Full Coded Caching Gains for Cache-less Users

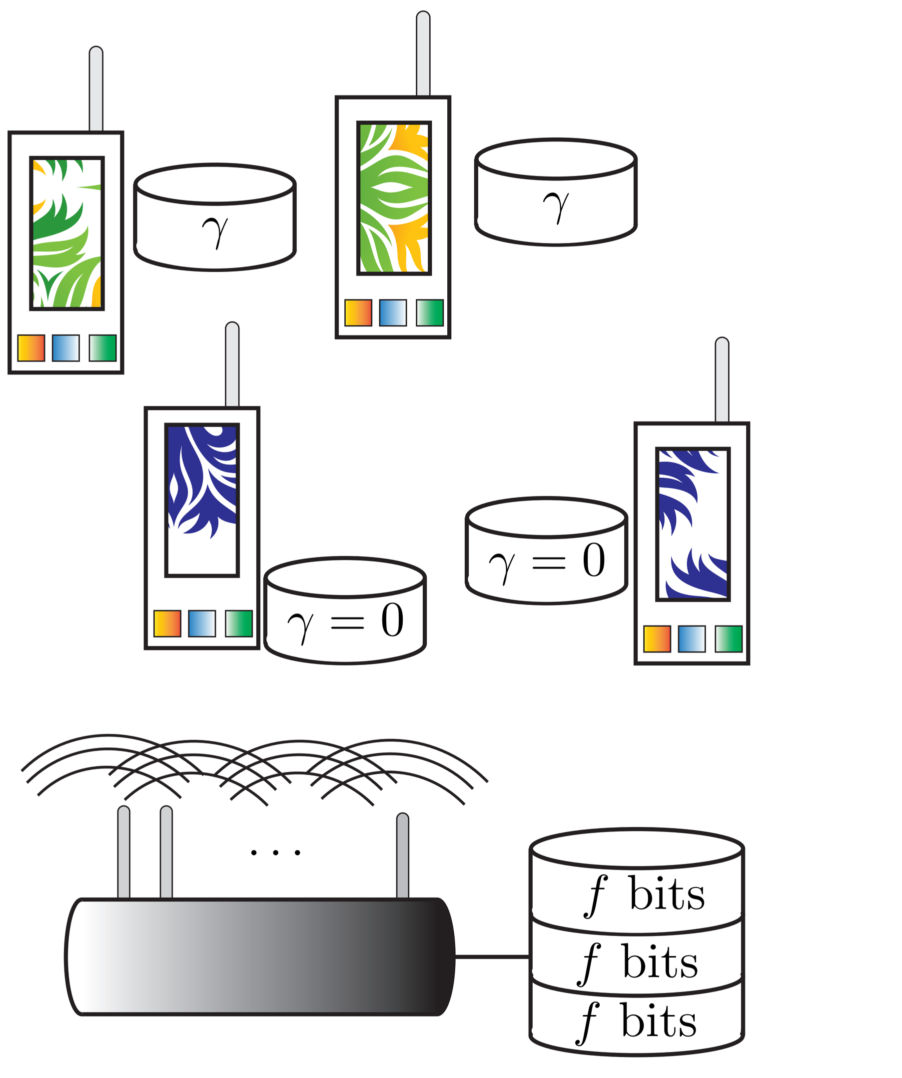
Eleftherios Lampiris, Petros Elia

EURECOM, Sophia Antipolis, France

lampiris, elia, (@eurecom.fr)

Setting

1. K_c **Cache-aided users:** Cache $\gamma > 0$.
2. K_n **Cache-less users:** No cache.
3. **Multiple Antennas:** L-MISO Broadcast fully-connected Channel.
4. **Content:** Users request files from the same library of N files.



Key Questions

- Can cache-less users experience Coded Caching performance?

  Fundamental Obstacle: In the single-antenna setting they can't.
- What are the fundamental limits?
- Can we transmit to both types simultaneously?

  Obstacle: Cache-less users cannot decode XORed messages.
- Does adding users *hurt* the theoretically optimal DoF performance of cache-aided users, i.e. $d_\Sigma = K_c\gamma + L$?

  Fundamental Obstacle: In the single-antenna setting it does.

