_1^\star(\boldsymbol{h}),\boldsymbol{h}) \geq z$ we have $z \leq \frac{\bar{r}}{T(m,1)}\log(1+c_1)$ and replacing z with its definition:

$$\frac{\bar{r}}{2T(m,\infty)}\log(1+\underline{\rho}\log 2) \leq \frac{\bar{r}}{T(m,1)}\log(1+c_1),$$

which is equivalent to

$$\frac{T(m,1)}{2T(m,\infty)}\log(1+\underline{\rho}\log 2) \leq \log(1+c_1),$$

a contradiction with (i) the definition of c_1. We have hence proven that $f(\mathcal{J}_1^\star(\boldsymbol{h}),\boldsymbol{h}) \geq z$ implies $\min_{i\in\mathcal{J}_1^\star(\boldsymbol{h})} h_i \geq c_1$.

Now assume that $\min_{i\in\mathcal{J}_1^\star(\boldsymbol{h})} h_i \geq c_2$ and that $f(\mathcal{J}_1^\star(\boldsymbol{h}),\boldsymbol{h}) \geq z$. If this event occurs, using the facts that

$$\begin{aligned}\text{(a)}\quad \log(1+\min_{j\in\mathcal{J}_1^\star(\boldsymbol{h})} h_j)\mathbf{1}\{i\in\mathcal{J}_1^\star(\boldsymbol{h})\} &\leq \log(1+h_i)\mathbf{1}\{i\in\mathcal{J}_1^\star(\boldsymbol{h})\}\\ &\leq \log(1+h_i)\mathbf{1}\{h_i\geq c_2\},\end{aligned}$$

since $i\in\mathcal{J}_1^\star(\boldsymbol{h})$ implies $h_i \geq c_2$, and (b) $T(m,|\mathcal{J}_1^\star(\boldsymbol{h})|) \geq T(m,1)$ since Property 2, we obtain the upper bound:

$$f(\mathcal{J}_1^\star(\boldsymbol{h}),\boldsymbol{h}) \leq \frac{1}{T(m,1)}\sum_{i\geq 2}\frac{\log(1+h_i)\mathbf{1}\{h_i\geq c_2\}}{(\bar{r}_i)^\alpha} \equiv Z'.$$

In summary $\min_{i\in\mathcal{J}_1^\star(\boldsymbol{h})} h_i \geq c_2$ and $f(\mathcal{J}_1^\star(\boldsymbol{h}),\boldsymbol{h}) \geq z$ implies $z \leq f(\mathcal{J}_1^\star(\boldsymbol{h}),\boldsymbol{h}) \leq Z'$. Let us upper bound the expectation of Z'. Since h_i has exponential distribution with mean ρ_i we have:

$$\begin{aligned}\mathbb{E}(\log(1+h_i)\mathbf{1}\{h_i\geq c_2\}) &= \int_{c_2}^\infty (\log(1+y)/\rho_i)e^{-y/\rho_i}dy\\ &\leq \int_{c_2}^\infty (\log(1+y)/\overline{\rho})e^{-y/\overline{\rho}}dy.\end{aligned}$$

Hence:

$$\mathbb{E}(Z') \leq \frac{\bar{r}\int_{c_2}^\infty(\log(1+y)/\overline{\rho})e^{-y/\overline{\rho}}dy}{T(m,1)}.$$

Using Markov's inequality, we get:

$$\begin{aligned}\mathbb{P}\left(Z' \geq z\right) &\leq \frac{\mathbb{E}(Z')}{z} \\ &\leq \frac{2T(m,\infty)\int_{c_2}^{\infty}(\log(1+y)/\overline{\rho})e^{-y/\overline{\rho}}dy}{T(m,1)\log(1+\underline{\rho}\log 2)} \\ &\leq \frac{1}{4},\end{aligned}$$

using the definition of c_2 for the final inequality.

In conclusion, we have proven that:

$$\begin{aligned}\mathbb{P}\left(\min_{i\in\mathcal{J}_1^\star(\boldsymbol{h})} h_i \notin [c_1,c_2]\right) &= \mathbb{P}\left(\min_{i\in\mathcal{J}_1^\star(\boldsymbol{h})} h_i \notin [c_1,c_2]; f(\mathcal{J}_1^\star(\boldsymbol{h}),\boldsymbol{h}) \leq z\right) \\ &+ \mathbb{P}\left(\min_{i\in\mathcal{J}_1^\star(\boldsymbol{h})} h_i \notin [c_1,c_2]; f(\mathcal{J}_1^\star(\boldsymbol{h}),\boldsymbol{h}) \geq z\right) \\ &\leq \mathbb{P}\left(f(\mathcal{J}_1^\star(\boldsymbol{h}),\boldsymbol{h}) \leq z\right) + \mathbb{P}\left(\min_{i\in\mathcal{J}_1^\star(\boldsymbol{h})} h_i \notin [c_1,c_2]; f(\mathcal{J}_1^\star(\boldsymbol{h}),\boldsymbol{h}) \geq z\right) \\ &= \mathbb{P}\left(f(\mathcal{J}_1^\star(\boldsymbol{h}),\boldsymbol{h}) \leq z\right) + \mathbb{P}\left(\min_{i\in\mathcal{J}_1^\star(\boldsymbol{h})} h_i \geq c_2; f(\mathcal{J}_1^\star(\boldsymbol{h}),\boldsymbol{h}) \geq z\right) \\ &\leq \mathbb{P}\left(Z \leq z\right) + \mathbb{P}\left(Z' \geq z\right) \\ &\leq \frac{1}{2} + \frac{1}{4} = \frac{3}{4},\end{aligned}$$

hence:

$$\mathbb{P}\left(\min_{i\in\mathcal{J}_1^\star(\boldsymbol{h})} h_i \in [c_1,c_2]\right) \geq \frac{1}{4}.$$

The second step involves lower bounding $\bar{r}_1$, using the previous result on the fluctuations of $\min_{i\in\mathcal{J}_1^\star(\boldsymbol{h})} h_i$. We will use the four following facts: (a) Since $\mathcal{J}_1^\star(\boldsymbol{h})$ depends solely on $h_2,...,h_K$, the event $\min_{i\in\mathcal{J}_1^\star(\boldsymbol{h})} h_i \in [c_1,c_2]$ is independent of h_1, (b) When both $\min_{i\in\mathcal{J}_1^\star(\boldsymbol{h})} h_i \in [c_1,c_2]$, and $h_1 > c_2$, then $1 \in \mathcal{J}^\star(\boldsymbol{h})$ since $\frac{1}{(\bar{r}_1)^\alpha} \geq \max_{i\geq 2}\frac{1}{(\bar{r}_i)^\alpha}$ and $\min_{i\in\mathcal{J}^\star(\boldsymbol{h})} h_i \leq c_2 \leq h_1$. Indeed, if $1 \notin \mathcal{J}^\star(\boldsymbol{h})$, for any $i \in \mathcal{J}^\star(\boldsymbol{h})$ we have $f(\mathcal{J}^\star(\boldsymbol{h}) \setminus \{i\} \cup \{1\},\boldsymbol{h}) > f(\mathcal{J}^\star(\boldsymbol{h}),\boldsymbol{h})$, a contradiction since $\mathcal{J}^\star(\boldsymbol{h})$ is a maximizer of $\mathcal{J} \mapsto f(\mathcal{J},\boldsymbol{h})$, (c) Since h_1 has exponential distribution with mean $\rho_1 \geq \underline{\rho}$, $\mathbb{P}(h_1 \geq c_2) = e^{-c_2/\rho_1} \geq e^{-c_2/\underline{\rho}}$ and (d) We have $T(m,|\mathcal{J}^\star(\boldsymbol{h})|) \leq T(m,\infty)$ since Property 2.

