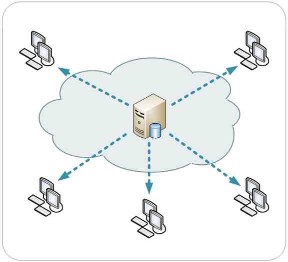
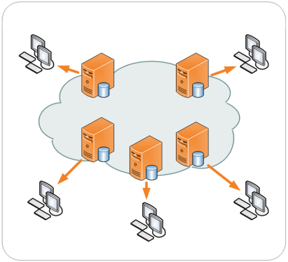
An Information-Theoretic View of Cache-Aided Networks: Part 1 – Coded Caching

Michèle Wigger

Telecom ParisTech, 9 february 2021

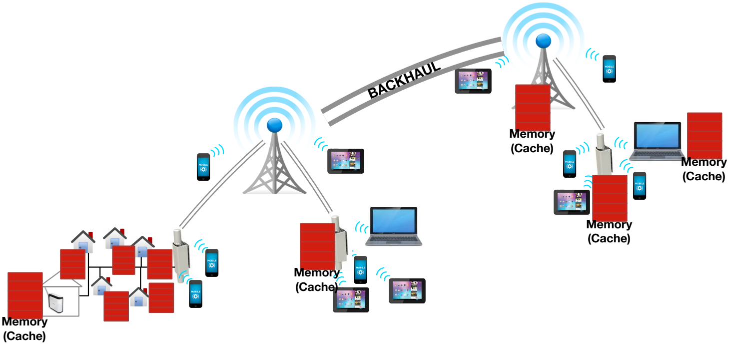
Content Delivery Networks





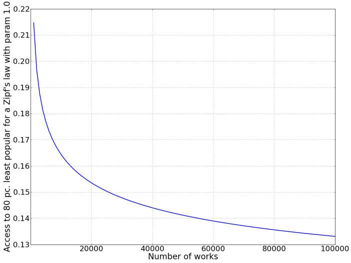
- Store contents in caches before file demands even known
- Reduce network load and latency during high-congestion periods
- Idea useful if certain files very popular and known in advance

Distributed Caches: Promising Solution for Cellular Networks



- Can cache at main BSs, picoBSs, femtoBSs, or directly at end users

File Popularities



- Static file popularity follows a Zipf distribution $P(x) = Cx^{-\alpha}$
- Evolution of file popularities (youtube videos) can also be predicted

Use pro-active caching to improve cellular systems!

- decrease network load
- decrease latency

Information-Theoretic View of Caching

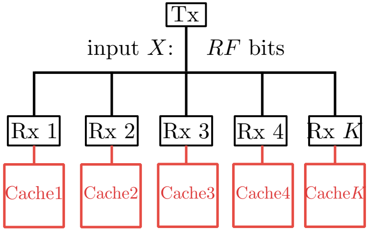
- Go beyond obvious *local* caching gain
- Create coding opportunities through smart cache placement
- Exploit multi-cast opportunities

  $\rightarrow$ serve many users/demands with same signals
- Global caching gain

  $\rightarrow$ receivers can profit from other receivers' cache memories

A Simple Network

Library: Files $W_1, W_2, \ldots, W_N$ of F bits each

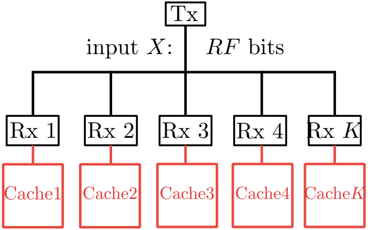


- All files equally popular $\rightarrow$ we consider only most popular files
- All files equally large.
- Before the actual transmission there is an idle period where the transmitter (server) can fill the receivers' cache memories.
- Cache placement phase only constrained by memory sizes.

[1] M. A. Maddah-Ali, U. Niesen, "Fundamental Limits of Caching." *IEEE Transactions on Information Theory.*

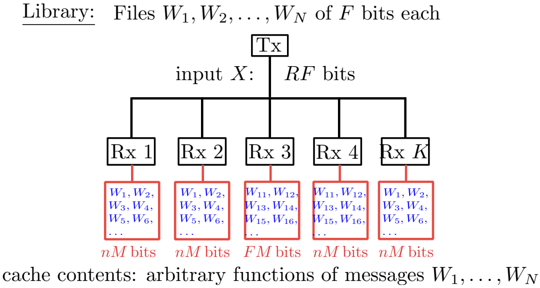
A Simple Network

Library: Files $W_1, W_2, \ldots, W_N$ of F bits each



Communication in two phases:

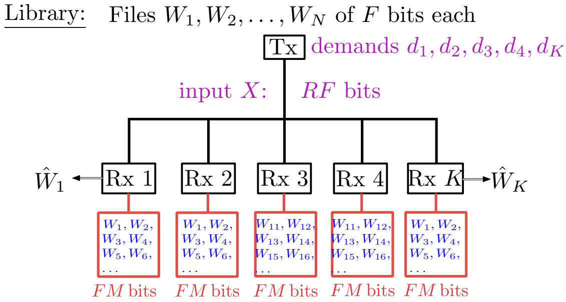
A Simple Network



Communication in two phases:

- Placement phase: Tx fills caches without knowing which receiver demands which message

A Simple Network



Communication in two phases:

