Achieving Full Multiplexing and Unbounded Caching Gains with Bounded Feedback Resources

Eleftherios Lampiris
EURECOM
06410 Biot, France
Email: lampiris@eurecom.fr

Petros Elia
EURECOM
06410 Biot, France
Email: elia@eurecom.fr

***Abstract*— In the context of the K-user MISO broadcast channel with cache-aided receivers, recent multi-antenna coded-caching techniques have sought to complement the traditional multiplexing gains associated to multiple (L) antennas, with the (potentially unbounded) caching gains (G) associated to coded caching. To date, all known existing efforts to combine the two gains, either resulted in a maximum known DoF $L+G$ that required though CSIT on all $(L+G)$ users served at a time (i.e., that induced potentially unbounded CSIT costs that matched the DoF gains), or resulted in a much compromised DoF where multiplexing gains came at the expense of bounded or vanishing caching gains. We present here a new multi-antenna coded caching algorithm that introduces a new XOR generation structure which completely untangles caching gains from CSIT, delivering the desired sum-DoF of $L+G$ but with a much reduced CSIT cost of only L channel vectors at a time ($L \times L$ CSIT matrix). This means that for the first time in multi-antenna coded caching, one can achieve full multiplexing gains and unbounded caching gains, at the mere CSIT cost associated to achieving the multiplexing gains. In the end, the result solidifies the role of coded caching as a method for reducing feedback requirements in multi-antenna environments.**

I. Introduction

The promise of Coded Caching [1] is to enable a single transmission to be useful to many users, thus decreasing the delivery time significantly. In a broadcast setting where a set of K users will each ask for some file from a library of N popular files, the main idea is to cache a portion $\gamma \in (0,1)$ from each file, such that users can benefit from the fact that desired content is already partially stored in the cache, as well as – most importantly – from the fact that the cached undesired content can be used as side information to remove interference stemming from other users' requested files.

In the original single-transmitter (single-antenna) setting of [1], this idea was shown to allow for treatment of $1+K\gamma$ users at a time, corresponding to an (additive) *caching gain* of $G = K\gamma$, i.e., being able to serve – as a consequence of caching – an additional $K\gamma$ users at a time. In the context of a single-stream broadcast channel (BC) with normalized link capacity of 1 file per unit of time, this implied a worst-case (normalized) delivery time of $\mathcal{T} = \frac{K(1-\gamma)}{1+K\gamma} < \frac{1}{\gamma}$, as well as implied (the equivalent of) a Degrees-of-Freedom (DoF) performance $d_\Sigma(\gamma) \triangleq \frac{1-\gamma}{\mathcal{T}} = 1+K\gamma$ which could in theory increase indefinitely with an increasing K.

Recent works have extended the above idea to the multi-transmitter case, with the ultimate goal of combining this caching gain with the multiplexing gain that comes from having multiple transmitters (multiple antennas). The work in [2] showed that — in a wired multi-server (L servers) setting which can easily be seen to correspond to the cache-aided MISO BC setting with L transmit antennas — the two gains could be combined additively, yielding (the equivalent of) a sum-DoF equal to

$$d_\Sigma = L + K\gamma.$$

This was rather surprising because the two gains are attributed to two seemingly 'opposing' approaches: multiplexing gain is generally due to signal separation, while caching gain is due to signal-combining, i.e., multicasting. Since then, many works such as [3]–[8] have developed different coded caching schemes for the multi transmit-antenna (MISO-BC) setting.

A. Scaling of CSIT Costs in Multi-Antenna Coded Caching

While the original coded caching approach [1] in the single-stream ($L=1$) setting, could achieve the near optimal (and under some basic assumptions, optimal [9], [10]) caching gain $K\gamma$ without requiring any channel state information at the transmitter (CSIT), a main problem with all known multi-antenna coded caching methods [2]–[6] that achieved the full (maximum known) DoF $L+K\gamma$, is that they came with a maximal CSIT cost of $L+K\gamma$ CSI vectors, as they required that *each* served user must feedback their full channel vector[1].

The following example aims to demonstrate the aforementioned CSIT costs, and it focuses on a simple instance of the original multiserver method in [2] which serves as a proxy to many other methods with similar CSIT requirements.

***Example* 1.** *Let us consider the cache-aided MISO BC setting with $K=4$ users, normalized cache size $\gamma = 1/2$, and $L=2$ transmit antennas, where the multiserver approach can treat $L+K\gamma = 4$ users at a time. Assuming that users $1,2,3,4$*

This work was supported by the ANR project ECOLOGICAL-BITS-AND-FLOPS.

[1]Other methods that considered reduced CSIT, experience reduced DoF. For example, the work in [7] considers reduced quality CSIT on fewer served users, but yields a maximum DoF that is bounded close to L.

