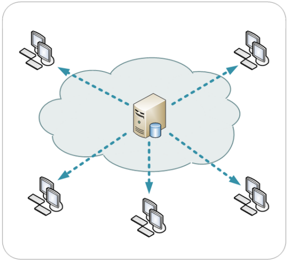
An Information-Theoretic View of Cache-Aided Communication, Compression, and Computation Systems

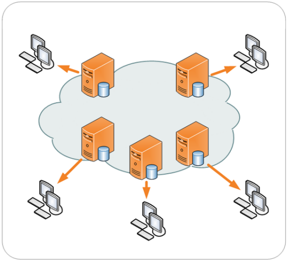
Michèle Wigger

GlobalSIP 2015, Orlando, Florida

14 December 2015

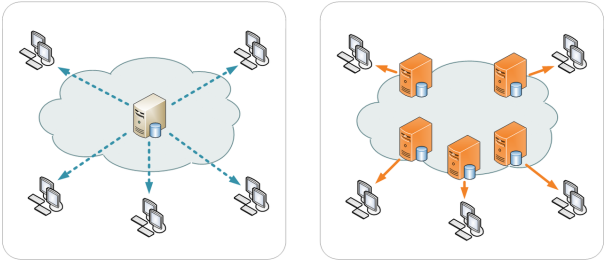
Content Delivery Networks





- 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

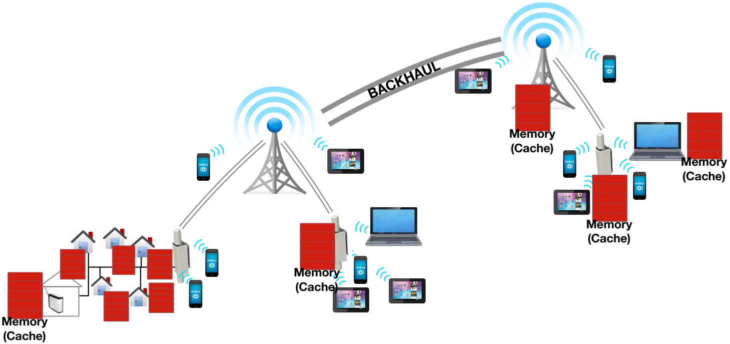
Content Delivery Networks



- 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

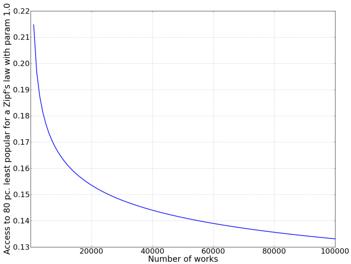
Let's see how caches enter the wireless story....

Promising Solution: Distribute Caches at Various Locations in Network



- 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!

Questions to Address for Cache-Aided Networks

- Sizes of caches?
- What to store in the caches?
- How to communicate in presence of cached data?
- Benefits in rate, delay, energy?

Energy Balance of Cache-Aided Communication

- Energy needed to cache/store data

- Energy needed to transmit caching information

- Energy needed to transmit delivery information

Energy Balance of Cache-Aided Communication

- Energy needed to cache/store data

  $\rightarrow$ can be relatively small & unused storage entity sometimes already available

- Energy needed to transmit caching information

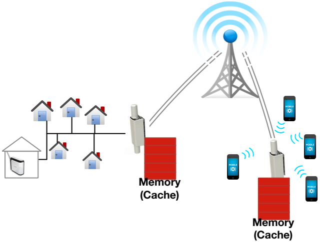
  $\rightarrow$ minimized by using simple low-rate codes and modulation schemes

- Energy needed to transmit delivery information

  $\rightarrow$ improved through local retrieval of information

  $\rightarrow$ global caching gain brings further reductions in rate and energy

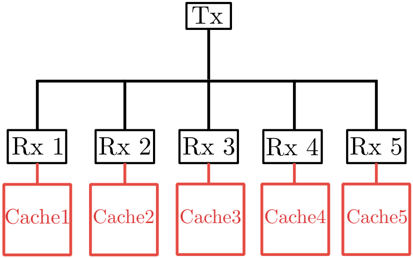
Our Scenario: One-To-Many Communication with Receiver-Caches



- All files equally popular → interested in worst-case performance
- Centralized protocol on how to fill caches
- Caches filled during nights when demands not yet known

Maddhah-Ali & Niesen Source Coding Setup

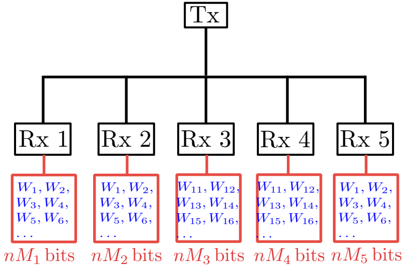
Library: Files $W_1, W_2, \ldots, W_D$ of $n\rho$ bits each (no popularities)



Communication in two phases:

Maddhah-Ali & Niesen Source Coding Setup

Library: Files $W_1, W_2, \ldots, W_D$ of $n\rho$ bits each

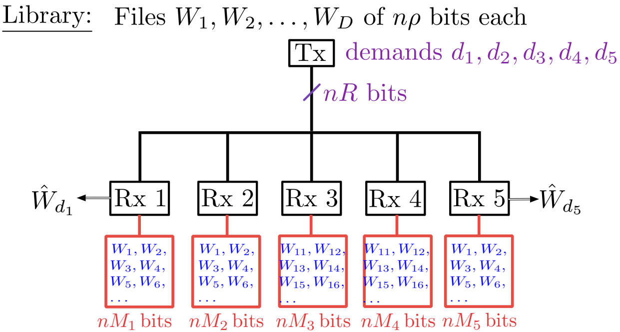


cache contents: arbitrary functions of messages $W_1, \ldots, W_D$

Communication in two phases:

