Channel Coding Techniques for Communication over Networks and Channels with Memory

Nadim Ghaddar

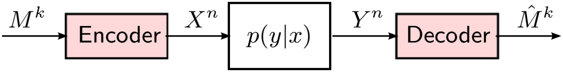
Department of Electrical and Computer Engineering
University of California San Diego

Ph.D. Defense

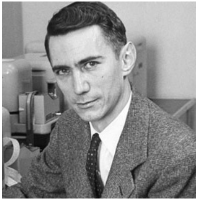
November 17, 2022

Point-to-Point Communication

- Shannon's model

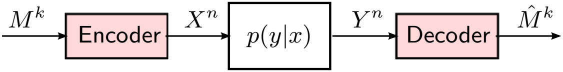


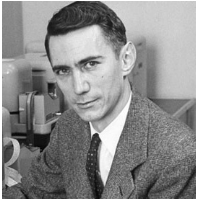
  - Rate $R = k/n$
  - Error probability $P_e = \mathsf{P}\{M^k \neq \hat{M}^k\}$



Point-to-Point Communication

- Shannon's model





  - Rate $R = k/n$
  - Error probability $P_e = \mathsf{P}\{M^k \neq \hat{M}^k\}$

P2P Channel Coding Theorem [Shannon 1948]

Channel $p(y\,|\,x)$ with **capacity** C:

- A family of codes with vanishing P_e exists only if $R \leq C$.
- For any $R < C$, a family of codes with vanishing P_e exists.

