ParisTech
INSTITUT DES SCIENCES ET TECHNOLOGIES
PARIS INSTITUTE OF TECHNOLOGY

TELECOM ParisTech

20176ENST-0019

EDITE - ED 130

Doctorat ParisTech

T H È S E

pour obtenir le grade de docteur délivré par

TELECOM ParisTech

Spécialité « Communication et Electronique »

présentée et soutenue publiquement par

Jingjing ZHANG

le 26 Avril 2017

L'interaction entre le caching, les feedback et la topologie dans le canal de diffusion sans fil

Directeur de thèse : **Petros ELIA**

Jury

M. Konstantin AVRACHENKOV, Directeur de Recherches, INRIA, France	Président
M. Leandros TASSIULAS , Professeur, Yale University, États-Unis	Rapporteur
M. Deniz GUNDUZ, Reader, Imperial College London, Royaume-Uni	Rapporteur
Mme Michèle WIGGER, Maître de Conférences, Télécom ParisTech, France	Examinateur

TELECOM ParisTech
école de l'Institut Télécom - membre de ParisTech

THÈSE

ParisTech
INSTITUT DES SCIENCES ET TECHNOLOGIES
PARIS INSTITUTE OF TECHNOLOGY

TELECOM ParisTech

20176ENST-0019

EDITE - ED 130

Doctor of Philosophy ParisTech

DISSERTATION

In Partial Fulfillment of the Requirements
for the Degree of Doctor of Philosophy from

TELECOM ParisTech

Specialization « Electronics and Communications »

publicly presented and defended by

Jingjing ZHANG

on 26 April 2017

Interplay between caching, feedback and topology in the wireless broadcast channel

Thesis Advisor : **Petros ELIA**

THÈSE

Jury

Konstantin AVRACHENKOV, Director of Research, INRIA, France	President
Leandros TASSIULAS, Professor, Yale University, US	Reviewer
Deniz GUNDUZ, Reader, Imperial College London, UK	Reviewer
Michèle WIGGER, Associate Professor, TELECOM ParisTech, France	Examiner

TELECOM ParisTech
An Institute Telecom School - Member of ParisTech

Abstract

Current PHY-based communication solutions do not successfully scale in large networks, because as the number of users increases, these solutions cannot fully separate all users' signals. This problem in turn can gradually leave each user (i.e., each receiver) with small communication rates, and it comes at a time when wireless data traffic is expected to increase dramatically. Recently a new type of technology was proposed, in the form of an upper-layer solution, that relied on caching network-coded data at the receiving devices. The proposed solution — termed as 'coded caching' — was motivated by the fact that wireless traffic is heavily predictable, and it was based on pre-caching content from an existing library of many popular files. This promising approach naturally came with its own limitations though, and the corresponding performance would still remain limited. These two approaches (advanced PHY-based techniques, and coded-caching which used caching at the receivers) had been treated independently; after all, one uses feedback on the PHY layer, the other uses receiver-side memory on MAC.

With this as a starting point, part of what we show here is that there are interesting interplays between the two, which can potentially allow for a combination that bypasses each approach's limitations. More generally, the main work of this thesis on advanced wireless caching, is about exploring the deep connections between caching and fundamental primitives of wireless networks, such as feedback and topology.

The first part of the work explores the synergistic gains between coded caching and delayed CSIT feedback (i.e., Channel State Information at the Transmitters). Here we consider the K-user cache-aided wireless MISO broadcast channel (BC) with random fading and delayed CSIT, and identify the optimal cache-aided degrees-of-freedom (DoF) performance within a factor of 4. To achieve this, we propose a new scheme that combines basic coded-caching with MAT-type schemes, and which efficiently exploits the prospective-hindsight similarities between these two methods. This delivers a powerful synergy between coded caching and delayed feedback, in the sense that the total synergistic DoF-gain can be much larger than the sum of the individual gains from delayed CSIT and from coded caching. The derived performance interestingly reveals — for the first time — substantial DoF gains from coded caching when the (normalized) cache size γ (fraction of the library stored at

each receiving device) is very small. Specifically, a microscopic $\gamma \approx e^{-G}$ can come within a factor of G from the interference-free optimal. For example, storing at each device only a *thousandth* of what is deemed as 'popular' content ($\gamma \approx 10^{-3}$), we approach the interference-free optimal within a factor of $ln(10^3) \approx 7$ (per user DoF of 1/7), for any number of users. This result carries an additional practical ramification as it reveals how to use coded caching to essentially buffer CSI, thus partially ameliorating the burden of having to acquire real-time CSIT (considered by many as the main bottleneck in wireless networks).

The second part of the work explores an interesting interplay (and trade-off) between caching and current CSIT feedback quality. For the same setting as before, but now with an additional ability to incorporate imperfect-quality current CSIT, this part explores the link between feedback-quality and the size of the caches. This is guided by the intuition that the bigger the caches, the more side information the receivers have in their caches, the less interference one needs to handle (because of the side information), and the lesser the feedback that is potentially needed to steer interference. Conversely, more feedback may diminish the utility of memory because the better the feedback is, the more pronounced the broadcast element should be, leading to more separated streams of information, and thus fewer commonly-heard (not separated) streams which are the basis of cache-aided multi-casting. This work provides another new scheme that utilizes current and delayed feedback, to again come within a factor of at most 4 from optimal. Furthermore we describe the savings in current CSIT that we can have due to coded caching, while we also show how much caching can allow us to entirely remove current CSIT without degrading performance, basically describing the memory cost for buffering CSI.

The third part of the thesis explores feedback-aided coded caching for the same MISO BC, but with emphasis on very small caches, focusing on the case where the cumulative cache size is smaller than the library size. Here the work proposes new schemes that boost the impact of small caches, dealing with the additional challenge of having some of the library content entirely uncached, which in turn forces us to dynamically change the caching redundancy to compensate for this. Our proposed scheme is near-optimal, and this part of the thesis identifies the optimal (small) cache-aided DoF performance within a factor of 4.

The fourth part of the thesis explores — for the same K-user BC setting — the connection between coded caching and current-quality CSIT, but now without delayed CSIT. Here we show how caching, when combined with a rate-splitting broadcast approach, can improve performance and reduce the need for CSIT feedback, in the sense that the cache-aided interference-free optimal DoF performance (directly associated to perfect-CSIT and local-caching gains), can in fact be achieved with reduced-quality CSIT. These CSIT savings can be traced back to an inherent relationship between caching, performance, and

CSIT; caching improves performance by leveraging multi-casting of common information, which automatically reduces the need for CSIT, by virtue of the fact that common information is not a cause of interference. At the same time though, too much multicasting of common information can be detrimental, as it does not utilize existing CSIT. For this, we designed simple schemes that build on the Maddah-Ali and Niesen coded caching scheme, by properly balancing multicast and broadcast opportunities, and by combing caching with rate-splitting communication schemes that are specifically designed to operate under imperfect-quality CSIT. The observed achievable CSIT savings here are more pronounced for smaller libraries and for smaller numbers of users. In the end, this part allows us to quantify the intuition that, in the presence of coded-caching, there is no reason to improve CSIT beyond a certain threshold quality.

The fifth part of the thesis explores wireless coded caching from a topological perspective. The fact that coded caching employs multicasting (i.e., communicating a common message to many receiving users at the same time), causes performance to deteriorate when the links have unequal capacities. Such uneven topologies, where some users have weaker channels than others, introduce the problem that any multicast transmission that is meant for at least one weak user, could conceivably have to be sent at a lower rate, thus 'slowing down' the rest of the strong users as well. With this as motivation, we explore coded caching in a SISO BC setting where some users have higher link capacities than others (all users have the same cache size). Focusing on a binary and fixed topological model where strong links have a fixed normalized capacity 1, and where weak links have reduced normalized capacity, we identify — as a function of the cache size and the topology — the optimal throughput performance, within a factor of at most 8. The transmission scheme that achieves this performance, employs a simple form of interference enhancement, and exploits the property that weak links attenuate interference, thus allowing for multicasting rates to remain high even when involving weak users. This approach ameliorates the negative effects of uneven topology in multicasting, now allowing all users to achieve the optimal performance associated to maximal capacity, even if the capacity of the weaker users decreases down to a certain threshold capacity. This leads to the interesting conclusion that for coded multicasting, the weak users need not bring down the performance of all users, but on the contrary to a certain extent, the strong users can lift the performance of the weak users without any penalties on their own performance.

At the end of the thesis we also present a result that does not involve caching. This part explores the DoF limits of the (two user) SISO X channel with imperfect-quality CSIT, and it shows that the same DoF-optimal performance — previously associated to perfect-quality current CSIT — can in fact be achieved with current CSIT that is of imperfect quality. The work also shows that the DoF performance previously associated to perfect-quality delayed CSIT, can in fact be achieved in the presence of imperfect-quality delayed

CSIT. These follow from the presented sum-DoF lower bound that bridges the gap — as a function of the quality of delayed CSIT — between the cases of having no feedback and having delayed feedback, and then another bound that bridges the DoF gap — as a function of the quality of current CSIT — between delayed and perfect current CSIT. The bounds are based on novel precoding schemes that are presented here and which employ imperfect-quality current and/or delayed feedback to align interference in space and in time.

Résumé

Due à l'augmentation de nombre des utilisateurs, les méthodes de communication basées sur le PHY ne permettent pas de séparer les différents signaux des utilisateurs dans un réseau à grandes dimensions. Ce problème peut progressivement laisser chaque utilisateur (i.e., chaque récepteur) avec de faibles débits de communication, et il apparait lorsque le flux de données augmente d'une façon spectaculaire. Récemment, une nouvelle technologie a été proposée pour les couches de niveaux supérieur. Cette technologie, nommée « coded caching » consiste à mémoriser les données décodées de réseaux dans les appareils réceptrices. La proposition de cette technologie a été motivée d'une part par le fait que le trafic sans fil est très prévisible et d'autre part par le fait qu'elle est basée sur la pré-mémorisation de contenu d'une bibliothèque de nombreux fichiers populaires. Cette approche prometteuse possèdent des limites et sa performance reste toujours limitée. Ces deux approches ont été traité séparément. Cependant, il existe des intersections intéressent entre les deux ce qui permet une éventuelle combinaison pour contourner les limites de chacune.

Le travail de cette thèse, en caching avancé, consiste à explorer des connexions profondes entre le caching et les primitifs principales des réseaux sans fils comme feedback et topologie.

La première partie de cette thèse explore les gains synergiques entre le « coded caching » et la rétroaction retardée du CSIT. Nous considérons ici K-utilisateur cache-aidé MISO BC sans fil avec une décoloration aléatoire et un CSIT retardé, et identifient les performances optimales en degrés de liberté (DoF) à mémoire cache dans un facteur de 4. Pour cela, nous proposons un nouveau système combinant le « coded caching » de base avec les approaches de type MAT, et qui exploite efficacement les similitudes prospectives et rétrospectives entre ces deux méthodes. Cela offre une synergie puissante entre le « coded caching » et la rétroaction retardée, en ce sens que le DoF-gain synergique peut être beaucoup plus grande que la somme de la personne gains de CSIT différée et de « coded caching » . La performance dérivée révèle intéressante — pour la première fois — gains substantiels DoF à partir du « coded caching » , même lorsque la taille du cache (normalisée) γ (fraction de la bibliothèque stockée à chaque récepteur) est très faible. Plus précisément, un $\gamma \approx e^{-G}$ microscopique peut atteindre un facteur de G par rapport à l'optimum sans interférence. Par

exemple, en stockant à chaque récepteur un emph millième de ce qui est considéré comme un contenu 'populaire' ($\gamma \approx 10^{-3}$), nous approchons l'optimal sans interférence dans un facteur de $\ln(10^3) \approx 7$ (par utilisateur DoF de 1/7), pour un nombre quelconque d'utilisateurs. Ce résultat entraîne une ramification pratique supplémentaire car il révèle comment utiliser coded caching pour amortir essentiellement CSI, ce qui améliore partiellement le fardeau d'avoir à acquérir en temps réel CSIT (considéré par beaucoup comme le principal goulet d'étranglement dans les réseaux sans fil).

La deuxième partie de cette thèse explore une interaction intéressante (et compromis) entre le coded caching et la rétroaction actuelle du CSIT. Comme nous l'avons mentionné avant mais maintenant avec une capacité supplémentaire à incorporer une qualité imparfaite CSIT. Cette partie explore le lien entre la rétroaction et la taille des caches. Cela est guidé par l'intuition que plus les caches sont grandes, plus les récepteurs ont des informations latérales dans leurs caches, moins d'interférences il faut (à cause des informations latérales), et moins la rétroaction qui est potentiellement nécessaire pour diriger l'interférence. Inversement, plus de rétroaction peut diminuer l'utilité de la mémoire parce que la meilleure rétroaction est, plus l'élément de diffusion doit être prononcé, Conduisant à des flux d'informations plus séparés, et donc moins Couramment entendus (non séparés) qui sont à la base de cache-aidé multi-casting.

La troisième partie de la thèse explore feedback-aidé « coded caching » pour le même MISO BC, mais en mettant l'accent sur les caches très petites et en se concentrant sur le cas où la taille du cache cumulative est plus petite que la taille de la bibliothèque. Ici, le travail propose de nouveaux stratégies qui stimulent l'impact des petites caches, traitant le défi supplémentaire d'avoir une partie du contenu de la bibliothèque entièrement uncached, ce qui nous oblige à changer dynamiquement la redondance de cache pour compenser cela. Les stratégies proposées sont presque optimales, et cette partie de la thèse identifie la performance optimale cache-aidé DoF dans un facteur de 4.

La quatrième partie de la thèse explore, pour les même réseaux de K-récepteurs BC; la connexion entre coded et la CSIT actuel, mais sans retardé CSIT. Dans cette partie, nous montrons comment le « caching » , combinée à une approche de diviseur des taux, peut améliorer les performances et réduire le besoin de retour d'information CSIT , en ce sens que la cache-aidé DoF performance optimale peuvent être réalisées avec un CSIT de qualité réduite. Ces économies CSIT peuvent être retracées à une relation inhérente entre caching, la performance et CSIT; caching améliore les performances en profitant du multi-casting d'informations communes, ce qui réduit automatiquement le besoin de CSIT, en raison du fait que l'information commune n'est pas une cause d'interférence. Cependant, beaucoup de multicasting de l'information commune peut être préjudiciable, car il n'utilise pas des CSIT existants. Pour cela, nous avons conçu des méthodes simples qui s'appuient sur le système de Maddah-Ali et Niesen coded caching, en équilibrant correcte-

ment les opportunités du multicasting et de la télédiffusion, et en combinant caching avec les schémas diviseur des taux qui sont spécifiquement conçus pour fonctionner sous CSIT de qualité imparfaite. Les économies de CSIT réalisables observées ici sont plus prononcées pour les petites bibliothèques et pour un plus petit nombre d'utilisateurs. En fin de compte, cette partie permet de quantifier l'intuition que, en présence de coded caching, il n'y a aucune raison d'améliorer CSIT au-delà d'un certain seuil de qualité.

La cinquième partie de la thèse explore le « coded caching » sans fil d'un point de vue topologique. Le fait que le « coded caching » emploie la multicasting (i.e., la communication d'un message commun à de nombreux utilisateurs en même temps), cause la détérioration des performances lorsque les liens ont des capacités inégales. Ces topologies inégales, où certains utilisateurs ont des canaux plus faibles que d'autres, introduisent le problème que toute transmission multidestinataire destinée à au moins un utilisateur faible, pourrait devoirêtre envoyée à un débit inférieur, ralentissant ainsi le reste du forte utilisateurs ainsi. Avec ceci comme motivation, nous explorons le « coded caching » dans un SISO BC où certains utilisateurs ont des capacités de lien plus élevées que d'autres. En se concentrant sur un modèle topologique binaire et fixe où les liens forts ont une capacité normale fixe 1 et où les liens faibles ont une capacité normalisée réduite. Nous identifions — en fonction de la taille du cache et de la topologie — la performance de débit optimale dans un facteur d'au plus 8. L'approche de transmission qui réalise cette performance utilise une forme simple d'amélioration de l'interférence et exploite la propriété que les liaisons faibles atténuent l'interférence, permettant ainsi aux taux de multicasting de rester élevés, même lorsqu'ils impliquent des utilisateurs faibles.

À la fin de cette thèse, nous présentons des résultats qui n'impliquent pas le caching. Cette dernière partie explore les DoF limites du SISO X canal (deux utilisateurs) avec CSIT de qualité imparfaite et montre que la même DoF performance optimale - précédemment associée au CSIT actuel de qualité parfaite - peut être atteinte avec le CSIT actuel qui est de qualité imparfaite. Le travail montre également que la performance du DoF précédemment associée au CSIT retardé de qualité parfaite peut être réalisée en présence d'un CSIT retardé de qualité imparfaite. Ceux-ci résultent de la limite inférieure de la somme-DoF présentée qui comble l'écart— en fonction de la qualité du CSIT différé— entre les cas de ne pas avoir de retour et ayant un retour retardé, puis une autre borne qui comble le DoF gap; en fonction de la qualité du courant CSIT— entre le retard et le courant parfait CSIT. Les limites sont basées sur des nouvelles approches de precoding qui sont présentées dans cette thèse et qui utilisent une rétroaction de qualité imparfaite et/ou retardée pour aligner l'interférence dans l'espace et dans le temps.

Acknowledgments

After three years and half's PhD study at Eurecom, I have reaped enormously not only in knowledge, but also in life. Many thanks to so many people.

Foremost, I would like to express my sincere gratitude to my supervisor, Prof. Petros Elia. Thanks to him for offering me this great opportunity and leading me to the hall of information theory. Thanks to him for his genius ideas and expertise from which I benefited immensely. Thanks for his patience and continual inspiration, helping me all the time of research. He taught me what I should insist on research. His passion and attitude about research and life will always light me in my future.

In addition to my supervisor, I would like to thank all other members of my thesis committee: Prof. Konstantin Avrachenkov, Prof. Leandros Tassiulas, Prof. Deniz Gündüz, and Prof. Michèle Angela Wigger, for agreeing to serve as the committee of my defense and for their insightful comments and encouragement.

I would like to express my thanks to the colleagues and friends at Eurecom, for their general care and wonderful events we had together. To Manijeh Bashar, Xiwen Jiang, Qianrui Li, Xiaolan Sha, Yongchao Tian, Paolo Viotti, Xinping Yi, Haifan Yin, Raj Haresh Patel, Konstantinos Alexandris, Alberto Benegiamo, Nikolaos Sapountzis, Paul de Kerret, George Arvanitakis, Laurent Gallo, Romain Favraud, and many others, with whom I spent a great time during my PhD study. To Eleftherios Lampiris, who has always been open and patient to a discussion. To Shengyun Liu and Sosina Mengistu Gashaw, my dearest flatmates, for the unforgettable moments.

I would also like to thank all my friends outside of Eurecom. To Lili Zhang, who is far away in China but always has taken the time to listen to me. To Dimitra Tsigkari, for the beautiful coffee afternoons we had together.

Last, but not the least, I would like to express my deep gratitude to my families for their unconditional love and support. Particularly to my parents, who sacrifice so much for raising me up and my education, and my brother, who always encourages me to strive for my dream. All the supports they have provided me throughout my life was the greatest gift anyone has ever given me. I would like to give a special thank to my brother's cute twin daughters, weiwei and hanhan, for the happiness they bring me.

Contents

List of Figures

List of Tables

Acronyms

We summarize here the acronyms that are commonly used in this dissertation. The meaning of an acronym is also indicated when it is first used. The English acronyms are also used for the French summary.

AWGN	Additive White Gaussian Noise
MIMO	Multi-Input Multi-Output
MISO	Multi-Input Single-Output
SISO	Single-Input Single-Output
BC	Broadcast Channel
XC	X Channel
CSIR	Channel State Information at the Receiver
CSIT	Channel State Information at the Transmitter
DoF	Degrees-of-Freedom
IA	Interference Alignment
i.i.d.	Independent and Identically Distributed
et al.	et alii, et alia (and others)
e.g.	exempli gratia (for the sake of example)
resp.	Respectively
ZF	Zero-Forcing
SNR	Signal-to-Noise Ratio
TIN	Treating Interference as Noise

Notations

Common notations and symbols are list as below, with the rest notations defined in the text where they occur.

N, M, γ, x	Variables
ψ	Set
$\boldsymbol{x}$	Vector
$\oplus$	The bitwise XOR operation
$\binom{n}{k}$	n-choose-k
$\|\psi\|$	The cardinality of the set ψ
$\|\|\boldsymbol{x}\|\|^2$	The magnitude of the vector $\boldsymbol{x}$
$\text{dur}(\boldsymbol{x})$	The transmission duration of the vector $\boldsymbol{x}$
$\psi_1 \backslash \psi_2$	The difference set
$\boldsymbol{x}^\perp$	Normalized orthogonal component of a non-zero vector $\boldsymbol{x}$
$\boldsymbol{x}^T$	Transpose of a vector $\boldsymbol{x}$
$\cap$	Union of two sets
$\bar{\psi}$	The complementary set of ψ
x^+	$\max\{x, 0\}$
H_n	The n_{th} harmonic number
$\log_2(\cdots)$	A logarithm of base 2
$\lfloor\rfloor, \lceil\rceil$	The floor or ceiling of a number
min, max	Minimum, maximum

Chapter 1

Introduction

1.1 Basic Motivation

Current communication solutions do not successfully scale in large networks, because as the number of users increases, these solutions cannot fully separate all users' signals. This in turn can gradually leave each user small communication rates, and it comes at a time when wireless data traffic is expected to increase by 10 times in just 5 years [3]. If data volumes continue to expand so rapidly, it is foreseen that wireless networks will soon come to a halt, thus inevitably compromising the informational foundations of society. This looming network-overflow can also have severe environmental consequences, because current systems would require exponential increases in transmit-power to accommodate future data volumes; Telecommunications has in fact already a higher carbon footprint than aviation [4]. Despite the imminence of this overload, the consensus is that there is no existing or currently envisioned technology that resolves this, not even with brute force increase of bandwidth, or of the number and size of base-stations.

The problem in addressing the above overload, mainly originates from the inherently real-time nature of communications, where data must be 'served' on time. This entails having to continuously adapt communications—in real-time—to the rapidly fluctuating states of a large wireless network. 'Feedback' in this context, refers to the action of disseminating large amounts of overhead information (referred to as channel-state information, or CSI) about the instantaneous strengths of each propagation-path between the different nodes. These channels fluctuate up to hundreds of times per second, hence as the network size increases, the overhead consumes more and more resources, eventually leaving no room for actual data.

The limitations of feedback-based PHY Most communication methods use feedback as their main resource. Feedback mainly works for separating users' signals, so the better the feedback process, the better the signal 'steering', the better the signal separation, the less the interference, the higher the rates. The problem is that the feedback overhead provably dominates all network resources. Not too long ago, it was revealed that because of feedback, many cellular settings could never escape vanishingly small throughputs, irrespective of any conceivable form of transmission cooperation [5]. This stood as a proxy to the limitations of many architectures, like in massive MIMO (see the award winning Grassmann-manifold approach [6]), or even in fully decentralized solutions such as the powerful interference-alignment methods [7], where there is a polynomial aggravation of the feedback problem, and a required super-exponential complexity just to get any non-vanishing rate. Even if feedback was relegated to being non-real-time (which is much easier to obtain), this again forces near-zero rates (cf [1,8] as well as [9,10], etc.). Finally even if networks were heavily densified with wireline infrastructure, the ensuing sampling-complexity and propagation bottlenecks, can far exceed feedback bottlenecks (on this, see the recent [11]).

A clear way to see this 'feedback-bottleneck' problem is in the simple setting of large MIMO, where the interference-free optimal performance (which can be achieved by simple channel inversion), comes under the assumption that the channel (which changes as many as a few hundred times per second) is known at the transmitter side, in an instantaneous manner, which in turn implies real-time no-delay feedback of channel estimates that have near-perfect accuracy. As argued, this is impossible, and the resulting CSIT imperfections result in dramatic performance deterioration [6]. To see this better, note that the channel comes with a coherence period T_C during which the channel is roughly time-invariant, and has a coherence bandwidth W_C which is the frequency interval over which the channel stays again fixed. The large overhead of acquiring each channel estimate, forces the per-user degrees of freedom d ($d \approx \frac{\text{Capacity}}{logSNR}$ is the normalized capacity — see more later) to be upper bounded as

$$d \leq \frac{\min(K, \frac{T_C W_C}{2})}{K}\big(1 - \frac{\min\left(K, \frac{T_c W_c}{2}\right)}{T_c W_c}\big) \leq \frac{T_c W_c}{4K} \to 0$$

which slowly vanishes to zero, because the overhead scales with K which can exceed the coherence window $T_C W_C$. This manifests itself in a variety of settings, including that of time division duplexing (TDD) with uplink-downlink reciprocity.

The emergence of coded caching Recently, an upper-layer solution was proposed that relied on caching network-coded data at the receiving devices [2]. The proposed solution was motivated by the fact that wireless traffic is heavily video or audio on-demand (over 60%), which entails an ability to predict

data requests in advance ('the night before'). This approach was based on caching content from an existing library of many popular files. Each user would pre-store (without knowing next day's requests) a carefully selected sequence of sub-files from the library, specifically designed to speed up (next day's) communications. In contrast to the above, memory was used to disseminate side-information (context), which can be best utilized by multicasting, i.e., by transmitting signals that must be 'heard' by many. This is exactly the opposite of feedback-aided separation where generally users only hear 'their own' signals. This promising approach — which was presented at the time for the single-stream error-free (wireline) broadcast channel, offers gains—in terms of per-user degrees of freedom—which can remain limited: the DoF gain remained approximately equal to the ratio between cache size and library size.

Insight: The idea behind the exploration of feedback and caching Our work will seek to fuse these two seemingly independent worlds of caching and of advanced feedback-aided PHY. This work is a natural response to the apparent need to jointly utilize these two powerful resources in wireless communications: memory and feedback. Why though would the feedback process of disseminating CSI today, be so powerfully linked to the process of preemptive data-storage yesterday? To see this, recall that feedback and memory are complementary; feedback separates signals, while memory brings them together (because side information enables multicasting, which is the opposite of signal separation). This means that when incomplete feedback inadvertently forces incomplete signal separation, it paradoxically boosts the impact of memory-aided multicasting. This also holds in reverse, and in essence, each element builds on the imperfections of the other. The second connection stems from the fact that both cases face parallel problems. How we cache on Tuesday, knowing that the file requests will be revealed on Wednesday, is a structurally similar problem to how we transmit now, knowing that the 'location' (channel) will be revealed later. It is this second structural link that allows memory to make non-real-time feedback relevant, and it is the first connection that gives a certain fail-safe property to this duality.

Insight: The power of preemptive micro-insertions of data in wireless settings Traditional caching (prefetching) methods mainly reduce the volume of the problem for the day after, reflecting the old saying "Do something today, so that you do not have to do it tomorrow". Rather than changing the volume of the problem though, a much more powerful approach is to use splashes of memory to change the structure of the problem. To make things clear, let us draw an analogy. Imagine you are sitting at a table where everyone speaks Italian, except you. You understand almost nothing, until a good Samaritan translator intermittently offers small hints (few words about what is said). We all know these can be very helpful. Assume now there are many

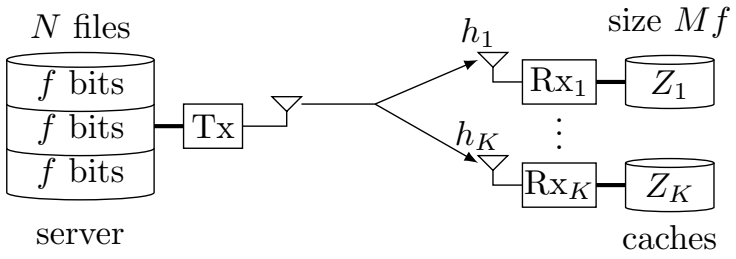

Figure 1.1: Cache-aided single-stream BC.

possible topics of conversation, and many non-Italian listeners, who were separately taught the day before a few things about each topic. Assume that this 'common training' was designed, so that the translator can use single sentences that are useful to many at a time (few words that—given the training—offer context on more than one discussion). The next day, listeners choose a topic they want to listen in. Then the translator must communicate in a setting where the ability of listeners to comprehend, is not stable. He does not fully know in which state they are, but must nonetheless compensate for dips in comprehension, so that at the end he can satisfy the curiosity of all listeners, as quickly as possible. This drew a parallel to our setting, which is though more involved.

1.1.1 Brief overview of existing challenges of coded caching

Consider a wireless network with one server and K receiving users, requesting data from a library of N possible files, where for simplicity, each file is of length f bits (see Fig. 1.1). The library (catalog) is known in advance, and it may for example consist of popular movies, where each file can be about a minute of one of many 'popular' movies that might be requested the day after. Consider a simplified single-stream noiseless shared link (broadcast), of normalized capacity f (one file per normalized unit-time). Serving one user at a time, would imply a required time $T = K$ needed to guarantee the completion of the delivery for any K requested files. This corresponds to a (normalized) per-user throughput of $d = \frac{1}{K}$ (files per unit time). As K increases, this per-user rate vanishes to zero. Suppose now that each user (each receiver) has a storage (cache) memory of capacity equal to $M < N$ files. By caching at each user 'during the night before' (off peak hours) a fraction $\gamma = \frac{M}{N} < 1$ of each file, meant that any 'day-time' request from the library could be handled in $T = K(1-\gamma)$ which yields an improved 'day-timeâĂŹ throughput of $d \approx \frac{1}{K(1-\gamma)} \to 0$ which still goes to zero as K increases. This is the aforementioned prefetching approach used in current systems. In general, such traditional routing- and popularity-based prefetching schemes would yield a vanishingly small normalized throughput $d(\gamma) \to \frac{1}{K\beta'} \to 0$,

where β' here captures the cache size γ, and the file popularities (cf. [12, 13]). This 'prefetching' line of research produced engaging results, which may work well in the wired internet with routing-based protocols, but not in modern wireless networks which are fundamentally different, because their structure and 'topology' fluctuates up to hundreds of times per second, as well as because of the involved structural manipulation of signals (for example, in wireless settings, interference is something to be used, rather than to be always avoided as a collision). The evolved coded variant of this in [2], showed that by caching data in a specific way, essentially coding across caches during the caching phase, and across different data requested by different users (during the next day), one could serve users with a single network-coded multicast transmission, and achieve—during the 'day-time' delivery period—a throughput

$$d \to \frac{\gamma}{1-\gamma}$$

that was directly proportional to the cache size M, and which did not approach zero as K increased. The problem is that for relatively small values of γ—the above per-user throughput would scale approximately as $d \to \frac{\gamma}{1-\gamma} \to \gamma$ which is highly problematic because γ is expected to be microscopically small, ranging in the order of $\gamma \approx 10^{-4} \to 10^{-2}$ or even less ([14]), reflecting the widely held belief that realistically, end-user devices will be able to cache only a small fraction of the library, because such a library must be very large in order for it to be requested sufficiently often the next day. The higher the number of library files N, the more probable it is that the requested content will draw from this library, and the more often the coded-caching gains will appear. Anything other than microscopically small γ would either imply unrealistically large caches, or impractically small libraries that almost never yield caching gains. The expectation is that $M \ll N \geq K \gg 1$.

The problem persists even in the presence of realistic file-popularity considerations, which have been pivotal in prefetching approaches that preceded coded-caching. As we saw in [13] for a realistic distribution of file popularities, the gains—even though they far exceeded the traditional local-caching gains from prefetching—were in fact again scaling as $d \to \gamma$ for smaller γ (see also [12]). In fact the same problem further persisted even in the presence of a powerful cluster of L servers/antennas (where $L = \lambda K, \lambda \in (0, 1)$) that served the K users over an error-free network, which—even in the presence of perfect-feedback assumptions—gave a throughput (cf. [15])

$$d(\gamma) = \lambda + \gamma$$

which meant that the gain attributed to memory, is again $d(\gamma) - d(\gamma = 0) = \gamma$ simply because the term $d(\gamma = 0) = \lambda$ (this is easy to see) is solely due to having multiple servers (multiple antennas) with perfect real-time feedback, which we argued before is unrealistic for larger K. All existing memory-based

solutions that we know, had a restricted impact for small γ. As we show, this limitation $d(\gamma) - d(\gamma = 0) \approx \gamma$, is not fundamental.

Another crucial ingredient is topology, and how this affects the use of memory. Again this is almost fully unexplored (with a notable exception in [16]) (In the absence of memory, some work has been done on understanding the relationship between feedback and topology, cf. [17–19] as well as [10]).

Let us further explore the state of art of some recent feedback-aided solutions (focusing on feedback timeliness and quality considerations) that are closer to the spirit of this thesis, and then some memory-aided solutions that have appeared over the last 2-3 years.

1.2 Further state-of-art

Our work here — which considers the joint application of CSIT and coded caching as two powerful and interconnected tools for removing sizable fractions of the interference in multiuser wireless networks — builds on new work on coded caching and on the many results on the performance ramifications of feedback timeliness and quality. Let us begin with a very brief overview of prior works in the area of feedback-aided interference management (focusing mainly on the BC, and on the context of this work here), which will then be followed by a more extensive overview of caching-related efforts, focusing almost exclusively on developments surrounding coded caching.

1.2.1 State of art on feedback-aided interference management (no caching)

The role of feedback in removing multiuser interference, naturally involves many directions and facets. One particular direction of research in this area has sought to derive DoF limits that reflect the effects of feedback quality and timeliness, and which often — under different settings and in different forms — employ communication schemes that work retrospectively to alleviate the effect of CSIT delays and imperfections. Such works include the work by Maddah-Ali and Tse [1] who showed how retrospective communications over a fast fading channel, can render completely obsolete CSIT useful, as well as other subsequent works [8, 9, 20–28] that build on this to further incorporate CSIT quality considerations. Other works that relate to the approach here, can be found in [29–32].

1.2.2 State of art on cache-aided interference management

The benefits of coded caching on reducing interference and improving performance, came more recently with the aforementioned work by Maddah-Ali and Niesen in [2] who considered a caching system where a server is connected to

multiple users through a shared link, and designed a novel caching and delivery method that jointly offers a multicast gain that helps mitigate the link load, and which was proven to have a gap from optimal that is at most 12. This work was subsequently generalized in different settings, which included the setting of different cache sizes for which Wang et al. in [37] developed a variant of the algorithm in [2] which achieves a gap of at most 12 from the information theoretic optimal. Other extensions included the work in [38] by Maddah-Ali and Niesen who considered the setting of decentralized caching where the achieved performance was shown to be comparable to that of the centralized case [2], despite the lack of coordination in content placement. For the same original single-stream setting of [2], the work of Ji et al. in [39] considered a scenario where users make multiple requests each, and proposed a scheme that has a gap to optimal that is less than 18. Again for the setting in [2], the work of Ghasemi and Ramamoorthy in [40], derived tighter outer (lower) bounds that improve upon existing bounds, and did so by recasting the bound problem as one of optimally labeling the leaves of a directed tree. Further work can be found in [41] where Wang et al. explored the interesting link between caching and distributed source coding with side information. Interesting conclusions are also drawn in the work of Ajaykrishnan et al. in [42], which revealed that the effectiveness of caching in the single stream case, is diminished when N approaches and exceeds K^2. Furthermore Amiri et al. improved upon the original performance of Maddah-Ali and Niesen when the system has more users than files ($K \geq N$), while in [43] Amiri and Gündüz managed to improve performance (reduced delay, to within a factor of up to 2 from optimal), for specific values of M.

Deviating from single-stream error-free links, different works have considered the use of coded caching in different wireless networks, without though particular consideration for CSIT feedback quality. For example, work by Huang et al. in [44], considered a cache-aided wireless fading BC where each user experiences a different link quality, and proposed a suboptimal communication scheme that is based on time- and frequency-division and power- and bandwidth-allocation, and which was evaluated using numerical simulations to eventually show that the produced throughput decreases as the number of users increases. Further work by Timo and Wigger in [16] considered an erasure broadcast channel and explored how the cache-aided system efficiency can improve by employing unequal cache sizes that are functions of the different channel qualities. Another work can be found in [45] where Maddah-Ali and Niesen studied the wireless interference channel where each transmitter has a local cache, and showed distinct benefits of coded caching that stem from the fact that content-overlap at the transmitters allows effective interference cancellation.

Different work has also considered the effects of caching in different non-classical channel paradigms. One of the earlier such works that focused on practical wireless network settings, includes the work by Golrezaei et al. in [46],

which considered a downlink cellular setting where the base station is assisted by helper nodes that jointly form a wireless distributed caching network (no coded caching) where popular files are cached, resulting in a substantial increase to the allowable number of users by as much as $400 - 500\%$. In a somewhat related setting, the work in [47] by Perabathini et al. accentuated the energy efficiency gains from caching. Interesting work can also be found in [48] and in [49] on using stochastic geometry and network coding for modeling wireless caches and storage devices, as well as in [50, 51] that explore higher-layer considerations relating to caching.

Further work by Ji et al. in [52] derived the limits of so-called combination caching networks in which a source is connected to multiple user nodes through a layer of relay nodes, such that each user node with caching is connected to a distinct subset of the relay nodes. Additional work can also be found in [53] where Niesen et al. considered a cache-aided network where each node is randomly located inside a square, and it requests a message that is available in different caches distributed around the square. Further related work on caching can be found in [39, 54–58].

Work that combines caching and feedback considerations in wireless networks, has only just recently started. A reference that combines these, can be found in [59] where Deghel et al. considered a MIMO interference channel (IC) with caches at the transmitters. In this setting, whenever the requested data resides within the pre-filled caches, the data-transfer load of the backhaul link is alleviated, thus allowing for these links to be instead used for exchanging CSIT that supports interference alignment. An even more recent concurrent work can be found in [60] where Ghorbel et al. studied the capacity of the cache-enabled broadcast packet erasure channel with ACK/NACK feedback. In this setting, Ghorbel et al. cleverly showed — interestingly also using a retrospective type algorithm, by Gatzianas et al. in [61] — how feedback can improve performance by informing the transmitter when to resend the packets that are not received by the intended user and which are received by unintended users, thus allowing for multicast opportunities.

The first work that considers the actual interplay between coded caching and CSIT quality, can be found in [62] which considered the easier problem of how the optimal cache-aided performance (with coded caching), can be achieved with reduced quality CSIT.

1.2.3 Measure of Performance

The measure of performance here will be the (normalized) duration T — in time slots, per file served per user — needed to complete the delivery process, *for any request*. Note that this is a 'worst case' measure (corresponding to the case where each user demands a distinct file). This T must not be perceived

as a traditional latency measure, but rather as a throughput measure[1], simply because of the normalization by the amount of information sent.

Equivalently, when meaningful, we will also consider the *cache-aided degrees of freedom per user* (cache-aided DoF) which is simply[2]

$$d = \frac{1-\gamma}{T} \in [0,1] \tag{1.1}$$

which is indeed a measure of throughput[3] of the delivery phase, and which does not include the benefits of having some content already available at the receivers (local caching gain), and rather focuses on capturing the effect of feedback and coded caching, in handling interference.

1.2.4 Cache-aided broadcast channel model

We will mainly consider the setting of the symmetric multiple-input single-output (MISO) BC where a transmitter that is equipped with K antennas, communicates to K single-antenna users. The transmitter has access to a library of N distinct files $W_1, W_2, \cdots, W_N$, each of size $|W_n| = f$ bits. Each user $k \in 1, 2, ..., K$ has a cache Z_k with size $|Z_k| = Mf$ (bits), and this size takes the normalized form

$$\gamma \triangleq \frac{M}{N}.$$

The communication has two phases, the placement phase and the delivery phase. During the first phase (off-peak hours), the caches $\{Z_k\}_{k=1}^K$ at the users are pre-filled with the information from the N files $\{W_n\}_{n=1}^N$. During the second phase, the transmission commences when each user k requests a single file W_{R_k} out of the library.

In this setting, the received signals at each user k take the form

$$y_k = \boldsymbol{h}_k^T \boldsymbol{x} + z_k, \quad k = 1, \ldots, K$$

where $\boldsymbol{x} \in \mathbb{C}^{K\times 1}$ denotes the transmitted vector satisfying a power constraint $\mathbb{E}(||\boldsymbol{x}||^2) \leq P$, where $\boldsymbol{h}_k \in \mathbb{C}^{K\times 1}$ denotes the vector fading coefficients of the channel of user k and where z_k represents unit-power AWGN noise at receiver k.

At the end of the communication, each receiving user k combines the received signals y_k — accumulated during the delivery phase — with the information available in their respective cache Z_k, to reconstruct their desired file W_{R_k}.

[1] In fact in [2] the same measure is referred to as the *rate*.

[2] We note that Kd is simply the coding gain $K(1-\gamma)/T$ that is often used to quantify the gain from coded-caching.

[3] By definition of the DoF, this throughput would now scale — in the high SNR setting of interest — as $d\log_2(P)$ bits ($d\log_2(SNR)$ bits) of resolved content per (delivery-phase) channel use.

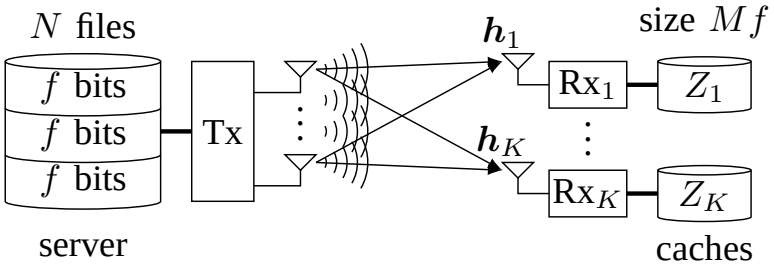

Figure 1.2: Cache-aided K-user MISO BC.

1.2.5 CSIT-type feedback model

Communication also takes place in the presence of channel state information at the transmitter. CSIT-type feedback is crucial in handling interference, and can thus substantially reduce the resulting duration T of the delivery phase. This CSIT is typically of imperfect-quality as it is hard to obtain in a timely and reliable manner. In the high-SNR (high power P) regime of interest, this current-CSIT quality — whenever this is assumed to be available — will be concisely represented in the form of the normalized quality exponent [20] (see also [9])

$$\alpha := -\lim_{P\to\infty} \frac{\log \mathbb{E}[||\boldsymbol{h}_k - \hat{\boldsymbol{h}}_k||^2]}{\log P},\ k \in \{1,\dots,K\} \tag{1.2}$$

where $\boldsymbol{h}_k - \hat{\boldsymbol{h}}_k$ denotes the estimation error between the current CSIT estimate $\hat{\boldsymbol{h}}_k$ and the estimated channel $\boldsymbol{h}_k$. The range of interest[4] is $\alpha \in [0,1]$. In some cases (Chapters 2 and 3), we will also assume availability of delayed CSIT (as in for example [1], as well as in a variety of subsequent works [9,20–27] as well as [64–66]) where now the delayed estimates of any channel, can be received without error but with arbitrary delay, even if this delay renders this CSIT completely obsolete.

Intuition: Mixed feedback This mixed CSI model (partial current CSIT, and delayed CSIT) nicely captures different realistic settings that might involve channel correlations and an ability to improve CSI as time progresses. This same *CSI model is particularly well suited for our caching-related setting here, because it explicitly reflects two key ingredients that — as we will see — are deeply intertwined with coded caching; namely, feedback quality (which will reveal a memory-vs-feedback tradeoff), and feedback timeliness (which introduces a non-linear aspect to the problem, which in turn can be translated into a dramatic — non-linear — boost in the impact of caching).*

[4] In the high SNR regime of interest here, $\alpha = 0$ corresponds to having essentially no current CSIT (cf. [63]), while having $\alpha = 1$ corresponds (again in the high SNR regime) to perfect and immediately available CSIT.

1.3 Extended summary of contributions

This thesis centers around coded caching and seeks to understand how coded caching can be meaningfully combined with different wireless network ingredients such as feedback, multiple antennas, and even topology. In this section, we summarize all the contributions of the thesis.

Chapter 2: The synergistic gains of coded caching and delayed-CSIT feedback The first part of the work explores the synergistic gains between coded caching and delayed CSIT feedback. Here we consider the K-user ($K \leq N$) cache-aided wireless MISO broadcast channel (BC) with random fading and delayed CSIT ($\alpha = 0$), and identify the optimal cache-aided degrees-of-freedom performance within a factor of 4. For $H_n \triangleq \sum_{i=1}^{n} \frac{1}{i}$ denoting the harmonic number, the first result says that

$$T = H_K - H_{K\gamma}$$

is achievable and has a gap-to-optimal that is less than 4, for all problem parameters (for all K, γ). In the presence of the well known logarithmic approximation $H_n \approx \log(n)$ (which becomes tight as K increases) then the above T takes the form

$$T = \log(\frac{1}{\gamma})$$

and the corresponding per-user DoF takes the form

$$d(\gamma) = \frac{1-\gamma}{\log(\frac{1}{\gamma})}. \tag{1.3}$$

What we see is that — for larger values of K — the corresponding gain that is directly attributed to caching

$$d(\gamma) - d^*(\gamma = 0) \to \frac{1-\gamma}{\log(\frac{1}{\gamma})} > \gamma, \ \forall \gamma \in (0, 1]$$

can substantially exceed the typical coded-caching (per-user DoF) gain γ. The gain appears very strongly at smaller values of γ, where we see that the derivative — when evaluated at $\gamma = 1/K$ — takes the form

$$\frac{\delta\big(d(\gamma) - d(\gamma = 0)\big)}{\delta\gamma}\Big|_{\gamma=\frac{1}{K}} \approx \frac{K}{\log^2 K}$$

revealing a substantial DoF boost at the early stages of γ. These can be compared to linear gains prior to this work, where the derivative is constant $\frac{\delta\big(d(\gamma)-d(\gamma=0)\big)}{\delta\gamma} = \frac{\delta\big(\gamma\big)}{\delta\gamma} = 1, \quad \forall\gamma.$

In the same regime, these gains in fact imply an exponential (rather than linear) effect of coded caching, in the sense that now a modest $\gamma = e^{-G}$ can offer a very satisfactory

$$d(\gamma = e^{-G}) \approx \frac{1}{G} \tag{1.4}$$

which is only a factor G from the interference-free (cache-free) optimal $d = 1$.

For example, storing at each device only a *thousandth* of what is deemed as 'popular' content ($\gamma \approx 10^{-3}$), we approach the interference-free optimal within a factor of $ln(10^3) \approx 7$ (per user DoF of 1/7), for any number of users.

Intuition on schemes:
The corresponding memory-aided schemes and information-theoretic outer bounds, will be discussed in detail later on. The key idea behind the main scheme is that the caching algorithm creates a multi-destination delivery problem that is the same as that which is efficiently solved by the last stages of the MAT scheme. In essence, caching allows us to skip the first $K\gamma$ phases (i.e., to skip the first phases) of the MAT scheme, which happen to have the longest time duration (phase i has duration $1/i$). This gives us an idea as to why the impact of small caches (small γ) is substantial; even small caches can remove a large fraction of the communication duration. Upon MAT decoding, we simply proceed with coded-caching decoding based on the algorithm in [2].

Intuition on the outer bound:
To lower bound T, we first consider an 'easier' (faster) case of having $s \leq K$ users that are not interfered by the remaining users, and then assume that the cache contents are shared among all the users. The we run the experiment where each user is served one file, repeating this experiment $\lfloor \frac{N}{s} \rfloor$ times (this part is from [2]). Then we switch to the (easier) case that the shared cache contents are 'transferred' via a side-information multicating link, to all receivers. This allows us a lower bound that is linked to that of the s-user MISO BC with a parallel side channel whose throughput is sMf bits (the cumulative caching size) during the whole transmission, all in the presence of delayed CSIT. The we combine this with the cut-set bound, to get a lower bound on T which is then maximized over all s.
These results were presented in

- Jingjing Zhang, Petros Elia, "The Synergistic Gains of Coded Caching and Delayed Feedback", arXiv:1604.06531, April 2016.

- Jingjing Zhang, Petros Elia, "Fundamental Limits of Cache-Aided Wireless BC: Interplay of Coded-Caching and CSIT Feedback", *IEEE Transactions on Information Theory*, to appear 2017.

Chapter 3: Tradeoff between memory and current CSIT In Chapter 3, we present in closed form (for the same K-user MISO BC, with a K-antenna transmitter serving K users with caches) the previously unexplored interplay between caching and feedback-quality. Specifically we will see that the achievable T and its corresponding DoF $d(\gamma, \alpha)$, take the form

$$T = \frac{(1-\gamma)(H_K - H_{K\gamma})}{\alpha(H_K - H_{K\gamma}) + (1-\alpha)(1-\gamma)}$$

$$d(\gamma, \alpha) = \alpha + (1-\alpha)\frac{1-\gamma}{H_K - H_{K\gamma}}$$

and are both proved to be at most a factor of 4 from optimal. Under the logarithmic approximation, the derived T takes the form

$$T(\gamma, \alpha) = \frac{(1-\gamma)\log(\frac{1}{\gamma})}{\alpha\log(\frac{1}{\gamma}) + (1-\alpha)(1-\gamma)}$$

and the derived DoF takes the form

$$d(\gamma, \alpha) = \alpha + (1-\alpha)\frac{1-\gamma}{\log\frac{1}{\gamma}}.$$

If we were to focus on the larger K setting (for the sake of crisp interpretation), what the above suggests is that current CSIT offers an initial DoF boost of $d^*(\gamma = 0, \alpha) = \alpha$ (cf. [67]), which is then supplemented by a DoF gain

$$d(\gamma, \alpha) - d^*(\gamma = 0, \alpha) \to (1-\alpha)\frac{1-\gamma}{\log(\frac{1}{\gamma})}$$

attributed to the synergy between delayed CSIT and caching.

The DoF expression clearly points to a certain tradeoff between α and γ, hence one can imagine trading-off one for the other. Hence to capture the (memory-aided) feedback savings, let us consider

$$\delta_\alpha(\gamma) \triangleq \arg\min_{\alpha'}\{\alpha' : (1-\gamma)T^*(\gamma = 0, \alpha') \leq T(\gamma, \alpha)\} - \alpha$$

describing the *CSIT reduction due to caching* (from α', down to an operational α, without loss in performance) and for which we show that caching can achieve a CSIT reduction which, under the logarithmic approximation, takes the form

$$\delta_\alpha(\gamma, \alpha) = (1-\alpha)d(\gamma, \alpha = 0) = (1-\alpha)\frac{1-\gamma}{\log(\frac{1}{\gamma})}.$$

Furthermore, a simple calculation — which we show here for the larger-K regime which yields crisper insight — can tell us that

$$\gamma_\alpha^{'} \triangleq \arg\min_{\gamma^{'}}\{\gamma^{'} : d(\gamma^{'}, \alpha = 0) \geq d^*(\gamma = 0, \alpha)\} = e^{-1/\alpha}$$

which means that $\gamma_\alpha^{'} = e^{-1/\alpha}$ suffices to achieve — in conjunction with delayed CSIT — the optimal DoF performance $d^*(\gamma = 0, \alpha)$ associated to a system with delayed CSIT and α-quality current CSIT. This tells us how much memory is needed to substitute current with delayed CSIT without loss in performance, which can be interpreted as the memory cost for gaining the ability to 'buffer CSI'.

Intuition on schemes:
To offer some intuition on the schemes, we note that the caching part is modified from [2] to *'fold'* (linearly combine) the different users' data into multi-layered blocks, in a way such that the subsequent transmission algorithm (QMAT algorithm, cf. [67]) (specifically the last $K - \eta_\alpha$ ($\eta_\alpha \in \{\Gamma, \dots, K-1\}$) phases of the QMAT algorithm) can efficiently deliver these blocks. Equivalently the algorithms are calibrated so that the caching algorithm creates a multi-destination delivery problem that is the same as that which is efficiently solved by the last stages of the QMAT-type communication scheme.

Then the intuition is that as α increases, we can have more private data, which means that there is less to be cached, which means that caching can have higher redundancy, which implies XORs of higher order, which means that we can multicast to more users at a time, which in turn means that we can skip more phases of QMAT. The intensity of the impact of small values of γ relates to the fact that the early phases of QMAT are the longest. So while a small γ can only skip a few phases, it skips the longest ones, thus managing to substantially reduce delay.

Intuition on the outer bound:
The proof of the lower bound on T starts similar to the case of having only delayed CSIT; first assume only $s \leq K$ users who share caches and who are not interfered by the rest, and then repeat this $\lfloor \frac{N}{s} \rfloor$ times. Then we had to derive novel outer bounds for the s-user MISO BC with mixed CSIT (the novelty here was to account for imperfect-quality current feedback) and with an additional presence of a parallel link of total throughput sMf bits. This bound uses a sequence of entropy-based inequalities. The proof continues then as in the case of $\alpha = 0$.

The results can be found in

- Jingjing Zhang, Petros Elia, "Fundamental Limits of Cache-Aided Wireless BC: Interplay of Coded-Caching and CSIT Feedback", in *Proc. 54th Annual Allerton Conf. Communication, Control and Computing (Allerton'16)*, Illinois, USA, October 2016.

- Jingjing Zhang, Petros Elia, "Fundamental Limits of Cache-Aided Wireless BC: Interplay of Coded-Caching and CSIT Feedback", *IEEE Transactions on Information Theory*, to appear 2017.

See also

- Paul de Kerret, David Gesbert, Jingjing Zhang, and Petros Elia, "Optimally bridging the gap from delayed to perfect CSIT in the K-user MISO BC", in *Proc. of IEEE Information Theory Workshop (ITW'16)*, Cambridge, UK, September 2016. (long version, arXiv:1604.01653).
- Paul de Kerret, David Gesbert, Jingjing Zhang, and Petros Elia, "Optimal DoF of the K-User broadcast channel with delayed and imperfect current CSIT", *IEEE Transactions on Information Theory*, submitted 2016. (arXiv:1604.01653).

Chapter 4: Feedback-aided coded caching with very small caches
Chapter 4 explores feedback-aided coded caching for the same symmetric MISO BC, but with emphasis on very small caches, focusing on the case where the cumulative cache size is smaller than the library size (i.e., $KM \leq N$, i.e., $\Gamma \triangleq K\gamma \leq 1$). The following identifies, up to a factor of 4, the optimal T^*, for all $\Gamma \in [0,1]$. We use the expression

$$\alpha_{b,\eta} = \frac{\eta - \Gamma}{\Gamma(H_K - H_\eta - 1) + \eta}, \ \eta = 1, \ldots, K-1. \tag{1.5}$$

The result tells us that for $KM \leq N$ ($\Gamma \leq 1$) and for $N \geq K$, then for $\eta = 1, \ldots, K-2$,

$$T = \begin{cases} \frac{H_K - \Gamma}{1-\alpha+\alpha H_K}, & 0 \leq \alpha < \alpha_{b,1} \\ \frac{(K-\Gamma)(H_K - H_\eta)}{(K-\eta)+\alpha(\eta+K(H_K-H_\eta-1))}, & \alpha_{b,\eta} \leq \alpha < \alpha_{b,\eta+1} \\ 1-\gamma, & \frac{K-1-\Gamma}{(K-1)(1-\gamma)} \leq \alpha \leq 1 \end{cases}$$

is achievable, and has a gap from optimal that is less than 4 ($\frac{T}{T^*} < 4$), for all α, K. For $\alpha \geq \frac{K-1-\Gamma}{(K-1)(1-\gamma)}$, T is optimal.

In the absence of current CSIT ($\alpha = 0$), again for $K\gamma < 1$, the result tells us that

$$T = H_K - \Gamma$$

is achievable and has a gap from optimal that is less than 4.

On the schemes:
Here the work proposes new schemes that boost the impact of small caches, as they manage to deal with the additional challenge of having some of the library content entirely uncached, which in turn forces us to dynamically change the caching redundancy to compensate for this.

On the outer bound:
The lower bound on T, draws directly from that in Chapter 3 which was derived for all N, K, M, γ, and which can thus be applied here for the range $\gamma \in [0, \frac{1}{K}]$.

- Jingjing Zhang, Petros Elia, "Feedback-Aided Coded Caching for the MISO BC with Small Caches", to appear in *Proc. of IEEE International Conference on Communications (ICC'17)*, Paris, France, May 2017. (long version, arXiv:1606.05396).

Chapter 5: Trading off feedback with memory: no delayed CSIT In Chapter 5, we consider the same MISO BC as before, except that now delayed feedback is removed. In this setting, and for $N = K$, the interference-free optimal $T^* = 1 - \frac{M}{N}$ can be achieved with an α that need not be bigger than

$$\alpha_{th} = \frac{N - 1 - M}{\frac{M}{K} + (N - 1 - M)}.$$

On the schemes:
This was achieved by combining caching with a rate-splitting broadcast method; an approach that not only improved performance, but also reduced the need for CSIT, in the sense that the cache-aided (interference-free) optimal DoF performance associated to caching and perfect CSIT, was in fact achieved with reduced-quality CSIT.

On the outer bound:
The only outer bound used is that $T \geq 1 - \gamma$ which is the simple bound after removing all interference.

The results were published in

- Jingjing Zhang, Felix Engelmann, and Petros Elia, "Coded caching for reducing CSIT-feedback in wireless communications", in *Proc. 53rd Annual Allerton Conf. Communication, Control and Computing (Allerton'15)*, Illinois, USA, October 2015.

Chapter 6: Wireless coded caching: A topological perspective In Chapter 6, we move away from feedback and consider the aspect of topology, in a basic SISO BC. Topology here simply means that some links are stronger than others. This asymmetry can be a liability, as we comment below, but it can also be a blessing; after all, one can readily 'hide' interference in the direction of week channels. In this context, the thesis explores the performance of coded caching in a wireless SISO BC setting where some users have higher link capacities than others. All users have the same cache size. Focusing on a binary and fixed topological model where strong links have a fixed normalized capacity 1, and where weak links have reduced normalized capacity $\tau < 1$, we identify — as a function of the cache size and τ — the optimal throughput performance, within a factor of at most 8. The transmission scheme that

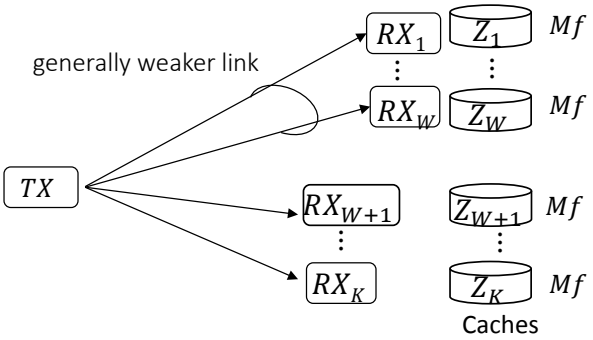

Figure 1.3: Cache-aided K-user MISO BC.

achieves this performance, employs a simple form of interference enhancement, and exploits the property that weak links attenuate interference, thus allowing for multicasting rates to remain high even when involving weak users. For any K, W, γ, τ, where W is the number of 'weak' users, the achievable $T(\tau)$ (which is at most 8 times from the theoretical optimal), takes the form

$$T(\tau) = \begin{cases} \frac{T(W)}{\tau}, & 0 \leq \tau < \bar{\tau}_{thr} \\ \min\{T(K-W)+T(W), \frac{\tau_{thr}T(K)}{\tau}\}, & \bar{\tau}_{thr} \leq \tau \leq \tau_{thr} \\ T(K), & \tau_{thr} < \tau \leq 1 \end{cases}$$

From the last part of the above expression, we see that this approach ameliorates the negative effects of uneven topology in multicasting, now allowing all users to achieve the optimal performance $T(K)$ associated to $\tau = 1$, even if τ is approximately as low as

$$\tau \geq 1 - (1-w)^g$$

where g is the coded-caching gain, and where $w = W/K$ is the fraction of users that are weak. This leads to the interesting conclusion that for coded multicasting, the weak users need not bring down the performance of all users, but on the contrary to a certain extent, the strong users can lift the performance of the weak users without any penalties on their own performance.