- Placement phase: Tx fills caches without knowing demands $d_1, \ldots, d_5$

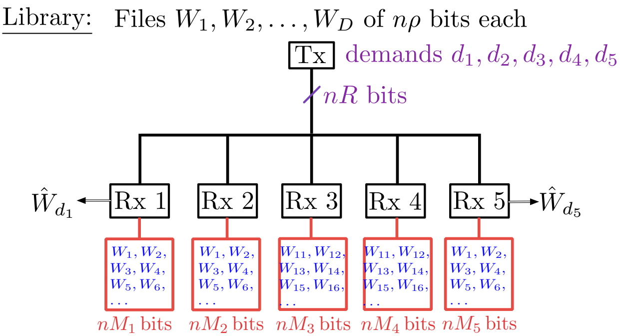
Maddhah-Ali & Niesen Source Coding Setup



Communication in two phases:

- Placement phase: Tx fills caches without knowing demands $d_1, \ldots, d_5$
- Delivery phase: Tx describes $W_{d_1}, \ldots, W_{d_5}$ to Rxs $1, \ldots, 5$, respectively, through nR common bits

Maddhah-Ali & Niesen Source Coding Setup

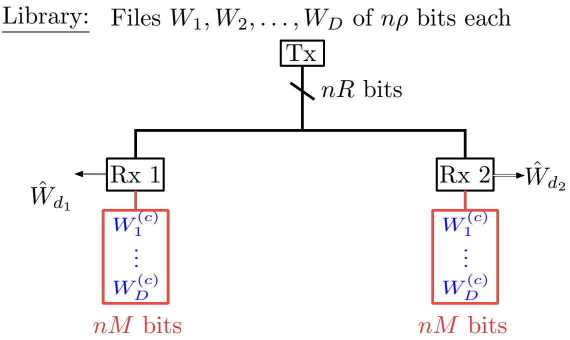


Communication in two phases:

Rates-Memories Tradeoff

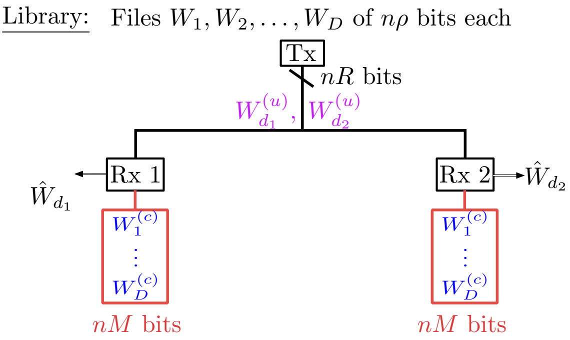
For which $(\rho, R, M_1, \ldots, M_K)$ is error-free data transmission possible?

Naive Uncoded Caching for $K = 2$ Receivers



- Split $W_d = (W_d^{(c)}, W_d^{(u)})$ of length $\left(\frac{\mathsf{M}}{D} n, (\rho - \frac{\mathsf{M}}{D}) n\right)$ bits

Naive Uncoded Caching for $K = 2$ Receivers

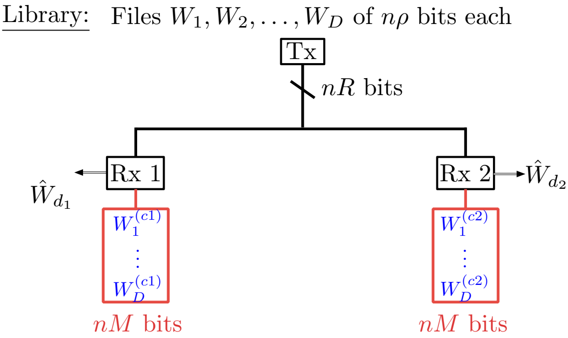


- Split $W_d = (W_d^{(c)}, W_d^{(u)})$ of length $\left(\frac{\mathsf{M}}{D} n, (\rho - \frac{\mathsf{M}}{D}) n\right)$ bits

Rates-Memory Trade-Off

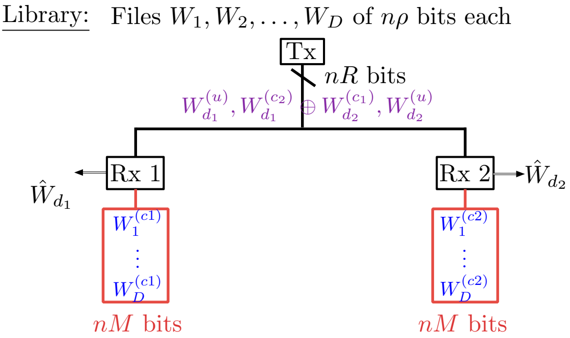
Reconstruction is possible, if $R \geq 2\left(\rho - \frac{\mathsf{M}}{D}\right)$

Coded caching for $K = 2$ Receivers [Maddah-Ali&Niesen 2013]



- Split $W_d = (W_d^{(c1)}, W_d^{(c2)}, W_d^{(u)})$ of length $\left(\frac{\mathsf{M}}{D}n, \frac{\mathsf{M}}{D}n, (\rho - 2\frac{\mathsf{M}}{D})n\right)$ bits

Coded caching for $K = 2$ Receivers [Maddah-Ali&Niesen 2013]