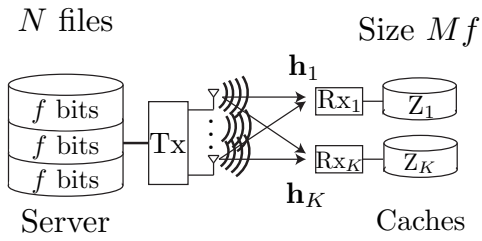

Fig. 1. An L-antenna transmitter having access to a library of N files and communicating with K receivers each with cache size of $M = \gamma N$ files.

respectively request files A, B, C, D, then each of the three transmissions takes the form

$$\begin{aligned}\mathbf{x} =&\mathbf{h}_4^{\perp} A_{23} \oplus B_{13} \oplus C_{12} + \mathbf{h}_3^{\perp} A_{24} \oplus B_{14} \oplus D_{12} + \\ &+ \mathbf{h}_2^{\perp} A_{34} \oplus C_{14} \oplus D_{13} + \mathbf{h}_1^{\perp} B_{34} \oplus C_{24} \oplus D_{23}\end{aligned} \quad (1)$$

where $\mathbf{h}_k^{\perp}$ denotes the precoder orthogonal to the channel of user k, and where A_{ij} (respectively B_{ij}, C_{ij}, D_{ij}) denotes the part of file A (respectively of B, C, D) that is cached at users i and j. We clearly see that the transmitter must know all users' channel vectors.

These maximal CSIT costs, place a fundamental limitation on the full utilization of multi-antenna coded caching. As we know from [11], the existence of bounded coherence periods, sets a hard limit on the achievable DoF, as it essentially bounds the number of CSIT vectors that can be communicated back to the transmitter within that coherence period.

In this work we will introduce a fundamentally new algorithm that requires feedback from only L users, irrespective of how large the achieved caching gain is. The algorithm will achieve the same maximal-known (near optimal[2]) sum-DoF performance of $d_{\Sigma} = L + K\gamma$, thus untangling the caching gains from the CSIT bottleneck, and proving for the first time that coded caching in multi-antenna settings can use CSIT to first provide the maximal multiplexing gain, and then use the caches to provide an additional near-optimal caching gain without any additional CSIT cost.

II. Setting & Notation

a) System Setting: We assume K single-antenna receiving users connected to an L-antenna transmitter. The received signal at user $k \in \{1, 2, \dots K\} \triangleq [K]$ takes the form

$$y_k = \mathbf{h}_k^T \mathbf{x} + w_k, \quad \forall k \in [K], \quad (2)$$

where $\mathbf{x} \in \mathbb{C}^{L\times 1}$ denotes the transmitted vector from the L-antenna array satisfying the power constraint $\mathbb{E}\left\{\|\mathbf{x}\|^2\right\} \leq P$, where $\mathbf{h}^T \in \mathbb{C}^{L\times 1}$ denotes the random-fading channel vector of user k, and where w_k is the AWGN noise at receiver k. The work considers DoF so the signal to noise ratio is considered to be large.

Communication happens in two phases. First in the *cache-placement phase* the caches are filled with content from the library of N files $\{W^{(n)},\ n = 1, 2, \cdots, N\}$, while then in the *delivery phase* each user requests a single file from the library, after which the base station serves these requested files, accounting for the requests and the cached content. The aim is to reduce the worst case (over all possible demands) delivery time $\mathcal{T}$. We will first assume that $K\gamma$ is an integer multiple L, while for the other cases, we will use the standard memory sharing[3] approach.

b) Notation: For some set $\lambda \subset [K]$ of $|\lambda| = L$ users, we will denote with H_{λ}^{-1} the normalized inverse of the $L \times L$ channel matrix to these users. We will use Z_k to denote the cache content of user $k \in [K]$, and $d_k \in [N]$ to denote the index of the file[4] requested by user k. Finally $\oplus$ will denote the bitwise-XOR operator, and $\binom{n}{k}$ will denote the n-choose-k operator for some $n \geq k,\ n, k \in \mathbb{N}$.

III. Main result and an example

We proceed with the main result, which is based on the algorithm that we will describe in the next section.

Theorem 1. *In the K-user cache-aided MISO-BC with L antennas and normalized cache size γ, the DoF $d_{\Sigma} = K\gamma + L$ can be achieved with CSIT from only L users.*

Proof. The proof is constructive, and will be provided by the scheme description in Section IV. □

a) Intuition and an example: Before fully describing the scheme, we proceed with some intuition on the design.

First of all, the cache-placement method will be exactly like in [1], while the XOR generation method will be fundamentally different. The first step is to construct vectors of L XORs, where each XOR is composed of $\frac{K\gamma}{L} + 1$ subfiles, resulting in vectors that hold a total of $L + K\gamma$ different subfiles aimed at simultaneously serving a total of $K\gamma + L$ users. Each such set of $L + K\gamma$ served ("active") users will be divided into two sets; the first set λ will have L users, while the other set π will have $K\gamma$ users. The vector of XORs will be multiplied by the inverse H_{λ}^{-1} of the channel matrix corresponding to the users in λ. As a result, each of the users in λ will only receive one of the XORs (the rest will be nulled-out by the precoder), while the remaining $K\gamma$ users (i.e., those in π) will receive a linear combination of all L XORs. We design the vector of XORs in such a way so that the first category of users in λ can each "cache-out" $\frac{K\gamma}{L}$ subfiles (leaving them only with their own desired file), while the second category of users in π will be able to cache-out $K\gamma + L - 1$ subfiles, i.e., all but one subfiles.

***Example* 2.** *For example, for the case of $K = 4$, $\gamma = 1/2$ and $L = 2$, a transmitted vector takes the form*[5]

[2]We note that this performance has been proven to have a gap of at most 2 from the one-shot linear optimal sum-DoF [3].