Putting (a), (b), (c) and (d) together we get:

$$\begin{aligned}\bar{r}_1 &\geq \frac{1}{T(m,\infty)}\log(1+c_1)\mathbb{P}(\min_{i\in\mathcal{J}_1^\star(\boldsymbol{h})} h_i \in [c_1,c_2], h_1 \geq c_2) \\ &= \frac{1}{T(m,\infty)}\log(1+c_1)\mathbb{P}(\min_{i\in\mathcal{J}_1^\star(\boldsymbol{h})} h_i \in [c_1,c_2])\mathbb{P}(h_1 \geq c_2) \\ &\geq \frac{1}{T(m,\infty)}\frac{1}{4}\log(1+c_1)e^{-c_2/\underline{\rho}} \equiv C_1(\underline{\rho},\overline{\rho}).\end{aligned}$$

Furthermore, for any $i = 1, ..., K$:

$$\begin{aligned}\bar{r}_i &\leq \frac{1}{T(m,1)}\mathbb{E}(\log(1+h_i)) \\ &\leq \frac{1}{T(m,1)}\log(1+\mathbb{E}(h_i)) \\ &= \frac{1}{T(m,1)}\log(1+\rho_i) \\ &\leq \frac{1}{T(m,1)}\log(1+\overline{\rho}) \equiv C_2(\underline{\rho},\overline{\rho}).\end{aligned}$$

We have proven that:

$$C_1(\underline{\rho},\overline{\rho}) \leq \bar{r}_i \leq C_2(\underline{\rho},\overline{\rho})$$

for all $i = 1, ..., K$ and all K as announced. □

C.1.2 Proof element 2: asymptotic size of $\mathcal{J}$

From the first proof element we deduce the second one, that is, only groups $\mathcal{J}^\star(\boldsymbol{h})$ of large size are chosen with high probability as the number of users grows. In turn this implies that $T(m, |\mathcal{J}^\star(\boldsymbol{h})|) \underset{K\to\infty}{\overset{\mathbb{P}}{\to}} T(m,\infty)$. This result is important, since it allows to take $T(m, |\mathcal{J}^\star(\boldsymbol{h})|)$ out of the equation when it comes to controlling which users are selected by the optimal policy.

Proposition 7. *For all $J \geq 0$ we have:*

$$\mathbb{P}(|\mathcal{J}^\star(\boldsymbol{h})| \geq J) \underset{K\to\infty}{\to} 1.$$

Furthermore, $T(m, |\mathcal{J}^\star(\boldsymbol{h})|) \underset{K\to\infty}{\overset{\mathbb{P}}{\to}} T(m,\infty)$.

Proof. Consider the following group of users:

$$\mathcal{J}(\boldsymbol{h}) = \{i \geq 1 : h_i \geq \rho_i \log 2\}.$$

Let us lower bound the value of $f(\mathcal{J}(\boldsymbol{h}), \boldsymbol{h}) = \frac{1}{T(m,|\mathcal{J}|)}\log(1+\min_{j\in\mathcal{J}} h_j)\sum_{i=1}^K \frac{\mathbf{1}\{i\in\mathcal{J}\}}{(\bar{r}_i)^\alpha}$ as defined in (C.1). Using the facts that (a) $T(m,\mathcal{J}(\boldsymbol{h})) \leq T(m,\infty)$ due to Property 2, (b) $i \in \mathcal{J}(\boldsymbol{h})$ implies $h_i \geq \rho_i \log 2 \geq \underline{\rho}\log 2$ so that $\min_{i\in\mathcal{J}(\boldsymbol{h})} h_i \geq \underline{\rho}\log 2$ and (c) $\bar{r}_i \leq C_2(\underline{\rho},\overline{\rho})$ so that $\frac{1}{(\bar{r}_i)^\alpha} \geq \frac{1}{C_2(\underline{\rho},\overline{\rho})^\alpha}$ we obtain the lower bound:

$$\frac{\log(1+\underline{\rho}\log 2)}{C_2(\underline{\rho},\overline{\rho})^\alpha T(m,\infty)}\sum_{i=1}^K \mathbf{1}\{h_i \geq \rho_i \log 2\} \leq f(\mathcal{J}(\boldsymbol{h}), \boldsymbol{h}).$$

Let us upper bound the value of $f(\mathcal{J}^\star(\boldsymbol{h}),\boldsymbol{h})$, using the facts that (a) $T(m,\mathcal{J}(\boldsymbol{h}))\geq T(m,1)$ due to Property 1, (b) $\bar{r}_i \geq C_1(\underline{\rho},\overline{\rho})$ so that $\frac{1}{(\bar{r}_i)^\alpha} \leq \frac{1}{C_1(\underline{\rho},\overline{\rho})^\alpha}$ and (c) $\min_{i\in\mathcal{J}^\star(\boldsymbol{h})} h_i \leq \max_{i=1,\dots,K} h_i \leq \overline{\rho}\max_{i=1,\dots,K}(h_i/\rho_i)$ we obtain:

$$f(\mathcal{J}^\star(\boldsymbol{h}),\boldsymbol{h}) \leq |\mathcal{J}^\star(\boldsymbol{h})|\frac{\log(1+\overline{\rho}\max_{i=1,\dots,K}(h_i/\rho_i))}{C_1(\underline{\rho},\overline{\rho})^\alpha T(m,1)}.$$

Since $\mathcal{J}^\star(\boldsymbol{h})$ is a maximizer of $\mathcal{J}\mapsto f(\mathcal{J},\boldsymbol{h})$ we have $f(\mathcal{J}(\boldsymbol{h}),\boldsymbol{h})\leq f(\mathcal{J}^\star(\boldsymbol{h}),\boldsymbol{h})$, and the two previous inequalities imply:

$$\log(1+\underline{\rho}\log 2)\frac{T(m,1)}{T(m,\infty)}\left(\frac{C_1(\underline{\rho},\overline{\rho})}{C_2(\underline{\rho},\overline{\rho})}\right)^\alpha \frac{\sum_{i=1}^K \mathbf{1}\{h_i\geq\rho_i\log 2\}}{\log(1+\overline{\rho}\max_{i=1,\dots,K}(h_i/\rho_i))} \leq |\mathcal{J}^\star(\boldsymbol{h})|.$$

To finish the proof, we prove that:

$$\frac{\sum_{i=1}^K \mathbf{1}\{h_i\geq\rho_i\log 2\}}{\log(1+\overline{\rho}\max_{i=1,\dots,K}(h_i/\rho_i))} \underset{K\to\infty}{\overset{a.s.}{\to}} \infty.$$

