Erasure Coding for Decentralized Caching

Soheil Mohajer
University of Minnesota

Mohammad Ali Maddah-Ali
Bell Labs, Alcatel-Lucent

14 Feb. 2017, ITA, San Diego
Problem Formulation

- $N$ files $X_1, X_2, X_3, \ldots, X_N$, $|X_i| = F$

- $K$ users, each with enough local storage to fetch $M$ files
  - without knowledge of the users’ request
    \[ Z_i = f_i(X_1, \ldots, X_N) \quad \text{and} \quad |Z_i| \leq MF \]
  - coded / uncoded data to be cached
  - centrally decided / random placement

- Each user requests for one file in the library
  \[ d = (d_1, d_2, \ldots, d_K) \in \{1, 2, \ldots, N\}^K \]
Problem Formulation

Delivery

\[ Y = g(X_1, \ldots, X_N; d; Z_1, \ldots, Z_K) \]

such that:

\[ (Y, Z_k) \sim X_{d_k} \quad k = 1, 2, \ldots, K \]

Questions:

- Optimal placement?
- Minimum delivery rate \[ R = \frac{|Y|}{F} \]
Uncoded vs. Coded Delivery

\[ N = 2 \left\{ \begin{array}{l} X_1 : \emptyset \hspace{1em} 1 \hspace{1em} 2 \hspace{1em} 12 \\ X_2 : \emptyset \hspace{1em} 1 \hspace{1em} 2 \hspace{1em} 12 \end{array} \right\} \]

\[ F \]

\[ Z_1 \]

\[ Z_2 \]

\[ d_1 = \]

\[ d_2 = \]
Centralized vs. Decentralized Placement

- Centralized: Placement is designed to maximize the utility

\[
R_C(M) = K \left(1 - \frac{M}{N}\right) \min \left\{ \frac{1}{1 + KM/N}, \frac{N}{K} \right\}.
\]


- Decentralized: No pre-setting is required, random fetching

\[
R_D(M) = K \left(1 - \frac{M}{N}\right) \min \left\{ \frac{N}{KM} (1 - (1 - M/N)^K), \frac{N}{K} \right\}.
\]

Centralized vs. Decentralized Caching

\[ \frac{R_D(M)}{R_C(M)} \leq 12 \]

numerically, 1.6

\( K = 20 \) users, \( N = 100 \) files

Uncoded vs. Coded Placement

- **Uncoded:** cache contents being union of pure packets from files

  - Several bounds with constant multiplicative gaps
    - [Maddah-Ali et al, 15-16], [Ghasemi et al, 16], [Wang et al, 16],
    [Sengupta at al, 16], Ajaykrishnan et al al, 15], [Tian, 15]

  - Proposed schemes are optimum when $K \leq N$
    - [Yu et al, 16], [Kai et al, 16]

  - Delivery rate can be improved when $K \geq N$
    - [Yu et al, 16], [Wan et al, 16]

- **Coded placement:** data can be coded before placement

  - Focus on joint coding (mixture) of files

  - Delivery rate can be improved! [Maddah-Ali et al, 15], [Tian & Chen.’16],…
Erasure Coding for Random Placement

\[ F \]

\[ N \]

\[ \{0, 0, 0\} \]

\[ 1 \oplus 2, 1 \oplus 3, 2 \oplus 3 \]

\[ 23 \oplus 13 \oplus 12 \]

\[ MF/N \]

\[ MF/N \]

\[ MF/N \]

\[ Z_1 \]

\[ Z_2 \]

\[ Z_3 \]

\[ erasure coding (r) \]

\[ F/r \]

\[ \{\emptyset, \emptyset, \emptyset\} \]

\[ 1 \oplus 2, 1 \oplus 3, 2 \oplus 3 \]

\[ 23 \oplus 13 \oplus 12 \]

\[ MF/N \]

\[ MF/N \]

\[ MF/N \]

\[ Z_1 \]

\[ Z_2 \]

\[ Z_3 \]

\[ MF/N \]

\[ MF/N \]

\[ MF/N \]
Erasure Coding for Caching

- Probability of caching each packet
- Number of packets can be jointly encoded for $u$ users
- Number of desired packets already cached

\[ q = \frac{MF}{NF} = \frac{Mr}{N} \]

\[ \beta_u = q^{u-1} (1 - q)^{K-u+1} \frac{F}{r} \]

\[ \frac{M}{N} F = \frac{q}{r} F \]
Erasure Coding for Caching

- Total number of desired packets cached or received at user

\[
\frac{q}{r} F + \sum_{u=1}^{K} \binom{K - 1}{u - 1} \lambda_u \beta_u \geq F
\]

pre-fetching delivery

- Total number of packets to be sent

\[
R(r; \lambda_1, \ldots, \lambda_K) = \frac{1}{F} \sum_{u=1}^{K} \binom{K}{u} \lambda_u \beta_u
\]
Optimization Problem

Optimization over $\lambda_i$'s

High-utility packets are preferable!

Optimization Problem

$$R(r) = \min_{\lambda_1, \ldots, \lambda_K} \sum_{u=1}^{K} \binom{K}{u} \lambda_u \beta_u$$

s.t.

$$\frac{q}{r} F + \sum_{u=1}^{K} \binom{K-1}{u-1} \lambda_u \beta_u \geq F$$

$$\lambda_u \in [0, 1] \quad u = 1, \ldots, K$$
Optimization Problem

\[ R(r) = \gamma \binom{K}{u^*} \beta_{u^*} + \sum_{u=u^*+1}^{K} \binom{K}{u} \beta_u \]

where \( \gamma \) and \( u^* \) are uniquely determined from

\[
\frac{q}{r} F + \gamma \binom{K-1}{u^*-1} \beta_{u^*} + \sum_{u=u^*+1}^{K} \binom{K-1}{u-1} \beta_u = F
\]

\[ R^* = \min_{0<r\leq 1} R(r) \]
Skewed Histogram vs. Cut-off

\( \frac{\beta_u}{F} \) vs. utility level

- Decentralized caching
- Erasure coding & Decentralized caching
- Required packets from Erasure & Decentralized caching

\[ u^* = u \]

number of packages \(/ F\)

utility level
It works!

![Graph showing delivery rate vs. r with lines \( R_D \), \( R_{ED}(r) \), and \( R_C \) representing Decentralized Caching, Erasure Coding + Decentralized Caching, and default caching, respectively.](image)

With parameters \( K=10, N=45, M=5 \).
Cache-Rate Trade-off

K=10, N=45

Decentralized Placement
Erasure Coding & Decentralized Placement
Centralized Placement
A few Remarks

- Erasure pre-coding can improve the delivery rate of decentralized caching
- It is low-complexity compared to file combination
- Asymptotically meets the centralized caching for small cache size

Questions!