Scheme Description

Placement

Placement follows the MN [1] paradigm, i.e.

$$Z_k \leftarrow \{W_n^\tau : k \in \tau,\ \tau \subset [K_c],\ |\tau| = K_c\gamma,\ \forall n \in [N]\}$$

Delivery

1. Create a vector of L elements.

$$\begin{bmatrix} s_1 \\ s_2 \\ \vdots \\ s_L \end{bmatrix}$$

2. First element is composed of XORed subfiles for $K_c\gamma + 1$ cache-aided users.
3. The rest $L-1$ elements are subfiles for cache-less users.

$$\begin{bmatrix} \bigoplus_{k\in\chi} W_{d_k}^{\chi\setminus\{k\}} \\ W_{d_p}^\tau \\ \vdots \\ W_{d_q}^\tau \end{bmatrix}$$

4. Multiply by a precoder to separate all the cache-less and one of the cache-aided users.

$$\mathbf{x}_{k,p_2,\dots,p_L} = \mathcal{H}^{-1}_{p_1,p_2,\dots,p_L} \begin{bmatrix} \bigoplus_{k\in\chi} W_{d_k}^{\chi\setminus\{k\}} \\ W_{d_{p_2}}^\tau \\ \vdots \\ W_{d_{p_L}}^\tau \end{bmatrix}$$

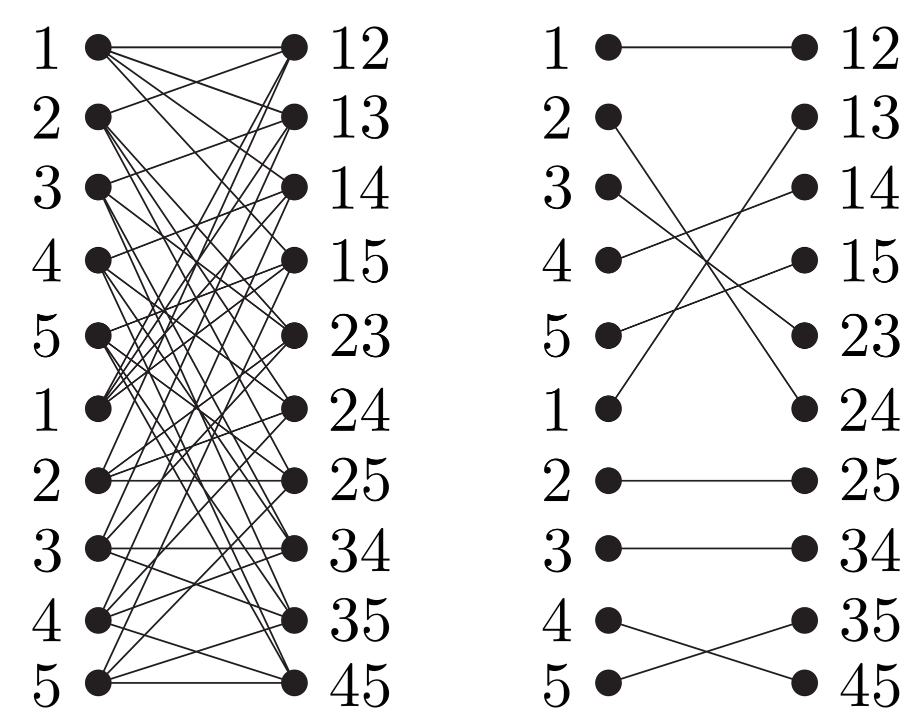
Decoding

- Cache-less users ($p_2 - p_L$) receive their subfiles via ZF-precoding.
- Cache-aided user p_1 receives the XOR and decodes a la Maddah-Ali - Niesen.
- Remaining $K_c\gamma$ users receive a linear combination of all subfiles. To decode they need to have cached all $K_c\gamma + L - 1$ subfiles.

Combinatorial Matching Challenge

- To achieve decoding, we need to pair the subfile indices with a transmission index.
- This creates a perfect matching in a bipartite graph and is combinatorially hard to solve.
- We solve it in $\mathcal{O}(1)$ time using a novel approach.

Matching Example



Scheme Example

Assume : $K_c = 5, \quad \gamma = \frac{1}{5}, \quad K_n = 2, \quad L = 2$

Placement

$$Z_1 = \{A_1, B_1, ..., G_1\}, \quad Z_2 = \{A_2, B_2, ..., G_2\},$$
$$Z_3 = \{A_3, B_3, ..., G_3\}, \quad Z_4 = \{A_4, B_4, ..., G_4\},$$
$$Z_5 = \{A_5, B_5, ..., G_5\}, \quad Z_6 = Z_7 = \emptyset.$$