Since $h_1/\rho_1,\dots,h_K/\rho_K$ are i.i.d exponentially distributed with mean 1, we have $\mathbb{P}(h_i\geq\rho_i\log 2)=\frac{1}{2}$ and the law of large numbers gives:

$$\frac{1}{K}\sum_{i=1}^K \mathbf{1}\{h_i\geq\rho_i\log 2\} \underset{K\to\infty}{\overset{\mathbb{P}}{\to}} \frac{1}{2}.$$

Since $\frac{1}{4}<\frac{1}{2}$, we have for $K\to\infty$, with high probability,

$$\sum_{i=1}^K \mathbf{1}\{h_i\geq\rho_i\log 2\} \geq \frac{K}{4}.$$

Furthermore,

$$\begin{aligned}\mathbb{P}(\max_{i=1,\dots,K}(h_i/\rho_i)\geq 2\log K) &= 1-\mathbb{P}(\max_{i=1,\dots,K}(h_i/\rho_i)\leq 2\log K)\\ &= 1-\prod_{i=1}^K \mathbb{P}(h_i/\rho_i\leq 2\log K)\\ &= 1-\left(1-\frac{1}{K^2}\right)^K \underset{K\to\infty}{\to} 0.\end{aligned}$$

Thus for $K\to\infty$, with high probability, we have

$$\max_{i=1,\dots,K}(h_i/\rho_i)\leq 2\log K.$$

Hence, the following occurs with high probability:

$$\log(1+\underline{\rho}\log 2)\frac{T(m,1)}{T(m,\infty)}\left(\frac{C_1(\underline{\rho},\overline{\rho})}{C_2(\underline{\rho},\overline{\rho})}\right)^\alpha \frac{K}{\log(1+2\overline{\rho}\log K)} \leq |\mathcal{J}^\star(\boldsymbol{h})|.$$

Since $\frac{K}{\log\log K} \underset{K\to\infty}{\to} \infty$, this implies that, for all $J \geq 0$:

$$\mathbb{P}(|\mathcal{J}^\star(\boldsymbol{h})| \geq J) \underset{K\to\infty}{\to} 1.$$

Therefore, for any $J \geq 0$:

$$\mathbb{P}\left(T(m,J) \leq T(m,|\mathcal{J}^\star(\boldsymbol{h})|) \leq T(m,\infty)\right) \underset{K\to\infty}{\to} 1.$$

This holds for all J, which proves the second statement. □

C.1.3 Proof element 3: convergence to a deterministic equivalent

The last proof element is to show that, when $K \to \infty$, maximizing $f(\mathcal{J}, \boldsymbol{h})$ reduces to a simpler, deterministic optimization problem, which we call a "deterministic equivalent" of the original problem. Define the following mapping:

$$\phi(\mathcal{J}, \boldsymbol{h}) = \log(1 + \min_{j\in\mathcal{J}} h_j)\frac{1}{K}\sum_{i=1}^{K} \frac{\mathbf{1}\{i \in \mathcal{J}\}}{(\bar{r}_i)^\alpha},$$

which corresponds to the value of $\frac{T(m,\infty)}{K} f(\mathcal{J}, \boldsymbol{h})$ when $|\mathcal{J}|$ goes to infinity. Further define ψ:

$$\psi(c, \boldsymbol{h}) = \log(1 + c)\frac{1}{K}\sum_{i=1}^{K} \frac{\mathbf{1}\{h_i \geq c\}}{(\bar{r}_i)^\alpha},$$

which is the value of ϕ when selecting only users whose channel realization is larger than c. It is noted that when $K \to \infty$, we have

$$\max_{\mathcal{J}\subset\{1,\ldots,K\}} \phi(\mathcal{J}, \boldsymbol{h}) = \max_{c\geq 0} \psi(c, \boldsymbol{h}).$$

Indeed, if $\min_{j\in\mathcal{J}} h_j = c$ for some c, then all users i such that $h_i \geq c$ should be included in $\mathcal{J}$ in order to maximize $\phi(\mathcal{J}, \boldsymbol{h})$. Hence maximizing $\phi(\mathcal{J}, \boldsymbol{h})$ over all subsets of users $\mathcal{J}$ reduces to a simple, one-dimensonnal search over the value of $\min_{j\in\mathcal{J}} h_j = c$, that is maximizing $\psi(c, \boldsymbol{h})$ over $c \geq 0$. We are now left to control the value of the random quantity $\max_{c\geq 0} \psi(c, \boldsymbol{h})$, which is not straightforward since its maximizer $\arg\max_{c\geq 0} \psi(c, \boldsymbol{h})$ is typically a random variable as well. For a fixed value of c, we define $\Psi(c)$ which is the expected value of $\psi(c, \boldsymbol{h})$:

$$\Psi(c) = \mathbb{E}(\psi(c, \boldsymbol{h})) = \log(1 + c)\frac{1}{K}\sum_{i=1}^{K} \frac{e^{-c/\rho_i}}{(\bar{r}_i)^\alpha}.$$

We will show that Ψ constitutes a *deterministic equivalent*, in the sense that maximizing $\psi(c, \boldsymbol{h})$ over $c \geq 0$ for a fixed value of $\boldsymbol{h}$ yields, asmptotically with high probability, the same outcome as maximizing $\Psi(c)$ over $c \geq 0$. In other words, a concentration phenomenon occurs as the number of users grows large and channel opportunism does yield any gains over choosing all users whose channel realization is above a fixed threshold.

Proposition 8. *We have:*

$$\max_{c\geq 0}\psi(c,\boldsymbol{h}) \underset{K\to\infty}{\overset{\mathbb{P}}{\to}} \max_{c\geq 0}\Psi(c).$$

Proof. We first show that, for any fixed c, $\psi(c,\boldsymbol{h})$ is concentrated around $\Psi(c)$ when $K\to\infty$. Since (a) the channel realizations $h_1,...,h_K$ are independent across users, and (b) $\text{var}(\mathbf{1}\{h_i\geq c\})\leq 1$, and (c) $\bar{r}_i\geq C_1(\underline{\rho},\overline{\rho})$ for $i=1,...,K$, we have:

$$\begin{aligned}\text{var}(\psi(c,\boldsymbol{h})) &= \frac{\log(1+c)^2}{K^2}\sum_{i=1}^{K}\frac{\text{var}(\mathbf{1}\{h_i\geq c\})}{(\bar{r}_i)^{2\alpha}}\\ &\leq \frac{\log(1+c)^2}{KC_1(\underline{\rho},\overline{\rho})^{2\alpha}} \underset{K\to\infty}{\to} 0.\end{aligned}$$

Hence, Chebychev's inequality proves that

$$\psi(c,\boldsymbol{h}) \underset{K\to\infty}{\overset{\mathbb{P}}{\to}} \mathbb{E}(\psi(c,\boldsymbol{h})) = \Psi(c).$$

We may now lower bound $\max_{c\geq 0}\psi(c,\boldsymbol{h})$ as follows. Consider $\tilde{c}\in\arg\max_{c\geq 0}\Psi(c)$, then we have $\psi(\tilde{c},\boldsymbol{h})\leq\max_{c\geq 0}\psi(c,\boldsymbol{h})$ and since $\psi(\tilde{c},\boldsymbol{h}) \underset{K\to\infty}{\overset{\mathbb{P}}{\to}} \Psi(\tilde{c}) = \max_{c\geq 0}\Psi(c)$, this proves that, for all $\epsilon>0$:

$$\mathbb{P}\left(\max_{c\geq 0}\Psi(c)-\epsilon\leq\max_{c\geq 0}\psi(c,\boldsymbol{h})\right)\underset{K\to\infty}{\to}1.$$