[3]Thus if $L > K\gamma$, we would apply memory sharing between points $K\gamma' = 0$ and $K\gamma'' = L$.

[4]In the examples we will assume the standard simplified notation where $W^{(d_1)} = A, W^{(d_2)} = B$, and so on.

[5]Here the reader should note that there is a notational discrepancy between the example here and the formal notation. The notation here was kept very simple in order to more easily provide basic intuition on the structure of the scheme. Example 4 provides the more detailed version of this example.

$$\mathbf{x} = H_{12}^{-1} \cdot \begin{bmatrix} A_{34} \oplus C_{14} \\ B_{34} \oplus D_{23} \end{bmatrix} \tag{3}$$

where H_{12}^{-1} can be the ZF precoder with respect to users $1, 2$, where files A, B, C, D are requested by users $1, 2, 3, 4$ respectively, and where A_{ij} represents the part of A that can be found in caches i and j (similarly for B_{ij}, C_{ij}, D_{ij}). Hence we see that user 1 and user 2 (users in first category λ), only receive the first and second XOR respectively, due to the design of H_{12}^{-1}, and hence can decode; user 1 can cache out C_{14} to get A_{34} and user 2 can cache out D_{23} to get the desired B_{34}. On the other hand, for users 3 and 4 corresponding to the second category π, we see that user 3 can cache out A_{34}, B_{34} and D_{23} to get the desired C_{14}, while user 4 can cache out A_{34}, B_{34} and C_{14} to get the desired subfile D_{23}.

IV. Scheme Description

We proceed to describe the scheme, first briefly describing the placement and then focusing on the delivery phase.

A. Placement Phase

The placement phase happens without knowledge of L, and it follows the original scheme in [1] so that each file, $W^{(n)}, n \in [N]$, is initially split into $\binom{K}{K\gamma}$ subfiles $W_\tau^{(n)}$, each indexed by a $K\gamma$-length set $\tau \subset [K]$, in which case each cache takes the form

$$Z_k = \left\{ W_\tau^{(n)} : \forall \tau \ni k, |\tau| = K\gamma, \forall n \in [N] \right\}. \tag{4}$$

B. Delivery Phase

Before designing the XORs, we must first further split each requested subfile. After the requests $\{W^{(d_k)},\ k \in [K]\}$ are made, and after the number of antennas becomes known, each requested subfile $W_\tau^{(d_k)}$ is further split twice as follows

$$\begin{aligned} W_\tau^{(d_k)} &\to \{W_{\sigma,\tau}^{(d_k)},\ \sigma \subset [K] \setminus (\tau \cup \{k\}), |\sigma| = L-1\} \\ W_{\sigma,\tau}^{(d_k)} &\to \{W_{\sigma,\tau}^{\phi,(d_k)},\ \phi \in [K\gamma + L]\}. \end{aligned} \tag{5}$$

We note that, for clarity of exposition and to avoid many indices, we will henceforth suppress the index ϕ, and thus any $W_{\sigma,\tau}^{\phi,(d_k)}$ will be denoted as $W_{\sigma,\tau}^{(d_k)}$ unless ϕ is explicitly described.

1) XOR design: For any two sets $\mu \subset [K], \nu \subset [K]$, where $|\mu| = \frac{K\gamma}{L} + 1$ and $|\nu| = K\gamma\frac{L-1}{L}$ and where $\mu \cap \nu = \emptyset$, and for any $\sigma \subset [K] \setminus \mu \setminus \nu,\ |\sigma| = L-1$, we construct the XOR

$$X_\mu^{\nu,\sigma} = \bigoplus_{k \in \mu} W_{\sigma,(\nu \cup \mu)\setminus\{k\}}^{(d_k)} \tag{6}$$

to consist of $\frac{K\gamma}{L} + 1$ subfiles which are desired by the users in μ and which are completely known by all the users in ν. The set $(\nu \cup \mu) \setminus \{k\}$ will play the role of τ from the placement phase, and the set σ will be a function of μ, ν and a function of the set λ of users we will ZF against.

***Example* 3.** *For the case of $K\gamma = 4$ and $L = 2$, let $\mu = \{1,2,3\}, \nu = \{4,5\}$ and consider any $\sigma \in [K] \setminus \{1,2,3,4,5\}$. Then the designed XOR*

$$X_{\{123\}}^{45,\sigma} = W_{\{\sigma,\underbrace{2345}_{\tau}\}}^{(d_1)} \oplus W_{\{\sigma,1345\}}^{(d_2)} \oplus W_{\{\sigma,1245\}}^{(d_3)}$$

delivers the subfiles requested by the users in μ, where these subfiles are all known by each user in ν.