- Placement phase: Tx fills caches without knowing which receiver demands which message
- Delivery phase:
  - Receiver k wants file W_{d_k} $\rightarrow$ sends demand d_k to transmitter
  - Tx describes $W_{d_1}, \ldots, W_{d_K}$ to Rxs $1, \ldots, K$ through input X
  - Tx describes demands $d_1, \ldots, d_K$ to all receivers

Fundamental Rate-Memory Tradeoff $R^\star(M)$

$$R^\star(M) \quad := \quad \min\Big\{R\colon \text{ such that for } (R,M) \text{ goal can be achieved} \\ \text{for } \textit{all demands } d_1,\ldots,d_K.\Big\}$$

Some properties:

- $R^\star(M)$ is decreasing in M.
- $R^\star(M)$ is bounded above by $\min\{N,K\}$. Moreover:

$$R^\star(M=0) = \min\{N,K\}.$$

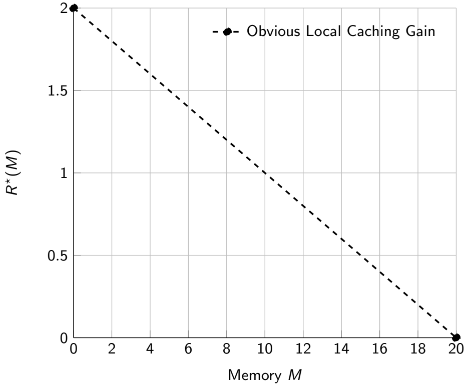
- $R^\star(M)$ is nonnegative. Moreover:

$$R^\star(M) = 0, \qquad \forall M \geq N.$$

Obvious Upper Bound on $R^\star(M)$

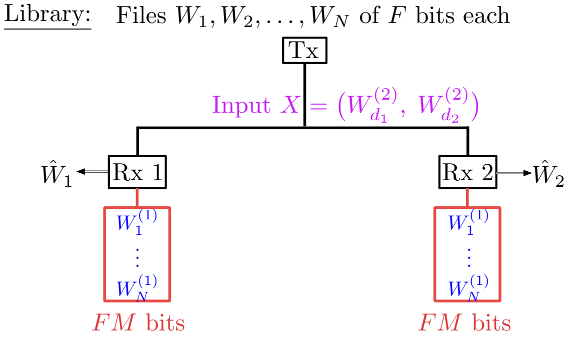
$$R^\star(M) \le \min\{K, N\}\Big(1 - \frac{M}{N}\Big).$$

- Example with $N = 20$ files and $K = 2$ users



- Achieved through time/memory sharing or by the following naive scheme...

Naive Scheme for $K = 2$ Receivers



- Split W_d into two parts $(W_d^{(1)}, W_d^{(2)})$ of sizes $\frac{FM}{N}$ and $F\left(1 - \frac{M}{N}\right)$ bits
- For $d = 1, \ldots, N$: cache part $W_d^{(1)}$ at both Rxs 1 and 2
- Delivery input $X = \left(W_{d_1}^{(2)},\ W_{d_2}^{(2)}\right)$ (if $d_1 \neq d_2$)

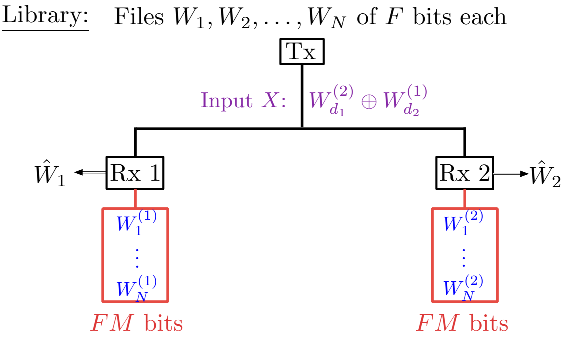
In the worst-case, delivery rate needs to be $R = 2\left(1 - \frac{M}{N}\right)$.

Naive Scheme for K Receivers

- Split W_d into two parts $(W_d^{(1)}, W_d^{(2)})$ of sizes $F\frac{M}{N}$ and $F\left(1-\frac{M}{N}\right)$ bits
- For $d = 1, \ldots, N$: cache part $W_d^{(1)}$ *at all* rxs
- Deliver part $W_d^{(2)}$ for each demanded message W_d.
  - If $K \geq N$, in the worst case:
  $$X = (W_{d_1}^{(2)},\ W_{d_2}^{(2)}, \ldots,\ W_{d_K}^{(2)}).$$
  - If $K < N$, in the worst case:
  $$X = (W_1^{(2)},\ W_2^{(2)}, \ldots,\ W_N^{(2)}).$$

Required Delivery Rate is $\quad R = \min\{K, N\} \cdot \left(1 - \frac{M}{N}\right).$

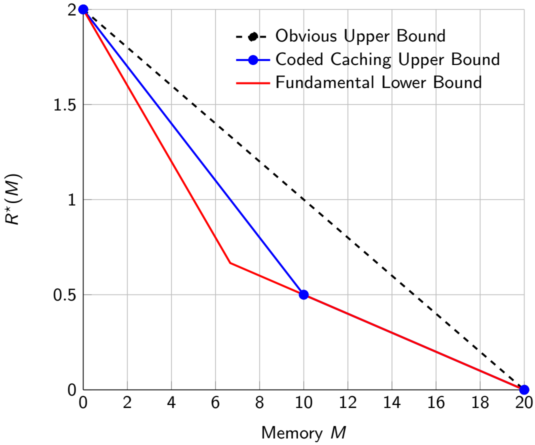
Coded caching for $K = 2$ Receivers [Maddah-Ali&Niesen 2014]



- Split W_d into two parts $(W_d^{(1)}, W_d^{(2)})$ each of $\frac{F}{2}$ bits
- For $d = 1, \ldots, N$: cache part $W_d^{(1)}$ at Rx1 and part $W_d^{(2)}$ at Rx2
- Delivery input $X = W_{d_1}^{(2)} \oplus W_{d_2}^{(1)}$

Achieves Rate-Memory Pair $\quad M = \frac{N}{2} \quad$ and $\quad R = \frac{1}{2}$.