Road to Capacity

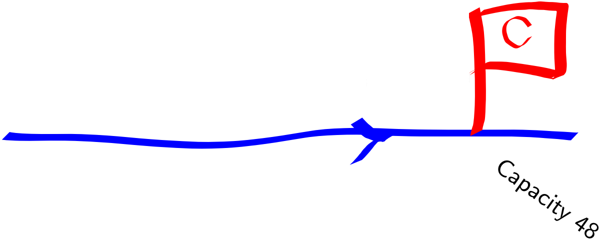


Image courtesy: Lele Wang

Road to Capacity

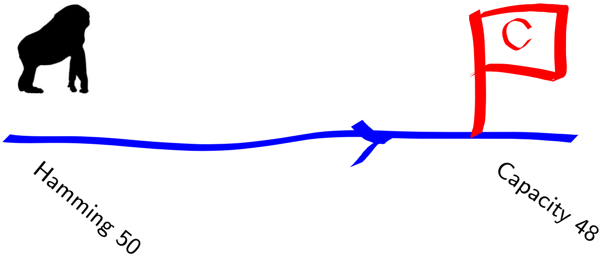


Image courtesy: Lele Wang

Road to Capacity

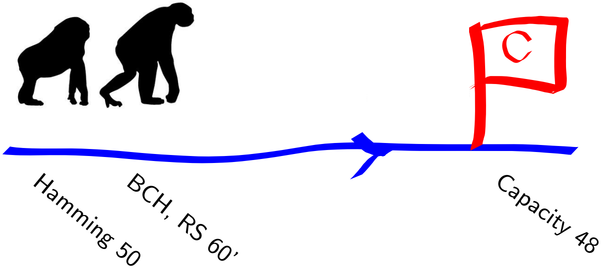


Image courtesy: Lele Wang

Road to Capacity

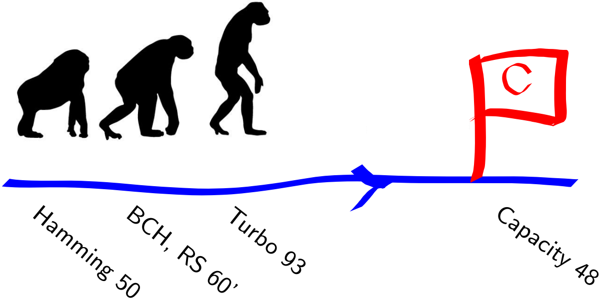


Image courtesy: Lele Wang

Road to Capacity

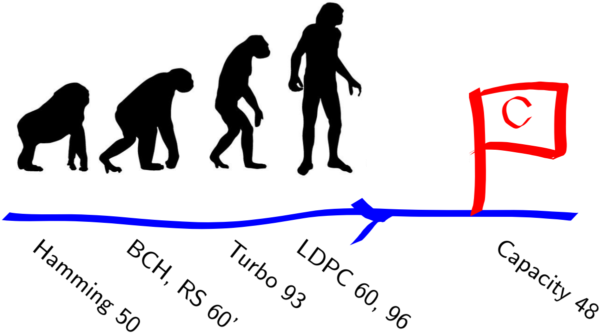


Image courtesy: Lele Wang

Road to Capacity

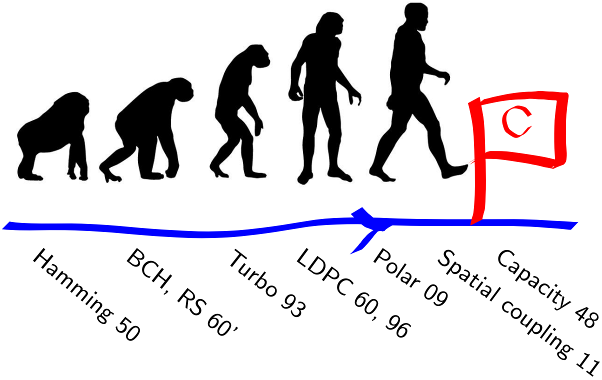
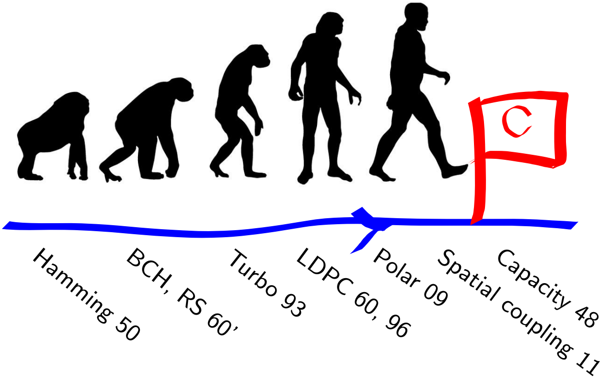


Image courtesy: Lele Wang

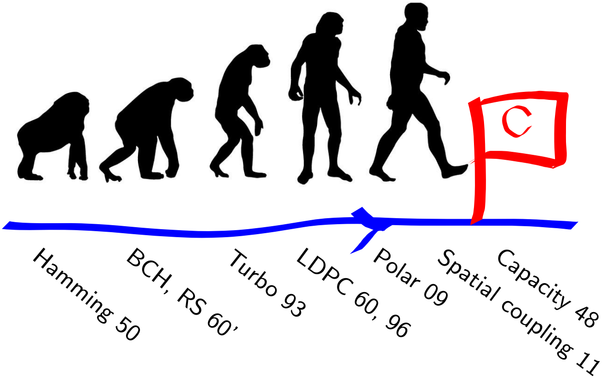
Road to Capacity



Approach capacity for all binary point-to-point memoryless symmetric (BMS) channels!

Image courtesy: Lele Wang

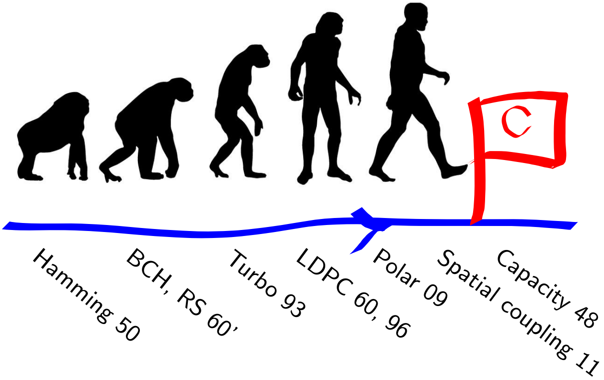
Road to Capacity



Approach capacity for all binary **point-to-point** **memoryless** **symmetric** (BMS) channels!

Image courtesy: Lele Wang

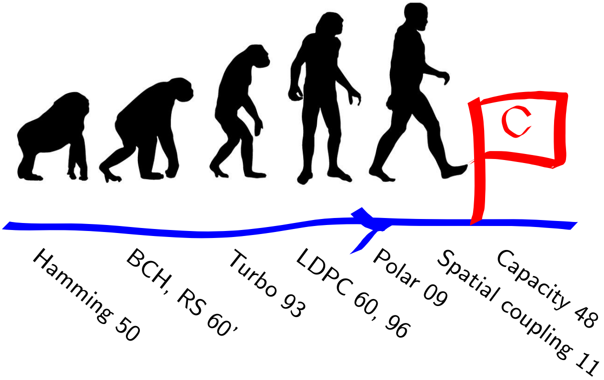
Road to Capacity



Approach capacity for all binary $\underbrace{\text{point-to-point}}_{\text{one sender, one receiver}}$ **memoryless** **symmetric** (BMS) channels!