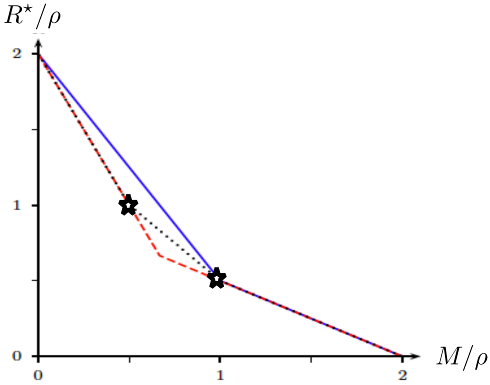


- Split $W_d = (W_d^{(c1)}, W_d^{(c2)}, W_d^{(u)})$ of length $\left(\frac{M}{D}n, \frac{M}{D}n, (\rho - 2\frac{M}{D})n\right)$ bits

Rates-Memory Trade-Off

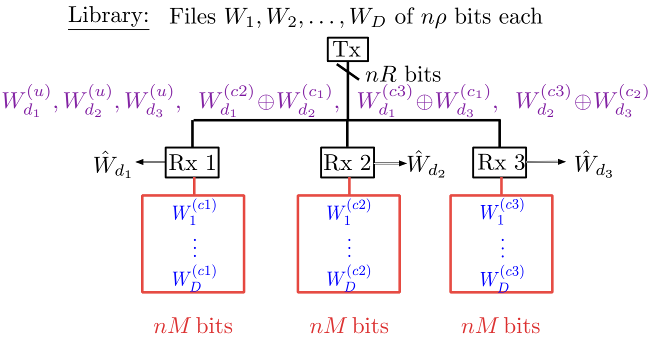
Reconstruction possible, if $R \geq 2\left(\rho - \frac{M}{D}\right) - \frac{M}{D}$

Optimal Rates-Memory Tradeoff $R^\star(\rho, M)$ for $K = D = 2$



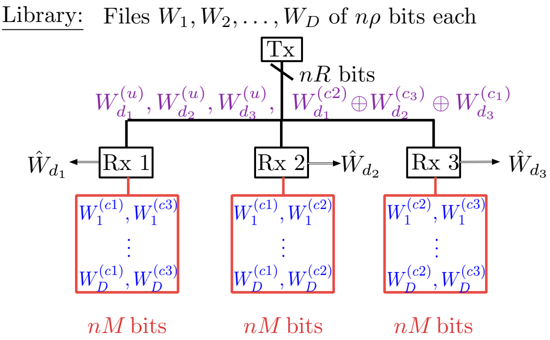
- Coded caching gives right-star
- Symmetry arguments for left-star $\rightarrow$ exchange caching and delivery phase

Coded caching for $K = 3$ Receivers [Maddah-Ali&Niesen 2013]



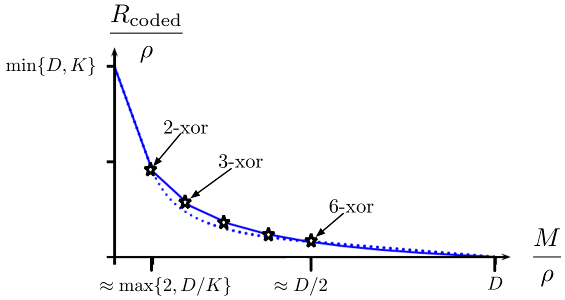
- Split $W_d = (W_d^{(c1)}, W_d^{(c2)}, W_d^{(c3)}, W_d^{(u)})$
- If M small: save single part at each receiver
- If M large: save two parts at each receiver

Coded caching for $K = 3$ Receivers [Maddah-Ali&Niesen 2013]



- Split $W_d = (W_d^{(c1)}, W_d^{(c2)}, W_d^{(c3)}, W_d^{(u)})$
- If M small: save single part at each receiver
- If M large: save two parts at each receiver

Local and Global Caching Gains $K \geq 2$ [Maddah-Ali&Niesen 2013]



Coded caching achieves

Reconstruction possible, if $R_{\text{coded}} \geq K(\rho - \frac{M}{D}) \cdot \min\left\{\frac{1}{1+KM/\rho/D}, \frac{D}{K}\right\}$

$$1 \leq \frac{R^\star(\rho, M)}{R_{\text{coded}}(\rho, M)} \leq 12, \qquad \forall K, \rho, D, M.$$

Extensions

- Decentralized caching

  [M. A. Maddah-Ali, U. Niesen, "Decentralized coded caching attains order-optimal memory-rate tradeoff"]

- Nonuniform or random demands

  [U. Niesen and M. A. Maddah-Ali, "Coded caching with nonuniform demands"]
  [Ji, Tulino, Llorca, and Caire, "Order-optimal rate of caching and coded multicasting with random demands"]

- Online caching phase

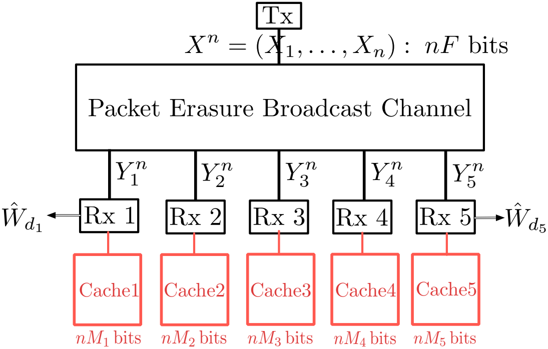
  [R. Pedarsani, M. A. Maddah-Ali and U. Niesen, "Online coded caching"]

- Hierarchical caching

  [Hachem, Karamchandani, Diggavi, "Coded caching for heterogeneous wireless networks with multi-level access"]

Delivery over Noisy Broadcast Channel (BC) [Saeedi, Timo, Wigger 2015]

Library: Files $W_1, W_2, \ldots, W_D$ of $n\rho$ bits each (no popularities)