We now upper bound $\max_{c\geq 0}\psi(c,\boldsymbol{h})$. We do so by splitting $[0,+\infty)$ into a finite number of intervals and control the behaviour of $c\mapsto\psi(c,\boldsymbol{h})$ in those intervals. Consider $\epsilon>0$ fixed. Define $\delta>0$, and $L\geq 0$ such that both of the following conditions are satisfied:

$$\begin{aligned}&\text{(i)}\ \frac{1}{C_1(\underline{\rho},\overline{\rho})^{\alpha}}\int_{L\delta}^{\infty}(\log(1+y)/\overline{\rho})e^{-y/\overline{\rho}}dy\leq\frac{\epsilon}{2},\\ &\text{(ii)}\frac{\delta}{C_1(\underline{\rho},\overline{\rho})^{\alpha}}\leq\frac{\epsilon}{2}.\end{aligned}$$

Such a choice is always possible since $\int_{L\delta}^{\infty}(\log(1+y)/\overline{\rho})e^{-y/\overline{\rho}}dy$ vanishes for $L\delta\to\infty$. Further define:

$$m_\ell=\begin{cases}\max_{c\in[(\ell-1)\delta,\ell\delta]}\psi(c,\boldsymbol{h}) & \text{if } \ell=1,...,L\\ \max_{c\in[L\delta,+\infty)}\psi(c,\boldsymbol{h}) & \text{if } \ell=L+1.\end{cases}$$

It is noted that $m_1,...,m_{L+1}$ are random variables and that:

$$\max_{c\geq 0}\psi(c,\boldsymbol{h})=\max_{\ell=1,...,L+1}m_\ell.$$

We may now upper bound the value of each m_ℓ individually. First consider $c \in [(\ell-1)\delta, \ell\delta]$, then we have:

$$\begin{aligned}
\psi(c, \boldsymbol{h}) &\leq \log(1+\ell\delta)\frac{1}{K}\sum_{i=1}^{K}\frac{\mathbf{1}\{h_i \geq (\ell-1)\delta\}}{(\bar{r}_i)^\alpha}. \\
&= \psi((\ell-1)\delta, \boldsymbol{h}) \\
&+ (\log(1+\ell\delta) - \log(1+(\ell-1)\delta))\frac{1}{K}\sum_{i=1}^{K}\frac{\mathbf{1}\{h_i \geq (\ell-1)\delta\}}{(\bar{r}_i)^\alpha} \\
&\leq \psi((\ell-1)\delta, \boldsymbol{h}) + \frac{\delta}{C_1(\underline{\rho}, \overline{\rho})^\alpha}, \\
&\leq \psi((\ell-1)\delta, \boldsymbol{h}) + \frac{\epsilon}{2},
\end{aligned}$$

since $c \mapsto \log(1+c)$ is increasing, $c \mapsto \mathbf{1}\{h_i \geq c\}$ is decreasing, $\log(1+\ell\delta) \leq \log(1+(\ell-1)\delta) + \delta$, and $\bar{r}_i \geq C_1(\underline{\rho}, \overline{\rho})$ for $i = 1, ..., K$. We have proven that:

$$m_\ell \leq \psi((\ell-1)\delta, \boldsymbol{h}) + \frac{\epsilon}{2}\,,\ \ell = 1, ..., L$$

and since

$$\psi((\ell-1)\delta, \boldsymbol{h}) \underset{K\to\infty}{\overset{\mathbb{P}}{\to}} \Psi((\ell-1)\delta) \leq \max_{c\geq 0}\Psi(c)\,,\ \ell = 1, ..., L,$$

we have that:

$$\mathbb{P}(m_\ell \leq \max_{c\geq 0}\Psi(c) + \epsilon) \underset{K\to\infty}{\to} 1\,,\ \ell = 1, ..., L.$$

Now consider $c \in [L\delta, \infty)$. We have the upper bound:

$$\psi(c, \boldsymbol{h}) \leq \frac{1}{KC_1(\underline{\rho}, \overline{\rho})^\alpha}\sum_{i=1}^{K}\log(1+h_i)\mathbf{1}\{h_i \geq L\delta\} \equiv Y,$$

using the fact that $\bar{r}_i \geq C_1(\underline{\rho}, \overline{\rho})$ for $i = 1, ..., K$ and:

$$\begin{aligned}
\log(1+c)\mathbf{1}\{h_i \geq c\} &\leq \log(1+h_i)\mathbf{1}\{h_i \geq c\} \\
&\leq \log(1+h_i)\mathbf{1}\{h_i \geq L\delta\}.
\end{aligned}$$

Hence $m_{L+1} \leq Y$, and we control the first and second moment of Y to show that Y is concentrated around its expectation. By definition of L and δ, since h_i has exponential distribution with mean ρ_i:

$$\begin{aligned}
\mathbb{E}(Y) &= \frac{1}{KC_1(\underline{\rho}, \overline{\rho})^\alpha}\sum_{i=1}^{K}\mathbb{E}(\log(1+h_i)\mathbf{1}\{h_i \geq L\delta\}) \\
&= \frac{1}{KC_1(\underline{\rho}, \overline{\rho})^\alpha}\sum_{i=1}^{K}\int_{L\delta}^{\infty}(\log(1+y)/\rho_i)e^{-y/\rho_i}dy, \\
&\leq \frac{1}{C_1(\underline{\rho}, \overline{\rho})^\alpha}\int_{L\delta}^{\infty}(\log(1+y)/\overline{\rho})e^{-y/\overline{\rho}}dy \\
&\leq \frac{\epsilon}{2},
\end{aligned}$$

and since $h_1, ..., h_K$ are independent:

$$\begin{aligned}\text{var}(Y) &= \frac{1}{K^2 C_1(\underline{\rho},\overline{\rho})^{2\alpha}} \sum_{i=1}^{K} \text{var}(\log(1+h_i)\mathbf{1}\{h_i \geq L\delta\}) \\ &\leq \frac{1}{KC_1(\underline{\rho},\overline{\rho})^{2\alpha}} \int_0^{+\infty} (\log(1+y)^2/\overline{\rho}) e^{-y/\overline{\rho}} dy \underset{K\to\infty}{\to} 0\end{aligned}$$

using the fact that for $i = 1, .., .K$:

$$\begin{aligned}\text{var}(\log(1+h_i)\mathbf{1}\{h_i \geq L\delta\}) &\leq \mathbb{E}(\log(1+h_i)^2 \mathbf{1}\{h_i \geq L\delta\}^2) \\ &\leq \mathbb{E}(\log(1+h_i)^2) \\ &= \int_0^{+\infty} (\log(1+y)^2/\rho_i) e^{-y/\rho_i} dy \\ &\leq \int_0^{+\infty} (\log(1+y)^2/\overline{\rho}) e^{-y/\overline{\rho}} dy.\end{aligned}$$