Fundamental Limit and Bounds for 2 Users ($N = 20$ files)



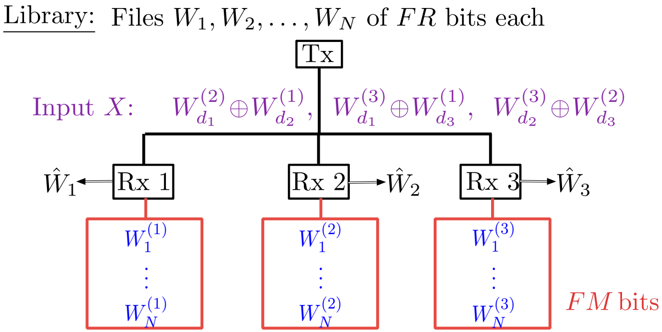
Time- and Memory Sharing for arbitrary parameter $\alpha \in [0,1]$

- Assume Scheme 1 achieves (R_1, M_1) and Scheme 2 achieves (R_2, M_2)
- Split each file $W_d = (W_d^{(1)}, W_d^{(2)})$ consisting of αF and $(1-\alpha)F$ bits each.
- Apply placements of Scheme 1 to $\{W_d^{(1)}\}$ using $M_1 \alpha F$ bits of memory and placements of Scheme 2 to $\{W_d^{(2)}\}$ using $M_2(1-\alpha)F$ bits of memory
- Apply delivery of Scheme 1 to $\{W_d^{(1)}\} \to$ signal $X^{(1)}$ of $R_1 \alpha F$ bits; and apply delivery of Scheme 2 to $\{W_d^{(2)}\} \to$ signal $X^{(2)}$ of $R_2(1-\alpha)F$ bits

Total cache memory $MF = M_1\alpha F + M_2(1-\alpha)F$; and
total number of delivery bits $RF = R_1 \alpha F + R_2(1-\alpha)F$

$$\Rightarrow (\alpha M_1 + (1-\alpha)M_2,\ \alpha R_1 + (1-\alpha)R_2) \quad \text{is achievable } \forall \alpha \in [0,1]$$

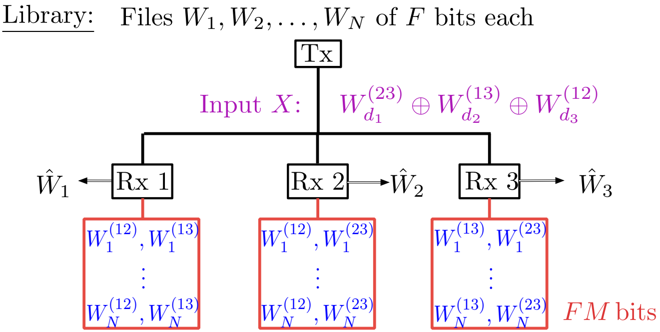
Coded caching for $K = 3$ Receivers, Parameter $t = 1$



- Split W_d into three parts $(W_d^{(1)}, W_d^{(2)}, W_d^{(3)})$ each $\frac{F}{3}$ bits
- For $d = 1, \ldots, N$: cache part $W_d^{(1)}$ at Rx1, part $W_d^{(2)}$ at Rx2, and part $W_d^{(3)}$ at Rx3
- Delivery input $X = W_{d_1}^{(2)} \oplus W_{d_2}^{(1)},\ W_{d_1}^{(3)} \oplus W_{d_3}^{(1)},\ W_{d_2}^{(3)} \oplus W_{d_3}^{(2)}$

Achieves Rate-Memory Pair $M = \frac{N}{3}$ and $R = 1$.

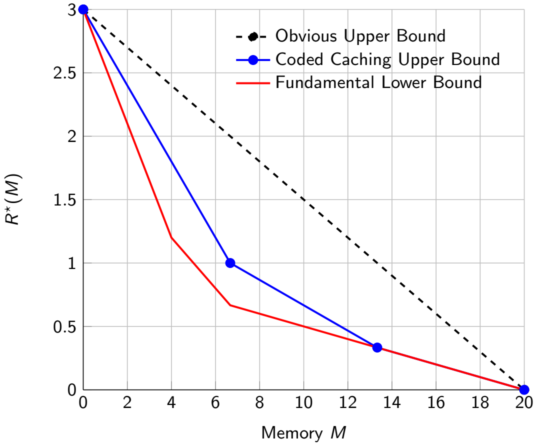
Coded caching for $K = 3$ Receivers, Parameter $t = 2$



- Split W_d into three parts $(W_d^{(12)}, W_d^{(13)}, W_d^{(23)})$ each of $\frac{F}{3}$ bits
- For $d = 1, \ldots, N$: cache part $W_d^{(12)}$ at Rxs 1 and 2, part $W_d^{(13)}$ at Rxs 1 and 3, and part $W_d^{(23)}$ at Rxs 2 and 3
- Delivery input $X = W_{d_1}^{(23)} \oplus W_{d_2}^{(13)} \oplus W_{d_3}^{(12)}$

Achieves Rate-Memory Pair $\quad M = \frac{2N}{3} \quad$ and $\quad R = \frac{1}{3}$.

