

Locally Private Compression

Aslan Tchamkerten

Telecom Paris

Venkat Chandar

DE Shaw

Sidharth Jaggi

U Bristol

June 19th

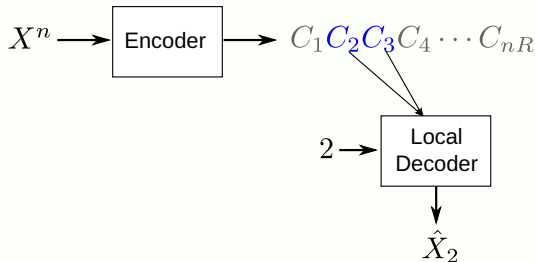
Shashank Vatedka

IIT Hyderabad

Locally private compression

Compress a DMS X^n such that

- X_i can be retrieved from a **small subset** of compressed bits
- which reveal **no information** about the other bits $X_{[n]\setminus i}$



Two extremes:

- No compression: perfect local decodability and privacy
- Virtually all existing compressors are not locally private.

Compression with local decodability and privacy

A rate- R privately locally decodable compression scheme consists of

- A randomized encoder

$$\left\{ p_{C^{nR}|x^n} : x^n \in \mathcal{X}^n \right\}$$

- n local decoders

$$\left\{ (\mathcal{I}_j, \widehat{X}_j = f_j(C_{\mathcal{I}_j})) : j \in [n] \right\}$$

Compression with local decodability and privacy


A rate- R privately locally decodable compression scheme consists of

- A randomized encoder


$$\left\{ p_{C^{nR}|x^n} : x^n \in \mathcal{X}^n \right\}$$

- n local decoders

$$\left\{ (\mathcal{I}_j, \widehat{X}_j = f_j(C_{\mathcal{I}_j})) : j \in [n] \right\}$$



locations
to probe
 $\mathcal{I}_j \subset [nR]$



local function
 $f_j : \{0,1\}^{|\mathcal{I}_j|} \rightarrow \{0,1\}$

Compression with local decodability and privacy

A rate- R privately locally decodable compression scheme consists of

- A randomized encoder

$$\left\{ p_{C^{nR}|x^n} : x^n \in \mathcal{X}^n \right\}$$

- n local decoders

$$\left\{ (\mathcal{I}_j, \widehat{X}_j = f_j(C_{\mathcal{I}_j})) : j \in [n] \right\}$$

Wanted: For $X^n \sim \text{i.i.d. } p_X$,

- **Reliability:** $\Pr[\widehat{X}_j \neq X_j] \rightarrow 0$ as $n \rightarrow \infty$
- **Privacy:** $C_{\mathcal{I}_j}$ should be independent of $X_{[n] \setminus j}$ for any $j \in [n]$

Related literature

- **Locally decodable source coding** without privacy constraints: (Makhdoumi et al. 2013, Mazumdar et al. 2015, Tatwawadi et al. 2018)
- **Locally private compression:**
 - (Chandar et al. 2023): not rate-optimal
 - (Chandar et al. 2024): rate-optimal, but complex scheme only for Bernoulli sources

Main result

Theorem

Let X^n be i.i.d. P . For any

$$R > H(P),$$

there exists a simple scheme with the following properties:

- compression at rate R ,
- perfect local privacy,
- error probability decays as $1/\text{poly}(n)$,
- encoding and local decoding run in $O(n \times \text{poly}(\log n))$ time.

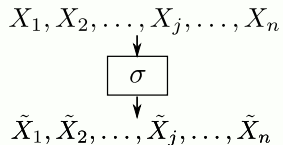
A natural scheme: permute-compress

Encoder picks a uniformly random permutation σ on $[n]$.

A natural scheme: permute-compress

Encoder picks a uniformly random permutation σ on $[n]$.

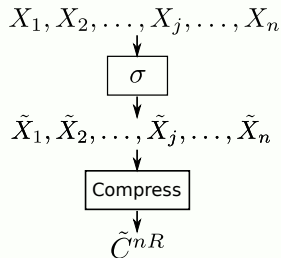
- Permute: $\tilde{X}_{\sigma(j)} = X_j$, for $j \in [n]$.



A natural scheme: permute-compress

Encoder picks a uniformly random permutation σ on $[n]$.