Hence Chebychev's inequality shows that $Y \underset{K\to\infty}{\overset{\mathbb{P}}{\to}} \mathbb{E}(Y) \leq \frac{\epsilon}{2}$, from which we deduce:

$$\mathbb{P}(m_{L+1} \leq \epsilon) \underset{K\to\infty}{\to} 1.$$

So combining both cases, we have that:

$$\mathbb{P}(m_\ell \leq \max_{c\geq 0} \Psi(c) + \epsilon) \underset{K\to\infty}{\to} 1 \ , \ \ell = 1, .., L+1.$$

We have proven that, for all $\epsilon > 0$:

$$\mathbb{P}(\max_{c\geq 0} \Psi(c) - \epsilon \leq \max_{c\geq 0} \psi(c, \boldsymbol{h}) \leq \max_{c\geq 0} \Psi(c) + \epsilon) \underset{K\to\infty}{\to} 1,$$

and $\max_{c\geq 0} \psi(c, \boldsymbol{h}) \underset{K\to\infty}{\overset{\mathbb{P}}{\to}} \max_{c\geq 0} \Psi(c)$ as announced. □

C.1.4 Putting it all together

We now complete the proof of Theorem 17. From proposition 8, asymptotically with high probability, utility optimal scheduling can be realized by selecting a threshold policy, where the threshold $c^\star$ is a maximizer of the deterministic mapping Φ. Under a theshold policy with threshold $c^\star$, the rate of user i is given by:

$$\bar{r}_i = \mathbb{E}\left(\frac{1}{T(m, J(c^\star))} \log(1+c^\star)\mathbf{1}\{h_i \geq c^\star\}\right),$$

where $J(c^\star)$ is the number of users whose channel realization is above $c^\star$:

$$J(c^\star) = \sum_{i=1}^{K} \mathbf{1}\{h_i \geq c^\star\}.$$

We have:

$$\left|\bar{r}_i - \mathbb{E}\left(\frac{1}{T(m,\infty)}\log(1+c^\star)\mathbf{1}\{h_i \geq c^\star\}\right)\right| \leq \log(1+c^\star)\mathbb{E}\left(\left|\frac{1}{T(m,\infty)} - \frac{1}{T(m,J(c^\star))}\right|\right).$$

From the law of large numbers:

$$\frac{J(c^\star)}{K} \geq \frac{1}{K}\sum_{i=1}^{K}\mathbf{1}\{\underline{\rho}(h_i/\rho_i) \geq c^\star\} \underset{K\to\infty}{\overset{a.s.}{\to}} e^{-\frac{c^\star}{\underline{\rho}}} > 0,$$

therefore $J(c^\star) \underset{K\to\infty}{\overset{a.s.}{\to}} \infty$. Hence

$$\frac{1}{T(m,J(c^\star))} \underset{K\to\infty}{\overset{a.s.}{\to}} \frac{1}{T(m,\infty)}$$

and

$$\mathbb{E}\left(\frac{1}{T(m,J(c^\star))}\right) \leq \frac{1}{T(m,1)}\ ,\ K \geq 1,$$

so we apply Lebesgue's theorem to yield:

$$\mathbb{E}\left(\left|\frac{1}{T(m,\infty)} - \frac{1}{T(m,J(c^\star))}\right|\right) \underset{K\to\infty}{\to} 0.$$

We have proven that:

$$\begin{aligned}\bar{r}_i &\underset{K\to\infty}{\to} \mathbb{E}\left(\frac{1}{T(m,\infty)}\log(1+c^\star)\mathbf{1}\{h_i \geq c^\star\}\right)\\ &= \frac{1}{T(m,\infty)}\log(1+c^\star)e^{-\frac{c^\star}{\rho_i}}.\end{aligned}$$

The value of $c^\star$ may be retrieved from the fact that applying theshold policy with threshold $c^\star$ maximizes the utility $\frac{1}{K}\sum_{i=1}^{K} g_\alpha(\bar{r}_i)$, hence:

$$\begin{aligned}c^\star &\in \underset{c\geq 0}{\operatorname{argmax}}\left\{\frac{1}{K}\sum_{i=1}^{K} g_\alpha\left(\frac{\log(1+c)e^{-\frac{c}{\rho_i}}}{T(m,\infty)}\right)\right\},\\ &= \underset{c\geq 0}{\operatorname{argmax}}\left\{\frac{1}{K}\sum_{i=1}^{K} g_\alpha\left(\log(1+c)e^{-\frac{c}{\rho_i}}\right)\right\}.\end{aligned}$$

This completes the proof of Theorem 17.

C.2 Proof of Proposition 3

In all cases, it is noted that $0 < c^\star < \infty$. Consider $\alpha = 1$. By definition, since $g_\alpha(x) = \log(x)$:

$$\begin{aligned} c^\star &\in \operatorname*{argmax}_{c\geq 0} \left\{ \frac{1}{K} \sum_{i=1}^{K} g_\alpha(\log(1+c)e^{-\frac{c}{\rho_i}}) \right\}, \\ &= \operatorname*{argmax}_{c\geq 0} \left\{ \log\log(1+c) - \frac{c}{K} \sum_{i=1}^{K} \frac{1}{\rho_i} \right\}. \end{aligned}$$

Since $c \mapsto \log\log(1+c)$ is strictly concave, mapping $c \mapsto \log\log(1+c) - \frac{c}{K}\sum_{i=1}^{K}\frac{1}{\rho_i}$ is strictly concave, hence it admits a unique local maximum which is $c^\star$. The optimal threshold $c^\star$ is thus the unique point at which the derivative is null. Differentiating we get:

$$(1+c^\star)\log(1+c^\star) = K\left(\sum_{i=1}^{K}\frac{1}{\rho_i}\right)^{-1}.$$

The result follows by definition of the Lambert function W_0.

Now consider $\alpha > 1$, so that $1-\alpha < 0$. By definition, since $g_\alpha(x) = \frac{x^{1-\alpha}-1}{1-\alpha}$:

$$\begin{aligned} c^\star &\in \operatorname*{argmax}_{c\geq 0} \left\{ \frac{1}{K} \sum_{i=1}^{K} g_\alpha(\log(1+c)e^{-\frac{c}{\rho_i}}) \right\}, \\ &= \operatorname*{argmin}_{c\geq 0} \left\{ \sum_{i=1}^{K} \log(1+c)^{1-\alpha} e^{-\frac{c(1-\alpha)}{\rho_i}} \right\}, \\ &= \operatorname*{argmin}_{c\geq 0} \left\{ (1-\alpha)\log\log(1+c) + \log\left(\sum_{i=1}^{K} e^{-\frac{c(1-\alpha)}{\rho_i}}\right) \right\}, \end{aligned}$$

where we took the logarithm to obtain the last expression. Now, since $\alpha > 1$, $c \mapsto (1-\alpha)\log\log(1+c)$ is convex, and so is $c \mapsto \log\left(\sum_{i=1}^{K} e^{-\frac{c(1-\alpha)}{\rho_i}}\right)$ (log-sum-exp function, see [77]). Hence the above admits a single local minimum, which equals $c^\star$ and may be found by solving:

$$(1+c)\log(1+c) = \frac{\sum_{i=1}^{K} e^{-\frac{c(1-\alpha)}{\rho_i}}}{\sum_{i=1}^{K} \frac{1}{\rho_i} e^{-\frac{c(1-\alpha)}{\rho_i}}}.$$

