A Rate-Distortion Approach to Caching

Roy Timo[†], Shirin Saeedi Bidokhti[*], Michèle Wigger[‡] and Bernhard C. Geiger[†]
[†]Institute for Communications Engineering, Technische Universität München
[*]Department of Electrical Engineering, Stanford University
[‡]Communications and Electronics Department, Telecom ParisTech
{roy.timo, bernhard.geiger}@tum.de, saeedi@stanford.edu, michele.wigger@telecom-paristech.fr

***Abstract*—This paper takes a rate-distortion approach to the caching problem of Maddah-Ali and Niesen. We characterise the optimal tradeoffs between compression rate, reconstruction distortion and cache capacity for a single-user problem and special cases of a two-user problem. These tradeoffs illustrate some interesting connections between optimal caching strategies, Gács-Körner common information, and Wyner's common information.**

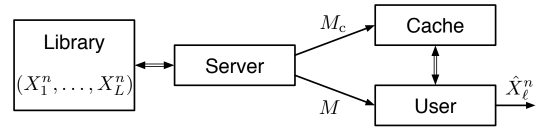

Fig. 1. Single-user RD cache problem.

I. Introduction and Setup

We address a communication scenario where users request files from a server during peak-traffic periods. The server reduces the peak-traffic by pre-placing information in cache memories close to the users during prior periods of low traffic. In these low-traffic periods, communication rate is not a limiting resource and the amount of pre-placed information is mainly restricted by the cache memory sizes.

More specifically, in this paper we consider the scenario in Figure 1. The server has access to a library with L files:

$$\text{Library} \quad \mathbf{X} := (X_1^n, X_2^n, \ldots, X_L^n),$$

where each file is a sequence of n symbols

$$X_\ell^n := (X_{\ell,1}, X_{\ell,2}, \ldots, X_{\ell,n})$$

taking value in a finite alphabet $\mathcal{X}_\ell$. For simplicity, we assume that each file is a sequence of independent and identically distributed (i.i.d.) symbols, where symbols pertaining to different files can be correlated:

$$(X_{1,1}, \ldots, X_{L,1}), \ldots, (X_{1,n}, \ldots, X_{L,n}) \quad \text{i.i.d.} \sim P_{\mathbf{X}}, \quad (1)$$

for some given joint law $P_{\mathbf{X}}$ over $\boldsymbol{\mathcal{X}} := \mathcal{X}_1 \times \cdots \times \mathcal{X}_L$.

Assume that there is a single user, which selects an index

$$\ell \in \mathcal{L} := \{1, 2, \ldots, L\}$$

arbitrarily and requests the corresponding file X_ℓ^n from the server. The user has a local cache memory of size nC bits where the server can pre-place information M_c, and which the user can access to reconstruct its requested file X_ℓ^n. A central assumption in our work is that the server has to place the information in the cache *before* it learns the user's request. The information M_c stored in the cache should thus be chosen such that it is useful for (or common to) as many files as possible.

Once the server learns the user's request $\ell \in \mathcal{L}$, it sends an nR-bit *delivery message* M to the user. Based on this message M and the cache content M_c, the user attempts to reconstruct its requested file X_ℓ^n. Hence, the delivery message M should contain all the information about X_ℓ^n that is relevant to the user and that is not yet stored in the cache memory.

Such a cache-aided setup was first considered by Maddah-Ali and Niesen in [1, 2] and triggered a series of fruitful results [3]–[9]. The works in [1]–[7] studied the problem where *independent* files $X_1^n, \ldots, X_L^n$ had to be reconstructed losslessly by *multiple* users. More specifically, these works presented various upper and lower bounds on the minimum required delivery-rate R for given cache capacity C. While we limit ourselves to a single user with cache memory, we extend the analysis to lossy reconstruction of potentially correlated files, cf. (1). We furthermore analyse the problem when a second user without cache memory is present, see the setup in Figure 4.

The main problem of interest in this paper is thus the optimal tradeoff between the *delivery rate* R, the *cache capacity* C, and the user's *reconstruction distortion*. Notice that the delivery rate R is a *worst-case* rate (or a compound rate) in the sense that it has to be sufficiently large so that the user can reconstruct every file X_ℓ^n, $\ell \in \mathcal{L}$, with desired accuracy. The problem setup by Wang, Lim and Gastpar [9], can be considered as an ergodic average-case equivalent of our worst-case (or compound) setup.

II. Single User

A. Formal Problem Definition

Let $\hat{\mathcal{X}}_1, \ldots, \hat{\mathcal{X}}_L$ be given reconstruction sets. A *joint rate-distortion-cache (RDC) code* for a given blocklength n consists of $(2L+1)$-mappings:

(i) A *cache encoder* $f_c : \boldsymbol{\mathcal{X}}^n \to \mathcal{M}_c$, where $\mathcal{M}_c$ is finite.
(ii) A *file encoder* $f_\ell : \boldsymbol{\mathcal{X}}^n \to \mathcal{M}$ for each $\ell \in \mathcal{L}$, where $\mathcal{M}$ is finite.
(iii) A *file decoder* $g_\ell : \mathcal{M} \times \mathcal{M}_c \to \hat{\mathcal{X}}_\ell^n$ for each $\ell \in \mathcal{L}$.