On the schemes:
To achieve this, we used interference enhancement, by performing power-level superposition coding, and sequential successive interference cancellation, where strong users first treat their own signals as noise to decode out the signals of the weak users, and then decode their own. Note that, had we simply sent the multicasting signals sequentially at the maximum rate allowed (depending on the destination), this threshold τ would have been 1 (and thus any reduction in capacity, for even $W = 1$ user, would have had a cost in terms of the overall

performance).

On the outer bounds:
To lower bound T, we proceed to apply the idea of considering an 'easier' case of having only W weak users that are not interfered by the remaining users. Now for this smaller system with $K = W$ users, we exploit the cut-set bound in [2], where each user has a link of capacity τ. By doing so, we can get a lower bound on T after simple normalization (division) by τ. For $\tau \geq \bar{\tau}_{th}$, we established a tighter lower bound on T, which is bounded by the system of having K strong users (i.e., $\tau = 1$) from [2] due to the limited capacity of the weak users.

- Jingjing Zhang, Petros Elia, "Wireless Coded Caching: A Topological Perspective", submitted to *IEEE International Symposium on Information Theory (ISIT'17)*, Aachen, Germany, June 2017. (long version, arXiv:1606.08253).

Chapter 7: The SISO-X channel with imperfect-quality CSIT In Chapter 7 we present a result that does not involve caching. This part explores the DoF limits of the (two user) SISO X channel with imperfect-quality CSIT, and it shows that the same DoF-optimal performance — previously associated to perfect-quality current CSIT — can in fact be achieved with current CSIT that is of imperfect quality. The work also shows that the DoF performance previously associated to perfect-quality delayed CSIT, can in fact be achieved in the presence of imperfect-quality delayed CSIT. These follow from the presented sum-DoF lower bound that bridges the gap — as a function of the quality of delayed CSIT — between the cases of having no feedback and having delayed feedback, and then another bound that bridges the DoF gap — as a function of the quality of current CSIT — between delayed and perfect current CSIT.

On the schemes:
The inner bounds are based on novel precoding schemes that are presented here and which employ imperfect-quality current and/or delayed feedback to align interference in space and in time.

Specifically we show that for the two-user XC with perfect-quality delayed CSIT, and with imperfect current CSIT of quality exponent α, the optimal sum DoF is lower bounded as

$$d_\Sigma \geq \min\Big(\frac{4}{3}, \frac{6}{5} + \frac{2\alpha(2-3\alpha)}{5(4-7\alpha)}\Big).$$

As a result, the optimal sum DoF $d_\Sigma = \frac{4}{3}$ can be achieved with imperfect current CSIT of quality that need not exceed $\alpha = \frac{4}{9}$.

Furthermore we show that for the same two-user XC with no current CSIT and with imperfect delayed CSIT of quality exponent β, the optimal sum DoF is lower bounded as

$$d_{\Sigma} \geq \min\left(\frac{6}{5}, 1+\frac{\beta}{3}\right)$$

and as a result, the (linear-) optimal sum-DoF $d_{\Sigma} = \frac{6}{5}$, previously associated to perfect-quality delayed feedback, can in fact be achieved with imperfect-quality delayed CSIT of quality that need not exceed $\beta = \frac{3}{5}$.

- Jingjing Zhang, Dirk TM Slock, and Petros Elia, "Achieving the DoF limits of the SISO X channel with imperfect-quality CSIT", in *Proc. of IEEE International Symposium on Information Theory (ISIT'15)*, Hong Kong, China, June 2015.

Chapter 8: Conclusions Chapter 8 offers some conclusions, summarizing some of the results, discussing some possible ramifications of having modest amounts of caching in designing larger BC systems and file libraries, some thoughts on whether caching is a more impactful resource than CSIT feedback, and a small discussion on the need for tighter information-theoretic outer bounds for the case of feedback-and-cache aided wireless networks.

Chapter 2

The Synergy between Coded Caching and Delayed CSIT Feedback

In this chapter, we consider the K-user cache-aided wireless MISO broadcast channel (BC) with random fading and delayed CSIT, and identify the optimal cache-aided degrees-of-freedom (DoF) performance within a factor of 4. The achieved performance is due to a scheme that combines basic coded-caching with MAT-type schemes, and which efficiently exploits the prospective-hindsight similarities between these two methods. This delivers a powerful synergy between coded caching and delayed feedback, in the sense that the total synergistic DoF-gain can be much larger than the sum of the individual gains from delayed CSIT and from coded caching.

The derived performance interestingly reveals — for the first time — substantial DoF gains from coded caching, even when the (normalized) cache size γ (fraction of the library stored at each receiving device) is very small. Specifically, a microscopic $\gamma \approx e^{-G}$ can come within a factor of G from the interference-free optimal. For example, storing at each device only a *thousandth* of what is deemed as 'popular' content ($\gamma \approx 10^{-3}$), we approach the interference-free optimal within a factor of $ln(10^3) \approx 7$ (per user DoF of 1/7), for any number of users. This result carries an additional practical ramification as it reveals how to use coded caching to essentially buffer CSI, thus partially ameliorating the burden of having to acquire real-time CSIT.

2.1 Introduction

In the setting of broadcast-type communication networks where one transmitter serves the interfering requests of more than one receiving user, recent work in [2] showed how properly-encoded caching of content at the receivers, and proper encoding across different users' requested data, can provide increased effective throughput and a reduced network load. This was achieved by creating — through coding — multicast opportunities where common symbols are simultaneously needed by more than one user, even if such users requested different data content. This *coded caching* approach has since motivated different works in [12, 15, 16, 37–42, 44, 45, 54–60, 62] which considered the utilization of coded caching over a variety of different settings, including the recent concurrent works in [60] that considered the cache-enabled broadcast packet erasure channel with ACK/NACK feedback, and the preliminary work in [62] that considered caching with imperfect-quality feedback.

Part of our motivation here is to explore the connection between coded caching and communications with imperfect feedback. Intuitively both cases face parallel problems: a transmitter with complete data knowledge, must retroactively compensate for only having partial knowledge of the 'destination', may this be the identity of the receiving user the 'next day', or the partially known channel. These connections will eventually allow for a non-separability property which we here translate into substantial synergistic gains between delayed feedback and coded caching. These gains are pertinent because there is a real need to boost the performance effect of generally modest cache sizes, and because delayed CSIT is often the only feedback resource that is available in larger networks with rapidly fluctuating channel states.

2.1.1 Caching-aided broadcast channel model

As stated in the introduction, in this chapter we consider the setting of the symmetric multiple-input single-output (MISO) BC where a transmitter that is equipped with K antennas, communicates to K single-antenna users. For the sake of independence, we here repeat some of the basic notation as well as clarify a bit further the basic assumptions for this specific chapter.

The transmitter has access to a library of N distinct files $W_1, W_2, \cdots, W_N$, each of size $|W_n| = f$ bits. Each user $k \in 1, 2, ..., K$ has a cache Z_k with size $|Z_k| = Mf$ (bits), and this size takes the normalized form

$$\gamma = \frac{M}{N}.$$

The communication has two phases, the placement phase and the delivery phase. During the first phase (off-peak hours), the caches $\{Z_k\}_{k=1}^K$ at the users are pre-filled with the information from the N files $\{W_n\}_{n=1}^N$. During the second phase, the transmission commences when each user k requests a

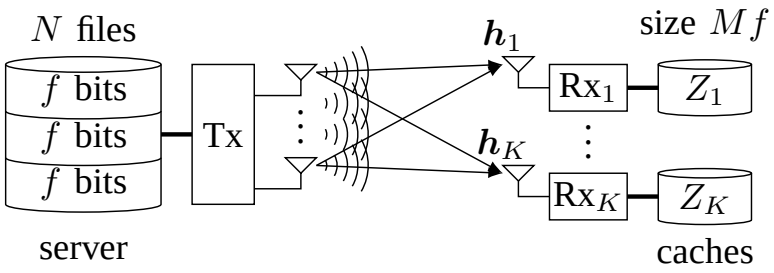

Figure 2.1: Cache-aided K-user MISO BC.

single file W_{R_k} out of the library. As noted, the received signals at each user k take the form

$$y_k = \boldsymbol{h}_k^T \boldsymbol{x} + z_k, \quad k = 1, \ldots, K \tag{2.1}$$

where $\boldsymbol{x} \in \mathbb{C}^{K\times 1}$ denotes the transmitted vector satisfying a power constraint $\mathbb{E}(||\boldsymbol{x}||^2) \leq P$, where $\boldsymbol{h}_k \in \mathbb{C}^{K\times 1}$ denotes the vector fading coefficients of the channel of user k and where z_k represents unit-power AWGN noise at receiver k. At the end of the communication, each receiving user k combines the received signals y_k — accumulated during the delivery phase — with the information available in their respective cache Z_k, to reconstruct their desired file W_{R_k}.

2.1.2 Notation and assumptions

For the sake of completeness we repeat some of the notation, while we also introduce here additional notation. We will use $\Gamma = \frac{KM}{N} = K\gamma$ to mean that the sum of the sizes of the caches across all users, is Γ times the volume of the N-file library. As in [2], we will consider the case where $\Gamma = \{1, 2, \cdots K\}$. We will also use the notation $H_n = \sum_{i=1}^{n} \frac{1}{i}$ to represent the n_{th} harmonic number, and we will use $\epsilon_n \triangleq H_n - \log(n)$ to represent its logarithmic approximation error, for some integer n. We remind the reader that ϵ_n decreases with n, and that $\epsilon_\infty \triangleq \lim_{n\to\infty}(H_n - \log(n)) \approx 0.5772$. $\binom{n}{k}$ will be the n-choose-k operator, and $\oplus$ will be the bitwise XOR operation. We will use $[K] \triangleq \{1, 2, \cdots, K\}$. If ψ is a set, then $|\psi|$ will denote its cardinality. For sets A and B, then $A\backslash B$ denotes the difference set. Complex vectors will be denoted by lower-case bold font. We will use $||\boldsymbol{x}||^2$ to denote the magnitude of a vector $\boldsymbol{x}$ of complex numbers. For a transmitted vector $\boldsymbol{x}$, we will use $\text{dur}(\boldsymbol{x})$ to denote the transmission duration of that vector, e.g., $\text{dur}(\boldsymbol{x}) = \frac{1}{10}T$ would mean that the transmission of vector $\boldsymbol{x}$ lasts one tenth of T. Logarithms are of base e, while $\log_2(\cdot)$ will represent a logarithm of base 2.

Main assumptions

Throughout this work, we assume availability of delayed CSIT (as in for example [1], as well as a variety of subsequent works [8,9,20–27]) where now the delayed estimates of any channel, can be received at the transmitter, without error but with arbitrary delay, even if this delay renders this CSIT completely obsolete. We hasten to note that delayed CSIT here is not meant to offer an additional performance boost by itself, but rather is employed solely as a tool that will link coded-caching to communications with non-perfect feedback.

We will also ask that each receiver knows their own channel perfectly. We further adhere to the common convention (see for example [1]) of assuming perfect and global knowledge of delayed channel state information at the receivers (delayed global CSIR), where each receiver must know (with delay) the CSIR of (some of the) other receivers. We will assume that the entries of *each specific* estimation error vector are i.i.d. Gaussian. For the outer (lower) bound to hold, we will make the common assumption that the current channel state must be independent of the previous channel-estimates and estimation errors, *conditioned on the current estimate* (there is no need for the channel to be i.i.d. in time). We will make the assumption that the channel is drawn from a continuous ergodic distribution such that all the channel matrices and all their sub-matrices are full rank almost surely.

2.2 Performance of the cache-aided MISO BC

We begin with an outer (lower) bound on the optimal duration T^*. The proof is found in the Appendix.

Lemma 2.1 *The optimal T^* for the (K, M, N) cache-aided K-user MISO BC with delayed CSIT, is lower bounded as*

$$T^* \geq \max_{s\in\{1,\ldots,min\{\lfloor \frac{N}{M} \rfloor, K\}\}} H_s - \frac{sM}{\lfloor \frac{N}{s} \rfloor}. \tag{2.2}$$

We now proceed with the main result.

Theorem 2.1 *In the (K, M, N) cache-aided MISO BC with $K \leq N$ users, and with $\Gamma \in \{1, 2, \cdots, K-1\}$, then*

$$T = H_K - H_\Gamma \tag{2.3}$$

is achievable and has a gap-to-optimal

$$\frac{T}{T^*} < 4 \tag{2.4}$$

that is less than 4, for all K.

Proof. The scheme that achieves the above performance is presented in Section 2.4, while the corresponding gap to the optimal performance is bounded in Section 3.8. ■

The following corollary offers some insight by adopting the logarithmic approximation $H_n \approx \log(n)$ (which becomes tight as K increases)[1].

Corollary 2.1 *Under the logarithmic approximation, the above T takes the form*

$$T = \log(\frac{1}{\gamma})$$

and the corresponding per-user DoF takes the form

$$d(\gamma) = \frac{1-\gamma}{\log(\frac{1}{\gamma})}. \tag{2.5}$$

Example 2.1 *In a MISO BC system with $\alpha = 0$, K antennas and K users, in the absence of caching, the optimal per-user DoF with delayed CSIT is $d^*(\gamma = 0) = 1/H_K$ (cf. [1]) which vanishes to zero as K increases. A DoF of $1/4$ can be guaranteed with $\gamma \approx \frac{1}{50}$ for all K, a DoF of $1/7$ with $\gamma \approx \frac{1}{1000}$, and a DoF of $1/11.7$ can be achieved with $\gamma \approx 10^{-5}$, again for all K.*

2.2.1 Synergistic DoF gains

We proceed to derive some insight from the above, and for this we look to the large K regime, where there is no ambiguity on which gains can be attributed solely to coded caching (in addition to possible DoF gains due to other resources such as feedback). In this regime, what the above says is that the gain that is directly attributed to caching

$$d(\gamma) - d^*(\gamma = 0) \to \frac{1-\gamma}{\log(\frac{1}{\gamma})} > \gamma, \ \forall \gamma \in (0, 1]$$

can substantially exceed[2] the typical coded-caching (per-user DoF) gain γ.

What we also see, again for larger K, is that while the individual component settings/algorithms (MAT from [1], and the Maddah-Ali and Niesen (MN) algorithm from [2]) respectively provided individual DoF gains of the form

[1] To avoid confusion, we clarify that the main theorem is simply a DoF-type result, that nothing but SNR scales to infinity, and the derived DoF holds for all K. The corollaries are simply the approximation of the above expression, under the logarithmic approximation, which becomes tight as K increases. The corollary is derived directly by approximating the expression in Theorem 2.1 (eq. 3.4) for larger values of K.

[2] In this larger K setting, we have $d_{\text{SS}}(\gamma)+d_{\text{MAT}} \to \gamma$. We clarify that this step is simply the result of a large-K approximation of the corresponding expression from the main theorem. In that sense, K scales after SNR does. We also recall from [1] that $d^*(\gamma = 0) = \frac{1}{H_K}$ which decreases with K.

$d_{\text{MAT}} = d^*(\gamma = 0) = \frac{1}{H_K}$ and $d_{\text{SS}}(\gamma) = \frac{1-\gamma}{\frac{K(1-\gamma)}{1+K\gamma}} = \gamma + \frac{1}{K}$ (cf. [2]), the combination of these two components results in a synergistic

$$d(\gamma) > d_{\text{SS}}(\gamma) + d_{\text{MAT}}, \quad \forall \gamma \in [0,1]$$

that — for larger K — exceeds the sum of the two individual components. This is the first time that such synergistic gains have been recorded. The gains become very striking for smaller values of γ in which case we have that $\frac{1-\gamma}{\log(\frac{1}{\gamma})} \gg \gamma$.

Derivative analysis for understanding the small-γ gains attributed to caching Let us fix K, and consider the derivative of the DoF gain attributed to caching

$$d(\gamma) - d(\gamma = 0) = \frac{1-\gamma}{H_K - H_{K\gamma}} - \frac{1}{H_K} \approx \frac{1-\gamma}{\log(1/\gamma)} - \frac{1}{H_K} \tag{2.6}$$

which takes the form

$$\frac{\delta\big(d(\gamma) - d(\gamma=0)\big)}{\delta\gamma} \approx \frac{\frac{1}{\gamma} - 1 - \log(\frac{1}{\gamma})}{(\log(\frac{1}{\gamma}))^2} \approx \frac{\frac{1}{\gamma}}{(\log(\frac{1}{\gamma}))^2} \tag{2.7}$$

which, when evaluated at $\gamma = 1/K$, gives

$$\frac{\delta\big(d(\gamma) - d(\gamma=0)\big)}{\delta\gamma}\Big|_{\gamma=\frac{1}{K}} \approx \frac{K}{\log^2 K}$$

revealing a substantial DoF boost at the early stages [3] of γ.

These can be compared to linear gains where the derivative is constant

$$\frac{\delta\big(d(\gamma) - d(\gamma=0)\big)}{\delta\gamma} = \frac{\delta\big(\gamma\big)}{\delta\gamma} = 1, \quad \forall\gamma. \tag{2.8}$$

These gains in fact imply[4] an exponential (rather than linear) effect of coded caching, in the sense that now a microscopic $\gamma = e^{-G}$ can offer a very satisfactory

$$d(\gamma = e^{-G}) \approx \frac{1}{G} \tag{2.9}$$

which is only a factor G from the interference-free (cache-free) optimal $d = 1$. The above only needs that $K \geq e^G$ for any fixed $G \geq 1$. It does not require K to be asymptotically large. Naturally the higher the K, the more of these gains can be attributed solely to caching (rather than MAT). When the value of K is moderate, naturally MAT has an impact, in terms of per-user DoF.

[3] Similarly for $\gamma = K^{-(1-\epsilon)}, \epsilon \in (0,1]$, we get $\frac{\delta\left(d(\gamma)-d(\gamma=0)\right)}{\delta\gamma}\Big|_{\gamma=K^{-(1-\epsilon)}} \approx \frac{K^{1-\epsilon}}{(1-\epsilon)^2\log^2 K}$.

[4] Here we make the assumption that $1-\gamma \approx 1$, which is a soft approximation that allows for simplicity of expressions, and which reflects the reality of small γ (cf. [14]).

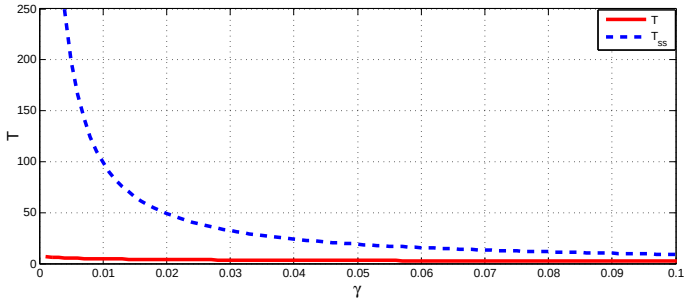

Figure 2.2: Single stream T_{ss} (no delayed CSIT, dotted line) vs. T after the introduction of delayed CSIT. Plot holds even for very large K, and the main gains appear for smaller values of γ.

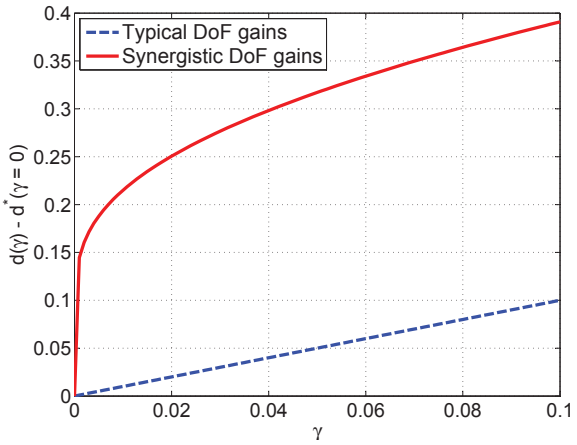

Figure 2.3: Typical gain $d(\gamma) - d^*(\gamma = 0)$ attributed solely to coded caching (dotted line) vs. synergistic gains derived here. Plot holds for large K, and the main gains appear for smaller values of γ.

2.2.2 Practical ramification: using coded caching to 'buffer' CSI

In addition to the substantial DoF gains that one can get by exploiting synergy, we also note that exploiting this interplay between caching and feedback timeliness, can additionally help alleviate the laborious task of sending feedback under the coherence period constraint. By using a modest γ, we are essentially endowing the system (for this specific setting that we are considering here) with a seemingly paradoxical ability of online buffering of CSI. To see this better, consider

$$\gamma_G^{'} \triangleq \arg\min_{\gamma^{'}} \{\gamma^{'} : d(\gamma^{'}) \geq \frac{1}{G}\} \tag{2.10}$$

which describes the minimum γ needed to achieve — in conjunction with delayed CSIT — a certain gap $G \geq 1$ from the interference-free (cache-free) optimal (associated to perfect real-time CSIT), for which we can quickly calculate that $\gamma_G^{'} = e^{-(G-\epsilon_K+\epsilon_\infty)}$, which for larger K, converges to[5] the aforementioned $\gamma_G^{'} = e^{-G}$.

2.3 Example of scheme

The scheme that achieves the above results, will be presented in its general form in Section 2.4. To offer some intuition on the schemes, we provide here an example (for the case of $K = N = 3, M = 1$).

In our examples here, for simplicity, the three distinct files in the library will be relabeled as $W_1 = A, W_2 = B, W_3 = C$. Finally we assume the worst-case request where A, B, C are requested by user 1, 2, 3, respectively.

Placement phase

After splitting each file into three equally-sized subfiles as $A = (A_1, A_2, A_3), B = (B_1, B_2, B_3), C = (C_1, C_2, C_3)$, we fill the cache Z_k of each user k, as follows $Z_k = (A_k, B_k, C_k), k = 1, 2, 3$.

Delivery phase

To satisfy the requests A, B, C, we must deliver the following three XORs $A_2 \oplus B_1, A_3 \oplus C_1, B_3 \oplus C_2$, each having size $\frac{f}{3}$ bits, and each intended for two users (users 1-2, 1-3, and 2-3 respectively). These messages, which we respectively denote as AB, AC, BC, are delivered by employing the last two phases of the ($K = 3$) MAT algorithm in [1].

Phase 2: Before transmission, we split each XOR into two *mini parts* as $AB = (AB_1, AB_2), AC = (AC_1, AC_2), BC = (BC_1, BC_2)$, where now each mini part has size $\frac{f}{6}$ bits. Then we form the following three vectors[6]

$$\boldsymbol{x}_1 = \begin{bmatrix} AB_1 \\ AB_2 \\ 0 \end{bmatrix}, \boldsymbol{x}_2 = \begin{bmatrix} AC_1 \\ AC_2 \\ 0 \end{bmatrix}, \boldsymbol{x}_3 = \begin{bmatrix} BC_1 \\ BC_2 \\ 0 \end{bmatrix} \tag{2.11}$$

which we transmit sequentially. This allows each user to receive different combinations of scalars, as we show below (ignoring noise)

$$\begin{aligned} &\text{user 1: } L_1(AB_1, AB_2), L_4(AC_1, AC_2), L_7(BC_1, BC_2) \\ &\text{user 2: } L_2(AB_1, AB_2), L_5(AC_1, AC_2), L_8(BC_1, BC_2) \\ &\text{user 3: } L_3(AB_1, AB_2), L_6(AC_1, AC_2), L_9(BC_1, BC_2) \end{aligned}$$

[5] The above holds because $T = H_K - H_\Gamma = G = \log(\frac{1}{\gamma}) + \epsilon_K - \epsilon_{K\gamma} \leq \log(\frac{1}{\gamma}) + \epsilon_K - \epsilon_\infty$.

[6] Here we assume a mapping from bits to QAM.

where we note that $L_7(BC_1, BC_2)$ is useful to both users 2 and 3 in decoding BC_1, BC_2, and similarly $L_5(AC_1, AC_2)$ is useful to both user 1 and 3, and $L_3(AB_1, AB_2)$ is useful to users 1 and 2. Recall that $|L_1(AB_1, AB_2)| = |AB_1| = \frac{f}{6}$ bits, which means that $\text{dur}(\boldsymbol{x}_i) = \frac{1}{6}, i = 1, \cdots, 3$, and that this phase has duration $\frac{3}{6}$.

Phase 3: We now transmit, using one antenna only, first a linear combination $f_1\big(L_3(AB_1, AB_2),\ L_5(AC_1, AC_2),\ L_7(BC_1, BC_2)\big)$, and then a second combination $f_2\big(L_3(AB_1, AB_2),\ L_5(AC_1, AC_2),\ L_7(BC_1, BC_2)\big)$, both of which carry a fully-common message (i.e., a message of order 3) that is useful to all users. Thus after the sequential transmission of

$$\begin{aligned} \boldsymbol{x}_4 &= [f_1, 0, 0]^T \\ \boldsymbol{x}_5 &= [f_2, 0, 0]^T \end{aligned} \tag{2.12}$$

each user can decode. To see this, let us focus on user 1. From before transmission of $\boldsymbol{x}_4, \boldsymbol{x}_5$, user 1 knew $L_7(BC_1, BC_2)$ (as this was its received signal during the third transmission, for BC). Now, with $\boldsymbol{x}_4, \boldsymbol{x}_5$, user 1 has two observations regarding L_3, L_5, L_7. L_7 can be removed, hence now user 1 can resolve $L_3(AB_1, AB_2)$ and $L_5(AC_1, AC_2)$. Thus now user 1 can combine $L_1(AB_1, AB_2)$ and $L_3(AB_1, AB_2)$ to resolve AB_1 and AB_2, to thus recover $A_2 \oplus B_1$. Similarly user 1 can combine $L_4(AC_1, AC_2)$ and $L_5(AC_1, AC_2)$ to resolve AC_1 and AC_2, to thus recover $A_3 \oplus C_1$. From $A_2 \oplus B_1$ user 1 can use its cache (which includes B_1) to recover A_2, and similarly from $A_3 \oplus C_1$ the same user can use its cache (which includes C_1) to recover A_3, and thus recover the desired A. Similarly users 2 and 3 can recover files B and C respectively.

With $\text{dur}(\boldsymbol{x}_4) = \text{dur}(\boldsymbol{x}_5) = \frac{1}{6}$, the second phase has duration $\frac{2}{6}$ and the total two-phase transmission has an overall duration

$$T = \frac{1}{2} + \frac{1}{3} = \frac{5}{6} \tag{2.13}$$

which matches the derived $T(\gamma) = T(\frac{1}{3}) = H_K - H_{K\gamma} = H_3 - H_2 = \frac{5}{6}$.

2.4 Cache-aided prospective-hindsight scheme

We proceed to describe some of the details of the scheme, and how it combines the coded caching algorithm in [2] (placement, folding-and-delivery, and decoding) with the MAT algorithm in [1].

Key idea behind the scheme First let us briefly describe the idea behind our simple scheme. As Figure 2.4 implies, the scheme starts by first applying the Maddah-Ali and Niesen (MN) sub-packetization based scheme [2] in order to place contents (sub-packets) in the caches, and to generate order-$(K\gamma + 1)$ messages in the form of XORs of the sub-packets, where each of these XORs

is meant for $K\gamma+1$ users. These XORs are delivered by the well known MAT method [1], and in particular the MAT variant that delivers order-$(K\gamma+1)$ messages. This allows us to skip the first $K\gamma$ phases of the MAT scheme, which happen to have the longest time duration. This is why the impact of even small caches (small γ) is substantial. Upon MAT decoding, we simply proceed with decoding based on the algorithm in [2]. In the end, the key idea is that the caching algorithm creates a multi-destination delivery problem that is the same as that which is efficiently solved by the last stages of the MAT scheme.

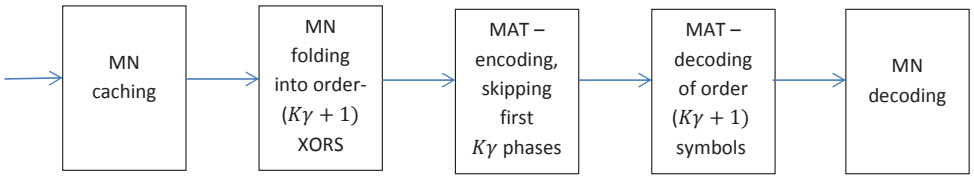

Figure 2.4: Basic composition of scheme. 'MAT encoding/decoding' corresponds to the scheme in [1], while 'MN caching/folding' corresponds to the scheme in [2].

2.4.1 Placement phase

The placement phase is taken from [2], where each of the N files $\{W_n\}_{n=1}^N$ ($|W_n| = f$ bits) in the library, is equally split into $\binom{K}{\Gamma}$ subfiles as follows, $W_n = \{W_{n,\tau}\}_{\tau\in\Psi_\Gamma}$, where $\Psi_\Gamma \triangleq \{\tau \subset [K] \ : \ |\tau| = \Gamma\}$, so each subfile has size

$$|W_{n,\tau}| = \frac{f}{\binom{K}{\Gamma}} \text{ bits.} \tag{2.14}$$

Based on the above, the caches are filled as follows $Z_k = \{W_{n,\tau}\}_{n\in[N],\tau\in\Psi_\Gamma,k\in\tau}$, so that each subfile $W_{n,\tau}$ is stored in Z_k as long as $k \in \tau$.

2.4.2 Delivery

At the beginning of the delivery phase, the transmitter must deliver each requested file W_{R_k}, by delivering the constituent subfiles $\{W_{R_k,\tau}\}_{k\notin\tau}$ to the corresponding user k. Thus, as in [2], we must send the entire set $\mathcal{X}_\Psi \triangleq \{X_\psi\}_{\psi\in\Psi_{\Gamma+1}}$, where each XOR is of the form $X_\psi \triangleq \oplus_{k\in\psi} W_{R_k,\psi\setminus\{k\}}, \psi \in \Psi_{\Gamma+1}$. There are $|\mathcal{X}_\Psi| = \binom{K}{\Gamma+1}$ folded messages (XORs), and each has size (cf. (4.9))

$$|X_\psi| = |W_{R_k,\tau}| = \frac{f}{\binom{K}{\Gamma}} \text{ bits.} \tag{2.15}$$

To deliver $\{X_\psi\}_{\psi\in\Psi_{\Gamma+1}}$, we will employ the last $K-\Gamma$ phases (phase $j=\Gamma+1,\ldots,K$) of the MAT algorithm. Each phase j delivers order-j folded messages. We describe the content that is carried during each of these phases.

Phase $\Gamma+1$*:* In this first phase of duration $T_{\Gamma+1}$, the information in $\{X_\psi\}_{\psi\in\Psi_{\Gamma+1}}$ is delivered by $\boldsymbol{x}_t,\ t\in[0,T_{\Gamma+1}]$, which can also be rewritten in the form of a sequential transmission of shorter-duration K-length vectors $\boldsymbol{x}_\psi=[x_{\psi,1},\ldots,x_{\psi,K-\Gamma},0,\ldots,0]^T$ for different ψ, where each vector $\boldsymbol{x}_\psi$ carries exclusively the information from each X_ψ, and where this information is uniformly split among the $K-\Gamma$ independent scalar entries $x_{\psi,i},\ i=1,\ldots,K-\Gamma$, each carrying $\frac{|X_\psi|}{(K-\Gamma)}=\frac{f}{\binom{K}{\Gamma}(K-\Gamma)}$ bits (cf. (6.14)). Hence, the duration of each $\boldsymbol{x}_\psi$ is $\mathrm{dur}(\boldsymbol{x}_\psi)=\frac{|X_\psi|}{(K-\Gamma)f}=\frac{1}{\binom{K}{\Gamma}(K-\Gamma)}$. Given that $|\mathcal{X}_\Psi|=\binom{K}{\Gamma+1}$, then

$$T_{\Gamma+1}=\binom{K}{\Gamma+1}\mathrm{dur}(\boldsymbol{x}_\psi)=\frac{1}{\Gamma+1}. \tag{2.16}$$

After each transmission of $\boldsymbol{x}_\psi$, each user $k\in[K]$ receives a linear combination $L_{\psi,k}$ of the transmitted $K-\Gamma$ symbols $x_{\psi,1},x_{\psi,2},\ldots,x_{\psi,K-\Gamma}$.

Next an additional $K-\Gamma-1$ signals $L_{\psi,k'},\ k'\in[K]\backslash\psi$ (linear combinations of $x_{\psi,1},x_{\psi,2},\ldots,x_{\psi,K-\Gamma}$ as received — up to noise level — at each user $k'\in[K]\backslash\psi$) will be sent, which will help each user $k\in\psi$ to resolve $x_{\psi,1},x_{\psi,2},\ldots,x_{\psi,K-\Gamma}$. This will be done in the next phase $j=\Gamma+2$.

Phase $\Gamma+2$*:* The challenge now is for signals

$$\boldsymbol{x}_{c,t},\ t\in(T_{\Gamma+1},T_{\Gamma+1}+T_{\Gamma+2}]$$

to convey all the messages of the form $L_{\psi,k'},\ \forall k'\in[K]\backslash\psi,\ \forall\psi\in\Psi_{\Gamma+1}$ to each receiver $k\in\psi$. Note that each of the above linear combinations, is now — during this phase — available (up to noise level) at the transmitter. Let

$$\Psi_{\Gamma+2}=\{\psi\in[K]\ :\ |\psi|=\Gamma+2\} \tag{2.17}$$

and consider for each $\psi\in\Psi_{\Gamma+2}$, a transmitted vector

$$\boldsymbol{x}_\psi=[x_{\psi,1},\ldots,x_{\psi,K-\Gamma-1},0,\ldots,0]^T$$

which carries the contents of $\Gamma+1$ different linear combinations

$$f_i(\{L_{\psi\backslash\{k\},k}\}_{k\in\psi}),i=1,\ldots,\Gamma+1$$

of the $\Gamma+2$ elements $\{L_{\psi\backslash\{k\},k}\}_{\forall k\in\psi}$ created by the transmitter. The linear combination coefficients defining each linear-combination function f_i, are predetermined and known at each receiver. The transmission of $\{\boldsymbol{x}_\psi\}_{\forall\psi\in\Psi_{\Gamma+2}}$ is sequential.

It is easy to see that there is a total of $(\Gamma+1)\binom{K}{\Gamma+2}$ symbols of the form $f_i(\{L_{\psi\backslash\{k\},k}\}_{k\in\psi}),i=1,\ldots,\Gamma+1,\ \psi\in\Psi_{\Gamma+2}$, each of which can be considered

as an order-$(\Gamma+2)$ signal intended for $\Gamma+2$ receivers in ψ. Using this, and following the same steps used in phase $\Gamma+1$, we calculate that

$$T_{\Gamma+2} = \binom{K}{\Gamma+2} \text{dur}(\boldsymbol{x}_\psi) = T_{\Gamma+1} \frac{\Gamma+1}{\Gamma+2}. \tag{2.18}$$

We now see that for each ψ, each receiver $k \in \psi$ recalls their own observation $L_{\psi\backslash\{k\},k}$ from the previous phase, and removes it from all the linear combinations $\{f_i(\{L_{\psi\backslash\{k\},k}\}_{\forall k\in\psi})\}_{i=1,\ldots,\Gamma+1}$, thus now being able to acquire the $\Gamma+1$ independent linear combinations $\{L_{\psi\backslash\{k'\},k'}\}_{\forall k'\in\psi\backslash\{k\}}$. It holds for each other user $k' \in \psi$.

After this phase, we use $L_{\psi,k}, \psi \in \Psi_{\Gamma+2}$ to denote the received signal at receiver k. Like before, each receiver $k, k \in \psi$ needs $K-\Gamma-2$ extra observations of $x_{\psi,1}, \ldots, x_{\psi,K-\Gamma-1}$ which will be seen from $L_{\psi,k'}, \forall k' \notin \psi$, which will come from order-$(\Gamma+3)$ messages that are created by the transmitter and which will be sent in the next phase.

Phase j $(\Gamma+3 \leq j \leq K)$: Generalizing the described approach to any phase $j \in [\Gamma+3, \ldots, K]$, we will use

$$\boldsymbol{x}_{c,t},\ t \in [\sum_{i=\Gamma+1}^{j-1} T_i,\ \sum_{i=\Gamma+1}^{j} T_i]$$

to convey all the messages of the form $L_{\psi,k'},\ \forall k' \in [K]\backslash\psi,\ \forall \psi \in \Psi_{j-1}$ to each user $k \in \psi$. For each $\psi \in \Psi_j \triangleq \{\psi \in [K]\ :\ |\psi| = j\}$, each transmitted vector

$$\boldsymbol{x}_\psi = [x_{\psi,1}, \ldots, x_{\psi,K-j-1}, 0, \ldots, 0]^T$$

will carry the contents of $j-1$ different linear combinations $f_i(\{L_{\psi\backslash\{k\},k}\}_{k\in\psi}), i = 1, \ldots, j-1$ of the j elements $\{L_{\psi\backslash\{k\},k}\}_{\forall k\in\psi}$ created by the transmitter. After the sequential transmission of $\{\boldsymbol{x}_\psi\}_{\forall\psi\in\Psi_j}$, each receiver k can obtain the $j-1$ independent linear combinations $\{L_{\psi\backslash\{k'\},k'}\}_{\forall k'\in\psi\backslash\{k\}}$. The same holds for each other user $k' \in \psi$. As with the previous phases, we can see that

$$T_j = T_{\Gamma+1} \frac{\Gamma+1}{j},\ j = \Gamma+3, \ldots, K. \tag{2.19}$$

This process terminates with phase $j = K$, during which each

$$\boldsymbol{x}_\psi = [x_{\psi,1}, 0, 0, \ldots, 0]^T$$

carries a single scalar that is decoded easily by all. Based on this, backwards decoding will allow for users to retrieve $\{X_\psi\}_{\psi\in\Psi_{\Gamma+1}}$. This is described below.

2.4.3 Decoding

Each receiver k will backwards reconstruct the sets of overheard equations, which now take the form

$$\begin{gathered}\{L_{\psi,k'},\ \forall k' \in [K]\backslash\psi\}_{\forall\psi\in\Psi_K} \\ \vdots \\ \downarrow \\ \{L_{\psi,k'},\ \forall k' \in [K]\backslash\psi\}_{\forall\psi\in\Psi_{\Gamma+2}}\end{gathered}$$

until phase $\Gamma+2$, thus gaining enough observations to recover the original $K-\Gamma$ symbols $x_{\psi,1}, x_{\psi,2}, \ldots, x_{\psi,K-\Gamma}$ that fully convey X_ψ, hence each user k can reconstruct their own set $\{W_{R_k,\psi\backslash\{k\}}\}_{\psi\in\Psi_{\Gamma+1}}$ by using Z_k, which in turn allows each user k to reconstruct their requested W_{R_k}.

2.4.4 Calculation of T

To calculate T, combining (3.40), (2.18) and (3.46) gives that

$$T = \sum_{j=\Gamma+1}^{K} T_j = T_{\Gamma+1} \sum_{j=\Gamma+1}^{K} \frac{\Gamma+1}{j} = \sum_{j=\Gamma+1}^{K} \frac{1}{j} = H_K - H_\Gamma.$$

2.5 Conclusions

The work explored the interesting connections between retrospective transmission schemes which alleviate the effect of the delay in knowing the channel, and coded caching schemes which alleviate the effect of the delay in knowing the content destination. These connections are at the core of the coded caching paradigm, and their applicability can extend to different settings. For the MISO-BC setting, the optimal cache-aided DoF were identified within a multiplicative factor of 4. The result also implies that a very modest amount of caching can have a substantial impact on performance, as well as can go a long way toward removing the burden of acquiring timely CSIT.

2.6 Appendix - A novel scheme that accentuates the retrospective nature of caching and communicating with delayed CSIT

Another way to offer small insight on the relationship between communicating with retrospective knowledge of the channel, and caching-and-communicating with retrospective knowledge of the destination of data, is found in the following example (done here for $N = K = 3, \Gamma = 1$) which describes how we can design an entirely new caching algorithm, based on the MAT algorithm. Now we will present a new caching strategy.

We use A, B, C to denote the database at the transmitter, where each file is evenly split into three subfiles, i.e., $A = (A_1, A_2, A_3)$, the same with B and C. Each subfile has size $\frac{1}{3}\log P$ bits. Towards this, the caches are filled as follow

$$Z_1 = \{w_1, w_4, w_7\}$$
$$Z_2 = \{w_2, w_5, w_8\}$$
$$Z_3 = \{w_3, w_6, w_9\}$$

where w_1, w_2, w_3 are three linearly independent combinations of A_1, A_2, A_3, similarly, w_4, w_5, w_6 and w_7, w_8, w_9 are three linearly independent combinations of B_1, B_2, B_3 and C_1, C_2, C_3, respectively. The constant coefficients are shared with the users.

Consider the worst case that A, B, C are requested by user 1, 2, 3, respectively. From $w_2 \oplus w_4$, we can see that user 1 can get a linear independent combination of three desired subfiles since w_4 is available in Z_1, while similarly, user 2 can get a equation of three desired subfiles. In the same way, from $w_3 \oplus w_7$, $w_6 \oplus f_8$, user 1 can recover w_3, user 2 can recover w_6 and user 3 can recover w_7, w_8. Thus, user 1 is able to solve A_1, A_2, A_3 from w_1, w_2, w_3 and then obtain A, user 2 and 3 acts the same. Hence,

- $w_2 \oplus w_4$ is desired by user 1 and 2;
- $w_3 \oplus w_7$ is desired by user 1 and 3;
- $w_6 \oplus w_8$ is desired by user 2 and 3;

These three messages should somehow be sent.

The transmission consists two phase.

Phase 1: $w_2 \oplus w_4$, $w_3 \oplus w_7$, $w_6 \oplus w_8$ are equally split into two supports of $\boldsymbol{x}_1 = \{x_{1,1}, x_{1,2}, 0\}, \boldsymbol{x}_2 = \{x_{2,1}, x_{2,2}, 0\}, \boldsymbol{x}_3 = \{x_{3,1}, x_{3,2}, 0\}$, respectively, and each is sent from two transmit antennas. Note that each support has size $\frac{1}{6}\log P$ bits. Thus, each user receives a combination of every two supports as follows,

$$\begin{aligned}
&\text{user 1 receives: } L_1(x_{1,1}, x_{1,2}) \quad L_4(x_{2,1}, x_{2,2}) \quad L_7(x_{3,1}, x_{3,2}) \\
&\text{user 2 receives: } L_2(x_{1,1}, x_{1,2}) \quad L_5(x_{2,1}, x_{2,2}) \quad L_8(x_{3,1}, x_{3,2}) \\
&\text{user 3 receives: } L_3(x_{1,1}, x_{1,2}) \quad L_6(x_{2,1}, x_{2,2}) \quad L_9(x_{3,1}, x_{3,2})
\end{aligned}$$

We can see that $L_7(x_{3,1}, x_{3,2})$ can be another observation for user 2 and 3 to solve $x_{3,1}, x_{3,2}$, thus it is useful for both user. Similarly, $L_5(x_{2,1}, x_{2,2})$ is useful for both user 1 and 3, $L_3(x_{1,1}, x_{1,2})$ is useful for both user 1 and 2.

Phase 2: Based on the first phase, we define f_i as two linear combinations of $L_3(x_{1,1}, x_{1,2}), L_5(x_{2,1}, x_{2,2}), L_7(x_{3,1}, x_{3,2})$ known at the transmitter after phase 1 and the coefficients are enjoyed by all the users,

$$f_i(L_3(x_{1,1}, x_{1,2}), L_5(x_{2,1}, x_{2,2}), L_7(x_{3,1}, x_{3,2})), i = 1, 2$$

Hence, common messages f_i are sent in this phase. As a result, each user can remove the known equation and obtain two other intended observations.

Consequently, from the two-phase transmission, we have

$$T = \frac{1}{2} + \frac{1}{3} = \frac{5}{6} \tag{2.20}$$

Our scheme is built on MAT algorithm. Instead of storing clean information in the caches, e.g., A_1, here a novel caching strategy that combinations of the information from the sourcing files are pre-filled is applied. Combined with rate-splitting approach under retrospective knowledge of CSIT, the caching content that are simultaneously useful for multi-users are sent back, allowing each user has enough observations to resolve the new data not be cached.

The similarities between communicating with and without caching is described as follows,

1. At the receivers, each receiver has received or stored undesired messages useful for multi-users;

2. In the presence of retrospective knowledge of CSIT, the side information are learnt at the transmitter and delivered in a manner that each user is able to get the desired observations.

2.7 Appendix - Vanishing fraction of delayed CSIT

In the following we briefly explore how caching allows for a reduced D-CSIT load.

The MAT-inspired schemes that we use and which we describe in Section 2.4, can have up to K phases which are of decreasing time duration and which use a decreasing number of transmit antennas. Essentially each phase is lighter than the previous one, in terms of implementation difficulty. What we will see is that caching will allow us to bypass the first Γ phases, which are the longest and most intensive, leaving us with the remaining $K - \Gamma$ communication phases that are easier to support with delayed feedback because they involve fewer transmissions, with fewer transmit antennas and to fewer users, and thus involve fewer D-CSIT scalars that must be communicated.

In brief — after normalization to account for the condition that each user receives a total of $\log_2(P)$ bits of data — each phase $j = \Gamma + 1, \Gamma + 2, \ldots, K$ will have a *normalized* duration $T_j = \frac{1}{j}$. During each phase j, we will need to send D-CSIT that describes the channel vectors for $K - j$ users, and during this same phase the transmitted vectors will have support $K - j + 1$ because only $K - j + 1$ transmit antennas are active. Thus during phase j, there will be a need to send $T_j(K-j+1)(K-j) = \frac{1}{j}(K-j+1)(K-j)$ D-CSIT scalars,

and thus a need to send D-CSIT for up to a total of

$$L(\Gamma) = \sum_{j=\Gamma+1}^{K} \frac{1}{j}(K-j+1)(K-j)$$
$$= (K^2+K)(H_K - H_\Gamma) - \frac{K(1-\gamma)(3K-K\gamma-1)}{2}$$

channel scalars, while in the absence of caching (corresponding to $\Gamma = 0$), we will have to send D-CSIT on

$$L(\Gamma=0) = \sum_{j=1}^{K} \frac{1}{j}(K-j+1)(K-j)$$
$$= (K^2+K)H_K - \frac{3K^2}{2} + \frac{K}{2}$$

channel scalars.

To reflect the frequency of having to gather D-CSIT, and to provide a fair comparison between different schemes of different performance that manage to convey different amounts of actual data to the users, we consider the measure $Q(\Gamma)$ that normalizes the above number $L(\Gamma)$ of full D-CSIT scalars, by the coherence period T_c and by the total number of full data symbols sent. In our case, under the assumption that each user receives a total of $\log_2(P)$ bits, the total number of full data symbols sent is K, and thus we have

$$Q(\Gamma) = \frac{L(\Gamma)}{T_c K} = \frac{(K^2+K)(H_K - H_\Gamma) - \frac{K(1-\gamma)(3K-K\gamma-1)}{2}}{T_c K}$$

while without caching, we have

$$Q(\Gamma=0) = \frac{L(\Gamma)}{T_c K} = \frac{(K+1)H_K - \frac{3}{2}K + \frac{1}{2}}{T_c}.$$

Consequently we see that in the large K limit,

$$Q(\Gamma) \to \frac{K\left(\log(\frac{1}{\gamma}) - \frac{3}{2} + 2\gamma - 2\gamma^2\right)}{T_c}$$

$$Q(\Gamma=0) \to \frac{1}{T_c} K \log(K)$$

which implies that

$$\lim_{K\to\infty} \frac{Q(\Gamma)}{Q(\Gamma=0)} = 0$$

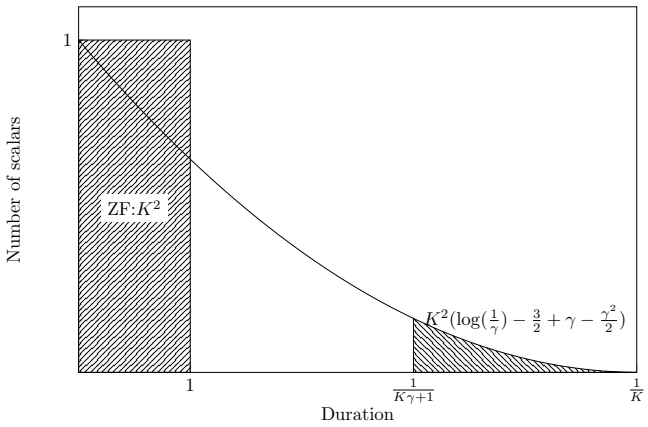

Figure 2.5: Illustration of the vanishing fraction of D-CSIT cost, due to caching.

which in turn tells us that as K increases, for any fixed γ, caching allows for a substantial reduction (down to a vanishingly small portion) from the original cost of D-CSIT. This is illustrated in Figure 2.5.

This reduction is important because retrospective delayed-feedback methods suffer from an increased cost of supporting their CSIT requirements (cf. [68]) (albeit at the benefit of allowing substantial delays in the feedback mechanisms); after all, in the presence of perfect CSIT and zero forcing (no caching), the same cost is

$$Q_{ZF} = \frac{K^2}{T_c K} = \frac{K}{T_c}$$

which gives that

$$\lim_{K\to\infty} \frac{Q(\Gamma = 0)}{Q_{ZF}} = \infty$$

which in turn verifies the above claim, and shows that the increase in the cost of supporting the D-CSIT (without caching) can be unbounded compared to ZF methods. On the other hand, we see that

$$\lim_{K\to\infty} \frac{Q(\Gamma)}{Q_{ZF}} = \log(\frac{1}{\gamma} - \frac{3}{2} + 2\gamma - 2\gamma^2)$$

which means that

$$\lim_{K\to\infty} \frac{Q(\Gamma)}{Q_{ZF}} < 1,\ \gamma \geq \frac{1}{10}.$$

One interesting conclusion that comes out of this, is that caching can allow for full substitution of current CSIT (as we have seen in Section 2.2.2), with a very substantial reduction of the cost of D-CSIT as well, where for $\gamma \geq \frac{1}{10}$ this cost is even less than that of the very efficient ZF, which has to additionally

deal though with harder-to-obtain current CSIT. This cost reduction is also translated into a reduction in the cost of disseminating global channel state information at the receivers (global CSIR), where each receiver must now know (again with delay that is allowed to be large) the CSIR of only a fraction of the other receivers.

2.8 Appendix - Proof of Lemma 2.1 (Lower bound on T^*)

Our aim is to lower bound the duration T, that guarantees the delivery of K different files to K users, via a MISO broadcast channel with delayed CSIT, and in the presence of K caches, each of size Mf. Let T_2 be the duration needed to resolve the simpler setting where we want to serve $s \leq K$ different files to s users, again each in the presence of their own caches. Naturally $T_2 \leq T$ since we ignore the interference from the remaining $K-s$ users (whose requests are ignored). Now let T_3 ($T_3 \leq T_2$), be the duration needed to resolve the same problem, except that now all the s caches are merged, and each of the s users has access to all s caches. We choose to repeat this last experiment $\lfloor \frac{N}{s} \rfloor$ times, thus spanning a total duration of $T_3 \lfloor \frac{N}{s} \rfloor$. At this point, we transfer to the equivalent setting of the s-user MISO BC with delayed CSIT, and a side-information multicasting link to the receivers, of capacity d_m (files per time slot). Under the assumption that in this latter setting, decoding happens at the end of communication, and once we set

$$d_m T_3 \lfloor \frac{N}{s} \rfloor = sM \tag{2.21}$$

(which guarantees that the side information from the side link, throughout the communication process, matches the maximum amount of information in the caches), we then have that

$$T_3 \lfloor \frac{N}{s} \rfloor d'(d_m) \geq \lfloor \frac{N}{s} \rfloor s \tag{2.22}$$

where $d'(d_m)$ is any upper bound on the above s-user MISO BC channel with delayed CSIT and the side link. Using the bound $d'(d_m) = \frac{s}{H_s}(1+d_m)$ from [69] and applying (2.21), we get $d'(d_m) = \frac{s}{H_s}(1 + \frac{sM}{T_3 \lfloor \frac{N}{s} \rfloor})$ and thus we get

$$T_3 \lfloor \frac{N}{s} \rfloor \frac{s}{H_s}(1 + \frac{sM}{T_3 \lfloor \frac{N}{s} \rfloor}) \geq \lfloor \frac{N}{s} \rfloor s \tag{2.23}$$

which means that

$$T_3 \geq H_s - \frac{sM}{\lfloor \frac{N}{s} \rfloor} \tag{2.24}$$

which implies that the optimal T^*, for the original s-user problem, is bounded as

$$T^* \geq T_3 \geq H_s - \frac{sM}{\lfloor \frac{N}{s} \rfloor}. \tag{2.25}$$

Maximization over all s, gives the desired result.

2.9 Appendix - Bounding the gap to optimal

This section presents the proof that the gap $\frac{T(\gamma)}{T^*(\gamma)}$, between the achievable $T(\gamma)$ and the optimal $T^*(\gamma)$, is always upper bounded by 4, which also serves as the proof of identifying the optimal $T^*(\gamma)$ within a factor of 4.

First recall from Theorem 2.1 that $T(\gamma) = H_K - H_{K\gamma}$ and from Lemma 2.1 that $T^*(\gamma) \geq \max_{s \in \{1,\ldots,\lfloor \frac{N}{M} \rfloor\}} H_s - \frac{Ms}{\lfloor \frac{N}{s} \rfloor}$. We want to prove that

$$\frac{T(\gamma)}{T^*(\gamma)} < 4, \ \forall K, \forall \Gamma = 1, 2, \ldots, K-1 \tag{2.26}$$

and the proof will be split into three cases: case 1 for $\gamma \in [\frac{1}{K}, \frac{1}{36}]$, case 2 for $\gamma \in [\frac{1}{36}, \frac{1}{2}]$, and case 3 for $\gamma \in [\frac{1}{2}, \frac{K-1}{K}]$. Recall that γ is bounded as $\gamma \geq \frac{1}{K}$.

Case 1 ($\gamma \leq \frac{1}{36}$)

First note that having $\gamma \leq \frac{1}{36}$ implies $K \geq 36$. To prove (2.26), we see that

$$\frac{T}{T^*} \leq \max_{\gamma \in [\frac{1}{K}, \frac{1}{36}] \cap (\mathbb{Z}/K)} \frac{H_K - H_{K\gamma}}{\max_{s \in [1,K] \cap \mathbb{Z}} H_s (1 - \frac{Ms}{H_s \lfloor \frac{N}{s} \rfloor})} \tag{2.27}$$

$$\leq \max_{\gamma \in [\frac{1}{K}, \frac{1}{36}] \cap (\mathbb{Z}/K)} \frac{H_K - H_{K\gamma}}{\max_{s \in [6, \lfloor \sqrt{K} \rfloor] \cap \mathbb{Z}} (H_s - \frac{Ms}{\lfloor \frac{N}{s} \rfloor})} \tag{2.28}$$

$$\leq \max_{\gamma \in [\frac{1}{K}, \frac{1}{36}]} \frac{H_K - H_{K\gamma}}{\max_{s \in [6, \lfloor \sqrt{K} \rfloor] \cap \mathbb{Z}} (H_s - \frac{Ms}{\lfloor \frac{N}{s} \rfloor})} \tag{2.29}$$

$$\leq \max_{\gamma \in [\frac{1}{K}, \frac{1}{36}]} \frac{\log(\frac{1}{\gamma}) + \epsilon_{36} - \epsilon_\infty}{\max_{s \in [6, \lfloor \sqrt{K} \rfloor] \cap \mathbb{Z}} \log(s) + \epsilon_\infty - \gamma s^2 \frac{7}{6}} \tag{2.30}$$

where (2.28) holds because $H_s - \frac{Ms}{\lfloor \frac{N}{s} \rfloor} < 0$ when $s > \lfloor \frac{1}{\gamma} \rfloor$ and because we reduced the maximizing region for s, where (2.29) holds because we increased the maximizing region for γ, and where (2.30) holds because ϵ_K decreases with K, because $H_K - \log(K) \leq \epsilon_{36}$, $H_{K\gamma} - \log(K\gamma) > \epsilon_\infty$, $H_s > \log(s) + \epsilon_\infty$, and

because $(\lfloor \frac{N}{s} \rfloor)/\frac{N}{s} \geq \frac{6}{7}$, $s \leq \frac{N}{6}$ (recall that $s \leq \lfloor \sqrt{K} \rfloor \leq \frac{K}{6} \leq \frac{N}{6}$). Continuing from (2.30), we have that

$$\frac{T}{T^*} \leq \max_{s_c \in [6, \lfloor \sqrt{K} \rfloor] \cap \mathbb{Z}} \max_{\gamma \in [\frac{1}{(s_c+1)^2}, \frac{1}{s_c^2}]} \frac{\log(\frac{1}{\gamma}) + \epsilon_{36} - \epsilon_\infty}{\log(s_c) + \epsilon_\infty - \gamma s_c^2 \frac{7}{6}} \tag{2.31}$$

because $\max_{s \in [6, \lfloor \sqrt{K} \rfloor] \cap \mathbb{Z}} \log(s) + \epsilon_\infty - \gamma s^2 \frac{7}{6} \geq \log(s_c) + \epsilon_\infty - \gamma s_c^2 \frac{7}{6}$ for any γ and for any $s_c \in [6, \lfloor \sqrt{K} \rfloor] \cap \mathbb{Z}$. The split of the maximization $\max_{\gamma \in [\frac{1}{K}, \frac{1}{36}]}$ into the double maximization $\max_{s_c \in [6, \lfloor \sqrt{K} \rfloor] \cap \mathbb{Z}} \max_{\gamma \in [\frac{1}{(s_c+1)^2}, \frac{1}{s_c^2}]}$ reflects the fact that we heuristically choose[7] $s = s_c \in \mathbb{Z}$ when $\gamma \in [\frac{1}{(s_c+1)^2}, \frac{1}{s_c^2}]$. Now we perform a simple change of variables, introducing a real valued s' ($s' \triangleq \sqrt{\frac{1}{\gamma}}$) such that $\gamma = \frac{1}{s'^2}$. Hence, a γ range of $\gamma \in [\frac{1}{(s_c+1)^2}, \frac{1}{s_c^2}]$, corresponds to an s' range of $s' \in [s_c, s_c + 1]$. Hence we rewrite (2.31) using this change of variables, to get

$$\frac{T}{T^*} \leq \max_{s_c \in [6, \lfloor \sqrt{K} \rfloor] \cap \mathbb{Z}} \max_{s' \in [s_c, s_c+1]} \frac{\log(s'^2) + \epsilon_{36} - \epsilon_\infty}{\log(s_c) + \epsilon_\infty - \frac{7}{6} \frac{s_c^2}{s'^2}} \tag{2.32}$$

$$\leq \max_{s_c \in [6, \lfloor \sqrt{K} \rfloor] \cap \mathbb{Z}} \underbrace{\frac{\log(s_c+1)^2 + \epsilon_{36} - \epsilon_\infty}{\log(s_c) + \epsilon_\infty - \frac{7}{6}}}_{f(s_c)} \tag{2.33}$$

$$\leq \frac{2 * \log(7) + \epsilon_{36} - \epsilon_\infty}{\log(6) + \epsilon_\infty - \frac{7}{6}} < 4 \tag{2.34}$$

where (2.33) holds because $\frac{s_c^2}{s'^2} \leq 1$, where (2.34) holds because $f(s_c)$ is decreasing in s_c.