Algorithm 1: Delivery Phase

for $\lambda \subset [K], |\lambda| = L$ *(precode users in λ)* **do**
  Create H_λ^{-1}
  for $\pi \subset ([K] \setminus \lambda),\ |\pi| = K\gamma$ **do**
    Break π into some $\mathcal{F}_i\ i \in [L]:\ |\mathcal{F}_i| = \frac{K\gamma}{L}$, $\bigcup_{i\in[L]} \mathcal{F}_i = \pi,\ \mathcal{F}_i \cap \mathcal{F}_j = \emptyset, \forall i,j \in [L]$
    for $s \in \{0, 1, ..., L-1\}$ **do**
      $r_i = ((s+i-1) \mod L) + 1, i \in [L]$
      Transmit

$$\mathbf{x}_{\lambda,\pi}^s = H_\lambda^{-1} \cdot \begin{bmatrix} X_{\lambda(1)\cup\mathcal{F}_{r_1}}^{\pi\setminus\mathcal{F}_{r_1},\lambda\setminus\lambda(1)} \\ X_{\lambda(2)\cup\mathcal{F}_{r_2}}^{\pi\setminus\mathcal{F}_{r_2},\lambda\setminus\lambda(2)} \\ \vdots \\ X_{\lambda(L)\cup\mathcal{F}_{r_L}}^{\pi\setminus\mathcal{F}_{r_L},\lambda\setminus\lambda(L)} \end{bmatrix}. \tag{7}$$

2) Vector design: At this point we describe how to generate the vector of XORs to be transmitted, after precoding, across the L antennas. Every transmission will serve completely different subfiles to $K\gamma + L$ users (there is no data repetition), while requiring CSIT from only L users.

- In Step 1, a set λ of L users is chosen.
- In Step 2, a (ZF-type) precoder H_λ^{-1} is designed to separate the L users in λ.
- In Step 3, another set $\pi \subset [K] \setminus \lambda$ of $K\gamma$ users is selected from the remaining users.
- In Step 4 this set π of $K\gamma$ users is partitioned into L non-overlapping sets $\mathcal{F}_i,\ i \in [L]$, each having $\frac{K\gamma}{L}$ users.
- In Step 5 a user from λ is associated to one of the sets $\mathcal{F}_i$, as a function of a parameter s that goes from 0 to $L-1$. For example, when $s = 0$, the first XOR of the vector will be intended for users in set $\lambda(1) \cup \mathcal{F}_1$, the second XOR will be intended for the users in the set $\lambda(2) \cup \mathcal{F}_2$ and so on. When on the other hand $s = 1$, then the first XOR will be intended for users in $\lambda(1) \cup \mathcal{F}_2$, the second XOR will be for users in $\lambda(2) \cup \mathcal{F}_3$, and so on, modulo L. In particular, step 5 (and the adjoint step 6) allows us to iterate over all sets $\mathcal{F}_i$, associating every time a distinct set $\mathcal{F}_i$ to a distinct user from group λ, until all users from set λ have been associated with all sets $\mathcal{F}_i$. Then, in the

last step (Step 7) the vector of the L XORs is transmitted after being precoded by H_λ^{-1}.

By design of the XORs (cf. (6)), the constructed vector guarantees (together with the precoder) that the users in λ can decode the single XOR that they receive, while also guaranteeing that each user in π has cached all subfiles in the entire vector, apart from their desired subfile.

C. Calculating the DoF performance

a) Showing that each desired subfile is transmitted: The first task here is to show exactly in which transmission each subfile appears. Let us take any arbitrary subfile $W_{\sigma,\tau}^{(d_k)}$. This first defines the set of active users to be $\sigma \cup \tau \cup \{k\} = \lambda \cup \pi$. Let us also recall that $\lambda \cap \pi = \emptyset, \sigma \cap \tau = \emptyset$ and $\sigma \subset \lambda$, where $|\sigma| = L-1, |\lambda| = L, |\pi| = |\tau| = K\gamma$. Recall that σ, τ, k are derived from the subfile and as a result are fixed.

We consider two distinct cases. In the *first case*, let $\lambda = \sigma \cup \{k\}$, in which case $W_{\sigma,\tau}^{(d_k)}$ will appear in transmission $\mathbf{x}_{\lambda,\pi}^s$, while $\pi = (\sigma \cup \tau \cup \{k\}) \setminus \lambda = \tau$ and for $s = 0, 1, \cdots, L-1$. The *second case* corresponds to when $k \notin \lambda$, in which case we can see that the set of all possible λ that can include σ, are $\lambda = \sigma \cup \tau(i),\ i = 1, 2, \cdots, K\gamma$. Hence the subfile $W_{\sigma,\tau}^{(d_k)}$ will appear L times when case 1 occurs, and $K\gamma$ times when case 2 occurs, thus showing that $W_{\sigma,\tau}^{(d_k)}$ will appear in a total of $L + K\gamma$ transmissions, and thus in each such transmission we will send a different subfile $W_{\sigma,\tau}^{\phi,(d_k)}$, $\phi = 1, 2, \cdots, L + K\gamma$. This proves that all the data is transmitted, and it also explains the reason for the last subpacketization into $W_{\sigma,\tau}^{\phi,(d_k)}$ for the $L + K\gamma$ different values of ϕ.

b) Decodability in each transmission: The decodability in each transmission was explained in the description of the algorithm; the type 1 users in set λ cache out $K\gamma/L$ files from the one visible XOR, while type 2 users (in set π) cache out $K\gamma + L - 1$ files.