For brevity, we will call the above collection of encoders and decoders an $(n, \mathcal{M}, \mathcal{M}_c)$-code. Given demand $\ell \in \mathcal{L}$, the cache content and the delivery message are

$$M_c := f_c(\mathbf{X}^n) \qquad \text{and} \qquad M := f_\ell(\mathbf{X}^n);$$

and the user's reconstruction is

$$\hat{X}_\ell^n := g_\ell(M, M_c) \in \hat{\mathcal{X}}_\ell^n.$$

As per the usual rate-distortion (RD) paradigm, let us assume that the quality of $\hat{X}_\ell^n$ can be meaningfully measured using average per-letter distortions. Specifically, for each $\ell \in \mathcal{L}$, let

$$\delta_\ell : \hat{\mathcal{X}}_\ell \times \mathcal{X}_\ell \to [0, \infty)$$

be a bounded *distortion function*. For simplicity, we assume that for each symbol $x_\ell \in \mathcal{X}_\ell$ there always exists an $\hat{x}_\ell$ in $\hat{\mathcal{X}}_\ell$ such that $\delta_\ell(\hat{x}_\ell, x_\ell) = 0$.

Definition 1: Let $\mathbf{D} := (D_1, D_2, \ldots, D_L)$ and C be arbitrary nonnegative reals. We say that a delivery rate $R \geq 0$ is $(\mathbf{D}, C)$*-admissible* if for every $\epsilon > 0$ there exists a sufficiently large blocklength n and an $(n, \mathcal{M}, \mathcal{M}_c)$-code such that

$$\forall\ \ell \in \mathcal{L}: \quad \mathbb{E}\left[\frac{1}{n}\sum_{i=1}^{n} \delta_\ell(\hat{X}_{\ell,i}, X_{\ell,i})\right] \leq D_\ell + \epsilon,$$

and

$$|\mathcal{M}_c| \leq 2^{n(C+\epsilon)} \quad \text{and} \quad |\mathcal{M}| \leq 2^{n(R+\epsilon)}. \tag{2}$$

We call C the *cache capacity* and $\mathbf{D}$ the *distortion constraints*. The optimal RDC tradeoff for blocklengths $n \to \infty$ is characterised by the following function.

Definition 2: The RDC function is

$$\mathsf{R}(\mathbf{D}, C) := \inf\left\{R \geq 0 : R \text{ is } (\mathbf{D}, C)\text{-admissible}\right\}.$$

B. Main Results

The RDC function has the following properties:

Proposition 1:

(i) $\mathsf{R}(\mathbf{D}, C)$ is jointly convex and non-increasing in $\mathbf{D}$ and C.

(ii) If $C \geq H(\mathbf{X})$, then $\mathsf{R}(\mathbf{D}, C) = 0$ for all $\mathbf{D}$.

(iii) If $C = 0$, then

$$\mathsf{R}(\mathbf{D}, 0) = \max_{\ell \in \mathcal{L}} \mathsf{R}_{X_\ell}(D_\ell),$$

where $\mathsf{R}_{X_\ell}(D_\ell)$ is the usual RD function for X_ℓ,

$$\mathsf{R}_{X_\ell}(D_\ell) := \min_{\substack{p_{\hat{X}_\ell|X_\ell}:\ \mathcal{X}_\ell \to \hat{\mathcal{X}}_\ell \\ \text{s.t. } \mathbb{E}[\delta_\ell(\hat{X}_\ell, X_\ell)] \leq D_\ell}} I(X_\ell; \hat{X}_\ell).$$

Let

$$\mathsf{R}^*(\mathbf{D}, C) := \min \max_\ell I(X_\ell; \hat{X}_\ell | U), \tag{3}$$

where the minimum is taken over all $(U, \hat{X}_1, \hat{X}_2, \ldots, \hat{X}_L)$ jointly distributed with $\mathbf{X}$ such that $I(\mathbf{X}; U) \leq C$ and $\mathbb{E}[\delta_\ell(\hat{X}_\ell, X_\ell)] \leq D_\ell$ for all $\ell \in \mathcal{L}$.

Theorem 1:

$$\mathsf{R}(\mathbf{D}, C) = \mathsf{R}^*(\mathbf{D}, C).$$

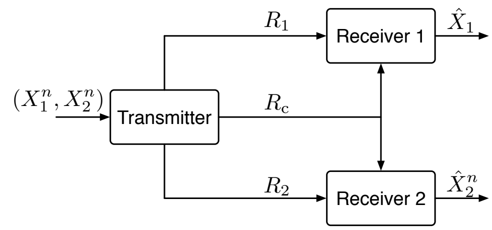

Fig. 2. Lossy Source Coding for a Simple Network.