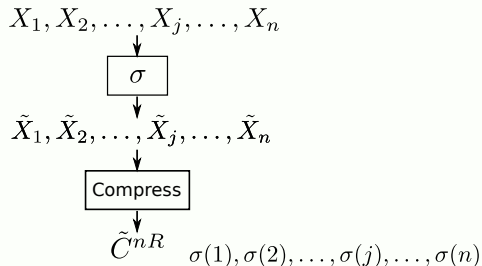
- Permute: $\tilde{X}_{\sigma(j)} = X_j$, for $j \in [n]$.
- Compress \tilde{X}^n to get \tilde{C}^{nR}



A natural scheme: permute-compress

Encoder picks a uniformly random permutation σ on $[n]$.

- Permute: $\tilde{X}_{\sigma(j)} = X_j$, for $j \in [n]$.
- Compress \tilde{X}^n to get \tilde{C}^{nR}
- Store: $(\tilde{C}^{nR}, \sigma(1), \dots, \sigma(n))$



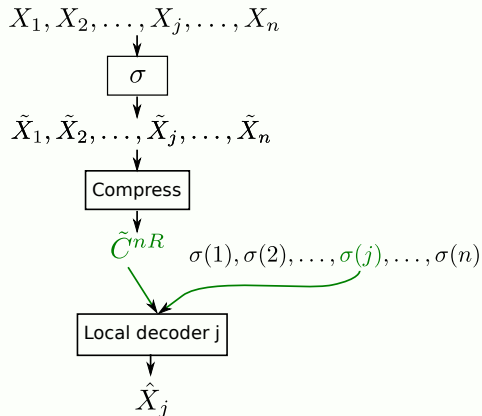
A natural scheme: permute-compress

Encoder picks a uniformly random permutation σ on $[n]$.

- **Permute:** $\tilde{X}_{\sigma(j)} = X_j$, for $j \in [n]$.
- **Compress** \tilde{X}^n to get \tilde{C}^{nR}
- **Store:** $(\tilde{C}^{nR}, \sigma(1), \dots, \sigma(n))$

To recover X_j :

- Decompress \tilde{C}^{nR} and read $\sigma(j)$ 'th location



A natural scheme: permute-compress

Encoder picks a uniformly random permutation σ on $[n]$.

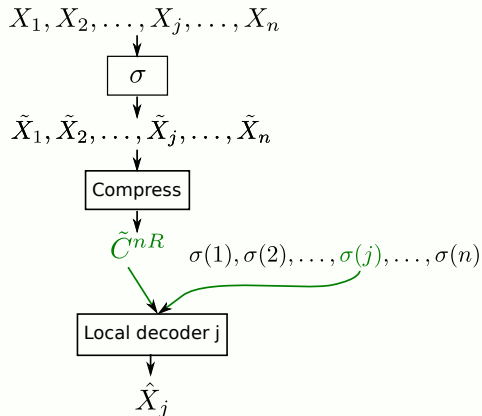
- **Permute:** $\tilde{X}_{\sigma(j)} = X_j$, for $j \in [n]$.
- **Compress** \tilde{X}^n to get \tilde{C}^{nR}
- **Store:** $(\tilde{C}^{nR}, \sigma(1), \dots, \sigma(n))$

To recover X_j :

- Decompress \tilde{C}^{nR} and read $\sigma(j)$ 'th location

Problems:

- No compression: storing σ requires $O(n \log n)$ bits!
- No privacy: decoder gets information about type of X^n



Fixing problems

- **Problem:** Storing σ requires $O(n \log n)$ bits
Solution: Break X^n into blocks of size $b = o\left(\frac{n}{\log n}\right)$, and use same σ for each block

Fixing problems

- **Problem:** Storing σ requires $O(n \log n)$ bits
Solution: Break X^n into blocks of size $b = o\left(\frac{n}{\log n}\right)$, and use same σ for each block
- **Problem:** \tilde{X}^n reveals type of X^n
Solution: Pad $\ll b$ extra symbols to each block to “freeze” type

Making each block constant composition

We partition X^n into blocks of size b each: $X^b(1), \dots, X^b(n/b)$

- W.h.p., blocks are typical. The empirical frequency:

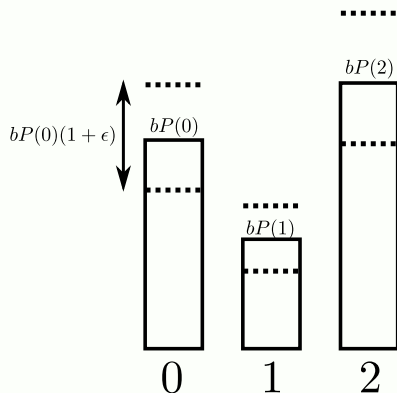
$$\eta_{X^b(i)}(x) \leq bP(x)(1 + \epsilon), \quad \forall x \in \mathcal{X}$$

