Weighted Parity-Check Codes for Channels with State and Asymmetric Channels

Chih Wei Ling, Yanxiao Liu and Cheuk Ting Li
chihweiLing@link.cuhk.edu.hk
yanxiaoliu@link.cuhk.edu.hk
ctli@ie.cuhk.edu.hk

Department of Information Engineering, The Chinese University of Hong Kong

1st July 2022

Table of Contents

Introduction and Motivation

- Background: Find practical construction for the Gelfand-Pinsker setting with cost constraint and for the asymmetric channels

Introduction and Motivation

- Background: Find practical construction for the Gelfand-Pinsker setting with cost constraint and for the asymmetric channels
  - Barron et al. [2003] proposed nested linear code, but not sparse

Introduction and Motivation

- Background: Find practical construction for the Gelfand-Pinsker setting with cost constraint and for the asymmetric channels
    - Barron et al. [2003] proposed nested linear code, but not sparse
    - Martinian and Wainwright [2006] used sparse graphical code to generate nested linear code which is practical

Introduction and Motivation

- Background: Find practical construction for the Gelfand-Pinsker setting with cost constraint and for the asymmetric channels
  - Barron et al. [2003] proposed nested linear code, but not sparse
  - Martinian and Wainwright [2006] used sparse graphical code to generate nested linear code which is practical
  - Li and Anantharam [2021] proposed Poisson functional representation construction, with the best known second-order error bound [Scarlett, 2015] compared to other finite-blocklength schemes, but is impractical

Introduction and Motivation

- Background: Find practical construction for the Gelfand-Pinsker setting with cost constraint and for the asymmetric channels
  - Barron et al. [2003] proposed nested linear code, but not sparse
  - Martinian and Wainwright [2006] used sparse graphical code to generate nested linear code which is practical
  - Li and Anantharam [2021] proposed Poisson functional representation construction, with the best known second-order error bound [Scarlett, 2015] compared to other finite-blocklength schemes, but is impractical
- Our goal is to construct a code that is:
  - Practical

Introduction and Motivation

- Background: Find practical construction for the Gelfand-Pinsker setting with cost constraint and for the asymmetric channels
  - Barron et al. [2003] proposed nested linear code, but not sparse
  - Martinian and Wainwright [2006] used sparse graphical code to generate nested linear code which is practical
  - Li and Anantharam [2021] proposed Poisson functional representation construction, with the best known second-order error bound [Scarlett, 2015] compared to other finite-blocklength schemes, but is impractical
- Our goal is to construct a code that is:
  - Practical
  - Applicable to asymmetric channels (unlike Barron et al. [2003])

Introduction and Motivation

- Background: Find practical construction for the Gelfand-Pinsker setting with cost constraint and for the asymmetric channels
  - Barron et al. [2003] proposed nested linear code, but not sparse
  - Martinian and Wainwright [2006] used sparse graphical code to generate nested linear code which is practical
  - Li and Anantharam [2021] proposed Poisson functional representation construction, with the best known second-order error bound [Scarlett, 2015] compared to other finite-blocklength schemes, but is impractical
- Our goal is to construct a code that is:
  - Practical
  - Applicable to asymmetric channels (unlike Barron et al. [2003])
  - Having error performance as good as (and sometimes better than) the construction in Barron et al. [2003]

Query Functions

- Let $\mathbf{H} \in \mathbb{F}_2^{n \times n}$ be a full-rank matrix, called the *full parity check matrix*
  - $\mathbf{H}$ uniformly chosen random full-rank matrix
  - Also works for sparse $\mathbf{H}$, but the analysis is left for future study

Query Functions

- Let $\mathbf{H} \in \mathbb{F}_2^{n\times n}$ be a full-rank matrix, called the *full parity check matrix*
  - $\mathbf{H}$ uniformly chosen random full-rank matrix
  - Also works for sparse $\mathbf{H}$, but the analysis is left for future study
- For a *bias vector* $\mathbf{q} = [q_1, \ldots, q_n] \in [0,1]^n$, define the $\mathbf{q}$-*weight* of a vector $\mathbf{u} \in \mathbb{F}_2^n$ as

$$w_{\mathbf{q}}(\mathbf{u}) := \prod_{i=1}^{n} q_i^{u_i}(1-q_i)^{1-u_i} = P_{x_i\sim\mathrm{Bern}(q_i)}(\mathbf{x} = \mathbf{u})$$