C. Connections to the Gray-Wyner Network

For the case of $L = 2$ files, $\mathbf{X}^n = (X_1^n, X_2^n)$, there is a close connection between the RDC function and Gray and Wyner's classic *"source coding for a simple network"* problem [10]. The Gray-Wyner network is illustrated in Figure 2: A transmitter is connected to two different receivers via a common link of rate R_c and two private links of rates R_1 and R_2. The set of all achievable rate tuples (R_c, R_1, R_2) for which receivers 1 and 2 can respectively reconstruct X_1^n and X_2^n to within distortions D_1 and D_2 is given by [10, Thm. 8]

$$\mathcal{R}_{\text{GW}}(D_1, D_2) := \bigcup \left\{ (R_c, R_1, R_2) : \begin{array}{r} R_c \geq I(X_1, X_2; U) \\ R_1 \geq I(X_1; \hat{X}_1 | U) \\ R_2 \geq I(X_2; \hat{X}_2 | U) \end{array} \right\},$$

where the union is over all tuples $(X_1, X_2, U, \hat{X}_1, \hat{X}_2)$ satisfying $\mathbb{E}[\delta_\ell(\hat{X}_\ell, X_\ell)] \leq D_\ell$, for $\ell \in \{1, 2\}$. The next proposition can be proved by associating the common rate R_c of the Gray-Wyner problem with the rate of the caching message M_c, and the two private rates R_1 and R_2 of the Gray-Wyner problem with the rates of our delivery message M when the user demands X_1^n and X_2^n, respectively.

Proposition 2:

$$\mathsf{R}((D_1, D_2), C) = \min_{(C, R_1, R_2) \in \mathcal{R}_{\text{GW}}(D_1, D_2)} \max\left\{R_1, R_2\right\}.$$

D. Almost Lossless Compression

Let us now restrict attention to the case where the user wants to reconstruct X_ℓ^n (almost) losslessy. Specifically, suppose that $\hat{\mathcal{X}}_\ell = \mathcal{X}_\ell$ and $\delta_\ell(\hat{x}_\ell, x_\ell) = \mathbb{1}\{\hat{x}_\ell \neq x_\ell\}$ for all $\ell \in \mathcal{L}$ are Hamming distortion functions; and $\mathbf{0} := (0, \ldots, 0)$ is a tuple of L zeros. Given these assumptions, define the rate-cache (RC) function

$$\mathsf{R}_{\mathbf{0}}(C) := \mathsf{R}(\mathbf{0}, C).$$

From Theorem 1 we have the next corollary.

Corollary 1.1:

$$\mathsf{R}_{\mathbf{0}}(C) = \mathsf{R}^*(\mathbf{0}, C) = \min_U \max_\ell H(X_\ell | U),$$

where the minimum is taken over all auxiliary random variables U, jointly distributed with $\mathbf{X}$, satisfying $I(\mathbf{X}; U) \leq C$.

Figure 3 shows the typical behaviour of $\mathsf{R}_{\mathbf{0}}(C)$. To obtain better understanding, we propose two lower bounds and study conditions when they are tight.

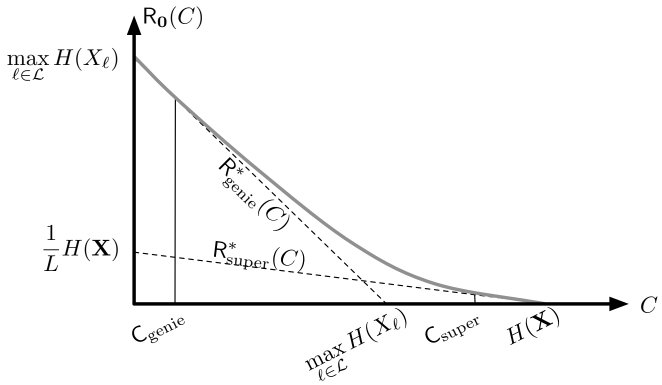

Fig. 3. An illustration of a typical RC function $\mathsf{R}_{\mathbf{0}}(C)$ and the lower bounds in Propositions 3 and 6.