Case 2 ($\gamma \in [\frac{1}{36}, \frac{1}{2}]$)

In the maximization of the lower bound, we will now choose $s = 1$.

For $K \geq 2$, we have

$$\frac{T}{T^*} \leq \frac{\log(\frac{1}{\gamma}) + \epsilon_2 - \epsilon_\infty}{1 - \gamma} =: f(\gamma) \tag{2.35}$$

because $H_K - \log(K) \leq \epsilon_2$, $H_{K\gamma} - \log(K\gamma) > \epsilon_\infty, \forall K \geq 2$.

For the above defined $f(\gamma)$, we calculate the derivative to take the form

$$\frac{df(\gamma)}{d\gamma} = \frac{\overbrace{1 - \gamma^{-1} - \log(\gamma) + \epsilon_2 - \epsilon_\infty}^{f'_N(\gamma)}}{\underbrace{(1-\gamma)^2}_{f'_D(\gamma)}}$$

[7] Essentially we choose an s that is approximately equal to $\lfloor \sqrt{\frac{1}{\gamma}} \rfloor$, and while this choice does not guarantee the exact maximizing s, it does manage to sufficiently raise the resulting lower bound.

where $f'_N(\gamma), f'_D(\gamma)$ respectively denote the numerator and denominator of this derivative. Since $f'_D(\gamma) > 0, \forall\gamma < 1$, and since

$$\frac{df'_N(\gamma)}{d\gamma} = \gamma^{-2} - \gamma^{-1} \geq 0, \forall\gamma \in [\frac{1}{36}, \frac{1}{2}].$$

To prove this, we use the following lemma, which we prove in Section 2.9.1 below.

Lemma 2.2 *Let $g'_N(\gamma)$ and $g'_D(\gamma)$ respectively denote the numerator and the denominator of the derivative $\frac{dg(\gamma)}{d\gamma}$ of some function $g(\gamma)$. If in the range $\gamma \in [\gamma_1, \gamma_2]$, $g'_N(\gamma)$ increases in γ, and if $g'_D(\gamma) > 0$, then*

$$\max_{\gamma\in[\gamma_1,\gamma_2]} g(\gamma) = \max\{g(\gamma = \gamma_1), g(\gamma = \gamma_2)\}. \tag{2.36}$$

We now continue with the main proof, and apply Lemma 2.2, to get

$$\max_{\gamma\in[\frac{1}{36},\frac{1}{2}]} f(\gamma) = \max\{f(\frac{1}{2}), f(\frac{1}{36})\} < 4 \tag{2.37}$$

which directly shows the desired $\frac{T}{T^*} < 4$ for $\frac{1}{36} \leq \gamma \leq \frac{1}{2}, K \geq 2$.

Case 3 ($\gamma \in [\frac{1}{2}, \frac{K-1}{K}]$)

In the maximization of the lower bound, we will again choose $s = 1$. Considering that now γ takes the values $\gamma = \frac{j}{K},\ j \in [\frac{K}{2}, K-1] \cap \mathbb{Z}$, we have

$$\begin{aligned}
\frac{T}{T^*} &\leq \frac{H_K - H_{K\gamma}}{1-\gamma} = \frac{H_K - H_{(K-j)}}{j/K} \\
&= \frac{1}{j}(\frac{K}{K-j+1} + \frac{K}{K-j+2} + \cdots + 1) \\
&= \frac{1}{j}(1 + \frac{j-1}{K-(j-1)} + 1 + \frac{j-2}{K-(j-2)} + \cdots + 1) \\
&= 1 + \frac{1}{j}(\frac{j-1}{K-(j-1)} + \frac{j-2}{K-(j-2)} + \cdots + \frac{1}{K-1}) \\
&< 2
\end{aligned}$$

because $j \leq \frac{K}{2}$.

This completes the proof for the entire case where $\Gamma = 1, 2, \ldots, K-1$.

2.9.1 Proof of Lemma 2.2

We first note that the condition $\frac{dg'_N(\gamma)}{d\gamma} \geq 0$ implies that $g'_N(\gamma)$ is increasing in γ. We also note that $g'_D(\gamma) \geq 0, \gamma \in [\gamma_1, \gamma_2]$ where naturally $\gamma_1 \leq \gamma_2$. We consider the following three cases.

Case 1 ($g'_N(\gamma_1) \geq 0$) If $g'_N(\gamma_1) \geq 0$ then $g'_N(\gamma) \geq 0$ for any $\gamma \in [\gamma_1, \gamma_2]$, which in turn means that $\frac{dg(\gamma)}{d\gamma} = \frac{g'_N(\gamma)}{g'_D(\gamma)} \geq 0, \gamma \in [\gamma_1, \gamma_2]$. This gives the desired

$$\max_{\gamma \in [\gamma_1, \gamma_2]} g(\gamma) = g(\gamma_2).$$

Case 2 ($g'_N(\gamma_1) < 0$ & $g'_N(\gamma_2) \leq 0$) For any $\gamma \in [\gamma_1, \gamma_2]$, then if $g'_N(\gamma_1) < 0$ & $g'_N(\gamma_2) \leq 0$ then $g'_N(\gamma) \leq 0$, thus $\frac{dg(\gamma)}{d\gamma} \leq 0$, which gives the desired

$$\max_{\gamma \in [\gamma_1, \gamma_2]} g(\gamma) = g(\gamma_1).$$

Case 3 ($g'_N(\gamma_1) < 0$ & $g'_N(\gamma_2) > 0$) For any $\gamma \in [\gamma_1, \gamma_2]$, then if $g'_N(\gamma_1) < 0$ & $g'_N(\gamma_2) > 0$, there exists a unique $\gamma = \gamma' \in [\gamma_1, \gamma_2]$ such that $g'_N(\gamma') = 0$. Hence $\frac{dg(\gamma)}{d\gamma} \leq 0, \forall \gamma \in [\gamma_1, \gamma']$ and $\frac{dg(\gamma)}{d\gamma} \geq 0, \forall \gamma \in [\gamma', \gamma_2]$. Consequently we have the desired

$$\max_{\gamma \in [\gamma_1, \gamma_2]} g(\gamma) = \max\{g(\gamma_1), g(\gamma_2)\}.$$

Combining the above three cases, yields the derived

$$\max_{\gamma \in [\gamma_1, \gamma_2]} g(\gamma) = \max\{g(\gamma_1), g(\gamma_2)\}$$

which completes the proof.

Chapter 3

The Interplay of Coded Caching and Current CSIT Feedback

This chapter here continues to consider the K-user cache-aided wireless K-antenna MISO symmetric broadcast channel with random fading, but now feedback is mixed: delayed feedback comes much later with perfect quality, and current CSIT feedback comes immediately but with reduced quality. We here analyze the throughput performance as a function of feedback statistics and cache size. In this setting, our work identifies the optimal cache-aided degrees-of-freedom (DoF) within a factor of 4, by identifying near-optimal schemes that exploit the new synergy between coded caching and delayed CSIT, as well as by exploiting the unexplored interplay between caching and feedback-quality.

The derived limits interestingly reveal that — the combination of imperfect quality current CSIT, delayed CSIT, and coded caching, guarantees that — the DoF gains have an initial offset defined by the quality of current CSIT, and then that the additional gains attributed to coded caching are exponential, in the sense that any linear decrease in the required DoF performance, allows for an exponential reduction in the required cache size.

3.1 Introduction

Our interest here is to explore coded caching, not in the original single-stream setting in [2], but rather in the feedback-aided multi-antenna wireless BC. This wireless and multi-antenna element now automatically brings to the fore a largely unexplored and involved relationship between coded caching and CSIT-type feedback quality. This relationship carries particular importance

because both CSIT and coded caching are powerful and crucial ingredients in handling interference, because they are both hard to implement individually, and because their utility is affected by one another (often adversely, as we will see). Our work tries to understand how CSIT and caching resources jointly improve performance, as well as tries to shed some light on the interplay between coded caching and feedback.

Motivation for the current work

A main motivation in [2] and in subsequent works, was to employ coded caching to remove interference. Naturally, in wireless networks, the ability to remove interference is very much linked to the quality and timeliness of the available feedback, and thus any attempt to further our understanding of the role of coded caching in these networks, stands to benefit from understanding the interplay between coded caching and (variable quality) feedback. This joint exposition becomes even more meaningful when we consider the connections that exist between feedback-usefulness and cached side-information at receivers, where principally the more side information receivers have, the less feedback information the transmitter might need.

As suggested before, this approach is also motivated by the fact that feedback is hard to get in a timely manner, and hence is typically far from ideal and perfect. Thus, given the underlying links between the two, perhaps the strongest reason to jointly consider coded caching and feedback, comes from the prospect of using coded caching to alleviate the constant need to gather and distribute CSIT, which — given typical coherence durations — is an intensive task that may have to be repeated hundreds of times per second during the transmission of content. This suggests that content prediction of a predetermined library of files during the night (off peak hours), and a subsequent caching of parts of this library content again during the night, may go beyond boosting performance, and may in fact offer the additional benefit of alleviating the need for prediction, estimation, and communication of CSIT during the day, whenever requested files are from the library. Our idea of exploring the interplay between feedback (timeliness and quality) and coded caching, hence draws directly from this attractive promise that content prediction, once a day, can offer repeated and prolonged savings in CSIT.

3.1.1 Cache-aided broadcast channel model

We remind the reader that we will consider the same MISO BC model as Chapter 2 (see Fig. 2.1), corresponding to the symmetric K-user multiple-input single-output (MISO) broadcast channel where a K-antenna transmitter, communicates to K single-antenna receiving users. Hence the channel model and caching model remain the same as in Chapter 2.

3.1.2 Coded caching and CSIT-type feedback

Communication also takes place in the presence of channel state information at the transmitter. CSIT-type feedback is crucial in handling interference, and can thus substantially reduce the resulting duration T of the delivery phase. This CSIT is typically of imperfect-quality as it is hard to obtain in a timely and reliable manner. In the high-SNR (high P) regime of interest, this current-CSIT quality is concisely represented in the form of the normalized quality exponent [20] [9]

$$\alpha := -\lim_{P\to\infty} \frac{\log \mathbb{E}[||\boldsymbol{h}_k - \hat{\boldsymbol{h}}_k||^2]}{\log P},\ k \in \{1,\dots,K\} \tag{3.1}$$

where $\boldsymbol{h}_k - \hat{\boldsymbol{h}}_k$ denotes the estimation error between the current CSIT estimate $\hat{\boldsymbol{h}}_k$ and the estimated channel $\boldsymbol{h}_k$. The range of interest[1] is $\alpha \in [0,1]$. We also assume availability of delayed CSIT (as in for example [1], as well as in a variety of subsequent works [9, 20–27] as well as [64–66]) where now the delayed estimates of any channel, can be received without error but with arbitrary delay, even if this delay renders this CSIT completely obsolete. As it is argued in [20], this mixed CSI model (partial current CSIT, and delayed CSIT) nicely captures different realistic settings that might involve channel correlations and an ability to improve CSI as time progresses. This same CSI model is particularly well suited for our caching-related setting here, because it explicitly reflects two key ingredients that are directly intertwined with coded caching; namely, feedback timeliness and feedback quality.

Intuitive links between α and γ

As we will see, α is not only linked to the performance — where a higher α allows for better interference management and higher performance over the wireless delivery link — but is also linked to caching; after all, the bigger the γ, the more side information the receivers have, the less interference one needs to handle (at least in symmetric systems), and the smaller the α that is potentially needed to steer interference. This means that principally, a higher γ implies that more common information needs to be transmitted, which may (in some cases) diminish the utility of feedback which primarily aims to facilitate the opposite which is the transmission of private information. It is for example easy to see (we will see this later) that in the presence of $\Gamma = K - 1$, there is no need for CSIT in order to achieve the optimal performance.

[1] In the high SNR regime of interest here, $\alpha = 0$ corresponds to having essentially no current CSIT (cf. [63]), while having $\alpha = 1$ corresponds (again in the high SNR regime) to perfect and immediately available CSIT.

3.1.3 Main assumptions

Throughout this work, we assume availability of current CSIT with some quality α, of delayed CSIT (D-CSIT), as well as ask that each receiver knows their own channel perfectly. We also adhere to the common convention (see for example [1]) of assuming perfect and global knowledge of delayed channel state information at the receivers (delayed global CSIR), where each receiver must know (with delay) the CSIR of (some of the) other receivers. We will assume that the entries of *each specific* estimation error vector are i.i.d. Gaussian. For the outer (lower) bound to hold, we will make the common assumption that the current channel state must be independent of the previous channel-estimates and estimation errors, *conditioned on the current estimate* (there is no need for the channel to be i.i.d. in time). We will make the assumption that the channel is drawn from a continuous ergodic distribution such that all the channel matrices and all their sub-matrices are full rank almost surely. We also make the soft assumption that the transmitter *during the delivery phase* is aware of the feedback statistics. We note though that, while our main scheme assumes knowledge of α during the caching phase, most results will be the outcome of a simpler scheme that does not require knowledge of α during this caching phase. Removing this assumption entails, for $\alpha > 0$, a performance penalty which is small.

3.1.4 Prior work

A pertinent work can be found in [45] where Maddah-Ali and Niesen studied the wireless interference channel where each transmitter has a local cache, and showed distinct benefits of coded caching that stem from the fact that content-overlap at the transmitters allows effective interference cancellation. The work in [15] by Shariatpanahi et al. explored caching and perfect current CSIT in the context of multi-server case. An even more general work can be found in [70], where Naderializadeh et al, studied the single-shot DoF with these two ingredients. The first work that considers the actual interplay between coded caching and CSIT quality is our work in [62] which considered the easier problem of how the optimal cache-aided performance (with coded caching), can be achieved with reduced quality CSIT.

3.1.5 Outline and contributions

In Section 4.2, Lemma 3.1, we offer a lower bound for the optimal $T^*(\gamma, \alpha)$. Then in Theorem 3.2 we calculate the achievable $T(\gamma, \alpha)$, for $\Gamma \in \{1, 2, \cdots, K\}$, $\alpha \in [0, 1]$, and prove it to be less than four times the optimal, thus identifying the optimal $T^*(\gamma, \alpha)$ within a factor of 4. A simpler expression for T (again within a factor of 4 from optimal), and its corresponding per-user DoF, are derived in Theorem 3.2, while a simple approximation of these is derived in Corollary 3.1, where we see that the per-user DoF takes the form

$d(\gamma,\alpha)=\alpha+(1-\alpha)\frac{1-\gamma}{\log\frac{1}{\gamma}}$, revealing that even a very small $\gamma=e^{-G}$ can offer a substantial DoF boost $d(\gamma=e^{-G},\alpha)-d(\gamma=0,\alpha)\approx(1-\alpha)\frac{1}{G}$. In Section 3.3 we discuss practical implications. In Corollary 3.2 we describe the savings in current CSIT that we can have due to coded caching, while in Corollary 3.4 we quantify the intuition that, in the presence of coded-caching, there is no reason to improve CSIT beyond a certain threshold quality.

In Section 3.5 we present the caching-and-delivery schemes, which build on the interesting connections between MAT-type retrospective transmission schemes (cf. [1]) and coded caching. The caching part is modified from [2] to essentially *'fold'* (linearly combine) the different users' data into multi-layered blocks, in a way such that the subsequent transmission algorithm (which employs parts of the QMAT algorithm in [67]) is suited to efficiently unfold these. The caching and transmission algorithms are calibrated so that the caching algorithm — which is modified from that in [2] to adapt the caching redundancy to α — creates the same multi-destination delivery problem that is efficiently solved by the last stages of the QMAT scheme. Appendix 3.7 presents the outer bound proof, and Appendix 3.8 the proof for the gap to optimal.

3.2 Throughput of cache-aided BC as a function of CSIT quality and caching resources

The following results hold for the (K,M,N,α) cache-aided K-user wireless MISO BC with random fading, $\alpha\in[0,1]$ and $N\geq K$, where $\gamma=\frac{M}{N}$ and $\Gamma=K\gamma$. We begin with an outer bound (lower bound) on the optimal T^*.

Lemma 3.1 *The optimal T^* for the (K,M,N,α) cache-aided K-user MISO BC, is lower bounded as*

$$T^*(\gamma,\alpha)\geq \max_{s\in\{1,\ldots,\lfloor\frac{N}{M}\rfloor\}}\frac{1}{(H_s\alpha+1-\alpha)}(H_s-\frac{Ms}{\lfloor\frac{N}{s}\rfloor}). \tag{3.2}$$

Proof. The proof is presented in Section 3.7, and it uses the bound from Lemma 3.2 whose proof can be found in Section 3.7.1. ■

3.2.1 Achievable throughput of the cache-aided BC

The following identifies, up to a factor of 4, the optimal T^*, for all $\Gamma\in\{1,2,\cdots,K\}$ (i.e., $M\in\frac{N}{K}\{1,\cdots,K\}$). The result uses the expression

$$\alpha_{b,\eta}=\frac{\eta-\Gamma}{\Gamma(H_K-H_\eta-1)+\eta},\ \eta=\lceil\Gamma\rceil,\ldots,K-1. \tag{3.3}$$

Note that the above does not hold for $\Gamma=K$, as this would imply no need for delivery.

Theorem 3.1 *In the (K, M, N, α) cache-aided MISO BC with N files, $K \leq N$ users, $\Gamma \in \{1, 2, \cdots, K\}$, and for $\eta = \arg\max_{\eta' \in [\Gamma, K-1] \cap \mathbb{Z}} \{\eta' \ : \ \alpha_{b,\eta'} \leq \alpha\}$, then*

$$T = \max\{1 - \gamma, \frac{(K - \Gamma)(H_K - H_\eta)}{(K - \eta) + \alpha(\eta + K(H_K - H_\eta - 1))}\} \tag{3.4}$$

is achievable and always has a gap-to-optimal that is less than 4, for all α, K. For $\alpha \geq \frac{K(1-\gamma)-1}{(K-1)(1-\gamma)}$, T is optimal.

Proof. The caching and delivery scheme that achieves the above performance is presented in Section 3.5, while the corresponding gap to optimal is bounded in Section 3.8. ■

The above is achieved with a general scheme whose caching phase is a function of α. We will henceforth consider a special case ($\eta = \Gamma$) of this scheme, which provides similar performance (it again has a gap to optimal that is bounded by 4), simpler expressions, and has the practical advantage that the caching phase need not depend on the CSIT statistics α of the delivery phase. For this case, we can achieve the following performance.

Theorem 3.2 *In the (K, M, N, α) cache-aided MISO BC with $\Gamma \in \{1, 2, \cdots, K\}$,*

$$T = \frac{(1 - \gamma)(H_K - H_\Gamma)}{\alpha(H_K - H_\Gamma) + (1 - \alpha)(1 - \gamma)} \tag{3.5}$$

is achievable and has a gap from optimal

$$\frac{T}{T^*} < 4 \tag{3.6}$$

that is less than 4, for all α, K. Thus the corresponding per-user DoF takes the form

$$d(\gamma, \alpha) = \alpha + (1 - \alpha)\frac{1 - \gamma}{H_K - H_\Gamma}. \tag{3.7}$$

Proof. The scheme that achieves the above performance will be described later on as a special (simpler) case of the scheme corresponding to Theorem 3.1. The corresponding gap to optimal is bounded in Section 3.8. ■

The following corollary describes the above achievable T, under the logarithmic approximation $H_n \approx \log(n)$. The presented expression is exact in the large K setting[2] where $\frac{H_K - H_\Gamma}{\log(\frac{1}{\gamma})} = 1$.

[2] For large K, this approximation $\frac{H_K - H_\Gamma}{\log(\frac{1}{\gamma})} = 1$ is tight for any *fixed* γ.

Corollary 3.1 *Under the logarithmic approximation $H_n \approx \log(n)$, the derived T takes the form*

$$T(\gamma,\alpha) = \frac{(1-\gamma)\log(\frac{1}{\gamma})}{\alpha\log(\frac{1}{\gamma}) + (1-\alpha)(1-\gamma)} \tag{3.8}$$

and the derived DoF takes the form

$$d(\gamma,\alpha) = \alpha + (1-\alpha)\frac{1-\gamma}{\log\frac{1}{\gamma}}. \tag{3.9}$$

For the large K setting, what the above suggests is that current CSIT offers an initial DoF boost of $d^*(\gamma=0,\alpha)=\alpha$ (cf. [67]), which is then supplemented by a DoF gain

$$d(\gamma,\alpha) - d^*(\gamma=0,\alpha) \to (1-\alpha)\frac{1-\gamma}{\log(\frac{1}{\gamma})}$$

attributed to the synergy between delayed CSIT and caching[3]. These synergistic gains (see also [71]) are accentuated for smaller values of γ, where we see an exponential effect of coded caching, in the sense that now a microscopic $\gamma = e^{-G}$ can offer a substantial DoF boost

$$d(\gamma=e^{-G},\alpha) - d(\gamma=0,\alpha) \approx (1-\alpha)\frac{1}{G}. \tag{3.10}$$

Interplay between CSIT quality and coded caching in the symmetric MISO BC

The derived form in (3.7) (and its approximation in (3.9)) nicely capture the synergistic as well as competing nature of feedback and coded caching. It is easy to see for example that the effect from coded-caching, reduces with α and is proportional to $1-\alpha$. This reflects the fact that in the symmetric MISO BC, feedback supports broadcasting by separating data streams, thus diminishing multi-casting by reducing the number of common streams. In the extreme case when $\alpha = 1$, we see — again for the symmetric MISO BC — that the caching gains are limited to local caching gains[4].

3.3 Cache-aided CSIT reductions

We proceed to explore how coded caching can alleviate the need for CSIT.

[3] We note that these interference-removal gains, particularly in the large K regime, are not a result of extra performance boost directly from D-CSIT, because in the large K setting, this latter performance boost is negligible (vanishes to zero) without caching.

[4] This conclusion is general (and not dependent on the specific schemes), because the used schemes are optimal for $\alpha = 1$. The statement holds because we can simply uniformly cache a fraction γ of each file in each cache, and upon request, use perfect-CSIT to zero-force the remaining requested information, to achieve the optimal $T^*(\gamma,\alpha=1) = 1-\gamma$, which leaves us with local (data push) caching gains only.

3.3.1 Cache-aided CSIT gains

To capture the cache-aided reductions on the CSIT load, let us consider

$$\bar{\alpha}(\gamma,\alpha) := \arg\min_{\alpha'}\{\alpha' : (1-\gamma)T^*(\gamma=0,\alpha') \leq T(\gamma,\alpha)\}$$

which is derived below in the form

$$\bar{\alpha}(\gamma,\alpha) = \alpha + \delta_\alpha(\gamma,\alpha)$$

for some $\delta_\alpha(\gamma,\alpha)$ that can be seen as the *CSIT reduction due to caching* (from $\bar{\alpha}(\gamma,\alpha)$ to the operational α).

Corollary 3.2 *In the (K,M,N,α) cache-aided MISO BC, then*

$$\bar{\alpha}(\gamma,\alpha) = \alpha + \frac{(1-\alpha)(H_{K\gamma}-\gamma H_K)}{(H_K-1)(H_K-H_{K\gamma})} \tag{3.11}$$

is achievable, and implies a cache-aided CSIT reduction

$$\delta_\alpha(\gamma,\alpha) = \frac{(1-\alpha)(H_{K\gamma}-\gamma H_K)}{(H_K-1)(H_K-H_{K\gamma})}.$$

Proof. The proof is direct from Theorem 3.2. ■

The above is made more insightful in the large K regime, for which we have the following.

Corollary 3.3 *In the (K,M,N,α) cache-aided MISO BC, then*

$$\bar{\alpha}(\gamma,\alpha) = \alpha + (1-\alpha)\frac{1-\gamma}{\log(\frac{1}{\gamma})} \tag{3.12}$$

which implies CSIT reductions of

$$\delta_\alpha(\gamma,\alpha) = (1-\alpha)d(\gamma,\alpha=0) = (1-\alpha)\frac{1-\gamma}{\log(\frac{1}{\gamma})}.$$

Proof. The proof is direct from the definition of $\bar{\alpha}(\gamma,\alpha)$ and from Theorem 3.2. ■

Furthermore we have the following which quantifies the intuition that, in the presence of coded-caching, there is no reason to improve CSIT beyond a certain threshold quality. The following uses the definition in (3.3), and it holds for all K.

Corollary 3.4 *For any* $\Gamma \in \{1, \ldots, K\}$*, then*

$$T^*(\gamma, \alpha) = T^*(\gamma, \alpha = 1) = 1 - \gamma \tag{3.13}$$

holds for any

$$\alpha \geq \alpha_{b,K-1} = \frac{K(1-\gamma) - 1}{(K-1)(1-\gamma)} \tag{3.14}$$

which reveals that CSIT quality $\alpha = \alpha_{b,K-1}$ *is the maximum needed, as it already offers the same optimal performance* $T^*(\gamma, \alpha = 1)$ *that would be achieved if CSIT was perfect.*

Proof. This is seen directly from Theorem 3.1 after noting that the achievable T matches $T^*(\gamma, \alpha = 1) = 1 - \gamma$. ■

How much caching is needed to partially substitute current CSIT with delayed CSIT (using coded caching to 'buffer' CSI)

As we have seen, in addition to offering substantial DoF gains, the synergy between feedback and caching can also be applied to reduce the burden of acquiring current CSIT. What the above results suggest is that a modest γ can allow a BC system with D-CSIT to approach the performance attributed to current CSIT, thus allowing us to partially substitute current with delayed CSIT, which can be interpreted as an ability to buffer CSI. A simple calculation — for the large-K regime — can tell us that

$$\gamma_\alpha^{'} := \arg\min_{\gamma^{'}} \{\gamma^{'} : d(\gamma^{'}, \alpha = 0) \geq d^*(\gamma = 0, \alpha)\} = e^{-1/\alpha}$$

which means that $\gamma_\alpha^{'} = e^{-1/\alpha}$ suffices to achieve — in conjunction with delayed CSIT — the optimal DoF performance $d^*(\gamma = 0, \alpha)$ associated to a system with delayed CSIT and α-quality current CSIT.

Example 3.1 *Let* K *be very large, and consider a BC system with delayed CSIT and* α*-quality current CSIT, where* $\alpha = 1/5$*. Then* $\gamma_{\alpha=1/5}^{'} = e^{-5} = 0.0067 \approx 1/150$ *which means that*

$$d^*(\gamma = 0.0067, \alpha = 0) \geq d^*(\gamma = 0, \alpha = 1/5)$$

which says that the same high-K *per-user DoF performance* $d^*(\gamma = 0, \alpha = 1/5)$*, can be achieved by substituting all current CSIT with coded caching employing* $\gamma \approx 1/150$*.*

3.4 Examples of schemes

The scheme that achieves the above results, will be presented in its general form in Section 3.5. To offer some intuition on the schemes, we provide here different examples (all for the case of $K = N = 3, M = 1$), first for the case of $\alpha = 1$, then for the general case of $\alpha \in (0, 1)$ corresponding to Theorem 3.2, and then a third example again for the general case of $\alpha \in (0, 1)$, now for the case corresponding to Theorem 3.1, where the caching redundancy increases with α.

In our examples here, for simplicity, the three distinct files in the library will be relabeled as $W_1 = A, W_2 = B, W_3 = C$. Finally we assume the worst-case request where A, B, C are requested by user 1, 2, 3, respectively.

3.4.1 Scheme for $\alpha = 1$

We offer this example as a warm up exercise, as the scheme is very simple. For caching, each user stores a fraction $\gamma = \frac{M}{N} = 1/3$ of each file, and then (upon notification of the three requests) the remaining $f(1-\gamma) = \frac{2f}{3}$ bits of each desired file are delivered using interference-free zero forcing which employs perfect CSIT. Hence after an optimal duration $T = (1-\gamma) = \frac{2}{3}$, the transmitter delivers file A to user 1, B to user 2 and C to user 3. In this symmetric setting, the complete separation of signals due to perfect-CSIT, renders multicasting unnecessary, and the optimal performance reflects only local caching gains.

3.4.2 Scheme for $\alpha \in (0, 1)$

We proceed with the general case of $\alpha \in (0, 1)$, again focusing on the case corresponding to Theorem 3.2 where η is forced to be $\eta = \Gamma$, for all α (in this case, $\eta = 1$).

Placement phase

The cache placement is the same as in the case of $\alpha = 0$ described in Section 2.4.1 in Chapter 2.

Data folding

Recall that users 1,2,3 respectively request files A, B, C, which will be delivered with delay T. Also recall than now, user 1 requires files A_2, A_3, user 2 files B_1, B_3, and user 3 files C_1, C_2. Each of these subfiles will be split into two mini parts, where for example $A_2 = (A_2^f, A_2^{\bar{f}}), A_3 = (A_3^f, A_3^{\bar{f}})$ and similarly for B_1, B_3, C_1, C_2, such that $|A_2^{\bar{f}}| = |A_3^{\bar{f}}| = |B_1^{\bar{f}}| = |B_3^{\bar{f}}| = |C_1^{\bar{f}}| = |C_2^{\bar{f}}| = \frac{f\alpha T}{2}$ bits. Now the three generated XORs will be $AB \triangleq A_2^f \oplus B_1^f$, $AC \triangleq A_3^f \oplus C_1^f$, and $BC \triangleq B_3^f \oplus C_2^f$, and which will delivered the 'folded' part each missing subfile,

while the remaining 'unfolded' parts $A_2^{\bar{f}}, A_3^{\bar{f}}, B_1^{\bar{f}}, B_3^{\bar{f}}, C_1^{\bar{f}}, C_2^{\bar{f}}$ will be delivered via a ZF component inside the employed QMAT scheme, as we describe below.

Transmission

We proceed to describe the transmission of the aforementioned folded and unfolded messages, employing the last two phases of the three-user QMAT algorithm from [67].

Phase 2: This phase will take as input the folded messages (which will be decoded upon completion of the third phase later on), and will also deliver some of the unfolded messages. For any $K\gamma$-length set (in this case, for any pair) $\psi \in \{\{1,2\},\{1,3\},\{2,3\}\}$, and for $\bar{\psi} \triangleq \{1,2,3\}\backslash\psi$, the transmission takes the form

$$\boldsymbol{x}_t = \mathbf{G}_{\psi,t}\boldsymbol{x}_{\psi,t} + \boldsymbol{g}_{\bar{\psi},t}a^*_{\bar{\psi},t} + \sum_{k=1}^{3}\boldsymbol{g}_{k,t}a_{k,t}. \tag{3.15}$$

where

- $\mathbf{G}_{\psi,t} \triangleq [\boldsymbol{g}_{\bar{\psi},t}, \mathbf{U}_{\psi,t}]$, for $\boldsymbol{g}_{\bar{\psi},t}$ being simultaneously orthogonal to the channel estimates of the two users in ψ, and for $\mathbf{U}_\psi \in \mathbb{C}^{3\times 2}$ being a randomly chosen sub-unitary matrix
- $\boldsymbol{g}_{k,t}$ is orthogonal to the current estimates of the channels to users $\{1,2,3\}\backslash k$
- $\boldsymbol{x}_{\psi,t} = [x_{\psi,1}, x_{\psi,2}, 0]^T$ is a $K = 3$-length vector with $K - K\gamma = 2$ non-zero scalar entries, where each scalar has rate $(1-\alpha)\log P$ bits per unit time, and where each scalar has power $\mathbb{E}[|x_{\psi,1}|^2] \doteq P$, $\mathbb{E}[|x_{\psi,2}|^2] \doteq P^{1-\alpha}$
- the bits in XOR AB ($\psi = \{1,2\}$) are split evenly between $x_{\{1,2\},1}$ and $x_{\{1,2\},2}$, the bits in XOR AC ($\psi = \{1,3\}$) are split evenly between $x_{\{1,3\},1}$ and $x_{\{1,3\},2}$, and the bits in XOR BC ($\psi = \{2,3\}$) are split evenly between $x_{\{2,3\},1}$ and $x_{\{2,3\},2}$
- irrespective of ψ, $a_{k,t}$ is the ZF symbol carrying the unfolded messages for user $k \in \{1,2,3\}$, with the rate $\alpha\log P$ bits per unit time, and with power P^α
- $a^*_{\bar{\psi},t}$ is an auxiliary symbol intended for user $\bar{\psi}$, carrying residual interference[5]. The symbol has power P, and rate $\min(1-\alpha,\alpha)\log P$ bits per unit time. Auxiliary symbols allow for the simultaneous delivery of private data (unfolded messages) and higher-order data (XORs)

[5]The interference is carried over from a previous round of the QMAT scheme. We spare the reader some of the details regarding rounds, and consider the scheme for just one round. The rounds are linked via the auxiliary variables, and having more than one round simply guarantees the QMAT DoF optimality, as it minimizes the cost of initialization.

- Transmission $\boldsymbol{x}_t$ is sequential: first for $\psi = \{1,2\}$, then for $\psi = \{1,3\}$, and then for $\psi = \{2,3\}$.

For any given ψ, the received signal (noise removed) $y_{k,t}$ of each desired user $k \in \psi$, takes the form

$$y_{k,t} = \underbrace{\boldsymbol{h}_{k,t}^T \mathbf{G}_{\psi,t} \boldsymbol{x}_{\psi,t}}_{\triangleq L_{\psi,k},\ \text{power} \doteq P} + \underbrace{\boldsymbol{h}_{k,t}^T \boldsymbol{g}_{\bar{\psi},t} a_{\bar{\psi},t}^*}_{\doteq P^{1-\alpha}} + \underbrace{\boldsymbol{h}_{k,t}^T \boldsymbol{g}_{k,t} a_{k,t}}_{\doteq P^{\alpha}} \tag{3.16}$$

while the received signal for the other user $\bar{\psi}$, takes the form

$$y_{\bar{\psi},t} = \underbrace{\boldsymbol{h}_{\bar{\psi},t}^T \boldsymbol{g}_{\bar{\psi},t} a_{\bar{\psi},t}^*}_{\text{power} \doteq P} + \underbrace{\boldsymbol{h}_{\bar{\psi},t}^T \mathbf{G}_{\psi,t} \boldsymbol{x}_{\psi,t}}_{\triangleq i_{\psi,\bar{\psi}},\ P^{1-\alpha}} + \underbrace{\boldsymbol{h}_{\bar{\psi},t}^T \boldsymbol{g}_{\bar{\psi},t} a_{\bar{\psi},t}}_{P^{\alpha}}. \tag{3.17}$$

It is easy to see that at the end of this phase, each user in ψ needs one more observation to resolve $x_{\psi,1}, x_{\psi,2}$, hence the overheard messages $i_{\psi,\bar{\psi}} \triangleq \boldsymbol{h}_{\bar{\psi},t}^T \mathbf{G}_{\psi,t} \boldsymbol{x}_{\psi,t}$ will be quantized and placed into a message that will be meant for $K\gamma + 1 = 3$ users, and which will be delivered in the next phase.

To calculate the duration of this second phase, we recall that the phase terminates when we transmit $AB = A_2^f \oplus B_1^f$, $AC = A_3^f \oplus C_1^f$, and $BC = B_3^f \oplus C_2^f$. Given that $|A_2^{\bar{f}}| = |A_3^{\bar{f}}| = |B_1^{\bar{f}}| = |B_3^{\bar{f}}| = |C_1^{\bar{f}}| = |C_2^{\bar{f}}| = \frac{f\alpha T}{2}$ bits (by design, as we have seen above), means that $|AB| = |AC| = |BC| = \frac{1}{3} - \frac{f\alpha T}{2}$. Given that the rate of each transmitted scalar $x_{\psi,1}$ and $x_{\psi,2}$ is $(1-\alpha)\log P$ bits per unit time, and given that we are transmitting all three XORS AB, AC, BC, means that the duration of the second phase is $3\frac{\frac{1}{3} - \frac{f\alpha T}{2}}{2(1-\alpha)}$.

Phase 3: We use $\bar{i}_{\psi,\bar{\psi}}$ to denote the quantized version of $i_{\psi,\bar{\psi}}$ which can be reconstructed by the transmitter in the next phase. Instead of creating two linear combinations as in the case of only having delayed CSIT, here we use the XOR operator to combine messages: we create $f_1 \triangleq \bar{i}_{12,3} \oplus \bar{i}_{13,2}$ and $f_2 \triangleq \bar{i}_{13,2} \oplus \bar{i}_{23,1}$. The transmission then takes the form

$$\boldsymbol{x}_t = [x_{c,t}, 0, 0]^T + \sum_{k=1}^{3} \boldsymbol{g}_{k,t} a_{k,t} \tag{3.18}$$

where $x_{c,t}$ carries information from f_1 and f_2, has power P, rate $(1-\alpha)\log P$ bits per unit time, and finally where $a_{k,t}$ is again the ZF symbols with power P^{α}, and rate $\alpha \log P$ bits per unit time, as before.

Decoding

Now each user can decode $x_{c,t}$ and its own private (ZF) messages in phase 3, and can then go back to phase 2 to decode the XORs and the private messages. To see this, we again focus on user 1 who already knows $L_{\{1,2\},1}, L_{\{1,3\},1}, \bar{i}_{\{2,3\},1}$

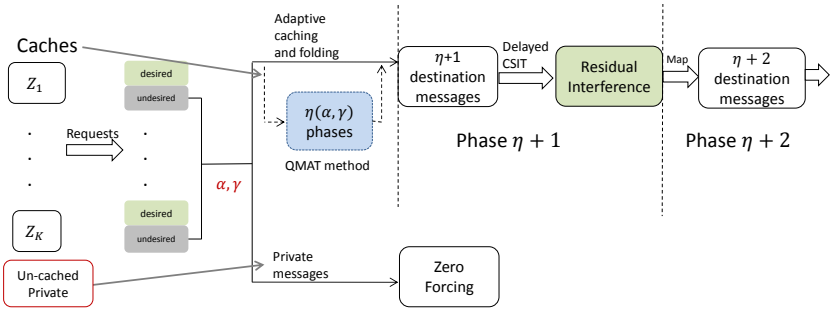

Figure 3.1: Cache-aided retrospective communications scheme.

(cf. (3.16)), where $L_{\psi,k} \triangleq \boldsymbol{h}_{k,t}^T \mathbf{G}_{\psi,t} \boldsymbol{x}_{\psi,t}$ (here, since we focus on user 1, we set $k = 1$).

From the common message $x_{c,t}$, user 1 knows $f_1 = \bar{i}_{12,3} \oplus \bar{i}_{13,2}$ and $f_2 = \bar{i}_{13,2} \oplus \bar{i}_{23,1}$. Using its knowledge of $\bar{i}_{23,1}$ with f_2, user 1 can get $\bar{i}_{13,2}$, and then from f_1 she can also have $\bar{i}_{12,3}$. Now user 1 combines $\bar{i}_{12,3}$ with $L_{\{1,2\},1}$ to get $x_{\{1,2\},1}$ and $x_{\{1,2\},2}$ which allows us to resolve $AB = A_2^f \oplus B_1^f$. Then user 1 combines $\bar{i}_{13,2}$ with $L_{\{1,3\},1}$ to get $x_{\{1,3\},1}$ and $x_{\{1,3\},2}$ which allows us to resolve $AC = A_3^f \oplus C_1^f$. Using $B_1^f \in Z_1$ yields A_2^f, and using $C_1^f \in Z_1$ yields A_3^f. Combined with the ZF-transmitted private data which delivers $A_2^{\bar{f}}$ and $A_3^{\bar{f}}$, completes the delivery of A_2, A_3 and thus of A.

To calculate the duration of the third phase, we recall that $x_{c,t}$ carries f_1 and f_2, each of size $|f_1| = |f_2| = \frac{\frac{1}{3} - \frac{f\alpha T}{2}}{2}$. Thus $x_{c,t}$ carries a total of $\frac{1}{3} - \frac{f\alpha T}{2}$ bits, at a rate of $(1-\alpha)\log P$ bits per unit time. Hence the total duration of phase 3 is $\frac{\frac{1}{3} - \frac{f\alpha T}{2}}{1-\alpha}$ Combined with the duration $3\frac{\frac{1}{3} - \frac{f\alpha T}{2}}{2(1-\alpha)}$ of phase 2, implies a total duration of

$$T = \frac{10}{12 + 3\alpha} \tag{3.19}$$

which matches the derived expression

$$T = \frac{(1-\gamma)(H_K - H_\Gamma)}{\alpha(H_K - H_\Gamma) + (1-\alpha)(1-\gamma)} = \frac{\frac{2}{3}(H_3 - H_1)}{\alpha(H_3 - H_1) + (1-\alpha)\frac{2}{3}} = \frac{10}{12 + 3\alpha}.$$

3.4.3 Scheme for adapting caching redundancy η as a function of α

In the previous example, we considered the case corresponding to Theorem 3.2 where the caching redundancy η is fixed as $\eta = K\gamma$. This incurs a certain (albeit small) degree of sub-optimality, because as we explain in the next section, a higher α can allow for higher caching redundancy because more private messages means reduced multicasting, thus allowing for some of the data to remain

uncached, which in turn allows for more copies of the same information across different users' caches (see Fig 3.1). Here we give an example of the general scheme that captures this interplay between η and α, and which corresponds to Theorem 3.1, which improves slightly upon Theorem 3.2. Our focus in this example is to show how we calibrate — as a function of α — the cache placement and the process of creating the XORs. This is again presented for the case of $K = N = 3$, $M = 1$.

Using the defined breaking points $\alpha_{b,\eta} = \frac{\eta-\Gamma}{\Gamma(H_K-H_\eta-1)+\eta}$, $\eta = \lceil\Gamma\rceil, \ldots, K-1$ in (3.3), the range of α is split into two intervals. The first interval is $\alpha \in [0, \alpha_{b,2}) = [0, \frac{3}{4}]$ during which the scheme remains the same as the scheme we presented in the previous example, where by choosing $\eta = K\gamma = 1$, we create XORs that are meant for two users at a time. In the second interval $\alpha \geq \alpha_{b,2} = \frac{3}{4}$, we set $\eta = 2$, and each XOR is meant for $\eta + 1 = 3$ users, allowing to skip the second phase of the previous example. Below we show how caching and folding adapts to this case.

Placement phase: We first split each file as

$$A = (A^c, A^{\bar{c}}), B = (B^c, B^{\bar{c}}), C = (C^c, C^{\bar{c}}), \tag{3.20}$$

where A^c, B^c, C^c denote the cached parts, and $A^{\bar{c}}, B^{\bar{c}}, C^{\bar{c}}$ the parts that are not cached. The split is such that $|A^c| = |B^c| = |C^c| = f\frac{K\gamma}{\eta} = \frac{f}{2}$. Then each cached part is again divided evenly into $\binom{K}{\eta} = 3$ mini parts as $A^c = (A_{12}, A_{13}, A_{23})$, $B^c = (B_{12}, B_{13}, B_{23}), C^c = (C_{12}, C_{13}, C_{23})$, and then the caches are filled as

$$\begin{aligned} Z_1 &= A_{12}, A_{13}, B_{12}, B_{13}, C_{12}, C_{13}, \\ Z_2 &= A_{12}, A_{23}, B_{12}, B_{23}, C_{12}, C_{23}, \\ Z_3 &= A_{13}, A_{23}, B_{13}, B_{23}, C_{13}, C_{23}. \end{aligned} \tag{3.21}$$

Now, to satisfy the requests A, B, C for users $1, 2, 3$ respectively, we must send $A_{23} \oplus B_{13} \oplus C_{12}$ as well as the uncached messages $A^{\bar{c}}, B^{\bar{c}}, C^{\bar{c}}$. This will be achieved by employing phase $\eta + 1 = 3$ of the scheme we saw in the previous example, where the transmission again takes the form $\boldsymbol{x}_t = [x_{c,t}, 0, 0]^T + \sum_{k=1}^{3} \boldsymbol{g}_{k,t} a_{k,t}$, where $x_{c,t}$ carries $A_{23} \oplus B_{13} \oplus C_{12}$ (again with power P and rate $(1-\alpha)\log P$ bits per unit time), while $a_{1,t}, a_{2,t}, a_{3,t}$ respectively carry $A^{\bar{c}}, B^{\bar{c}}, C^{\bar{c}}$ (each with power P^α and rate $\alpha \log P$ bits per unit time).

This adaptation of η as a function of α provides for a slightly better performance, which now — for any $\alpha \geq \frac{3}{4}$ — takes the form $T = 1 - \gamma = \frac{2}{3}$, which is the interference-free optimal, despite having imperfect CSIT, as this was discussed in Corollary (3.4). The above performance is an improvement over the previously derived $T = \frac{10}{12+3\alpha}$ for all $\alpha \geq \frac{K(1-\gamma)-1}{(K-1)(1-\gamma)} = \frac{3}{4}$.

3.5 Cache-aided retrospective communications

We proceed to describe the communication scheme, and in particular the process of placement, folding-and-delivery, and decoding. In the end we calculate the achievable duration T.

The caching part is modified from [2] to *'fold'* (linearly combine) the different users' data into multi-layered blocks, in a way such that the subsequent Q-MAT transmission algorithm (cf. [67]) (specifically the last $K - \eta_\alpha$ ($\eta_\alpha \in \{\Gamma, \dots, K-1\}$) phases of the QMAT algorithm) can efficiently deliver these blocks. Equivalently the algorithms are calibrated so that the caching algorithm creates a multi-destination delivery problem that is the same as that which is efficiently solved by the last stages of the QMAT-type communication scheme. We henceforth remove the subscript in η_α and simply use η, where now the dependence on α is implied.

3.5.1 Placement phase

We proceed with the placement phase which modifies on the work of [2] such that when the CSIT quality α increases, the algorithm caches a decreasing portion from each file, but does so with increasing redundancy. The idea is that the higher the α, the more private messages one can deliver directly without the need to multicast, thus allowing for some of the data to remain entirely uncached, which in turn allows for more copies of the same information across different users' caches.

Here each of the N files $W_n, n = 1, 2, \dots, N$ ($|W_n| = f$ bits) in the library, is split into two parts

$$W_n = (W_n^c, W_n^{\bar{c}}) \tag{3.22}$$

where W_n^c (c for 'cached') will be placed into one or more caches, while the content of $W_n^{\bar{c}}$ ($\bar{c}$ for 'non-cached') will never be cached anywhere, but will instead be communicated — using CSIT — in a manner that causes manageable interference and hence does not necessarily benefit from coded caching. The split is such that

$$|W_n^c| = \frac{KMf}{N\eta} \tag{3.23}$$

where $\eta \in \{\Gamma, \dots, K-1\}$ is a positive integer, the value of which will be decided later on such that it properly regulates how much to cache from each W_n. Now for any specific η, we equally divide W_n^c into $\binom{K}{\eta}$ subfiles $\{W_{n,\tau}^c\}_{\tau \in \Psi_\eta}$,

$$W_n^c = \{W_{n,\tau}^c\}_{\tau \in \Psi_\eta} \tag{3.24}$$

where[6] $\Psi_\eta := \{\tau \subset [K] \ : \ |\tau| = \eta\}$ and where each subfile has size

$$|W^c_{n,\tau}| = \frac{KMf}{N\eta\binom{K}{\eta}} = \frac{Mf}{N\binom{K-1}{\eta-1}} \text{ bits.} \tag{3.25}$$

Now drawing from [2], the caches are filled as follows

$$Z_k = \{W^c_{n,\tau}\}_{n\in[N],\tau\in\Psi^{(k)}_\eta} \tag{3.26}$$

where $\Psi^{(k)}_\eta := \{\tau \in \Psi_\eta \ : \ k \in \tau\}$. Hence each subfile $W^c_{n,\tau}$ is stored in Z_k as long as $k \in \tau$, which means that each $W^c_{n,\tau}$ (and thus each part of W^c_n) is repeated η times in the caches. As η increases with α, this means that CSIT allows for a higher redundancy in the caches; instead of content appearing in Γ different caches, it appears in $\eta \geq \Gamma$ caches instead, which will translate into multicast messages that are intended for more receivers.

3.5.2 Data folding

At this point, the transmitter becomes aware of the file requests $R_k, k = 1,\dots,K$, and must now deliver each requested file W_{R_k}, by delivering the constituent subfiles $\{W^c_{R_k,\tau}\}_{\tau\in\Psi_\eta\backslash\Psi^{(k)}_\eta}$ as well as $W^{\overline{c}}_{R_k}$, all to the corresponding receiver k. We quickly recall that:

1. subfiles $\{W^c_{R_k,\tau}\}_{\tau\in\Psi^{(k)}_\eta}$ are already in Z_k;

2. subfiles $\{W^c_{R_k,\tau}\}_{\tau\in\Psi_\eta\backslash\Psi^{(k)}_\eta}$ are directly requested by user k, but are not cached in Z_k;

3. subfiles $Z_k\backslash\{W^c_{R_k,\tau}\}_{\tau\in\Psi^{(k)}_\eta} = Z_k\backslash W^c_{R_k}$ are cached in Z_k, are not directly requested by user k, but will be useful in removing interference.

We assume the communication here has duration T. Thus for each k and a chosen η, we split each subfile $W^c_{R_k,\tau}$, $\tau \in \Psi_\eta\backslash\Psi^{(k)}_\eta$ (each of size $|W^c_{R_k,\tau}| = \frac{Mf}{N\binom{K-1}{\eta-1}}$ as we saw in (4.9)) into

$$W^c_{R_k,\tau} = [W^{c,f}_{R_k,\tau} \ \ W^{c,\overline{f}}_{R_k,\tau}] \tag{3.27}$$

where $W^{c,f}_{R_k,\tau}$ corresponds to information that appears in a cache somewhere and that will be eventually 'folded' (XORed) with other information, whereas

[6] We recall that in the above, τ and $W^c_{n,\tau}$ are sets, thus $|\tau|, |W^c_{n,\tau}|$ denote cardinalities; $|\tau| = \eta$ means that τ has η different elements from $[K]$, while $|W^c_{n,\tau}|$ describes the size of $W^c_{n,\tau}$ in bits.

$W^{c,\overline{f}}_{R_k,\tau}$ corresponds to information that is cached somewhere but that will not be folded with other information. The split yields

$$|W^{c,\overline{f}}_{R_k,\tau}| = \frac{f\alpha T - f(1-\frac{KM}{N\eta})}{\binom{K-1}{\eta}} \text{ (bits)} \tag{3.28}$$

where in the above, $f\alpha T$ represents the load for each user without causing interference during the delivery phase, where $f(1-\frac{KM}{N\eta})$ is the amount of uncached information, and where $|W^{c,f}_{R_k,\tau}| = |W^{c}_{R_k,\tau}| - |W^{c,\overline{f}}_{R_k,\tau}|$.

We proceed to fold cached content, by creating linear combinations (XORs) from $\{W^{c,f}_{R_k,\tau}\}_{\tau\in\Psi_\eta\setminus\Psi^{(k)}_\eta}, \forall k$. We will use $P_{k,k'}(\tau)$ to be the function that replaces inside τ, the entry $k'\in\tau$, with the entry k. As in [2], the idea is that if we deliver

$$W^{c,f}_{R_k,\tau} \oplus (\oplus_{k'\in\tau} \underbrace{W^{c,f}_{R_{k'},P_{k,k'}(\tau)}}_{\in Z_k}) \tag{3.29}$$

the fact that $W^{c,f}_{R_{k'},P_{k,k'}(\tau)} \in Z_k$, guarantees that receiver k can recover $W^{c,f}_{R_k,\tau}$, while at the same time guarantees that each other user $k'\in\tau$ can recover its own desired subfile $W^{c,f}_{R'_k,P_{k,k'}(\tau)} \notin Z_{k'}, \forall k'\in\tau$.

Hence delivery of each $W^{c,f}_{R_k,\tau} \oplus (\oplus_{k'\in\tau} W^{c,f}_{R_{k'},P_{k,k'}(\tau)})$ of size

$$|W^{c,f}_{R_k,\tau} \oplus (\oplus_{k'\in\tau} W^{c,f}_{R_{k'},P_{k,k'}(\tau)})| = |W^{c,f}_{R_k,\tau}|$$

(cf. (4.9)), automatically guarantees delivery of $W^{c,f}_{R_{k'},P_{k,k'}(\tau)}$ to each user $k'\in\tau$, i.e., simultaneously delivers a total of $\eta+1$ distinct subfiles (each again of size $|W^{c,f}_{R_{k'},P_{k,k'}(\tau)}| = |W^{c,f}_{R_k,\tau}|$ bits) to $\eta+1$ distinct users. Hence *any*

$$X_\psi := \oplus_{k\in\psi} W^{c,f}_{R_k,\psi\setminus\{k\}}, \psi\in\Psi_{\eta+1} \tag{3.30}$$

— which is of the same form as in (3.29), and which is referred to here as an *order-$(\eta+1)$ folded message* — can similarly deliver to user $k\in\psi$, her requested file $W^{c,f}_{R_k,\psi\setminus k}$, which in turn means that each order-$(\eta+1)$ folded message X_ψ can deliver — with the assistance of the side information in the caches — a distinct, individually requested subfile, to each of the $\eta+1$ users $k\in\psi$ ($\psi\in\Psi_{\eta+1}$).

Thus to satisfy all requests $\{W_{R_k}\setminus Z_k\}_{k=1}^K$, the transmitter must deliver

- uncached messages $W^{\overline{c}}_{R_k},\ k=1,\ldots,K$
- cached but unfolded messages $\{W^{c,\overline{f}}_{R_k,\psi\setminus\{k\}}\}_{\psi\in\Psi_{\eta+1}},\ k=1,\ldots,K$

- and the entire set

$$\mathcal{X}_\Psi := \{X_\psi = \oplus_{k\in\psi} W^{c,f}_{R_k,\psi\backslash\{k\}}\}_{\psi\in\Psi_{\eta+1}} \tag{3.31}$$

consisting of $|\mathcal{X}_\Psi| = \binom{K}{\eta+1}$ folded messages of order-$(\eta+1)$, each of size (cf. (3.28),(4.9))

$$\begin{aligned} |X_\psi| &= |W^{c,f}_{R_k,\tau}| = |W^{c}_{R_k,\tau}| - |W^{c,\overline{f}}_{R_k,\tau}| \\ &= \frac{f(1-\gamma-\alpha T)}{\binom{K-1}{\eta}} \text{ (bits)}. \end{aligned} \tag{3.32}$$

3.5.3 Transmission

We proceed to describe the transmission of the aforementioned messages by adapting the QMAT algorithm from [67], with delay T.

The QMAT algorithm has K transmission phases. For each phase $i = 1, \cdots, K$, the QMAT data symbols are intended for a subset $\mathcal{S} \subset [K]$ of users, where $|\mathcal{S}| = i$. Here by adapting the algorithm, at each instance $t \in [0, T]$ through the transmission, the transmitted vector takes the form

$$\boldsymbol{x}_t = \mathbf{G}_{c,t}\boldsymbol{x}_{c,t} + \sum_{\ell\in\bar{\mathcal{S}}} \boldsymbol{g}_{\ell,t} a^*_{\ell,t} + \sum_{k=1}^{K} \boldsymbol{g}_{k,t} a_{k,t} \tag{3.33}$$

with $\boldsymbol{x}_{c,t}$ being a K-length vector for QMAT data symbols, with $a^*_{\ell,t}$ being an auxiliary symbol that carries residual interference, where $\bar{\mathcal{S}}$ is a set of 'undesired' users that changes every phase, and where each unit-norm precoder $\boldsymbol{g}_{k,t}$ for user $k = 1, 2, \ldots, K$, is simultaneously orthogonal to the CSI estimate for the channels of all other users ($\boldsymbol{g}_{l,t}$ acts the same), thus guaranteeing

$$\hat{\boldsymbol{h}}^T_{k',t}\boldsymbol{g}_{k,t} = 0, \quad \forall k' \in [K]\backslash k. \tag{3.34}$$

Each precoder $\mathbf{G}_{c,t}$ is defined as $\mathbf{G}_{c,t} = [\boldsymbol{g}_{c,t}, \mathbf{U}_{c,t}]$, where $\boldsymbol{g}_{c,t}$ is simultaneously orthogonal to the channel estimates of the undesired receivers, and $\mathbf{U}_{c,t} \in \mathbb{C}^{K\times(K-1)}$ is a randomly chosen, isotropically distributed unitary matrix [7].

We will allocate the rates such that

[7] Whenever possible, we will henceforth avoid going into the details of the QMAT scheme. Some aspects of this scheme are similar to MAT, and a main new element is that QMAT applies digital transmission of interference, and a double-quantization method that collects and distributes residual interference across different rounds, in a manner that allows for ZF and MAT to coexist at maximal rates. The important element for the decoding part later on, will be how to load the symbols, the rate of each symbol, and the corresponding allocated power. An element that is hidden from the presentation here is that, while the QMAT scheme has many rounds, and while decoding spans more than one round, we will — in a slight abuse of notation — focus on describing just one round, which we believe is sufficient for the purposes of this paper here.

- each $\boldsymbol{x}_{c,t}$ carries $f(1-\alpha)\text{dur}(\boldsymbol{x}_{c,t})$ bits,
- each $a^*_{\ell,t}$ carries $\min\{f(1-\alpha), f\alpha\}\text{dur}(\boldsymbol{g}_{\ell,t}a^*_{\ell,t})$ bits,
- each $a_{k,t}$ carries $f\alpha\text{dur}(\boldsymbol{g}_{k,t}a_{k,t})$ bits,

and we will allocate power such that

$$\mathbb{E}\{|\boldsymbol{x}_{c,t}|_1^2\} = \mathbb{E}\{|a^*_{\ell,t}|^2\} \doteq P,\ \mathbb{E}\{|\boldsymbol{x}_{c,t}|_i^2\} = \mathbb{E}\{|a_{k,t}|^2\} \doteq P^\alpha$$

where $|\boldsymbol{x}_{c,t}|_i, i = 1, 2, \cdots, K$, denotes the magnitude of the i^{th} entry of vector $\boldsymbol{x}_{c,t}$.

Remark 1 *Recall that instead of employing matrix notation, after normalization, we use the concept of signal duration dur($\boldsymbol{x}$) required for the transmission of some vector $\boldsymbol{x}$. We also note that due to time normalization, the time index $t \in [0, T]$, need not be an integer.*

For any α, our scheme will be defined by an integer $\eta \in [\Gamma, K-1] \cap \mathbb{Z}$, which will be chosen as

$$\eta = \arg \max_{\eta' \in [\Gamma, K-1] \cap \mathbb{Z}} \{\eta' \ :\ \alpha_{b,\eta'} \leq \alpha\} \tag{3.35}$$

for

$$\alpha_{b,\eta} = \frac{\eta - \Gamma}{\Gamma(H_K - H_\eta - 1) + \eta}. \tag{3.36}$$

η defines the amount of cached information that will be folded ($\{W^{c,f}_{R_k,\tau}\}_{\tau \in \Psi_\eta \backslash \Psi^{(k)}_\eta}$), and thus also the amount of cached information that will not be folded ($\{W^{c,\overline{f}}_{R_k,\tau}\}_{\tau \in \Psi_\eta \backslash \Psi^{(k)}_\eta}$) and which will be exclusively carried by the different $a_{k,t}$. In all cases,

- all of $\{X_\psi\}_{\psi \in \Psi_{\eta+1}}$ (which are functions of the cached-and-to-be-folded $\{W^{c,f}_{R_k,\tau}\}_{\tau \in \Psi_\eta \backslash \Psi^{(k)}_\eta}$) will be exclusively carried by $\boldsymbol{x}_{c,t},\ t \in [0, T]$, while
- all of the uncached $W^{\overline{c}}_{R_k}$ (for each $k = 1, \ldots, K$) and all of the cached but unfolded $\{W^{c,\overline{f}}_{R_k,\tau}\}_{\tau \in \Psi_\eta \backslash \Psi^{(k)}_\eta}$ will be exclusively carried by $a_{k,t}, t \in [0, T]$.