c) Calculating the DoF performance: The resulting DoF can now be easily seen to be $d_\Sigma = L + K\gamma$ by recalling that each transmission includes $K\gamma + L$ different subfiles, and that no subfile is ever repeated. A quick verification, accounting for the number of iterations in each step and the total subpacketization $Q = \binom{K}{K\gamma}\binom{K-K\gamma-1}{L-1}(K\gamma + L)$, yields $\mathcal{T} = \frac{1}{Q}\overbrace{\binom{K}{L}}^{Step\ 1}\overbrace{\binom{K-L}{K\gamma}}^{Step\ 3}\cdot\overbrace{L}^{Step\ 5} = \frac{K(1-\gamma)\cdot L}{L(K\gamma+L)} = \frac{K(1-\gamma)}{K\gamma+L}$ which implies as sum-DoF of $d_\Sigma = K(1-\gamma)/\mathcal{T} = L + K\gamma$. □

The following example employs the complete notation $W_{\sigma,\tau}^{\phi,(d_k)}$ to demonstrate the iteration over all subfiles. As before, we use $A_{\sigma,\tau}^{(\phi)}$ to refer to $W_{\sigma,\tau}^{\phi,(d_1)}$, $B_{\sigma,\tau}^{(\phi)}$ to refer to $W_{\sigma,\tau}^{\phi,(d_2)}$, and so on.

***Example* 4** (Example of scheme)**.** *Consider a transmitter with $L = 2$ antennas, serving $K = 4$ users with caching redundancy $K\gamma = 2$. Each file is split into $Q = \overbrace{(K\gamma + L)}^{\phi}\overbrace{\binom{K-K\gamma-1}{L-1}}^{\sigma}\overbrace{\binom{K}{K\gamma}}^{\tau} = 24$ subfiles, and the following are the $\binom{K}{L}\binom{K-L}{K\gamma}L = 12$ transmissions that will satisfy all the users' requests.*

$$\mathbf{x}_{12,34}^1 = H_{12}^{-1}\begin{bmatrix} A_{2,34}^{(1)} \oplus C_{2,14}^{(1)} \\ B_{1,34}^{(1)} \oplus D_{1,23}^{(1)} \end{bmatrix}, \mathbf{x}_{12,34}^2 = H_{12}^{-1}\begin{bmatrix} A_{2,34}^{(2)} \oplus D_{2,13}^{(1)} \\ B_{1,34}^{(2)} \oplus C_{1,24}^{(1)} \end{bmatrix}$$

$$\mathbf{x}_{34,12}^1 = H_{34}^{-1}\begin{bmatrix} B_{4,13}^{(1)} \oplus C_{4,12}^{(1)} \\ A_{3,24}^{(1)} \oplus D_{3,12}^{(1)} \end{bmatrix}, \mathbf{x}_{34,12}^2 = H_{34}^{-1}\begin{bmatrix} A_{4,23}^{(1)} \oplus C_{4,12}^{(2)} \\ B_{3,14}^{(1)} \oplus D_{3,12}^{(2)} \end{bmatrix}$$

$$\mathbf{x}_{24,13}^1 = H_{24}^{-1}\begin{bmatrix} A_{4,23}^{(2)} \oplus B_{4,13}^{(2)} \\ C_{2,14}^{(2)} \oplus D_{2,13}^{(2)} \end{bmatrix}, \mathbf{x}_{24,13}^2 = H_{24}^{-1}\begin{bmatrix} B_{4,13}^{(3)} \oplus C_{4,12}^{(3)} \\ A_{2,34}^{(3)} \oplus D_{2,13}^{(3)} \end{bmatrix}$$

$$\mathbf{x}_{13,24}^1 = H_{13}^{-1}\begin{bmatrix} A_{3,24}^{(2)} \oplus B_{3,14}^{(2)} \\ C_{1,24}^{(2)} \oplus D_{1,23}^{(2)} \end{bmatrix}, \mathbf{x}_{13,24}^2 = H_{13}^{-1}\begin{bmatrix} A_{3,24}^{(3)} \oplus D_{3,12}^{(2)} \\ B_{1,34}^{(3)} \oplus C_{1,24}^{(3)} \end{bmatrix}$$

$$\mathbf{x}_{14,23}^1 = H_{14}^{-1}\begin{bmatrix} A_{4,23}^{(3)} \oplus B_{4,13}^{(4)} \\ D_{1,23}^{(3)} \oplus C_{1,24}^{(4)} \end{bmatrix}, \mathbf{x}_{14,23}^2 = H_{14}^{-1}\begin{bmatrix} A_{4,23}^{(4)} \oplus C_{4,12}^{(4)} \\ B_{1,34}^{(4)} \oplus D_{1,23}^{(4)} \end{bmatrix}$$

$$\mathbf{x}_{23,14}^1 = H_{23}^{-1}\begin{bmatrix} A_{3,24}^{(4)} \oplus B_{3,14}^{(3)} \\ C_{2,14}^{(3)} \oplus D_{2,13}^{(4)} \end{bmatrix}, \mathbf{x}_{23,14}^2 = H_{23}^{-1}\begin{bmatrix} B_{3,14}^{(4)} \oplus D_{3,12}^{(4)} \\ C_{2,14}^{(4)} \oplus A_{2,34}^{(4)} \end{bmatrix}.$$

Observing for example the first transmission, we see that user 1 only receives $A_{2,34}^{(1)} \oplus C_{2,14}^{(1)}$ and can thus decode $A_{2,34}^{(1)}$ by caching out $C_{2,14}^{(1)}$ and similarly user 2 receives only $B_{1,34}^{(1)} \oplus D_{1,23}^{(1)}$ and can decode $B_{1,34}^{(1)}$ by caching out $D_{1,23}^{(1)}$. On the other hand, user 3 can decode $C_{2,14}^{(1)}$ by caching out $A_{2,34}^{(1)}, B_{1,34}^{(1)}, D_{1,23}^{(1)}$, and user 4 can decode $D_{1,23}^{(1)}$ by caching out $A_{2,34}^{(1)}, B_{1,34}^{(1)}, C_{2,14}^{(1)}$. As we see, the delay is $\mathcal{T} = \frac{12}{24}$ and the sum-DoF is $d_\Sigma = \frac{K(1-\gamma)}{\mathcal{T}} = 4$.