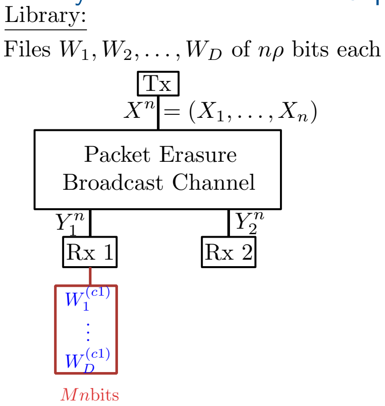


- Receiver k gets erasure with probability δ_k where $\delta_1 \geq \delta_2 \geq \ldots \geq \delta_K$

$$Y_k^n = (X_1, X_2, \Delta, X_4, \Delta, \ldots, X_{n-1}, \Delta)$$

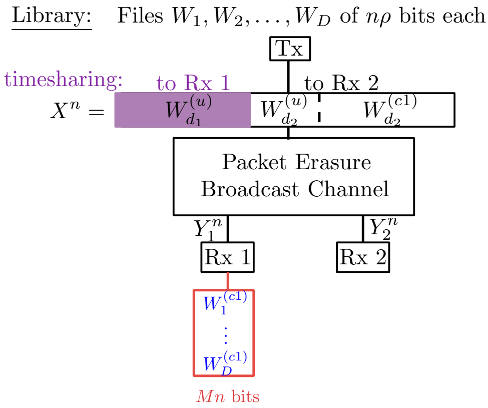
$\rightarrow$ fraction of Δs $\approx \delta_k$

Example: Asymmetric Caches and Separate Channel Coding



- $W_d = (W_d^{(c1)},\ W_d^{(u)})$ of sub-rates $(\frac{\mathsf{M}}{D},\ \rho - \frac{\mathsf{M}}{D})$

Example: Asymmetric Caches and Separate Channel Coding



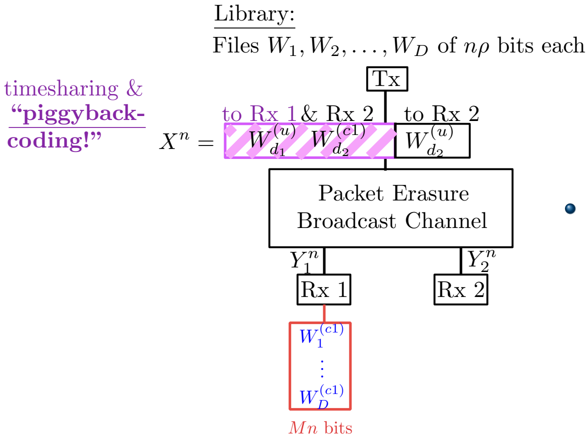
- $W_d = (W_d^{(c1)},\ W_d^{(u)})$ of sub-rates $(\frac{\mathsf{M}}{D},\ \rho - \frac{\mathsf{M}}{D})$

Separate Cache-Channel Coding $\to$ No Global Caching Gain

$p(\text{error}) \to 0$ if: $$\frac{\rho - \frac{\mathsf{M}}{D}}{F(1-\delta_1)} + \frac{\rho}{F(1-\delta_2)} \leq 1$$

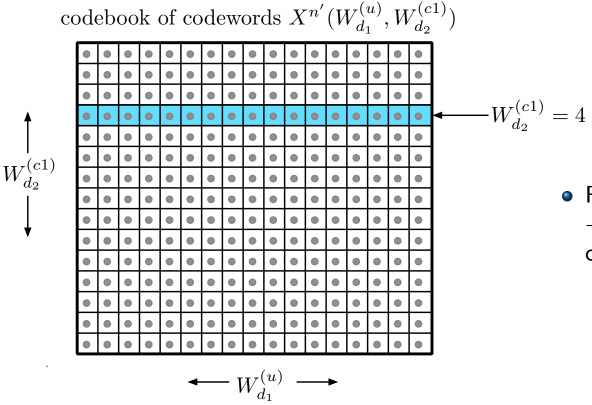
Standard Erasure BC: $p(error)$ if : $$\frac{\rho_1}{F(1-\delta_1)} + \frac{\rho_2}{F(1-\delta_2)} \leq 1$$

Example: Our Joint Cache-Channel Scheme for $K = 2$



- $W_d = (W_d^{(c1)},\ W_d^{(u)})$ of sub-rates $(\frac{M}{D},\ \rho - \frac{M}{D})$

Piggyback Coding to Send $(W_{d_1}^{(u)}, W_{d_2}^{(c1)})$ to Both Rxs

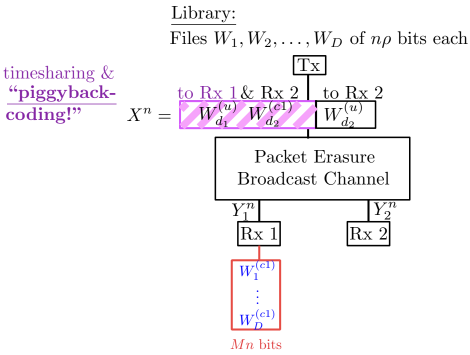


- Rx 1 knows $W_{d_2}^{(c1)}$ $\rightarrow$ restrict decoding to corresponding row

Transmission of $W_{d_2}^{(c1)}$ not affecting Rx 1 at all!

$$p(error) \rightarrow 0 \text{ as } n \rightarrow \infty: \qquad \max\left\{ \frac{\rho - \frac{M}{D}}{F(1-\delta_1)}, \frac{\rho}{F(1-\delta_2)} \right\} \leq \frac{n'}{n}$$