Transmission of $\{X_\psi\}_{\psi \in \Psi_{\eta+1}}$: From [67], we know that the transmission relating to $\boldsymbol{x}_{c,t}$ can be treated independently from that of $a_{k,t}$ with the assistance of $a^*_{\ell,t}$, simply because — as we will further clarify later on — the $a_{k,t}$ do not actually interfere with decoding of $\boldsymbol{x}_{c,t}$, as a result of the scheme, and as a result of the chosen power and rate allocations which jointly adapt to the CSIT quality α. For this reason, we can treat the transmission of $\boldsymbol{x}_{c,t}$ separately.

Hence we first focus on the transmission of $\{X_\psi\}_{\psi\in\Psi_{\eta+1}}$, which will be sent using $\boldsymbol{x}_{c,t},\ t\in[0,T]$ using the last $K-\eta$ phases of the QMAT algorithm in [67] corresponding to having the ZF symbols $a_{k,t}$ set to zero. For ease of notation, we will label these phases starting from phase $\eta+1$ and terminating in phase K. Each phase $j=\eta+1,\dots,K$ aims to deliver order-j folded messages (cf. (3.31)), and will do so gradually: phase j will try to deliver (in addition to other information) $N_j:=(K-j+1)\binom{K}{j}$ order-j messages which carry information that has been requested by j users, and in doing so, it will generate $N_{j+1}:=j\binom{K}{j+1}$ signals that are linear combinations of received signals from $j+1$ different users, and where these N_{j+1} signals will be conveyed in the next phase $j+1$. During the last phase $j=K$, the transmitter will send fully common symbols that are useful and decoded by all users, thus allowing each user to go back and retroactively decode the information of phase $j=K-1$, which will then be used to decode the information in phase $j=K-2$ and so on, until they reach phase $j=\eta+1$ (first transmission phase) which will complete the task. We proceed to describe these phases. We will use T_j to denote the duration of phase j.

Phase $\eta+1$: In this first phase of duration $T_{\eta+1}$, the information in $\{X_\psi\}_{\psi\in\Psi_{\eta+1}}$ is delivered by $\boldsymbol{x}_{c,t},\ t\in[0,T_{\eta+1}]$, which can also be rewritten in the form of a sequential transmission of shorter-duration K-length vectors

$$\boldsymbol{x}_\psi=[x_{\psi,1},\dots,x_{\psi,K-\eta},0,\dots,0]^T \tag{3.37}$$

for different ψ, where each vector $\boldsymbol{x}_\psi$ carries exclusively the information from each X_ψ, and where this information is uniformly split among the $K-\eta$ independent scalar entries $x_{\psi,i},\ i=1,\dots,K-\eta$, each carrying

$$\frac{|X_\psi|}{(K-\eta)}=\frac{f(1-\gamma-\alpha T)}{\binom{K-1}{\eta}(K-\eta)} \tag{3.38}$$

bits (cf. (6.14)). Hence, given that the allocated rate for $\boldsymbol{x}_{c,t}$ (and thus the allocated rate for each $\boldsymbol{x}_\psi$) is $(1-\alpha)f$, we have that the duration of each $\boldsymbol{x}_\psi$ is

$$\mathrm{dur}(\boldsymbol{x}_\psi)=\frac{|X_\psi|}{(K-\eta)(1-\alpha)f}. \tag{3.39}$$

Given that $|\mathcal{X}_\Psi|=\binom{K}{\eta+1}$, then

$$T_{\eta+1}=\binom{K}{\eta+1}\mathrm{dur}(\boldsymbol{x}_\psi)=\frac{\binom{K}{\eta+1}|X_\psi|}{(K-\eta)(1-\alpha)f}. \tag{3.40}$$

After each transmission of $\boldsymbol{x}_\psi$, the received signal $y_{k,t},\ t\in[0,T_{\eta+1}]$ of desired user k ($k\in\psi$) takes the form

$$y_{k,t}=\underbrace{\boldsymbol{h}_{k,t}^T\mathbf{G}_{c,t}\boldsymbol{x}_{c,t}}_{L_{\psi,k},\mathrm{power}\ \doteq P}+\underbrace{\boldsymbol{h}_{k,t}^T\sum_{\ell\in\psi}\boldsymbol{g}_{\ell,t}a_{\ell,t}^*}_{\doteq P^{1-\alpha}}+\underbrace{\boldsymbol{h}_{k,t}^T\boldsymbol{g}_{k,t}a_{k,t}}_{P^\alpha} \tag{3.41}$$

while the received signal for the other users $k \in [K]\backslash\psi$ takes the form

$$y_{k,t}=\underbrace{\boldsymbol{h}_{k,t}^T\boldsymbol{g}_{k,t}a_{k,t}^*}_{\text{power}\doteq P}+\underbrace{\boldsymbol{h}_{k,t}^T\sum_{\substack{\ell\in\psi\\ \ell\neq k}}\boldsymbol{g}_{\ell,t}a_{\ell,t}^*+\underbrace{\boldsymbol{h}_{k,t}^T\mathbf{G}_{c,t}\boldsymbol{x}_{c,t}}_{L_{\psi,k},\doteq P^{1-\alpha}}}_{i_{\psi,k},\doteq P^{1-\alpha}}+\underbrace{\boldsymbol{h}_{k,t}^T\boldsymbol{g}_{k,t}a_{k,t}}_{P^\alpha} \tag{3.42}$$

where in both cases, we ignored the Gaussian noise and the ZF noise up to P^0. Each user $k \in [K]$ receives a linear combination $L_{\psi,k}$ of the transmitted $K-\eta$ symbols $x_{\psi,1}, x_{\psi,2}, \ldots, x_{\psi,K-\eta}$. Next the transmitter will somehow send an additional $K-\eta-1$ signals $L_{\psi,k},\ k \in [K]\backslash\psi$ (linear combinations of $x_{\psi,1}, x_{\psi,2}, \ldots, x_{\psi,K-\eta}$ as received — up to noise level — at each user $k' \in [K]\backslash\psi$) which will help each user $k \in \psi$ resolve the already sent $x_{\psi,1}, x_{\psi,2}, \ldots, x_{\psi,K-\eta}$. This will be done in the next phase $j = \eta+2$.

Phase $\eta+2$: The challenge now is for signals $\boldsymbol{x}_{c,t},\ t \in (T_{\eta+1}, T_{\eta+1}+T_{\eta+2}]$ to convey all the messages of the form

$$i_{\psi,k},\ \forall k \in [K]\backslash\psi,\ \forall\psi \in \Psi_{\eta+1}$$

to each receiver $k \in \psi$. Note that $\boldsymbol{h}_{k,t}^T\sum_{\substack{\ell\in\psi\\ \ell\neq k}}\boldsymbol{g}_{\ell,t}a_{\ell,t}^*$ is the residual interference of the previous round which can be removed easily and each of the above linear combinations, is now — during this phase — available (up to noise level) at the transmitter. Let

$$\Psi_{\eta+2} = \{\psi \in [K]\ :\ |\psi| = \eta+2\} \tag{3.43}$$

and consider for each $\psi \in \Psi_{\eta+2}$, a transmitted vector

$$\boldsymbol{x}_\psi = [x_{\psi,1}, \ldots, x_{\psi,K-\eta-1}, 0, \ldots, 0]^T$$

which carries the contents of $\eta+1$ ($l = 1, \cdots, \eta+1$) different elements

$$f_l = (\bar{i}_{\psi\backslash\{k\},k} \oplus \bar{i}_{\psi\backslash\{k'\},k'}), k \neq k', k, k' \in \psi$$

where $\bar{i}_{\psi\backslash\{P_k\},P_k}$ is the quantization of $i_{\psi\backslash\{P_k\},P_k}$ from phase 1. f_l are predetermined and known at each receiver. The transmission of $\{\boldsymbol{x}_\psi\}_{\forall\psi\in\Psi_{\eta+2}}$ is sequential.

It is easy to see that there is a total of $(\eta+1)\binom{K}{\eta+2}$ XORs in the form of f_l, each of which can be considered as an order-$(\eta+2)$ signal intended for $\eta+2$ receivers in ψ. Using this, and following the same steps used in phase $\eta+1$, we calculate that

$$T_{\eta+2} = \binom{K}{\eta+2}\text{dur}(\boldsymbol{x}_\psi) = T_{\eta+1}\frac{\eta+1}{\eta+2}. \tag{3.44}$$

We now see that for each ψ, each receiver $k \in \psi$ recalls their own observation $i_{\psi\backslash\{k\},k}$ from the previous phase, and removes it from f_l, thus now being

able to acquire the $\eta+1$ independent linear combinations $\{L_{\psi\setminus\{k\},k}\}_{\forall k\in\psi\setminus\{k\}}$ by easily removing the auxiliary symobls. The same holds for each other user $k\in\psi$.

After this phase, we use $L_{\psi,k},\psi\in\Psi_{\eta+2}$ to denote the received signal of QMAT at receiver k. Like before, each receiver $k,k\in\psi$ needs $K-\eta-2$ extra observations of $x_{\psi,1},\ldots,x_{\psi,K-\eta-1}$ which will be seen from $L_{\psi,k},\forall k\notin\psi$, which will come from order-$(\eta+3)$ messages that are created by the transmitter and which will be sent in the next phase.

Phase j $(\eta+3\leq j\leq K)$: Generalizing the described approach to any phase $j\in[\eta+3,\ldots,K]$, we will use $\boldsymbol{x}_{c,t},\ t\in[\sum_{i=\eta+1}^{j-1}T_i,\sum_{i=\eta+1}^{j}T_i]$ to convey all the messages of the form

$$i_{\psi,k},\ \forall k\in[K]\backslash\psi,\ \forall\psi\in\Psi_{j-1}$$

to each receiver $k\in\psi$. For each

$$\psi\in\Psi_j:=\{\psi\in[K]\ :\ |\psi|=j\} \tag{3.45}$$

each transmitted vector

$$\boldsymbol{x}_\psi=[x_{\psi,1},\ldots,x_{\psi,K-j-1},0,\ldots,0]^T$$

will carry the contents of $j-1$ different XORs $f_l,l=1,\ldots,j-1$ of the j elements $\{\bar{i}_{\psi\setminus\{k\},k}\}_{\forall k\in\psi}$ created by the transmitter. After the sequential transmission of $\{\boldsymbol{x}_\psi\}_{\forall\psi\in\Psi_j}$, each receiver k can obtain the $j-1$ independent linear combinations $\{L_{\psi\setminus\{k\},k}\}_{\forall k\in\psi\setminus\{k\}}$ again by removing the auxiliary symbols. The same holds for each other user $k'\in\psi$. As with the previous phases, we can see that

$$T_j=T_{\eta+1}\frac{\eta+1}{j},\ j=\eta+3,\ldots,K. \tag{3.46}$$

This process terminates with phase $j=K$, during which each

$$\boldsymbol{x}_\psi=[x_{\psi,1},0,0,\ldots,0]^T$$

carries a single scalar that is decoded easily by all. Based on this, backwards decoding will allow for users to retrieve $\{X_\psi\}_{\psi\in\Psi_{\eta+1}}$. This is described immediately afterwards. In treating the decoding part, we briefly recall that each $a_{k,t},\ k=1,\ldots,K$ carries (during period $t\in[0,T]$), all of the uncached $W^{\bar{c}}_{R_k}$ and all of the unfolded $\{W^{c,\bar{f}}_{R_k,\tau}\}_{\tau\in\Psi_\eta\setminus\Psi_\eta^{(k)}}$.

3.5.4 Decoding

The whole transmission lasts $K-\eta$ phases. For each phase $j,j=\eta+1,\cdots,K$ and the corresponding ψ, the received signal $y_{k,t},\ t\in[\sum_{i=\eta+1}^{j-1}T_i,\sum_{i=\eta+1}^{j}T_i]$

of desired user k ($k \in \psi$) takes the form

$$y_{k,t} = \underbrace{\boldsymbol{h}_{k,t}^T \mathbf{G}_{c,t} \boldsymbol{x}_{c,t}}_{\text{power} \doteq P} + \underbrace{\boldsymbol{h}_{k,t}^T \sum_{\ell \in \psi} \boldsymbol{g}_{\ell,t} a_{\ell,t}^*}_{\doteq P^{1-\alpha}} + \underbrace{\boldsymbol{h}_{k,t}^T \boldsymbol{g}_{k,t} a_{k,t}}_{P^\alpha} \tag{3.47}$$

while the received signal for the other users $k \in [K] \backslash \psi$ takes the form

$$y_{k,t} = \underbrace{\boldsymbol{h}_{k,t}^T \boldsymbol{g}_{k,t} a_{k,t}^*}_{\text{power} \doteq P} + \underbrace{\underbrace{\boldsymbol{h}_{k,t}^T \sum_{\substack{\ell \in \psi \\ \ell \neq k}} \boldsymbol{g}_{\ell,t} a_{\ell,t}^*}_{L_{\psi,k'} \doteq P^{1-\alpha}} + \underbrace{\boldsymbol{h}_{k,t}^T \mathbf{G}_{c,t} \boldsymbol{x}_{c,t}}_{\doteq P^{1-\alpha}}}_{i_{\psi,k} \doteq P^{1-\alpha}} + \underbrace{\boldsymbol{h}_{k,t}^T \boldsymbol{g}_{k,t} a_{k,t}}_{P^\alpha}$$

where again we ignored the Gaussian noise and the ZF noise up to P^0. As we see in [67], after each phase, $i_{\psi,k}$ is first quantized with $(1-2\alpha)^+ \log P$ bits, which results in a residual quantizaton noise $n_{\psi,k}$ with power scaling as P^α. Then, the transmitter quantizes the quantization noise $n_{\psi,k}$ with an additional $\alpha \log P$ bits, which will be carried by the auxiliary data symbols in the corresponding phase in the next round (here we 'load' this round with additional requests from the users). In this way, we can see that the 'common' signal $\boldsymbol{x}_{c,t}$ can also be decoded at user $k \in [K] \backslash \psi$ with the assistance of an auxiliary data symbol from the next round. After this, each user k will remove $\boldsymbol{h}_{k,t}^T \mathbf{G}_{c,t} \boldsymbol{x}_{c,t}$ from their received signals, and readily decode their private symbols $a_{k,t},\ t \in [0,T]$, thus allowing for retrieval of their own unfolded $\{W_{R_k,\psi\backslash\{k\}}^{c,\overline{f}}\}_{\psi \in \Psi_{\eta+1}}$ and uncached $W_{R_k}^{\overline{c}}$. In terms of decoding the common information, as discussed above, each receiver k will perform a backwards reconstruction of the sets of overheard equations

$$\begin{gathered} \{L_{\psi,k},\ \forall k \in [K] \backslash \psi\}_{\forall \psi \in \Psi_K} \\ \vdots \\ \downarrow \\ \{L_{\psi,k},\ \forall k \in [K] \backslash \psi\}_{\forall \psi \in \Psi_{\eta+2}} \end{gathered}$$

until phase $\eta + 2$. At this point, each user k has enough observations to recover the original $K - \eta$ symbols $x_{\psi,1}, x_{\psi,2}, \ldots, x_{\psi,K-\eta}$ that fully convey X_ψ, hence each user k can reconstruct their own set $\{W_{R_k,\psi\backslash\{k\}}^{c,f}\}_{\psi \in \Psi_{\eta+1}}$ which, combined with the information from the $a_{k,t},\ t = [0,T]$ allow for each user k to reconstruct $\{W_{R_k,\psi\backslash\{k\}}^{c}\}_{\psi \in \Psi_{\eta+1}}$ which is then combined with Z_k to allow for reconstruction of the requested file W_{R_k}.

3.5.5 Calculation of T

To calculate T, we recall from (3.46) that

$$T = \sum_{j=\eta+1}^{K} T_j = T_{\eta+1} \sum_{j=\eta+1}^{K} \frac{\eta+1}{j} = (\eta+1)(H_K - H_\eta) T_{\eta+1}$$

which combines with (3.38) and (3.40) to give

$$T = \frac{(K-\Gamma)(H_K - H_\eta)}{(K-\eta) + \alpha(\eta + K(H_K - H_\eta - 1))} \tag{3.48}$$

as stated in Theorem 3.1. The bound by $T = 1-\gamma$ seen in the theorem, corresponds to the fact that the above expression (3.48) applies, as is, only when $\alpha \leq \alpha_{b,K-1} = \frac{K(1-\gamma)-1}{(K-1)(1-\gamma)}$ which corresponds to $\eta = K-1$ (where X_ψ are fully common messages, directly desired by all), for which we already get the best possible $T = 1-\gamma$, and hence there is no need to go beyond $\alpha = \frac{K(1-\gamma)-1}{(K-1)(1-\gamma)}$.

3.6 Conclusions

This work studied the previously unexplored interplay between coded-caching and CSIT feedback quality and timeliness. This is motivated by the fact that CSIT and coded caching are two powerful ingredients that are hard to obtain, and by the fact that these ingredients are intertwined in a synergistic and competing manner. In addition to the substantial cache-aided DoF gains revealed here, the results suggest the interesting practical ramification that distributing predicted content 'during the night', can offer continuous amelioration of the load of predicting and disseminating CSIT during the day.

3.7 Appendix - Proof of Lemma 3.1 (Lower bound on T^*)

We here note that for the outer (lower) bound to hold, we will make the common assumption that the current channel state must be independent of the previous channel-estimates and estimation errors, *conditioned on the current estimate* (there is no need for the channel to be i.i.d. in time). We will also make the common assumption that the channel is drawn from a continuous ergodic distribution such that all the channel matrices and all their sub-matrices are full rank almost surely.

To lower bound T, we first consider the easier problem where we want to serve $s \leq K$ different files to s users, each with access to all caches. We also consider that we repeat this (easier) last experiment $\lfloor \frac{N}{s} \rfloor$ times, thus spanning a total duration of $T\lfloor \frac{N}{s} \rfloor$ (and up to $\lfloor \frac{N}{s} \rfloor s$ files delivered). At this point, we transfer to the equivalent setting of the s-user MISO BC with delayed CSIT and imperfect current CSIT, and a side-information multicasting link to the receivers, of capacity d_m (files per time slot). Under the assumption that in this latter setting, decoding happens at the end of communication, and once we set

$$d_m T \lfloor \frac{N}{s} \rfloor = sM \tag{3.49}$$

(which guarantees that the side information from the side link, throughout the communication process, matches the maximum amount of information in the caches), we have that

$$T\lfloor\frac{N}{s}\rfloor d'_{\Sigma}(d_m) \geq \lfloor\frac{N}{s}\rfloor s \tag{3.50}$$

where $d'_{\Sigma}(d_m)$ is any sum-DoF upper bound on the above s-user MISO BC channel with delayed CSIT and the aforementioned side link. Using the bound

$$d'_{\Sigma}(d_m) = s\alpha + \frac{s}{H_s}(1-\alpha+d_m)$$

from Lemma 3.2 (see below), and applying (3.49), we get

$$T\lfloor\frac{N}{s}\rfloor\big(s\alpha + \frac{s}{H_s}(1-\alpha+\frac{sM}{T\lfloor\frac{N}{s}\rfloor})\big) \geq \lfloor\frac{N}{s}\rfloor s \tag{3.51}$$

and thus that

$$T \geq \frac{1}{(H_s\alpha+1-\alpha)}(H_s - \frac{Ms}{\lfloor\frac{N}{s}\rfloor}) \tag{3.52}$$

which implies a lower bound on the original s-user problem. Maximization over all s, gives the desired bound on the optimal T^*

$$T^* \geq \max_{s\in\{1,\ldots,\min(N,K)\}} \frac{1}{(H_s\alpha+1-\alpha)}(H_s - \frac{Ms}{\lfloor\frac{N}{s}\rfloor}) \tag{3.53}$$

required for the original K-user problem. This concludes the proof of Lemma 3.1.

3.7.1 Sum-DoF bound for the s-user MISO BC, with delayed CSIT, α-quality current CSIT, and additional side information

We begin with the statement of the lemma, which we prove immediately below.

Lemma 3.2 *For the s-user MISO BC, with delayed CSIT, α-quality current CSIT, and an additional parallel side-link of capacity that scales as $d_m \log P$, the sum-DoF is upper bounded as*

$$d_{\Sigma}(d_m) \leq s\alpha + \frac{s}{H_s}(1-\alpha+d_m). \tag{3.54}$$

Proof. Our proof traces the proof of [69], adapting for the additional α-quality current CSIT.

Consider a permutation π of the set $\mathcal{E} = \{1, 2, \cdots, s\}$. For any user $k, k \in \mathcal{E}$, we provide the received signals $y_k^{[n]}$ as well as the message W_k of user k to

user $k+1, k+2, \cdots, s$. We use $y_0^{[n]}$ to denote the output of the side-link and we also define the following notations

$$\Omega^{[n]} := \{\boldsymbol{h}_k^{[n]}\}_{k=1}^{s}, \quad \hat{\Omega}^{[n]} := \{\hat{\boldsymbol{h}}_k^{[n]}\}_{k=1}^{s}, \quad \mathcal{U}^{[n]} := \{\Omega^{[n]}, \hat{\Omega}^{[n]}\},$$
$$\boldsymbol{h}_k^{[t]} := \{\boldsymbol{h}_k^{(i)}\}_{i=1}^{t}, \quad y_k^{[t]} := \{y_k^{(i)}\}_{i=1}^{t}, t = 1, 2, \cdots, n,$$
$$W_{[k]} := \{W_1, W_2, \cdots, W_k\}, \quad y_{[k]}^{[n]} := \{y_1^{[n]}, y_2^{[n]}, \cdots, y_k^{[n]}\}.$$

Then for $k = 1, 2, \cdots, s$, we have

$$\begin{aligned}
n(R_k - \epsilon_n) &\le I(W_k; y_{[k]}^{[n]}, y_0^{[n]}, W_{[k-1]} | \mathcal{U}^{[n]}) && (3.55)\\
&= I(W_k; y_{[k]}^{[n]}, y_0^{[n]} | W_{[k-1]}, \mathcal{U}^{[n]}) && (3.56)\\
&= I(W_k; y_{[k]}^{[n]} | W_{[k-1]}, \mathcal{U}^{[n]}) + I(W_k; y_0^{[n]} | y_{[k]}^{[n]}, W_{[k-1]}, \mathcal{U}^{[n]})\\
&= h(y_{[k]}^{[n]} | W_{[k-1]}, \mathcal{U}^{[n]}) - h(y_{[k]}^{[n]} | W_{[k]}, \mathcal{U}^{[n]})\\
&\quad + h(y_0^{[n]} | y_{[k]}^{[n]}, W_{[k-1]}, \mathcal{U}^{[n]}) - h(y_0^{[n]} | y_{[k]}^{[n]}, W_{[k]}, \mathcal{U}^{[n]}) && (3.57)
\end{aligned}$$

where (3.55) follows from Fano's inequality, where (3.56) holds due to the fact that the messages are independent, and where the last two steps use the basic chain rule. Note that $W_0 = 0$.

$$\begin{aligned}
&\sum_{k=1}^{s-1} \Big(\frac{h(y_{[k+1]}^{[n]} | W_{[k]}, \mathcal{U}^{[n]})}{k+1} - \frac{h(y_{[k]}^{[n]} | W_{[k]}, \mathcal{U}^{[n]})}{k} \Big)\\
&= \sum_{t=1}^{n} \sum_{k=1}^{s-1} \Big(\frac{h(y_1^{(t)}, \cdots, y_{k+1}^{(t)} | y_1^{[t-1]}, \cdots, y_{k+1}^{[t-1]}, W_{[k]}, \mathcal{U}^{[n]})}{k+1}\\
&\quad - \frac{h(y_1^{(t)}, \cdots, y_k^{(t)} | y_1^{[t-1]}, \cdots, y_k^{[t-1]}, W_{[k]}, \mathcal{U}^{[n]})}{k} \Big) && (3.58)\\
&= \sum_{t=1}^{n} \sum_{k=1}^{s-1} \Big(\frac{h(y_1^{(t)}, \cdots, y_{k+1}^{(t)} | y_1^{[t-1]}, \cdots, y_{k+1}^{[t-1]}, W_{[k]}, \mathcal{U}^{[t]})}{k+1}\\
&\quad - \frac{h(y_1^{(t)}, \cdots, y_k^{(t)} | y_1^{[t-1]}, \cdots, y_k^{[t-1]}, W_{[k]}, \mathcal{U}^{[t]})}{k} \Big) && (3.59)\\
&\le \sum_{t=1}^{n} \sum_{k=1}^{s-1} \Big(\frac{h(y_1^{(t)}, \cdots, y_{k+1}^{(t)} | y_1^{[t-1]}, \cdots, y_{k+1}^{[t-1]}, W_{[k]}, \mathcal{U}^{[t]})}{k+1}\\
&\quad - \frac{h(y_1^{(t)}, \cdots, y_k^{(t)} | y_1^{[t-1]}, \cdots, y_{k+1}^{[t-1]}, W_{[k]}, \mathcal{U}^{[t]})}{k} \Big) && (3.60)\\
&\le \sum_{t=1}^{n} \sum_{k=1}^{s-1} \frac{1}{k+1} \alpha \log P + n \cdot o(\log P) && (3.61)\\
&= n(H_s - 1)\alpha \log P + n \cdot o(\log P) && (3.62)
\end{aligned}$$

where (3.58) follows from the linearity of the summation, where (3.59) holds since the received signal is independent of the future channel state information, where (3.60) uses the fact that conditioning reduces entropy, and where (3.62) is from the fact that Gaussian distribution maximizes differential entropy under the covariance constraint and from [23, Lemma 2]. From (3.57), we then have

$$\begin{aligned}
\sum_{k=1}^{s}\frac{n(R_k-\epsilon_n)}{k} &\le \sum_{k=1}^{s-1}\Big(\frac{h(y_{[k+1]}^{[n]}|W_{[k]},\mathcal{U}^{[n]})}{k+1}-\frac{h(y_{[k]}^{[n]}|W_{[k]},\mathcal{U}^{[n]})}{k}\Big)+h(y_1^{[n]}|\mathcal{U}^{[n]})\\
&\quad+\sum_{k=1}^{s-1}\Big(\frac{h(y_0^{[n]}|y_{[k+1]}^{[n]},W_{[k]},\mathcal{U}^{[n]})}{k+1}-\frac{h(y_0^{[n]}|y_{[k]}^{[n]},W_{[k]},\mathcal{U}^{[n]})}{k}\Big)\\
&\quad-\frac{1}{s}h(y_{[s]}^{[n]}|W_{[s]},\mathcal{U}^{[n]})+h(y_0^{[n]}|y_1^{[n]},\mathcal{U}^{[n]})-\frac{1}{s}h(y_0^{[n]}|y_{[s]}^{[n]},W_{[s]},\mathcal{U}^{[n]})\\
&\le n(H_s-1)\alpha\log P+\underbrace{h(y_1^{[n]}|\mathcal{U}^{[n]})}_{\le n\log P}+\underbrace{h(y_0^{[n]}|y_1^{[n]},\mathcal{U}^{[n]})}_{\le n\cdot d_m\log P}\\
&\quad+\sum_{k=1}^{s-1}\Big((\frac{1}{k+1}-\frac{1}{k})h(y_0^{[n]}|y_{[k]}^{[n]},W_{[k]},\mathcal{U}^{[n]})\Big)+n\cdot o(\log P)\\
&\le n(H_s-1)\alpha\log P+n\log P+n\cdot d_m\log P+n\cdot o(\log P).
\end{aligned}\tag{3.63}$$

where the second inequality is from (3.62), where the last inequality follows from the fact that entropy is non-negative and $\frac{1}{k+1}-\frac{1}{k}\le 0$. Dividing by $n\log P$ and letting $P\to\infty$ gives

$$\sum_{k=1}^{s}\frac{d_k}{k}\le(H_s-1)\alpha+1+d_m \tag{3.64}$$

which implies that

$$d_\Sigma(d_m)\le s\alpha+\frac{s}{H_s}(1-\alpha+d_m) \tag{3.65}$$

which completes the proof of Lemma 3.2. ■

3.8 Appendix - Bounding the gap to optimal

Our aim here is to show that

$$\frac{T(\gamma,\alpha>0)}{T^*(\gamma,\alpha>0)}<4$$

and we will do so by showing that the above gap is smaller than the gap we calculated above for $\alpha=0$ in Appendix in the Chapter 2, which was again

bounded above by 4. For this, we will use the expression[8]

$$T(\gamma, \alpha > 0) = \frac{(1-\gamma)(H_K - H_\Gamma)}{\alpha(H_K - H_\Gamma) + (1-\alpha)(1-\gamma)} \tag{3.66}$$

from Theorem 3.2, and the expression

$$T^*(\gamma, \alpha > 0) \geq \max_{s \in \{1,\ldots,\lfloor \frac{N}{M} \rfloor\}} \frac{1}{(H_s\alpha + 1 - \alpha)}(H_s - \frac{Ms}{\lfloor \frac{N}{s} \rfloor})$$

from Lemma 3.1. Hence we have

$$\frac{T}{T^*} \leq \frac{\frac{(1-\gamma)(H_K - H_{K\gamma})}{\alpha(H_K - H_{K\gamma}) + (1-\alpha)(1-\gamma)}}{\max_{s \in \{1,\ldots,\lfloor \frac{N}{M} \rfloor\}} \frac{1}{(H_s\alpha+1-\alpha)}(H_s - \frac{Ms}{\lfloor \frac{N}{s} \rfloor})} \tag{3.67}$$

$$\leq \underbrace{\frac{\frac{(1-\gamma)(H_K - H_{K\gamma})}{\alpha(H_K - H_{K\gamma}) + (1-\alpha)(1-\gamma)}}{\frac{1}{(H_{s_c}\alpha+1-\alpha)}(H_{s_c} - \frac{Ms_c}{\lfloor \frac{N}{s_c} \rfloor})}}_{g(s_c,\gamma)} \tag{3.68}$$

where $s = s_c \in \{1, \ldots, \lfloor \frac{N}{M} \rfloor\}$, but where this s_c will be chosen here to be exactly the same as in the case of $\alpha = 0$. This will be useful because, for that case of $\alpha = 0$, we have already proved that the same specific s_c guarantees that

$$\frac{H_K - H_{K\gamma}}{H_{s_c} - \frac{Ms_c}{\lfloor \frac{N}{s_c} \rfloor}} < 4 \tag{3.69}$$

for the appropriate ranges of γ. This will apply towards bounding (3.68).

The proof is broken in two cases, corresponding to $\gamma \in [\frac{1}{36}, \frac{K-1}{K}]$, and $\gamma \in [0, \frac{1}{36}]$.

Case 1 $(\alpha > 0, \gamma \in [\frac{1}{36}, \frac{K-1}{K}])$

As when $\alpha = 0$ (cf. [71]), we again set $s = 1$, which reduces (3.68) to

$$\frac{T(\alpha > 0, \gamma)}{T^*(\alpha > 0, \gamma)} \leq \frac{\frac{(1-\gamma)(H_K - H_{K\gamma})}{\alpha(H_K - H_{K\gamma}) + (1-\alpha)(1-\gamma)}}{1-\gamma}.$$

For this case — when α was zero, and when we chose the same $s = 1$ — we have already proved that $\frac{T(\alpha=0,\gamma)}{1-\gamma} < 4$. As a result, since $T(\alpha > 0, \gamma) < T(\alpha = 0, \gamma)$, and since $1 - \gamma \leq T^*$, we conclude that $\frac{T(\alpha>0,\gamma)}{T^*} < 4$, $\gamma \in [\frac{1}{36}, \frac{K-1}{K}]$ which completes this part of the proof.

[8] We note that the here derived upper bound on the gap corresponding to the T in Theorem 3.2, automatically applies as an upper bound to the gap corresponding to the T from Theorem 3.1, because the latter T is smaller than the former.

Case 2 ($\alpha > 0, \gamma \in [0, \frac{1}{36}]$)

Going back to (3.68), we now aim to bound

$$g(s_c, \gamma) \triangleq \frac{\frac{(1-\gamma)(H_K - H_{K\gamma})}{\alpha(H_K - H_{K\gamma}) + (1-\alpha)(1-\gamma)}}{\frac{1}{(H_{s_c}\alpha + 1 - \alpha)}(H_{s_c} - \frac{Ms_c}{\lfloor \frac{N}{s_c} \rfloor})} < 4. \tag{3.70}$$

We already know from the case of $\alpha = 0$ (cf. (3.69)) that

$$\frac{H_K - H_{K\gamma}}{H_{s_c} - \frac{Ms_c}{\lfloor \frac{N}{s_c} \rfloor}} < 4 \tag{3.71}$$

holds. Hence we will prove that

$$g(s_c, \gamma) \leq \frac{H_K - H_{K\gamma}}{H_{s_c} - \frac{Ms_c}{\lfloor \frac{N}{s_c} \rfloor}} \tag{3.72}$$

to guarantee the bound. We note that (3.72) is implied by

$$H_{s_c} \leq \frac{H_K - H_{K\gamma}}{1 - \gamma} \tag{3.73}$$

which is implied by

$$\log(s_c) \leq \frac{\log(1/\gamma)}{1 - \gamma} - \epsilon_6, \ \epsilon_6 = H_6 - \log(6) \tag{3.74}$$

because $H_{s_c} \leq \log(s_c) + \epsilon_6, \forall s_c \geq 6, \forall \gamma \in [0, \frac{1}{36}], \forall K$. Furthermore (3.74) is implied by

$$\frac{1}{2}\log(\frac{1}{\gamma}) \leq \frac{\log(1/\gamma)}{1 - \gamma} - \epsilon_6 \tag{3.75}$$

because $\gamma \in [\frac{1}{(s_c+1)^2}, \frac{1}{s_c^2}]$ means that $s_c \leq \sqrt{\frac{1}{\gamma}}$. Since $\frac{1}{1-\gamma} \geq 1$, then (3.75) is implied by

$$\frac{1}{2}\log(\frac{1}{\gamma}) \leq \log(\frac{1}{\gamma}) - \epsilon_6. \tag{3.76}$$

It is obvious that (3.76) holds since $\gamma \leq \frac{1}{36}$. Towards this, by proving (3.76), we guarantee (3.70) and the desired bound. This completes the proof.

Chapter 4

Feedback-Aided Coded Caching with Small Caches

This work continues to explore coded caching in the symmetric K-user cache-aided MISO BC with imperfect CSIT-type feedback, but for the specific case where the cache size is much smaller than the library size. Building on the recently explored synergy between caching and delayed-CSIT, and building on the tradeoff between caching and CSIT quality, the work proposes new schemes that boost the impact of small caches, focusing on the case where the cumulative cache size is smaller than the library size. This small-cache setting places an additional challenge due to the fact that some of the library content must remain entirely uncached, which forces us to dynamically change the caching redundancy to compensate for this. Our proposed scheme is near-optimal, and the work identifies the optimal cache-aided degrees-of-freedom (DoF) performance within a factor of 4.

4.1 Introduction

Our aim here is to explore the ideas in the previous chapters, and to further our understanding of the *joint* effect of coded caching — now with small caches — and (variable quality) feedback, in removing interference and in eventually improving performance. The connections between these ingredients (caches and feedback) will prove particularly crucial here, in boosting the effect of otherwise insufficiently large caches.

4.1.1 K-user feedback-aided symmetric MISO BC with small caches

The model remains the same as in Chapter 3, and it corresponds to the symmetric K-user wireless MISO BC with a K-antenna transmitter, and K single-antenna receiving users. Our emphasis here will be on the small cache regime where the cumulative cache size is less than the library size ($K\gamma \leq 1$, i.e., $KM \leq N$), and which will force us to account for the fact that not all content can be cached. We will also touch upon the general small-cache setting where the individual cache size is much less than the library size ($M \ll N$, i.e., $\gamma \ll 1$). As before, we will consider the mixed CSIT model.

Related work on smaller-sized caches can be found in [73] which considered coded caching — in the single stream case with $K \geq N$ — in the presence of very small caches with $KM \leq 1$, corresponding to the case where pooling all the caches together, can at most match the size of a single file.

4.2 Main results

The following identifies, up to a factor of 4, the optimal T^*, for all $\Gamma \in [0,1]$. We use the expression

$$\alpha_{b,\eta} = \frac{\eta - \Gamma}{\Gamma(H_K - H_\eta - 1) + \eta},\ \eta = 1, \ldots, K-1. \tag{4.1}$$

Theorem 4.1 *In the (K, M, N, α) cache-aided MISO BC with N files, $K \leq N$ users, and $KM \leq N$ ($\Gamma \leq 1$), then for $\eta = 1, \ldots, K-2$,*

$$T = \begin{cases} \frac{H_K - \Gamma}{1-\alpha+\alpha H_K}, & 0 \leq \alpha < \alpha_{b,1} \\ \frac{(K-\Gamma)(H_K - H_\eta)}{(K-\eta)+\alpha(\eta + K(H_K - H_\eta - 1))}, & \alpha_{b,\eta} \leq \alpha < \alpha_{b,\eta+1} \\ 1-\gamma, & \frac{K-1-\Gamma}{(K-1)(1-\gamma)} \leq \alpha \leq 1 \end{cases} \tag{4.2}$$

is achievable, and has a gap from optimal that is less than 4 ($\frac{T}{T^*} < 4$), for all α, K. For $\alpha \geq \frac{K-1-\Gamma}{(K-1)(1-\gamma)}$, T is optimal.

Proof. The scheme that achieves the above performance is presented in Section 4.3, while the corresponding gap to optimal is bounded in Section 4.4. ■

Furthermore directly from the above, for $\alpha = 0$, we have the following.

Corollary 4.1 *In the MISO BC with $\Gamma \leq 1, \alpha = 0$, then*

$$T = H_K - \Gamma \tag{4.3}$$

is achievable and has a gap from optimal that is less than 4.

Directly from Theorem 4.1, we have the following corollary which quantifies the CSIT savings

$$\delta(\gamma,\alpha) := \arg\min_{\alpha'}\{\alpha' : (1-\gamma)T^*(\gamma=0,\alpha') \leq T(\gamma,\alpha)\} - \alpha$$

that we can have as a result of properly exploiting small caches. This reflects the CSIT reductions (from $\alpha+\delta(\gamma,\alpha)$ to the operational α) that can be achieved due to coded caching, without loss in performance.

Corollary 4.2 *In the (K,M,N,α) cache-aided BC with $\Gamma \leq 1$, then*

$$\delta(\gamma,\alpha) = \begin{cases} \frac{\gamma(K-H_K)}{H_K-K\gamma}(\alpha+\frac{1}{H_K-1}), & 0 \leq \alpha < \alpha_{b,1} \\ \frac{(1-\alpha)(KH_\eta-\eta H_K)}{KH_{\eta+1}(H_K-1)}, & \alpha_{b,\eta} \leq \alpha < \alpha_{b,\eta+1} \\ 1-\alpha, & \alpha \geq \frac{K(1-\gamma)-1}{(K-1)(1-\gamma)}. \end{cases} \tag{4.4}$$

The last case in the above equation shows how, in the presence of caching, we need not acquire CSIT quality that exceeds $\alpha = \frac{K(1-\gamma)-1}{(K-1)(1-\gamma)}$.

Tightening the bounds for the large BC with scalably small caches $(K \gg 1, \quad \gamma \ll 1)$ We now briefly touch upon the more general small-cache setting of $\gamma \ll 1$, where we have a large number of users $K \gg 1$. In this setting — which captures our case of $\Gamma \leq 1$, as well as the case where $\Gamma > 1$ but where still $\gamma \ll 1$ — we tighten the gap to optimal for the achievable performance, here (from Theorem 4.1), as well as for the $\Gamma \geq 1$ setting in [74] which stated that

$$T_{\Gamma\geq 1} := \frac{(1-\gamma)(H_K - H_\Gamma)}{\alpha(H_K-H_\Gamma)+(1-\alpha)(1-\gamma)}. \tag{4.5}$$

Theorem 4.2 *In the (K,M,N,α) cache-aided MISO BC, in the limit of large K and reduced cache size $M \ll N$, the achieved T from Theorem 4.1 (as well as $T_{\Gamma\geq 1}$), are at most a factor of 2 from optimal, for all values of α.*

Proof. The proof is found in Appendix 4.6.1. ■

The following shows (for the case of $\alpha = 0$) how, even a vanishing $\gamma = \frac{M}{N} \to 0$, can provide a non-vanishing gain. [1]

[1] To avoid confusion, we clarify that the main Theorem is simply a DoF-type result, where SNR scales to infinity, and where the derived DoF holds for all K. The corollary below is simply based on the original DoF expression (i.e., SNR diverges first), which is then approximated in the large K setting (K diverges second, simultaneously with γ).

Corollary 4.3 *In the $(K, M, N, \alpha = 0)$ cache-aided MISO BC, as K scales to infinity and as γ scales as $\gamma = K^{-(1-\zeta)}$ for any $\zeta \in [0, 1)$, the gain from caching is*

$$\lim_{K\to\infty} \frac{T(\gamma = K^{-(1-\zeta)}, \alpha = 0)}{T^*(\gamma = 0, \alpha = 0)} = 1 - \zeta. \tag{4.6}$$

Proof. The expression follows directly from (4.5). ■

4.3 Cache-aided QMAT with very small caches

We now describe the communication scheme. Part of the challenge, and a notable difference from the case of larger caches, is that due to the fact that now $\Gamma < 1$, some of the library content must remain entirely uncached. This uncached part is delivered by employing a combination of multicasting and ZF which uses current CSIT. The problem though remains when α is small because then current CSIT can only support a weak ZF component, which in turn forces us to send some of this uncached private information using multicasting, which itself must be calibrated not to intervene with the multicasting that utilizes side information from the caches. For this range of smaller α, our scheme here will differ from that when α is big (as well as from the scheme for $\Gamma \geq 1$). When α is bigger than a certain threshold value $\alpha_{b,1}$, we will choose to cache even less data from the library, which though we will cache with higher redundancy[2]. Calibrating this redundancy as a function of α, allows us to strike the proper balance between ZF and delayed-CSIT aided coded caching. For this latter part, we will use our scheme from [74] which we do not describe here.

We consider the range[3] $\alpha \in [0, \alpha_{b,1}]$, and proceed to set $\eta = 1$ (cf. (4.1) from Theorem 4.1), such that there is no overlapping content in the caches ($Z_k \cap Z_i = \varnothing$).

4.3.1 Placement phase

During the placement phase, each of the N files $W_n, n = 1, 2, \ldots, N$ ($|W_n| = f$ bits) in the library, is split into two parts

$$W_n = (W_n^c, W_n^{\overline{c}}) \tag{4.7}$$

where W_n^c (c for 'cached') will be placed in different caches, while the content of $W_n^{\overline{c}}$ ($\overline{c}$ for 'non-cached') will never be cached anywhere, but will instead be communicated — using current and delayed CSIT — in a manner that avoids interference without depending on caches. The split is such that

$$|W_n^c| = \frac{KMf}{N} = K\gamma f \text{ bits}. \tag{4.8}$$

[2] Higher redundancy here implies that parts of files will be replicated in more caches.

[3] The remaining range of α will be briefly addressed at the end of this section.

Then, we equally divide W_n^c into K subfiles $\{W_{n,k}^c\}_{k\in[K]}$, where each subfile has size

$$|W_{n,k}^c| = \frac{Mf}{N} = \gamma f \text{ bits} \tag{4.9}$$

and the caches are filled as follows

$$Z_k = \{W_{n,k}^c\}_{n\in[N]} \tag{4.10}$$

such that each subfile $W_{n,k}^c$ is stored in Z_k.

4.3.2 Delivery phase

Upon notification of the requests $W_{R_k}, k = 1,\dots,K$, we first further split $W_{R_k,k}^{\overline{c}}$ into two parts, $W_{R_k,k}^{\overline{c},p}$ and $W_{R_k,k}^{\overline{c},\overline{p}}$ that will be delivered in two different ways that we describe later, and whose sizes are such that

$$|W_{R_k,k}^{\overline{c},p}| = \alpha f T, \quad |W_{R_k,k}^{\overline{c},\overline{p}}| = f(1 - K\gamma - \alpha T). \tag{4.11}$$

Then we fold all $W_{R_k,\psi\setminus\{k\}}^c$ to get a set

$$X_\psi := \oplus_{k\in\psi} W_{R_k,\psi\setminus\{k\}}^c, \psi \in \Psi_2 \tag{4.12}$$

of so-called *order-2 XORs* (each XOR is meant for two users), and where $\Psi_2 := \{\psi \in [K] \ : \ |\psi| = 2\}$. Each of these XORs has size

$$|X_\psi| = \gamma f \text{ bits} \tag{4.13}$$

and they jointly form the XOR set

$$\mathcal{X}_\Psi := \{X_\psi = \oplus_{k\in\psi} W_{R_k,\psi\setminus\{k\}}^c\}_{\psi\in\Psi_2} \tag{4.14}$$

of cardinality $|\mathcal{X}_\Psi| = \binom{K}{2}$.

In the end, we must deliver

- $W_{R_k}^{\overline{c},p},\ k = 1,\cdots,K$, privately to user k, using mainly current CSIT
- $W_{R_k}^{\overline{c},\overline{p}},\ k = 1,\cdots,K$, using mainly delayed CSIT
- $\{W_{R_k,\psi\setminus\{k\}}^c\}_{\psi\in\Psi_2},\ k = 1,\cdots,K$ by delivering the XORs from $\mathcal{X}_\Psi$, each to their intended pair of receivers.

This delivery is described in the following.

Transmission We describe how we adapt the QMAT algorithm from [67] to deliver the aforementioned messages, with delay T.

While we will not go into all the details of the QMAT scheme, we note that some aspects of this scheme are similar to MAT schemes (cf. [1]), and a main new element is that QMAT applies digital transmission of interference, and a double-quantization method that collects and distributes residual interference across different rounds, in a manner that allows for ZF and MAT to coexist at maximal rates. The main ingredients include MAT-type symbols of different degrees of multicasting, ZF-type symbols for each user, and auxiliary symbols that diffuse interference across different phases and rounds. Many of the details of this scheme are 'hidden' behind the choice of $\mathbf{G}_{c,t}$ and behind the loading of the MAT-type symbols and additional auxiliary symbols that are all represented by $\boldsymbol{x}_{c,t}$ below. Another important element involves the use of caches to 'skip' MAT phases, as well as a careful rate- and power-allocation policy.

The QMAT algorithm has K transmission phases. For each phase $i = 1, \cdots, K$, the QMAT data symbols are intended for a subset $\mathcal{S} \subset [K]$ of users, where $|\mathcal{S}| = i$. Here by adapting the algorithm, at each instance $t \in [0, T]$ through the transmission, the transmitted vector takes the form

$$\boldsymbol{x}_t = \mathbf{G}_{c,t}\boldsymbol{x}_{c,t} + \sum_{\ell \in \bar{\mathcal{S}}} \boldsymbol{g}_{\ell,t} a^*_{\ell,t} + \sum_{k=1}^{K} \boldsymbol{g}_{k,t} a_{k,t} \tag{4.15}$$

with $\boldsymbol{x}_{c,t}$ being a K-length vector for QMAT data symbols, with $a^*_{\ell,t}$ being an auxiliary symbol that carries residual interference, where $\bar{\mathcal{S}}$ is a set of 'undesired' users that changes every phase, and where each unit-norm precoder $\boldsymbol{g}_{k,t}$ for user $k = 1, 2, \ldots, K$, is simultaneously orthogonal to the CSI estimate for the channels of all other users ($\boldsymbol{g}_{l,t}$ acts the same), thus guaranteeing

$$\hat{\boldsymbol{h}}^T_{k',t} \boldsymbol{g}_{k,t} = 0, \quad \forall k' \in [K] \backslash k. \tag{4.16}$$

Each precoder $\mathbf{G}_{c,t}$ is defined as $\mathbf{G}_{c,t} = [\boldsymbol{g}_{c,t}, \mathbf{U}_{c,t}]$, where $\boldsymbol{g}_{c,t}$ is simultaneously orthogonal to the channel estimates of the undesired receivers, and $\mathbf{U}_{c,t} \in \mathbb{C}^{K \times (K-1)}$ is a randomly chosen, isotropically distributed unitary matrix.

The rates and the power are set by the QMAT algorithm, such that:

- each $\boldsymbol{x}_{c,t}$ carries $f(1-\alpha)\mathrm{dur}(\boldsymbol{x}_{c,t})$ bits,
- each $a^*_{\ell,t}$ carries $\min\{f(1-\alpha), f\alpha\}\mathrm{dur}(\boldsymbol{g}_{\ell,t}a^*_{\ell,t})$ bits,
- each $a_{k,t}$ carries $f\alpha \mathrm{dur}(\boldsymbol{g}_{k,t}a_{k,t})$ bits,
- and

$$\mathbb{E}\{|\boldsymbol{x}_{c,t}|_1^2\} = \mathbb{E}\{|a^*_{\ell,t}|^2\} \doteq P$$
$$\mathbb{E}\{|\boldsymbol{x}_{c,t}|_i^2\} \doteq P^{1-\alpha}, \mathbb{E}\{|a_{k,t}|^2\} \doteq P^{\alpha}$$

where $|\boldsymbol{x}_{c,t}|_i, i = 1, 2, \cdots, K$, denotes the magnitude of the i^{th} entry of vector $\boldsymbol{x}_{c,t}$.

The scheme here employs a total of $2K-1$ phases (rather than the K phases in the original Q-MAT), where during the first $K-1$ phases (labeled here as phases $j = 1, \ldots, K-1$), the vector $\boldsymbol{x}_{c,t}$ carries the folded messages $X_\psi \in \mathcal{X}_\Psi$ using the last $K-1$ phases of the MAT algorithm from [1], while for phases $j = K, \ldots, 2K-1$, $\boldsymbol{x}_{c,t}$ now carries $\{W_{R_k}^{\overline{c},\overline{p}}\}_{k\in[K]}$ using the entirety of MAT. In addition, for all $2K-1$ phases, the different $a_{k,t}$ will carry (via ZF) all of the uncached $W_{R_k}^{\overline{c},p}, k = 1, \ldots, K$. The power and rate allocation guarantee that these MAT and ZF components can be carried out in parallel with the assistance of the auxiliary symbols from the next round[4].

In the following, we use T_j to denote the duration of phase j, and $T^{(1)} := \sum_{j=1}^{K-1} T_j$ to denote the duration of the first $K-1$ phases.

Summary of transmission scheme for delivery of $\{X_\psi\}_{\psi\in\Psi_2}$ Here, $\boldsymbol{x}_{c,t}$, $t \in [0, T^{(1)}]$ will have the structure defined by the last $K-1$ phases of (one round of) the QMAT algorithm.

During the first phase ($t \in [0, T_1]$, corresponding to phase 2 of QMAT, where $|\mathcal{S}| = 2$), $\boldsymbol{x}_{c,t}$ will convey all the order-2 messages in $\{X_\psi\}_{\psi\in\Psi_2}$ (each ψ corresponds to each $\mathcal{S}$). Then, at the end of this phase, for each $\psi \in \Psi_2$, and for each $k \in \psi$, the received signal at user k, takes the form

$$y_{k,t} = \underbrace{\boldsymbol{h}_{k,t}^T \mathbf{G}_{c,t} \boldsymbol{x}_{c,t}}_{\text{power } \doteq P} + \underbrace{\boldsymbol{h}_{k,t}^T \sum_{\ell\in\psi} \boldsymbol{g}_{\ell,t} a_{\ell,t}}_{\doteq P^{1-\alpha}} + \underbrace{\boldsymbol{h}_{k,t}^T \boldsymbol{g}_{k,t} a_{k,t}}_{P^\alpha} \tag{4.17}$$

while the received signal for the other users $k \in [K]\backslash\psi$ takes the form

$$y_{k,t} = \underbrace{\boldsymbol{h}_{k,t}^T \boldsymbol{g}_{k,t} a_{k,t}^*}_{\text{power } \doteq P} + \underbrace{\boldsymbol{h}_{k,t}^T \Big(\sum_{\substack{\ell\in\psi \\ \ell\neq k}} \boldsymbol{g}_{\ell,t} a_{\ell,t}^* + \mathbf{G}_{c,t}\boldsymbol{x}_{c,t}\Big)}_{L_{\psi,k'},\ \doteq P^{1-\alpha}} + \underbrace{\boldsymbol{h}_{k,t}^T \boldsymbol{g}_{k,t} a_{k,t}}_{P^\alpha} \tag{4.18}$$

where in both cases, we ignored the Gaussian noise and the ZF noise up to P^0. Following basic MAT techniques, the interference $L_{\psi,k'}, \forall k'$ is translated into order-3 messages and will be sent in phase $j = 2$. In addition, to separate $\boldsymbol{h}_{k,t}^T \boldsymbol{g}_{k,t} a_{k,t}$ from the MAT component, as in [67], we use auxiliary data symbols $a_{\ell,t}^*$. Specifically, $L_{\psi,k'}$ is first quantized with $(1-2\alpha)^+ \log P$ bits, leaving the quantizaiton noise $n_{\psi,k'}$ with power scaling in P^α. Then, the transmitter quantizes this quantization noise $n_{\psi,k'}$ with $\alpha \log P$ bits up to the noise level, which will be carried by the auxiliary data symbols in the corresponding phase

[4]We here focus, for ease of description, on describing only one round. For more details on the multi-round structure of the QMAT, please see [67].

in the next round. In this way, $\boldsymbol{x}_{c,t}$ *can be decoded using the auxiliary data symbols of the next round, and using order-3 messages from the next phase.*

Given that the allocated 'rate' for $\boldsymbol{x}_{c,t}$ is $(1-\alpha)f$, and given that there is a total of $|\mathcal{X}_\Psi| = \binom{K}{2}$ different order-2 folded messages X_ψ ($|X_\psi| = \gamma f$ bits), the duration T_1 of the first phase, takes the form

$$T_1 = \frac{\binom{K}{2}|X_\psi|}{(K-1)(1-\alpha)f} = \frac{\gamma\binom{K}{2}}{(K-1)(1-\alpha)}. \tag{4.19}$$

For phases $j = 2, \cdots, K-1$ here (which draw from the last $K-2$ phases in [67]), we can similarly calculate their duration T_j to be $T_j = \frac{2}{j+1}T_1$, which in turn implies that

$$T^{(1)} = \sum_{j=1}^{K-1} T_j = T_1 \sum_{j=1}^{K-1} \frac{2}{1+j} = \frac{\Gamma(H_K - 1)}{1-\alpha}. \tag{4.20}$$

Transmission of $\{W_{R_k}^{\bar{c},\bar{p}}\}_{k\in[K]}$ Now the remaining information from $\{W_{R_k}^{\bar{c},\bar{p}}\}_{k\in[K]}$, will be conveyed by $\boldsymbol{x}_{c,t}, t \in [T^{(1)}, T]$ (phases $K, \ldots, 2K-1$), where now though we will use all the phases of the Q-MAT algorithm because now there is no corresponding side information in the caches to help us 'skip' phases. During the first phase of this second part (i.e., during phase $j = K$), we place all of $\{W_{R_k}^{\bar{c},\bar{p}}\}_{k\in[K]}$ in $\boldsymbol{x}_{c,t}, t \in [T^{(1)}, T^{(1)} + T_K]$. Given the allocated rate $(1-\alpha)f$ for $\boldsymbol{x}_{c,t}$, and given that $|\{W_{R_k}^{\bar{c},\bar{p}}\}_{k\in[K]}| = Kf(1 - \frac{KM}{N} - \alpha T)$, we see that

$$T_K = \frac{Kf(1 - \frac{KM}{N} - \alpha T)}{K(1-\alpha)f} = \frac{(1 - \Gamma - \alpha T)}{(1-\alpha)}. \tag{4.21}$$

Similarly we see that $T_j = \frac{1}{j-K+1}T_K,\ j = K, \ldots, 2K-1$, which means that

$$\begin{aligned} T - T^{(1)} &= \sum_{j=K}^{2K-1} T_j = T_K \sum_{j=K}^{2K-1} \frac{1}{j-K+1} \\ &= \frac{H_K(1 - \Gamma - \alpha T)}{(1-\alpha)}. \end{aligned} \tag{4.22}$$

Combining (4.20) and (4.22), gives the desired

$$T = \frac{H_K - \Gamma}{1 - \alpha + \alpha H_K}. \tag{4.23}$$

Communication scheme for $\alpha \in [\alpha_{b,1}, \alpha_{b,K-1}]$ Here, when $\alpha \geq \alpha_{b,1}$, the scheme already exists; we use

$$\eta = \arg \max_{\eta' \in [\Gamma, K-1] \cap \mathbb{Z}} \{\eta' \ :\ \alpha_{b,\eta'} \leq \alpha\} \tag{4.24}$$

where

$$\alpha_{b,\eta} = \frac{\eta - \Gamma}{\Gamma(H_K - H_\eta - 1) + \eta} \tag{4.25}$$

and directly from the algorithm designed for the case of $\Gamma \geq 1$ in [74], we get

$$T = \max\{1 - \gamma,\ \frac{(K-\Gamma)(H_K - H_\eta)}{(K-\eta) + \alpha(\eta + K(H_K - H_\eta - 1))}\}. \tag{4.26}$$

4.4 Bounding the gap to optimal

This section presents the proof that the gap $\frac{T(\gamma,\alpha)}{T^*(\gamma,\alpha)}$, between the achievable $T(\gamma,\alpha)$ and the optimal $T^*(\gamma,\alpha)$, is always upper bounded by 4, which also serves as the proof of identifying the optimal $T^*(\gamma,\alpha)$ within a factor of 4. The outer bound (lower bound) on the optimal T^*, is taken from Lemma 3.1 in Chapter 3, and it takes the form

$$T^*(\gamma,\alpha) \geq \max_{s\in\{1,\ldots,\lfloor\frac{N}{M}\rfloor\}} \frac{1}{(H_s\alpha + 1 - \alpha)}(H_s - \frac{Ms}{\lfloor\frac{N}{s}\rfloor}). \tag{4.27}$$

We proceed with the fist case of $\alpha = 0, \Gamma \leq 1$.

4.4.1 Gap for $\alpha = 0, \Gamma < 1$

Our aim here is to show that $\frac{T}{T^*} < 4$, where we use the above lower bound, and where we recall that the achievable T took the form

$$T = H_K - K\gamma.$$

We first see that

$$\frac{T}{T^*} \leq \frac{H_K - K\gamma}{\max\limits_{s\in\{1,\ldots,\lfloor\frac{N}{M}\rfloor\}} H_s - \frac{Ms}{\lfloor\frac{N}{s}\rfloor}} \leq \frac{H_K - K\gamma}{\max\limits_{s\in\{1,\ldots,\lfloor\frac{N}{M}\rfloor\}} H_s - \frac{\gamma s^2}{1-\frac{s-1}{N}}} \tag{4.28}$$

$$\leq \frac{H_K - K\gamma}{\max\limits_{s\in\{1,\ldots,\lfloor\frac{N}{M}\rfloor\}} H_s - \frac{\gamma s^2}{1-\frac{s-1}{K}}} \tag{4.29}$$

$$\leq \frac{H_K - K\gamma}{H_{s_c} - \frac{K\gamma s_c^2}{K-s_c+1}} =: f_o(\gamma, s_c) \tag{4.30}$$

where (4.28) holds because $\lfloor\frac{N}{s}\rfloor \leq \frac{N-(s-1)}{s}$, where (4.29) holds because $N \geq K$, and where the last step holds because $\gamma \leq \frac{1}{K}$ and because we choose $s_c = \lfloor\sqrt{K}\rfloor$. We proceed to split the proof in two parts: one for $K \geq 25$, and one for $2 \leq K \leq 25$.