1) Lower Bound $\mathsf{R}^*_{\mathbf{0},\text{Genie}}(C)$ on $\mathsf{R}_{\mathbf{0}}(C)$: Imagine that, before the caching phase, a genie tells the server which $\ell \in \mathcal{L}$ the user will select in the future. The optimal caching strategy for this hypothetical *genie-aided* problem is obvious, because for each $\ell \in \mathcal{L}$ we have a standard RD problem: The server uses an optimal code to losslessly compress the source X_ℓ^n, stores the first nC bits produced by this code in the user's cache memory, and sends the remaining bits as the delivery message. The user assembles the bits from the cache memory and the delivery message and reconstructs the requested file. The RC function of this genie-aided system, $\mathsf{R}_{\mathbf{0},\text{Genie}}(C)$, hence equals[1]

$$\mathsf{R}_{\mathbf{0},\text{Genie}}(C) = \mathsf{R}^*_{\mathbf{0},\text{Genie}}(C) := \max\big\{0,\ \max_{\ell\in\mathcal{L}} H(X_\ell) - C\big\}.$$

Since the server can always choose to ignore the genie-information, the RC function of the genie-aided system cannot exceed the RC function of the original scenario:

Proposition 3:

$$\mathsf{R}_{\mathbf{0}}(C) \geq \mathsf{R}_{\mathbf{0},\text{Genie}}(C).$$

For degraded file sets, above lower bound is tight.

Example 1: Let the DMS $\mathbf{X}$ be given by $X_\ell := (A_1, \ldots, A_\ell)$ for all $\ell \in \mathcal{L}$, where $(A_1, \ldots, A_L)$ have an arbitrary joint distribution. Then,

$$\mathsf{R}_{\mathbf{0}}(C) = \mathsf{R}^*_{\mathbf{0},\text{Genie}}(C) = \max\{0, H(X_L) - C\}.$$

2) Connection to the Gács-Körner Common Information: The lower bound $\mathsf{R}^*_{\mathbf{0},\text{Genie}}(C)$ is also trivially tight at zero cache capacity, $C = 0$; for example, see Assertion (ii) in Proposition 1. It is therefore natural to define

$$C_{\text{Genie}} := \sup\Big\{C \leq H(\mathbf{X}) : \mathsf{R}_{\mathbf{0}}(C) = \mathsf{R}^*_{\mathbf{0},\text{Genie}}(C)\Big\}$$

to be the largest cache capacity for which there is *no rate loss* with respect to the optimal genie-aided system.

Define the subset $\mathcal{L}^* \subseteq \mathcal{L}$ as

$$\mathcal{L}^* := \arg\max_{\ell\in\mathcal{L}} H(X_\ell).$$

[1]The maximum over $\ell \in \mathcal{L}$ is needed because we again consider a worst-case (compound) setup over all possible demands $\ell \in \mathcal{L}$.

Further, let

$$C^*_{\text{Genie}} := \max_{U}\ I(\mathbf{X}; U),$$

where the maximum is taken over all auxiliary random variables U jointly distributed with $\mathbf{X}$ for which the following statements hold:

(i) For every $\ell^* \in \mathcal{L}^*$, we have $U \leftrightarrow X_{\ell^*} \leftrightarrow X_{\mathcal{L}\setminus\ell^*}$, where $X_{\mathcal{L}\setminus\ell^*} := \big(X_1, X_2, \ldots, X_{\ell^*-1}, X_{\ell^*+1}, \ldots, X_L\big)$.

(ii) For every $\ell^* \in \mathcal{L}^*$,

$$H(X_{\ell^*}|U) = \max_{\ell\in\mathcal{L}} H(X_\ell|U),$$

(iii) U is defined on an alphabet $\mathcal{U}$ with $|\mathcal{U}| \leq |\boldsymbol{\mathcal{X}}| + |\mathcal{L}^*| + L$.

Proposition 4:

$$C_{\text{Genie}} = C^*_{\text{Genie}}.$$

The critical cache capacity C^*_{Genie} is related to the natural L-variable generalisation [12] of Gács and Körner's *common information*:

$$\mathsf{K}^*_{\text{GK}} := \max_{U:\ H(U|X_\ell)=0,\ \forall\ \ell\in\mathcal{L}} H(U).$$

Proposition 5:

$$C^*_{\text{Genie}} \geq \mathsf{K}^*_{\text{GK}}. \tag{4}$$

If $H(X_1) = \cdots = H(X_L)$, then (4) holds with equality.

3) Lower Bound $\mathsf{R}^*_{\mathbf{0},\text{Super}}(C)$ on $\mathsf{R}_{\mathbf{0}}(C)$: Now imagine a situation where we have a *superuser* that requests all the L sources $X_1^n, \ldots, X_L^n$ and that obtains L *delivery messages* of rate R each. Moreover, suppose that as before this superuser has a local cache memory of size nC bits that can be filled by the server. The optimal strategy for this superuser problem is again obvious, since it is equivalent to a standard RD problem with a single compression message of rate $LR+C$: The server takes an optimal code to compress the entire library $\mathbf{X}^n$ and distributes the produced bits in the cache memory and over the L delivery messages. The RC function of this superuser system, $\mathsf{R}_{\mathbf{0},\text{Super}}(C)$, hence is:

$$\mathsf{R}_{\mathbf{0},\text{Super}}(C) = \mathsf{R}^*_{\mathbf{0},\text{Super}}(C) := \max\left\{0,\ \frac{1}{L}\big(H(\mathbf{X}) - C\big)\right\}.$$