C.3 Proof of Proposition 5

From (4.10) we have

$$\frac{1}{K}\sum_{i=1}^{K} g_0(\bar{r}_{sc,i}) = \frac{1}{K}\sum_{i=1}^{K} \bar{r}_{sc,i} \tag{C.2}$$

$$= \frac{1}{K}\mathbb{E}\left[\sum_{i=1}^{K} r_{sc,i}\right] \tag{C.3}$$

$$= \frac{1}{K}\mathbb{E}\left[\max_{\mathcal{J}\subseteq[K]} \frac{|\mathcal{J}|}{T(m,|\mathcal{J}|)} \log(1+\min_{j\in\mathcal{J}} h_j)\right]. \tag{C.4}$$

Since the channel coefficients $\{h_j\}_{j\in[K]}$ are proportional to P, we have for $P\to\infty$:

$$\frac{\max_{\mathcal{J}\subseteq[K]} \frac{|\mathcal{J}|}{T(m,|\mathcal{J}|)} \log(1+\min_{j\in\mathcal{J}} h_j)}{\log(P)} \xrightarrow{a.s.} \frac{K}{T(m,K)}. \tag{C.5}$$

which means that at each slot, the selection scheme selects all K users to perform coded caching content delivery. Thus all users have the same rate. Furthermore,

$$\sup_{P\geq 0} \frac{\mathbb{E}\left[\max_{\mathcal{J}\subseteq[K]} \frac{|\mathcal{J}|}{T(m,|\mathcal{J}|)} \log(1+\min_{j\in\mathcal{J}} h_j)\right]}{\log(P)} = \sup_{P\geq 0} \frac{\mathbb{E}\left[\max_{\mathcal{J}\subseteq[K]} \frac{|\mathcal{J}|}{T(m,|\mathcal{J}|)} \log(1+\min_{j\in\mathcal{J}} h_j)\right]}{\log(P)} < \infty \tag{C.6}$$

so by Lebesgue's theorem $\frac{\mathbb{E}\left[\max_{\mathcal{J}\subseteq[K]} \frac{|\mathcal{J}|}{T(m,|\mathcal{J}|)} \log(1+\min_{j\in\mathcal{J}} h_j)\right]}{\log(P)} \to \frac{K}{T(m,K)}$. Thus,

$$\bar{r}_{sc} = \frac{1}{K}\mathbb{E}\left[\max_{\mathcal{J}\subseteq[K]} \frac{|\mathcal{J}|}{T(m,|\mathcal{J}|)} \log(1+\min_{j\in\mathcal{J}} h_j)\right] \sim \frac{\log(P)}{T(m,K)}, \tag{C.7}$$

which coincided with the baseline scheme in Proposition 1 of Chapter 3.

Bibliography

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

[2] ——, "Decentralized coded caching attains order-optimal memory-rate tradeoff," *IEEE/ACM Transactions on Networking (TON)*, vol. 23, no. 4, pp. 1029–1040, 2015.

[3] J. Agrqwal, R. Patel, P. Mor, P. Dubey, and J. Keller, "Evolution of mobile communication network: From 1g to 4g," *International Journal of Multidisciplinary and Current Research*, vol. 3, pp. 1100–1103, 2015.

[4] F. Boccardi, R. W. Heath, A. Lozano, T. L. Marzetta, and P. Popovski, "Five disruptive technology directions for 5g," *IEEE Communications Magazine*, vol. 52, no. 2, pp. 74–80, 2014.

[5] C. V. N. Index, "Global mobile data traffic forecast update, 2015-2020," *Cisco white paper*, 2015.

[6] G. Paschos, E. Baştuğ, I. Land, G. Caire, and M. Debbah, "Wireless caching: Technical misconceptions and business barriers," *IEEE Communications Magazine*, vol. 54, no. 8, pp. 16–22, 2016.

[7] K. Shanmugam, N. Golrezaei, A. G. Dimakis, A. F. Molisch, and G. Caire, "Femtocaching: Wireless content delivery through distributed caching helpers," *IEEE Transactions on Information Theory*, vol. 59, no. 12, pp. 8402–8413, 2013.

[8] N. Golrezaei, A. F. Molisch, A. G. Dimakis, and G. Caire, "Femtocaching and device-to-device collaboration: A new architecture for wireless video distribution," *IEEE Communications Magazine*, vol. 51, no. 4, pp. 142–149, 2013.

[9] E. Baştuğ, 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, p. 41, 2015.

[10] P. Blasco and D. Gündüz, "Learning-based optimization of cache content in a small cell base station," in *2014 IEEE International Conference on Communications (ICC), Sydney, Australia*, June 2014.

[11] K. Poularakis, G. Iosifidis, V. Sourlas, and L. Tassiulas, "Multicast-aware caching for small cell networks," in *2014 IEEE Wireless Communications and Networking Conference (WCNC)*, 2014.

[12] C. Yang, Y. Yao, Z. Chen, and B. Xia, "Analysis on cache-enabled wireless heterogeneous networks," *IEEE Transactions on Wireless Communications*, vol. 15, no. 1, pp. 131–145, 2016.

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

[14] E. Baştuğ, M. Bennis, M. Kountouris, and M. Debbah, "Edge caching for coverage and capacity-aided heterogeneous networks," in *2016 IEEE International Symposium on Information Theory (ISIT), Barcelona, Spain*, July 2016.

[15] N. Golrezaei, P. Mansourifard, A. F. Molisch, and A. G. Dimakis, "Base-station assisted device-to-device communications for high-throughput wireless video networks," *IEEE Transactions on Wireless Communications*, vol. 13, no. 7, pp. 3665–3676, 2014.

[16] B. Chen, C. Yang, and Z. Xiong, "Optimal caching and scheduling for cache-enabled d2d communications," *IEEE Communications Letters*, vol. 21, no. 5, pp. 1155–1158, 2017.

[17] B. Chen, C. Yang, and G. Wang, "High-throughput opportunistic cooperative device-to-device communications with caching," *IEEE Transactions on Vehicular Technology*, vol. 66, no. 8, pp. 7527–7539, 2017.

[18] B. Bai, L. Wang, Z. Han, W. Chen, and T. Svensson, "Caching based socially-aware d2d communications in wireless content delivery networks: a hypergraph framework," *IEEE Wireless Communications*, vol. 23, no. 4, pp. 74–81, 2016.

[19] M. Ji, R.-R. Chen, G. Caire, and A. F. Molisch, "Fundamental limits of distributed caching in multihop d2d wireless networks," in *2017 IEEE International Symposium on Information Theory (ISIT), Aachen, Germany*, June 2017.

[20] A. Liu, V. Lau, and G. Caire, "Capacity scaling of wireless device-to-device caching networks under the physical model," in *2017 IEEE International Symposium on Information Theory (ISIT), Aachen, Germany*, June 2017.

[21] M. Ji, G. Caire, and A. F. Molisch, "Wireless device-to-device caching networks: Basic principles and system performance," *IEEE Journal on Selected Areas in Communications*, vol. 34, no. 1, pp. 176–189, 2016.

[22] L. W. Dowdy and D. V. Foster, "Comparative models of the file assignment problem," *ACM Computing Surveys (CSUR)*, vol. 14, no. 2, pp. 287–313, 1982.