Image courtesy: Lele Wang

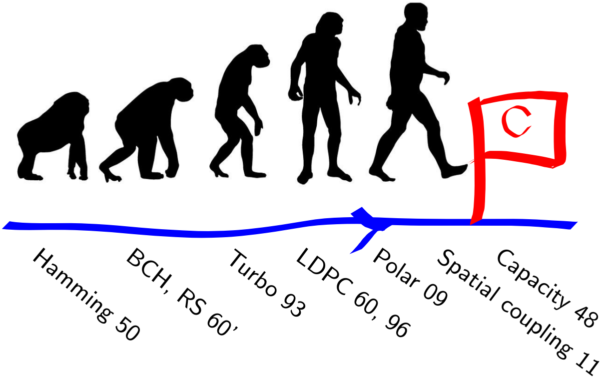
Road to Capacity



Approach capacity for all binary **point-to-point** $\underbrace{\textbf{memoryless}}_{\text{indep. channel uses}}$ **symmetric** (BMS) channels!

Image courtesy: Lele Wang

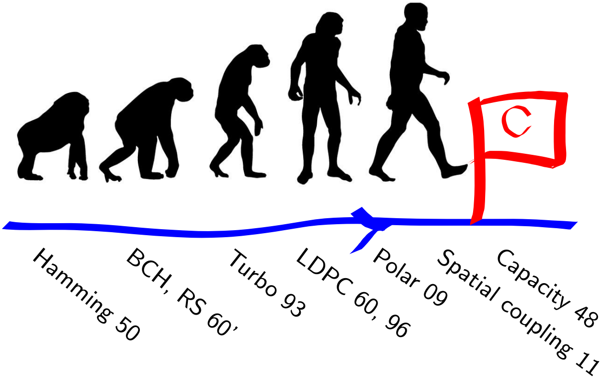
Road to Capacity



Approach capacity for all binary **point-to-point** **memoryless** **symmetric** (BMS) channels!

Image courtesy: Lele Wang

Road to Capacity



Approach capacity for all binary **point-to-point** **memoryless** **symmetric** (BMS) channels!

Image courtesy: Lele Wang

This Dissertation: Networks and Channels with Memory

I. Coding over networks: A Lego-brick approach

II. Joint channel estimation and polar coding over channels with memory

This Dissertation: Networks and Channels with Memory

I. Coding over networks: A Lego-brick approach

- Gelfand–Pinsker coding, asymmetric channel coding
- Marton coding over broadcast channels
- Distributed lossy compression
- Coding over cloud radio access networks (C-RAN's)

II. Joint channel estimation and polar coding over channels with memory

This Dissertation: Networks and Channels with Memory

I. Coding over networks: A Lego-brick approach

- Gelfand–Pinsker coding, asymmetric channel coding
- Marton coding over broadcast channels
- Distributed lossy compression
- Coding over cloud radio access networks (C-RAN's)

II. Joint channel estimation and polar coding over channels with memory

- Decoding algorithms that take into account the channel memory
- Pilot arrangement pattern that uses code structure
- Finite-state Markov channels
- Gauss-Markov channels
- Flat-fading channels

This Dissertation: Networks and Channels with Memory

I. Coding over networks: A Lego-brick approach

- **Gelfand–Pinsker coding, asymmetric channel coding**
- **Marton coding over broadcast channels**
- Distributed lossy compression
- Coding over cloud radio access networks (C-RAN's)

II. Joint channel estimation and polar coding over channels with memory

- Decoding algorithms that take into account the channel memory
- Pilot arrangement pattern that uses code structure
- **Finite-state Markov channels**
- Gauss-Markov channels
- Flat-fading channels

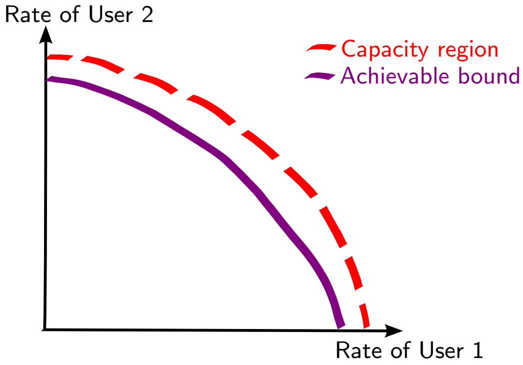
Coding over Networks: A Lego-Brick Approach

Coding over Networks

Network information theory: Characterizes achievable rates for network communication.