Case 1 ($\alpha = 0, K \geq 25$)

Here we see that the derivative of $f_o(\gamma, s_c)$ takes the form

$$\frac{df_o(\gamma, s_c)}{d\gamma} = \frac{K}{A}\left(\frac{H_K s_c^2}{K - s_c + 1} - H_{s_c}\right) \tag{4.31}$$

$$\geq \frac{K \log K}{A}\left(\frac{(\sqrt{K}-1)^2}{K - \sqrt{K} + 2} - \frac{1}{2}\right) \tag{4.32}$$

$$= \frac{K \log(K)}{A}\left(\frac{1}{2} - \frac{\sqrt{K}+1}{K - \sqrt{K} + 2}\right) \tag{4.33}$$

$$\geq 0 \tag{4.34}$$

where A is easily seen to be positive, where the second step is because $\sqrt{K}-1 \leq s_c \leq \sqrt{K}$ and $H_K \geq \log(K)$, and where the last step is because $0 \leq \frac{\sqrt{K}+1}{K-\sqrt{K}+2} \leq \frac{1}{2}$. Hence

$$\max_{\gamma \in [0, \frac{1}{K}]} f_o(\gamma, s_c) = f_o(\gamma = \frac{1}{K}, s_c) = \frac{H_K - 1}{H_{s_c} - \frac{s_c^2}{K - s_c + 1}}. \tag{4.35}$$

Now it is easy to see that $\frac{s_c^2}{K - s_c - 1} \leq \frac{K}{K - \sqrt{K} + 1}$ since $s_c = \lfloor \sqrt{K} \rfloor \leq \sqrt{K}$. Now consider the function

$$f(K) := \frac{K}{K - \sqrt{K} + 1} - \frac{\log K}{4}$$

and let us calculate its derivative

$$\frac{df(K)}{dK} = \frac{1 - \frac{\sqrt{K}}{2}}{(K - \sqrt{K} + 1)^2} - \frac{1}{4K} < 0$$

which we see to be negative for any $K \geq 36$. This allows us to conclude that $\max_{K \in [25, \infty]} f(K) = f(25) = 0.3858$, and also that $\frac{s_c^2}{K - s_c - 1} \leq \frac{\log K}{4} + 0.3858$.

Now let us go back to (4.35), where using the above maximization, we can get

$$f_o(\gamma = \frac{1}{K}, s_c) \leq \frac{H_K - 1}{H_{s_c} - (\frac{\log K}{4} + 0.3858)}$$

$$\leq \frac{H_K - 1}{\frac{1}{2}\log K + \epsilon_\infty + \log\frac{4}{5} - (\frac{\log K}{4} + 0.3858)}$$

$$= \frac{\log K + \epsilon_{25} - 1}{\frac{1}{4}\log K + \epsilon_\infty + \log\frac{4}{5} - 0.3858} \tag{4.36}$$

$$\leq 4 \tag{4.37}$$

where the second step is because $H_{s_c} \geq \log s_c + \epsilon_\infty \geq \log(\sqrt{K} - 1) + \epsilon_\infty \geq \log(\frac{5}{6}\sqrt{K}) + \epsilon_\infty = \frac{1}{2}\log K + \epsilon_\infty + \log\frac{5}{6}$, and where (4.36) holds because $H_K \leq \log K + \epsilon_{25}$ since $K \geq 25$.

Case 2 ($\alpha = 0, K = 2, \dots, 24$)

This is an easy step, and it follows after choosing $s = 1$ in the outer bound, which gives

$$\frac{T}{T^*} \leq \frac{H_K - K\gamma}{1-\gamma} \leq \frac{KH_K}{K-1} \leq 4 \tag{4.38}$$

because $\gamma \leq \frac{1}{K}$ and $K \leq 24$.

This completes the whole proof for $\alpha = 0, \Gamma < 1$.

4.4.2 Gap for $\alpha > 0, \Gamma < 1$

To bound the gap between the achievable

$$T = \begin{cases} \frac{H_K - \Gamma}{1-\alpha+\alpha H_K}, & 0 \leq \alpha < \alpha_{b,1} \\ \frac{(K-\Gamma)(H_K - H_\eta)}{(K-\eta)+\alpha(\eta+K(H_K-H_\eta-1))}, & \alpha_{b,\eta} \leq \alpha < \alpha_{b,\eta+1} \\ 1-\gamma, & \frac{K-1-\Gamma}{(K-1)(1-\gamma)} \leq \alpha \leq 1 \end{cases} \tag{4.39}$$

from Theorem 4.1 (recall that $\eta = 1, \dots, K-2$), to the optimal T^* bounded in (4.27), we will use the fact that

$$\frac{(H_K - \Gamma)}{\max\limits_{s \in \{1,\dots,\lfloor \frac{N}{M} \rfloor\}} H_s - \frac{sM}{\lfloor \frac{N}{s} \rfloor}} < 4, \forall N \geq K \geq 2, \forall \Gamma < 1. \tag{4.40}$$

We will split our proof in two main cases: one for $\alpha \in [0, \alpha_{b,1})$, and another for $\alpha \in [\alpha_{b,1}, 1]$ (recall from (4.1) that $\alpha_{b,\eta} = \frac{\eta - \Gamma}{\Gamma(H_K - H_\eta - 1) + \eta}$).

Case 1 ($\alpha \in [0, \alpha_{b,1}]$)

Directly from (4.39), let us use $T' := \frac{H_K - \Gamma}{1-\alpha+\alpha H_K}$ to denote T when $\alpha \in [0, \alpha_{b,1}]$. Now the gap is simply bounded as

$$\begin{aligned} \frac{T'}{T^*} &\leq \frac{H_K - \Gamma}{\max\limits_{s \in \{1,\dots,\lfloor \frac{N}{M} \rfloor\}} \frac{1}{H_s \alpha + 1 - \alpha}(H_s - \frac{Ms}{\lfloor \frac{N}{s} \rfloor})(1 - \alpha + \alpha H_K)} \\ &= \frac{(H_K - \Gamma)(1 - \alpha + \alpha H_s)}{\max\limits_{s \in \{1,\dots,\lfloor \frac{N}{M} \rfloor\}} (H_s - \frac{sM}{\lfloor \frac{N}{s} \rfloor})(1 - \alpha + \alpha H_K)} < 4 \end{aligned}$$

after observing that $s \leq K$ and after applying (4.40).

Case 2 ($\alpha \in [\alpha_{b,1}, 1]$)

Let us use

$$T'^{,\eta} := \frac{(K - \Gamma)(H_K - H_\eta)}{(K - \eta) + \alpha(\eta + K(H_K - H_\eta - 1))} \tag{4.41}$$

to denote $T(\gamma, \alpha)$ in (4.39) when $\alpha \in [\alpha_{b,\eta}, \alpha_{b,\eta+1}]$, $\eta = 1, \ldots, K-2$. For the rest of the proof, we will use the following lemma.

Lemma 4.1 *$T'^{,\eta}$ is decreasing with η, while $\alpha_{b,\eta}$ is increasing with η.*

Proof. See Appendix 4.6.2. ■

Given that $T'^{,\eta}$ decreases in η, we will just prove that $\frac{T'^{,1}}{T^*} < 4$, which will automatically guarantee $\frac{T'^{,\eta}}{T^*} < 4$ for all η and thus for all values of α.

Subcase 2-a ($K \geq 25$): From (4.40), we see that

$$\begin{aligned}\frac{T'^{,1}}{T^*} &\leq \frac{(K-\Gamma)(H_K-1)(1-\alpha+\alpha H_s)}{\max\limits_{s\in\{1,\ldots,K\}}(H_s - \frac{sM}{\lfloor \frac{N}{s} \rfloor})(K-1+\alpha(1+K(H_K-2)))} \\ &\leq \frac{4(K-\Gamma)(H_K-1)(1-\alpha+\alpha H_s)}{(K-1+\alpha(1+K(H_K-2)))(H_K-\Gamma)} =: \frac{4A_1}{B_1}\end{aligned}$$

where we use $A_1 := (K-\Gamma)(H_K-1)(1-\alpha+\alpha H_s)$, and where we use $B_1 := (K-1+\alpha(1+K(H_K-2)))(H_K-\Gamma)$ to denote the denominator of the last expression. To upper bound the gap by 4, we will simply show that $A_1 < B_1$. Towards this we will first show that

$$\begin{aligned}A_1 - B_1 = \alpha(1-H_K)(K(H_K-H_s-\Gamma)+\Gamma H_s) \\ +(\alpha-1)(K-H_K)(1-\Gamma)\end{aligned} \tag{4.42}$$

is negative. To see this, we easily note that $(\alpha-1)(K-H_K)(1-\Gamma) \leq 0$. To guarantee that the first term above is also negative when $\Gamma \leq 1$, we just need to show (for the same s that we chose before given the same parameters, but when α was zero) that

$$K(H_K - H_s - \Gamma) + \Gamma H_s = K(H_K - H_s - \Gamma + \gamma H_s) \geq 0$$

which is easy to see because $H_K - H_s - \Gamma + \gamma H_s = H_K - H_s + \gamma(H_s - K) \geq H_K - K\gamma \geq 0$ since $H_s - K \leq 0$ and $\gamma \in [0,1]$. This completes this part of the proof.

Subcase 2-b ($K \leq 24$): Here we choose $s = 1$ in the outer bound, and we directly have

$$\frac{T'^{,1}}{T^*} \leq \frac{H_K - K\gamma}{1-\gamma} \leq \frac{KH_K}{K-1} \leq 4 \tag{4.43}$$

because $\gamma \leq \frac{1}{K}$ and $K \leq 24$.

This completes the whole proof for $\alpha > 0, \Gamma \leq 1$.

4.5 Conclusions

Motivated by realistic expectations that cache size — at wireless receivers/end-users — will be small ([14]), and motivated by the well known limitations in getting good-quality and timely feedback, the work combines these scarce and highly connected resources, to conclude that we can attribute non-trivial performance gains even if the caches are with vanishingly small $\gamma \to 0$. This synergy between feedback and caching, allows for a serious consideration of scenarios where even microscopic fractions of the library can be placed at different caches across the network, better facilitating the coexistence of modestly-sized caches and large libraries.

4.6 Appendix

4.6.1 Proof of Corollary 4.3

Our aim here is to show that for large K, and when $\gamma \ll 1$, the achievable T (both from Theorem 4.1 corresponding to the case of $\Gamma < 1$, but also from (4.5)), has a gap to optimal that is at most 2, for all α. We first consider the scenario where $\alpha = 0$, and note that

$$\begin{aligned} T(\Gamma \geq 1, \alpha = 0) &= H_K - H_{K\gamma} \leq \log(K) + \epsilon_2 - \log(K\gamma) \\ &\leq \log(\frac{1}{\gamma}) + \epsilon_2 \\ T(\Gamma < 1, \alpha = 0) &= H_K - K\gamma \leq \log(K) + \epsilon_2 \\ &\leq \log(\frac{1}{\gamma}) + \epsilon_2 \end{aligned}$$

and thus note that in both cases, we have that

$$T \leq \log(\frac{1}{\gamma}) + \epsilon_2, \forall \Gamma \geq 0$$

which means that

$$\frac{T}{T^*} \leq \frac{\log(\frac{1}{\gamma}) + \epsilon_2}{\max\limits_{s \in \{1,\ldots,\lfloor \frac{N}{M} \rfloor\}} H_s - \frac{sM}{\lfloor \frac{N}{s} \rfloor}} \leq \frac{\log(\frac{1}{\gamma}) + \epsilon_2}{H_{s_c} - \frac{Ms_c}{\lfloor \frac{N}{s_c} \rfloor}} \tag{4.44}$$

for any $s_c \in \{1, \ldots, \lfloor \frac{N}{M} \rfloor\}$. Now let us choose $s_c = \lfloor \sqrt{\frac{1}{\gamma}} \rfloor$, and note that $N \geq Ms_c^2 \gg s_c$, which means that $\frac{\lfloor \frac{N}{s_c} \rfloor}{\frac{N}{s_c}} \to 1$. Consequently, from (4.44), for both cases, we have

$$\begin{aligned} \lim_{K\to\infty} \frac{T}{T^*} &\leq \lim_{K\to\infty} \frac{\log(\frac{1}{\gamma}) + \epsilon_2}{\log(s_c) - \gamma s_c^2} \\ &= \lim_{K\to\infty} \frac{2\log(s_c) + \epsilon_2}{\log(s_c) - 1} = 2 \end{aligned} \tag{4.45}$$

proving the tighter gap to optimal, which is at most 2, for the case of $\alpha = 0$.

The additional case for $\alpha > 0, \Gamma < 1$ is handled in the extended version in [75], while the case of $\alpha > 0, \Gamma \geq 1$ (but still with $\gamma \ll 1$) can be found in [74].

4.6.2 Proof of Lemma 4.1

We need to show that $\alpha_{b,\eta}$ is increasing in η, and that $T'^{,\eta}$ is decreasing in η. The first follows by noting that

$$\begin{aligned}\alpha_{b,\eta+1} - \alpha_{b,\eta} &= \frac{(H_K - H_\eta)}{(H_K - H_\eta - 1) + \frac{\eta}{\Gamma}} - \frac{(H_K - H_{\eta+1})}{(H_K - H_{\eta+1} - 1) + \frac{\eta+1}{\Gamma}} \\ &= \frac{H_K - H_\eta + \frac{\eta - \Gamma}{\eta+1}}{((H_K - H_\eta - 1) + \frac{\eta}{\Gamma})((H_K - H_{\eta+1} - 1) + \frac{\eta+1}{\Gamma})} > 0 \end{aligned} \tag{4.46}$$

which holds because $\eta \geq \Gamma$.

To see that $T'^{,\eta}$ decreases in η, after simplifying notation by letting

$$\begin{aligned} D_\eta &:= (K - \eta) + \alpha(\eta + K(H_K - H_\eta - 1)) \\ D_{\eta+1} &= (K - \eta - 1) + \alpha(\eta + 1 + K(H_K - H_{\eta+1} - 1)) \end{aligned}$$

to denote the denominators of $T'^{,\eta}$ and of $T'^{,\eta+1}$ respectively, we see that

$$\begin{aligned} \frac{T'^{,\eta} - T'^{,\eta+1}}{K - \Gamma} &= \frac{H_K - H_\eta}{D_\eta} - \frac{H_K - H_{\eta+1}}{D_{\eta+1}} \\ &= \frac{(\frac{K-\eta}{\eta+1} + H_\eta - H_K)(1 - \alpha)}{D_\eta D_{\eta+1}} \\ &= \frac{(\frac{K-\eta}{\eta+1} - (\frac{1}{\eta+1} + \frac{1}{\eta+2} + \cdots + \frac{1}{K}))(1 - \alpha)}{D_\eta D_{\eta+1}} > 0 \end{aligned} \tag{4.47}$$

which holds because $\eta \leq K$ and $\frac{1}{\eta+1} \geq \frac{1}{\eta+i}, \forall i \in [1, K - \eta] \cap \mathbb{Z}$. The above inequality is strict when $\eta > \Gamma$. This completes the proof.

Chapter 5

Coded caching for reducing CSIT-feedback in wireless communications

In the same K-user symmetric MISO BC setting as before, the work explores the role of caching content at receiving users for the purpose of reducing the need for current feedback. Emphasis is entirely on current CSIT quality, and this chapter does not consider delayed CSIT. In this setting, we show how caching, when combined with a rate-splitting broadcast approach, can not only improve performance, but can also reduce the need for CSIT, in the sense that the identified cache-aided optimal DoF associated to caching and perfect CSIT, can in fact be achieved with reduced-quality CSIT. These CSIT savings can be traced back to an inherent relationship between caching, performance, and CSIT; caching improves performance by leveraging multi-casting of common information, which automatically reduces the need for CSIT, by virtue of the fact that common information is not a cause of interference. At the same time though, too much multicasting of common information can be detrimental, as it does not utilize existing CSIT. Our caching method builds on the Maddah-Ali and Niesen coded caching scheme, by properly balancing multicast and broadcast opportunities, and by combing caching with rate-splitting communication schemes that are specifically designed to operate under imperfect-quality CSIT. The observed achievable CSIT savings here, are more pronounced for smaller values of K users and N files.

5.1 Introduction

5.1.1 Cache-aided K-user broadcast channel

We continue to explore the same wireless communication setting where a K-antenna transmitter communicates information to K single-antenna receiving users (see Fig 2.1). The entire information content at the transmitter consists of N distinct files $W_1, W_2, \ldots, W_N$, each of size f bits. Each user $k = 1, 2, \ldots, K$ has a cache Z_k of size Mf bits, where $M < N$. After each transmission, the corresponding received signals at receiver k, can be modeled as

$$y_k = \boldsymbol{h}_k^T \boldsymbol{x} + z_k, \quad k = 1, \ldots, K \tag{5.1}$$

where $\boldsymbol{x} \in \mathbb{C}^{K\times 1}$ denotes the transmitted vector which satisfies a power constraint $\mathbb{E}(|\boldsymbol{x}|^2) \leq P$, where $\boldsymbol{h}_k \in \mathbb{C}^{K\times 1}$ denotes the vector fading coefficients, and where z_k represents unit power AWGN noise at user k.

We first proceed with motivating examples that provide insight on different elements of the problem.

5.1.2 Motivating examples

Example - ($M = 0$)

Let us first consider the no-caching case where $M = 0$, which — again without loss of generality — can be taken to correspond to a delivery phase that is strictly of a broadcast nature. In this broadcast channel, the sum DoF with perfect CSIT, is equal to K, which means that the K requested files[1] will take 1 time slot to deliver, corresponding to an optimal $T^*(M = 0, \alpha = 1) = T^*(0) = 1$. Applying the new result by Davoodi and Jafar [63], which states that the maximum DoF of K can only be achieved in the presence of $\alpha = 1$, immediately reveals that to achieve the optimal $T^*(0) = 1$, we need $\alpha = \alpha_{th} = 1$.

Example - ($N = K = 2$)

Let us now offer a more involved example, which reveals an interesting achievable tradeoff between T, M and α. Specifically let us consider the case where $N = K = 2$, and let $0 \leq M \leq 1$. There are two files, which we relabel as $W_1 = A, W_2 = B$, each of size $f = \log P$ bits.

In this example, for the *placement phase*, we first split both files A and B into three subfiles, i.e, $A = (A_1, A_2, A_3), B = (B_1, B_2, B_3)$, where the subfiles $A_i, B_i, i = 1, 2$ are each of size $\frac{Mf}{2}$ bits, and where A_3 and B_3 are each of size $(1 - M)f$ bits. We fill up the caches as follows $Z_1 = (A_1, B_1)$ and $Z_2 =$

[1] Recall that the request considered over the delivery phase, is one where each user requests a different file.

(A_2, B_2), so that each user has in their cache, an equal part of clean information for each file.

For the *delivery phase* — and again focusing on the request $W_{F_1} = W_1 = A, W_{F_2} = W_2 = B$ — we see that to complete the task, user 1 needs subfile A_2 (which is available in the cache of user 2) as well as A_3, and user 2 needs subfile B_1 (available at the cache of user 1), as well as B_3. As a result, $A_2 \oplus B_1$ (containing $\frac{Mf}{2}$ bits) has information that can be useful to both users, while A_3 and B_3 has private information for user 1 and 2 respectively. The challenge will be to communicate this information, as efficiently as possible, over a channel with imperfect feedback, corresponding to some $\alpha < 1$. Towards this, let us consider a scheme that sends a single transmission of the form

$$\boldsymbol{x} = \boldsymbol{w}c + \hat{\boldsymbol{h}}_2^{\perp} a_1 + \hat{\boldsymbol{h}}_1^{\perp} a_2 \tag{5.2}$$

where $\boldsymbol{x} \in \mathbb{C}^{2\times 1}$, where $\boldsymbol{w} \in \mathbb{C}^{2\times 1}$ is a randomly chosen precoder, and where $\hat{\boldsymbol{h}}_k^{\perp} \in \mathbb{C}^{2\times 1}$ is a precoder that is orthogonal to the estimate $\hat{\boldsymbol{h}}_k$ for $\boldsymbol{h}_k$. Additionally, the above symbols c, a_1, a_2 are respectively allocated power as follows[2] given by

$$P^{(c)} \doteq P, \; P^{(a_1)} \doteq P^{(a_2)} \doteq P^{\alpha}$$

and are allocated rate as follows

$$r^{(c)} = (1-\alpha)f, \quad r^{(a_1)} = r^{(a_2)} = \alpha f.$$

In particular, a_1 is loaded with $\alpha \log P$ bits from A_3, and a_2 is loaded with $\alpha \log P$ bits from B_3, while c is loaded with the $\frac{Mf}{2}$ bits of $A_2 \oplus B_1$, as well as with the $\max\{2((1-M)f - T\alpha f), 0\}$ bits of A_3, B_3 that did not fit inside a_1 and a_2. The precoders, power-allocation and rate-allocation are known to all nodes. As a result the received signals at the two users, take the form

$$\begin{aligned} y_1 &= \underbrace{\boldsymbol{h}_1^T \boldsymbol{w} c}_{P} + \underbrace{\boldsymbol{h}_1^T \hat{\boldsymbol{h}}_2^{\perp} a_1}_{P^{\alpha}} + \underbrace{\boldsymbol{h}_1^T \hat{\boldsymbol{h}}_1^{\perp} a_2}_{P^0} + \underbrace{z_1}_{P^0} \\ y_2 &= \underbrace{\boldsymbol{h}_2^T \boldsymbol{w} c}_{P} + \underbrace{\boldsymbol{h}_2^T \hat{\boldsymbol{h}}_2^{\perp} a_1}_{P^0} + \underbrace{\boldsymbol{h}_2^T \hat{\boldsymbol{h}}_1^{\perp} a_2}_{P^{\alpha}} + \underbrace{z_2}_{P^0} \end{aligned} \tag{5.3}$$

where

$$\mathbb{E}|\boldsymbol{h}_1^T \hat{\boldsymbol{h}}_1^{\perp} a_2|^2 = \mathbb{E}|(\hat{\boldsymbol{h}}_1^T + \tilde{\boldsymbol{h}}_1^T)\hat{\boldsymbol{h}}_1^{\perp} a_2|^2 = \mathbb{E}|\tilde{\boldsymbol{h}}_1^T \hat{\boldsymbol{h}}_1^{\perp} a_2|^2 \doteq P^0$$

$$\mathbb{E}|\boldsymbol{h}_2^T \hat{\boldsymbol{h}}_2^{\perp} a_1|^2 = \mathbb{E}|(\hat{\boldsymbol{h}}_2^T + \tilde{\boldsymbol{h}}_2^T)\hat{\boldsymbol{h}}_2^{\perp} a_1|^2 = \mathbb{E}|\tilde{\boldsymbol{h}}_2^T \hat{\boldsymbol{h}}_2^{\perp} a_1|^2 \doteq P^0.$$

[2]We here use $\doteq$ to denote *exponential equality*, i.e., we write $f(P) \doteq P^B$ to denote $\lim_{P\to\infty} \frac{\log f(P)}{\log P} = B$.

At this point, user 1 can decode common symbol c by treating all other signals as noise. Consequently, user 1 removes $\boldsymbol{h}_1^T \boldsymbol{w} c$ from y_1 and decodes private symbol a_1. From c, user 1 can recover $A_2 \oplus B_1$, which combined with Z_1 allows for recovery of A_2. Finally from c and a_1, user 1 can recover A_3. Given that A_1 is already available in its cache, user 1 can thus reconstruct A. User 2 similarly obtains B.

To calculate the achieved T for this example, we note that the total of $\frac{Mf}{2} + (1-M)f + (1-M)f = (\frac{4-3M}{2}) \log P$ bits in $A_2 \oplus B_1, A_3, B_3$, was sent at an achievable rate (provided by this specific rate-splitting scheme) of $(1-\alpha+\alpha+\alpha)f = (1+\alpha)\log P$ bits per time slot. Hence the corresponding achievable duration is

$$T(M,\alpha) = \frac{4-3M}{2(1+\alpha)}. \tag{5.4}$$

As we will show later using basic cut-set bound arguments, the optimal $T^*(M)$ — associated to perfect CSIT — takes the form $T^*(M) = 1 - \frac{M}{N} = 1 - \frac{M}{2}$. Hence equating

$$T(M,\alpha) = \frac{4-3M}{2(1+\alpha)} = T^*(M) = 1 - \frac{M}{2}$$

and solving for α, gives that any α bigger than

$$\alpha_{th} = 1 - \frac{M}{2-M}, \quad 0 \leq M \leq 1 \tag{5.5}$$

suffices to achieve the optimal $T^*(M, \alpha = 1) = 1 - \frac{M}{2}$. A few simple observations include the fact that, as expected, α_{th} reduces with M (Fig. 5.1), as well as the fact that for $M = 0$, we have $\alpha_{th} = 1$, which correctly reflects the discussion in the previous example, which reminded us that in the broadcast channel, in the limit of high P, perfect CSIT is necessary for DoF optimality, i.e., perfect CSIT is necessary to transmit one file to each user within a time duration that is asymptotically optimal. On the other hand, we see that having $M \geq 1$, leads to $\alpha_{th} = 0$ because, as we have seen in [2], when $M \geq 1$ ($N = K = 2$), a simple transmission of a common message, suffices to achieve $T^*(M) = 1 - \frac{M}{2}$. Since the transmission is limited to a common symbol, it does not require CSIT.

Example - ($N = K = 2, M = 1/2$). Modifying coded caching for the BC

Let us now look at the interesting instance of $N = K = 2, M = 1/2$, which showcases some of the differences between the multicast case in [2] and the broadcast approach here, and which motivates caching specifically for the multi-antenna wireless setting, as compared to caching for the multicast case with a single shared medium with only serial multicast possibilities, that was

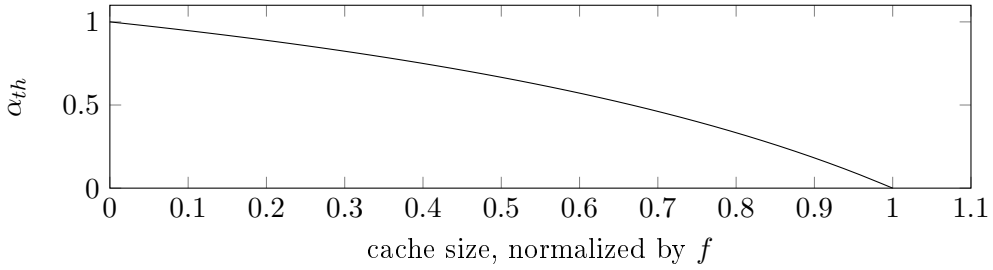

Figure 5.1: Required α_{tr} to achieve the optimal $T^*(M)$ in the $K = N = 2$ cache-aided MISO BC.

explored in [2]. Towards this, we recall that in [2], the *optimal* $T = 1$ was achieved by splitting files A and B into two halves, i.e., as $A = (A_1, A_2)$ and $B = (B_1, B_2)$, by setting $Z_k = A_k \oplus B_k, k = 1, 2$, and by sequentially transmitting[3] two common messages, B_1 and then A_2, to achieve the aforementioned optimal $T = 1$. What we point out here is that this caching would not work for the MISO-BC case (where the optimal T is $T^* = 1 - \frac{M}{2} = \frac{3}{4}$) because it leads to a delivery phase that only transmits common information, and thus does not leverage existing CSIT to improve performance.

We proceed with the description of the main results.

5.2 Main Results

We now describe the optimal $T^*(M)$ that is achievable with perfect CSIT ($\alpha = 1$), and then provide an achievability bound on $T(M, \alpha)$, and thus an achievability bound on the smallest α_{th} that achieves the optimal $T^*(M)$ above. We recall that the following results hold for $f = \log P$, in the limit of large P.

Lemma 5.1 *In the cache-aided K-user MISO BC, with N files of size f, and with caches of size Mf, the optimal $T^*(M, \alpha = 1)$ takes the form*

$$T^*(M) = 1 - \frac{M}{N}. \tag{5.6}$$

Proof. Let us first create the outer (lower) bound on T, using basic cut-set bound arguments in a manner that is similar to that in [2], [37]. Consider a simplified setting, where there are s users ($s \in \{1, \ldots, min\{\lfloor \frac{N}{M} \rfloor, K\}\}$). During the placement phase, the users' corresponding caches $Z_1, \ldots, Z_s$ are filled, while during the delivery phase, each of the s users makes $\lfloor \frac{N}{s} \rfloor$ sequential requests (one after the other), corresponding to a total of $s\lfloor \frac{N}{s} \rfloor$ requested

[3]For the worst-case request that is assumed without loss of generality.

files $W_1, \ldots, W_{s\lfloor \frac{N}{s} \rfloor}$ by all the users together. Note that for integer $\frac{N}{s}$, these requests span all N files. We now consider a total of $\lfloor \frac{N}{s} \rfloor$ sequential transmissions $X_1, \ldots, X_{\lfloor \frac{N}{s} \rfloor}$, such that X_1 and $Z_1, \ldots, Z_s$ can reconstruct $W_1, \ldots, W_s$, such that similarly X_2 and $Z_1, \ldots, Z_s$ can reconstruct $W_{s+1}, \ldots, W_{2s}$, and so on, until we have that $X_1, \ldots, X_{\lfloor \frac{N}{s} \rfloor}$ and $Z_1, \ldots, Z_s$ can reconstruct all the requested files $W_1, \ldots, W_{s\lfloor \frac{N}{s} \rfloor}$.

To apply the cut-set bound, we place the $\lfloor \frac{N}{s} \rfloor$ broadcasting signals $X_1, \ldots, X_{\lfloor \frac{N}{s} \rfloor}$, each of duration T, on one side of the cut, together with all the caches $Z_1, \ldots, Z_s$, and then on the other side of the cut, we place all the requests of s users for a total of $s\lfloor \frac{N}{s} \rfloor$ files, each of size f. Hence it follows that

$$\begin{aligned}
&\lfloor \frac{N}{s} \rfloor sT + sM \geq H(Z_1, \ldots, Z_s, X_1, \ldots, X_{\lfloor \frac{N}{s} \rfloor}) \\
&\geq H(Z_1, \ldots, Z_s, X_1, \ldots, X_{\lfloor \frac{N}{s} \rfloor} | W_1, \ldots, W_{s\lfloor \frac{N}{s} \rfloor}) \\
&\quad + s\lfloor \frac{N}{s} \rfloor (1 - \epsilon_f) \\
&\geq s\lfloor \frac{N}{s} \rfloor (1 - \epsilon_f) \qquad (5.7)
\end{aligned}$$

where we have used that the $K \times s$ interference-free MIMO channel provides s degrees of freedom (this is in the limit of $f \to \infty$), and where we have used Fano's inequality. In the same limit of $f \to \infty$, we have that $\epsilon_f \to 0$. Thus solving for T, and optimizing over all possible choices of s, we obtain

$$T \geq \max_{s \in \{1, \ldots, min\{\lfloor \frac{N}{M} \rfloor, K\}\}} (1 - \frac{M}{\lfloor \frac{N}{s} \rfloor}) \qquad (5.8)$$

which obviously gives that $T \geq 1 - \frac{M}{N}$.

To achieve this with perfect CSIT, is very easy. First let each user cache any $f\frac{M}{N}$ bits from each file, which leaves for $(1 - \frac{M}{N})f$ bits, per user, that must be delivered during the delivery phase. For the worst case where each user requests a different file, this corresponds to a broadcast transmission, which can be handled in $T = 1 - \frac{M}{N}$ time slots, in the presence of perfect CSIT. This completes the proof. ■

Having established the optimal $T^*(M, \alpha = 1) = 1 - \frac{M}{N}$, let us now establish an inner (achievability) bound on $T(M, \alpha)$, and translate that onto a bound on

$$\alpha_{th} = \arg\min\{\alpha : T(M, \alpha) = 1 - \frac{M}{N}\}.$$

The result will be presented for the simpler case where $K = N$.

Proposition 5.1 *In the cache-aided K-user MISO BC, with $N = K$ files of size f, and with caches of size Mf, an achievable $T(M, \alpha)$ takes the form*

$$T(M, \alpha) = \frac{K - \frac{M(1+K)}{K}}{1 + (K-1)\alpha} \qquad (5.9)$$

which implies that the optimal $T^* = 1 - \frac{M}{N}$ *can be achieved with an* α *that need not be bigger than*

$$\alpha_{th} = \frac{N - 1 - M}{\frac{M}{K} + (N - 1 - M)}.$$

Proof. The proof is presented in the following subsection, by presenting the caching and delivery scheme that achieves the above performance in the presence of imperfect CSIT. ■

5.2.1 Coded caching and delivery with imperfect CSIT

To design the caching, each of the N files $W_n,\ n = 1, 2, \ldots, N$ is first divided into two parts,

$$W_n = (W_n^c, W_n^p)$$

where the information in W_n^p is never cached. For $p = \frac{M}{N-1}$, W_n^c has size pf, and W_n^p has size $(1-p)f$. The main idea is to apply the caching method of [2], but to restrict this to the subfiles $\{W_n^c\}_{n=1}^N$, rather than applying it on the whole $\{W_n\}_{n=1}^N$. Towards this, let us first split each subfile W_n^c into N subfiles $W_{n,\tau}, \tau \in \Omega$, where $\Omega = \{\tau \subset [K],\ s.t.\ |\tau| = N - 1\}$, and where we have used the notation $[K] \triangleq \{1, 2, \ldots, K\}$. We note that the union of the above subfiles forms W_n^c, and that each subfile $W_{n,\tau}$ has size $\frac{Pf}{N}$. Based on the above, and following in the footsteps of [2], we form the caches as follows

$$Z_k \leftarrow W_{n,\tau}, \forall n = \{1, 2, \ldots, N\}, \forall \tau \in \Omega,\ \text{such that } k \in \tau.$$

It is easy to see that each cache Z_k has $\frac{Mf}{N}$ bits originating from any specific W_n^c.

For the *delivery phase*, the transmitter sends

$$\boldsymbol{x} = \boldsymbol{w}c + \boldsymbol{g}_1 a_1 + \cdots + \boldsymbol{g}_k a_k + \cdots + \boldsymbol{g}_K a_K \tag{5.10}$$

where each a_k carries information from W_k^p, i.e., information that has not been cached anywhere, while c carries all the $\frac{pf}{N}$ bits of

$$X_c = \oplus_{k=1}^K W_{F_k, [K]\setminus\{k\}}$$

which can be seen as common information that is simultaneously useful to all receivers. Additionally, c carries the extra private information that could not fit in each a_k. In the above, $\boldsymbol{g}_k,\ k = 1, 2, \ldots, K$ are precoders that are designed to be orthogonal to the channel estimates of all users other than k. Finally the power and rate allocation was given by

$$\begin{aligned} &P^{(c)} \doteq P,\ P^{(a_k)} \doteq P^{\alpha} \\ &r^{(c)} = (1-\alpha)f,\ r^{(a_k)} = \alpha f, \quad k = 1, \ldots, K. \end{aligned} \tag{5.11}$$

As a result, the received signals $y_k,\ k = 1, 2, \ldots, K$ take the form

$$y_k = \underbrace{\boldsymbol{h}_k^T \boldsymbol{w} c}_{P} + \underbrace{\boldsymbol{h}_k^T \boldsymbol{g}_k a_k}_{P^\alpha} + \underbrace{\sum_{i=1, i\neq k}^{K} \boldsymbol{h}_k^T \boldsymbol{g}_i a_i}_{P^0} + \underbrace{z_k}_{P^0} \tag{5.12}$$

and we can see that, due to the power allocation and CSIT quality, symbols a_k do not cause interference to unintended users; at least not above the noise level. At this point, user k can decode the common symbol c by treating all other signals as noise. Consequently, user k removes $\boldsymbol{h}_k^T \boldsymbol{w} c$ from y_k, and decodes its private symbol a_k. Then it can recover $W_{F_k,[K]\setminus\{k\}}$ from Z_k and c, and $W_{F_k}^p$ from c and a_k. Since it already has $W_{F_k,\tau}$, user 1 reconstructs W_{F_k}. The same approach resolves the requests of the other users.

As before, we see that we were able to communicate

$$(K - \frac{MK}{N-1} + \frac{M}{K(N-1)})f = (K - \frac{MK}{N-1} + \frac{M}{K(N-1)}) \log P$$

bits of information. With the achievable rate of the communication scheme scaling as $(1 + (K-1)\alpha) \log P + o(\log P)$ per channel use, it becomes clear that as f increases, the transmission duration becomes

$$\begin{aligned} T(M, \alpha) &= \frac{K - \frac{MK}{N-1} + \frac{M}{K(N-1)}}{1 + (K-1)\alpha} \\ &= \frac{K - \frac{M(1+K)}{K}}{1 + (K-1)\alpha}. \end{aligned} \tag{5.13}$$

Setting $T(M, \alpha) = T^*(M) = 1 - \frac{M}{N}$, and solving for α, provides the achievable CSIT threshold[4] α_{th}.

Example: We here present a final example to offer some clarity on the designed scheme. We do this for the case of $N = K = 3, M = 1$. As before, we rename the files $(W_1, W_2, W_3) =: (A, B, C)$, and since $\tau \in \{\{1,2\},\{1,3\},\{2,3\}\}$, we split these as $A = (A_{12}, A_{13}, A_{23}, A^p)$, $B = (B_{12}, B_{13}, B_{23}, B^p)$ and $C = (C_{12}, C_{13}, C_{23}, C^p)$, where A^p, B^p, C^p will be private information for user 1, 2 and 3 respectively. In the above, the subfiles A_τ, B_τ, C_τ are each of size $\frac{f}{6}$, and they appear in Z_k for any $k \in \tau$ (see Fig. 5.2).

For request A, B, C by user $1, 2, 3$ respectively, we can see that user 1 needs A_{23} which is available at Z_2, Z_3, user 2 needs B_{13} which is available at Z_1, Z_3, and user 3 needs C_{12} which is available at Z_1, Z_2. Hence considering the common information $A_{23} \oplus B_{13} \oplus C_{12}$, we see that upon decoding this common information, user 1 can automatically reconstruct A_{23} (by removing

[4] It is interesting to note that for $\alpha = \alpha_{th}$, all private information is stored in symbols $a_k,\ k = 1, 2, \ldots, K$.

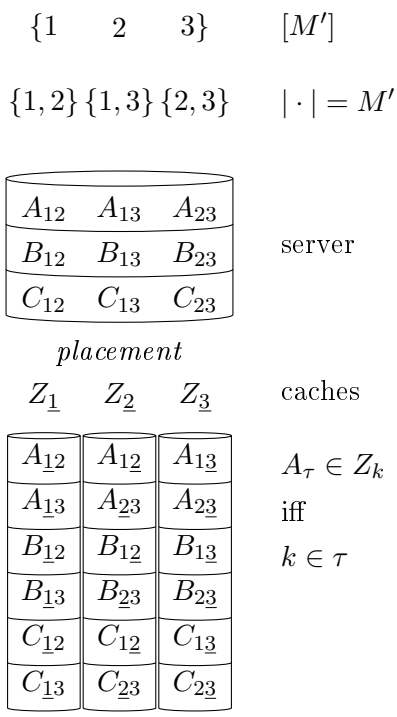

Figure 5.2: Placement of parts into user caches for $N = K = 3$, $M' \triangleq N-1 = 2$.

$B_{13} \oplus C_{12}$), and users 2 and 3 can act similarly to respectively reconstruct B_{13} and C_{12}.

During the delivery phase, the transmitter sends

$$\boldsymbol{x} = \boldsymbol{w}c + \boldsymbol{g}_1 a_1 + \boldsymbol{g}_2 a_2 + \boldsymbol{g}_3 a_3 \tag{5.14}$$

with power and rates set as

$$\begin{aligned} &P^{(c)} \doteq P, \ P^{(a_k)} \doteq P^{\alpha} \\ &r^{(c)} = 1-\alpha, \ r^{(a_k)} = \alpha, \quad k = 1,2,3 \end{aligned} \tag{5.15}$$

where $A_{23} \oplus B_{13} \oplus C_{12}$ is carried exclusively by c, while A^p, B^p, C^p are respectively placed in a_1, a_2, a_3, and any leftover information is placed in c.

The received signals y_k take the form

$$y_k = \underbrace{\boldsymbol{h}_k^T \boldsymbol{w} c}_{P} + \underbrace{\boldsymbol{h}_k^T \boldsymbol{g}_k a_k}_{P^{\alpha}} + \underbrace{\sum_{i=1, i\neq k}^{3} \boldsymbol{h}_k^T \boldsymbol{g}_i a_i}_{P^0} + \underbrace{z_k}_{P^0} \tag{5.16}$$

and as before, user 1 can decode c and a_1 to reconstruct all of A, and similarly for user 2 and 3 which reconstruct B and C respectively.

To calculate T, we note that there is a total of $3 - \frac{4M}{3}$ bits of information to be communicated (total information in $A_{23} \oplus B_{13} \oplus C_{12}$, A^p, B^p, C^p). Since the rate-splitting scheme has an achievable rate of $(1+2\alpha)\log P$ bits per time slot, we have that

$$T(M=1,\alpha) = \frac{3 - \frac{4M}{3}}{1+2\alpha}$$

which, when equated with $T^*(M, \alpha = 1) = 1 - \frac{M}{N} = \frac{2}{3}$, gives

$$\alpha_{th} = \frac{1}{\frac{1}{3}+1} = \frac{3}{4}.$$

It is worth comparing the above scheme which achieves $\alpha_{th} = \frac{3}{4}$, to a scheme that uses the caching method in [2], which — for these values of M, N — would not immediately allow for the possibility to have a common symbol that is simultaneously useful to everyone, and would thus not allow for the CSIT savings presented above. This is because in [2], the files are divided as $A = (A_1, A_2, A_3), B = (B_1, B_2, B_3), C = (C_1, C_2, C_3)$, forming caches $Z_k = (A_k, B_k, C_k),\ k = 1, \ldots, K$, which means that (again for delivery of different files $W_{F_1} = A, W_{F_2} = B, W_{F_3} = C$), $A \setminus Z_1 = (A_2, A_3), B \setminus Z_2 = (B_1, B_3), C \setminus Z_3 = (C_1, C_2)$, which in turn implies transmission of two-pair XORs $(A_2 \oplus B_1, A_3 \oplus C_1, B_3 \oplus C_2)$ (rather than triplet XORs in our case) which allows for the optimal $T^* = 3/4$, but only under the condition of perfect CSIT.

5.3 Conclusions

Motivated by recent advances in caching content at users (cf. [2,37,38]), which utilize *multicast gains* to increase throughput and reduce the network load, and motivated by sophisticated transmission schemes in multiuser settings that utilize precoder-enabled *broadcast gains* to increase the overall capacity of the system (sometimes in the presence of imperfect feedback [9,20,21,63]), we have here jointly treated these multicast and broadcast efforts, in a complementary manner that jointly compensated for each approach's limitations.

Particularly we focused on the effect of jointly treating caching and communication in broadcast-type communications, and explored the benefits of caching, not only in improving performance, but also in reducing the CSIT required to achieve this optimal performance.

Future work will include an effort to reduce the achievable bound $T(M, \alpha)$, as well as efforts to further reduce the CSIT-quality exponent α_{th} associated to the optimal performance.

Chapter 6

Wireless Coded Caching: A Topological Perspective

In this work, we move away from feedback and consider the aspect of topology. We explore the performance of coded caching in a SISO BC setting where some users have higher link capacities than others. Focusing on a binary and fixed topological model where strong links have a fixed normalized capacity 1, and where weak links have reduced normalized capacity $\tau < 1$, we identify — as a function of the cache size and τ — the optimal throughput performance, within a factor of at most 8. The transmission scheme that achieves this performance, employs a simple form of interference enhancement, and exploits the property that weak links attenuate interference, thus allowing for multicasting rates to remain high even when involving weak users. This approach ameliorates the negative effects of uneven topology in multicasting, now allowing all users to achieve the optimal performance associated to $\tau = 1$, even if τ is approximately as low as $\tau \geq 1-(1-w)^g$ where g is the coded-caching gain, and where w is the fraction of users that are weak. This leads to the interesting conclusion that for coded multicasting, the weak users need not bring down the performance of all users, but on the contrary to a certain extent, the strong users can lift the performance of the weak users without any penalties on their own performance. Furthermore for smaller ranges of τ, we also see that achieving the near-optimal performance comes with the advantage that the strong users do not suffer any additional delays compared to the case where $\tau = 1$.

6.1 Introduction

As we have noted, recently the seminal work in [2] introduced coded caching as a means of using caches at the receivers in order to induce multicasting opportunities that lead to substantial removal of interference. This breakthrough provided impressive throughput gains, and inspired a sequence of other works such as [13,38,57,58,73,76–81], as well as [46,56,82–85], and even extensions that are specific to wireless networks [15,45,60,62,70,74,86].

Emphasis in [2] was placed on the symmetric, error free, single-stream BC, where each link from the transmitter to any of the receivers was identical, with normalized capacity equal to 1 file per unit of time. For this topologically symmetric setting, it was shown that a delivery phase with delay $T(K) \triangleq \frac{K(1-\gamma)}{1+K\gamma}$ suffices to guarantee the delivery of any K requested files to the users. This was achieved by caching a fraction γ of each file at each cache, and then by using cache-aided multicasting to send the remaining information to $K\gamma + 1$ users at a time. In this symmetric setting, the resulting coding gain $g_{max} \triangleq \frac{K(1-\gamma)}{T(K)} = 1 + K\gamma$ far exceeded the local caching gains typically associated to receiver-side caching.

What was also noticed though is that, because of multicasting, the performance suffered when the links had unequal capacities. Such uneven topologies, where some users have weaker channels than others, introduce the problem that any multicast transmission that is meant for at least one weak user, could conceivably have to be sent at a lower rate, thus 'slowing down' the rest of the strong users as well. For example, if we were to naively apply the delivery scheme in [2] — which consisted of a sequential transmission of $\binom{K}{K\gamma+1}$ different XORs (one XOR for each subset of $K\gamma+1$ users) — we would have the case that even a single weak user would suffice for the performance to deteriorate such that $T(K,\tau) > T(K,\tau=1),\ \forall \tau < 1$. Such topological considerations[1] have motivated work such as that in [81] which — for the setting of the broadcast erasure channel — includes a 'balancing' solution where only weak users have access to caches, while strong users do not.

Our motivation is to mitigate the performance degradation that coded caching experiences when some link capacities are reduced. The key to mitigating this topology-induced degradation, is a simple form of interference enhancement which exploits the natural interference attenuation in the direction of the weak links, and which allows us to maintain — to a certain degree — a constant multicasting flow of normalized rate 1.

[1] In wireless communications, there is a variety of topological factors — including propagation path loss, shadow fading and inter-cell interference [87] — which lead to having some links that are much weaker or stronger than others; a reality that has motivated a variety of works (e.g. [18,19,88–92]) relating to *generalized* degrees of freedom (GDoF).

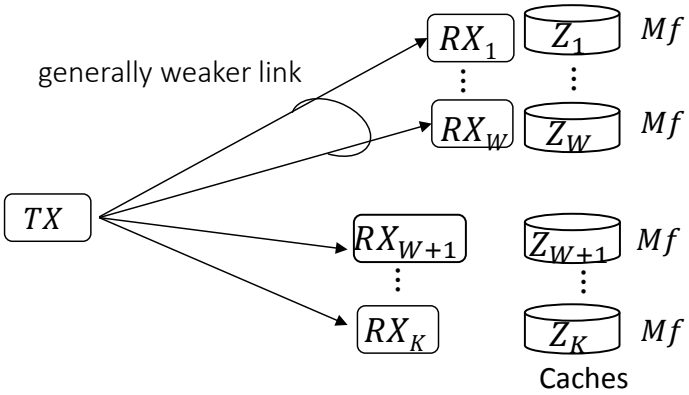

Figure 6.1: Cache-aided K-user SISO BC.

6.1.1 Cache-aided SISO BC

We focus on the topologically-uneven wireless SISO K-user broadcast channel, where $K - W$ users have strong links with unit-normalized capacity, while the remaining W users have links that are weak with normalized capacity τ for some fixed $\tau \in [0, 1]$. For notational convenience we will assume that users $1, 2, \ldots, W$ are weak, and that users $W + 1, \ldots, K$ are strong. In this setting, where a single-antenna transmitter communicates to K single-antenna receiving users, at any time t, the received signal at user k takes the form

$$y_{k,t} = \sqrt{P^{\tau_k}} h_{k,t} x_t + z_{k,t} \quad k = 1, 2, \cdots, K \tag{6.1}$$

where the input signal x_t has bounded power $\mathbb{E}\{|x_t|^2\} \leq 1$, where the fading $h_{k,t}$ and the noise $z_{k,t}$ are assumed to be Gaussian with zero mean and unit variance, and where the link strength is $\tau_k = 1$ for strong users, and $\tau_k = \tau$ for weak users. In this setting, the average received signal to noise ratio (SNR) for the link to user k is given as[2],[3]

$$\mathbb{E}\{|\sqrt{P^{\tau_k}} h_{k,t} x_t|^2\} = P^{\tau_k}.$$

As before, we focus on the high SNR (high P) setting, and we make the normalization — without loss of generality — that each library file W_n, $n = 1, \ldots, N$, has size f (bits) which — in the high SNR setting of interest here — is set equal to $f = \log_2(P)$. Consequently the aforementioned capacity of a strong (interference free) link, is now *1 file per unit of time*, while the capacity

[2]Additionally in the high P regime of interest here, it is easy to see that $Pr(|\sqrt{P^{\tau_k}} h_{k,t}|^2 \doteq P^{\tau_k}) = 1$.

[3]We here use $\doteq$ to denote *exponential equality*, i.e., we write $g(P) \doteq P^B$ to denote $\lim_{P\to\infty} \frac{\log g(P)}{\log P} = B$. Similarly $\dot{\geq}$ and $\dot{\leq}$ will denote exponential inequalities.

of a weak link is τ files per unit of time. The cache Z_k of user k has size Mf bits, where M ($M \leq N$) defines the aforementioned normalized cache size

$$\gamma \triangleq \frac{M}{N}. \tag{6.2}$$

Our results consider the case where $N \geq K$, and consider the measure of performance T — in time slots, per file served per user — needed to complete the delivery process, *for any request*. After the aforementioned normalization $f = \log(P)$, this measure matches that in [2].

6.1.2 Notation and conventions

We will use $\mathcal{K} \triangleq \{1, 2, \cdots, K\}$ to denote the (indices of the) set of all users, $\mathcal{W} \triangleq \{1, 2, \cdots, W\}$ to denote the set of weak users, and $\mathcal{S} \triangleq \{W+1, \cdots, K\}$ to denote the set of strong users. We will also use $w \triangleq W/K$ to define the fraction of the users that are weak. We will use $\Gamma \triangleq \frac{KM}{N} = K\gamma$ to denote the cumulative (normalized) cache size, and for any integer L, we will use

$$T(L) \triangleq \frac{L(1-\gamma)}{1+L\gamma} \tag{6.3}$$

to denote the delay associated to the original coded caching solution in [2] with L strong users and no weak users ($\tau = 1$).

Consequently we will use $T(K) \triangleq \frac{K(1-\gamma)}{1+K\gamma}$ to describe the performance for the case of $L = K$ users, as this was derived in [2] for integer $K\gamma = \{0, 1, \cdots, K\}$ (for the general $K\gamma$, the lower convex envelope of the integer points is achievable). Similarly $T(K-W) = \frac{(K-W)(1-\gamma)}{1+(K-W)\gamma}$ will simply correspond to the case of $L = K - W$, and $T(W) = \frac{W(1-\gamma)}{1+W\gamma}$ to the case of $L = W$, and *we stress that* $T(K), T(K-W), T(W)$ *all correspond to the case of* $\tau = 1$. We here note that for clarity of exposition, we allow for an integer relaxation on $(K-W)\gamma$ and $W\gamma$. This relaxation, which allows for crisp expressions, will be lifted in Section 6.5.4 which, for completeness, presents the extension of the algorithm in [2] for any γ, using memory-sharing between files (see also [40]).

6.2 Throughput of topological cache-aided BC

The following describes, within a factor of 8, the optimal $T(\tau)$ as a function of K, W, γ, τ. The results use the expression

$$\bar{\tau}_{thr} = \frac{T(W)}{T(W) + T(K-W)}$$

and

$$\tau_{thr} = \begin{cases} 1 - \frac{\binom{K-W}{K\gamma+1}}{\binom{K}{K\gamma+1}}, & \text{for } W < K(1-\gamma) \\ 1, & \text{otherwise.} \end{cases} \tag{6.4}$$

The following applies to the case of centralized placement.

Theorem 6.1 *In the K-user topological cache-aided SISO BC with W weak users,*

$$T(\tau)=\begin{cases} \frac{T(W)}{\tau}, & 0 \leq \tau < \bar{\tau}_{thr} \\ \min\{T(K-W)+T(W), \frac{\tau_{thr}T(K)}{\tau}\}, & \bar{\tau}_{thr} \leq \tau \leq \tau_{thr} \\ T(K), & \tau_{thr} < \tau \leq 1 \end{cases} \tag{6.5}$$

is achievable, and has a gap from optimal

$$\frac{T(\tau)}{T^*(\tau)} \leq 8 \tag{6.6}$$

that is always less than 8.

Proof. The scheme that achieves the above performance is presented in Section 6.3, while the corresponding gap to optimal is bounded in Appendix 6.5.1. ■

What the above shows is that there are three regions of interest. In the first region where $\tau \geq \tau_{thr}$, despite the degradation in the link strengths, the performance of all users remains as if all links were uniformly strong (as if $\tau = 1$). In this setting, instead of experiencing the phenomenon that the weak users 'pull down' the performance of all users, we observe the interesting effect of strong users bringing up the performance of the weak users, to the optimal $T(K)$ associated to $\tau = 1$. The conclusion is that in this first region, the reduction in the capacity of the weak links τ, does not translate into a performance degradation. This is because, even when multicasting involves weak users, the employed superposition scheme allows for an overall multicasting rate of 1. Then, there is an intermediate region where there is a degradation in the overall performance by a factor $\frac{\tau_{thr}}{\tau}$ (rather than by a factor $\frac{1}{\tau}$). Finally there is the third region $\tau \leq \bar{\tau}_{thr}$, where due to the substantially limited capacity of the weak links, the transmission to the weak users becomes the bottleneck and the performance is dominated by the delay of serving the weak users, and it deteriorates by a factor $\frac{1}{\tau}$. Interestingly, within this region, and particularly when $\tau \in [0, \frac{w}{1+K(1-w)\gamma}]$, while the near optimal performance reflects the bottleneck due to the weak users, it is also the case (this can be seen in the description of the scheme) that the delivery to the strong users finishes much earlier, and that the strong users do not suffer any additional delays compared to the case where $\tau = 1$; each strong user completes reception of their file with delay that is not bigger than $T(K)$.

In all cases, we see an improvement over the aforementioned naive sequential transmission of XORs, for which it is easy to show that the performance

takes the form

$$T_{nv} = T(K)\big(1 + \frac{\tau_{thr}}{\tau}(1-\tau)\big) \tag{6.7}$$

$$= \frac{T(K)}{\tau}\big(1 - (1-\tau_{thr})(1-\tau)\big) \tag{6.8}$$

where we see that $T_{nv}(\tau) > T(K)$ for any $\tau < 1$. The gains of the proposed method, compared to the naive sequential multicasting, are more prominent when τ is reduced ($0 \leq \tau < \bar{\tau}_{thr}$), and when $K\gamma > 1$ and $W\gamma < 1$, in which case the gains are bounded as

$$\frac{T_{nv}}{T(\tau)} < \frac{2}{W\gamma}$$

and can become large when $W\gamma$ becomes substantially small.

Example 6.1 *($K = 500, W = 50, \gamma = \frac{1}{50}$) Directly from the above we see that*

$$T = \begin{cases} \frac{24.5}{\tau}, & 0 \leq \tau < 0.36 \\ \min\{68.6, \frac{30.7}{\tau}\}, & 0.36 \leq \tau \leq 0.69 \\ T(K) = 44.5, & 0.69 < \tau \leq 1 \end{cases} \tag{6.9}$$

which means that, with a tenth of the users being weak, as long as $\tau \geq 0.69$, there is no performance degradation due to reduced-capacity links, and every user receives their file with delay $T(K) = \frac{K(1-\gamma)}{1+K\gamma} = 44.5$ associated to $\tau = 1$.

Regarding the first region, the following quantifies the intuition that the topology threshold τ_{thr} (until which, capacity reductions do not degrade performance), is a function of the degree of multicasting (coding gain) $g_{max} \triangleq K\gamma + 1 = K(1-\gamma)/T(K)$.

Corollary 6.1 *The threshold τ_{thr} which guarantees full-capacity performance $T(K)$, lies inside the region $\tau_{thr} \in [1-(1-w)^{g_{max}}, 1-(1-w-\frac{w\gamma}{1-\gamma})^{g_{max}}]$, which also means that*

$$T(\tau) = T(K),\ \forall \tau \geq 1 - (1-w)^{g_{max}} + \gamma^{g_{max}}.$$

Thus as γ decreases, this threshold approaches

$$\tau_{thr} \approx 1 - (1-w)^{g_{max}}.$$

Proof. The proof consists of basic algebraic manipulations and can be found in the Appendix. ■

We again note that a simple sequential delivery of the XORs would have resulted in $\tau_{thr} = 1$.

We extend the above to the link-capacity threshold

$$\tau_{thr,G} \triangleq \arg\min\{\tau : T(\tau) \leq G \cdot T(K),\ G \geq 2\} \tag{6.10}$$

until which, the performance loss is restricted to a factor of $G \geq 2$. For example, for any $\tau \geq \tau_{thr,2}$, the scheme guarantees that $T(\tau) \leq 2T(K)$.

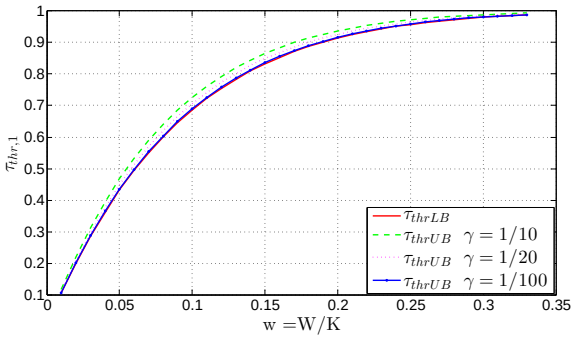

Figure 6.2: $\tau_{thrLB} = 1-(1-w)^{g_{max}}$ denotes the lower bound of τ_{thr}, while $\tau_{thrUB} = 1-(1-w-\frac{w\gamma}{1-\gamma})^{g_{max}}$ denotes the upper bound.

Corollary 6.2 *For any $\tau \geq \tau_{thr,G} = \frac{w}{1+w(g_{max}-1)}\frac{g_{max}}{G}$ $(G \geq 2)$, the performance degradation is bounded as $T(\tau) \leq G \cdot T(K)$.*

Proof. The proof is presented in Appendix 6.5.3. ■

Example 6.2 *$(w = \frac{1}{10}, g_{max} = 11)$ Here, as we have seen, $\tau_{thr} = 0.686$, whereas*

$$\tau_{thr,G} = \frac{0.55}{G}, \ G \geq 2 \tag{6.11}$$

which means that any link-capacity reduction down to, for example, $\tau \geq \tau_{thr,2} = \frac{0.55}{2} = 0.275$, only comes with a performance deterioration of at most 2 $(T(\tau) \leq 2T(K), \ \forall \tau \geq 0.275)$.