Performance of Joint Cache-Channel Scheme with Piggyback Coding



- $W_d = (W_d^{(c1)}, \ W_d^{(u)})$ of sub-rates $(\frac{\mathsf{M}}{D}, \ \rho - \frac{\mathsf{M}}{D})$

Joint Cache-Channel Coding → Global Caching Gain!

$$p(error) \to 0 \quad \text{if :} \qquad \underbrace{\max\left\{\frac{\rho - \frac{\mathsf{M}}{D}}{F(1-\delta_1)}, \ \frac{\rho}{F(1-\delta_2)}\right\}}_{\text{piggyback coding}} + \frac{\rho - \frac{\mathsf{M}}{D}}{F(1-\delta_2)} \leq 1$$

Example for $\delta_1 = 4/5$ and $\delta_2 = 1/5$ and M$\leq \rho 3D/8$

2. Asymmetric caches $\mathsf{M}_1 = \mathsf{M}$ and $\mathsf{M}_2 = 0$ & separate source-channel coding

$$\rho \leq \frac{4}{5} F(1-\delta_1) + \frac{4}{5}\frac{\mathsf{M}}{D}$$

3. Asymmetric caches $\mathsf{M}_1 = \mathsf{M}$ and $\mathsf{M}_2 = 0$ & joint cache-channel coding

$$\rho \leq \frac{4}{5} F(1-\delta_1) + 1\frac{\mathsf{M}}{D}$$

Example for $\delta_1 = 4/5$ and $\delta_2 = 1/5$ and M$\leq \rho 3D/8$

1. Symmetric caches $\mathsf{M}_1 = \mathsf{M}_2 = \mathsf{M}/2$ & coded caching as before & separate source-channel coding

$$\rho \leq \frac{4}{5} F(1-\delta_1) + \frac{3}{5}\frac{\mathsf{M}}{D}$$

2. Asymmetric caches $\mathsf{M}_1 = \mathsf{M}$ and $\mathsf{M}_2 = 0$ & separate source-channel coding

$$\rho \leq \frac{4}{5} F(1-\delta_1) + \frac{4}{5}\frac{\mathsf{M}}{D}$$

3. Asymmetric caches $\mathsf{M}_1 = \mathsf{M}$ and $\mathsf{M}_2 = 0$ & joint cache-channel coding

$$\rho \leq \frac{4}{5} F(1-\delta_1) + 1\frac{\mathsf{M}}{D}$$

Fundamental Limits of Caching

A caching/delivery scheme cannot have $p(\text{error}) \to 0$ as $n \to \infty$, if

$$\frac{\rho - M_1}{F(1-\delta_1)} + \frac{\rho - M_2}{1-\delta_2} \leq 1$$

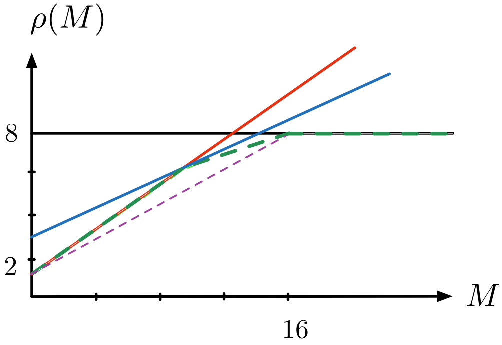
$$2\rho \leq 2F(1-\delta_1) + M_1$$

$$2\rho \leq 2F(1-\delta_2) + M_2$$

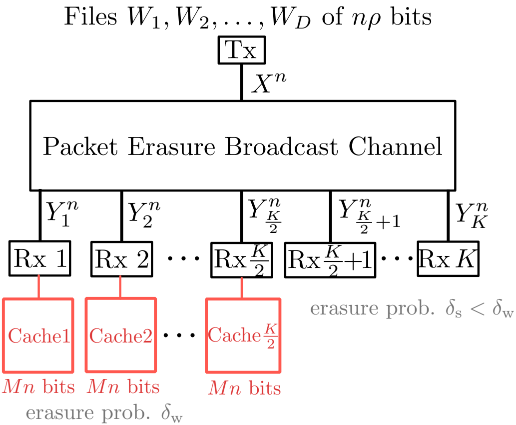
$$3\rho \leq F(1-\delta_1) + F(1-\delta_2) + M_1 + M_2$$

Achievable and Infeasible Maximum Rates $\rho(M)$

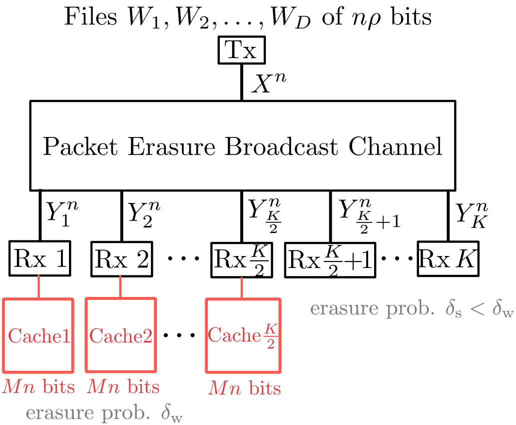
- $K = D = 2$ and asymmetric cache sizes $M_1 = M$ and $M_2 = 0$
- $\delta_1 = 0.8$ and $\delta_2 = 0.2$ and $F = 10$
- Maximum rates $\rho(M)$ in bits per channel use



Extension to K Receivers



Extension to K Receivers



Example:

- Small cache size M
- $\delta_{\text{w}} = \frac{4}{5}$ and $\delta_{\text{s}} = \frac{1}{5}$