[23] D. D. Sleator and R. E. Tarjan, "Amortized efficiency of list update and paging rules," *Communications of the ACM*, vol. 28, no. 2, pp. 202–208, 1985.

[24] A. Dan, D. Sitaram, and P. Shahabuddin, "Dynamic batching policies for an on-demand video server," *Multimedia systems*, vol. 4, no. 3, pp. 112–121, 1996.

[25] I. Baev, R. Rajaraman, and C. Swamy, "Approximation algorithms for data placement problems," *SIAM Journal on Computing*, vol. 38, no. 4, pp. 1411–1429, 2008.

[26] S. Borst, V. Gupta, and A. Walid, "Distributed caching algorithms for content distribution networks," in *2010 Proceedings IEEE INFOCOM*, 2010.

[27] U. Niesen and M. A. Maddah-Ali, "Coded caching with nonuniform demands," *IEEE Transactions on Information Theory*, vol. 63, no. 2, pp. 1146–1158, 2017.

[28] J. Zhang, X. Lin, and X. Wang, "Coded caching under arbitrary popularity distributions," *IEEE Transactions on Information Theory*, vol. 64, no. 1, pp. 349–366, 2018.

[29] J. Hachem, N. Karamchandani, and S. Diggavi, "Multi-level coded caching," in *2014 IEEE International Symposium on Information Theory (ISIT), HI, USA*, June 2014.

[30] M. Ji, A. M. Tulino, J. Llorca, and G. Caire, "Order-optimal rate of caching and coded multicasting with random demands," *IEEE Transactions on Information Theory*, 2017.

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

[32] A. Ramakrishnan, C. Westphal, and A. Markopoulou, "An efficient delivery scheme for coded caching," in *2015 27th IEEE International,Teletraffic Congress (ITC 27)*, 2015.

[33] M. Ji, K. Shanmugam, G. Vettigli, J. Llorca, A. M. Tulino, and G. Caire, "An efficient multiple-groupcast coded multicasting scheme for finite fractional caching," in *2015 IEEE International Conference on Communications (ICC), London UK*, June 2015.

[34] S. Jin, Y. Cui, H. Liu, and G. Caire, "Order-optimal decentralized coded caching schemes with good performance in finite file size regime," in *2016 IEEE, Global Communications Conference (GLOBECOM)*. IEEE, 2016.

[35] L. Eleftherios and E. P., "Adding transmitters dramatically boosts coded-caching gains for finite file sizes," *arXiv preprint arXiv:1802.03389*, 2018.

[36] R. Pedarsani, M. A. Maddah-Ali, and U. Niesen, "Online coded caching," *IEEE/ACM Transactions on Networking*, vol. 24, no. 2, pp. 836–845, 2016.

[37] H. Ghasemi and A. Ramamoorthy, "Asynchronous coded caching," in *2017 IEEE International Symposium onInformation Theory (ISIT), Aachen, Germany*, June 2017.

[38] U. Niesen and M. A. Maddah-Ali, "Coded caching for delay-sensitive content," in *2015 IEEE International Conference on Communications (ICC), London, UK*, June 2015.

[39] Y. Birk and T. Kol, "Coding on demand by an informed source (iscod) for efficient broadcast of different supplemental data to caching clients," *IEEE Transactions on Information Theory*, vol. 52, no. 6, pp. 2825–2830, 2006.

[40] Z. Bar-Yossef, Y. Birk, T. Jayram, and T. Kol, "Index coding with side information," *IEEE Transactions on Information Theory*, vol. 57, no. 3, pp. 1479–1494, 2011.

[41] M. Ji, A. M. Tulino, J. Llorca, and G. Caire, "Caching and coded multicasting: Multiple groupcast index coding," in *2014 IEEE Global Conference on Signal and Information Processing (GlobalSIP)*. IEEE, 2014.

[42] W. Huang, S. Wang, L. Ding, F. Yang, and W. Zhang, "The performance analysis of coded cache in wireless fading channel," *arXiv preprint arXiv:1504.01452*, 2015.

[43] A. El Gamal and Y.-H. Kim, *Network information theory*. Cambridge university press, 2011.

[44] S. P. Shariatpanahi, S. A. Motahari, and B. H. Khalaj, "Multi-server coded caching," *IEEE Transactions on Information Theory*, vol. 62, no. 12, pp. 7253–7271, 2016.

[45] K.-H. Ngo, S. Yang, and M. Kobayashi, "Cache-Aided Content Delivery in MIMO Channels," in *2016 54th IEEE Annual Allerton Conference on Communication, Control, and Computing (Allerton), Chicago, USA*, September 2016.

[46] J. Zhang and P. Elia, "Fundamental limits of cache-aided wireless BC: Interplay of coded-caching and CSIT feedback," *IEEE Transactions on Information Theory*, vol. 63, no. 5, pp. 3142–3160, 2017.

[47] K.-H. Ngo, S. Yang, and M. Kobayashi, "Scalable Content Delivery with Coded Caching in Multi-Antenna Fading Channels," *IEEE Transactions on Wireless Communications*, vol. 17, no. 1, pp. 548–562, 2018.

[48] S. P. Shariatpanahi, G. Caire, and B. H. Khalaj, "Multi-Antenna Coded Caching," in *2017 IEEE International Symposium on Information Theory (ISIT), Aachen, Germany*, June 2017.

[49] ——, "Physical-layer schemes for wireless coded caching," *arXiv preprint arXiv:1711.05969*, 2017.

[50] E. Piovano, H. Joudeh, and B. Clerckx, "On coded caching in the overloaded miso broadcast channel," in *2017 IEEE International Symposium on Information Theory (ISIT), Aachen, Germany*. IEEE, June 2017.

[51] J. Zhang, F. Engelmann, and P. Elia, "Coded caching for reducing CSIT-feedback in wireless communications," in *2015 IEEE 53rd Annual Allerton Conference on Communication, Control, and Computing (Allerton), pp. 1099–1105, IL, USA*, September 2015.

[52] S. S. Bidokhti, M. Wigger, and R. Timo, "Noisy broadcast networks with receiver caching," *arXiv preprint arXiv:1605.02317*, 2016.

[53] M. M. Amiri and D. Gündüz, "Cache-aided content delivery over erasure broadcast channels," *IEEE Transactions on Communications*, vol. 66, no. 1, pp. 370–381, 2018.

[54] J. Zhang and P. Elia, "Wireless coded caching: A topological perspective," in *2017 IEEE International Symposium on Information Theory (ISIT), Aachen, Germany*, June 2017.

[55] S. S. Bidokhti, M. Wigger, and A. Yener, "Benefits of cache assignment on degraded broadcast channels," in *2017 IEEE International Symposium on Information Theory (ISIT), Aachen, Germany*, June 2017.

[56] N. Naderializadeh, M. A. Maddah-Ali, and A. S. Avestimehr, "Fundamental limits of cache-aided interference management," *IEEE Transactions on Information Theory*, vol. 63, no. 5, pp. 3092–3107, 2017.

[57] E. Tuncel, "Slepian-wolf coding over broadcast channels," *IEEE Transactions on Information Theory*, vol. 52, no. 4, pp. 1469–1482, 2006.

[58] S. S. Bidokhti, M. Wigger, and A. Yener, "Gaussian broadcast channels with receiver cache assignment," in *2017 IEEE International Conference onCommunications (ICC), Paris, France*, May 2017.