6.2.1 Decentralized case

We proceed to provide similar results for the case of decentralized placement, where as described in [38], the caching phase is a random process. The result takes the same form as above, except that now we substitute $T(L)$ from (6.3) with the decentralized equivalent $T'(L) = \frac{1-\gamma}{\gamma}(1-(1-\gamma)^L)$ $(L = K, K-W, W)$ (cf. [38]), and where we substitute $\tau_{thr}, \bar{\tau}_{thr}$ with

$$\tau'_{thr} = \frac{1-(1-\gamma)^W}{1-(1-\gamma)^K}, \ \bar{\tau}'_{thr} = \frac{1-(1-\gamma)^W}{2-(1-\gamma)^W-(1-\gamma)^{K-W}}.$$

For completeness we present the result below.

Theorem 6.2 *In the K-user topological cache-aided SISO BC with W weak users, and decentralized cache placement,*

$$T = \begin{cases} \frac{T'(W)}{\tau}, & 0 \leq \tau < \bar{\tau}'_{thr} \\ T'(K-W)+T'(W), & \bar{\tau}'_{thr} \leq \tau \leq \tau'_{thr} \\ T'(K), & \tau'_{thr} < \tau \leq 1 \end{cases} \tag{6.12}$$

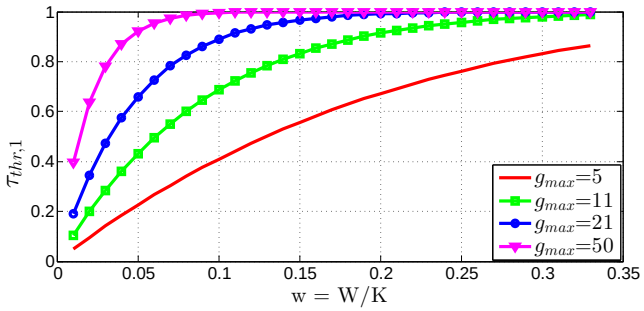

Figure 6.3: τ_{thr} corresponding to distinct values for gains g_{max}. For example, for $g_{max} = 5$ and $w = 0.1$ then $\tau_{thr} \approx 0.4$.

is achievable and order optimal.

The delivery scheme that allows for the above, is identical to the one in the centralized setting (see below), and the only difference is in the analysis of $T(\tau)$ which accounts for the new thresholds $\tau'_{thr}, \bar{\tau}'_{thr}$. The claim that the scheme is order optimal, follows from the arguments in [38] and the arguments in the proof of the gap in the previous theorem.

6.3 Coded caching with simple interference enhancement

We proceed to describe the scheme, for the cases in Theorem 6.1.

6.3.1 Scheme for $\tau \geq \tau_{thr}$

The following applies to the case where $W < K(1-\gamma)$.

Placement phase

The placement phase is identical to that in [2], where we recall that each file W_n, $n = 1, \ldots, N$ is equally split into $\binom{K}{\Gamma}$ subfiles $\{W_{n,\tau}\}_{\tau \in \Psi_\Gamma}$ where $\Psi_\Gamma \triangleq \{\tau \subset \mathcal{K} \ : \ |\tau| = \Gamma\}$, such that each cache Z_k is then filled according to $Z_k = \{W_{n,\tau}\}_{n \in [N], \tau \in \Psi_\Gamma, k \in \tau}$.

Delivery phase

At the beginning of the delivery phase, the transmitter must deliver each requested file W_{R_k} to each receiver k, by delivering the remaining (uncached) subfiles $\{W_{R_k,\tau}\}_{k \notin \tau}$ for each user.

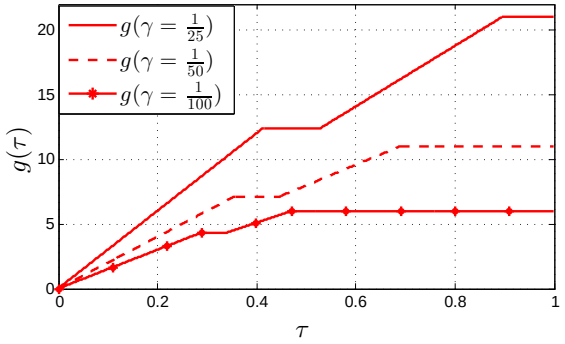

Figure 6.4: The plot shows the gain as a function of τ when $K = 500, W = 50$. The horizontal lines represent the maximum gain g_{max} corresponding to $\tau = 1$, and demonstrate how these can be achieved even with lesser link capacities.

We first recall from [2] that for any $\psi \in \Psi_{\Gamma+1} \triangleq \{\psi \in \mathcal{K} \ : \ |\psi| = \Gamma + 1\}$, then

$$X_\psi \triangleq \oplus_{k\in\psi} W_{R_k,\psi\backslash\{k\}} \tag{6.13}$$

suffices to deliver to each user $k \in \psi$, their requested file $W_{R_k,\psi\backslash k}$. To satisfy all requests $\{W_{R_k}\backslash Z_k\}_{k=1}^K$, the entire set $\mathcal{X}_\Psi \triangleq \{X_\psi\}_{\psi\in\Psi_{\Gamma+1}}$ consisting of $|\mathcal{X}_\Psi| = \binom{K}{\Gamma+1}$ folded messages (XORs), must be delivered, where each XOR contains (has size)

$$|X_\psi| = |W_{R_k,\tau}| = \frac{f}{\binom{K}{\Gamma}} \text{ (bits)}. \tag{6.14}$$

We distinguish between the subset of XORs $\mathcal{X}_{\Psi,s} \triangleq \{X_\psi : \forall\psi, \ s.t. \ \psi \cap \mathcal{W} = \emptyset\} \subset \mathcal{X}_\Psi$ that are only intended for strong users, and the remaining subset $\mathcal{X}_{\Psi,w} \triangleq \mathcal{X}_\Psi\backslash\mathcal{X}_{\Psi,s}$ that have at least one weak user as an intended recipient.

Let T_1 be the duration required to deliver all of $\mathcal{X}_{\Psi,w}$, to all weak users $k \in \mathcal{W}$. Let the transmission first take the form

$$x_t = c_t + b_t, \quad t \in [0, T_1] \tag{6.15}$$

where the power and rate of the symbols are allocated such that

$$\mathbb{E}\{|c_t|^2\} \doteq P^0, \quad r_t^{(c)} = \tau \tag{6.16}$$

$$\mathbb{E}\{|b_t|^2\} \doteq P^{-\tau}, r_t^{(b)} = 1 - \tau \tag{6.17}$$

where $r_t^{(c)}$ (resp. $r_t^{(b)}$) denotes the prelog factor of the number of bits $r_t^{(c)} f$ carried by symbol c_t (resp. $r_t^{(b)}$) at time t. In the above, c_t will carry information from $\mathcal{X}_{\Psi,w}$, while b_t will carry the information from $\mathcal{X}_{\Psi,s}$. As we see, the

reduced power of b_t guarantees that it does not interfere with weak users (at least not above the noise level).

During this period, the received signals $y_{k,t}$ take the form

$$y_{k,t} = \underbrace{\sqrt{P}h_{k,t}c_t}_{P} + \underbrace{\sqrt{P}h_{k,t}b_t}_{P^{1-\tau}} + \underbrace{z_{k,t}}_{P^0}, \quad k \in \mathcal{S} \tag{6.18}$$

$$y_{k,t} = \underbrace{\sqrt{P^\tau}h_{k,t}c_t}_{P^\tau} + \underbrace{\sqrt{P^\tau}h_{k,t}b_t}_{P^0} + \underbrace{z_{k,t}}_{P^0}, \quad k \in \mathcal{W} \tag{6.19}$$

allowing each weak user to directly decode c_t, and allowing each strong user $k \in \mathcal{S}$ to first decode c_t by treating b_t as noise, and to then decode b_t by removing c_t. This is achieved because the interference to the strong users was enhanced (see [93] and [94]) in order for it to be removed.

Depending on the size of $\mathcal{X}_{\Psi,w}$ and $\mathcal{X}_{\Psi,s}$, we will have two cases. In the first case, all the information in $\mathcal{X}_{\Psi,s}$ is delivered by b_t within the aforementioned duration T_1, and thus $T = T_1$. In the second case though, the delivery of $\mathcal{X}_{\Psi,s}$ takes longer than the delivery of $\mathcal{X}_{\Psi,w}$ (longer than T_1), in which case the remaining information is transmitted during an additional period of duration T_2, during which the transmission (as it is intended only for strong users) takes the simpler form

$$x_t = c_t, \quad t \in [T_1, T_1 + T_2] \tag{6.20}$$

during which the power and rate are set as

$$\mathbb{E}\{|c_t|^2\} \doteq P^0, \quad r_t^{(c)} = 1 \tag{6.21}$$

which allows each strong user to directly decode c_t.

In both cases, each strong user can decode $\mathcal{X}_{\Psi,w}$ and $\mathcal{X}_{\Psi,s}$, while each weak user can decode $\mathcal{X}_{\Psi,w}$, and the delivery process is completed.

Calculation of T

To calculate the duration of the delivery phase, let us use

$$Q_{\bar{w}} \triangleq |\mathcal{X}_{\Psi,s}||X_\psi| = \frac{\binom{K-W}{\Gamma+1} f}{\binom{K}{\Gamma}} \quad \text{(bits)}$$

to denote the size (in bits) of $\mathcal{X}_{\Psi,s}$, and let us use

$$Q_w = |\mathcal{X}_\Psi||X_\psi| - Q_{\bar{w}} \quad \text{(bits)}$$

to denote the size of $\mathcal{X}_{\Psi,w}$. We now treat the aforementioned two cases.

Case 1a: $T_1 > \frac{Q_{\bar{w}}}{(1-\tau)f}$ **(this corresponds to** $\tau \in [0, \tau_{thr}]$**)** Here $T = T_1$ is directly calculated, and takes the form

$$T = T_1 = \frac{Q_w}{\tau f} = \frac{1}{\tau}\Big(1 - \frac{\binom{K-W}{\Gamma+1}}{\binom{K}{\Gamma+1}}\Big)\frac{K(1-\gamma)}{1+K\gamma} = \frac{\tau_{thr}T(K)}{\tau}. \tag{6.22}$$

Case 1b: $T_1 \leq \frac{Q_{\bar{w}}}{(1-\tau)f}$ **(this corresponds to** $\tau \in (\tau_{thr}, 1]$**)** The transition to this new case, happens as soon as $T_1 < \frac{Q_{\bar{w}}}{(1-\tau)f}$, which happens as soon as $\tau > \tau_{thr}$ (i.e., $\tau = \tau_{thr}$ is derived by setting $T_1 = \frac{Q_{\bar{w}}}{(1-\tau)f}$). Recall that now $T = T_1 + T_2$. We can easily calculate that the second period (during which we multicast to strong users at full rate) has duration

$$T_2 = \frac{Q_{\bar{w}} - (1-\tau)fT_1}{f}$$

where $Q_{\bar{w}} - (1-\tau)fT_1$ is the amount of the remaining information of $\mathcal{X}_{\Psi,s}$ that had not been handled during the first period of duration T_1. Adding the two components gives us

$$T = T_1 + T_2 = \frac{K(1-\gamma)}{1+K\gamma} = T(K) \tag{6.23}$$

which matches the aforementioned performance $T(K)$ corresponding to uniformly strong topology ($\tau = 1$).

6.3.2 Scheme for the case of $\tau \leq \tau_{thr}$

The following applies for all $W \leq K$. Here the idea is that, because the weak link capacities are small, we treat the weak users separately from the strong users. While we generally transmit to both strong and weak users simultaneously, caching at the strong users is independent of the caching at the weak users, and each XOR is meant either for strong users exclusively, or for weak users exclusively. Transmission again takes the form $x_t = c_t + b_t$, and c_t will deliver the group of XORs meant for weak users, while b_t will deliver the group of XORs for the strong users.

For the case of the weak users, the total information that will be sent is $fT(W)\log(P)$ bits, while for the strong users, this will be $fT(K-W)\log(P)$ bits. There will be again two cases, where the split is again a function of the amount of information that needs to be delivered to the weak vs. to the strong users. In the first case, the transmission and allocation of power and rate, are the same as in (6.15) and (6.16), while in the second case they will be the same as in (6.20) and (6.21).

Case 2a: $\frac{fT(K-W)}{(1-\tau)f} < \frac{fT(W)}{\tau f}$ **(corresponds to** $\tau \in [\bar{\tau}_{thr}, \tau_{thr}]$**)** For this case — corresponding to the scenario where the delivery to the strong users does not take longer than the delivery to the weak users — T can be readily calculated to be

$$T = \frac{fT(W)}{\tau f} = \frac{T(W)}{\tau}.$$

Case 2b: $\frac{fT(K-W)}{(1-\tau)f} \geq \frac{fT(W)}{\tau f}$**, (corresponds to** $\tau \in [0, \bar{\tau}_{thr}]$**)** In this second case, in addition to the above mentioned $T_1 = \frac{T(W)}{\tau}$, the second period duration T_2 is readily calculated to be

$$T_2 = \frac{fT(K-W) - (1-\tau)fT_1}{f}$$

which eventually gives

$$T = T_1 + T_2 = T(K-W) + T(W). \tag{6.24}$$

Combining this with the results corresponding to cases 1a and 2b, gives the desired

$$T(\tau) = \min\{T(K-W) + T(W), \frac{\tau_{thr}T(K)}{\tau}\}.$$

6.4 Conclusion

In this work we explored the behavior of coded caching in the topological broadcast channel (BC), identifying the optimal cache-aided performance within a multiplicative factor of 8. Our proposed scheme uses a simple form of interference enhancement to alleviate the negative effect of having to multicast to both strong and weak links. By showing that the optimal performance can be achieved even in the presence of weaker links, the work reveals a new role of coded caching which is to partially balance the performance between weaker and stronger users, and to a certain degree without any penalty to the performance of the stronger users.

6.5 Appendix

6.5.1 Proving the gap to optimal

To prove the gap to optimal in Theorem 6.1, we first recall from [40] (which corresponds to the case of $\tau = 1$) that $\frac{T(K)}{T^*(\tau=1)} \leq 4$. Let us consider the following three cases.

Case 1 ($\tau_{thr} < \tau \leq 1$) In this case, the bound is direct, by seeing the following

$$\frac{T(\tau)}{T^*(\tau)} = \frac{T(K)}{T^*(\tau)} \leq \frac{T(K)}{T^*(\tau = 1)} \leq 4.$$

Case 2 ($\bar{\tau}_{thr} \leq \tau \leq \tau_{thr}$) We first recall that $T(K)$ is increasing with K, since

$$T(K) = \frac{K(1-\gamma)}{1+K\gamma} = \frac{1-\gamma}{\gamma}(1 - \frac{1}{1+K\gamma}).$$

This means that $T(K-W) \leq T(K)$ and $T(W) \leq T(K)$, and consequently that

$$T(\tau) = \min\{T(K-W) + T(W), \frac{\tau_{thr}T(K)}{\tau}\} \tag{6.25}$$

$$\leq T(K-W) + T(W) \leq 2T(K) \tag{6.26}$$

which yields the desired

$$\frac{T(\tau)}{T^*(\tau)} \leq \frac{2T(K)}{T^*(\tau)} \leq \frac{2T(K)}{T^*(\tau = 1)} \leq 8.$$

Case 3 ($0 < \tau \leq \bar{\tau}_{thr}$) For this case, to get a lower bound on $T(\tau)$, we use the bound in [40] for a system with $K = W$ users, all of them having a link of capacity τ. This means that the lower bound in [40] holds, after simple normalization (division) by τ. At the same time, we know that for this case, the achievable performance here is $\frac{T(W)}{\tau}$. Given that the normalization of the lower bound, matches the normalization of the achievable performance, then the gap remains, as in [40], equal to $\frac{T}{T^*} \leq 4$.

Combining the above three cases, yields the desired

$$\frac{T}{T^*} \leq 8$$

which completes the proof.

6.5.2 Proof of Corollary 6.1

From (6.4) we recall that for $W < K(1-\gamma)$ then $\tau_{thr} = 1 - \frac{\binom{K-W}{K\gamma+1}}{\binom{K}{K\gamma+1}}$. To simplify we note that

$$\frac{\binom{K-W}{K\gamma+1}}{\binom{K}{K\gamma+1}} = \frac{K-W}{K}\frac{K-W-1}{K-1}\cdots\frac{K-W-K\gamma}{K-K\gamma}$$

$$= (1-w)(1-w-\frac{w}{K-1})\cdots(1-w-\frac{wK\gamma}{K-K\gamma})$$

$$= \prod_{i=0}^{K\gamma}(1-w-\frac{wi}{K-i}) \tag{6.27}$$

where the first equation comes from expanding the binomial coefficients $\binom{K-W}{K\gamma+1}$ and $\binom{K}{K\gamma+1}$. Since $\frac{wi}{K-i}$ is increasing with i, we have $0 \leq \frac{wi}{K-i} \leq \frac{wK\gamma}{K-K\gamma}$. Applying this inequality to the last equation above (cf.(6.27)), gives

$$(1-w-\frac{w\gamma}{1-\gamma})^{g_{max}} \leq \frac{\binom{K-W}{K\gamma+1}}{\binom{K}{K\gamma+1}} \leq (1-w)^{g_{max}}$$

which in turn gives the lower and upper bound of τ_{thr}, in the form $\tau_{thrLB} = 1-(1-w)^{g_{max}}$ and $\tau_{thrUB} = 1-(1-w-\frac{w\gamma}{1-\gamma})^{g_{max}}$. It is easy to show that the difference between the upper and lower bound is not larger than $\gamma^{K\gamma+1}$, which vanishes as γ decreases.

6.5.3 Proof of Corollary 6.2

Let us recall from (6.25) that when $\bar{\tau}_{thr} \leq \tau \leq \tau_{thr}$ then

$$T(\tau) = \min\{T(K-W)+T(W), \frac{\tau_{thr}T(K)}{\tau}\} \tag{6.28}$$

$$\leq T(K-W)+T(W) \leq 2T(K) \tag{6.29}$$

which, together with the fact that $G \geq 2$, implies that such a performance degradation (beyond a factor of 2), requires that $\tau < \bar{\tau}_{thr}$, which in turn says that the achievable $T(\tau)$ takes the form $T(\tau) = \frac{T(W)}{\tau}$. Applying this in the definition in (6.10), yields the presented $\tau_{thr,G}$.

6.5.4 Removing the integer relaxation constraint

To remove the aforementioned integer relaxation, we consider the extension of the centralized MN algorithm in [2], to any value of γ (not just when $K\gamma$ is an integer). This has already been addressed in [40] which plots the intermediate values. For the sake of completeness we proceed to explicitly describe the corresponding performance, achieved here by the memory-sharing scheme described below. The following holds for any γ and for $\tau = 1$.

Proposition 6.1 *In the K-user cache-aided SISO BC, with $N \geq K$ files and cache size such that $K\gamma \in [t, t+1], t = 0, 1, \cdots, K-1$, then*

$$\begin{aligned} T''(K) &= \big((t+1)-K\gamma\big)\frac{K-t}{t+1} + (K\gamma - t)\frac{K-(t+1)}{t+2} \\ &= \frac{K-t}{t+1} + \frac{(K\gamma-t)(K+1)}{(t+1)(t+2)} \end{aligned} \tag{6.30}$$

is achievable and it has a gap from optimal

$$\frac{T''(K)}{T^*} \leq 4 \tag{6.31}$$

that is less than 4.

The above maintains the gap from optimal of 4, simply because the interpolation gives an improved performance over the case where $K\gamma \in [1, 2, \dots, K]$ (see also [40]). The expression coincides with the original $T(K)$ for integer values of $K\gamma$. The purpose of this proposition is to allow for the applicability of Theorem 6.1 without the integer relaxation assumption. With $T''(L)$ in place, Theorem 6.1 can apply, simply now with slightly different values for $\bar{\tau}_{thr}$ and τ_{thr}, which though are more complicated and which do not offer any additional insight and are thus omitted.

Below we briefly describe the scheme.

Proof of Proposition 6.1

Let $\Gamma = \frac{KM}{N} \in [t, t+1]$, for some $t = 0, 1, \cdots, K-1$. Let us start by splitting each file W_n into two parts, where the first part $W_n^{(1)}$ has size $\big((t+1) - K\gamma\big)f$ and the second part $W_n^{(2)}$ has size $(K\gamma - t)f$. Split each cache Z_k into two parts, $Z_{k,1}, Z_{k,2}$ such that $\frac{|Z_{k,1}|}{|Z_{k,2}|} = \frac{\big((t+1)-K\gamma\big)}{(K\gamma - t)}$. Focusing on the first part, apply the original MN algorithm, where now the library is $\{W_n^{(1)}\}_{n=1}^N$, the caches are $\{Z_{k,1}\}_{k=1}^K$, and caching is performed as though $K\gamma = t$, i.e., by splitting each half-file $W_n^{(1)}$ into $\binom{K}{t}$ equally-sized subfiles $W_{n,\tau}^{(1)}, \tau \in \Psi_t$ (each subfile now has size $((t+1)-K\gamma)f/\binom{K}{t}$), and by filling the caches according to $Z_{k,1} = \{W_{n,\tau}^{(1)}\}_{n\in[N],\tau\in\Psi_t,k\in\tau}$. Then simply create the sequence of $\binom{K}{t+1}$ XORs (where now each XOR is intended for $t+1$ users), the delivery of which requires

$$T^{(1)} = (t + 1 - K\gamma)\frac{\binom{K}{t+1}}{\binom{K}{t}}. \tag{6.32}$$

We then do the same for the second half of the files (second library $\{W_n^{(2)}\}_{n=1}^N$) except that now we substitute t with $t+1$, to get a corresponding duration of

$$T^{(2)} = (K\gamma - t)\frac{\binom{K}{t+2}}{\binom{K}{t+1}}. \tag{6.33}$$

Combining the two cases yields the whole duration of the delivery phase to be

$$T = T^{(1)} + T^{(2)} = \frac{K-t}{t+1} + \frac{(K\gamma - t)(K+1)}{(t+1)(t+2)} \tag{6.34}$$

which completes the proof.

Chapter 7

Achieving the DoF Limits with Imperfect-Quality CSIT

In this chapter, we investigate the problem of how to exploit imperfect CSIT without involving caching. This is done here in the setting of the two-user single-input single-output X channel. In this setting, recent works have explored the DoF limits in the presence of perfect CSIT, as well as in the presence of perfect-quality delayed CSIT. Our work shows that the same DoF-optimal performance — previously associated to perfect-quality current CSIT — can in fact be achieved with current CSIT that is of imperfect quality. The work also shows that the DoF performance previously associated to perfect-quality delayed CSIT, can in fact be achieved in the presence of imperfect-quality delayed CSIT. These follow from the presented sum-DoF lower bound that bridges the gap — as a function of the quality of delayed CSIT — between the cases of having no feedback and having delayed feedback, and then another bound that bridges the DoF gap — as a function of the quality of current CSIT — between delayed and perfect current CSIT. The bounds are based on novel precoding schemes that are presented here and which employ imperfect-quality current and/or delayed feedback to align interference in space and in time.

7.1 Introduction

We consider the two-user Gaussian single-input single-output (SISO) X channel (XC), with two single-antenna transmitters and two single-antenna receivers, where each transmitter has an independent message for each of the

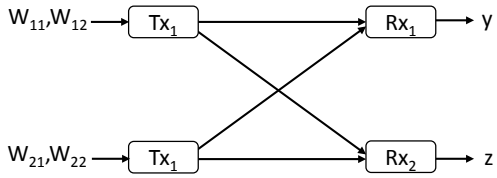

Figure 7.1: 2-user SISO X channel.

two receivers. The corresponding channel model takes the form

$$\begin{aligned} y_t &= h_t^{(1)} x_t^{(1)} + h_t^{(2)} x_t^{(2)} + m_t \\ z_t &= g_t^{(1)} x_t^{(1)} + g_t^{(2)} x_t^{(2)} + n_t \end{aligned} \tag{7.1}$$

where at any time t, $h_t^{(i)}$, $g_t^{(i)}$ denote the scalar fading coefficients of the channel from transmitter i to receiver 1 and 2 respectively, where m_t, n_t denote the unit-power AWGN noise at the two receivers, and where $x_t^{(i)}$, $i = 1, 2$ denotes the transmitted signals at transmitter i, satisfying a power constraint $\mathbb{E}(|x_t^{(i)}|^2) \leq P$. Naturally each $x_t^{(i)}$ may include some private information – originating from transmitter i – intended for receiver 1, and some private information intended for receiver 2.

In this setting, for a quadruple of achievable rates R_{ij}, $i, j = 1, 2$ corresponding to communication from transmitter i to receiver j, we adopt the high-SNR DoF approximation

$$d_{ij} = \lim_{P \to \infty} \frac{R_{ij}}{\log P}, \quad i, j = 1, 2 \tag{7.2}$$

to describe the limits of performance over the XC, particularly focusing on the sum DoF measure $d_\Sigma := d_{11} + d_{21} + d_{12} + d_{22}$.

In this context, the challenge originates from the fact that each transmitter is both an interferer as well as an intended transmitter to both receivers. Crucial in addressing this challenge is the role of feedback — and specifically of CSIT — which can allow for separation, at each receiver, of the intended and the interfering signals. In particular, while the optimal sum DoF without CSIT has been shown to be $d_\Sigma = 1$ (cf. [25]), the DoF increases to $d_\Sigma = \frac{6}{5}$ in the presence of perfect-quality delayed CSIT (see [98] which proved that this performance is optimal over all linear schemes), and the DoF further increases to an optimal sum-DoF of $d_\Sigma = \frac{4}{3}$ (see [99]) in the presence of perfect-quality and instantaneously available CSIT (perfect current CSIT), other related work can be see in [100–102].

7.1.1 Feedback quality model

Motivated by practical settings of limited feedback links, we here consider the case where feedback can be of imperfect-quality, and potentially also de-

layed. Towards this, let $\hat{h}_t^{(i)}, \hat{g}_t^{(i)}$ denote the current CSIT estimates of channels $h_t^{(i)}, g_t^{(i)}$ respectively, and let

$$\tilde{h}_t^{(i)} = h_t^{(i)} - \hat{h}_t^{i}, \quad \tilde{g}_t^{(i)} = g_t^{(i)} - \hat{g}_t^{(i)} \tag{7.3}$$

be the estimation errors, modeled here as having i.i.d Gaussian entries.

Similarly for delayed CSIT, along the same lines as in [1, 103], let $\check{h}_t^{(i)}, \check{g}_t^{(i)}$ denote the delayed estimates of channels $h_t^{(i)}, g_t^{(i)}$, where these estimates are obtained sometime after the channel elapses, and let

$$\ddot{h}_t^{(i)} = h_t^{(i)} - \check{h}_t^{(i)}, \quad \ddot{g}_t^{(i)} = g_t^{(i)} - \check{g}_t^{(i)} \tag{7.4}$$

be the associated CSIT errors, modeled here as having i.i.d Gaussian entries.

In our context, we again consider

$$\alpha = -\lim_{P\to\infty} \frac{\log \mathbb{E}[||\tilde{h}_t^{(i)}||^2]}{\log P} = -\lim_{P\to\infty} \frac{\log \mathbb{E}[||\tilde{g}_t^{(i)}||^2]}{\log P}$$

to here be the *current CSIT quality exponent* describing the quality of current CSIT, equally for both $i = 1, 2$, and we similarly consider

$$\beta = -\lim_{P\to\infty} \frac{\log \mathbb{E}[||\ddot{h}_t^{(i)}||^2]}{\log P} = -\lim_{P\to\infty} \frac{\log \mathbb{E}[||\ddot{g}_t^{(i)}||^2]}{\log P}$$

to be the *delayed-CSIT quality exponent*. In our DoF setting, and following arguments directly from [29], we can safely consider that the two quality exponents are bounded as

$$0 \leq \alpha \leq \beta \leq 1.$$

Having $\alpha = 1$ corresponds to the case of perfect CSIT, for which case — as stated above — the optimal sum DoF was established to be equal to $d_\Sigma = \frac{4}{3}$, while having $\beta = 1$ ($\alpha = 0$), corresponds to the case of perfect-quality delayed CSIT, for which case the optimal linear sum DoF was established in [98] to be $d_\Sigma = \frac{6}{5}$.

7.2 DoF performance with imperfect-quality current and delayed CSIT

Motivated by works such as that in [104] which presented a distributed interference management technique which can obtain the optimal DoF with local and perfect current CSIT with a certain *fractional* delay, and by works on imperfect current and delayed CSIT over the broadcast channel [9, 20, 21, 105], we here explore the role of feedback in moving between these extremal points ($\alpha = 1$ and $\beta = 1, \alpha = 0$) by considering different values of α and β.

Theorem 7.1 *For the two-user XC with perfect-quality delayed CSIT, and with imperfect current CSIT of quality exponent α, the optimal sum DoF is lower bounded as*

$$d_\Sigma \geq \min\left(\frac{4}{3}, \frac{6}{5} + \frac{2\alpha(2-3\alpha)}{5(4-7\alpha)}\right). \tag{7.5}$$

As a result, the optimal sum DoF $d_\Sigma = \frac{4}{3}$ can be achieved with imperfect current CSIT of quality that need not exceed $\alpha = \frac{4}{9}$.

Proof. The result follows by analyzing the performance of the communication scheme which will be presented in section 7.3.1. ■

We note that the above expression, evaluated at $\alpha = 0$, yields the aforementioned sum DoF $d_\Sigma = \frac{6}{5}$.

We now shift attention to the case of imperfect-quality delayed feedback, and of no (or very limited) current feedback, corresponding to having $\beta < 1$ and $\alpha = 0$. As argued in [9], interest in imperfect-quality *delayed* CSIT relates to the fact that β is more indicative of the quality of the *entirety* of feedback (timely plus delayed), and hence, any attempt to limit the total amount of feedback — that is communicated during a certain communication process — must focus on reducing β, rather than just focusing on reducing α. We proceed with the associated result.

Theorem 7.2 *For the two-user XC with no current CSIT and with imperfect delayed CSIT of quality exponent β, the optimal sum DoF is lower bounded as*

$$d_\Sigma \geq \min\left(\frac{6}{5}, 1 + \frac{\beta}{3}\right). \tag{7.6}$$

As a result, the (linear-) optimal sum-DoF $d_\Sigma = \frac{6}{5}$, previously associated to perfect-quality delayed feedback, can in fact be achieved with imperfect-quality delayed CSIT of quality that need not exceed $\beta = \frac{3}{5}$.

Proof. The result follows by analyzing the performance of the communication scheme which will be presented in section 7.3.2. ■

7.3 Achievable schemes for SISO XC

We proceed with the description of the scheme corresponding to Theorems 7.1 and 7.2.

The schemes are designed to have S phases, where the sth phase ($s = 1, \cdots, S$) consists of T_s channel uses. In describing the schemes, we will use a double time index s, t to correspond to the t^{th} time slot, $t = 1, \cdots, T_s$, of phase s.

The general structure of the transmitted signals at any timeslot t of phase s, will be

$$\boldsymbol{x}_{s,t} = \begin{bmatrix} u_{s,t}^{(1)} a_{s,t}^{(1)} + u'^{(1)}_{s,t} a'^{(1)}_{s,t} + v_{s,t}^{(1)} b_{s,t}^{(1)} + v'^{(1)}_{s,t} b'^{(1)}_{s,t} \\ c_{s,t} + u_{s,t}^{(2)} a_{s,t}^{(2)} + u'^{(2)}_{s,t} a'^{(2)}_{s,t} + v_{s,t}^{(2)} b_{s,t}^{(2)} + v'^{(2)}_{s,t} b'^{(2)}_{s,t} \end{bmatrix}$$

where, depending on the instance, some of these symbols will be deactivated resulting in a simpler transmitted signal. In the above, $a_{s,t}^{(i)}, a'^{(i)}_{s,t}$ will denote independent information symbols, from transmitter i to receiver 1, while symbols $b_s^{(i)}, b'^{(i)}_s$ are intended for receiver 2, again from transmitter i. In addition, $c_{s,t}$ will represent a common information symbol generally intended for both receivers. Furthermore $u_{s,t}^{(i)}, u'^{(i)}_{s,t}, v_{s,t}^{(i)}, v'^{(i)}_{s,t}$ are unit-norm 'precoding' scalars which — when combined in time and space — help align the interference from the distributed transmitters at the unintended receivers.

Communication takes place under an average power constraint P on both transmitters. We use the following notation to describe the allocated power on different symbols

$$P_s^{(a)} \triangleq \mathbb{E}|a_{s,t}^{(i)}|^2,\ P_s^{(a')} \triangleq \mathbb{E}|a'^{(i)}_{s,t}|^2,\ P_s^{(b)} \triangleq \mathbb{E}|b_{s,t}^{(i)}|^2,\ P_s^{(b')} \triangleq \mathbb{E}|b'^{(i)}_{s,t}|^2$$

and note that this holds equally for both transmitters, $i = 1, 2$. Furthermore, we use r_s^a to mean that, during phase s, each symbol $a_{s,t}, t = 1, \cdots, T_s$, carries $r_s^a \log P + o(\log P)$ bits. Similarly, we use $r_s^{a'}, r_s^b, r_s^{b'}, r_s^c$ to describe the prelog factor of the number of bits carried by $a'^{(i)}_{s,t}, b_{s,t}^{(i)}, b'^{(i)}_{s,t}, c_{s,t}$ respectively.

We now proceed with the details of the first scheme.

7.3.1 Schemes for XC with imperfect current CSIT

We note that typically, a receiver encounters interference originating from two transmitters. The general idea behind our scheme is that a receiver uses linear combinations of received signals to remove as much interference as possible from one transmitter, and then have the other transmitter help out — with precoding that employs imperfect-current and delayed feedback — in removing the remaining interference. This will be achieved with a proper choice of precoding scalars that are functions of imperfect current and delayed CSIT. Given that this CSIT can be of imperfect quality, the interference may not be fully removed immediately, thus forcing power-and-rate regulation of the information symbols, as well as a multiphase scheme that uses proactive encoding which handles interference at later stages of the communication process, and which then allows for retrospective decoding of the original private information.

Coding

The phase durations $T_1, T_2, \cdots, T_S$ are chosen to be integers such that

$$\begin{aligned} &T_2 = T_1\xi, \quad T_s = T_{s-1}\mu, \ \forall s \in \{3, 4, \cdots, S-1\}, \\ &T_S = T_{S-1}\gamma = T_1\xi\mu^{S-3}\gamma \end{aligned} \tag{7.7}$$

where $\xi = \frac{8(1-\alpha)}{3(4-7\alpha)}, \mu = \frac{2\alpha}{4-7\alpha}, \gamma = \frac{\alpha}{2(1-\alpha)}$.

Phase 1 Phase 1 consists of $\frac{T_1}{3}$ sub-phases, with each sub-phase consisting of three consecutive time slots. We will focus on the first such sub-phase (i.e., the first 3 time slots of the first phase), corresponding to time $(1,1),(1,2),(1,3)$. The rest of the sub-phases will simply be a repetition of this first sub-phase, with each sub-phase corresponding to new information symbols. In this first sub-phase, the transmitted signals are

$$\boldsymbol{x}_{1,1} = \begin{bmatrix} a_{1,1}^{(1)} \\ a_{1,1}^{(2)} + a'^{(2)}_{1,1} \end{bmatrix}, \boldsymbol{x}_{1,2} = \begin{bmatrix} b_{1,2}^{(1)} \\ b_{1,2}^{(2)} + b'^{(2)}_{1,2} \end{bmatrix}, \boldsymbol{x}_{1,3} = \begin{bmatrix} a_{1,1}^{(1)} + b_{1,2}^{(1)} \\ u_{1,3}^{(2)} a_{1,1}^{(2)} + v_{1,3}^{(2)} b_{1,2}^{(2)} \end{bmatrix}$$

where the power and normalized rates are set as

$$\begin{aligned} P_1^{(a)} \doteq P_1^{(b)} \doteq P, \ P_1^{(a')} \doteq P_1^{(b')} \doteq P^{1-\alpha} \\ r_1^{(a)} = r_1^{(b)} = 1, \ r_1^{(a')} = r_1^{(b')} = 1-\alpha \end{aligned} \tag{7.8}$$

and where

$$u_{1,3}^{(2)} = \frac{g_{1,1}^{(2)} \hat{g}_{1,3}^{(1)}}{g_{1,1}^{(1)} \hat{g}_{1,3}^{(2)}}, \qquad v_{1,3}^{(2)} = \frac{h_{1,2}^{(2)} \hat{h}_{1,3}^{(1)}}{h_{1,2}^{(1)} \hat{h}_{1,3}^{(2)}}.$$

To gain insight into the workings of the scheme, we note that, for example, $u_{1,3}^{(2)}$ is chosen to assist receiver 2 remove the interference from transmitter 2 using delayed estimates $g_{1,t}^{(1)}, g_{1,t}^{(2)}$ as well as using current imperfect estimates of the two channels leading to receiver 2 from the two transmitters. We note that the above expression reflects our assumption that delayed CSIT here is of perfect quality.

Excluding the noise term for the sake of brevity, the received signals at receiver 1 take the form

$$\begin{aligned} y_{1,1} &= h_{1,1}^{(1)} a_{1,1}^{(1)} + h_{1,1}^{(2)} (a_{1,1}^{(2)} + a'^{(2)}_{1,1}) \\ y_{1,2} &= h_{1,2}^{(1)} b_{1,2}^{(1)} + h_{1,2}^{(2)} (b_{1,2}^{(2)} + b'^{(2)}_{1,2}) \\ y_{1,3} &= h_{1,3}^{(1)} (a_{1,1}^{(1)} + b_{1,2}^{(1)}) + h_{1,3}^{(2)} (u_{1,3}^{(2)} a_{1,1}^{(2)} + v_{1,3}^{(2)} b_{1,2}^{(2)}). \end{aligned} \tag{7.9}$$

Upon receiving the above, receiver 1 removes the unintended symbol $b_{1,2}^{(1)}$ from transmitter 1, using the following linear combination, to get

$$\begin{aligned} & y_{1,3}/h_{1,3}^{(1)} - y_{1,2}/h_{1,2}^{(1)} \\ & = \underbrace{a_{1,1}^{(1)}}_{P} + \underbrace{\frac{h_{1,3}^{(2)}}{h_{1,3}^{(1)}} u_{1,3}^{(2)} a_{1,1}^{(2)}}_{P} + \overbrace{\underbrace{(\frac{h_{1,3}^{(2)}}{h_{1,3}^{(1)}} v_{1,3}^{(2)} - \frac{h_{1,2}^{(2)}}{h_{1,2}^{(1)}}) b_{1,2}^{(2)} - \frac{h_{1,2}^{(2)}}{h_{1,2}^{(1)}} b'^{(2)}_{1,2}}_{P^{(1-\alpha)}}}^{i_{1,1}} \end{aligned}$$

where under each term we noted the order of the summand's average power, where $i_{1,1}$ denotes the interference from transmitter 2 onto receiver 1 during this first sub-phase of the first phase, and where the power of this interference is bounded as

$$\begin{aligned} \mathbb{E}|i_{1,1}|^2 &= \mathbb{E}\left| \frac{h_{1,2}^{(2)}}{h_{1,2}^{(1)}} (\frac{h_{1,3}^{(2)}}{h_{1,3}^{(1)}} \frac{\hat{h}_{1,3}^{(1)}}{\hat{h}_{1,3}^{(2)}} - 1) b_{1,2}^{(2)} \right|^2 + \mathbb{E}\left| \frac{h_{1,2}^{(2)}}{h_{1,2}^{(1)}} b'^{(2)}_{1,2} \right|^2 \\ &\dot{=} \mathbb{E}\left| \frac{h_{1,2}^{(2)}}{h_{1,2}^{(1)}} \frac{\tilde{h}_{1,3}^{(2)} \hat{h}_{1,3}^{(1)} - \hat{h}_{1,3}^{(2)} \tilde{h}_{1,3}^{(1)}}{h_{1,3}^{(1)} \hat{h}_{1,3}^{(2)}} b_{1,2}^{(2)} \right|^2 + P^{1-\alpha} \dot{=} P^{1-\alpha}. \end{aligned} \tag{7.10}$$

In the above, precoding with $v_{1,3}^{(2)}$, managed to bring down the residual interference of $b_{1,2}^{(2)}$ (due to imperfections in current CSIT), to the levels of the interference imposed by $b'^{(2)}_{1,2}$.

Receiver 2, which follows a parallel course of action, now experiences interference $\theta_{1,1}$ from transmitter 2, where this interference is similarly bounded above by $P^{1-\alpha}$. At the end of this first sub-phase (3 time slots), transmitter 2 uses its partial knowledge of current CSIT to reconstruct $\{i_{1,1}, \theta_{1,1}\}$, and to quantize each term to get

$$\bar{i}_{1,1} = i_{1,1} - \tilde{i}_{1,1}, \quad \bar{\theta}_{1,1} = \theta_{1,1} - \tilde{\theta}_{1,1}.$$

Allowing for $(1-\alpha)\log P$ bits per interference term, allows in turn for

$$\mathbb{E}(|\tilde{i}_{1,1}|^2) \dot{=} \mathbb{E}(|\tilde{\theta}_{1,1}|^2) \dot{=} 1.$$

At this point, this same procedure described here for the first sub-phase of the first phase, is repeated for the remaining $\frac{T_1}{3} - 1$ sub-phases, corresponding though to new information. This process results in the accumulation of a total of $\frac{2T_1}{3}(1-\alpha)\log P$ quantization bits representing residual interference. These bits will be distributed evenly into the common symbols $\{c_{2,t}\}_{t=1}^{T_2}$ of the second phase.

2) Phase s, $2 \leq s \leq S-1$: Phase s consists of $\frac{T_s}{4}$ sub-phases, each consisting of 4 consecutive channel uses. As before, we describe the first sub-phase, corresponding to time $(s,1),(s,2),(s,3),(s,4)$, where the transmitted signals are

$$\boldsymbol{x}_{s,1} = \begin{bmatrix} a_{s,1}^{(1)} + a'^{(1)}_{s,1} \\ c_{s,1} + a_{s,1}^{(2)} \end{bmatrix}, \quad \boldsymbol{x}_{s,2} = \begin{bmatrix} b_{s,2}^{(1)} + b'^{(1)}_{s,2} \\ c_{s,2} + b_{s,2}^{(2)} \end{bmatrix}$$

$$\boldsymbol{x}_{s,3} = \begin{bmatrix} a_{s,1}^{(1)} + a'^{(1)}_{s,1} + b_{s,2}^{(1)} + b'^{(1)}_{s,2} \\ c_{s,3} + u_{s,3}^{(2)}(a_{s,1}^{(2)} + a'^{(2)}_{s,3}) + v_{s,3}^{(2)}(b_{s,2}^{(2)} + b'^{(2)}_{s,3}) \end{bmatrix}$$

$$\boldsymbol{x}_{s,4} = \begin{bmatrix} u_{s,4}^{(1)} a_{s,1}^{(1)} + v_{s,4}^{(1)} b_{s,2}^{(1)} \\ c_{s,4} + a'^{(2)}_{s,3} + b'^{(2)}_{s,3} \end{bmatrix}$$

where

$$\begin{aligned} u_{s,3}^{(2)} &= \frac{g_{s,1}^{(2)} \hat{g}_{s,3}^{(1)}}{g_{s,1}^{(1)} \hat{g}_{s,3}^{(2)}}, \quad u_{s,4}^{(1)} = \frac{g_{s,1}^{(1)}}{g_{s,1}^{(2)}} \left(\frac{g_{s,3}^{(1)} \hat{g}_{s,3}^{(2)}}{g_{s,3}^{(2)} \hat{g}_{s,3}^{(1)}} - 1\right) \frac{\hat{g}_{s,4}^{(2)}}{\hat{g}_{s,4}^{(1)}} \\ v_{s,3}^{(2)} &= \frac{h_{s,2}^{(2)} \hat{h}_{s,3}^{(1)}}{h_{s,2}^{(1)} \hat{h}_{s,3}^{(2)}}, \quad v_{s,4}^{(1)} = \frac{h_{s,2}^{(1)}}{h_{s,2}^{(2)}} \left(\frac{h_{s,3}^{(1)} \hat{h}_{s,3}^{(2)}}{h_{s,3}^{(2)} \hat{h}_{s,3}^{(1)}} - 1\right) \frac{\hat{h}_{s,4}^{(2)}}{\hat{h}_{s,4}^{(1)}} \end{aligned} \tag{7.11}$$

and where the rates and power are allocated as follows

$$\begin{aligned} P_1^{(c)} \doteq P,\ P_1^{(a)} = P_1^{(b)} \doteq P^{2\alpha},\ P_1^{(a')} = P_1^{(b')} \doteq P^{\alpha} \\ r_1^{(c_{s,1})} = r_1^{(c_{s,2})} = r_1^{(c_{s,3})} = 1 - 2\alpha, r_1^{(c_{s,3})} = 1 - \alpha, \\ r_1^{(a)} = r_1^{(b)} = 2\alpha, r_1^{(a')} = r_1^{(b')} = \alpha. \end{aligned} \tag{7.12}$$

We focus on the noiseless version of the signals of the first receiver, which take the form

$$\begin{aligned} y_{s,1} &= h_{s,1}^{(2)} c_{s,1} + h_{s,1}^{(1)}(a_{s,1}^{(1)} + a'^{(1)}_{s,1}) + h_{s,1}^{(2)} a_{s,1}^{(2)} \\ y_{s,2} &= h_{s,2}^{(2)} c_{s,2} + h_{s,2}^{(1)}(b_{s,2}^{(1)} + b'^{(1)}_{s,2}) + h_{s,2}^{(2)} b_{s,2}^{(2)} \\ y_{s,3} &= h_{s,3}^{(2)} c_{s,3} + h_{s,3}^{(1)}(a_{s,1}^{(1)} + a'^{(1)}_{s,1} + b_{s,2}^{(1)} + b'^{(1)}_{s,2}) \\ &\quad + h_{s,3}^{(2)}\Big(u_{s,3}^{(2)}(a_{s,1}^{(2)} + a'^{(2)}_{s,3}) + v_{s,3}^{(2)}(b_{s,2}^{(2)} + b'^{(2)}_{s,3})\Big) \\ y_{s,4} &= h_{s,4}^{(2)} c_{s,4} + h_{s,4}^{(1)}(u_{s,4}^{(1)} a_{s,1}^{(1)} + v_{s,4}^{(1)} b_{s,2}^{(1)}) + h_{s,4}^{(2)}(a'^{(2)}_{s,3} + b'^{(2)}_{s,3}). \end{aligned}$$

At this point, receiver 1 decodes $c_{s,1}, c_{s,2}, c_{s,3}, c_{s,4}$ by treating the other signals as noise[1]. After removal of these common symbols, receiver one has $y'_{s,t} =$

[1] For the case of $c_{s,4}$, note that this is achieved by proper choice of $u_{s,4}^{(1)}$.

$y_{s,t} - h^{(2)}_{s,t} c_{s,t}, t = 1, \cdots, 4$, where

$$\begin{aligned}
y'_{s,1} &= h^{(1)}_{s,1}(a^{(1)}_{s,1} + a'^{(1)}_{s,1}) + h^{(2)}_{s,1} a^{(2)}_{s,1} \\
y'_{s,2} &= h^{(1)}_{s,2}(b^{(1)}_{s,2} + b'^{(1)}_{s,2}) + h^{(2)}_{s,2} b^{(2)}_{s,2} \\
y'_{s,3} &= h^{(1)}_{s,3}(a^{(1)}_{s,1} + a'^{(1)}_{s,1} + b^{(1)}_{s,2} + b'^{(1)}_{s,2}) \\
&\quad + h^{(2)}_{s,3}\Big(u^{(2)}_{s,3}(a^{(2)}_{s,1} + a'^{(2)}_{s,3}) + v^{(2)}_{s,3}(b^{(2)}_{s,2} + b'^{(2)}_{s,3})\Big) \\
y'_{s,4} &= h^{(1)}_{s,4}(u^{(1)}_{s,4} a^{(1)}_{s,1} + v^{(1)}_{s,4} b^{(1)}_{s,2}) + h^{(2)}_{s,4}(a'^{(2)}_{s,3} + b'^{(2)}_{s,3}).
\end{aligned} \tag{7.13}$$

Interference alignment and power reducing Receiver 1 considers the following linear combination

$$\frac{y'_{s,3}}{h^{(1)}_{s,3}} - \frac{y'_{s,2}}{h^{(1)}_{s,2}} = \sigma_{s,1} + \overbrace{\underbrace{(\frac{h^{(2)}_{s,3}}{h^{(1)}_{s,3}} v^{(2)}_{s,3} - \frac{h^{(2)}_{s,2}}{h^{(1)}_{s,2}}) b^{(2)}_{s,2} + \frac{h^{(2)}_{s,3}}{h^{(1)}_{s,3}} v^{(2)}_{s,3} b'^{(2)}_{s,3}}_{P^\alpha}}^{i_{s,1}}$$

as a means of canceling the unintended information from transmitter 1. In the above, we use $\sigma_{s,1}$ to simply denote the part of the received signal — during this sub-phase — that consists of desired symbols, while we also recall that $i_{s,1}$ denotes the interference from transmitter 2. The interference relating to $b^{(1)}_{s,2} + b'^{(1)}_{s,2}$ has been removed by the actions of transmitter 1. Choosing $v^{(2)}_{s,3} = \frac{h^{(2)}_{s,2}\hat{h}^{(1)}_{s,3}}{h^{(1)}_{s,2}\hat{h}^{(2)}_{s,3}}$, guarantees that

$$\begin{aligned}
\mathbb{E}|i_{s,1}|^2 &= \mathbb{E}\left|(\frac{h^{(2)}_{s,3}}{h^{(1)}_{s,3}} v^{(2)}_{s,3} - \frac{h^{(2)}_{s,2}}{h^{(1)}_{s,2}}) b^{(2)}_{s,2}\right|^2 + \mathbb{E}\left|\frac{h^{(2)}_{s,3}}{h^{(1)}_{s,3}} v^{(2)}_{s,3} b'^{(2)}_{s,3}\right|^2 \\
&\doteq \mathbb{E}\left|\frac{h^{(2)}_{s,2}}{h^{(1)}_{s,2}} \frac{(\tilde{h}^{(2)}_{s,3}\hat{h}^{(1)}_{s,3} - \tilde{h}^{(1)}_{s,3}\hat{h}^{(2)}_{s,3})}{h^{(1)}_{s,3}\hat{h}^{(2)}_{s,3}} b^{(2)}_{s,2}\right|^2 + P^\alpha \doteq P^\alpha
\end{aligned} \tag{7.14}$$

and using $y'_{s,2}$ and $y'_{s,3}$, receiver 1 removes the interference from transmitter 1 and also reduces interference from transmitter 2. Similar arguments apply also for the interference $\theta_{s,1}$ at receiver 2 originating from transmitter 2.

We also consider

$$\frac{y'_{s,3}}{h^{(2)}_{s,3} v^{(2)}_{s,3}} - \frac{y'_{s,2}}{h^{(2)}_{s,2}} - \frac{y'_{s,4}}{h^{(2)}_{s,4}} = \sigma_{s,2} + \underbrace{(\eta - \frac{h^{(1)}_{s,4}}{h^{(2)}_{s,4}} v^{(1)}_{s,4}) b^{(1)}_{s,2}}_{P^0} + \underbrace{\eta b'^{(1)}_{s,2}}_{P^0}$$

where $\sigma_{s,2}$ is the desired signal, and where $\eta = \frac{h_{s,3}^{(1)}}{h_{s,3}^{(2)}v_{s,3}^{(2)}} - \frac{h_{s,2}^{(1)}}{h_{s,2}^{(2)}}$. With proper choice of precoding scalars, we have

$$\mathbb{E}\left|(\eta - \frac{h_{s,4}^{(1)}}{h_{s,4}^{(2)}}v_{s,4}^{(1)})b_{s,2}^{(1)}\right|^2 = \mathbb{E}\left|\frac{h_{s,2}^{(1)}}{h_{s,2}^{(2)}}(\frac{h_{s,3}^{(1)}\hat{h}_{s,3}^{(2)}}{h_{s,3}^{(2)}\hat{h}_{s,3}^{(1)}} - 1)(1 - \frac{h_{s,3}^{(1)}\hat{h}_{s,3}^{(2)}}{h_{s,3}^{(2)}\hat{h}_{s,3}^{(1)}})b_{s,2}^{(1)}\right|^2$$
$$= \mathbb{E}\left|\frac{h_{s,2}^{(1)}}{h_{s,2}^{(2)}}\frac{(\tilde{h}_{s,3}^{(1)}\hat{h}_{s,3}^{(2)} - \tilde{h}_{s,3}^{(2)}\hat{h}_{s,3}^{(1)})}{h_{s,3}^{(2)}\hat{h}_{s,3}^{(1)}}\frac{(\tilde{h}_{s,3}^{(2)}\hat{h}_{s,3}^{(1)} - \tilde{h}_{s,3}^{(1)}\hat{h}_{s,3}^{(2)})}{h_{s,3}^{(2)}\hat{h}_{s,3}^{(1)}}b_{s,2}^{(1)}\right|^2 \doteq P^0$$

and

$$\mathbb{E}\left|\eta b'^{(1)}_{s,2}\right|^2 = \mathbb{E}\left|\frac{h_{s,2}^{(1)}}{h_{s,2}^{(2)}}\frac{(\tilde{h}_{s,3}^{(1)}\hat{h}_{s,3}^{(2)} - \tilde{h}_{s,3}^{(2)}\hat{h}_{s,3}^{(1)})}{h_{s,3}^{(2)}\hat{h}_{s,3}^{(1)}}b'^{(1)}_{s,2}\right|^2 \doteq P^0$$

Using $y'_{s,2}$ and $y'_{s,4}$, receiver 1 removes the interference relating to $b_{s,2}^{(2)} + b'^{(2)}_{s,3}$ in $y'_{s,3}$, and the power of $b_{s,2}^{(1)}$ and $b'^{(2)}_{s,2}$ is reduced below the noise level.

Similarly receiver 1 can also get

$$\sigma_{s,3} = \sigma_{s,2} - \frac{h_{s,3}^{(1)}}{h_{s,3}^{(2)}v_{s,3}^{(2)}}\sigma_{s,1} = -\frac{h_{s,4}^{(1)}}{h_{s,4}^{(2)}}u_{s,4}^{(1)}a_{s,1}^{(1)} - a'^{(2)}_{s,3}$$

which will be used later to decode.

Quantizing and retransmitting the interference Up to this point, we have removed the unintended signals from transmitter 1, and now focus on the interference originating from transmitter 2. After the first sub-phase of phase s, transmitter 1 reconstructs $i_{s,1}, \theta_{s,1}$ using its knowledge of delayed CSIT, and quantizes these into

$$\bar{i}_{s,1} = i_{s,1} - \tilde{i}_{s,1}, \quad \bar{\theta}_{s,1} = \theta_{s,1} - \tilde{\theta}_{s,1}$$

requiring a total of $2\alpha \log P$ bits, allowing for bounded noise

$$\mathbb{E}(|\tilde{i}_{s,1}|^2) \doteq \mathbb{E}(|\tilde{\theta}_{s,1}|^2) \doteq 1.$$

Consequently during phase s, a total of $\frac{\alpha}{2}T_s \log P$ bits are accumulated, and will be distributed evenly into the common symbol sets $\{c_{s+1,t}\}_{t=1}^{T_{s+1}}$ of the next phase.

3) Phase S: Phase S has $\frac{T_S}{3}$ sub-phases, each consisting of 3 consecutive time slots. Focusing on $(S,1),(S,2),(S,3)$, we have

$$\boldsymbol{x}_{S,1} = \begin{bmatrix} a_{S,1}^{(1)} \\ c_{S,1} + a_{S,1}^{(1)} \end{bmatrix}, \ \boldsymbol{x}_{S,2} = \begin{bmatrix} b_{S,2}^{(1)} \\ c_{S,2} + b_{S,2}^{(2)} \end{bmatrix},$$

$$\boldsymbol{x}_{S,3} = \begin{bmatrix} a_{S,1}^{(1)} + b_{S,2}^{(1)} \\ c_{S,3} + u_{S,3}^{(2)} a_{S,1}^{(2)} + v_{S,3}^{(2)} b_{S,2}^{(2)} \end{bmatrix}$$

where

$$u_{S,3}^{(2)} = \frac{g_{S,1}^{(2)} \hat{g}_{S,3}^{(1)}}{g_{S,1}^{(1)} \hat{g}_{S,3}^{(2)}}, \qquad v_{S,3}^{(2)} = \frac{h_{S,2}^{(2)} \hat{h}_{S,3}^{(1)}}{h_{S,2}^{(1)} \hat{h}_{S,3}^{(2)}}$$

and where the power and rate are set to

$$\begin{aligned} P_1^{(c)} \doteq P, P_1^{(a)} \doteq P^{\alpha}, P_1^{(b)} \doteq P^{\alpha} \\ r_1^{(c)} = 1 - \alpha, r_1^{(a)} = r_1^{(b)} = \alpha. \end{aligned} \tag{7.15}$$

As before, receiver 1 decodes and removes the common symbols to get

$$\begin{aligned} y'_{S,1} &= h_{S,1}^{(1)} a_{S,1}^{(1)} + h_{S,1}^{(2)} a_{S,1}^{(2)} \\ y'_{S,2} &= h_{S,2}^{(1)} b_{S,2}^{(1)} + h_{S,1}^{(2)} b_{S,2}^{(2)} \\ y'_{S,3} &= h_{S,3}^{(1)} (a_{S,1}^{(1)} + b_{S,2}^{(1)}) + h_{S,3}^{(2)} (u_{S,3}^{(2)} a_{S,1}^{(2)} + v_{S,3}^{(2)} b_{S,2}^{(2)}). \end{aligned} \tag{7.16}$$

which are then linearly combined to get

$$\frac{y'_{S,3}}{h_{S,3}^{(1)}} - \frac{y'_{S,2}}{h_{S,2}^{(1)}} = \underbrace{a_{S,1}^{(1)} + \frac{h_{S,3}^{(2)}}{h_{S,3}^{(1)}} u_{S,3}^{(2)} a_{S,1}^{(2)}}_{P^{\alpha}} + \underbrace{(\frac{h_{S,3}^{(2)}}{h_{S,3}^{(1)}} v_{S,3}^{(2)} - \frac{h_{S,2}^{(2)}}{h_{S,2}^{(1)}}) b_{S,2}^{(2)}}_{P^{0}}$$

where

$$\mathbb{E}\left| (\frac{h_{S,3}^{(2)}}{h_{S,3}^{(1)}} v_{S,3}^{(2)} - \frac{h_{S,2}^{(2)}}{h_{S,2}^{(1)}}) b_{S,2}^{(2)} \right|^2 = \mathbb{E}\left| \frac{\tilde{h}_{S,3}^{(2)} \hat{h}_{S,3}^{(1)} - \tilde{h}_{S,3}^{(1)} \hat{h}_{S,3}^{(1)}}{h_{S,3}^{(1)} \hat{h}_{S,3}^{(1)}} b_{S,2}^{(2)} \right|^2 \doteq P^0$$

Decoding

1) Phase S: Receiver 1 first decodes $c_{S,t}$ by treating all other signals as noise, and then removes $h_{S,t}^{(2)} c_{S,t}$ from $y_{S,t}$. At this point, for each sub-phase, the receiver experiences an equivalent 2×2 MIMO channel of the form (with bounded noise)

$$\begin{bmatrix} y'_{S,1} \\ \frac{y'_{S,3}}{h_{S,3}^{(1)}} - \frac{y'_{S,2}}{h_{S,2}^{(1)}} \end{bmatrix} = \begin{bmatrix} h_{S,1}^{(1)} & h_{S,2}^{(1)} \\ 1 & \frac{h_{S,3}^{(2)}}{h_{S,3}^{(1)}} u_{S,3}^{(2)} \end{bmatrix} \begin{bmatrix} a_{S,1}^{(1)} \\ a_{S,1}^{(2)} \end{bmatrix}$$

allowing receiver 1 to decode $a_{S,1}^{(1)}, a_{S,1}^{(2)}$. Now receiver 1 can go back one phase and reconstruct $\{\bar{i}_{S-1,t}\}_{t=1}^{T_{S-1}}$ using knowledge of the common symbols of this last phase. Similar actions are performed by receiver 2.