D. Complete example of a more involved scheme

To further understand the algorithm we will provide a more involved example.

Here we will present all the transmissions for the case where $K = 6$, $L = 2$, and $\gamma = 2/3$ ($K\gamma = 4$).

$$\mathbf{x}_{12,3456}^1 = H_{12}^{-1}\begin{bmatrix} A_{2,3456}^{(1)} \oplus C_{2,1456}^{(1)} \oplus D_{2,1356}^{(1)} \\ B_{1,3456}^{(1)} \oplus E_{1,2346}^{(1)} \oplus F_{1,2345}^{(1)} \end{bmatrix}$$

$$\mathbf{x}_{12,3456}^2 = H_{12}^{-1}\begin{bmatrix} A_{2,3456}^{(2)} \oplus E_{2,1346}^{(1)} \oplus F_{2,1345}^{(1)} \\ B_{1,3456}^{(2)} \oplus C_{1,2456}^{(1)} \oplus D_{1,2356}^{(1)} \end{bmatrix}$$

$$\mathbf{x}_{13,2456}^{(1)} = H_{13}^{-1}\begin{bmatrix} A_{3,2456}^{(1)} \oplus B_{3,1456}^{(1)} \oplus D_{3,1256}^{(1)} \\ C_{1,2456}^{(2)} \oplus E_{1,2346}^{(2)} \oplus F_{1,2345}^{(2)} \end{bmatrix}$$

$$\mathbf{x}_{13,2456}^2 = H_{13}^{-1}\begin{bmatrix} A_{3,2456}^{(2)} \oplus E_{3,1246}^{(1)} \oplus F_{3,1245}^{(1)} \\ C_{1,2456}^{(3)} \oplus B_{1,3456}^{(3)} \oplus D_{1,2356}^{(2)} \end{bmatrix}$$

$$\mathbf{x}_{14,2356}^1 = H_{14}^{-1}\begin{bmatrix} A_{4,2356}^{(1)} \oplus B_{4,1356}^{(1)} \oplus C_{4,1256}^{(1)} \\ D_{1,2356}^{(3)} \oplus E_{1,2346}^{(3)} \oplus F_{1,2345}^{(3)} \end{bmatrix}$$

$$\mathbf{x}_{14,2356}^2 = H_{14}^{-1}\begin{bmatrix} A_{4,2356}^{(2)} \oplus E_{4,1236}^{(1)} \oplus F_{4,1235}^{(1)} \\ D_{1,2356}^{(4)} \oplus B_{1,3456}^{(4)} \oplus C_{1,2456}^{(4)} \end{bmatrix}$$

$$\mathbf{x}_{15,2346}^1 = H_{15}^{-1}\begin{bmatrix} A_{5,2346}^{(1)} \oplus B_{5,1346}^{(1)} \oplus C_{5,1246}^{(1)} \\ E_{1,2346}^{(4)} \oplus D_{1,2356}^{(5)} \oplus F_{1,2345}^{(4)} \end{bmatrix}$$

$$\mathbf{x}_{15,2346}^2 = H_{15}^{-1}\begin{bmatrix} A_{5,2346}^{(2)} \oplus D_{5,1236}^{(1)} \oplus F_{5,1234}^{(1)} \\ E_{1,2346}^{(5)} \oplus B_{1,3456}^{(5)} \oplus C_{1,2456}^{(5)} \end{bmatrix}$$

$$\mathbf{x}^1_{16,2345} = H_{16}^{-1}\begin{bmatrix} A^{(1)}_{6,2345} \oplus B^{(1)}_{6,1345} \oplus C^{(1)}_{6,1245} \\ F^{(5)}_{1,2345} \oplus D^{(6)}_{1,2356} \oplus E^{(6)}_{1,2346} \end{bmatrix}$$

$$\mathbf{x}^2_{16,2345} = H_{16}^{-1}\begin{bmatrix} A^{(2)}_{6,2345} \oplus D^{(1)}_{6,1235} \oplus E^{(1)}_{6,1234} \\ F^{(6)}_{1,2345} \oplus B^{(6)}_{1,3456} \oplus C^{(6)}_{1,2456} \end{bmatrix}$$

$$\mathbf{x}^1_{23,1456} = H_{23}^{-1}\begin{bmatrix} B^{(2)}_{3,1456} \oplus A^{(3)}_{3,2456} \oplus D^{(2)}_{3,1256} \\ C^{(2)}_{2,1456} \oplus E^{(2)}_{2,1346} \oplus F^{(2)}_{2,1345} \end{bmatrix}$$