If one limits the superuser to reconstruct each source X_ℓ^n, $\ell \in \mathcal{L}$, solely based on the content in the cache memory and the ℓ-th delivery message, one obtains our original setup. The RC function of the superuser system thus can not exceed the RC function of the original setup:

Proposition 6:

$$\mathsf{R}_{\mathbf{0}}(C) \geq \mathsf{R}_{\mathbf{0},\text{Super}}(C).$$

For independent and identically distributed files, above lower bound is tight:

Example 2: Let the DMS $\mathbf{X}$ follow the product distribution $P_{\mathbf{X}} = \prod_{\ell=1}^{L} P_X$. In this case,

$$\mathsf{R}_{\mathbf{0}}(C) = \mathsf{R}^*_{\mathbf{0},\text{Super}}(C) = \max\left\{0, H(X) - \frac{C}{L}\right\}.$$

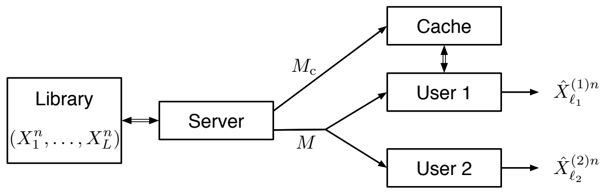

Fig. 4. Two-user RD cache problem.

4) Connection to Wyner's Common Information: The superuser lower bound is trivially tight when $C \geq H(\mathbf{X})$. So it is natural to consider the smallest cache capacity for which there is *no rate loss* with respect to the optimal superuser system,

$$C_{\text{Super}} := \inf\left\{C \geq 0 : \mathsf{R}_{\mathbf{0}}(C) = \mathsf{R}_{\mathbf{0},\text{Super}}(C)\right\}.$$

Let

$$C^*_{\text{Super}} := \min_U I(\mathbf{X}; U),$$

where the minimum is taken over all auxiliary random variables U jointly distributed with $\mathbf{X}$ such that

(i) $X_\ell \leftrightarrow U \leftrightarrow X_{\mathcal{L}\setminus\ell}$ for all $\ell \in \mathcal{L}$;
(ii) $H(X_1|U) = \cdots = H(X_L|U)$; and
(iii) U is defined on $\mathcal{U}$ with $|\mathcal{U}| \leq |\boldsymbol{\mathcal{X}}| + 2L$.

Proposition 7:

$$C_{\text{Super}} = C^*_{\text{Super}}.$$

The critical cache capacity C^*_{Super} is related to the natural L-variable generalisation [11] of *Wyner's common information*:

$$\mathsf{K}^*_{\text{W}}(\mathbf{X}) := \min_U \ I(\mathbf{X}; U),$$

where the minimum is taken over all U jointly distributed with $\mathbf{X}$ for which

(i) $X_\ell \leftrightarrow U \leftrightarrow X_{\mathcal{L}\setminus\ell}$ for all $\ell \in \mathcal{L}$; and
(ii) U is defined on an alphabet $\mathcal{U}$ with $|\mathcal{U}| \leq |\boldsymbol{\mathcal{X}}| + L$.

Proposition 8:

$$C^*_{\text{Super}} \geq \mathsf{K}^*_{\text{W}}.$$

If the source $\mathbf{X}$ is sufficiently symmetric, above inequality holds with equality.

III. Two-Users with One Cache

A. Setup

We now consider a two-user extension of the problem in Section II. Let us assume that user 1 has a cache with capacity C, while user 2 does not have a cache; see Figure 4. The library consists of the same L files $\mathbf{X}^n := (X_1^n, \dots, X_L^n)$ used in Section II, and communication again takes place in two phases — a *caching phase* and a *delivery phase*. Let $\mathcal{L}_1, \mathcal{L}_2 \subseteq \mathcal{L}$ denote those indices that can be potentially selected by users 1 and 2, respectively. That is, user k (for $k = 1, 2$) will request a file from $\left\{X^n_{\ell_k} : \ell_k \in \mathcal{L}_k\right\}$. Let $L_1 := |\mathcal{L}_1|$ and $L_2 := |\mathcal{L}_2|$.

A *two-user joint RDC code* with blocklength n consists of

(i) A *cache encoder*

$$f_{\text{c}} \colon \boldsymbol{\mathcal{X}}^n \to \mathcal{M}_{\text{c}}.$$

(ii) A *file encoder*

$$f_{(\ell_1,\ell_2)} \colon \boldsymbol{\mathcal{X}}^n \to \mathcal{M}, \quad (\ell_1, \ell_2) \in \mathcal{L}_1 \times \mathcal{L}_2.$$

(iii) A *user-1 file decoder*

$$g^{(1)}_{\ell_1,\ell_2} : \mathcal{M} \times \mathcal{M}_{\text{c}} \to \hat{\mathcal{X}}^{(1),n}_{\ell_1}, \quad (\ell_1, \ell_2) \in \mathcal{L}_1 \times \mathcal{L}_2.$$

(iv) A *user-2 file decoder*

$$g^{(2)}_{\ell_1,\ell_2} : \mathcal{M} \to \hat{\mathcal{X}}^{(2),n}_{\ell_2}, \quad (\ell_1, \ell_2) \in \mathcal{L}_1 \times \mathcal{L}_2.$$

Notice that we allow the decoders to depend on the demands of both users. We call the above collection of encoders and decoders an $(n, \mathcal{M}, \mathcal{M}_{\text{c}})$-two-user-code.