- Separate coding:
  $$\rho \le \frac{\frac{4}{5}F}{5K} + \frac{KM}{D}\left(\frac{2}{5} + \frac{2}{5}\frac{1}{K}\right)$$
- With piggybacking:
  $$\rho \le \frac{\frac{4}{5}F}{5K} + \frac{KM}{D}\left(\frac{3}{5} + \frac{2}{5}\frac{1}{K}\right)$$

- Split $W_d = \left(W_d^{(c1)}, \ldots, W_d^{(cK/2)}, W_d^{(u)}\right) \to$ cache $W_d^{(ck)}$ at Rx k
- Deliver Maddah-Ali&Niesen x-ors and piggyback $\{W_{d_\ell}^{(ck)}\}_{\ell=K/2+1}^{K}$ on $W_{d_k}^{(u)}$

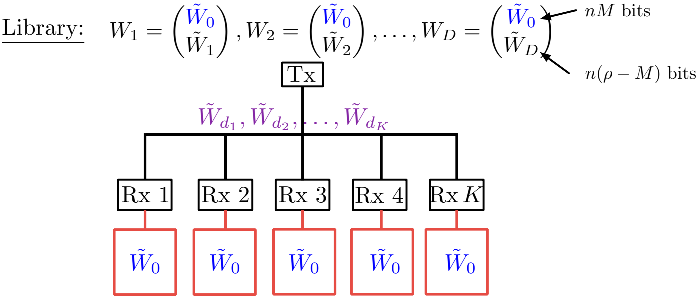
Insights and Intuition

- Important to consider noisy communication channel:
  - Joint cache-channel coding (piggyback coding)
  - Caching gains combine with feedback gains

    [A. Ghorbel, M. Kobayashi, S. Yang, "Cache-enabled broadcast packet erasure channels with state feedback"]
  - Interplay between caching gains and CSI gains

    [J. Zhang and P. Elia, "Fundamental limits of cache-aided wireless BC: interplay of coded-caching and CSIT feedback"]
- Larger caches for weak receivers $\rightarrow$ even more important with joint cache-channel coding
- Piggyback coding useful whenever info for strong Rx in cache of weak Rx!

Situation that Motivates our Next Problem [Timo, Saeedi, Wigger, Geiger 2015]



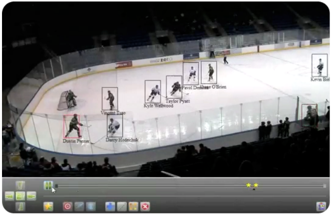
- Ignoring file correlation with M small: $\to R \geq K(\rho - \frac{M}{D}) - \frac{M}{D}$
- Storing common information $\tilde{W}_0$ in each cache: $\to R \geq K(\rho - M)$

Main Insights

Cache-contents now useful for multiple demands without need for coding

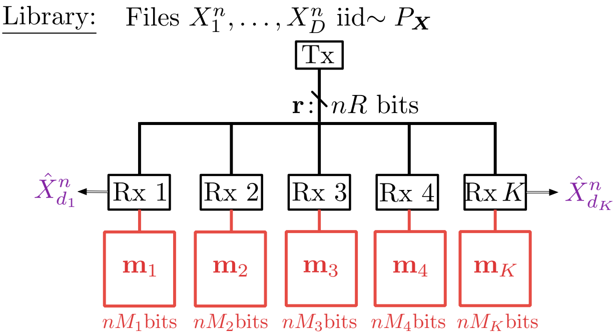
Files $X_1^n, X_2^n, \ldots, X_D^n$ Might be Correlated!

- Interactive videos: users can choose different angles, segments, etc



- Large data bases: users retrieve different functions of measured samples
  - Different features of biological data stored in a data base
- Cloud computing: different users download processed versions of data
  - Profiles of people in social networks

General One-To-Many Scenario



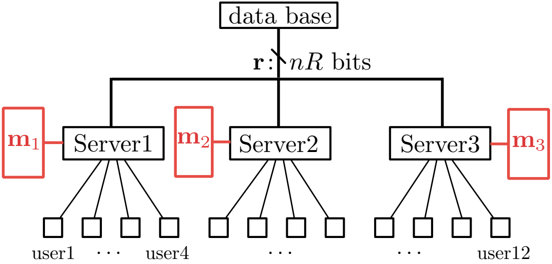
Lossless reconstruction of demanded files

$$\forall d_1, \ldots, d_K: \quad \Pr\left(\hat{X}_k^n(\mathbf{m}_k, \mathbf{r}, d_1, \ldots, d_K) \neq X_{d_k}^n\right) \to 0 \text{ as } n \to \infty$$

Application: Computation of Different Functions from Common Data

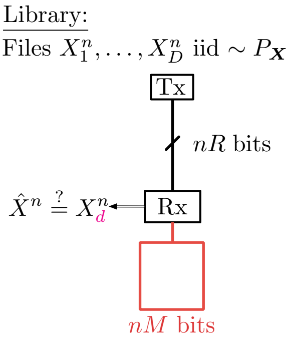
Data: $X^n = (X_1, \ldots, X_n)$

Files $X_1^n, \ldots, X_D^n$ where $X_{d,t} = f_d(X_t)$



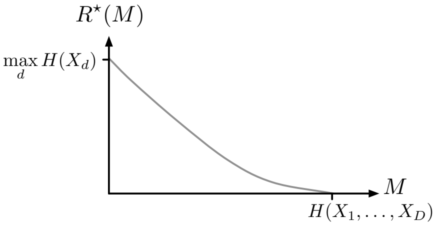
- Data stored on a central data base
- Each user wishes to retrieve one function $X_d^n = f_d(X^n)$
- Demand not known when caching at servers