Bounds for 3 Users ($N = 20$ files)



Coded Caching for K Users

- Parameter $t \in \{1, \dots, K-1\}$
- *Placement:* Split each W_d into $\binom{K}{t}$ parts and save each part at a different subset of receivers
  Let for each size-t subset $\mathcal{G}$ denote $W_d^{\mathcal{G}}$ the part of W_d placed in caches of all receivers in $\mathcal{G}$.
- *Delivery transmission:* For each set $\mathcal{S} = \{s_1, \dots, s_{t+1}\}$ of size $t+1$, send

$$W_{\text{XOR},\mathcal{S}} := \bigoplus_{\ell=1}^{t+1} W_{d_{s_\ell}}^{(\mathcal{S}\setminus\{s_\ell\})}$$

- *Delivery reception:* Receiver s_j has stored in its cache memory

$$W_{d_{s_\ell}}^{(\mathcal{S}\setminus\{s_\ell\})}, \qquad \forall \ell \in \{1, \dots, j-1, j+1, \dots, t\}.$$

  So, with $W_{\text{XOR},\mathcal{S}}$ it can recover $W_{d_{s_j}}^{(\mathcal{S}\setminus\{s_j\})}$.

  This way it can recover all missing parts of $W_{d_{s_j}}$.

Analysis of Coded Caching for K Users

- Fix parameter $t \in \{1, \dots, K-1\}$
- Each part of a file is of size

$$F \cdot \binom{K}{t}^{-1} \quad \text{bits.}$$

- Each receiver stores $\binom{K-1}{t-1}$ parts of each file. So the placement requires cache memory

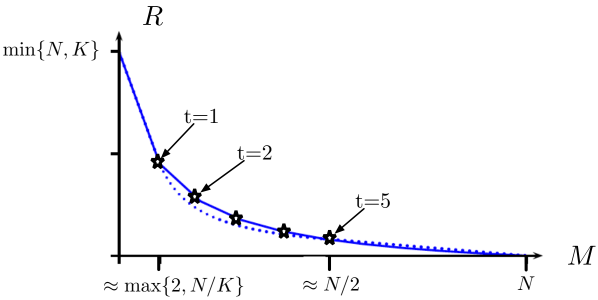
$$M = N \frac{\binom{K-1}{t-1}}{\binom{K}{t}} = N \frac{t}{K}. \qquad \text{(increasing in } t\text{)}$$

- Coded caching sends an XOR-message to each subset of $t+1$ receivers. So the total rate is

$$R = \frac{\binom{K}{t+1}}{\binom{K}{t}} = \frac{K-t}{t+1}. \qquad \text{(decreasing in } t\text{)}$$

Performance of Coded Caching

- $K = 6$



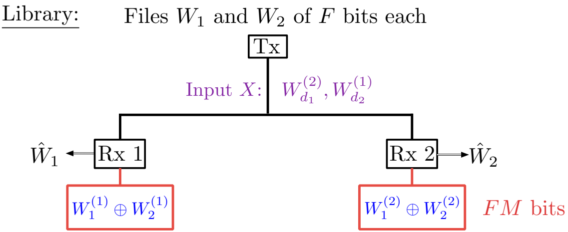
Coded Caching Upper Bound

For all $M \in \frac{N}{K} \cdot \{0, 1, \ldots, K-1, K\}$:

$$R^\star(M) \leq \min\left\{K\left(1 - \frac{M}{N}\right)\left(1 + \frac{MK}{N}\right)^{-1},\ N\left(1 - \frac{M}{N}\right)\right\}.$$

Achievability can be Improved!

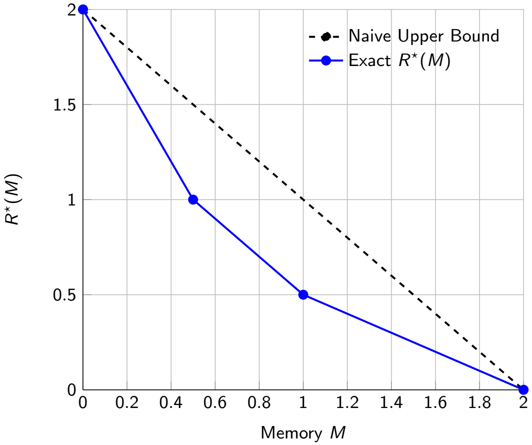
Example: $K = 2$ and $N = 2$:



- Split W_d into two parts $(W_d^{(1)}, W_d^{(2)})$ each of $\frac{F}{2}$ bits
- For $d = 1, \ldots, N$: cache part $W_d^{(1)}$ at Rx1 and part $W_d^{(2)}$ at Rx2
- Delivery input $X = W_{d_1}^{(2)}, W_{d_2}^{(1)}$