$$\mathbf{x}^2_{23,1456} = H_{23}^{-1}\begin{bmatrix} B^{(3)}_{3,1456} \oplus E^{(2)}_{3,1246} \oplus F^{(2)}_{3,1245} \\ C^{(3)}_{2,1456} \oplus A^{(3)}_{2,3456} \oplus D^{(2)}_{2,1356} \end{bmatrix}$$

$$\mathbf{x}^1_{24,1356} = H_{24}^{-1}\begin{bmatrix} B^{(2)}_{4,1356} \oplus A^{(3)}_{4,2356} \oplus C^{(2)}_{4,1256} \\ D^{(3)}_{2,1356} \oplus E^{(3)}_{2,1346} \oplus F^{(3)}_{2,1345} \end{bmatrix}$$

$$\mathbf{x}^2_{24,1356} = H_{24}^{-1}\begin{bmatrix} B^{(3)}_{4,1356} \oplus E^{(2)}_{4,1236} \oplus F^{(2)}_{4,1235} \\ D^{(4)}_{2,1356} \oplus A^{(4)}_{2,3456} \oplus C^{(4)}_{2,1456} \end{bmatrix}$$

$$\mathbf{x}^1_{25,1346} = H_{25}^{-1}\begin{bmatrix} B^{(2)}_{5,1346} \oplus A^{(3)}_{5,2346} \oplus C^{(2)}_{5,1246} \\ E^{(4)}_{2,1346} \oplus D^{(5)}_{2,1356} \oplus F^{(4)}_{2,1345} \end{bmatrix}$$

$$\mathbf{x}^2_{25,1346} = H_{25}^{-1}\begin{bmatrix} B^{(3)}_{5,1346} \oplus D^{(2)}_{5,1236} \oplus F^{(2)}_{5,1234} \\ E^{(5)}_{2,1346} \oplus A^{(5)}_{2,3456} \oplus C^{(5)}_{2,1456} \end{bmatrix}$$

$$\mathbf{x}^1_{26,1345} = H_{26}^{-1}\begin{bmatrix} B^{(2)}_{6,1345} \oplus A^{(3)}_{6,2345} \oplus C^{(2)}_{6,1245} \\ F^{(5)}_{2,1345} \oplus D^{(6)}_{2,1356} \oplus E^{(6)}_{2,1346} \end{bmatrix}$$

$$\mathbf{x}^2_{26,1345} = H_{26}^{-1}\begin{bmatrix} B^{(3)}_{6,1345} \oplus D^{(2)}_{6,1235} \oplus E^{(2)}_{6,1234} \\ F^{(6)}_{2,1345} \oplus A^{(6)}_{2,3456} \oplus C^{(6)}_{2,1456} \end{bmatrix}$$

$$\mathbf{x}^1_{34,1256} = H_{34}^{-1}\begin{bmatrix} C^{(3)}_{4,1256} \oplus A^{(4)}_{4,2356} \oplus B^{(4)}_{4,1356} \\ D^{(3)}_{3,1256} \oplus E^{(3)}_{3,1246} \oplus F^{(3)}_{3,1245} \end{bmatrix}$$

$$\mathbf{x}^2_{34,1256} = H_{34}^{-1}\begin{bmatrix} C^{(4)}_{4,1256} \oplus E^{(3)}_{4,1236} \oplus F^{(3)}_{4,1235} \\ D^{(4)}_{3,1256} \oplus A^{(4)}_{3,2456} \oplus B^{(4)}_{3,1456} \end{bmatrix}$$

$$\mathbf{x}^1_{35,1246} = H_{35}^{-1}\begin{bmatrix} C^{(3)}_{5,1246} \oplus A^{(4)}_{5,2346} \oplus B^{(4)}_{5,1346} \\ E^{(4)}_{3,1246} \oplus D^{(5)}_{3,1256} \oplus F^{(4)}_{3,1245} \end{bmatrix}$$

$$\mathbf{x}^2_{35,1246} = H_{35}^{-1}\begin{bmatrix} C^{(4)}_{5,1246} \oplus D^{(3)}_{5,1236} \oplus F^{(3)}_{5,1234} \\ E^{(5)}_{3,1246} \oplus A^{(5)}_{3,2456} \oplus B^{(5)}_{3,1456} \end{bmatrix}$$

$$\mathbf{x}^1_{36,1245} = H_{36}^{-1}\begin{bmatrix} C^{(3)}_{6,1245} \oplus A^{(4)}_{6,2345} \oplus B^{(4)}_{6,1345} \\ F^{(5)}_{3,1245} \oplus D^{(6)}_{3,1256} \oplus E^{(6)}_{3,1246} \end{bmatrix}$$

$$\mathbf{x}^2_{36,1245} = H_{36}^{-1}\begin{bmatrix} C^{(4)}_{6,1245} \oplus D^{(3)}_{6,1235} \oplus E^{(3)}_{6,1234} \\ F^{(6)}_{3,1245} \oplus A^{(6)}_{3,2456} \oplus B^{(6)}_{3,1456} \end{bmatrix}$$

$$\mathbf{x}^1_{45,1236} = H_{45}^{-1}\begin{bmatrix} D^{(4)}_{5,1236} \oplus A^{(5)}_{5,2346} \oplus B^{(5)}_{5,1346} \\ E^{(4)}_{4,1236} \oplus C^{(5)}_{4,1256} \oplus F^{(4)}_{4,1235} \end{bmatrix}$$