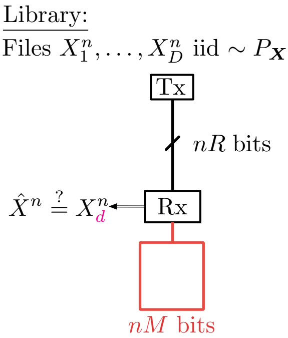
Single Receiver Problem and a Typical Rate-Memory Tradeoff $R^\star(M)$



Single Receiver Problem and a Typical Rate-Memory Tradeoff $R^\star(M)$

- A typical rate-memory tradeoff

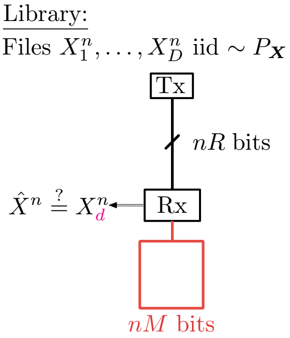




Single-Receiver: Optimal Rate-Memory Tradeoff

Scheme for Optimal Rate-Memory Tradeoff

- Compress $(X_1^n, \ldots, X_D^n)$ by U^n, and cache compression index
- Deliver X_d^n under side-info U^n



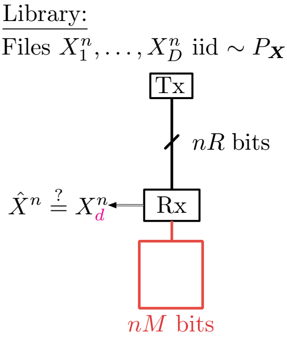
- $R^\star(M) = \min_{P_{U|X_1,\ldots,X_D}} \max_{d \in \{1,\ldots,D\}} H(X_d|U)$

  s.t. $I(X_1, \ldots, X_D; U) \leq M$

Single-Receiver: Optimal Rate-Memory Tradeoff

Scheme for Optimal Rate-Memory Tradeoff

- Compress $(X_1^n, \ldots, X_D^n)$ by U^n, and cache compression index
- Deliver X_d^n under side-info U^n



- $R^\star(M) = \min\limits_{P_{U|X_1,\ldots,X_D}} \max\limits_{d \in \{1,\ldots,D\}} H(X_d|U)$

  s.t. $I(X_1, \ldots, X_D; U) \le M$

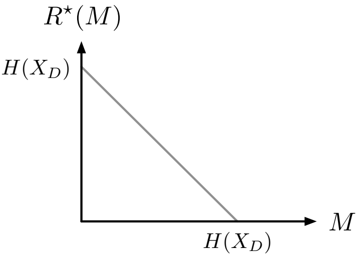
- How to choose $P_{U|X_1,\ldots,X_D}$?

Example 1: Degenerate Sources

- $X_1 \subseteq X_2 \subseteq \cdots X_D$

Rate-Memory Tradeoff

$$R^\star(M) = \max\{0,\ H(X_D) - M\}$$



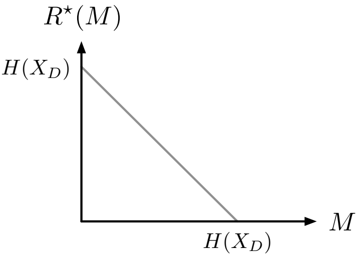
- U: store largest X_d that fits into cache

Example 1: Degenerate Sources

- $X_1 \subseteq X_2 \subseteq \cdots X_D$

Rate-Memory Tradeoff

$$R^\star(M) = \max\{0,\ H(X_D) - M\}$$



- U: store largest X_d that fits into cache

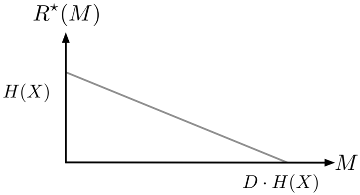
→ same rate-memory tradeoff as if genie revealed demand d before caching

Example 2: Independent and Identical Sources

- $X_{1,t}, \ldots, X_{D,t}$ i.i.d. $\sim P_X$

Rate-Memory Tradeoff

$$R^\star(M) = \max\left\{0,\ H(X) - \frac{M}{D}\right\}$$



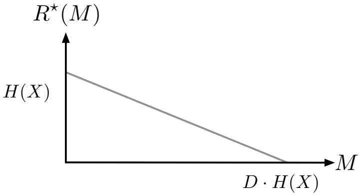
- Compress each source X_d^n independently with $\frac{M}{D}$ bits and cache these compression bits

Example 2: Independent and Identical Sources

- $X_{1,t}, \ldots, X_{D,t}$ i.i.d. $\sim P_X$

Rate-Memory Tradeoff

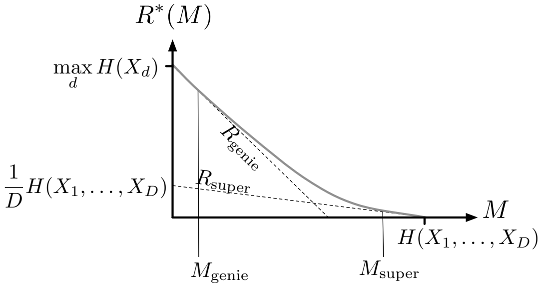
$$R^\star(M) = \max\left\{0,\ H(X) - \frac{M}{D}\right\}$$



- Compress each source X_d^n independently with $\frac{M}{D}$ bits and cache these compression bits

$\rightarrow$ same rate-memory tradeoff as for a super-user with D delivery pipes of rate R that reconstructs all sources $X_1^n, \ldots, X_D^n$