- Probability that even one block is not typical $\leq b \times 2^{-\Theta(b)} = 1/\text{poly}(n \log n)$
- Pad $\approx \epsilon b$ symbols to give $\bar{X}^b(i)$ such that

$$\eta_{\bar{X}^b(i)}(x) = \lceil bP(x)(1 + \epsilon) \rceil, \quad \forall x \in \mathcal{X}$$

Question: Why are we guaranteed fixed-length pad that gives constant composition?

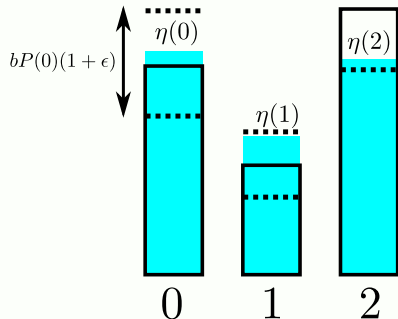
Fixed-length pad to get constant composition



Observation: If $\eta(x) > bP(x)$ for some x , then $\exists x'$ such that $\eta(x') < bP(x')$, since $\sum_{x \in \mathcal{X}} \eta(x) = b$.

Fixed-length pad to get constant composition

.....



Observation: If $\eta(x) > bP(x)$ for some x , then $\exists x'$ such that $\eta(x') < bP(x')$, since $\sum_{x \in \mathcal{X}} \eta(x) = b$.

For typical chunk, number of symbols to pad:

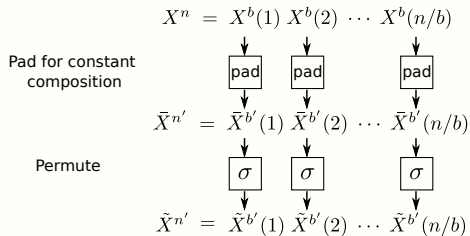
$$\begin{aligned}
 &= \sum_{x \in \mathcal{X}} \left(\underbrace{\lceil (1+\epsilon)bP(x) \rceil - \eta(x)}_{\geq 0} \right) \\
 &= \underbrace{\sum_{x \in \mathcal{X}} \lceil (1+\epsilon)bP(x) \rceil}_{\text{independent of } X^b} - \underbrace{\sum_{x \in \mathcal{X}} \eta(x)}_{=b} \\
 &\approx (1+\epsilon)b - b \\
 &= \epsilon b
 \end{aligned}$$

Summary: Pad-Permute-Compress (PPC) scheme

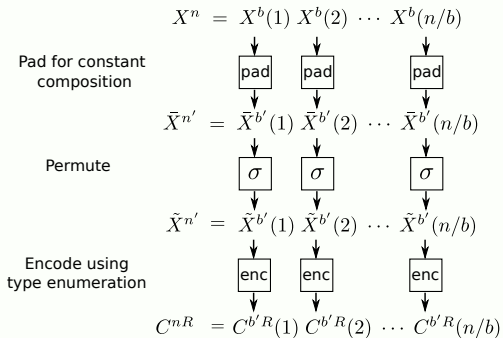
$$\begin{array}{c} X^n = X^{b(1)} X^{b(2)} \cdots X^{b(n/b)} \\ \downarrow \quad \downarrow \quad \downarrow \\ \text{pad} \quad \text{pad} \quad \text{pad} \\ \downarrow \quad \downarrow \quad \downarrow \\ \bar{X}^{n'} = \bar{X}^{b'}(1) \bar{X}^{b'}(2) \cdots \bar{X}^{b'}(n/b) \end{array}$$

Pad for constant composition

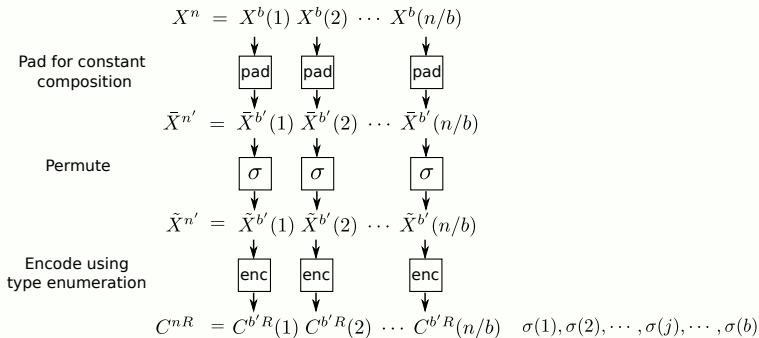
Summary: Pad-Permute-Compress (PPC) scheme