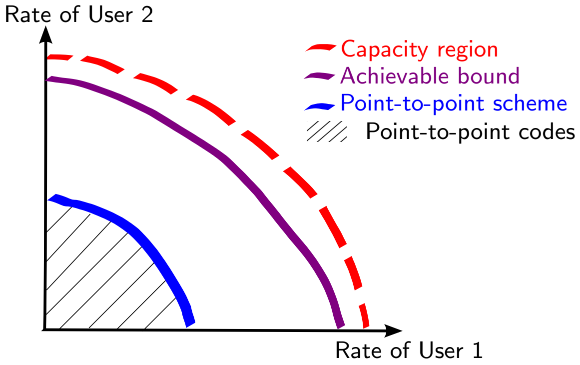
Coding over Networks

Network information theory: Characterizes achievable rates for network communication.



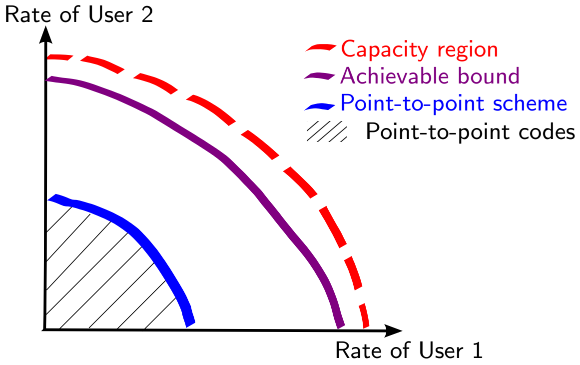
Coding over Networks

Network information theory: Characterizes achievable rates for network communication.



Coding over Networks

Network information theory: Characterizes achievable rates for network communication.



Goal: Construct low-complexity coding schemes over networks!

Previous Work

- Polar codes
  - Slepian–Wolf coding [Arıkan 2012]
  - Lossy source coding of a symmetric source [Korada–Urbanke 2010]
  - Multiple access channels [Şaşoğlu–Telatar–Yeh 2010, Abbe–Telatar 2012]
  - Broadcast channels [Mondelli–Hassani–Sason–Urbanke 2015]
  - Interference channels [Wang–Şaşoğlu 2014]
  - Relay channels [Wang 2015]

Previous Work

- Polar codes
  - Slepian–Wolf coding [Arıkan 2012]
  - Lossy source coding of a symmetric source [Korada–Urbanke 2010]
  - Multiple access channels [Şaşoğlu–Telatar–Yeh 2010, Abbe–Telatar 2012]
  - Broadcast channels [Mondelli–Hassani–Sason–Urbanke 2015]
  - Interference channels [Wang–Şaşoğlu 2014]
  - Relay channels [Wang 2015]

- Sparse graph codes with optimal decoding
  - Lossy source coding of a symmetric source [Matsunaga–Yamamoto 2003]
  - Gelfand–Pinsker and Wyner–Ziv coding [Muramatsu–Miyake 2010]

Previous Work

- Polar codes
  - Slepian–Wolf coding [Arıkan 2012]
  - Lossy source coding of a symmetric source [Korada–Urbanke 2010]
  - Multiple access channels [Şaşoğlu–Telatar–Yeh 2010, Abbe–Telatar 2012]
  - Broadcast channels [Mondelli–Hassani–Sason–Urbanke 2015]
  - Interference channels [Wang–Şaşoğlu 2014]
  - Relay channels [Wang 2015]

- Sparse graph codes with optimal decoding
  - Lossy source coding of a symmetric source [Matsunaga–Yamamoto 2003]
  - Gelfand–Pinsker and Wyner–Ziv coding [Muramatsu–Miyake 2010]

- Low-density generator matrix (LDGM) codes with message-passing decoding
  - Lossy source coding of a symmetric source [Wainwright–Maneva 2005, Aref–Macris–Vuffray 2015]