During the caching phase, the server pre-places the message $M_{\text{c}} := f_{\text{c}}(\mathbf{X}^n)$ in the cache of user 1. After the demands $(\ell_1, \ell_2) \in \mathcal{L}_1 \times \mathcal{L}_2$ are revealed to the server and both users, the server sends the message $M := f_{(\ell_1,\ell_2)}(\mathbf{X}^n)$ to both users. Users 1 and 2 respectively output

$$\hat{X}^{(1),n}_{\ell_1} := g^{(1)}_{\ell_1,\ell_2}(M, M_{\text{c}}) \quad \text{and} \quad \hat{X}^{(2),n}_{\ell_2} := g^{(2)}_{\ell_1,\ell_2}(M).$$

For convenience, we index user 1's reconstruction sequence only with its own demand ℓ_1; it can however also depend on user 2's demand ℓ_2. Similarly, for user 2's reconstruction.

The users might have differing exigencies regarding the files in the library. To account for this, we admit both users to measure reconstruction accuracy with different bounded per-letter distortion functions $\delta^{(1)}_{\ell_1} \colon \hat{\mathcal{X}}^{(1)}_{\ell_1} \times \mathcal{X}_{\ell_1} \to [0, \infty)$ and $\delta^{(2)}_{\ell_2} \colon \hat{\mathcal{X}}^{(2)}_{\ell_2} \times \mathcal{X}_{\ell_2} \to [0, \infty)$ (for indices $\ell_1 \in \mathcal{L}_1$ and $\ell_2 \in \mathcal{L}_2$).

Definition 3: Let C be a nonnegative real number, and let $\mathbf{D}^{(1)} := \left\{D^{(1)}_{\ell_1}\right\}_{\ell_1 \in \mathcal{L}_1}$ and $\mathbf{D}^{(2)} := \left\{D^{(2)}_{\ell_2}\right\}_{\ell_2 \in \mathcal{L}_2}$ be L_1- and L_2-tuples of nonnegative real numbers.

We say that a compression rate $R \geq 0$ is $(\mathbf{D}^{(1)}, \mathbf{D}^{(2)}, C)$-*admissible* if for any $\epsilon > 0$ there exists a sufficiently large blocklength n and an $(n, \mathcal{M}, \mathcal{M}_{\text{c}})$-code satisfying (2) and

$$\forall\ k \in \{1, 2\} \colon \forall\ \ell \in \mathcal{L}_k \colon$$
$$\mathbb{E}\left[\frac{1}{n}\sum_{i=1}^{n} \delta^{(k)}_{\ell_k}\left(\hat{X}^{(k)}_{\ell,i}, X_{\ell,i}\right)\right] \leq D^{(k)}_\ell + \epsilon. \quad (5)$$

Definition 4: The *two-user RDC function* is

$$\mathsf{R}_{\text{2user}}(\mathbf{D}^{(1)}, \mathbf{D}^{(2)}, C) := \inf\left\{R \geq 0 : R \text{ is } \left(\mathbf{D}^{(1)}, \mathbf{D}^{(2)}, C\right)\text{-admissible}\right\}.$$

B. Genie-Aided Lower Bound on the RDC Function

If both users' demands were revealed by a genie to the server even before the caching phase, our setup would coincide with a "worst-case" (or compound) successive-refinement setup. The rate-distortions function of this worst-demands successive refinement problem thus forms a lower bound on $\mathsf{R}_{\text{2user}}(\mathbf{D}^{(1)}, \mathbf{D}^{(2)}, C)$.

Definition 5: Let $\mathsf{R}^*_{\text{SuccRef}}(\mathbf{D}^{(1)}, \mathbf{D}^{(2)}, C)$ be the RDC function defined in (6) on top of the next page, where the minimum is taken over all tuples $(\mathbf{X}, \hat{\mathbf{X}}^{(1)}, \hat{\mathbf{X}}^{(2)})$ such that for $k \in \{1, 2\}$:

$$\forall\ \ell \in \mathcal{L}_k \colon \quad \mathbb{E}\left[\delta^{(k)}_\ell\left(\hat{X}^{(k)}_\ell, X_\ell\right)\right] \leq D^{(k)}_\ell. \quad (7)$$