2) Phase $s, s = S-1, \cdots, 2$. As before, receiver 1 first decodes $c_{s,t}$. Using the already decoded $\{c_{s+1,t}\}_{t=1}^{T_{s+1}}$, receiver 1 reconstructs $\{\bar{i}_{s,t}, \bar{\theta}_{s,t}\}_{t=1}^{T_s}$, and for each sub-phase, subtracts $\bar{i}_{s,1}$ to get $\sigma_{s,1}$, up to bounded noise. The same receiver also employs the estimate $\bar{\theta}_{s,1}$ as an extra observation. Thus now receiver 1 sees a 4×4 MIMO channel of the form

$$\begin{bmatrix} y'_{s,1} \\ \sigma_{s,1} \\ \sigma_{s,3} \\ \bar{\theta}_{s,1} \end{bmatrix} = \underbrace{\begin{bmatrix} h_{s,1}^{(1)} & h_{s,1}^{(2)} & h_{s,1}^{(1)} & 0 \\ 1 & \frac{h_{s,3}^{(2)}}{h_{s,3}^{(1)}} u_{s,3}^{(2)} & 1 & \frac{h_{s,3}^{(2)}}{h_{s,3}^{(1)}} u_{s,3}^{(2)} \\ -\frac{h_{s,4}^{(1)}}{h_{s,4}^{(2)}} u_{s,4}^{(1)} & 0 & 0 & -1 \\ 0 & \frac{g_{s,3}^{(2)}}{g_{s,3}^{(1)}} u_{s,3}^{(2)} - \frac{g_{s,1}^{(2)}}{g_{s,1}^{(1)}} & 0 & \frac{g_{s,3}^{(2)}}{g_{s,3}^{(1)}} u_{s,3}^{(2)} \end{bmatrix}}_{A} \begin{bmatrix} a_{s,1}^{(1)} \\ a_{s,1}^{(2)} \\ a'^{(1)}_{s,1} \\ a'^{(2)}_{s,3} \end{bmatrix}$$

where one can check that matrix has a full rank almost surely. With the above linear independence established, we now see that we have accumulated two observations of power $P^{2\alpha}$, and two observations of power P^{α}, while at the same time, there are two information symbols of power $P^{2\alpha}$ and of rate 2α and two information symbols of power P^{α} and of rate α. This suffices for receiver 1 to decode $a_{s,1}^{(1)}, a_{s,1}^{(2)}, a'^{(1)}_{s,1}, a'^{(2)}_{s,3}$. The process is the similar for receiver 2.

3) Phase 1*:* Similarly, receiver 1 first reconstructs $\{\bar{i}_{1,1}, \bar{\theta}_{1,1}\}_{t=1}^{T_1}$ from $\{c_{2,t}\}_{t=1}^{T_2}$ for each sub-phase to get a 3×3 MIMO channel of the form

$$\begin{bmatrix} y_{1,1} \\ \frac{y_{1,3}}{h_{1,3}^{(1)}} - \frac{y_{1,2}}{h_{1,2}^{(1)}} - \bar{i}_{1,1} \\ \bar{\theta}_{1,1} \end{bmatrix} = \begin{bmatrix} h_{1,1}^{(1)} & h_{1,1}^{(2)} & h_{1,1}^{(2)} \\ 1 & \frac{h_{1,3}^{(2)}}{h_{1,3}^{(1)}} u_{1,3}^{(2)} & 0 \\ 0 & \frac{g_{1,3}^{(2)}}{g_{1,3}^{(1)}} u_{1,3}^{(2)} - \frac{g_{1,1}^{(2)}}{g_{1,1}^{(1)}} & -\frac{g_{1,1}^{(2)}}{g_{1,1}^{(1)}} \end{bmatrix} \begin{bmatrix} a_{1,1}^{(1)} \\ a_{1,1}^{(2)} \\ a'^{(2)}_{1,1} \end{bmatrix}$$

where obviously the matrix has a full rank almost surely. Therefore, receiver can decode $a_{1,1}^{(1)}, a_{1,1}^{(2)}$ with rate 1 respectively, and $a'^{(2)}_{1,1}$ with rate $1-\alpha$. Receiver 2 acts the same.

DoF calculation

Adding up the private information transmitted over the different phases, we get that

$$\begin{aligned} d_{\sum} &= (\frac{T_1}{3}(6-2\alpha) + \sum_{i=2}^{S-1} \frac{T_s}{4}(12\alpha) + \frac{T_S}{3}4\alpha)/(\sum_{i=1}^{S} T_s) \\ &= 3\alpha + (T_1(2 - \frac{11}{3}\alpha) - T_S\frac{5}{3}\alpha)/(\sum_{i=1}^{S} T_s) \end{aligned}$$

Considering that $0 \leq \mu \leq 1$, for an asymptotically high S, we get

$$d_{\sum} = 3\alpha + \frac{\frac{T_2}{\xi}(2 - \frac{11}{3}\alpha) + \frac{5}{3}T_2\mu^{S-3}\gamma\alpha}{\frac{T_2}{\xi} + T_2(\frac{1}{1-\mu} + \mu^{S-3}(\gamma - \frac{\mu}{1-\mu}))} = \frac{6}{5} + \frac{2\alpha(2-3\alpha)}{5(4-7\alpha)}$$

which proves the result, and which additionally shows that the optimal sum DoF $\frac{4}{3}$ is achievable even with $\alpha = \frac{4}{9}$.

7.3.2 Schemes for XC with imperfect delayed CSIT

We proceed to focus on the achievability with imperfect delayed CSIT.

The phase durations $T_1, T_2, \cdots, T_S$ are chosen to be integers such that

$$\begin{aligned} &T_s = T_{s-1}\xi, \ \forall s \in \{2, 3, \cdots, S-1\}, \\ &T_S = T_{S-1}\gamma = T_1\xi^{S-2}\gamma \end{aligned} \tag{7.17}$$

where $\xi = \frac{2\beta}{3(1-\beta)}, \gamma = \frac{2\beta}{3}$.

Phase 1 Like before, phase 1 consists of $\frac{T_1}{3}$ sub-phases, with each sub-phase consisting of three consecutive time slots. Focus on the first such sub-phase denoted as $(1,1), (1,2), (1,3)$. The rest of the sub-phases will simply be a repetition of this first sub-phase, with each sub-phase corresponding to new information symbols. In this first sub-phase, the transmitted signals are

$$\boldsymbol{x}_{1,1} = \begin{bmatrix} c_{1,1} + a_{1,1}^{(1)} + a'^{(1)}_{1,1} \\ a_{1,1}^{(2)} \end{bmatrix}, \boldsymbol{x}_{1,2} = \begin{bmatrix} c_{1,2} + b_{1,2}^{(1)} + b'^{(1)}_{1,2} \\ b_{1,2}^{(2)} \end{bmatrix}, \boldsymbol{x}_{1,3} = \begin{bmatrix} c_{1,3} + a_{1,1}^{(1)} + b_{1,2}^{(1)} \\ a_{1,1}^{(2)} + b_{1,2}^{(2)} \end{bmatrix} \tag{7.18}$$

where the power and normalized rates are set as

$$\begin{aligned} &P_1^{(c)} \doteq P, P_1^{(a)} \doteq P_1^{(b)} \doteq P_1^{(a')} \doteq P_1^{(b')} \doteq P^{\beta} \\ &r_1^{(c)} = 1 - \beta, r_1^{(a)} = r_1^{(b)} = r_1^{(a')} = r_1^{(b')} = \beta. \end{aligned} \tag{7.19}$$

After the transmission, the received signals at receiver 1 take the form

$$\begin{aligned} y_{1,1} &= h_{1,1}^{(1)}c_{1,1} + h_{1,1}^{(1)}(a_{1,1}^{(1)} + a'^{(1)}_{1,1}) + h_{1,1}^{(2)}a_{1,1}^{(2)} \\ y_{1,2} &= h_{1,2}^{(1)}c_{1,2} + h_{1,2}^{(1)}(b_{1,2}^{(1)} + b'^{(1)}_{1,2}) + h_{1,2}^{(2)}b_{1,2}^{(2)} \\ y_{1,3} &= h_{1,3}^{(1)}c_{1,3} + h_{1,3}^{(1)}(a_{1,1}^{(1)} + b_{1,2}^{(1)}) + h_{1,3}^{(2)}(a_{1,1}^{(2)} + b_{1,2}^{(2)}) \end{aligned} \tag{7.20}$$

where we ignore the Gaussian noise. At this point, we see that user 1 can decode $c_{1,1}, c_{1,2}, c_{1,3}$ from the three received signals with rate $1-\beta$ by treating

all the other signals as noise, respectively. After removal of these common symbols, receiver 1 has $y'_{1,t} = y_{1,t} - h^{(1)}_{1,t} c_{1,t}, t = 1, 2, 3$ where

$$\begin{aligned} y'_{1,1} &= h^{(1)}_{1,1}(a^{(1)}_{1,1} + a'^{(1)}_{1,1}) + h^{(2)}_{1,1} a^{(2)}_{1,1} \\ y'_{1,2} &= h^{(1)}_{1,2}(b^{(1)}_{1,2} + b'^{(1)}_{1,2}) + h^{(2)}_{1,2} b^{(2)}_{1,2} \\ y'_{1,3} &= h^{(1)}_{1,3}(a^{(1)}_{1,1} + b^{(1)}_{1,2}) + h^{(2)}_{1,3}(a^{(2)}_{1,1} + b^{(2)}_{1,2}) \end{aligned} \tag{7.21}$$

Upon the above, receiver 1 removes the unintended symbol $b^{(2)}_{1,2}$ from transmitter 2, using the following linear combination, to get

$$\frac{y'_{1,3}}{h^{(2)}_{1,3}} - \frac{y'_{1,2}}{h^{(2)}_{1,2}} = \underbrace{\frac{h^{(1)}_{1,3}}{h^{(2)}_{1,3}} a^{(1)}_{1,1} + a^{(2)}_{1,1}}_{P^\beta} + \overbrace{\underbrace{(\frac{h^{(1)}_{1,3}}{h^{(2)}_{1,3}} - \frac{h^{(1)}_{1,2}}{h^{(2)}_{1,2}}) b^{(1)}_{1,2} - \frac{h^{(1)}_{1,2}}{h^{(2)}_{1,2}} b'^{(1)}_{1,2}}_{P^\beta}}^{i_{1,1}} \tag{7.22}$$

where we noted the order of the summand's average power of the desired signals and interference, where $i_{1,1}$ denotes the interference from transmitter 1 to receiver 1 during this first sub-phase.

Receiver 2, which acts similarly, now experiences interference $\theta_{1,1}$, where this interference is similarly bounded above by P^β. At the end of this first sub-phase (3 time slots), transmitter 1 uses its partial knowledge of delayed CSIT to reconstruct $\{i_{1,1}, \theta_{1,1}\}$, and to quantize each term to get

$$\bar{i}_{1,1} = i_{1,1} - \tilde{i}_{1,1}, \quad \bar{\theta}_{1,1} = \theta_{1,1} - \tilde{\theta}_{1,1}.$$

We have that $\mathbb{E}(|i_{1,1}|^2) = \mathbb{E}(|\theta_{1,1}|^2) \doteq P^\beta$, we choose a quantization rate that each $\bar{i}_{1,1}, \bar{\theta}_{1,1}$ has $\beta \log P$ bits, thus allowing in turn for $\mathbb{E}(|\tilde{i}_{1,1}|^2) \doteq \mathbb{E}(|\tilde{\theta}_{1,1}|^2) \doteq 1$.

The same procedure is repeated for the rest $\frac{T_1}{3} - 1$ sub-phases, with new information carried by each symbol from the transmitters. Consequently, the accumulated interference $\{\bar{i}_{1,t}\}_{t=1}^{T_1}$ and $\{\bar{\theta}_{1,t}\}_{t=1}^{T_1}$ is of size $\frac{2\beta T_1}{3} \log P$ bits, which will be assigned evenly into the common symbols $\{c_{2,t}\}_{t=1}^{T_2}$ of the second phase. By doing so, each user's own interference can be removed to the noise level (cf. (7.22)), and also it can be served as an extra observation for the other.

2) Phase s, $2 \le s \le S-1$: Phase s ($T_s = \frac{2\beta}{3(1-\beta)} T_{s-1}$) is similar to phase 1, which consists of $\frac{T_s}{3}$ sub-phases, each consisting of 3 consecutive channel uses. The transmitted signal takes the same form as in phase 1 (cf. (7.18)), as well as the corresponding power and rate of the symbols (cf. (7.19)), where the common symbols $\{c_{s,t}\}_{t=1}^{T_s}$ carry the residual interference from phase $s-1$, and all the other symbols carry new private information to the corresponding user.

At the end of the transmission of phase s, the received signals are of the same form as phase 1 (e.g, for user 1, $\{y_{s,t}\}_{t=1}^{T_s}$ is the same as in (7.20)). It

is easy to see that each receiver can decode the common symbols $\{c_{s,t}\}_{t=1}^{T_s}$ by treating interference as noise.

Now we go back to the previous phase $s-1$. With the knowledge of $\{c_{s,t}\}_{t=1}^{T_s}$, user 1, 2 can reconstruct the accumulated estimates

$$\{\bar{i}_{s-1,t}\}_{t=1}^{T_{s-1}}, \{\bar{\theta}_{s-1,t}\}_{t=1}^{T_{s-1}}$$

of all the interference, respectively. Take user 1 for example, it first subtracts $\bar{i}_{s-1,1}$ from $\frac{y'_{s-1,3}}{h^{(2)}_{s-1,3}} - \frac{y'_{s-1,2}}{h^{(2)}_{s-1,2}}$ (cf. (7.22)) up to noise level in the first sub-phase of phase $s-1$, and each sub-phase follows the same course of actions. Then together with $y'_{s,1}$ and the estimates $\bar{\theta}_{s-1,1}$ being an extra observation, it can obtain a 3×3 MIMO channel of the form

$$\begin{bmatrix} y'_{s-1,1} \\ \frac{y'_{s-1,3}}{h^{(2)}_{s-1,3}} - \frac{y'_{s-1,2}}{h^{(2)}_{s-1,2}} - \bar{i}_{s-1,1} \\ \bar{\theta}_{s-1,1} \end{bmatrix} = \begin{bmatrix} h^{(1)}_{s-1,1} & h^{(1)}_{s-1,1} & h^{(2)}_{s-1,1} \\ \frac{h^{(1)}_{s-1,3}}{h^{(2)}_{s-1,3}} & 0 & 1 \\ \frac{g^{(1)}_{s-1,3}}{g^{(2)}_{s-1,3}} - \frac{g^{(1)}_{s-1,1}}{g^{(2)}_{s-1,1}} & -\frac{g^{(1)}_{s-1,1}}{g^{(2)}_{s-1,1}} & 0 \end{bmatrix} \begin{bmatrix} a^{(1)}_{s-1,1} \\ a'^{(1)}_{s-1,1} \\ a^{(2)}_{s-1,1} \end{bmatrix}$$

where obviously the matrix has a full rank almost surely. Therefore, receiver can decode $a^{(1)}_{s-1,1}, a'^{(1)}_{s-1,1}, a^{(2)}_{s-1,1}$ with rate β respectively. Receiver 2 acts the same.

Now we come back to phase s. Like phase 1, we know that each user can remove $\{c_{s,t}\}_{t=1}^{T_s}$ part from the received signals and creates residual interference after linear combination. With partial knowledge of delayed CSIT, the transmitter can reconstruct the interference $\{i_{s,t}\}_{t=1}^{T_s}, \{\theta_{s,t}\}_{t=1}^{T_s}$ with quantized estimates $\{\bar{i}_{s,t}\}_{t=1}^{T_s}, \{\bar{\theta}_{s,t}\}_{t=1}^{T_s}$ of size $\frac{2\beta T_1}{3}\log P$ bits. Finally, the information of $\{\bar{i}_{s,t}\}_{t=1}^{T_s}, \{\bar{\theta}_{s,t}\}_{t=1}^{T_s}$ will be carried by $\{c_{s+1,t}\}_{t=1}^{T_{s+1}}$ sequentially in the next phase $s+1$.

3) Phase S: In this phase, for any $t\in[0,T_S]$, the transmitted signals take the form

$$\boldsymbol{x}_{S,t} = \begin{bmatrix} c_{S,t} \\ 0 \end{bmatrix}$$

where the power and rate are set to

$$P^{(c)} \doteq P, r^{(c)} = 1. \tag{7.23}$$

This phase delivers the residual interference from phase $S-1$, i.e. $\{\bar{i}_{S-1,t}\}_{t=1}^{T_{S-1}}$ and $\{\bar{\theta}_{S-1,t}\}_{t=1}^{T_{S-1}}$. It is easy to see that each user can decode the common symbol with rate 1.

DoF calculation

Adding up the private information transmitted over the different phases, we get that

$$d_{\sum} = \big(\frac{T_1}{3}(3(1-\beta)+6\beta) + \sum_{i=2}^{S-1}\frac{T_s}{3}(6\beta)\big)/\big(\sum_{i=1}^{S} T_s\big)$$
$$= 2\beta + \big(T_1(1-\beta) - T_S(2\beta)\big)/\big(\sum_{i=1}^{S} T_s\big)$$

Considering that $0 \leq \xi \leq 1$, for an asymptotically high S, we get

$$d_{\sum} = 2\beta + \frac{T_1(1-\beta) - T_1\xi^{S-2}(2\beta)}{T_1\frac{1-\xi^{S-1}}{1-\xi} + T_1\xi^{S-2}} = 1 + \frac{\beta}{3}$$

which proves the result, and which additionally shows that the linear optimal sum DoF $\frac{6}{5}$ is achievable even with $\beta = \frac{3}{5}$.

7.4 Conclusions

This work provided analysis and novel schemes for the setting of the two-user SISO X channel with imperfect quality current and delayed CSIT, offering insight on how much delayed and/or current feedback quality suffices to achieve a certain target sum-DoF performance.

Chapter 8

Conclusion and Future work

8.1 Conclusions

In terms of feedback, the work explored the fundamental limits of the cache-aided wireless broadcast channel, identifying the optimal cache-aided DoF within a multiplicative factor of 4. The constructed caching-and-delivery schemes adapted caching and transmission, to the CSIT quality in order to meet these limits. More importantly, the schemes exploited the interesting connections between retrospective transmission schemes which alleviate the effect of the delay in knowing the channel, and coded caching schemes which alleviate the effect of the delay in knowing the content destination. These connections are at the core of the coded caching paradigm, and their applicability can extend to different settings which we plan to explore in the future. Such connections exist because, both delayed feedback, as well as caching, can be used to set up multicasting.

The work advocated that the interplay between coded-caching and CSIT feedback is particularly important because CSIT and coded caching are two powerful tools that are simultaneously synergistic as well as competing, in handling interference. This joint exposition of these two intertwined tools, was motivated directly from the prospect of distributing predicted content 'during the night', in order to alleviate a very large number of CSIT predictions and disseminations per second during the day. What the work shows is that a modest amount of caching can go a long way in removing the burden of having to acquire high-quality timely CSIT. Along the same lines, an interesting way to interpret the results here, is to say that content prediction during the night, allowed for CSIT delays during the day.

In terms of this interplay and tradeoff, what we have seen (Chapters 2,3)

is that there is a certain amount of caching that is needed to take a system that had perfect CSIT ($\alpha = 1, \gamma = 0$), and completely substitute current CSIT with caching ($\gamma > 0, \alpha = 0$), without degrading performance. The conclusion is that $\gamma \approx e^{-1}$ suffices for this. Extending this rationale, we also draw the conclusion that if we allow for a somewhat degraded performance $d = 1/G$ (for some $G > 1$) — which essentially means that a fraction $1/G$ of interference is removed — then we will see an exponential reduction in the required cache size that is needed for this performance, down to a more manageable $\gamma \approx e^{-G}$. This is in contrast to the single-stream case (and all known instances that have been studied to date - even those employing many antennas and full feedback) where the equivalent reduction in γ would be inversely proportional to G, corresponding to a much more demanding $\gamma \approx \frac{1}{G}$. This exponential dependency suggests that, for increasing K, caches that are vanishingly small compared to the file library, can go a long way in removing the burden of having to acquire timely and high-quality CSIT.

Chapters 2-3 also discuss the good match between coded caching and retrospective transmission methods, showing that the latter methods offer a near optimal utilization of the increase in γ as compared to a large class of communication schemes that have similar performance without caching, but which fail to match well the coded caching framework. This suggests that these retrospective methods are a main ingredient in better utilizing caching. In a somewhat converse manner, we show how cache-aided communications can utilize a vanishingly-small portion of D-CSIT compared to traditional D-CSIT schemes, simply because caching helps 'skip' the parts of the schemes that require the highest D-CSIT load. The comparison carefully normalizes the number of full D-CSIT scalars sent, by the coherence period T_c and by the total amount of data delivered. The derived substantial reduction of the cost of D-CSIT, is accompanied by the surprising finding that for $\gamma \geq \frac{1}{10}$ this D-CSIT cost is even less than that of the the very efficient zero forcing precoder, which has to additionally deal with harder-to-obtain current CSIT.

Table 8.1: Summary of results (as K increases to infinity)

	local caching (LC)	single stream (SS)	derived here
$\alpha = 0$	$T_{\text{LC}} \to \infty$	$T_{\text{SS}} \to \frac{1-\gamma}{\gamma}$	$T \to \log(\frac{1}{\gamma})$
	$d_{\text{LC}} = 0$	$d_{\text{SS}} \to \gamma$	$d \to \frac{1-\gamma}{\log(\frac{1}{\gamma})}$
$\alpha > 0$	$T_{\text{LC}} = \frac{1-\gamma}{\alpha}$	N/A	$T \to \frac{(1-\gamma)\log(\frac{1}{\gamma})}{\alpha\log(\frac{1}{\gamma})+(1-\alpha)(1-\gamma)}$
	$d_{\text{LC}} = 0$	N/A	$d \to \alpha + (1-\alpha)\frac{1-\gamma}{\log(\frac{1}{\gamma})}$

Ramifications of a little caching in larger BC systems While implementation of coded caching in large BC systems might currently be problematic for different practical reasons (see for example [58]), the derived near-optimal limits in this large K setting, suggest very substantial gains for small values of γ. One of the most promising elements comes from the fact — as we have mentioned before — that the derived interference removal gains are no longer linear in γ as they were in the single stream case where we saw that the per-user DoF $d_{\text{ss}}(\gamma, \alpha = 0) \to \gamma$, but rather have an exponential element in the sense that any linear decrease in the required fraction d of the interference removed, allows for an exponential reduction in the required γ. So assume that you wanted to half a gap G to (the interference-free) optimal, we would now only need to *additively* increase γ by about $e^{-\frac{G}{2}}$ (i.e., $\gamma \to \gamma + e^{-\frac{G}{2}}$), unlike a linear system that would require an additive increase of γ by about $\frac{1}{G}$ (see Table 8.2)[1].

Table 8.2: Cache size increase $\gamma_1 \to \gamma_2$ needed to half the gap, i.e., needed for improvement $T(\gamma_1) = G \to T(\gamma_2) = \frac{G}{2},\ G \geq 2$ (large K)

local caching	single stream	derived here
$\gamma_1 \approx \gamma_2 \approx 1$	$\gamma_2 - \gamma_1 \approx \frac{1}{G}$	$\gamma_2 - \gamma_1 \approx e^{-\frac{G}{2}}$

It is interesting to note that these conclusions are made in the presence of a small gap to optimal (at most 4), which additionally converges to 2 as K increases and γ decreases to smaller values.

Library design The fact that now, small γ values can theoretically have a substantial impact, may offer food for thought on how we design libraries. In the past, having caches that were very small compared to the library size ($M \ll N$, i.e., γ very small), might not have been interesting (in terms of exploiting receiver-side caches in wireless settings), simply because a small γ carried lesser impact (limited impact with only local caching gains, and impact proportional to γ in single-stream coded caching systems). Now though, one might consider having a much larger library, even if caches are relatively small, because this will allow a frequent applicability (due to the large N) of the now *non-trivial* coded caching gains that are offered even by comparably very small cache sizes.

[1] Note that if there are only local caching gains, the achieved $T = K(1-\gamma)$, approaches $T \to G$ only if $\gamma \to 1$, because $K \gg G$.

A little caching has more impact than a little feedback One could argue that the fact that

$$d(\gamma, \alpha = 0) \to \frac{1-\gamma}{\log(\frac{1}{\gamma})} > d(\gamma = 0, \alpha) \to \alpha$$

implies that caching can be more impactful than 'comparable amounts' of CSIT (i.e., when $\alpha = \gamma$). This conclusion of course suffers from the fact that it is hard to quantify what it means to have comparable amounts of these two different resources. The fact though that now — for small γ, α — α has a linear effect on the above interference removal fraction, while γ has an exponential element in this, may allow with some more confidence the conclusion that for smaller values of γ and α, caching carries more impact. The intuition could be that if we started off with a system that had no current CSIT and no caching ($\alpha = \gamma = 0$), and we had to choose between injecting a little bit of caching (γ small) or a little bit of CSIT feedback (α small), then the exponential gains associated to small values of γ would be good motivation to consider the caching option.

Need to tighten the gaps to optimal The fact that the gains now have an exponential element in them, further accentuates the need to have tight bounds to optimal. This is because — based on the above discussion — the significance of a gap-to-optimal Q, can be proportional to the significance we attribute to a reduction of cache sizes, from $\gamma \approx e^{-1}$ to $\gamma = e^{-Q}$.

This need to tighten multiplicative gaps to optimal in high-f (high file size, i.e., DoF) settings, is further accentuated by the fact that such multiplicative gaps, spread the ambiguity that comes with asymptotic approximations, to all parameters of the system, and not just the parameters of approximation. While having a gap-to-optimal that does not scale with K, certainly offers very valuable information, the larger these gaps are, the more they can hinder many meaningful comparisons in settings of operational significance. Taking as an example the single-stream case, a gap to optimal that can be as high as some $Q \geq 1$ ($\frac{T}{T^*} \leq Q$) does not (directly) allow for simple conclusions of the form $T^*(\gamma_2, \alpha) \leq T^*(\gamma_1, \alpha)$, $\gamma_2 > \gamma_1$ for any K that is less than $K \leq \frac{Q(1-\gamma_2)-(1-\gamma_1)}{(1-\gamma_1)\gamma_2 - Q(1-\gamma_2)\gamma_1}$ (assuming the denominator is positive). This is simply because the gap might be maximal at γ_1 and minimal at γ_2. In any case, in DoF terms, guaranteeing a bounded gap to optimal simply says that *there exists a non-zero capacity slope as log(SNR) increases.* This is of course a non-trivial piece of information, but indeed it should be taken at its own value. Stating that a scheme is 'near optimal' simply because the gap to optimal is non-infinite, can be counter productive, especially when we consider that K is finite.

Chapitre 9

French Summary

9.1 Motivation

Les solutions de communication actuelles ne se basent pas sur les grands réseaux car, à mesure que le nombre d'utilisateurs augmente, ces solutions ne peuvent pas séparer complètement les signaux de tous les utilisateurs. Ceci à son tour peut progressivement laisser à chaque utilisateur de petits débits de communication, et il arrive à un moment où le trafic de données sans fil devrait augmenter de 10 fois en seulement 5 ans [1]. Si les volumes de données continuent à se développer si rapidement, il est prévu que les réseaux sans fil se feront bientôt s'arrêter, compromettant inévitablement les fondements informatifs de la société. Ce dépassement de réseau débordant peut également avoir des conséquences environnementales sévères, car les systèmes actuels nécessiteraient des augmentations exponentielles de la puissance d'émission pour tenir compte des futurs volumes de données ; Les télécommunications ont en fait déjà une empreinte carbone plus élevée que aviation [2]. Malgré l'imminence de cette surcharge, le consensus est qu'il n'existe pas de technologie existante ou actuellement envisagée qui résout cela, même pas avec l'augmentation de la bande passante de la force brute, ni du nombre et de la taille des stations de base.

Le problème à résoudre la surcharge ci-dessus, provient principalement de la nature intrinsèquement en temps réel des communications, où les données doivent être « servies » à temps. Cela implique d'adapter en permanence les communications — en temps réel — aux états rapidement fluctuants d'un grand réseau sans fil. Dans ce contexte, « Feedback » se réfère à l'action consistant à diffuser de grandes quantités d'informations générales (appelées information d'état des canaux, ou CSI) sur les forces instantanées de chaque

chemin de propagation entre les différents noeuds. Ces canaux fluctuent jusqu'à des centaines de fois par seconde, donc à mesure que la taille du réseau augmente, les frais généraux consomment de plus en plus de ressources, ne laissant éventuellement aucune place aux données réelles.

Les limites de la PHY basée sur les feedback La plupart des méthodes de communication utilisent les feedback comme principale ressource. Les feedback servent principalement à séparer les signaux des utilisateurs, donc, plus le processus de rétroaction est amélioré, plus le signal «direction» est efficace, plus la séparation du signal est élevée, moins les interférences sont élevées, plus les taux sont élevés. Le problème est que les feedback consomment toutes les ressources du réseau. Il n'y a pas si longtemps, il a été révélé que, en raison des feedback, de nombreux réglages cellulaires ne pouvaient jamais échapper à du débit petite, indépendamment de toute forme possible de coopération en matière de transmission [3]. Cela constituait un proxy pour les limites de nombreuses architectures, comme dans le MIMO massif (voir l'approche « Grassmann-manifold » [4]), ou même dans des solutions totalement décentralisées telles que les méthodes puissantes d'alignement des interférences [5], où il y a une aggravation polynomiale du problème de rétroaction et une complexité super exponentielle requise pour obtenir un débit non disparaissant. Même si le retour d'informations a été relégué en temps différé (ce qui est beaucoup plus facile à obtenir), cela force encore les taux proches de zéro (cf [6, 7] ainsi que [8,9], etc.). Enfin, même si les réseaux étaient fortement densifiés avec une infrastructure filaire, les blocages d'échantillonnage-complexité et de propagation qui en découlent, peuvent largement dépasser les goulots d'étranglement de rétroaction (voir le recent [10]).

Un moyen clair de voir ce problème du goulet de 'rétroaction' est que dans le simple réglage du grand MIMO, où la performance optimale sans interférence (qui peut être obtenue par simple inversion de canal) suppose que la chaîne (qui change jusqu'à quelques centaines de fois par seconde) est connu du côté de l'émetteur, de manière instantanée, ce qui implique une rétroaction en temps réel sans délai des estimations des canaux qui ont une précision presque parfaite. Comme on l'a soutenu, c'est impossible, et les imperfections CSIT qui en résultent entraînent une détérioration spectaculaire de la performance [4]. Pour voir cela mieux, notez que le canal est accompagné d'une période de cohérence T_C pendant laquelle le canal est à peu près invariant dans le temps et a une largeur de bande de cohérence W_C qui est l'intervalle de fréquence sur lequel la chaîne reste encore fixée. Les grands frais généraux d'acquisition de chaque estimation de la chaîne forcent les degrés de liberté par utilisateur d ($d \approx \frac{\text{Capacity}}{logSNR}$ est la capacité normalisée — voir plus tard) être borne supérieure comme

$$d \leq \frac{\min(K, \frac{T_C W_C}{2})}{K}\big(1 - \frac{\min(K, \frac{T_c W_c}{2})}{T_c W_c}\big) \leq \frac{T_c W_c}{4K} \to 0$$

qui s'évanouit lentement à zéro, car le frais généraux supérieures avec K qui peuvent dépasser la fenêtre de cohérence $T_C W_C$. Cela se manifeste dans une variété de paramètres, y compris celui du duplexage par division de temps (TDD) avec une réciprocité de liaison montante descendante.

L'émergence de « coded caching » Récemment, une solution de couche supérieure a été proposée qui compté sur les données du caching codées par réseau aux dispositifs de réception [11]. La solution proposée a été motivée par le fait que le trafic sans fil est lourdement vidéo ou audio à la demande (plus de 60%), ce qui implique une capacité à prévoir les demandes de données à l'avance (« la nuit précédente »). Cette approche était basée sur la contenu du caching d'une bibliothèque existante de nombreux fichiers populaires. Chaque utilisateur pré-sauvegarderait (sans connaître les demandes du lendemain) une séquence de sous-fichiers soigneusement sélectionnée de la bibliothèque, spécifiquement conçue pour accélérer les communications (du lendemain). Contrairement à ce qui précède, la mémoire a été utilisée pour diffuser des informations sur le côté (contexte), qui peuvent être mieux utilisées par multidiffusion, c'est-à-dire en transmettant des signaux qui doivent être «entendus» par plusieurs. Ceci est exactement le contraire de la séparation assistée par la rétroaction, où en général, les utilisateurs n'entendent que leurs propres signaux. Cette approche prometteuse - qui a été présentée à l'époque pour la chaîne de diffusion d'un seul flux sans erreur (ligne filaire), offre des gains — en termes de degrés de liberté par utilisateur - ce qui peut rester limité : le gain de DoF est resté approximativement égal au rapport entre la taille du cache et la taille de la bibliothèque.

Insight : l'idée derrière l'exploration des feedback et le caching Notre travail cherchera à fusionner ces deux mondes apparemment indépendants du caching et de la PHY avancée assistée par des feedback. Ce travail est une réponse naturelle au besoin apparent d'utiliser conjointement ces deux ressources puissantes dans les communications sans fil : la mémoire et les feedback. Pourquoi le processus de rétroaction de la diffusion de CSI aujourd'hui serait-il si fortement lié au processus de stockage de données préventif hier ? Pour voir cela, rappelez-vous que les feedback et la mémoire sont complémentaires ; la rétroaction sépare les signaux, tandis que la mémoire les rassemble (car les informations latérales permettent la multidiffusion, ce qui est le contraire de la séparation du signal). Cela signifie que lorsque les feedback incomplet entraîne par inadvertance une séparation de signal incomplète, cela renforce paradoxalement l'impact de la multidiffusion assistée par mémoire. Cela tient également à l'inverse, et en substance, chaque élément s'appuie sur les imperfections de l'autre. La deuxième connexion découle du fait que les deux cas sont confrontés à des problèmes parallèles. La façon dont nous cachons mardi, sachant que les demandes de fichiers seront révélées mercredi, est un problème structurel-

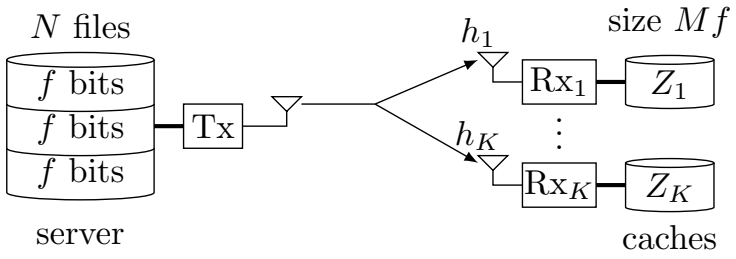

Figure 9.1 – Cache-aided single-stream BC.

lement similaire à la façon dont nous transmettons maintenant, sachant que le « emplacement » (canal) sera révélé plus tard. C'est ce deuxième lien structurel qui permet à la mémoire de faire des feedback en temps réel sans délai pertinents, et c'est la première connexion qui donne une certaine propriété de secours fiables à cette dualité.

Insight : Le pouvoir des micro-insertions préemptives de données dans des paramètres sans fil Les méthodes traditionnelles du caching (préfetching) réduisent principalement le volume du problème pour le lendemain, reflétant l'ancien dicton 'Faites quelque chose aujourd'hui, afin que vous ne deviez pas le faire demain'. Plutôt que de modifier le volume du problème, une approche beaucoup plus puissante consiste à utiliser des éclaboussures de mémoire pour changer la structure du problème. Pour clarifier les choses, établissons une analogie. Imaginez que vous êtes assis à une table où tout le monde parle italien, sauf vous. Vous ne comprenez presque rien, jusqu'à ce qu'un bon traducteur samaritain propose par intermittence de petits conseils (peu de mots sur ce qui est dit). Nous savons tous que cela peut être très utile. Supposons maintenant qu'il existe de nombreux sujets de conversation possibles et de nombreux auditeurs non-italiens, qui ont été enseignés séparément la veille de quelques points sur chaque sujet. Supposons que cette « formation commune » a été conçue, de sorte que le traducteur puisse utiliser des phrases simples qui sont utiles à beaucoup à la fois (peu de mots - en raison du contexte d'offre de formation — plus d'une discussion). Le lendemain, les auditeurs choisissent un sujet qu'ils veulent écouter. Ensuite, le traducteur doit communiquer dans un contexte où la capacité des auditeurs à comprendre, n'est pas stable. Il ne sait pas complètement dans quel état ils sont, mais doit néanmoins compenser les crevasses dans la compréhension, de sorte que, à la fin, il puisse satisfaire la curiosité de tous les auditeurs aussi rapidement que possible. Cela a marqué un parallèle à notre contexte, qui est encore plus impliqué.

Bref aperçu des défis existants du « coded caching »

Considérez un réseau sans fil avec un serveur et K recevant des utilisateurs, en demandant des données d'une bibliothèque de N fichiers possibles, où, pour simplifier, chaque fichier est de longueur f bits (voir Fig. 9.1). La bibliothèque (catalogue) est connue à l'avance, et peut par exemple se composer de films populaires, où chaque fichier peut être environ une minute de l'un des nombreux films « populaires » qui pourraient être demandés le lendemain. Considérons un lien (diffusion) partagé et simplifié sans bruit. Le lien est un flux unique, et a une capacité normalisée f (un fichier par unité-temps normalisée). Servir un utilisateur à la fois, impliquerait un temps requis $T = K$ nécessaire pour garantir l'achèvement de la livraison pour tout K fichier demandé. Cela correspond à un débit (normalisé) par utilisateur de $d = \frac{1}{K}$ (fichiers par unité de temps). Comme K augmente, ce taux par utilisateur disparaît à zéro. Supposons maintenant que chaque utilisateur (chaque récepteur) dispose d'une mémoire de stockage (cache) de capacité égale à $M < N$ files. En mettant un cache à chaque utilisateur 'pendant la nuit précédente' (hors heures de pointe), une fraction $\gamma = \frac{M}{N} < 1$ de chaque fichier signifie que toute demande de 'jour' de la bibliothèque pourrait être traité dans $T = K(1-\gamma)$ qui produit un rendement amélioré de 'jour' de $d \approx \frac{1}{K(1-\gamma)} \to 0$ qui va toujours à zéro à mesure que K augmente. C'est l'approche de prélecture précitée utilisée dans les systèmes actuels.

En général, des schémas de pré-traçage traditionnels basés sur la routage et la popularité produiraient un faible débit normalisé

$$d(\gamma) \to \frac{1}{K\beta'} \to 0$$

où β' ici capture la taille du cache γ, et les popularités du fichier (cf. citeJTLC :15, NA :17). Cette ligne de recherche «préfetching» a produit des résultats attrayants, qui peuvent bien fonctionner dans l'Internet filaire avec des protocoles basés sur le routage, mais pas dans les réseaux sans fil modernes qui sont fondamentalement différents, car leur structure et leur « topologie » fluctuent jusqu'à des centaines de fois par seconde, aussi bien qu'à cause de la manipulation structurelle impliquée des signaux (par exemple, dans les paramètres sans fil, les interférences sont quelque chose à utiliser, plutôt que d'être toujours évité comme une collision). La variante codée évoluée de ceci dans [11], a montré qu'en mettant en cache les données d'une manière spécifique, essentiellement le codage à travers les caches pendant la phase de mise en cache, et à travers différentes données demandées par différents utilisateurs (au lendemain), on pourrait servir Utilisateurs avec une seule transmission de multidiffusion codée par réseau, et atteindre — pendant la période de livraison « jour » — un débit

$$d \to \frac{\gamma}{1-\gamma}$$

qui était directement proportionnel à la taille du cache M, et qui ne s'est pas rapproché de zéro à mesure que K augmentait. Le problème est que, pour des valeurs relativement faibles de γ — Le débit par utilisateur indiqué ci-dessus serait à peu près

$$d \to \frac{\gamma}{1-\gamma} \to \gamma$$

qui est très problématique car γ devrait être microscopiquement petit, dans l'ordre de $\gamma \approx 10^{-4} \to 10^{-2}$ ou même moins ([12]), reflétant la croyance largement répandue que de manière réaliste, les périphériques utilisateurs finaux ne pourront cacher qu'une petite fraction de la bibliothèque, parce qu'une telle bibliothèque doit être très importante pour qu'elle soit demandée assez souvent le lendemain. Plus le nombre de fichiers de bibliothèque N est élevé, plus il est probable que le contenu demandé découlera de cette bibliothèque, et plus souvent les gains de mise en cache codés apparaîtront. Tout autre chose que le microscopiquement petit γ impliquerait soit des caches de taille irréaliste, soit des bibliothèques peu pratiques qui n'aboutissent presque jamais à des gains de mise en cache. L'attente est que $M \ll N \geq K \gg 1$.

Le problème persiste même en présence de considérations de popularité de fichiers réalistes, qui ont joué un rôle central dans les approches de prélecture qui ont précédé la mise en cache codée. Comme nous l'avons vu dans [13] pour une distribution réaliste des popularités de fichiers, les gains, même s'ils dépassaient de loin les gains traditionnels de mise en cache locale de la prélecture, étaient mettant à l'échelle en fait encore en tant que $d \to \Gamma$ pour plus petit γ (voir aussi [14]). En fait, le même problème persistait même en présence d'un puissant groupe de L serveurs/antennes (où $L = \lambda K, \lambda \in (0, 1)$) qui a servi les K utilisateurs sur un réseau sans erreur qui, même en présence d'hypothèses de rétroaction parfaite, a donné un débit (voir [15])

$$d(\gamma) = \lambda + \gamma$$

ce qui signifie que le gain attribué à la mémoire est de nouveau $d(\gamma) - d(\gamma = 0) = \gamma$ simplement parce que le terme $d(\gamma = 0) = \lambda$ (c'est facile à Voir) est uniquement dû à avoir plusieurs serveurs (antennes multiples) avec les feedback en temps réel parfait, dont nous avons discuté avant, n'est pas réaliste pour grand K. Toutes les solutions existantes basées sur la mémoire que nous connaissons ont eu un impact restreint pour les petits γ. Comme nous le montrons, cette limitation $d(\gamma) - d(\gamma = 0) \approx \gamma$, n'est pas fondamentale.

Un autre ingrédient essentiel est la topologie et la façon dont cela affecte l'utilisation de la mémoire. Encore une fois, cela est presque entièrement inexploré (avec une exception notable dans [16]) (En l'absence de mémoire, un certain travail a été fait sur la compréhension de la relation entre rétroaction et topologie, cf. [17–19] ainsi que [9]).

Permettez-nous d'explorer davantage l'état de l'art de certaines solutions aidées par la rétroaction récemment (en mettant l'accent sur les délais de

rétroaction et les considérations de qualité) qui sont plus proches de l'esprit de cette thèse, puis des solutions assistées par la mémoire apparues au cours des 2-3 dernières années.

9.2 Autre état de l'art

Notre travail ici — qui considère l'application conjointe de CSIT et la mise en cache codée comme deux outils puissants et interconnectés pour éliminer les fractions importantes de l'interférence dans les réseaux sans fil multi-utilisateurs — se construit sur de nouveaux travaux sur la mise en cache codée et sur les nombreux résultats sur les ramifications de la performance de la rapidité de rétroaction et de la qualité. Commençons par un très bref aperçu des travaux antérieurs dans le domaine de la gestion de l'interférence aidée par la rétroaction (en se concentrant principalement sur la BC et sur le contexte de ce travail ici), qui sera ensuite suivi d'un aperçu plus complet de la mise en cache- Des efforts connexes, se concentrant presque exclusivement sur les développements entourant la mise en cache codée.

Etat de l'art sur la gestion des interférences aidée par rétroaction (pas de caching)

Le rôle de la rétroaction dans l'élimination des interférences multi-utilisateurs implique naturellement de nombreuses directions et facettes. Une direction particulière de la recherche dans ce domaine a cherché à dériver des limites de DoF qui reflètent les effets de la qualité et de la rapidité des feedback, et qui souvent — sous différents contextes et sous différentes formes — employent des schémas de communication qui fonctionnent rétrospectivement pour atténuer l'effet Des retards de CSIT et des imperfections. De tels travaux incluent le travail de Maddah-Ali and Tse [6] qui a montré comment les communications rétrospectives sur un canal de décoloration rapide, peuvent rendre l'utilité CSIT complètement obsolète, ainsi que d'autres travaux ultérieurs [7, 8, 20–28] qui s'appuient sur cela pour intégrer davantage les considérations de qualité CSIT. D'autres travaux qui se rapportent à l'approche ici, se trouvent dans [29–32].

Etat de l'art sur la gestion de l'interférence assistée par cache

Les avantages de la mise en cache codée sur la réduction des interférences et l'amélioration des performances sont venus plus récemment avec le travail précité de Maddah-Ali et Niesen dans [11] qui a considéré un système de mise en cache où un serveur est connecté à plusieurs utilisateurs via un lien partagé, Et a conçu une nouvelle méthode de mise en cache et de livraison qui offre conjointement un gain de multidiffusion qui contribue à atténuer la charge de la liaison et qui a démontré qu'il y avait un écart optimal d'au maximum 12. Ce travail a ensuite été généralisé dans différents paramètres, notamment le

paramètre De différentes tailles de cache pour lesquelles Wang et al. In [37] a développé une variante de l'algorithme dans [11] qui permet d'obtenir un écart maximal de 12 à partir de la l'optimale de la théorie de l'information. D'autres extensions ont inclus le travail dans [38] par Maddah-Ali et Niesen qui ont considéré le paramétrage de la mise en cache décentralisée où la performance obtenue était comparable à celle du cas central [11], malgré le manque De coordination dans le placement de contenu. Pour le même réglage unique en un seul flux de [11], le travail de Ji et al. dans [39] considéré comme un scénario où les utilisateurs font plusieurs requêtes et ont proposé un schéma qui a un écart optimal à moins de 18. Encore une fois pour le paramètre [11], le travail de Ghasemi et Ramamoorthy dans [40], ont dérivé des limites extérieures (inférieures) plus serrées qui s'améliorent en fonction des limites existantes, et l'ont fait en refaisant le problème lié comme l'une des étiquettes optimales des feuilles d'un arbre dirigé. D'autres travaux peuvent être trouvés dans [41] où Wang et al. ont exploré le lien intéressant entre la mise en cache et le codage source distribué avec des information sur le côté. Des conclusions intéressantes sont également tirées dans le travail d'Ajaykrishnan et al. dans [42], qui a révélé que l'efficacité de la mise en cache dans le cas d'un seul flux est diminuée lorsque N approche et dépasse K^2. En outre, Amiri et al. Amélioré la performance originale de Maddah-Ali et Niesen lorsque le système a plus d'utilisateurs que de fichiers ($K \geq N$), tandis que dans [43] Amiri et Gündü a réussi à améliorer les performances (retard réduit, jusqu'à un maximum de 2), pour des valeurs spécifiques de M.

En évitant les liens sans erreur d'un seul flux, différents travaux ont considéré l'utilisation de la mise en cache codée dans différents réseaux sans fil, sans toutefois prendre en compte la qualité de rétroaction CSIT. Par exemple, les travaux de Huang et al. dans [44], considéré un BC de décoloration sans fil assistée par le cache où chaque utilisateur connaît une qualité de lien différente et a proposé un schéma de communication sous-optimal basé sur la répartition du temps et de la fréquence et de l'attribution de la puissance et de la bande passante et qui a été évalué à l'aide de simulations numériques pour éventuellement montrer que le débit produit diminue à mesure que le nombre d'utilisateurs augmente. Les travaux ultérieurs de Timo et Wigger dans [16] ont été considérés un canal de diffusion d'effacement et ont exploré comment l'efficacité du système aidé par le cache peut s'améliorer en employant des tailles inégales de cache qui sont des fonctions des différentes qualités de canal. Un autre travail peut être trouvé dans [45] où Maddah-Ali et Niesen ont étudié la chaîne d'interférence sans fil où chaque émetteur possède un cache local et a montré des avantages distincts de la mise en cache codée qui résulte du fait que le contenu se chevauche à Les émetteurs permettent une annulation efficace des interférences.

Un travail différent a également examiné les effets de la mise en cache dans différents paradigmes de chaînes non classiques. L'un des premiers travaux de ce genre axés sur les paramètres pratiques du réseau sans fil, comprend le

travail de Golrezaei et al. dans [46], qui a considéré un paramètre cellulaire de liaison descendante où la station de base est assistée par des noeuds auxiliaires qui forment conjointement un réseau de mise en cache distribué sans fil (pas de mise en cache codée) où les fichiers populaires sont mis en cache, ce qui entraîne une augmentation substantielle du nombre d'utilisateurs autorisé par autant que $400 - 500\%$. Dans un contexte quelque peu apparenté, le travail dans [47] par Perabathini et al. ont accentué les gains d'efficacité énergétique de la mise en cache. Des travaux intéressants peuvent également être trouvés dans [48] et dans [49] sur l'utilisation de la géométrie stochastique et du codage réseau pour la modélisation de caches et périphériques de stockage sans fil, ainsi que dans [50, 51] que Explorez les considérations de couche supérieure relatives à la mise en cache.

D'autres travaux de Ji et al. dans [52] dérivent les limites des réseaux de mise en cache appelés combinés dans lesquels une source est connectée à plusieurs nœuds utilisateur via une couche de noeuds de relais, de sorte que chaque nœud d'utilisateur avec mise en cache est connecté à un sous-ensemble distinct de Les nœuds de relais. Un travail supplémentaire peut également être trouvé dans [53] où Niesen et al. considéré comme un réseau assisté par le cache où chaque nœud est situé au hasard dans un carré, et il demande un message disponible dans différents caches répartis autour du carré. D'autres travaux connexes sur la mise en cache peuvent être trouvés dans [39, 54–58].

Les travaux qui combinent des considérations de mise en cache et de rétroaction dans les réseaux sans fil ne sont que récemment lancés. Une référence qui les combine, se trouve dans [59] où Deghel et al. considéré comme un MIMO canal d'interférence (IC) avec des caches sur les émetteurs. Dans ce réglage, chaque fois que les données demandées résident dans les caches pré-remplies, la charge de transfert de données de la liaison backhaul est atténuée, permettant ainsi que ces liens soient plutôt utilisés pour échanger CSIT qui supporte l'alignement des interférences. Un travail simultané encore plus récent peut être trouvé dans [60] où Ghorbel et al. ont étudié la capacité du canal d'effacement de paquets de diffusion activé par le cache avec des retours ACK/NACK. Dans ce contexte, Ghorbel et al. Astucieusement montré — d'une manière intéressante également en utilisant un algorithme de type rétrospectif, par Gatzianas et al. Dans [61] — comment les feedback peuvent améliorer les performances en informant l'émetteur lors de la renvoi des paquets qui ne sont pas reçus par l'utilisateur final et qui sont reçus par des utilisateurs involontaires, ce qui permet des opportunités de multidiffusion.

Le premier travail qui considère l'interaction réelle entre la mise en cache codée et la qualité CSIT, peut être trouvé dans [62] qui a considéré le problème plus facile de la manière dont la performance optimisée par cache (avec la mise en cache codée) peut être obtenue avec CSIT de qualité réduite.

Measure of Performance

La mesure de la performance ici sera la durée T (normalisée) — dans les créneaux horaires, par fichier servi par utilisateur — nécessaire pour compléter le processus de livraison, *pour toute demande*. Notez qu'il s'agit d'une mesure « pire cas » (correspondant au cas où chaque utilisateur exige un fichier distinct). Ce T ne doit pas être perçu comme une mesure de latence traditionnelle, mais plutôt comme mesure de débit [1], simplement parce que de la normalisation par la quantité d'informations envoyées.

De manière équivalente, lorsque cela est significatif, nous considérons également le *degrés de liberté assistés par cache par utilisateur* (DoF assisté par cache) qui est simplement [2]

$$d = \frac{1-\gamma}{T} \in [0,1] \tag{9.1}$$

qui est en effet une mesure de débit [3] de la phase de livraison et qui n'inclut pas les avantages d'avoir un contenu déjà disponible aux récepteurs (gain de mise en cache local) et se concentre plutôt sur la capture L'effet de la rétroaction et la mise en cache codée, dans le traitement des interférences.

Modèle de canal de diffusion assisté par le cache

Nous considérerons principalement le réglage de la multiple-input single-output (MISO) BC symétrique où un émetteur équipé des K antennes, communique aux K utilisateurs avec une seule antenne. L'émetteur a accès à une bibliothèque de N fichiers distincts $W_1, W_2, \cdots, W_N$, chaque taille $|W_n| = f$ bits. Chaque utilisateur $k \in 1, 2, ..., K$ a un cache Z_k avec la taille $|Z_k| = Mf$ (bits), et cette taille prend la forme normalisée

$$\gamma \triangleq \frac{M}{N}$$

La taille cumulative prend également la forme

$$\Gamma \triangleq \frac{KM}{N} = K\gamma$$

ce qui signifie simplement que la somme des tailles des caches sur tous les utilisateurs est *Gamma* fois le volume de la bibliothèque de N -file.

La communication comporte deux phases, la phase de placement et la phase de livraison. Au cours de la première phase (heures creuses), les caches $\{Z_k\}_{k=1}^{K}$

1. En fait, dans [11], la même mesure est appelée *rate*.
2. Nous notons que Kd est simplement le gain de codage $K(1-\gamma)/T$ qui est souvent utilisé pour quantifier le gain de la mise en cache codée.
3. Par définition du DoF, ce débit serait maintenant — dans le paramètre SNR élevé d'intérêt — comme $d\log_2(P)$ bits ($d\log_2(SNR)$ bits) du contenu résolu par utilisation de la chaîne (phase de livraison)

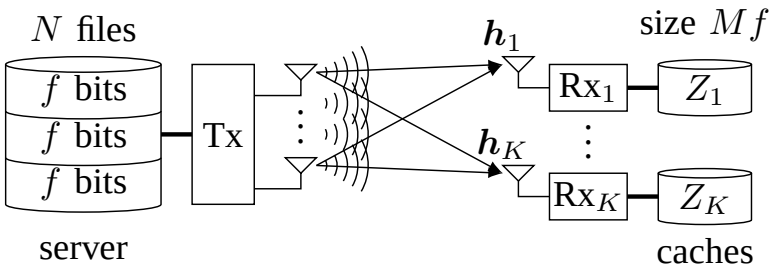

FIGURE 9.2 – Cache-aided K-user MISO BC.

aux utilisateurs sont pré-remplis avec les informations provenant des N files $\{W_n\}_{n=1}^N$. Au cours de la deuxième phase, la transmission commence lorsque chaque utilisateur k demande un seul fichier W_{R_k} hors de la bibliothèque.

Dans ce réglage, les signaux reçus à chaque utilisateur k prennent la forme

$$y_k = \boldsymbol{h}_k^T \boldsymbol{x} + z_k, \quad k = 1, \ldots, K$$

où $\boldsymbol{x} \in \mathbb{C}^{K\times 1}$ désigne le vecteur transmis satisfaisant une contrainte de puissance $\mathbb{E}(||\boldsymbol{x}||^2) \leq P$, où $\boldsymbol{h}_k \in \mathbb{C}^{K\times 1}$ désigne les coefficients de décoloration vectorielle du canal de l'utilisateur k et où z_k représente le bruit AWGN de l'unité au récepteur k.

À la fin de la communication, chaque utilisateur récepteur k combine les signaux reçus y_k — accumulés pendant la phase de livraison — avec les informations disponibles dans leur cache respectif Z_k, pour reconstituer leur souhaité fichier W_{R_k}.

Modèle de rétroaction de type CSIT

La communication se déroule également en présence d'informations d'état de canal sur l'émetteur. La rétroaction du type CSIT est cruciale dans la manipulation des interférences et peut donc réduire considérablement la durée résultante T de la phase de livraison. Ce CSIT est généralement de qualité imparfaite, car il est difficile à obtenir de manière rapide et fiable. Dans le régime d'intérêt SNR (haute puissance P), cette qualité CSIT actuelle — chaque fois que cela est supposé être disponible — sera concisément représentée sous la forme de l'exposant de qualité normalisé [20] (voir aussi [8])

$$\alpha := -\lim_{P\to\infty} \frac{\log \mathbb{E}[||\boldsymbol{h}_k - \hat{\boldsymbol{h}}_k||^2]}{\log P}, \; k \in \{1, \ldots, K\} \tag{9.2}$$

où $\boldsymbol{h}_k - \hat{\boldsymbol{h}}_k$ désigne l'erreur d'estimation entre l'estimation CSIT actuelle $\hat{\boldsymbol{h}}_k$ et la chaîne parfait $\boldsymbol{h}_k$. La gamme d'intérêt [4] est $\alpha \in [0, 1]$. Dans certains

4. Dans le régime SNR élevé d'intérêt ici, $\alpha = 0$ correspond à n'avoir essentiellement aucun CSIT actuel (cf. [63]), tout en ayant $\alpha = 1$ correspond (encore dans le régime SNR élevé) au CSIT parfait et immédiatement disponible.

cas (Chapitres 2 et 3), nous supposerons également la disponibilité du CSIT retardé (comme par exemple [6], ainsi que dans une variété d'œuvres suivantes [8, 20–27] ainsi que [64–66]) où maintenant les estimations retardées de n'importe quel canal, peuvent être reçues sans erreur mais avec un retard arbitraire, même si ce délai rend ce CSIT complètement obsolète.

Intuition : feedback mitigés Ce modèle CSI mixte (partiel CSIT et CSIT retardé) capture bien différents réglages réalistes qui peuvent impliquer des corrélations de canaux et une capacité à améliorer CSI à mesure que le temps progresse. Ce même *CSI modèle est particulièrement bien adapté à notre réglage de mise en cache ici, car il reflète explicitement deux ingrédients clés qui, comme nous le verrons, sont profondément liés avec la mise en cache codée ; à savoir la qualité de la rétroaction (qui révélera un compromis entre la mémoire et la rétroaction) et la rapidité de rétroaction (qui introduit un aspect non linéaire du problème, qui à son tour peut être traduit en un drame — non linéaire — augmentation dans l'impact de la mise en cache).*

9.3 Résumé détaillé des contributions

Cette thèse se concentre sur la mise en cache codée et cherche à comprendre comment la mise en cache codée peut être combinée de manière significative avec différents ingrédients du réseau sans fil tels que les feedback, les antennes multiples et même la topologie. Dans cette section, nous résumons toutes les contributions de la thèse.

Chapitre 2 : Les gains synergiques de la mise en cache codée et des retards de CSIT retardés La première partie du travail explore les gains synergiques entre la mise en cache codée et lesfeedback retardés du CSIT. Ici, nous considérons le MISO (BC) canal de diffusion sans fil à K -user ($K \leq N$) avec évanouissement aléatoire et CSIT retardé ($\alpha = 0$), et identifiez la performance optimale des degrés de liberté assistée par le cache dans un facteur de 4. Pour $H_n \triangleq \sum_{i=1}^{n} \frac{1}{i}$ désignant le numéro harmonique, le premier résultat indique que

$$T = H_K - H_{K\gamma}$$

est réalisable et a un écart-pour-optimal qui est inférieur à 4, pour tous les paramètres de problème (pour tous K, γ).