Query Functions

- Let $\mathbf{H} \in \mathbb{F}_2^{n \times n}$ be a full-rank matrix, called the *full parity check matrix*
  - $\mathbf{H}$ uniformly chosen random full-rank matrix
  - Also works for sparse $\mathbf{H}$, but the analysis is left for future study
- For a *bias vector* $\mathbf{q} = [q_1, \ldots, q_n] \in [0,1]^n$, define the $\mathbf{q}$-*weight* of a vector $\mathbf{u} \in \mathbb{F}_2^n$ as

$$w_{\mathbf{q}}(\mathbf{u}) := \prod_{i=1}^{n} q_i^{u_i}(1-q_i)^{1-u_i} = P_{x_i \sim \mathrm{Bern}(q_i)}(\mathbf{x} = \mathbf{u})$$

Definition

Given the bias vectors $\mathbf{p}, \mathbf{q} \in [0,1]^n$ (we call $\mathbf{p}$ the *codeword bias*, and $\mathbf{q}$ the *parity bias*), the *query function* with respect to $\mathbf{H}$ is given by

$$f_{\mathbf{H}}(\mathbf{p}, \mathbf{q}) := \mathrm{argmax}_{\mathbf{x} \in \mathbb{F}_2^n} w_{\mathbf{p}}(\mathbf{x}) w_{\mathbf{q}}(\mathbf{x}\mathbf{H}^T) \tag{1}$$

Weighted Parity-Check Codes (WPC)

Definition: Encoder

Given the *encoder codeword bias function* $\mathbf{p}_e : \mathbb{F}_2^k \to [0,1]^n$, which maps a message $\mathbf{m} \in \mathbb{F}_2^k$ (and other information available at the encoder) to a bias vector $\mathbf{p}_e(\mathbf{m})$, and the *encoder parity bias function* $\mathbf{q}_e : \mathbb{F}_2^k \to [0,1]^n$. The encoding function is

$$\mathbf{m} \mapsto \mathbf{x} = f_{\mathbf{H}}\left(\mathbf{p}_e(\mathbf{m}), \mathbf{q}_e(\mathbf{m})\right) \tag{2}$$

Weighted Parity-Check Codes (WPC)

Definition: Encoder

Given the *encoder codeword bias function* $\mathbf{p}_e : \mathbb{F}_2^k \to [0,1]^n$, which maps a message $\mathbf{m} \in \mathbb{F}_2^k$ (and other information available at the encoder) to a bias vector $\mathbf{p}_e(\mathbf{m})$, and the *encoder parity bias function* $\mathbf{q}_e : \mathbb{F}_2^k \to [0,1]^n$. The encoding function is

$$\mathbf{m} \mapsto \mathbf{x} = f_{\mathbf{H}}\left(\mathbf{p}_e(\mathbf{m}),\, \mathbf{q}_e(\mathbf{m})\right) \tag{2}$$

Definition: Decoder

Similarly, given the *decoder codeword and parity bias functions* $\mathbf{p}_d, \mathbf{q}_d : \mathbb{F}_2^n \to [0,1]^n$. For a corrupted version $\mathbf{y}$ of $\mathbf{x}$, the decoding function is

$$\mathbf{y} \mapsto \hat{\mathbf{m}} = \left[(\hat{\mathbf{x}}\mathbf{H}^T)_1, \ldots, (\hat{\mathbf{x}}\mathbf{H}^T)_k\right], \tag{3}$$

where

$$\hat{\mathbf{x}} := f_{\mathbf{H}}\left(\mathbf{p}_d(\mathbf{y}),\, \mathbf{q}_d(\mathbf{y})\right) \tag{4}$$

Recovering Conventional Linear Codes by WPC

- Binary symmetric channel with parameter β, i.e., $P(y_i|x_i)$ is $\mathrm{BSC}(\beta)$

Recovering Conventional Linear Codes by WPC

- Binary symmetric channel with parameter β, i.e., $P(y_i|x_i)$ is $\mathrm{BSC}(\beta)$
- To recover the conventional linear code, we take

$$\mathbf{p}_e(\mathbf{m}) = \frac{1}{2}\mathbf{1}^n, \qquad \mathbf{q}_e(\mathbf{m}) = [\mathbf{m}, \mathbf{0}^{n-k}],$$

$$\mathbf{p}_d(\mathbf{y}) = \beta\mathbf{1}^n + (1-2\beta)\mathbf{y}, \qquad \mathbf{q}_d(\mathbf{y}) = \frac{1}{2}\mathbf{1}^n,$$

and substitute into Equations (2) and (4)