Achieves Rate-Memory Pair $\quad M = \frac{1}{2} \quad$ and $\quad R = 1.$

Exact Rate-Memory Tradeoff $R^{\star}(M)$ for $K = N = 2$



First Lower Bound $R \geq s - \frac{s}{\lfloor N/s \rfloor} M$

- Consider only Receivers $1, \ldots, s$, where $s \leq \min\{N, K\}$
- Consider demand vectors

$$\begin{aligned} \mathbf{d}^{(1)} &:= (1, \ldots, s) \\ \mathbf{d}^{(2)} &:= (s+1, \ldots, 2s) \\ &\ldots \\ \mathbf{d}^{(\lfloor N/s \rfloor)} &:= ((\lfloor N/s \rfloor - 1)s + 1, \ldots, \lfloor N/s \rfloor s) \end{aligned}$$

- Let $X^{(\ell)}$ denote the delivery input for demand $\mathbf{d}^{(\ell)}$
- From $X^{(1)}, \ldots, X^{(\lfloor N/s \rfloor)}$ and $Z_1, \ldots, Z_s$ one can calculate $W_1, \ldots, W_{\lfloor N/s \rfloor s}$:

$$\begin{aligned} & H\big(X^{(1)}, \ldots, X^{(\lfloor N/s \rfloor)}, Z_1, \ldots, Z_s\big) \geq H(W_1, \ldots, W_{\lfloor N/s \rfloor s}) \\ \iff \quad & H\big(X^{(1)}, \ldots, X^{(\lfloor N/s \rfloor)}\big) + H\big(Z_1, \ldots, Z_s\big) \geq H(W_1, \ldots, W_{\lfloor N/s \rfloor s}) \\ \iff \quad & FR\lfloor N/s \rfloor + sFM \geq \lfloor N/s \rfloor sF \\ \iff \quad & R \geq s - \frac{s}{\lfloor N/s \rfloor} M. \end{aligned}$$

Second Lower Bound $R \geq s - \frac{s^2}{N} M$

- Consider only Receivers $1, \ldots, s$, where $s \leq \min\{N, K\}$
- $\mathcal{D}_s$: all demand vectors $\mathbf{d}_s$ of s users having s different demands.

$$|\mathcal{D}_s| = \binom{N}{s} s!$$

- Let $X^{(\mathbf{d}_s)}$ denote the delivery input for demand $\mathbf{d}_s$
- $\forall \mathbf{d}_s \in \mathcal{D}_s$ it holds that:

$$\begin{aligned} F \cdot R &\geq H(X^{\mathbf{d}_s}) \geq I(X^{\mathbf{d}_s}; W_{d_1}, \ldots, W_{d_s}, Z_1, \ldots, Z_s) \\ &\geq I(X^{\mathbf{d}_s}; W_{d_1}, \ldots, W_{d_s} | Z_1, \ldots, Z_s) \\ &= H(W_{d_1}, \ldots, W_{d_s} | Z_1, \ldots, Z_s). \end{aligned}$$

- By averaging over all demands $\mathbf{d}_s \in \mathcal{D}_s$ and by Han's inequality:

$$\begin{aligned} FR &\geq \frac{s}{N} \Big(H(W_1, \ldots, W_N) - I(W_1, \ldots, W_N; Z_1, \ldots, Z_s) \Big) \\ &\geq F \left(s - \frac{s^2}{N} M \right). \end{aligned}$$

Third Lower Bound $R \geq s - \sum_{i=1}^{s} \frac{i}{N-i+1} M$

- Consider only Receivers $1, \ldots, s$, where $s \leq \min\{N, K\}$
- $\mathcal{D}_s$: all demand vectors $\mathbf{d}_s$ of s users having s different demands.

$$|\mathcal{D}_s| = \binom{N}{s} s!$$

- Let $X^{(\mathbf{d}_s)}$ denote the delivery input for demand $\mathbf{d}_s$
- $\forall \mathbf{d}_s \in \mathcal{D}_s$ it holds that:

$$F \cdot R \geq sF - \sum_{i=1}^{s} I(W_{d_i}; Z_1, \ldots, Z_i | W_{d_1}, \ldots, W_{d_{i-1}})$$

- By averaging over all demands $\mathbf{d}_s \in \mathcal{D}_s$

$$\begin{aligned} FR &\geq sF - \sum_{i=1}^{s} \frac{1}{\binom{N}{s} s!} \sum_{\mathbf{d}_s \in \mathcal{D}_s} I(W_{d_i}; Z_1, \ldots, Z_i | W_{d_1}, \ldots, W_{d_{i-1}}) \\ &\geq F\left(s - \sum_{i=1}^{s} \frac{i}{N-i+1} M \right). \end{aligned}$$

Gap between Upper and Lower Bounds

- Multiplicative Gap:

$$\frac{\text{Best Upper Bound}}{\text{Best Lower Bound}} \leq 2.315.$$

- Upper bound from a more constrained scenario

$$R_{\text{Dec}} := \frac{N-M}{M}\left(1-\left(1-\frac{M}{N}\right)^{\min\{K,N\}}\right)$$

- Third lower bound is piecewise linear over intervals $[M_{\ell+1}, M_\ell)$ with

$$M_\ell = \begin{cases} \frac{N-\ell}{\ell+1} & \text{if } \ell \in \{0,1,\ldots,\min\{K,N\}-1\}, \\ 0 & \text{if } \ell = \min\{N,K\}. \end{cases}$$

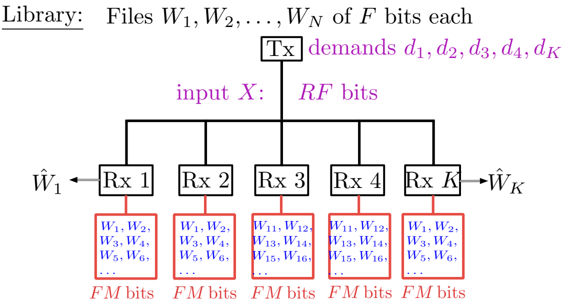
- Make R_{Dec} piecewise linear over same intervals.
- Ratio is bilinear and it suffices to consider end-points of intervals.
  $\rightarrow$ check all end-points!