Previous Work

- Polar codes
  - Slepian–Wolf coding [Arıkan 2012]
  - Lossy source coding of a symmetric source [Korada–Urbanke 2010]
  - Multiple access channels [Şaşoğlu–Telatar–Yeh 2010, Abbe–Telatar 2012]
  - Broadcast channels [Mondelli–Hassani–Sason–Urbanke 2015]
  - Interference channels [Wang–Şaşoğlu 2014]
  - Relay channels [Wang 2015]
- Sparse graph codes with optimal decoding
  - Lossy source coding of a symmetric source [Matsunaga–Yamamoto 2003]
  - Gelfand–Pinsker and Wyner–Ziv coding [Muramatsu–Miyake 2010]
- Low-density generator matrix (LDGM) codes with message-passing decoding
  - Lossy source coding of a symmetric source [Wainwright–Maneva 2005, Aref–Macris–Vuffray 2015]
- Lattice codes
  - Gaussian channels with Gaussian state (dirty paper coding) [Erez–Shamai–Zamir 2005]

Lego-Brick Approach to Coding

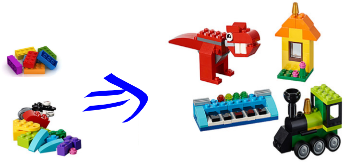
Question: What properties should P2P codes satisfy to be used for lossy source coding?

Lego-Brick Approach to Coding

Question: What properties should P2P codes satisfy to be used for lossy source coding?

"Lego-brick" approach to coding

Assemble codes in one communication setting $\Longrightarrow$ A code in a different setting

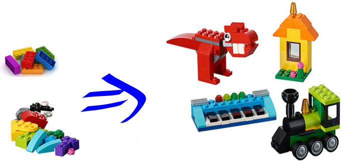


Lego-Brick Approach to Coding

Question: What properties should P2P codes satisfy to be used for lossy source coding?

"Lego-brick" approach to coding

Assemble codes in one communication setting $\Longrightarrow$ A code in a different setting



For a given coding problem,

- What "Lego bricks" to assemble, and what properties should they satisfy?
- How to assemble Lego bricks?
- How do performance guarantees translate?

Lego Bricks

- $p(y|x)$ is symmetric if $\exists\, \pi : \mathcal{Y} \to \mathcal{Y}$ s.t. $\pi^{-1} = \pi$ and $p(y|0) = p(\pi(y)|1)$, $\forall y$
  - BSC: $\pi(y) = y \oplus 1$
  - Binary-input AWGN: $\pi(y) = -y$

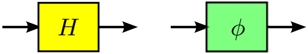
Lego Bricks

- $p(y|x)$ is symmetric if $\exists\, \pi: \mathcal{Y} \to \mathcal{Y}$ s.t. $\pi^{-1} = \pi$ and $p(y|0) = p(\pi(y)|1)$, $\forall y$
  - BSC: $\pi(y) = y \oplus 1$
  - Binary-input AWGN: $\pi(y) = -y$

- Note: Linear code ensembles can only achieve capacity of symmetric channels!

Lego Bricks

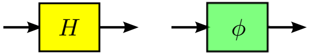
- $p(y|x)$ is symmetric if $\exists\, \pi : \mathcal{Y} \to \mathcal{Y}$ s.t. $\pi^{-1} = \pi$ and $p(y|0) = p(\pi(y)|1)$, $\forall y$
  - BSC: $\pi(y) = y \oplus 1$
  - Binary-input AWGN: $\pi(y) = -y$
- Note: Linear code ensembles can only achieve capacity of symmetric channels!
- Basic Lego Bricks:

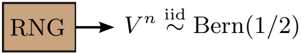


  - P2P code (H, ϕ) for BMS channel $p(y\,|\,x)$
  - Parity-check matrix H, decoder ϕ
  - Dimension k, blocklength n
  - Probability of error ϵ
  - $\cdots$

Lego Bricks

- $p(y|x)$ is symmetric if $\exists\, \pi : \mathcal{Y} \to \mathcal{Y}$ s.t. $\pi^{-1} = \pi$ and $p(y|0) = p(\pi(y)|1)$, $\forall y$
  - BSC: $\pi(y) = y \oplus 1$
  - Binary-input AWGN: $\pi(y) = -y$
- Note: Linear code ensembles can only achieve capacity of symmetric channels!
- Basic Lego Bricks:



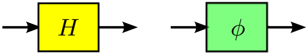


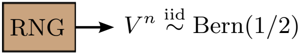
- P2P code (H, ϕ) for BMS channel $p(y\,|\,x)$
- Parity-check matrix H, decoder ϕ
- Dimension k, blocklength n
- Probability of error ϵ
- $\cdots$