$$\mathsf{R}^*_{\text{SuccRef}}\big(\mathbf{D}^{(1)}, \mathbf{D}^{(2)}, C\big) := \max_{(\ell_1,\ell_2)\in\mathcal{L}_1\times\mathcal{L}_2} \min_{P_{\mathbf{X},\hat{\mathbf{X}}^{(1)},\hat{\mathbf{X}}^{(2)}}} \max\Big\{ I\big(\mathbf{X};\hat{X}^{(2)}_{\ell_2}\big),\ I\big(\mathbf{X};\hat{X}^{(1)}_{\ell_1},\hat{X}^{(2)}_{\ell_2}\big) - C \Big\} \tag{6}$$

$$\mathsf{R}_{\text{2user,Ach}}\big(\mathbf{D}^{(1)}, \mathbf{D}^{(2)}, C\big) := \min \max_{(\ell_1,\ell_2)\in\mathcal{L}_1\times\mathcal{L}_2} \max\Big\{ I\big(\mathbf{X};\hat{X}^{(2)}_{\ell_2}\big) + I\big(\mathbf{X};\hat{X}^{(1)}_{\ell_1}|U,\hat{X}^{(2)}_{\ell_2}\big),\ I\big(\mathbf{X};U,\hat{X}^{(1)}_{\ell_1},\hat{X}^{(2)}_{\ell_2}\big) - C \Big\} \tag{8}$$

$$\mathsf{R}_{\text{2user}}(\mathbf{0},\mathbf{0},C) \le \min_{P_{U|\mathbf{X}}} \max_{(\ell_1,\ell_2)\in\mathcal{L}_1\times\mathcal{L}_2} \max\Big\{ H(X_{\ell_2}) + H(X_{\ell_1}|U,X_{\ell_2}),\ H(U,X_{\ell_1},X_{\ell_2}) - C \Big\} \tag{9}$$

Theorem 2:

$$\mathsf{R}_{\text{2user}}\big(\mathbf{D}^{(1)}, \mathbf{D}^{(2)}, C\big) \ge \mathsf{R}^*_{\text{SuccRef}}\big(\mathbf{D}^{(1)}, \mathbf{D}^{(2)}, C\big).$$

C. Upper Bound on the RDC Function

We have the following upper bound on the RDC function.

Definition 6: Let $\mathsf{R}_{\text{2user,Ach}}(\mathbf{D}^{(1)}, \mathbf{D}^{(2)}, C)$ be defined as in (8) on top of the next page, where the minimum is taken over all tuples $\big(U,\, \hat{\mathbf{X}}^{(1)}{:=}\{\hat{X}^{(1)}_{\ell_1}\}_{\ell_1\in\mathcal{L}_1}, \hat{\mathbf{X}}^{(2)}{:=}\{\hat{X}^{(2)}_{\ell_2}\}_{\ell_2\in\mathcal{L}_2}\big)$ jointly distributed with $\mathbf{X}$ such that (7) holds for $k \in \{1,2\}$.

Theorem 3:

$$\mathsf{R}_{\text{2user}}\big(\mathbf{D}^{(1)}, \mathbf{D}^{(2)}, C\big) \le \mathsf{R}_{\text{2user,Ach}}\big(\mathbf{D}^{(1)}, \mathbf{D}^{(2)}, C\big).$$

Theorem 3 can equivalently be stated as follows: a rate $R > 0$ is $(\mathbf{D}^{(1)}, \mathbf{D}^{(2)}, C)$-admissible whenever there is a tuple $\big(U,\, \hat{\mathbf{X}}^{(1)}, \hat{\mathbf{X}}^{(2)}\big)$ and a collection of auxiliary rates $\{\tilde{R}_{\ell_2}\}_{\ell_2\in\mathcal{L}_2}$ such that for every pair $(\ell_1,\ell_2) \in \mathcal{L}_1 \times \mathcal{L}_2$:

$$C + \tilde{R}_{\ell_2} \ge I\big(U;\mathbf{X},\hat{X}^{(2)}_{\ell_2}\big) - I\big(U;\hat{X}^{(2)}_{\ell_2}\big) = I\big(U;\mathbf{X}\big|\hat{X}^{(2)}_{\ell_2}\big)$$
$$R - \tilde{R}_{\ell_2} \ge I\big(\mathbf{X};\hat{X}^{(2)}_{\ell_2}\big) + I\big(\mathbf{X};\hat{X}^{(1)}_{\ell_1}|U,\hat{X}^{(2)}_{\ell_2}\big).$$

These rates are achieved by the following scheme. The server compresses the entire library $\mathbf{X}^n$ into U^n using the *adaptive conditional RD code* for side-information $\hat{X}^n_{\ell_2}$ that we describe in the next paragraph. Our adaptive RD code produces a first message of nC bits which the server stores in user 1's cache, and a second message of $n\tilde{R}_{\ell_2}$ bits which the server sends as part of the delivery message. In the delivery message it also sends a standard RD message that allows both users to reconstruct $\hat{X}^{(2),n}_{\ell_2}$, and a standard conditional RD message that allows user 1 to reconstruct $\hat{X}^{(1),n}_{\ell_1}$ given that it already knows $(U^n, \hat{X}^{(2),n}_{\ell_2})$. Both users first reconstruct $\hat{X}^{(2),n}_{\ell_2}$. User 1 subsequently reconstructs U^n and $\hat{X}^{(1),n}_{\ell_1}$, always using previously reconstructed sequences as side-information.