A Closer Look at the Typical Rate-Memory Function



- $M_{\text{genie}} = \max \left\{ H(U): \quad H(U|X_d) = 0, \; \forall d = 1, \ldots, D \right\}$

  Gács-Körner common info in symmetric setups

- $M_{\text{super}} = \min \left\{ I(U; X_1, \ldots, X_d): \quad X_d \to U \to X_{\mathcal{D} \setminus \{d\}}, \; \forall d = 1, \ldots, D \right\}$

  Wyner common info in symmetric setups

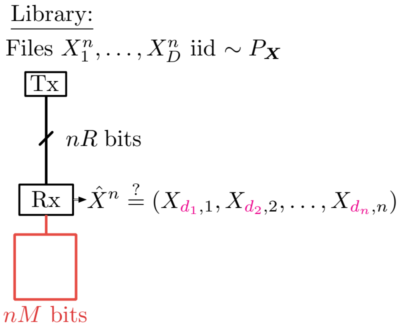
Related Single-Rx Scenario: Per-Symbol Demands [Wang, Lim, Gastpar 2015]

- Sequence of demands $d_1, \ldots, d_n$ i.i.d. $\sim P_{\mathbb{D}}$

Optimal Rate-Memory Tradeoff

$$R^\star(M) = \min H(X_d | U, \mathbb{D}),$$

over $P_{U|X_1,\ldots,X_D}$: $I(X_1, \ldots, X_D; U|\mathbb{D}) \leq M$

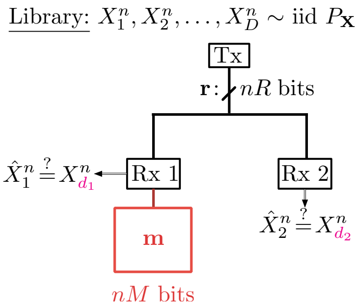


- Equivalent to information bottleneck (IB) method

  $\rightarrow$ efficient algorithms exist

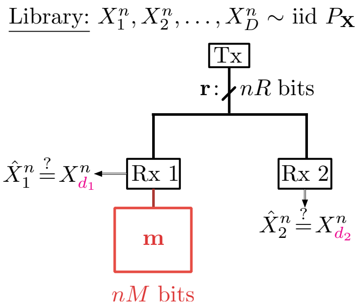
Two-User Lossy-Source Coding with One Cache

- Caching:
  $\mathbf{m} = \text{caching}(X_1^n, \ldots, X_D^n)$
- Delivery:
  $\mathbf{r} = \text{delivery}(X_1^n, \ldots, X_D^n, d_1, d_2)$



Two-User Lossy-Source Coding with One Cache

- Caching:
  $\mathbf{m} = \text{caching}(X_1^n, \ldots, X_D^n)$
- Delivery:
  $\mathbf{r} = \text{delivery}(X_1^n, \ldots, X_D^n, d_1, d_2)$

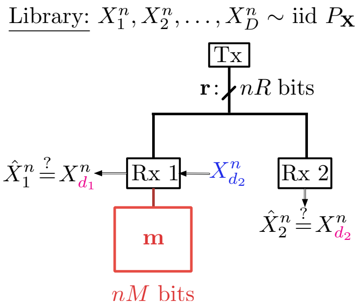


- Rx 1 more powerful than Rx 2 $\rightarrow$ Rx 1 can reconstruct $X_{d_2}^n$

  $\rightarrow$ But only in delivery phase!

Coding Scheme for an Idealized Setup

- Genie-aided scenario: Rx 1 (and Tx) learns $X^n_{d_2}$ even before caching



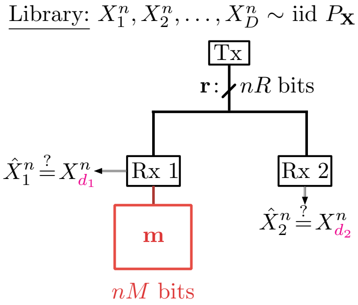
Coding scheme:

- Describe $X^n_{d_2}$ for Rx 2 (delivery)
- For Rx 1, use single-user caching with Rx side-info

Problem: compression with side-info. (Wyner-Ziv and Slepian-Wolf coding) require statistical knowledge of SI

$\rightarrow$ need to adjust bin-size!

Coding Scheme for 2 Users and Rx 1 Caching using Adaptive Binning



Coding scheme:

- Describe $X_{d_2}^n$ for Rx 2 (delivery)
- For Rx 1, use single-user caching with Rx side-info with adaptive binning and rate-transfer (caching and delivery)

Rate of idealized scenario is achievable:

$$R^\star \geq \min_{P_{U|X_1,\ldots,X_D}} \max_{(d_1,d_2)} H(X_{d_2}) + H(X_{d_1} \,|\, U, X_{d_2}) + \tilde{R}$$

s.t. $$I(U; X_1, \ldots, X_D | X_{d_2}) \leq M + \tilde{R}.$$

Insights

- Cache common information (Wyner or Gács-Körner) if possible
- Use previously delivered correlated files as side-information (Wyner-Ziv coding, Slepian-Wolf coding)
- New tools (adaptive binning and rate-transfer) needed to implement Wyner-Ziv or Slepian-Wolf with "unknown" side-information

Summary

- Clever choices of cache contents can highly improve caching gains:
  - Diversify cache contents if files independent
  - Extract common information if files dependent
- Delivery should be based on joint cache-channel schemes
  - Piggyback information to stronger receiver on weak receiver, if the former in cache of the latter
  - Adaptive binning to compensate missing side-information
  - More insignts for interference channels [Maddah-Ali&Niesen2015]
- Cache design is influenced by optimal coding schemes
  $\rightarrow$ e.g., piggyback coding improves benefits of asymmetric cache sizes