- Random dither

Lego Bricks

- $p(y|x)$ is symmetric if $\exists\, \pi : \mathcal{Y} \to \mathcal{Y}$ s.t. $\pi^{-1} = \pi$ and $p(y|0) = p(\pi(y)|1)$, $\forall y$
  - BSC: $\pi(y) = y \oplus 1$
  - Binary-input AWGN: $\pi(y) = -y$
- Note: Linear code ensembles can only achieve capacity of symmetric channels!
- Basic Lego Bricks:





  - P2P code (H, ϕ) for BMS channel $p(y\,|\,x)$
  - Parity-check matrix H, decoder ϕ
  - Dimension k, blocklength n
  - Probability of error ϵ
  - $\cdots$

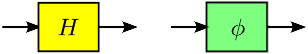
  - Random dither

- **Notation:** WLOG, let $H = \begin{bmatrix} A & B \end{bmatrix}$ where B is nonsingular, and define

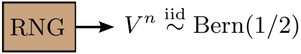
$$\widetilde{H} \triangleq \begin{bmatrix} \mathbf{0} \\ B^{-1} H \end{bmatrix}.$$

Lego Bricks

- $p(y|x)$ is symmetric if $\exists\, \pi\colon \mathcal{Y} \to \mathcal{Y}$ s.t. $\pi^{-1} = \pi$ and $p(y|0) = p(\pi(y)|1)$, $\forall y$
  - BSC: $\pi(y) = y \oplus 1$
  - Binary-input AWGN: $\pi(y) = -y$
- Note: Linear code ensembles can only achieve capacity of symmetric channels!
- Basic Lego Bricks:



  - P2P code (H, ϕ) for BMS channel $p(y\,|\,x)$
  - Parity-check matrix H, decoder ϕ
  - Dimension k, blocklength n
  - Probability of error ϵ
  - $\cdots$



  - Random dither

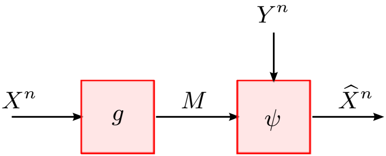
- **Notation:** WLOG, let $H = \begin{bmatrix} A & B \end{bmatrix}$ where B is nonsingular, and define

$$\widetilde{H} \triangleq \begin{bmatrix} \mathbf{0} \\ B^{-1}H \end{bmatrix}.$$

- Note: $\widetilde{H} \begin{bmatrix} \mathbf{0} \\ s^{n-k} \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ s^{n-k} \end{bmatrix}$, $\forall\, s^{n-k}$.

Example: Slepian–Wolf Problem

- Slepian–Wolf problem $p(x, y)$



$(X^n, Y^n) \overset{\text{iid}}{\sim} p(x, y)$

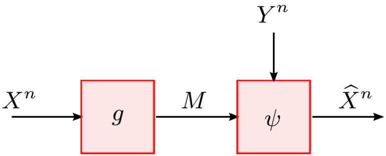
Encoder g

Decoder ψ

$P_e^{\text{SW}} = \mathsf{P}\{X^n \neq \widehat{X}^n\}$

Example: Slepian–Wolf Problem

- Slepian–Wolf problem $p(x, y)$



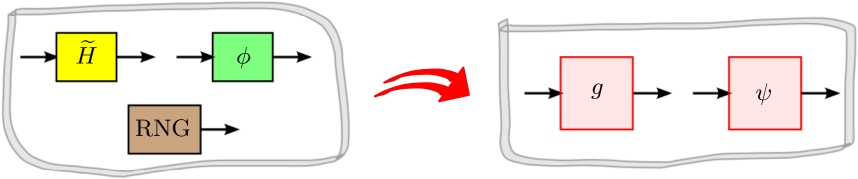
$(X^n, Y^n) \overset{\text{iid}}{\sim} p(x, y)$

Encoder g

Decoder ψ

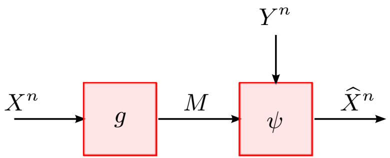
$P_e^{\text{SW}} = \mathsf{P}\{X^n \neq \widehat{X}^n\}$