Delivery

$$\mathbf{x}_{126} = \mathcal{H}^{-1}_{26}\begin{bmatrix} A_2 \oplus B_1 \\ F_1 \end{bmatrix}, \quad \mathbf{x}_{137} = \mathcal{H}^{-1}_{37}\begin{bmatrix} A_3 \oplus C_1 \\ G_1 \end{bmatrix},$$
$$\mathbf{x}_{146} = \mathcal{H}^{-1}_{16}\begin{bmatrix} A_4 \oplus D_1 \\ F_4 \end{bmatrix}, \quad \mathbf{x}_{157} = \mathcal{H}^{-1}_{17}\begin{bmatrix} A_5 \oplus E_1 \\ G_5 \end{bmatrix},$$
$$\mathbf{x}_{236} = \mathcal{H}^{-1}_{26}\begin{bmatrix} B_3 \oplus C_2 \\ F_3 \end{bmatrix}, \quad \mathbf{x}_{246} = \mathcal{H}^{-1}_{46}\begin{bmatrix} B_4 \oplus D_2 \\ F_2 \end{bmatrix},$$
$$\mathbf{x}_{257} = \mathcal{H}^{-1}_{57}\begin{bmatrix} B_5 \oplus E_2 \\ G_2 \end{bmatrix}, \quad \mathbf{x}_{347} = \mathcal{H}^{-1}_{47}\begin{bmatrix} C_4 \oplus D_3 \\ G_3 \end{bmatrix},$$
$$\mathbf{x}_{356} = \mathcal{H}^{-1}_{36}\begin{bmatrix} C_5 \oplus E_3 \\ F_5 \end{bmatrix}, \quad \mathbf{x}_{457} = \mathcal{H}^{-1}_{57}\begin{bmatrix} D_5 \oplus E_4 \\ G_4 \end{bmatrix}.$$

$\mathcal{H}^{-1}_{i,j}$ precoder to users i, j

Main Results

Theorem 1 *In a single antenna setting with K_c cache-aided users with normalized cache γ and K_n cache-less users the delivery time*

$$T = \frac{K_c(1-\gamma)}{1+K_c\gamma} + K_n$$

is exactly optimal.

Theorem 2 *In the MISO BC with L antennas, K_c cache-aided users, fractional cache size γ, and $K_n \geq (L-1)T_1$ cache-less users, the delivery time*

$$T = \frac{(L-1)T_1 + K_c(1-\gamma)}{K_c\gamma + L} + \frac{K_n - (L-1)T_1}{\min\{L, K_n - (L-1)T_1\}}$$

is achievable and within a factor of 2 from optimal, while if $K_n \leq (L-1)T_1(K_c,\gamma)$ then

$$T = \frac{K_c(1-\gamma)}{K_c\gamma + L} + \boxed{\frac{K_n}{K_c\gamma + L}}$$

is achievable and within a factor of 3 from optimal.

Corollary 1 - *Cache-less users with Coded Caching Gains*

Setting: $K_c, \gamma, K_n \leq (L-1)\frac{K_c(1-\gamma)}{1+K_c\gamma}, L.$
Result: *Cache-less users have DoF $K_c\gamma + L$, i.e. experience caching gains.*
Example: *See previous example.*

Corollary 2 - *Free Cache-less Users*

Setting: $K_c, \gamma.$
Result: *Every time you add an antenna you can serve **for free** $\frac{K_c(1-\gamma)}{1+K_c\gamma}$ cache-less users.*
Example: $K_c = 300, \gamma = 2/100$

Every new antenna can serve 42 extra cache-less users for free.

Corollary 3 - *L-fold Boost*

Setting: $K_c, \gamma, K_n = (L_1 - 1)\frac{K_c(1-\gamma)}{1+K_c\gamma}.$
Result: *Going from 1 to $L \leq L_1$ antennas, reduces delay by L times.*
Example: $K_c = 101, \gamma = 1/101, K_n = 300$

L	1	2	3	4	5	6	7
T	350	175	81.6	87.5	70	40.8	50

Corollary 4 - *Add Cache Users Cheaply*

Setting: $K_n, L.$
Result: *Serve infinite amount of K_c with $\gamma \geq \frac{L}{K_n}$, while delay increases by a factor of $\frac{L}{L-1}$.*
Example: $K_n = 300, L = 11.$
Add arbitrary K_c with $\gamma = 0.037$. Delay increases by 10%, number of users can increase infinitely.

Corollary 5 - *Even-out Assymetry*

Setting: $K_c, \gamma, K_n \leq (L-1)\frac{K_c(1-\gamma)}{1+K_c\gamma}, L.$
Result: *System performs as the equivalent with average cache of $\gamma_{av} = \frac{K_c\gamma}{K_c+K_n}$.*
Example: *See example in previous column, where system performs as if every user had cache $\frac{1}{7}$.*

Benefits of Combining Antennas with Coded Caching

- L-fold increase in Coded Caching Gains [2]
- Caching gains without CSIT cost [3]
- Free users with more antennas (this work)
- Cache-less users with CC gains (this work)

References

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

[2] E. Lampiris and P. Elia, "Adding transmitters dramatically boosts Coded-Caching gains for finite file sizes," *Journal of Selected Areas in Communications (JSAC)*, 2018.

[3] E. Lampiris and P. Elia, "Achieving full multiplexing and unbounded caching gains with bounded feedback resources," *Int. Symp. of Inf. Theory (ISIT)*, 2018.