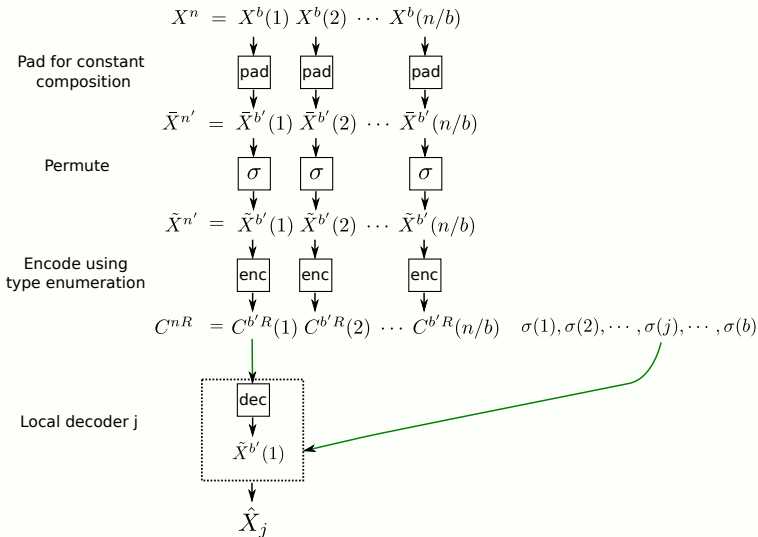
Summary: Pad-Permute-Compress (PPC) scheme



Summary: Pad-Permute-Compress (PPC) scheme



Summary: Pad-Permute-Compress (PPC) scheme



Analysis of PPC scheme

- Probability of error: Error occurs only if chunk is atypical.

$$P_e = 2^{-\Theta(b)} = \frac{1}{\text{poly}(n)}, \quad \text{if } b = \Omega(\log n)$$

- Rate: As long as $b = O(n/\log n)$,

$$nR = \frac{n}{b} \times \left[\underbrace{bH(P) + o(b)}_{\text{type enumeration}} + \underbrace{\epsilon b}_{\text{pad}} \right] + \underbrace{O(b \log b)}_{\text{permutation}} = n(H(P) + \epsilon + o(1))$$

- Privacy: Decoder only gets \hat{X}_j and type of $\tilde{X}^b(i)$ (independent of X^n given X_j)
- Computational complexity: Blocks are processed independently, type enumeration (which is most complex) can be performed in $O(b^3 \log b)$ time¹

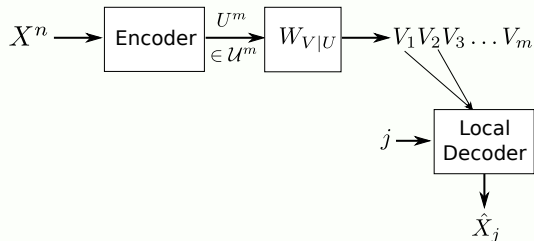
¹T. Cover, "Enumerative Source Encoding," IEEE Trans. Inf. Theory, 1973

Extensions

- **Recovery of contiguous substrings:** Easy extension of scheme to recover $X_{j_1:j_2}$, for arbitrary j_1, j_2 such that $j_2 - j_1 \leq j_{\max} \ll n$ is known.
Need to use multiple random permutations depending on j_{\max}

Extensions

- **Recovery of contiguous substrings:** Easy extension of scheme to recover $X_{j_1:j_2}$, for arbitrary j_1, j_2 such that $j_2 - j_1 \leq j_{\max} \ll n$ is known. Need to use multiple random permutations depending on j_{\max}
- **Joint source-channel coding:** Given DMS $X^n \sim P$ and DMC $W_{V|U}$. Recover any X_j w.h.p. from subset of symbols of V^m , while ensuring these symbols do not reveal $X_{[n]\setminus i}$



Open problems

- Sources with memory
- Recovery of X_{j_1}, X_{j_2} for *arbitrary* $j_1 \neq j_2$
- Minimum number of bits to be probed for locally private decoding?