- P2P code for BMS channel $\rightarrow$ Slepian–Wolf code



Basic Lego bricks **Slepian–Wolf code**

- Slepian–Wolf problem $p(x, y)$



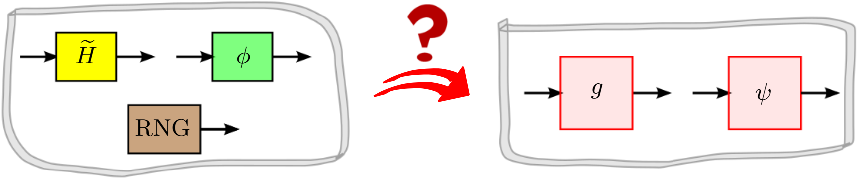
$(X^n, Y^n) \overset{\text{iid}}{\sim} p(x, y)$

Encoder g

Decoder ψ

$P_e^{\text{SW}} = \mathsf{P}\{X^n \neq \widehat{X}^n\}$

- P2P code for BMS channel $\to$ Slepian–Wolf code



Basic Lego bricks **Slepian–Wolf code**

Example: P2P code $\to$ Slepian–Wolf code [Wang–Kim 2015]

1) **"Codifying"**:

Lemma

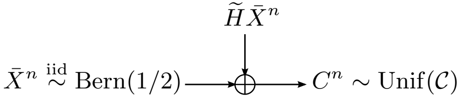
$$\bar{X}^n \overset{\text{iid}}{\sim} \text{Bern}(1/2) \quad \Longrightarrow \quad \bar{X}^n \oplus \widetilde{H}\bar{X}^n \sim \text{Unif}(\mathcal{C}).$$

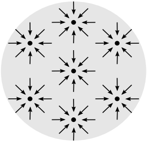
Example: P2P code $\to$ Slepian–Wolf code [Wang–Kim 2015]

1) **"Codifying"**:

Lemma

$$\bar{X}^n \overset{\text{iid}}{\sim} \mathrm{Bern}(1/2) \quad \Longrightarrow \quad \bar{X}^n \oplus \widetilde{H}\bar{X}^n \sim \mathrm{Unif}(\mathcal{C}).$$





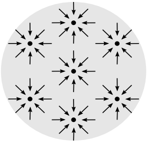
Example: P2P code $\to$ Slepian–Wolf code [Wang–Kim 2015]

1) **"Codifying"**:

Lemma

$$\bar{X}^n \overset{\text{iid}}{\sim} \text{Bern}(1/2) \quad \Longrightarrow \quad \bar{X}^n \oplus \widetilde{H}\bar{X}^n \sim \text{Unif}(\mathcal{C}).$$

$$\begin{tikzcd} & \widetilde{H}\bar{X}^n \arrow[d] & \\ \bar{X}^n \overset{\text{iid}}{\sim} \text{Bern}(1/2) \arrow[r] & \oplus \arrow[r] & C^n \sim \text{Unif}(\mathcal{C}) \end{tikzcd}$$



2) **"Symmetrization"**:

Lemma [Chen et al. 2009]

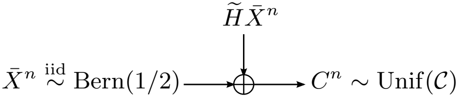
If $V \sim \text{Bern}(1/2) \perp\!\!\!\perp (X,Y)$, then $\bar{p}(y,v\,|\,x) := p_{X,Y}(x \oplus v, y)$ is symmetric

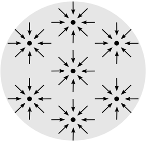
Example: P2P code → Slepian–Wolf code [Wang–Kim 2015]

1) **“Codifying”**:

Lemma

$\bar{X}^n \stackrel{\text{iid}}{\sim} \text{Bern}(1/2) \quad \Longrightarrow \quad \bar{X}^n \oplus \widetilde{H}\bar{X}^n \sim \text{Unif}(\mathcal{C}).$

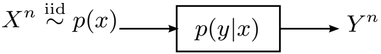




2) **“Symmetrization”**:

Lemma [Chen et al. 2009]

If $V \sim \text{Bern}(1/2) \perp\!\!\!\perp (X,Y)$, then $\bar{p}(y,v\,|\,x) := p_{X,Y}(x \oplus v, y)$ is symmetric



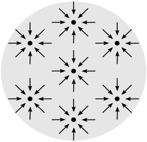
Example: P2P code → Slepian–Wolf code [Wang–Kim 2015]

1) **"Codifying"**:

Lemma

$$\bar{X}^n \overset{\text{iid}}{\sim} \text{Bern}(1/2) \quad \Longrightarrow \quad \bar{X}^n \oplus \widetilde{H}\bar{X}^n \sim \text{Unif}(\mathcal{C}).$$