Recovering Conventional Linear Codes by WPC

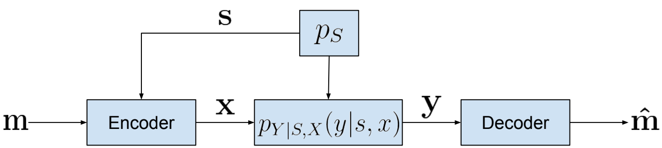
- Binary symmetric channel with parameter β, i.e., $P(y_i|x_i)$ is $\mathrm{BSC}(\beta)$
- To recover the conventional linear code, we take

$$\mathbf{p}_e(\mathbf{m}) = \frac{1}{2}\mathbf{1}^n, \qquad \mathbf{q}_e(\mathbf{m}) = [\mathbf{m},\, \mathbf{0}^{n-k}],$$

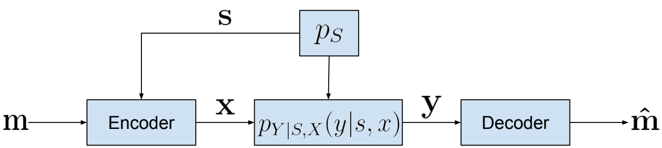
$$\mathbf{p}_d(\mathbf{y}) = \beta\mathbf{1}^n + (1-2\beta)\mathbf{y}, \qquad \mathbf{q}_d(\mathbf{y}) = \frac{1}{2}\mathbf{1}^n,$$

and substitute into Equations (2) and (4)
- Note that $w_{\mathbf{p}_d(\mathbf{y})}(\mathbf{x}) = P(\mathbf{x}|\mathbf{y})$ is the posterior distribution of $\mathbf{x}$

WPC for Gelfand-Pinsker Setting

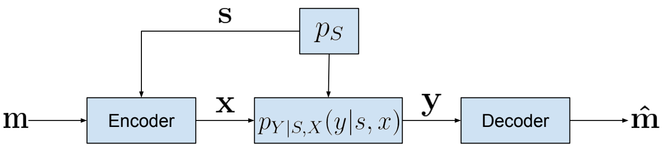


WPC for Gelfand-Pinsker Setting



- Assume x_i is binary, and s_i, y_i are arbitrary
  - Can generalize to larger x_i by considering F_l instead of F_2

WPC for Gelfand-Pinsker Setting

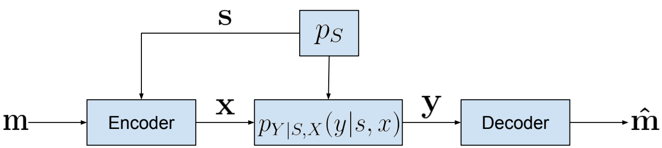


- Assume x_i is binary, and s_i, y_i are arbitrary
  - Can generalize to larger x_i by considering F_l instead of F_2
- Encoder: after observing $\mathbf{m}$ and $\mathbf{s}$, takes

$$\mathbf{p}_e(\mathbf{m},\mathbf{s}) = [p_e(s_1),\ldots,p_e(s_n)], \quad \mathbf{q}_e(\mathbf{m},\mathbf{s}) = [\mathbf{m},\ \mathbf{q}], \tag{5}$$

where we choose $p_e(s) = P_{X|S}(1|s)$ so $\mathbf{x}$ approximately follows $P_{X|S}$

WPC for Gelfand-Pinsker Setting



- Assume x_i is binary, and s_i, y_i are arbitrary
  - Can generalize to larger x_i by considering F_l instead of F_2
- Encoder: after observing $\mathbf{m}$ and $\mathbf{s}$, takes

$$\mathbf{p}_e(\mathbf{m},\mathbf{s}) = [p_e(s_1),\ldots,p_e(s_n)], \quad \mathbf{q}_e(\mathbf{m},\mathbf{s}) = [\mathbf{m},\,\mathbf{q}], \tag{5}$$

where we choose $p_e(s) = P_{X|S}(1|s)$ so $\mathbf{x}$ approximately follows $P_{X|S}$

- Decoder: after observing $\mathbf{y}$, takes

$$\mathbf{p}_d(\mathbf{y}) = [p_d(y_1),\ldots,p_d(y_n)], \quad \mathbf{q}_d(\mathbf{y}) = [\frac{1}{2}\mathbf{1}^k,\,\mathbf{q}], \tag{6}$$

and outputs $\hat{\mathbf{m}} = [(\hat{\mathbf{x}}\mathbf{H}^T)_1,\ldots,(\hat{\mathbf{x}}\mathbf{H}^T)_k]$, where $p_d(y) = P_{X|Y}(1|y)$

WPC is Capacity-Achieving

- We first state our main result as follows:

Theorem 1

- Assume $\mathbf{q} \sim P_Q$ i.i.d., where P_Q is a discrete distribution over $[0,1]$ with finite support satisfying

$$\mathbf{E}[H_b(Q)] = \frac{1 - H(X|S)}{1 - R}, \tag{7}$$

where $H_b : [0,1] \to [0,1]$ is the binary entropy function

WPC is Capacity-Achieving

- We first state our main result as follows:

Theorem 1

- Assume $\mathbf{q} \sim P_Q$ i.i.d., where P_Q is a discrete distribution over $[0, 1]$ with finite support satisfying

$$\mathbf{E}[H_b(Q)] = \frac{1 - H(X|S)}{1 - R}, \tag{7}$$

where $H_b : [0, 1] \to [0, 1]$ is the binary entropy function
- Then, for any $R < I(X; Y) - I(X; S)$, the probability of error of the code tends to 0, and the empirical joint distribution of $\{(s_i, x_i)\}_{i=1,\ldots,n}$ tends to $P_S P_{X|S}$ in probability as $n \to \infty$

WPC is Capacity-Achieving

- We first state our main result as follows:

Theorem 1

- Assume $\mathbf{q} \sim P_Q$ i.i.d., where P_Q is a discrete distribution over $[0, 1]$ with finite support satisfying

$$\mathbf{E}[H_b(Q)] = \frac{1 - H(X|S)}{1 - R}, \tag{7}$$

  where $H_b : [0, 1] \to [0, 1]$ is the binary entropy function
- Then, for any $R < I(X; Y) - I(X; S)$, the probability of error of the code tends to 0, and the empirical joint distribution of $\{(s_i, x_i)\}_{i=1,\ldots,n}$ tends to $P_S P_{X|S}$ in probability as $n \to \infty$

- Unlike nested linear codes, WPC also works for asymmetric $P_{X|S}$

WPC is Capacity-Achieving

- We first state our main result as follows:

Theorem 1

- Assume $\mathbf{q} \sim P_Q$ i.i.d., where P_Q is a discrete distribution over $[0, 1]$ with finite support satisfying

$$\mathbf{E}[H_b(Q)] = \frac{1 - H(X|S)}{1 - R}, \tag{7}$$

where $H_b : [0, 1] \to [0, 1]$ is the binary entropy function

- Then, for any $R < I(X; Y) - I(X; S)$, the probability of error of the code tends to 0, and the empirical joint distribution of $\{(s_i, x_i)\}_{i=1,\ldots,n}$ tends to $P_S P_{X|S}$ in probability as $n \to \infty$

- Unlike nested linear codes, WPC also works for asymmetric $P_{X|S}$
- Proof uses Sanov's theorem and robust typicality

How to Choose P_Q satisfying Equation (7)

- To achieve capacity, we need $\mathbf{E}[H_b(Q)] = \frac{1-H(X|S)}{1-R}$

How to Choose P_Q satisfying Equation (7)

- To achieve capacity, we need $\mathbf{E}[H_b(Q)] = \frac{1-H(X|S)}{1-R}$
- (Threshold) Take $P_Q(0) = P_Q(1) = (1-\gamma)/2$, $P_Q(1/2) = \gamma$, where $\gamma = (1 - H(X|S))/(1-R)$
  - Essentially equivalent to the nested linear code

How to Choose P_Q satisfying Equation (7)

- To achieve capacity, we need $\mathbf{E}[H_b(Q)] = \frac{1-H(X|S)}{1-R}$
- (Threshold) Take $P_Q(0) = P_Q(1) = (1-\gamma)/2$, $P_Q(1/2) = \gamma$, where $\gamma = (1 - H(X|S))/(1 - R)$
  - Essentially equivalent to the nested linear code
- (Linear) Take P_Q to be the uniform distribution $\mathrm{Unif}[0,1]$
  - $\mathbf{E}[H_b(Q)] = \frac{1-H(X|S)}{1-R}$ may not hold, not capacity achieving

How to Choose P_Q satisfying Equation (7)