Our adaptive conditional RD code uses a codebook $\mathcal{C} := \{U^n(m_u)\}$ with a nested binning structure: it contains $\approx 2^{nC}$ outer bins that each consist of $\approx 2^{n\tilde{R}_{\ell_2}}$ inner bins. The outer binning rate C is fixed in advanced; the inner binning rate however adapts to the quality of the side-information $\hat{X}^{(2),n}_{\ell_2}$ and is fixed only after the demand ℓ_2 is revealed. Encoding is in two steps. In a first step the server picks the unique codeword $U^n(m^*_u)$ that for every $\ell_2 \in \mathcal{L}_2$ is jointly typical with the pair $\big(\mathbf{X}^n, \hat{X}^{(2),n}_{\ell_2}\big)$. The outer bin index of $U^n(m^*_u)$ is immediately available and the server stores the nC bits representing this index in user 1's cache. Once the demand ℓ_2 is fixed, also the inner bin index is available and the server sends it as part of the delivery message. Decoding is standard using both bin indices and the side-information $\hat{X}^{(2),n}_{\ell_2}$.

D. Almost Lossless Reconstructions

Let now both users reconstruct their demanded files $X^n_{\ell_1}$ and $X^n_{\ell_2}$ (almost) losslessy. From Theorem 3:

Corollary 3.1: The RC-function for the lossless setup satisfies the upper bound in (9) on top of this page.

Corollary 3.2: Bound (9) holds with equality when

1) $\mathcal{L}_1 = \mathcal{L}_2 = \{\ell, \ell'\}$ for $\ell, \ell' \in \mathcal{L}$;
2) $\mathcal{L}_1 = \{\ell\}$ for some $\ell \in \mathcal{L}$; or
3) $\mathcal{L}_2 = \{\ell\}$ for some $\ell \in \mathcal{L}$.

Proof: To prove cases 1.) and 2.), specialise the lower bound in Theorem 2 to the lossless case and to $U = (X_\ell, X_{\ell'})$ and $U = X_\ell$, respectively. For case 3.) a new converse is required. ∎

Interestingly, in the first two cases there is no penalty for not knowing the demands during the caching phase.

Acknowledgements

The work of R. Timo was supported by the Alexander von Humboldt Foundation. The work of S. Saeedi Bidokhti was supported by the Swiss National Science Foundation fellowship no. 158487.

References

[1] M. A. Maddah-Ali and U. Niesen, "Fundamental limits of caching," *IEEE Trans. Inform. Theory*, vol. 60, no. 5, pp. 2856–2867, 2014.
[2] U. Niesen and M. A. Maddah-Ali, "Coded caching with nonuniform demands," *arXiv:1308.0178v3*, 2013.
[3] S. Sahraei and M. Gastpar, "K users caching two files: an improved achievable rate", *online* http://arxiv.org/abs/1512.06682v1.
[4] Z. Chen, P. Fan, and K. Ben Letaief, "Fundamental limits of caching: Improved bounds for small buffer users," *online* http://arXiv.org/abs/1407.1935v2.pdf, Nov. 2015.
[5] H. Ghasemi and A. Ramamoorthy, "Improved lower bounds for coded caching," *online* http://arxiv.org/abs/1501.06003.
[6] A. Sengupta, R. Tandon, and T. C. Clancy, "Improved approximation of storage-rate tradeoff for caching via new outer bounds," in *IEEE Intl. Symp. Inform. Theory*, Hong Kong, China, 2015.
[7] C. Tian, "A note on the fundamental limits of coded caching," *online* http://arXiv.org/abs/1503.00010.
[8] R. Timo and M. Wigger, "Joint cache-channel coding over erasure broadcast channels," in *IEEE Intl. Symp. Wireless Commun. Systems*, Brussels, Belgium, 2015.
[9] C. Y. Wang, S. H. Lim, and M. Gastpar, "Information-theoretic caching: sequential coding for computing," *arXiv:1504.00553*, 2015.
[10] R. Gray and A. Wyner, "Source coding for a simple network," *Bell Sys. Tech. J.*, vol. 53, no. 9, pp. 1681–1721, 1974.
[11] W. Liu, G. Xu, and B. Chen, "The common information of N dependent random variables," in *Allerton Conf. Commun. Control Comp.*, 2010.
[12] R. Tandon, L. Sankar, and H.V. Poor, "Multi-user privacy: The Gray-Wyner system and generalized common information" in *Proc. IEEE Int. Sym. on Information Theory*, 2011