An Information-Theoretic View of Cache-Aided Networks: Part 2 – Decentralized Coded Caching

Michèle Wigger

Telecom ParisTech, 9 February 2021

Decentralized Placement

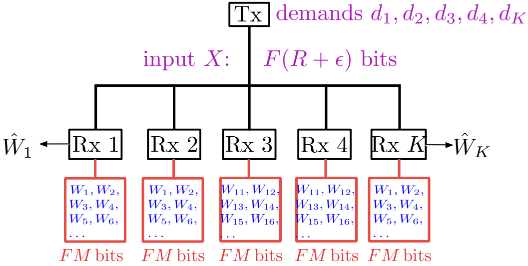


Communication in two phases:

- Placement phase: Each receiver randomly downloads bits from library without knowing demands or the number of receivers K!
  $\rightarrow Z_k = g(W_1, \ldots, W_N, \Theta_k)$
- Delivery phase:
  - Receiver k wants file W_{d_k} $\rightarrow$ sends demand d_k to transmitter
  - Tx describes $W_{d_1}, \ldots, W_{d_K}$ to Rxs $1, \ldots, K$ through input X

Formal Problem Statement

Library: Files $W_1, W_2, \ldots, W_N$ of F bits each



- Cache placement $\quad Z_k = g(W_1, \ldots, W_N, \Theta_k)$,

  where Θ_k is a randomness known to everyone

- Delivery encoding $\quad X = f(W_1, \ldots, W_N, d_1, \ldots, d_K, \Theta_1, \ldots, \Theta_K)$

- Delivery decoding $\quad \hat{W}_k = \varphi_k(X, Z_k, d_1, \ldots, d_K, \Theta_1, \ldots, \Theta_K)$.

- Goal: $\quad \hat{W}_k = W_{d_k}$ for all $k = 1, \ldots, K$ $\quad$ with high probability

Fundamental Rate-Memory Tradeoff $R^{\star}_{\text{Dec}}(M)$

$$R^{\star}_{\text{Dec}}(M) \quad := \quad \min\Big\{R\colon \text{ such that for } (R,M) \text{ each receiver} k \in \{1,\dots,K\} \\ \text{learns } W_{d_k} \text{ with high probability } \Big\}$$

Obvious bounds:

- Local caching gain achievable $R^{\star}_{\text{Dec}}(M) \leq min\{K,N\}\left(1-\frac{M}{N}\right)$
- Cannot improve on centralized setup: $R^{\star}_{\text{Dec}}(M) \geq R^{\star}(M)$.

Decentralized Coded Caching Algorithm

- Placement: Each Receiver sequentially samples and stores each bit of the library with probability $p = \frac{M}{N}$

- For each subset $\mathcal{S} \subseteq \{1, \ldots, K\}$, define now $W_d^{\mathcal{S}}$ the set of all bits of message W_d exclusively cached at all receivers of set $\mathcal{S}$.

- *Delivery Encoding:*
  - Send all demanded bits that are not cached anywhere
  - For $t = 1, \ldots, K-1$ use the coded caching delivery scheme of parameter t to send the demanded bits

    $$\left\{ W_{d_1}^{\mathcal{S}}, \ldots, W_{d_K}^{\mathcal{S}} : \qquad |\mathcal{S}| = t \right\}$$

    Zero-padding might be required for this operation!
  - Delivery decoding similar to coded caching scheme, but again for all parameters $t = 0, 1, 2, \ldots, K-1$.

---

[2] M. A. Maddah-Ali, U. Niesen, "Decentralized coded caching attains order-optimal memory-rate tradeoff," *IEEE Trans. on Inf. Theory*.

Analysis of Decentralized Scheme

- By the weak law of large numbers, for all $\epsilon > 0$:

$$\Pr\left[\left|\left|W_d^{\mathcal{S}}\right| - p^{|\mathcal{S}|}(1-p)^{K-|\mathcal{S}|}F\right| > \epsilon\right] \to 0 \quad \text{as} \quad F \to \infty.$$

- Expected storage

$$M = N\sum_{t=1}^{K}\binom{K-1}{t-1}p^t(1-p)^{K-t}$$
$$= Np\sum_{t'=0}^{K-1}\binom{K-1}{t'}p^{t'}(1-p)^{K-1-t'} = Np = M.$$

- Expected rate:

$$R = \sum_{t=0}^{K-1}\binom{K}{t+1}p^t(1-p)^{K-t} = \frac{1-p}{p}\sum_{t'=1}^{K}\binom{K}{t'}p^{t'}(1-p)^{K-t'}$$
$$= \frac{1-p}{p}\sum_{t'=0}^{K}\binom{K}{t'}p^{t'}(1-p)^{K-t'} - \frac{1-p}{p}(1-p)^K$$
$$= \frac{1-p}{p}\left(1-(1-p)^K\right) = \frac{N-M}{M}\left(1-\left(1-\frac{M}{N}\right)^K\right)$$