- To achieve capacity, we need $\mathbf{E}[H_b(Q)] = \frac{1-H(X|S)}{1-R}$
- (Threshold) Take $P_Q(0) = P_Q(1) = (1-\gamma)/2$, $P_Q(1/2) = \gamma$, where $\gamma = (1 - H(X|S))/(1-R)$
  - Essentially equivalent to the nested linear code
- (Linear) Take P_Q to be the uniform distribution $\mathrm{Unif}[0,1]$
  - $\mathbf{E}[H_b(Q)] = \frac{1-H(X|S)}{1-R}$ may not hold, not capacity achieving
- (Threshold linear) Construct P_Q using the cdf

$$F_Q(t) := \begin{cases} 0 & \text{if } t < 0 \\ \max\{\theta/2,\, 0\} & \text{if } 0 \le t < |\theta|/2 \\ t & \text{if } |\theta|/2 \le t < 1 - |\theta|/2 \\ 1 - \max\{\theta/2,\, 0\} & \text{if } 1 - |\theta|/2 \le t < 1 \\ 1 & \text{if } t \ge 1 \end{cases} \tag{8}$$

where $\theta \in [-1, 1]$ is chosen s.t. $\mathbf{E}[H_b(Q)] = \frac{1-H(X|S)}{1-R}$

- Combines the linear method for t close to $1/2$, and the threshold method for smaller and larger t's

Simulation Result

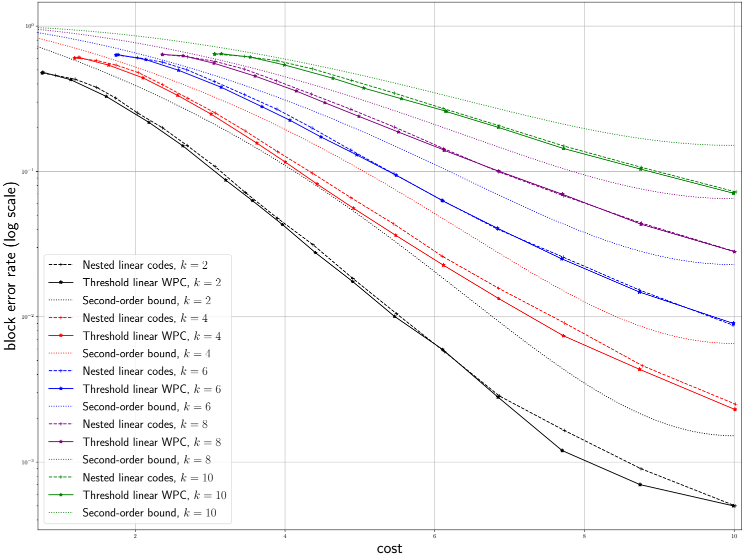


Figure: Performance evaluation with $n = 20$, $\beta = 0.05$

Conclusion and Discussions

- We propose the weighted parity check code, applicable to channels with state and asymmetric channels

Conclusion and Discussions

- We propose the weighted parity check code, applicable to channels with state and asymmetric channels
- Simulation results show that WPC achieves a smaller error probability compared to nested linear codes

Conclusion and Discussions

- We propose the weighted parity check code, applicable to channels with state and asymmetric channels
- Simulation results show that WPC achieves a smaller error probability compared to nested linear codes
- In the full paper [Ling et al., 2022], we show that our weighted construction also applies to the Wyner-Ziv setting [Wyner and Ziv, 1976]

Conclusion and Discussions

- We propose the weighted parity check code, applicable to channels with state and asymmetric channels
- Simulation results show that WPC achieves a smaller error probability compared to nested linear codes
- In the full paper [Ling et al., 2022], we show that our weighted construction also applies to the Wyner-Ziv setting [Wyner and Ziv, 1976]
- The code can be made more practical by considering a sparse parity-check matrix, though this is left for future work

Acknowledgments

The work of Cheuk Ting Li was supported in part by the Hong Kong Research Grant Council Grant ECS No. CUHK 24205621, and the Direct Grant for Research, The Chinese University of Hong Kong (Project ID: 4055133)

Reference

Richard J Barron, Brian Chen, and Gregory W Wornell. The duality between information embedding and source coding with side information and some applications. *IEEE Transactions on Information Theory*, 49 (5):1159–1180, 2003.

Cheuk Ting Li and Venkat Anantharam. A unified framework for one-shot achievability via the Poisson matching lemma. *IEEE Transactions on Information Theory*, 67(5):2624–2651, 2021.

Chih Wei Ling, Yanxiao Liu, and Cheuk Ting Li. Weighted parity-check codes for channels with state and asymmetric channels. *arXiv preprint arXiv:2201.10171*, 2022.

Emin Martinian and Martin J Wainwright. Low-density constructions can achieve the Wyner-Ziv and Gelfand-Pinsker bounds. pages 484–488, 2006.

Jonathan Scarlett. On the dispersions of the Gel'fand–Pinsker channel and dirty paper coding. *IEEE Trans. Inf. Theory*, 61(9):4569–4586, 2015.

A. D. Wyner and J. Ziv. The rate-distortion function for source coding with side information at the decoder. *IEEE Trans. Inf. Theory*, 22(1):