En présence de l'approximation logarithmique bien connue $H_n \approx \log(n)$ (qui devient serré lorsque K augmente) alors le T ci-dessus prend la forme

$$T = \log(\frac{1}{\gamma})$$

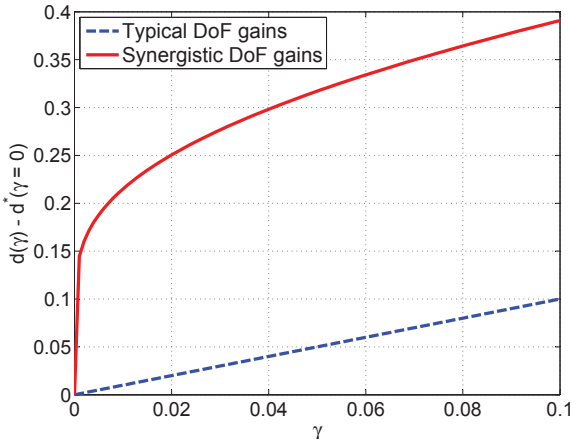

FIGURE 9.3 – Gain typique $d(\gamma) - d^*(\gamma = 0)$ attribué uniquement à la mise en cache codée (ligne pointillée) par rapport aux gains synergiques dérivés ici. Plot tient pour K élevé, et les gains principaux apparaissent pour des valeurs plus faibles de γ.

et le DoF correspondant par utilisateur prend la forme

$$d(\gamma) = \frac{1-\gamma}{\log(\frac{1}{\gamma})}. \tag{9.3}$$

Ce que nous voyons, c'est que — pour des valeurs plus importantes de K — le gain correspondant qui est directement attribué à la mise en cache

$$d(\gamma) - d^*(\gamma = 0) \to \frac{1-\gamma}{\log(\frac{1}{\gamma})} > \gamma, \ \forall \gamma \in (0, 1]$$

peut considérablement dépasser le gain typique de mise en cache codée (par utilisateur DoF) γ. Le gain apparaît très fortement à des valeurs inférieures de γ, où nous voyons que la dérivée — lorsqu'elle est évaluée à $\gamma = 1/K$ — prend la forme

$$\frac{\delta\big(d(\gamma) - d(\gamma = 0)\big)}{\delta\gamma}\Big|_{\gamma=\frac{1}{K}} \approx \frac{K}{\log^2 K}$$

révélant un boost substantiel de DoF aux premiers stades de γ. Ceux-ci peuvent être comparés aux gains linéaires avant ce travail (voir la figure Fig.9.3), où la dérivée est constante $\frac{\delta\big(d(\gamma)-d(\gamma=0)\big)}{\delta\gamma} = \frac{\delta\big(\gamma\big)}{\delta\gamma} = 1, \quad \forall\gamma.$

Dans le même régime, ces gains impliquent en fait un effet exponentiel (plutôt que linéaire) de la mise en cache codée, en ce sens que maintenant, un

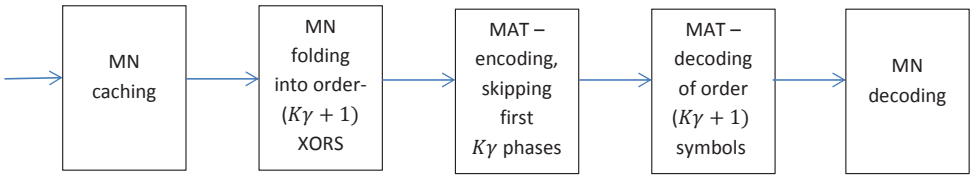

Figure 9.4 – Composition de base du schéma. Le « codage / décodage MAT » correspond au schéma dans [6], tandis que 'MN cache / pliage' correspond au schéma dans [11].

modeste $\gamma = e^{-G}$ peut offrir un très satisfaisant

$$d(\gamma = e^{-G}) \approx \frac{1}{G} \tag{9.4}$$

qui est seulement un facteur G de l'optimal sans interférence (sans cache) $d = 1$.

Par exemple, stocker sur chaque périphérique uniquement un *millième* de ce qui est considéré comme un contenu 'populaire' ($\gamma \approx 10^{-3}$), nous abordons l'optimisation sans interférence dans un facteur de $ln(10^3) \approx 7$ (par utilisateur DoF de $1/7$), pour n'importe quel nombre d'utilisateurs.

Intuition sur les schémas :
L'idée clé derrière le schéma principal est que l'algorithme de mise en cache crée un problème de livraison multi-destination qui est identique à celui qui est résolu efficacement par les dernières étapes du schéma MAT (voir Fig.9.4). Essentiellement, la mise en cache nous permet de sauter les premières $K\gamma$ phases (c.-à-d. sauter les premières phases) du schéma MAT, qui ont la durée la plus longue (la phase i a une durée de $1/i$) . Cela nous donne une idée de la raison pour laquelle l'impact des petits caches (petit γ) est important ; Même les petits caches peuvent supprimer une grande partie de la durée de la communication.

Lors du décodage MAT, nous procédons simplement au décodage du coded caching basé sur l'algorithme dans [11].

Preuve de la borne extérieure :
Notre objectif est de réduire la durée T, qui garantit la livraison de K différents fichiers à K utilisateurs, via un MISO canal de diffusion avec CSIT retardé et en présence de K caches, chacun de taille Mf. Laissez T_2 être la durée nécessaire pour résoudre le réglage plus simple où nous voulons servir $s \leq K$ différents fichiers à s utilisateurs, encore une fois en présence de leurs propres caches. Naturellement $T_2 \leq T$ car nous ignorons l'interférence des utilisateurs $K - s$ restants (dont les demandes sont ignorées). Maintenant, laissez T_3 ($T_3 \leq T_2$), soit la durée nécessaire pour résoudre le même problème, sauf

que maintenant tous les s caches sont fusionnés, et chacun des s les utilisateurs ont accès à tous s Caches. Nous choisissons de répéter cette dernière expérience $\lfloor \frac{N}{s} \rfloor$ fois, couvrant ainsi une durée totale de $T_3 \lfloor \frac{N}{s} \rfloor$. À ce stade, nous transférons au réglage équivalent de s -user MISO BC avec CSIT retardé et un lien de multidiffusion aux informations latérales aux récepteurs, de capacité d_m (fichiers par intervalle de temps). En supposant que dans ce dernier contexte, le décodage se produit à la fin de la communication, et une fois que nous

$$d_m T_3 \lfloor \frac{N}{s} \rfloor = sM \tag{9.5}$$

(Ce qui garantit que les informations latérales du lien latéral, tout au long du processus de communication, correspondent à la quantité maximale d'informations dans les caches), nous avons alors cela

$$T_3 \lfloor \frac{N}{s} \rfloor d'(d_m) \geq \lfloor \frac{N}{s} \rfloor s \tag{9.6}$$

où $d'(d_m)$ est une borne supérieure de la MISO BC de s utilisateurs avec CSIT retardé et la liaison latérale. Utilisation de la limite

$$d'(d_m) = \frac{s}{H_s}(1 + d_m)$$

de [67] et en appliquant (9.5), on obtient

$$d'(d_m) = \frac{s}{H_s}(1 + \frac{sM}{T_3 \lfloor \frac{N}{s} \rfloor})$$

et nous obtenons donc

$$T_3 \lfloor \frac{N}{s} \rfloor \frac{s}{H_s}(1 + \frac{sM}{T_3 \lfloor \frac{N}{s} \rfloor}) \geq \lfloor \frac{N}{s} \rfloor s \tag{9.7}$$

ce qui signifie que

$$T_3 \geq H_s - \frac{sM}{\lfloor \frac{N}{s} \rfloor} \tag{9.8}$$

ce qui implique que le optimal T^*, pour le problème origina de s utilisateurs, est limité comme

$$T^* \geq T_3 \geq H_s - \frac{sM}{\lfloor \frac{N}{s} \rfloor}. \tag{9.9}$$

La maximisation sur tous les s, donne

$$T^* \geq \max_{s \in \{1,\ldots,\min\{\lfloor \frac{N}{M} \rfloor, K\}\}} H_s - \frac{sM}{\lfloor \frac{N}{s} \rfloor}. \tag{9.10}$$

Ces résultats ont été présentés en

— Jingjing Zhang, Petros Elia, "The Synergistic Gains of Coded Caching and Delayed Feedback", arXiv :1604.06531, April 2016.

— Jingjing Zhang, Petros Elia, "Fundamental Limits of Cache-Aided Wireless BC : Interplay of Coded-Caching and CSIT Feedback", *IEEE Transactions on Information Theory*, to appear 2017.

Chapitre 3 : compromis entre mémoire et CSIT actuel Dans le chapitre 3, nous présentons sous forme fermée (pour le même MISO BC de K utilisateurs, avec un émetteur de K anthennas desservant des K utilisateurs avec des caches) l'interaction préalablement inexplorée entre la mise en cache et la qualité des feedback. Plus précisément, nous verrons que le T réalisable et son DoF $d(\gamma, \alpha)$ correspondant, prennent le formulaire

$$T = \frac{(1-\gamma)(H_K - H_{K\gamma})}{\alpha(H_K - H_{K\gamma}) + (1-\alpha)(1-\gamma)}$$

$$d(\gamma, \alpha) = \alpha + (1-\alpha)\frac{1-\gamma}{H_K - H_{K\gamma}}$$

et sont tous les deux prouvés être au maximum un facteur de 4 de l'optimal. Sous l'approximation logarithmique, le T dérivé prend la forme

$$T(\gamma, \alpha) = \frac{(1-\gamma)\log(\frac{1}{\gamma})}{\alpha\log(\frac{1}{\gamma}) + (1-\alpha)(1-\gamma)}$$

et le DoG dérivé prend la forme

$$d(\gamma, \alpha) = \alpha + (1-\alpha)\frac{1-\gamma}{\log\frac{1}{\gamma}}.$$

Si nous devions nous concentrer sur le plus grand réglage de K (pour des interprétations croquantes), ce qui suggère ci-dessus est que le CSIT actuel offre une augmentation initiale de DoF de $d^*(\gamma = 0, \alpha) = \alpha$ (cf. [68]), qui est ensuite complété par un gain DoF

$$d(\gamma, \alpha) - d^*(\gamma = 0, \alpha) \to (1-\alpha)\frac{1-\gamma}{\log(\frac{1}{\gamma})}$$

attribué à la synergie entre CSIT retardé et mise en cache.

L'expression de DoF indique clairement un certain troquer entre α et γ, d'où l'on peut imaginer échanger l'un pour l'autre. Par conséquent, pour capturer les économies de rétroaction (aidées par la mémoire), considérons

$$\delta_\alpha(\gamma) \triangleq \arg\min_{\alpha'}\{\alpha' : (1-\gamma)T^*(\gamma = 0, \alpha') \leq T(\gamma, \alpha)\} - \alpha \tag{9.11}$$

décrivant *la réduction de CSIT en raison de la mise en cache* (de α', jusqu'à un fonctionnement opérationnel α, sans perte de performance) et pour lequel

nous montrons que la mise en cache peut réaliser une réduction CSIT qui, sous l'approximation logarithmique, prend la forme

$$\delta_\alpha(\gamma,\alpha) = (1-\alpha)d(\gamma,\alpha=0) = (1-\alpha)\frac{1-\gamma}{\log(\frac{1}{\gamma})}.$$

En outre, un simple calcul — que nous montrons ici pour le régime plus grand de K qui donne une meilleure compréhension — peut nous dire que

$$\gamma_\alpha^{'} \triangleq \arg\min_{\gamma^{'}}\{\gamma^{'} : d(\gamma^{'},\alpha=0) \geq d^*(\gamma=0,\alpha)\} = e^{-1/\alpha}$$

ce qui signifie que $\gamma_\alpha^{'} = e^{-1/\alpha}$ suffit d'atteindre — en conjonction avec CSIT retardé — la DoF performance optimale $d^*(\gamma=0,\alpha)$ associée à un système avec CSIT retardé et CSIT actuel de la qualité α. Cela nous indique la quantité de mémoire nécessaire pour remplacer le CSIT actue par CSIT retardé sans perte de performance, ce qui peut être interprété comme le coût de la mémoire pour acquérir la capacité de « tampon CSI ».

Intuition sur les schémas :
Pour offrir une certaine intuition sur les schémas, nous notons que la partie de mise en cache est modifiée de [11] à *'fold '* (combiner linéairement) les données des différents utilisateurs en blocs multicouches, d'une manière telle que l'algorithme de transmission subséquent (algorithme Q-MAT, voir [68]) (spécifiquement Les dernières $K-\eta_\alpha$ ($\eta_\alpha \in \{\Gamma,\ldots,K-1\}$) phases de l'algorithme QMAT) peuvent délivrer efficacement ces blocs. De manière équivalente, les algorithmes sont étalonnés de sorte que l'algorithme de mise en cache crée un problème de livraison des destinations multiples qui soit identique à celui résolu efficacement par les dernières étapes du schéma de communication du QMAT type.

Ensuite, l'intuition est que lorsque α augmente, nous pouvons avoir plus de données privées, ce qui signifie qu'il y a moins à mettre en cache, ce qui signifie que la mise en cache peut avoir une redondance plus élevée, ce qui implique des XOR d'ordre supérieur, ce qui signifie que nous pouvons multicast à plus d'utilisateurs à la fois, ce qui signifie que nous pouvons sauter plus de phases de QMAT. L'intensité de l'impact de petites valeurs de γ se rapporte au fait que les premières phases de QMAT sont les plus longues. Donc, alors qu'un petit γ ne peut que sauter quelques phases, il saute les plus longs, ce qui permet de réduire considérablement le délai (voir Fig.9.4).

Intuition sur la limite extérieure :
La preuve de la limite inférieure sur T commence comme le cas d'avoir seulement retardé le CSIT ; D'abord, supposons seulement $s \leq K$ utilisateurs qui partagent des caches et qui ne sont pas entravés par le reste, puis répétez cette $\lfloor \frac{N}{s} \rfloor$ fois. Ensuite, nous avons dû dériver de nouvelles limites extérieures pour

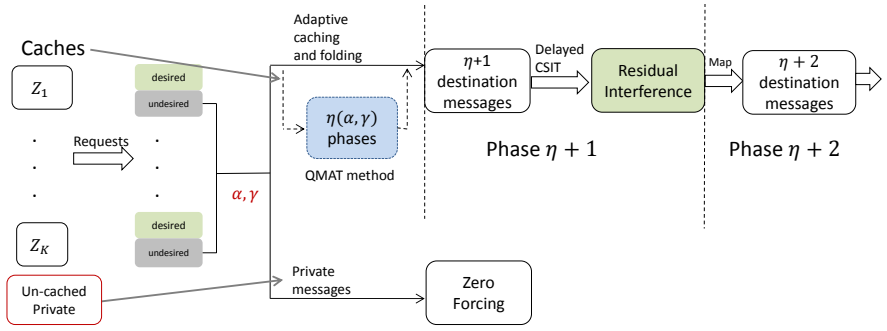

Figure 9.5 – Système de communication rétrospective assisté par le cache.

le MISO BC de s utilisateurs avec CSIT mixte (la nouveauté ici devait tenir compte de la rétroaction actuelle de qualité imparfaite) et avec une présence supplémentaire d'un lien parallèle du de s débit total sMf bits . Cette limite utilise une séquence d'inégalités fondées sur l'entropie. La preuve continue alors comme dans le cas de $\alpha = 0$. La limite inférieure T^* prend la forme

$$T^*(\gamma,\alpha) \geq \max_{s\in\{1,\ldots,\lfloor \frac{N}{M} \rfloor\}} \frac{1}{(H_s\alpha + 1 - \alpha)}(H_s - \frac{Ms}{\lfloor \frac{N}{s} \rfloor}). \tag{9.12}$$

Les résultats peuvent être trouvés dans

— Jingjing Zhang, Petros Elia, "Fundamental Limits of Cache-Aided Wireless BC : Interplay of Coded-Caching and CSIT Feedback", in *Proc. 54th Annual Allerton Conf. Communication, Control and Computing (Allerton'16)*, Illinois, USA, October 2016.

— Jingjing Zhang, Petros Elia, "Fundamental Limits of Cache-Aided Wireless BC : Interplay of Coded-Caching and CSIT Feedback", *IEEE Transactions on Information Theory*, to appear 2017.

Voir également

— Paul de Kerret, David Gesbert, Jingjing Zhang, and Petros Elia, "Optimally bridging the gap from delayed to perfect CSIT in the K-user MISO BC", in *Proc. of IEEE Information Theory Workshop (ITW'16)*, Cambridge, UK, September 2016. (long version, arXiv :1604.01653).

— Paul de Kerret, David Gesbert, Jingjing Zhang, and Petros Elia, "Optimal DoF of the K-User broadcast channel with delayed and imperfect current CSIT", *IEEE Transactions on Information Theory*, submitted 2016. (arXiv :1604.01653).

Chapitre 4 : mise en cache codée assistée par rétroaction avec de très petits caches Le chapitre 4 explore la mise en cache codée assistée

par rétroaction pour le même MISO BC symétrique, mais en mettant l'accent sur des caches très petites, en mettant l'accent sur le cas où la taille du cache cumulatif est inférieure à la taille de la bibliothèque (c.-à-d. $KM \leq N$, $\Gamma \triangleq K\gamma \leq 1$). Ce qui suit identifie, jusqu'à un facteur de 4, le optimal T^*, pour tous $\Gamma \in [0,1]$. Nous utilisons l'expression

$$\alpha_{b,\eta} = \frac{\eta - \Gamma}{\Gamma(H_K - H_\eta - 1) + \eta}, \ \eta = 1, \ldots, K-1. \tag{9.13}$$

Le résultat nous indique que pour $KM \leq N$ ($\Gamma \leq 1$) et pour $N \geq K$, alors pour $\eta = 1, \ldots, K-2$,

$$T = \begin{cases} \frac{H_K - \Gamma}{1-\alpha+\alpha H_K}, & 0 \leq \alpha < \alpha_{b,1} \\ \frac{(K-\Gamma)(H_K - H_\eta)}{(K-\eta)+\alpha(\eta+K(H_K-H_\eta-1))}, & \alpha_{b,\eta} \leq \alpha < \alpha_{b,\eta+1} \\ 1-\gamma, & \frac{K-1-\Gamma}{(K-1)(1-\gamma)} \leq \alpha \leq 1 \end{cases} \tag{9.14}$$

est réalisable et a un écart d'optimisation qui est inférieur à 4 ($\frac{T}{T^*} < 4$ pour tous α, K. Pour $\alpha \geq \frac{K-1-\Gamma}{(K-1)(1-\gamma)}$, T est optimal.

En l'absence de CSIT actuel ($\alpha = 0$), à nouveau pour $K\gamma < 1$, le résultat nous indique que

$$T = H_K - \Gamma$$

est réalisable et a un écart d'optimisation qui est inférieur à 4.

Directement à partir de la performance réalisable (9.14) et de la définition d'économie de CSIT (9.11), nous pouvons avoir les réductions de CSIT (de $\alpha + \delta(\gamma, \alpha)$ à la α opérationnelle) qui peut être réalisé en raison de la mise en cache codée, sans perte de performance. Dans la (K, M, N, α) BC aidée par caching avec $\Gamma \leq 1$, alors

$$\delta(\gamma, \alpha) = \begin{cases} \frac{\gamma(K-H_K)}{H_K - K\gamma}(\alpha + \frac{1}{H_K - 1}), & 0 \leq \alpha < \alpha_{b,1} \\ \frac{(1-\alpha)(KH_\eta - \eta H_K)}{KH_{\eta+1}(H_K-1)}, & \alpha_{b,\eta} \leq \alpha < \alpha_{b,\eta+1} \\ 1-\alpha, & \alpha \geq \frac{K(1-\gamma)-1}{(K-1)(1-\gamma)}. \end{cases} \tag{9.15}$$

Le dernier cas de l'équation ci-dessus montre comment, en présence de la mise en cache, nous n'avons pas besoin d'acquérir une qualité CSIT supérieure à $\alpha = \frac{K(1-\gamma)-1}{(K-1)(1-\gamma)}$.

Sur les schémas :
Ici, le travail propose de nouveaux schémas qui stimulent l'impact des petits caches, car ils parviennent à relever le défi supplémentaire d'avoir une partie du contenu de la bibliothèque non mis en cache, ce qui nous oblige à changer dynamiquement la redondance de la mise en cache pour compenser cela.

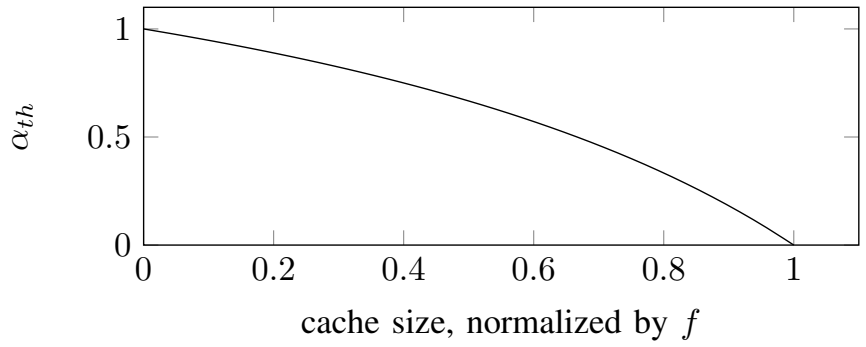

Figure 9.6 – α_{th} obligatoire pour obtenir le optimal $T^*(M)$ dans le MISO BC assisté par cache avec $K = N = 2$.

Sur la bordure extérieure :
La limite inférieure de T s'inspire directement de celle du chapitre 3 qui a été dérivée pour tous N, K, M, γ, et qui peut donc être appliquée ici pour la plage $\gamma \in [0, \frac{1}{K}]$.

Les résultats ont été publiés en

— Jingjing Zhang, Petros Elia, "Feedback-Aided Coded Caching for the MISO BC with Small Caches", to appear in *Proc. of IEEE International Conference on Communications (ICC'17)*, Paris, France, May 2017. (long version, arXiv :1606.05396).

Chapitre 5 : Échange des feedback avec la mémoire : pas de CSIT retardé Dans le chapitre 5, nous considérons le même MISO BC comme précédemment, sauf que les commentaires retardés sont supprimés. Dans ce réglage, et pour $N = K$, la valeur optimale $T^* = 1 - \frac{M}{N}$ sans interférence peut être obtenue avec un α qui ne doit pas être plus grand que (Voir Fig. 9.6 for $N = K = 2$).

$$\alpha_{th} = \frac{N - 1 - M}{\frac{M}{K} + (N - 1 - M)}.$$

Sur les schémas :
Ceci a été obtenu en combinant la mise en cache avec une méthode de diffusion par répartition des taux ; Une approche qui non seulement a amélioré les performances, mais aussi réduit le besoin de CSIT, en ce sens que la DoF performance optimale (sans interférence) associée à la mise en cache et au CSIT parfait a été réalisée avec un CSIT de qualité réduite.

Sur la bordure extérieure :

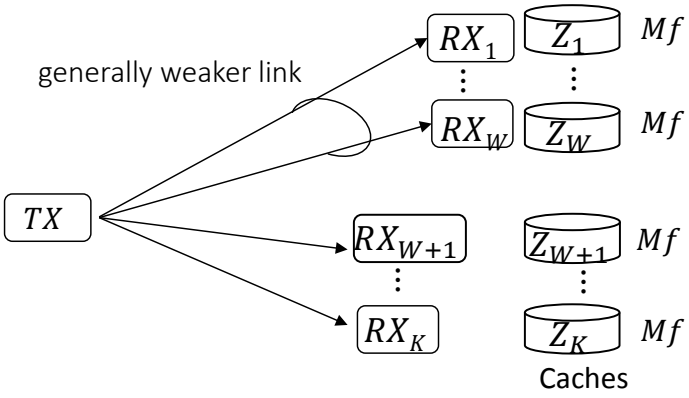

FIGURE 9.7 – MISO BC de K utilisateurs assisté par cache

La seule borne extérieure utilisée est que $T \geq 1 - \gamma$ qui est la borne simple après avoir supprimé toutes les interférences.

Les résultats ont été publiés en

— Jingjing Zhang, Felix Engelmann, and Petros Elia, "Coded caching for reducing CSIT-feedback in wireless communications", in *Proc. 53rd Annual Allerton Conf. Communication, Control and Computing (Allerton'15)*, Illinois, USA, October 2015.

Chapitre 6 : Mise en cache codée sans fil : une perspective topologique Dans le chapitre 6, nous nous éloignons des feedback et considérons l'aspect de la topologie dans un SISO BC de base. La topologie ici signifie simplement que certains liens sont plus forts que d'autres. Cette asymétrie peut être une responsabilité, comme nous commentons ci-dessous, mais cela peut aussi être une bénédiction ; après tout, on peut facilement « cacher » les interférences dans la direction des canaux faibles. Dans ce contexte, la thèse explore les performances de la mise en cache codée dans un réglage sans fil de SISO BC où certains utilisateurs ont des capacités de liaison plus élevées que d'autres. Tous les utilisateurs ont la même taille de cache. En mettant l'accent sur un modèle topologique binaire et fixe où les liens forts ont une capacité normalisée fixe 1 et où les liens faibles ont la capacité normalisée réduit $\tau < 1$, nous identifions — en fonction de la taille du cache et τ — la performance optimale du débit, dans un facteur au maximum de 8. Le schéma de transmission qui réalise cette performance, utilise une forme simple d'amélioration de l'interférence et exploite la propriété que les liens faibles atténuent les interférences, ce qui permet aux taux de multidiffusion de rester élevés Même lorsqu'ils impliquent des utilisateurs faibles. Pour tout K, W, γ, τ, où W est le nombre d'utilisateurs « faibles », le $T(\tau)$ réalisable (ce qui est au maximum 8 fois par rapport à l'optimum théorique) prend la forme

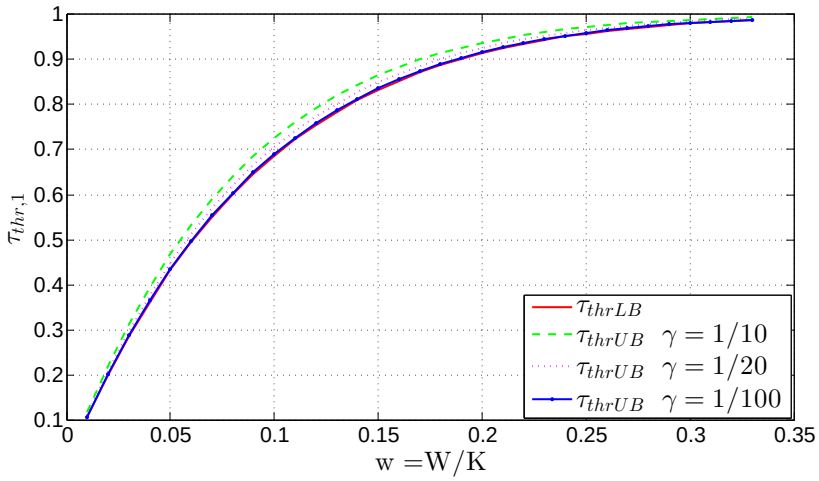

Figure 9.8 – $\tau_{thrLB} = 1-(1-w)^{g_{max}}$ désigne la limite inférieure de τ_{thr}, alors que $\tau_{thrUB} = 1-(1-w-\frac{w\gamma}{1-\gamma})^{g_{max}}$ désigne la limite supérieure.

$$T(\tau) = \begin{cases} \frac{T(W)}{\tau}, & 0 \leq \tau < \bar{\tau}_{thr} \\ \min\{T(K-W)+T(W), \frac{\tau_{thr}T(K)}{\tau}\}, & \bar{\tau}_{thr} \leq \tau \leq \tau_{thr} \\ T(K), & \tau_{thr} < \tau \leq 1 \end{cases}$$

À partir de la dernière partie de l'expression ci-dessus, nous voyons que cette approche améliore les effets négatifs de la topologie inégale en multidiffusion, permettant maintenant à tous les utilisateurs d'atteindre la performance optimale $T(K)$ associée à $\tau = 1$, même si τ est approximativement aussi bas que

$$\tau \geq 1-(1-w)^g$$

où g est le gain de mise en cache codée, et où $w = W/K$ est la fraction d'utilisateurs faible (voir Fig.9.8). Cela conduit à la conclusion intéressante que, pour la multidiffusion codée, les utilisateurs faibles n'ont pas besoin de réduire les performances de tous les utilisateurs, mais au contraire, dans une certaine mesure, les utilisateurs puissants peuvent élever les performances des utilisateurs faibles sans aucune pénalité performance.

Example 1 *($K = 500, W = 50, \gamma = \frac{1}{50}$) Directement de ce qui précède, nous voyons cela*

$$T = \begin{cases} \frac{24.5}{\tau}, & 0 \leq \tau < 0.36 \\ \min\{68.6, \frac{30.7}{\tau}\}, & 0.36 \leq \tau \leq 0.69 \\ T(K) = 44.5, & 0.69 < \tau \leq 1 \end{cases} \tag{9.16}$$

ce qui signifie que, avec un dixième des utilisateurs faible, aussi longtemps que $\tau \geq 0.69$*, il n'y a pas de dégradation de performance due à des liens à capacité réduite et chaque utilisateur reçoit son fichier avec un délai* $T(K) = \frac{K(1-\gamma)}{1+K\gamma} = 44.5$ *associé à* $\tau = 1$.

Sur les schémas :
Pour ce faire, nous avons utilisé l'amélioration des interférences, en effectuant un codage de superposition au niveau de la puissance et l'annulation des interférences successive, où les utilisateurs forts traitent d'abord leurs propres signaux comme bruit pour décoder les signaux des utilisateurs faibles, puis décodent le leur. Notez que, si nous avions simplement envoyé les signaux de multidiffusion séquentiellement au débit maximum autorisé (selon la destination), ce seuil τ aurait été 1 (et donc toute réduction de capacité, même pour $W = 1$ utilisateur, Aurait eu un coût en termes de performance globale).

Sur les limites extérieures :
Pour abaisser la limite de T, nous procédons à l'idée d'envisager un cas « plus facile » d'avoir seulement des utilisateurs faibles de W qui ne sont pas entravés par les utilisateurs restants. Maintenant, pour ce système plus petit avec des utilisateurs de $K = W$, nous exploitons le jeu de coupe lié dans [11], où chaque utilisateur a un lien de capacité τ. Ce faisant, nous pouvons obtenir une limite inférieure de T après une simple normalisation (division) par τ. Pour $\tau \geq \bar{\tau}_{th}$, nous avons établi une limite inférieure plus serrée sur T, qui est bornée par le système d'avoir des K utilisateurs forts (c.-à-d. $\tau = 1$) de [11] en raison de la capacité limitée des utilisateurs faibles.

Les résultats ont été publiés en

— Jingjing Zhang, Petros Elia, "Wireless Coded Caching : A Topological Perspective", submitted to *IEEE International Symposium on Information Theory (ISIT'17)*, Aachen, Germany, June 2017. (long version, arXiv :1606.08253).

Chapitre 7 : Le SISO XC avec CSIT de qualité imparfaite Dans le chapitre 7, nous présentons un résultat qui n'implique pas la mise en cache. Cette partie explore les limites DoF de la chaîne SISO X (deux utilisateurs) avec un CSIT de qualité imparfaite (voir Fig.9.9), et cela montre que la même performance DoF optimale — précédemment associée à une parfaite - CSIT de qualité actuelle - peut en fait être atteint avec CSIT actuel qui est d'une qualité imparfaite. Le travail montre également que la performance DoF précédemment associée au CSIT retardé de qualité parfaite, peut en fait être réalisée en présence d'un CSIT retardé de qualité imparfait. Ceux-ci découlent de la limite inférieure de somme-DoF présentée qui relie l'écart — en fonction de la qualité du CSIT retardé — entre les cas d'absence de rétroaction et de rétro-

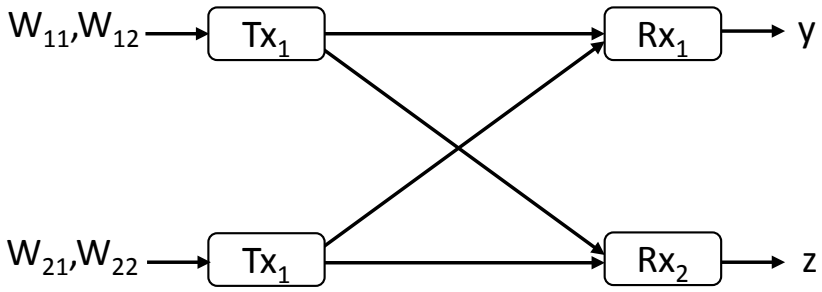

Figure 9.9 – Système de communication rétrospective assisté par le cache.

action retardée, puis une autre borne qui relie l'écart DoF — en fonction de la qualité du CSIT actuel — entre CSIT retardé et parfait.

Sur les schémas :
Les limites intérieures sont basées sur de nouveaux schémas de précodage qui sont présentés ici et qui utilisent un courant de qualité imparfait et / ou une rétroaction retardée pour aligner les interférences dans l'espace et dans le temps.

Plus précisément, nous montrons que pour le XC à deux utilisateurs avec CSIT retardé de qualité parfaite, et avec CSIT actuel imperfect de l'exposant de qualité α, la somme optimale DoF est inférieure à

$$d_\Sigma \geq \min\left(\frac{4}{3}, \frac{6}{5} + \frac{2\alpha(2-3\alpha)}{5(4-7\alpha)}\right).$$

Par conséquent, la somme optimale DoF $d_\Sigma = \frac{4}{3}$ peut être obtenue avec une CSIT actuelle imparfaite de qualité qui ne doit pas dépasser $\alpha = \frac{4}{9}$.

En outre, nous montrons que pour le même XC à deux utilisateurs sans CSIT actuel et avec CSIT retardé imparfait de l'exposant de qualité β, la somme optimale DoF est inférieure à

$$d_\Sigma \geq \min\left(\frac{6}{5}, 1 + \frac{\beta}{3}\right)$$

et, par conséquent, la somme (linéaire) optimale-DoF $d_\Sigma = \frac{6}{5}$, précédemment associée à une rétroaction différée de qualité parfaite, peut en fait être réalisée avec un CSIT retardé de qualité imparfait Qualité qui ne doit pas dépasser $\beta = \frac{3}{5}$.

Les résultats ont été publiés en

— Jingjing Zhang, Dirk TM Slock, and Petros Elia, "Achieving the DoF limits of the SISO X channel with imperfect-quality CSIT", in *Proc. of IEEE International Symposium on Information Theory (ISIT'15)*, Hong Kong, China, June 2015.

Chapitre 8 : Conclusions Le chapitre 8 présente quelques conclusions, résumant certains des résultats, en discutant certaines ramifications possibles de la quantité de mise en cache modérée dans la conception de BC systèmes et de bibliothèques de fichiers de plus grande envergure, certaines réflexions sur le fait que la mise en cache est une ressource plus impactante que les CSIT feedback et une petite discussion sur La nécessité d'établir des limites extérieures plus strictes en matière d'information pour le cas des réseaux sans fil assistés par le retour et le cache.

Bibliography

[1] M. A. Maddah-Ali and D. N. C. Tse, "Completely stale transmitter channel state information is still very useful," *IEEE Trans. Inf. Theory*, vol. 58, no. 7, pp. 4418 – 4431, Jul. 2012.

[2] M. Maddah-Ali and U. Niesen, "Fundamental limits of caching," *IEEE Trans. Inf. Theory*, vol. 60, no. 5, pp. 2856–2867, May 2014.

[3] Cisco, "Cisco visual networking index: Global mobile data traffic forecast update 2014âĂŞ2019," *White Paper*, 2015. [Online]. Available: http://goo.gl/tZ6QMk

[4] A. Fehske, G. Fettweis, J. Malmodin, and G. Biczok, "The global footprint of mobile communications: The ecological and economic perspective," *IEEE Communications Magazine*, vol. 49, no. 8, pp. 55–62, August 2011.

[5] A. Lozano, R. W. Heath, and J. G. Andrews, "Fundamental limits of cooperation," *IEEE Trans. Inf. Theory*, vol. 59, no. 9, pp. 5213–5226, Sept 2013.

[6] L. Zheng and D. N. C. Tse, "Communication on the grassmann manifold: a geometric approach to the noncoherent multiple-antenna channel," *IEEE Trans. Inf. Theory*, vol. 48, no. 2, pp. 359–383, Feb 2002.

[7] V. R. Cadambe and S. A. Jafar, "Interference alignment and degrees of freedom of the K-user interference channel," *IEEE Trans. Inf. Theory*, vol. 54, no. 8, pp. 3425 – 3441, Aug. 2008.

[8] R. Tandon, S. A. Jafar, S. Shamai, and H. V. Poor, "On the synergistic benefits of alternating CSIT for the MISO broadcast channel," *IEEE Trans. Inf. Theory*, vol. 59, no. 7, pp. 4106 – 4128, Jul. 2013.

[9] J. Chen and P. Elia, "Toward the performance vs. feedback tradeoff for the two-user MISO broadcast channel," *IEEE Trans. Inf. Theory*, vol. 59, no. 12, pp. 8336–8356, Dec. 2013.

[10] J. Chen, P. Elia, and S. A. Jafar, "On the two-user MISO broadcast channel with alternating CSIT: A topological perspective," *IEEE Transactions on Information Theory*, vol. 61, no. 8, pp. 4345–4366, Aug 2015.

[11] J. G. Andrews, X. Zhang, G. D. Durgin, and A. K. Gupta, "Are we approaching the fundamental limits of wireless network densification?" *IEEE Communications Magazine*, vol. 54, no. 10, pp. 184–190, October 2016.

[12] M. Ji, A. M. Tulino, J. Llorca, and G. Caire, "Caching-aided coded multicasting with multiple random requests," *CoRR*, vol. abs/1511.07542, 2015. [Online]. Available: http://arxiv.org/abs/1511.07542

[13] U. Niesen and M. A. Maddah-Ali, "Coded caching with nonuniform demands," *IEEE Trans. Inf. Theory*, vol. 63, no. 2, pp. 1146–1158, Feb 2017.

[14] S.-E. Elayoubi and J. Roberts, "Performance and cost effectiveness of caching in mobile access networks," in *Proc. of the 2nd International Conference on Information-Centric Networking*, 2015.

[15] S. P. Shariatpanahi, S. A. Motahari, and B. H. Khalaj, "Multi-server coded caching," *IEEE Trans. Inf. Theory*, vol. 62, no. 12, pp. 7253–7271, Dec 2016.

[16] R. Timo and M. Wigger, "Joint cache-channel coding over erasure broadcast channels," in *2015 International Symposium on Wireless Communication Systems (ISWCS)*, Aug 2015, pp. 201–205.

[17] R. H. Etkin, D. N. C. Tse, and H. Wang, "Gaussian interference channel capacity to within one bit," *IEEE Trans. Inf. Theory*, vol. 54, no. 12, pp. 5534–5562, Dec 2008.

[18] C. S. Vaze, S. Karmakar, and M. K. Varanasi, "On the generalized degrees of freedom region of the MIMO interference channel with no CSIT," in *Proc. IEEE Int. Symp. Information Theory (ISIT)*, July 2011, pp. 757–761.

[19] S. Gherekhloo, A. Chaaban, and A. Sezgin, "Topological interference management with alternating connectivity: The wyner-type three user interference channel," *CoRR*, vol. abs/1310.2385, 2013. [Online]. Available: http://arxiv.org/abs/1310.2385

[20] S. Yang, M. Kobayashi, D. Gesbert, and X. Yi, "Degrees of freedom of time correlated MISO broadcast channel with delayed CSIT," *IEEE Trans. Inf. Theory*, vol. 59, no. 1, pp. 315–328, Jan. 2013.

[21] T. Gou and S. Jafar, "Optimal use of current and outdated channel state information: Degrees of freedom of the MISO BC with mixed CSIT," *IEEE Communications Letters*, vol. 16, no. 7, pp. 1084 – 1087, Jul. 2012.

[22] J. Chen and P. Elia, "Degrees-of-freedom region of the MISO broadcast channel with general mixed-CSIT," in *Proc. Information Theory and Applications Workshop (ITA)*, Feb. 2013.

[23] P. de Kerret, X. Yi, and D. Gesbert, "On the degrees of freedom of the K-user time correlated broadcast channel with delayed CSIT," Jan. 2013, available on arXiv:1301.2138.

[24] J. Chen, S. Yang, and P. Elia, "On the fundamental feedback-vs-performance tradeoff over the MISO-BC with imperfect and delayed CSIT," in *Proc. IEEE Int. Symp. Information Theory (ISIT)*, Jul. 2013.

[25] C. Vaze and M. Varanasi, "The degree-of-freedom regions of MIMO broadcast, interference, and cognitive radio channels with no CSIT," *IEEE Trans. Inf. Theory*, vol. 58, no. 8, pp. 5254 – 5374, Aug. 2012.

[26] N. Lee and R. W. Heath Jr., "Not too delayed CSIT achieves the optimal degrees of freedom," in *Proc. Allerton Conf. Communication, Control and Computing*, Oct. 2012.

[27] C. Hao and B. Clerckx, "Imperfect and unmatched CSIT is still useful for the frequency correlated MISO broadcast channel," in *Proc. IEEE Int. Conf. Communications (ICC)*, Budapest, Hungary, Jun. 2013.

[28] G. Caire, N. Jindal, and S. Shamai, "On the required accuracy of transmitter channel state information in multiple antenna broadcast channels," in *Asilomar Conference on Signals, Systems and Computers*, Asilomar, CA, Nov. 2007.

[29] G. Caire, N. Jindal, M. Kobayashi, and N. Ravindran, "Multiuser MIMO achievable rates with downlink training and channel state feedback," *IEEE Trans. Inf. Theory*, vol. 56, no. 6, pp. 2845 – 2866, Jun. 2010.

[30] N. Jindal, "MIMO broadcast channels with finite-rate feedback," *IEEE Trans. Inf. Theory*, vol. 52, no. 11, pp. 5045 – 5060, Nov. 2006.

[31] O. Shayevitz and M. Wigger, "On the capacity of the discrete memoryless broadcast channel with feedback," *IEEE Trans. Inf. Theory*, vol. 59, no. 3, pp. 1329–1345, March 2013.

[32] Y. Wu and M. Wigger, "Coding schemes with rate-limited feedback that improve over the no feedback capacity for a large class of broadcast channels," *IEEE Transactions on Information Theory*, vol. 62, no. 4, pp. 2009–2033, April 2016.

[33] M. R. Korupolu, C. G. Plaxton, and R. Rajaraman, "Placement algorithms for hierarchical cooperative caching," in *Proc. ACM-SIAM SODA*, Jan. 1999, pp. 586–595.

[34] B.-J. Ko and D. Rubenstein, "Distributed self-stabilizing placement of replicated resources in emerging networks," *IEEE/ACM Trans. Netw.*, vol. 13, no. 3, pp. 476–487, Jun. 2005.

[35] Y. Birk and T. Kol, "Coding on demand by an informed source (ISCOD) for efficient broadcast of different supplemental data to caching clients," *IEEE Trans. Inf. Theory*, vol. 52, no. 6, pp. 2825–2830, Jun. 2006.

[36] S. Borst, V. Gupta, and A. Walid, "Distributed caching algorithms for content distribution networks," in *INFOCOM, 2010 Proceedings IEEE*, Mar. 2010, pp. 1–9.

[37] S. Wang, W. Li, X. Tian, and H. Liu, "Fundamental limits of heterogenous cache," *CoRR*, vol. abs/1504.01123, 2015.

[38] M. A. Maddah-Ali and U. Niesen, "Decentralized caching attains order-optimal memory-rate tradeoff," *CoRR*, vol. abs/1301.5848, 2013. [Online]. Available: http://arxiv.org/abs/1301.5848

[39] M. Ji, A. M. Tulino, J. Llorca, and G. Caire, "Order optimal coded delivery and caching: Multiple groupcast index coding," *CoRR*, vol. abs/1402.4572, 2014. [Online]. Available: http://arxiv.org/abs/1402.4572

[40] H. Ghasemi and A. Ramamoorthy, "Improved lower bounds for coded caching," *CoRR*, vol. abs/1501.06003, 2015. [Online]. Available: http://arxiv.org/abs/1501.06003

[41] C. Wang, S. H. Lim, and M. Gastpar, "Information-theoretic caching: Sequential coding for computing," *CoRR*, vol. abs/1504.00553, 2015. [Online]. Available: http://arxiv.org/abs/1504.00553

[42] A. N., N. S. Prem, V. M. Prabhakaran, and R. Vaze, "Critical database size for effective caching," *CoRR*, vol. abs/1501.02549, 2015. [Online]. Available: http://arxiv.org/abs/1501.02549

[43] M. M. Amiri and D. Gündüz, "Fundamental limits of coded caching: Improved delivery rate-cache capacity tradeoff," *IEEE Trans. Commun.*, vol. 65, no. 2, pp. 806–815, Feb 2017.

[44] W. Huang, S. Wang, L. Ding, F. Yang, and W. Zhang, "The performance analysis of coded cache in wireless fading channel," *CoRR*, vol. abs/1504.01452, 2015. [Online]. Available: http://arxiv.org/abs/1504.01452

[45] M. A. Maddah-Ali and U. Niesen, "Cache-aided interference channels," in *Proc. IEEE Int. Symp. Information Theory (ISIT)*, Jun. 2015.

[46] N. Golrezaei, K. Shanmugam, A. Dimakis, A. Molisch, and G. Caire, "Femtocaching: Wireless video content delivery through distributed caching helpers," in *Proc. IEEE INFOCOM*, Mar. 2012.

[47] B. Perabathini, E. Bastug, M. Kountouris, M. Debbah, and A. Conte, "Caching at the edge: a green perspective for 5G networks," *CoRR*, vol. abs/1503.05365, 2015. [Online]. Available: http://arxiv.org/abs/1503.05365

[48] E. Altman, K. Avrachenkov, and J. Goseling, "Coding for caches in the plane," *CoRR*, vol. abs/1309.0604, 2013. [Online]. Available: http://arxiv.org/abs/1309.0604

[49] K. Avrachenkov, X. Bai, and J. Goseling, "Optimization of caching devices with geometric constraints," *CoRR*, vol. abs/1602.03635, 2016. [Online]. Available: http://arxiv.org/abs/1602.03635

[50] V. Sourlas, L. Gkatzikis, P. Flegkas, and L. Tassiulas, "Distributed cache management in information-centric networks," *IEEE Transactions on Network and Service Management*, vol. 10, no. 3, pp. 286–299, September 2013.

[51] V. Sourlas, P. Flegkas, P. Georgatsos, and L. Tassiulas, "Cache-aware traffic engineering in information-centric networks," in *2014 IEEE 19th International Workshop on Computer Aided Modeling and Design of Communication Links and Networks (CAMAD)*, Dec 2014, pp. 295–299.

[52] M. Ji, M. F. Wong, A. Tulino, J. Llorca, G. Caire, M. Effros, and M. Langberg, "On the fundamental limits of caching in combination networks," in *Signal Processing Advances in Wireless Communications (SPAWC)*, Jun. 2015.

[53] U. Niesen, D. Shah, and G. W. Wornell, "Caching in wireless networks," *IEEE Trans. Inf. Theory*, vol. 58, no. 10, pp. 6524–6540, Oct 2012.

[54] E. Bastug, M. Bennis, and M. Debbah, "A transfer learning approach for cache-enabled wireless networks," *CoRR*, vol. abs/1503.05448, 2015. [Online]. Available: http://arxiv.org/abs/1503.05448

[55] A. F. Molisch, G. Caire, D. Ott, J. R. Foerster, D. Bethanabhotla, and M. Ji, "Caching eliminates the wireless bottleneck in video-aware wireless networks," *CoRR*, vol. abs/1405.5864, 2014. [Online]. Available: http://arxiv.org/abs/1405.5864

[56] J. Hachem, N. Karamchandani, and S. N. Diggavi, "Coded caching for heterogeneous wireless networks with multi-level access," *CoRR*, vol. abs/1404.6560, 2014. [Online]. Available: http://arxiv.org/abs/1404.6560

[57] J. Hachem, N. Karamchandani, and S. Diggavi, "Effect of number of users in multi-level coded caching," in *Proc. IEEE Int. Symp. Information Theory (ISIT)*, Hong-Kong, China, 2015.

[58] K. Shanmugam, M. Ji, A. M. Tulino, J. Llorca, and A. G. Dimakis., "Finite-length analysis of caching-aided coded multicasting," *IEEE Trans. Inf. Theory*, vol. 62, no. 10, pp. 5524–5537, Oct 2016.

[59] M. Deghel, E. Bastug, M. Assaad, and M. Debbah, "On the benefits of edge caching for MIMO interference alignment," in *Signal Processing Advances in Wireless Communications (SPAWC)*, Jun. 2015.

[60] A. Ghorbel, M. Kobayashi, and S. Yang, "Cache-enabled broadcast packet erasure channels with state feedback," *CoRR*, vol. abs/1509.02074, 2015. [Online]. Available: http://arxiv.org/abs/1509.02074

[61] M. Gatzianas, L. Georgiadis, and L. Tassiulas, "Multiuser broadcast erasure channel with feedback — capacity and algorithms," *IEEE Trans. Inf. Theory*, vol. 59, no. 9, pp. 5779–5804, Sept 2013.

[62] J. Zhang, F. Engelmann, and P. Elia, "Coded caching for reducing CSIT-feedback in wireless communications," in *Proc. Allerton Conf. Communication, Control and Computing*, Sep. 2015.

[63] A. G. Davoodi and S. A. Jafar, "Aligned image sets under channel uncertainty: Settling a conjecture by Lapidoth, Shamai and Wigger on the collapse of degrees of freedom under finite precision CSIT," *CoRR*, vol. abs/1403.1541, 2014. [Online]. Available: http://arxiv.org/abs/1403.1541

[64] M. Torrellas, A. Agustin, and J. Vidal, "Retrospective interference alignment for the MIMO interference broadcast channel," in *Proc. IEEE Int. Symp. Information Theory (ISIT)*, June 2015.

[65] A. Bracher and M. A. Wigger, "Feedback and partial message side-information on the semideterministic broadcast channel," in *Proc. IEEE Int. Symp. Information Theory (ISIT)*, June 2015.

[66] S. Lashgari, R. Tandon, and S. Avestimehr, "Three-user MISO broadcast channel: How much can CSIT heterogeneity help?" in *Proc. IEEE Int. Conf. Communications (ICC)*, June 2015, pp. 4187–4192.

[67] P. de Kerret, D. Gesbert, J. Zhang, and P. Elia, "Optimally bridging the gap from delayed to perfect csit in the k-user miso bc," in *2016 IEEE Information Theory Workshop (ITW)*, Sept 2016, pp. 300–304.

[68] M. Kobayashi and G. Caire, "On the net DoF comparison between ZF and MAT over time-varying MISO broadcast channels," in *Proc. IEEE Int. Symp. Information Theory (ISIT)*, Jul. 2012.

[69] J. Chen, S. Yang, A. Özgür, and A. Goldsmith, "Achieving full dof in heterogeneous parallel broadcast channels with outdated CSIT," *CoRR*, vol. abs/1409.6808, 2014. [Online]. Available: http://arxiv.org/abs/1409.6808

[70] N. Naderializadeh, M. A. Maddah-Ali, and A. S. Avestimehr, "Fundamental limits of cache-aided interference management," *CoRR*, vol. abs/1602.04207, 2016. [Online]. Available: http://arxiv.org/abs/1602.04207

[71] J. Zhang and P. Elia, "The synergistic gains of coded caching and delayed feedback," *CoRR*, vol. abs/1511.03961, 2016. [Online]. Available: http://arxiv.org/abs/1511.03961

[72] ——, "Fundamental limits of cache-aided wireless BC: Interplay of coded-caching and CSIT feedback," August 25 2015, *EURECOM report No. RR-15-307*, available on: http://www.eurecom.fr/publication/4723.

[73] Z. Chen, P. Fan, and K. B. Letaief, "Fundamental limits of caching: improved bounds for users with small buffers," *IET Communications*, vol. 10, no. 17, pp. 2315–2318, 2016.

[74] J. Zhang and P. Elia, "Fundamental limits of cache-aided wireless BC: Interplay of coded-caching and CSIT feedback," *IEEE Trans. Inf. Theory*, vol. PP, no. 99, pp. 1–1, Feb. 2017.

[75] ——, "Feedback-aided coded caching for the MISO BC with small caches," *CoRR*, vol. abs/1606.05396, 2016. [Online]. Available: http://arxiv.org/abs/1606.05396

[76] K. Wan, D. Tuninetti, and P. Piantanida, "On caching with more users than files," *CoRR*, vol. abs/1601.06383, 2016. [Online]. Available: http://arxiv.org/abs/1601.06383

[77] S. Sahraei and M. Gastpar, "K users caching two files: An improved achievable rate," in *Annual Conference on Information Science and Systems (CISS)*, March 2016, pp. 620–624.

[78] M. M. Amiri and D. Gündüz, "Fundamental limits of caching: Improved delivery rate-cache capacity trade-off," *CoRR*, vol. abs/1604.03888, 2016. [Online]. Available: http://arxiv.org/abs/1604.03888

[79] R. Pedarsani, M. A. Maddah-Ali, and U. Niesen, "Online coded caching," *CoRR*, vol. abs/1311.3646, 2013. [Online]. Available: http://arxiv.org/abs/1311.3646

[80] J. Zhang, X. Lin, and X. Wang, "Coded caching under arbitrary popularity distributions," in *Information Theory and Applications Workshop (ITA), 2015*, Feb 2015, pp. 98–107.

[81] S. S. Bidokhti, M. A. Wigger, and R. Timo, "Noisy broadcast networks with receiver caching," *CoRR*, vol. abs/1605.02317, 2016.

[82] N. Karamchandani, U. Niesen, M. A. Maddah-Ali, and S. N. Diggavi, "Hierarchical coded caching," *CoRR*, vol. abs/1403.7007, 2014. [Online]. Available: http://arxiv.org/abs/1403.7007

[83] M. Ji, G. Caire, and A. F. Molisch, "Fundamental limits of distributed caching in D2D wireless networks," *CoRR*, vol. abs/1304.5856, 2013. [Online]. Available: http://arxiv.org/abs/1304.5856

[84] Y. Ugur, Z. H. Awan, and A. Sezgin, "Cloud radio access networks with coded caching," *CoRR*, vol. abs/1512.02385, 2015. [Online]. Available: http://arxiv.org/abs/1512.02385

[85] S. H. Lim, C. Wang, and M. Gastpar, "Information theoretic caching: The multi-user case," *CoRR*, vol. abs/1604.02333, 2016. [Online]. Available: http://arxiv.org/abs/1604.02333

[86] M. Wigger, R. Timo, and S. Shamai, "Complete interference mitigation through receiver-caching in wyner's networks," in *Proc. IEEE Information Theory Workshop (ITW)*, Sept 2016, pp. 335–339.

[87] D. Tse and P. Viswanath, *Fundamentals of wireless communication*. Cambridge University Press, 2005.

[88] R. H. Etkin, D. N. C. Tse, and H. Wang, "Gaussian interference channel capacity to within one bit," *IEEE Trans. Inf. Theory*, vol. 54, no. 12, pp. 5534 – 5562, Dec. 2008.

[89] S. Karmakar and M. K. Varanasi, "The generalized degrees of freedom of the MIMO interference channel," in *Proc. IEEE Int. Symp. Information Theory (ISIT)*, July 2011, pp. 2198–2202.

[90] ——, "The generalized multiplexing gain region of the slow fading MIMO interference channel and its achievability with limited feedback," in *Proc. IEEE Int. Symp. Information Theory (ISIT)*, Jul. 2012.

[91] ——, "The generalized degrees of freedom region of the MIMO interference channel and its achievability," *IEEE Trans. Inf. Theory*, vol. 58, no. 12, pp. 7188 – 7203, Dec. 2012.

[92] C. Huang, V. R. Cadambe, and S. A. Jafar, "Interference alignment and the generalized degrees of freedom of the X channel," *IEEE Trans. Inf. Theory*, vol. 58, no. 8, pp. 5130 – 5150, May 2012.

[93] A. G. Davoodi and S. A. Jafar, "Transmitter cooperation under finite precision CSIT: A GDoF perspective," in *2015 IEEE Global Communications Conference (GLOBECOM)*, Dec 2015, pp. 1–6.

[94] I. Maric, R. Dabora, and A. J. Goldsmith, "Relaying in the presence of interference: Achievable rates, interference forwarding, and outer bounds," *IEEE Trans. Inf. Theory*, vol. 58, no. 7, pp. 4342–4354, July 2012.

[95] J. Zhang and P. Elia, "Fundamental limits of cache-aided wireless BC: Interplay of coded-caching and CSIT feedback," in *Proc. Allerton Conf. Communication, Control and Computing*, Sept 2016, pp. 924–932.

[96] ——, "Wireless coded caching: A topological perspective," *CoRR*, vol. abs/1606.08253, 2016. [Online]. Available: http://arxiv.org/abs/1606.08253

[97] P. de Kerret, D. Gesbert, J. Zhang, and P. Elia, "Optimal Sum-DoF of the K-user MISO BC with current and delayed feedback," *CoRR*, vol. abs/1604.01653, 2016. [Online]. Available: http://arxiv.org/abs/1604.01653

[98] S. Lashgari, A. S. Avestimehr, and C. Suh, "Linear degrees of freedom of the X-channel with delayed CSIT," *IEEE Trans. Inf. Theory*, vol. 60, no. 4, pp. 2180 – 2189, Apr. 2014.

[99] S. Jafar and S. Shamai, "Degrees of freedom region of the MIMO X channel," *IEEE Trans. Inf. Theory*, vol. 54, no. 1, pp. 151 – 170, Jan 2008.

[100] C. Huang, V. R. Cadambe, and S. A. Jafar, "Interference alignment and the generalized degrees of freedom of the X channel," *IEEE Trans. Inf. Theory*, vol. 58, no. 8, pp. 5130–5150, Aug 2012.

[101] D. T. H. Kao and A. S. Avestimehr, "Linear degrees of freedom of the mimo X-channel with delayed CSIT," *IEEE Trans. Inf. Theory*, vol. 63, no. 1, pp. 297–319, Jan 2017.

[102] R. Tandon, S. Mohajer, H. V. Poor, and S. Shamai, "On X-channels with feedback and delayed CSI," in *Proc. IEEE Int. Symp. Information Theory (ISIT)*, July 2012, pp. 1877–1881.

[103] M. J. Abdoli, A. Ghasemi, and A. K. Khandani, "On the degrees of freedom of K-user SISO interference and X channels with delayed CSIT," *IEEE Trans. Inf. Theory*, vol. 59, no. 10, pp. 6542–6561, Oct 2013.

[104] N. Lee, R. Tandon, and R. W. Heath Jr., "Distributed space-time interference alignment with moderately-delayed CSIT," *IEEE Trans. Wireless Commun.*, vol. 14, no. 2, pp. 1048–1059, Feb. 2015.

[105] J. Chen and P. Elia, "Can imperfect delayed CSIT be as useful as perfect delayed CSIT? DoF analysis and constructions for the BC," in *Proc. Allerton Conf. Communication, Control and Computing*, Oct. 2012.