[59] M. Gatzianas, L. Georgiadis, and L. Tassiulas, "Multiuser broadcast erasure channel with feedback—capacity and algorithms," *IEEE Transactions on Information Theory*, vol. 59, no. 9, pp. 5779–5804, 2013.

[60] C.-C. Wang, "On the capacity of 1-to-k broadcast packet erasure channels with channel output feedback," *IEEE Transactions on Information Theory*, vol. 58, no. 2, pp. 931–956, 2012.

[61] M. J. Neely, E. Modiano, and C.-P. Li, "Fairness and optimal stochastic control for heterogeneous networks," *IEEE/ACM Transactions on Networking (TON)*, vol. 16, no. 2, pp. 396–409, 2008.

[62] J. Mo and J. Walrand, "Fair end-to-end window-based congestion control," *IEEE/ACM Transactions on Networking (ToN)*, vol. 8, no. 5, pp. 556–567, 2000.

[63] A. L. Stolyar, "On the asymptotic optimality of the gradient scheduling algorithm for multiuser throughput allocation," *Operations research*, vol. 53, no. 1, pp. 12–25, 2005.

[64] N. David, "Optimal power allocation over parallel gaussian broadcast channels," 2001.

[65] M. J. Neely, "Stochastic network optimization with application to communication and queueing systems," *Synthesis Lectures on Communication Networks*, vol. 3, no. 1, pp. 1–211, 2010.

[66] L. Georgiadis, M. J. Neely, L. Tassiulas *et al.*, "Resource allocation and cross-layer control in wireless networks," *Foundations and Trends® in Networking*, vol. 1, no. 1, pp. 1–144, 2006.

[67] S. Wang, W. Li, X. Tian, and H. Liu, "Coded caching with heterogenous cache sizes," *arXiv preprint arXiv:1504.01123*, 2015.

[68] S. Yang and M. Kobayashi, "Secure communication in k-user multi-antenna broadcast channel with state feedback," in *2015 IEEE International Symposium on Information Theory (ISIT), Hong-Kong, China*, June 2015.

[69] M. A. Maddah-Ali and D. Tse, "Completely stale transmitter channel state information is still very useful," *IEEE Transactions on Information Theory*, vol. 58, no. 7, pp. 4418–4431, 2012.

[70] P. Piantanida, M. Kobayashi, and G. Caire, "Analog index coding over block-fading miso broadcast channels with feedback," in *2013 IEEE Information Theory Workshop (ITW), pp. 1–5, Sevilla, Spain*, September 2013.

[71] R. Timo and M. Wigger, "Joint cache-channel coding over erasure broadcast channels," in *2015 IEEE International Symposium on Wireless Communication Systems (ISWCS), Brussels, Belgium*, August 2015.

[72] N. Jindal and Z.-Q. Luo, "Capacity limits of multiple antenna multicast," in *2006 IEEE International Symposium on Information Theory, pp. 1841–1845*, 2006.

[73] R. Knopp and P. A. Humblet, "Information capacity and power control in single-cell multiuser communications," *1995 IEEE International Conference on Communications. ICC'95 Seattle, Gateway to Globalization*, vol. 1, pp. 331–335, 1995.

[74] H. Shirani-Mehr, G. Caire, and M. J. Neely, "Mimo downlink scheduling with non-perfect channel state knowledge," *IEEE Transactions on Communications*, vol. 58, no. 7, pp. 2055–2066, 2010.

[75] G. Caire, R. R. Muller, and R. Knopp, "Hard fairness versus proportional fairness in wireless communications: The single-cell case," *IEEE Transactions on Information Theory*, vol. 53, no. 4, pp. 1366–1385, 2007.

[76] F. Kelly, "Charging and rate control for elastic traffic," *European transactions on Telecommunications*, vol. 8, no. 1, pp. 33–37, 1997.

[77] S. Boyd and L. Vandenberghe, *Convex optimization*. Cambridge university press, 2004.

université PARIS-SACLAY | ÉCOLE DOCTORALE Sciences et technologies de l'information et de la communication (STIC)

Titre : Limites Fondamentales De Stockage Dans Les Réseaux Sans Fil

Mots clés : Stockage, réseaux sans fil, planification

Résumé : Le stockage de contenu populaire dans des caches disponibles aux utilisateurs, est une technique émergente qui permet de réduire le trafic dans les réseaux sans fil. En particulier, le coded caching proposée par Maddah-Ali et Niesen a été considéré comme une approche prometteuse pour atteindre un temps de livraison constant au fur et à mesure que la dimension augmente. Toutefois, plusieurs limitations empêchent ses applications. Nous avons adressé les limitations de coded caching dans les réseaux sans fil et avons proposé des schémas de livraison qui exploitent le gain de coded caching. Dans la première partie de la thèse, nous étudions la région de capacité pour un canal à effacement avec cache et retour d'information. Nous proposons un schéma et prouvons son optimalité pour des cas particuliers. Ces résultats sont généralisés pour le canal à diffusion avec des antennes multiples et retour d'information. Dans la deuxième partie, nous étudions la livraison de contenu sur un canal d'atténuation asymétrique, où la qualité du canal varie à travers les utilisateurs et le temps. En supposant que les demandes des utilisateurs arrivent de manière dynamique, nous concevons un schéma basé sur une structure de queues et nous prouvons qu'il maximise la fonction d'utilité par rapport à tous les schémas limités au cache décentralisé. Dans la dernière partie, nous étudions la planification opportuniste pour un canal d'atténuation asymétrique, en assurant une métrique de justice entre des utilisateurs. Nous proposons une politique de planification simple à base de seuil avec une complexité linéaire et qui exige seulement un bit de retour de chaque utilisateur.

Title : Fundamental Limits of Coded Caching in Wireless Networks

Keywords : Coded caching, wireless networks, scheduling

Abstract : Caching, i.e. storing popular contents at caches available at end users, has received a significant interest as a technique to reduce the peak traffic in wireless networks. In particular, coded caching proposed by Maddah-Ali and Niesen has been considered as a promising approach to achieve a constant delivery time as the dimension grows. However, several limitations prevent its applications in practical wireless systems. Throughout the thesis, we address the limitations of classical coded caching in various wireless channels. Then, we propose novel delivery schemes that exploit opportunistically the underlying wireless channels while preserving partly the promising gain of coded caching. In the first part of the thesis, we study the achievable rate region of the erasure broadcast channel with cache and state feedback. We propose an achievable scheme and prove its optimality for special cases of interest. These results are generalized to the multi-antenna broadcast channel with state feedback. In the second part, we study the content delivery over asymmetric block-fading broadcast channels, where the channel quality varies across users and time. Assuming that user requests arrive dynamically, we design an online scheme based on queuing structure and prove that it maximizes the alpha-fair utility among all schemes restricted to decentralized placement. In the last part, we study opportunistic scheduling over the asymmetric fading broadcast channel and aim to design a scalable delivery scheme while ensuring fairness among users. We propose a simple threshold-based scheduling policy of linear complexity that requires only a one-bit feedback from each user.

Université Paris-Saclay
Espace Technologique / Immeuble Discovery
Route de l'Orme aux Merisiers RD 128 / 91190 Saint-Aubin, France