Results for Decentralized Caching

Upper Bound $R^\star(M)$ for Decentralized Caching

$$R^\star(M) \leq K\left(1-\frac{M}{N}\right)\min\left\{\frac{N}{KM}\left(1-\left(1-\frac{M}{N}\right)^K\right), \frac{N}{K}\right\}$$

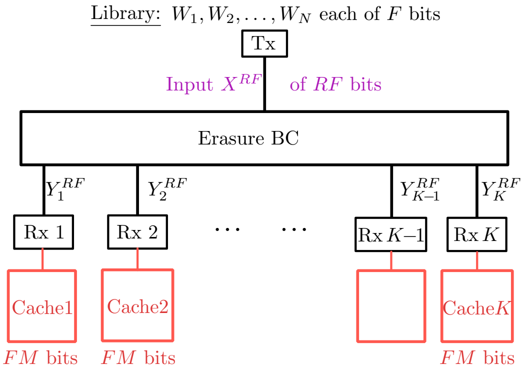
- Above upper bound matches $R^\star(M)$ up to a factor of at least 12. (Proved analytically in [2].)
- Above upper bound matches coded caching upper bound for centralized scenario up to a factor of 1.6. (Shown numerically.)

An Information-Theoretic View of Cache-Aided Networks:
Part 3 – Caching for Erasure Broadcast Channels

Michèle Wigger

Telecom ParisTech, 17 November 2020

Delivery over Homogeneous Erasure Broadcast Channels (BC)



- Each transmitted bit erased at each receiver with probability $\delta > 0$, irrespective of all other bits:

$$\Pr[Y_{k,t} = X_t] = 1 - \delta \qquad \text{and} \qquad \Pr[Y_{k,t} = \Delta] = \delta$$

Noisy Setup Requires Vanishing Error Probability

- Cache placement $Z_k = g_k(W_1, \ldots, W_N)$ consists of FM bits
- Delivery encoding $X^{RF} := (X_1, \ldots, X_{RF}) = f(W_1, \ldots, W_N, d_1, \ldots, d_K)$
- Delivery decoding:
  - Receiver k receives $Y_k^{RF} = (Y_{k,1}, \ldots, Y_{k,RF})$
  - It produces $\hat{W}_k = \varphi_k(Y_k^{RF}, Z_k, d_1, \ldots, d_K)$.
- Goal: $\Pr\left[\hat{W}_k = W_{d_k}\right] \to 0$ as $F \to \infty$ for all $k = 1, \ldots, K$

$\to$ Need to tolerate errors because the channel has nonzero probability of experiencing lots of erasures.

Rate-Memory Tradeoff for Erasure Broadcast Channels ($N \geq K$)

- Capacity of a erasure broadcast channel (EBC) is $(1-\delta)$, both for sending common messages and private messages.
- That means, by sending FR inputs, we can convey $FR(1-\delta)$ information bits with arbitrarily small probability of error as $F \to \infty$.
- So now each XOR packet requires $(1-\delta)^{-1}$ times more slots for reliable transmission than before. As a consequence:

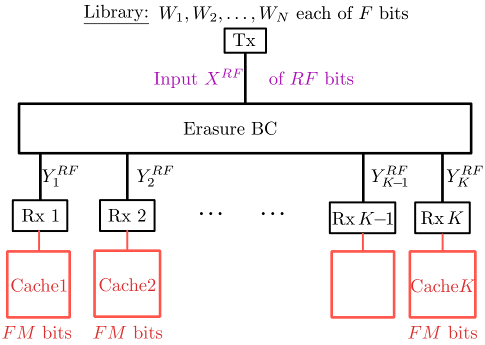
$$R^\star(M) \leq \text{convhull}\Big\{(M_t, R_t):\ t = 0, 1, \ldots, K\Big\}$$

where

$$M_t = \frac{tN}{K}$$

$$R_t = \frac{K}{1-\delta}\Big(1 - \frac{M_t}{N}\Big)\left(\frac{KM_t}{N}\right)^{-1}.$$

Delivery over Heterogeneous Erasure BCs



- Erasure probability at receivers $1, \ldots, K_w$ is δ_1, where $K_w < K$
- Erasure probability at receivers $K_w + 1, \ldots, K$ is $\delta_2 < \delta_1$
- Adapt the coded caching upper bound on $R^\star(M)$ to this setup!

Rate-Memory Tradeoff for Asymmetric EBCs

- The XOR packets that are meant for at least one weak receiver require $(1-\delta_1)^{-1}$ times more channel inputs than over the noisefree channel
- The XOR packets that are meant for only strong receivers require $(1-\delta_2)^{-1}$ times more channel inputs than over the noisefree channel
- $\binom{K_s}{t+1}$ XOR packets intended for only the strong receivers, where t denotes the parameter of the coded caching scheme
- $\binom{K}{t+1} - \binom{K_s}{t+1}$ of the XOR packets intended for at least one weak receiver