$$\mathbf{x}^2_{45,1236} = H_{45}^{-1}\begin{bmatrix} D^{(5)}_{5,1236} \oplus C^{(5)}_{5,1246} \oplus F^{(4)}_{5,1234} \\ E^{(5)}_{4,1236} \oplus A^{(5)}_{4,2356} \oplus B^{(5)}_{4,1356} \end{bmatrix}$$

$$\mathbf{x}^1_{46,1235} = H_{46}^{-1}\begin{bmatrix} D^{(4)}_{6,1235} \oplus A^{(5)}_{6,2345} \oplus B^{(5)}_{6,1345} \\ F^{(5)}_{4,1235} \oplus C^{(6)}_{4,1256} \oplus E^{(6)}_{4,1236} \end{bmatrix}$$

$$\mathbf{x}^2_{46,1235} = H_{46}^{-1}\begin{bmatrix} D^{(5)}_{6,1235} \oplus C^{(5)}_{6,1245} \oplus E^{(4)}_{6,1234} \\ F^{(6)}_{4,1235} \oplus A^{(6)}_{4,2356} \oplus B^{(6)}_{4,1356} \end{bmatrix}$$

$$\mathbf{x}^1_{56,1234} = H_{56}^{-1}\begin{bmatrix} E^{(5)}_{6,1234} \oplus A^{(6)}_{6,2345} \oplus B^{(6)}_{6,1345} \\ F^{(5)}_{5,1234} \oplus C^{(6)}_{5,1246} \oplus D^{(6)}_{5,1236} \end{bmatrix}$$

$$\mathbf{x}^2_{56,1234} = H_{56}^{-1}\begin{bmatrix} E^{(6)}_{6,1234} \oplus C^{(6)}_{6,1245} \oplus D^{(6)}_{6,1235} \\ F^{(6)}_{5,1234} \oplus A^{(6)}_{5,2346} \oplus B^{(6)}_{5,1346} \end{bmatrix}$$

V. Conclusion

In this work we provided a new algorithm designed for the L antenna MISO-BC with K cache-aided receivers, and we have shown that the algorithm can achieve the theoretical order optimal[6] DoF of $K\gamma + L$ served users at a time, with a CSIT cost that is a function only of the number of antennas.

A. Various benefits of reducing feedback: higher effective DoF, separability of design, and feedback reuse

This feedback reduction has multiple beneficial effects. Firstly, the reduced feedback requirements increase the effective DoF simply because they allow for more time (within any given coherence block) to transmit actual data. Secondly, the algorithm allows us to increase the number of users and/or their cache size, without having to change the amount of CSIT feedback, and without additional training overhead. Thirdly, the structure of the algorithm allows for training to happen less often, since by choosing one group of users for the precoding (i.e., by choosing λ), the resulting H_λ^{-1} can stay fixed for a large number of time slots (can stay fixed for all possible τ), thus allowing — within a coherence period — for a substantial reuse of the acquired feedback.

References

[1] M. A. Maddah-Ali and U. Niesen, "Fundamental limits of caching," *IEEE Transactions on Information Theory*, 2014.

[2] S. P. Shariatpanahi, S. A. Motahari, and B. H. Khalaj, "Multi-server Coded Caching," *IEEE Transactions on Information Theory*, 2016.

[3] N. Naderializadeh, M. A. Maddah-Ali, and A. S. Avestimehr, "Fundamental limits of cache-aided interference management," *IEEE Transactions on Information Theory*, 2017.

[4] S. P. Shariatpanahi, G. Caire, and B. H. Khalaj, "Multi-antenna Coded Caching," in *IEEE International Symposium on Information Theory (ISIT)*, 2017.

[5] A. Tölli, S. P. Shariatpanahi, J. Kaleva, and B. Khalaj, "Multi-antenna interference management for coded caching," *arXiv preprint arXiv:1711.03364*, 2017.

[6] J. S. P. Roig, D. Gündüz, and F. Tosato, "Interference networks with caches at both ends," in *2017 IEEE International Conference on Communications (ICC)*, May 2017.

[7] E. Piovano, H. Joudeh, and B. Clerckx, "On Coded Caching in the overloaded MISO Broadcast Channel," in *IEEE International Symposium on Information Theory (ISIT)*, June 2017.

[8] E. Lampiris and P. Elia, "Adding transmitters dramatically boosts coded-caching gains for finite file sizes," *arXiv preprint arXiv:1802.03389*, 2018.

[9] K. Wan, D. Tuninetti, and P. Piantanida, "On the optimality of uncoded cache placement," in *IEEE Information Theory Workshop (ITW)*, 2016.

[10] Q. Yu, M. A. Maddah-Ali, and A. S. Avestimehr, "The exact rate-memory tradeoff for caching with uncoded prefetching," *IEEE Transactions on Information Theory*, 2018.

[11] L. Zheng and D. Tse, "Communication on the Grassmann Manifold: A geometric approach to the noncoherent multiple-antenna channel," *IEEE Transactions on Information Theory*, 2002.

[6]This is the maximal known DoF and it has a gap of at most 2 from the one-shot linear DoF as shown in [3].