EBCs with Unequal Channel Strengths

- In this coded caching algorithm requires:

$$R^\star(M) \leq \text{convhull}\big\{(M_t, R_t):\ t = 0, 1, \ldots, N\big\}$$

where

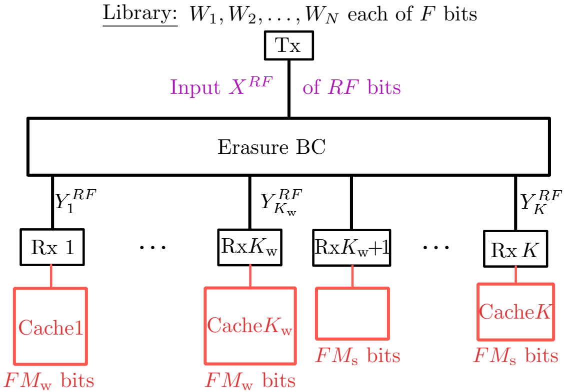
$$M_t = \frac{tN}{K}$$

$$R_t = \frac{K}{(1-\delta_2)}\left(1 - \frac{M_t}{N}\right)\left(1 + \frac{KM_t}{N}\right)^{-1} + \underbrace{\frac{\binom{K}{t+1} - \binom{K-K_w}{t+1}}{\binom{K}{t}}\left(\frac{1}{1-\delta_1} - \frac{1}{1-\delta_2}\right)}_{\text{penalty caused by weak receivers}}$$

Penalty Caused By Weak Receivers

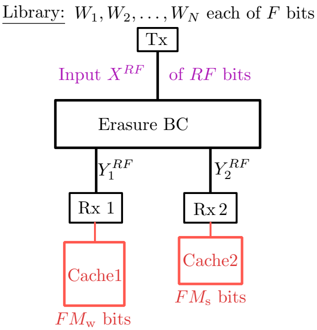
- Weak receivers cause rate-penalty!
- As we will see, this holds only for asymmetric situations
- Penalty caused by weak receivers can partially be removed in asymmetric setups where weak receivers need less information or when they have larger cache memories
- Efficient elimination of rate penalty requires new coding approach!

Can Cache Assignment Resolve Penalty?



- Move part of the cache memories from strong receivers to weak receivers
  $\rightarrow$ Idea: Help more the weak receivers to make the network more balanced
- How to exploit additional cache memories?
  $\rightarrow$ Coded caching only works with equal cache memories at all receivers

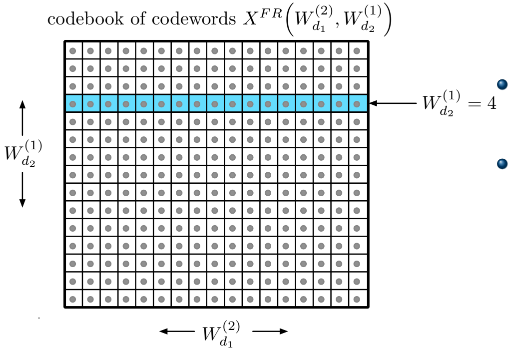
Two-User Example with Asymmetric Cache Memories



- Assign cache memory inverse proportionally to channel capacities: $M_{\text{w}} = \frac{1-\delta_2}{2-\delta_1-\delta_2} N$ and $M_{\text{s}} = \frac{1-\delta_1}{2-\delta_1-\delta_2} N$
- Split each $W_d = (W_d^{(1)}, W_d^{(2)})$ with sizes $\frac{1-\delta_2}{2-\delta_1-\delta_2} F$ and $\frac{1-\delta_1}{2-\delta_1-\delta_2} F$ bits
- Placement: store $\left\{W_d^{(1)}\right\}_{d=1}^{N}$ in Cache 1 and $\left\{W_d^{(2)}\right\}_{d=1}^{N}$ in Cache 2

Use a "Piggyback-Code" to Delivery $W_{d_1}^{(2)}$ and $W_{d_2}^{(1)}$

Randomly generate all codewords IID by choosing all entries IID.



- Rx 1 knows $W_{d_2}^{(1)}$ $\rightarrow$ restricts decoding to a row
- Rx 2 knows $W_{d_1}^{(2)}$ $\rightarrow$ restricts decoding to a column

- For Rx 1 to be able to decode, X^{FR} needs to be of size $\frac{F}{2-\delta_1-\delta_2}$ bits
- For Rx 2 to be able to decode, X^{FR} needs to be of size $\frac{F}{2-\delta_1-\delta_2}$ bits

"Piggyback-Code" removes Penalty caused by Weak Receiver

For $p(\mathit{error}) \to 0$ as $n \to \infty$ we need $R \geq \dfrac{1}{2 - \delta_1 - \delta_2}$

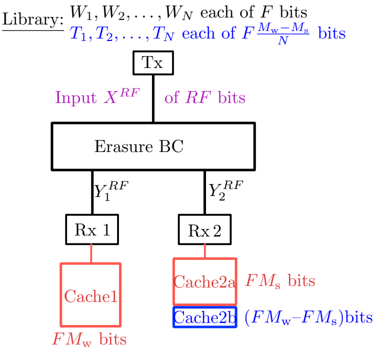
- Same performance as if only one of the receivers was present!
- Weaker receiver does not penalize stronger receiver!

Piggyback-Coding Extends to K Receivers

- Piggyback coding extends to arbitrary number of receivers and different erasure probabilities
- Extends to general degraded broadcast channels (BC)
- Modify coded caching as follows:
  - Size of a subpart of files depends on channel strengths of the receivers caching this subpart
  - Choose $t+1$-dimensional piggyback codebook for delivery communication to each subset of $t+1$ receivers

Size of subparts (and thus of cache memories) is chosen so that each piggyback codebook is decoded using the same time by each of the involved receivers

Higher Resolution At Stronger Receivers



- Let $T_1, \ldots, T_N$ be higher resolution info. required at Rx 2, not at Rx 1
- Store all $T_1, \ldots, T_N$ in Rx 2's cache memory
- Apply placement and delivery strategies described before