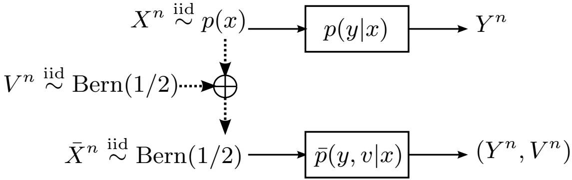
$$\bar{X}^n \overset{\text{iid}}{\sim} \text{Bern}(1/2) \longrightarrow \oplus \xleftarrow{} \widetilde{H}\bar{X}^n, \quad \oplus \longrightarrow C^n \sim \text{Unif}(\mathcal{C})$$



2) **"Symmetrization"**:

Lemma [Chen et al. 2009]

If $V \sim \text{Bern}(1/2) \perp\!\!\!\perp (X, Y)$, then $\bar{p}(y, v \,|\, x) := p_{X,Y}(x \oplus v, y)$ is symmetric



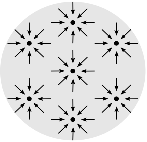
$\bar{p}$ is symmetric under $\pi\big((y, v)\big) = (y, v \oplus 1)$

1) **“Codifying”**:

Lemma

$$\bar{X}^n \overset{\text{iid}}{\sim} \mathrm{Bern}(1/2) \quad \Longrightarrow \quad \bar{X}^n \oplus \widetilde{H}\bar{X}^n \sim \mathrm{Unif}(\mathcal{C}).$$

$$\begin{tikzcd} & \widetilde{H}\bar{X}^n \arrow[d] & \\ \bar{X}^n \overset{\text{iid}}{\sim} \mathrm{Bern}(1/2) \arrow[r] & \oplus \arrow[r] & C^n \sim \mathrm{Unif}(\mathcal{C}) \end{tikzcd}$$



2) **“Symmetrization”**:

Lemma [Chen et al. 2009]

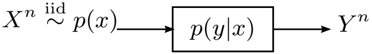
If $V \sim \mathrm{Bern}(1/2) \perp\!\!\!\perp (X, Y)$, then $\bar{p}(y, v \,|\, x) := p_{X,Y}(x \oplus v, y)$ is symmetric

$\bar{p}$ is symmetric under $\pi\big((y, v)\big) = (y, v \oplus 1)$

$\bar{p}$ is the *“symmetrized channel”* of $p(x, y)$

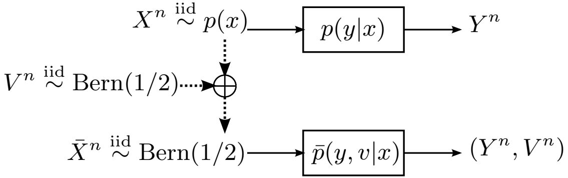
Example: P2P code $\to$ Slepian–Wolf code [Wang–Kim 2015]

- Slepian–Wolf problem $p(x, y)$



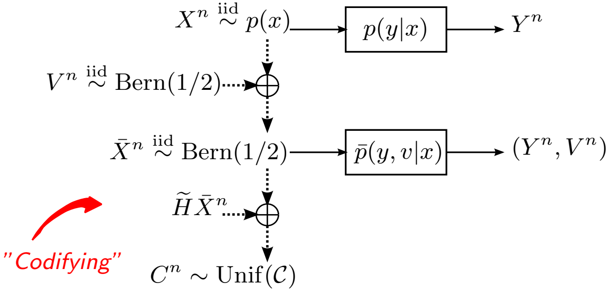
Example: P2P code $\to$ Slepian–Wolf code [Wang–Kim 2015]

- Slepian–Wolf problem $p(x, y)$



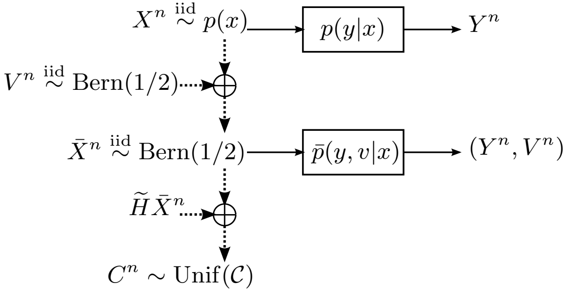
Example: P2P code → Slepian–Wolf code [Wang–Kim 2015]

- Slepian–Wolf problem $p(x, y)$



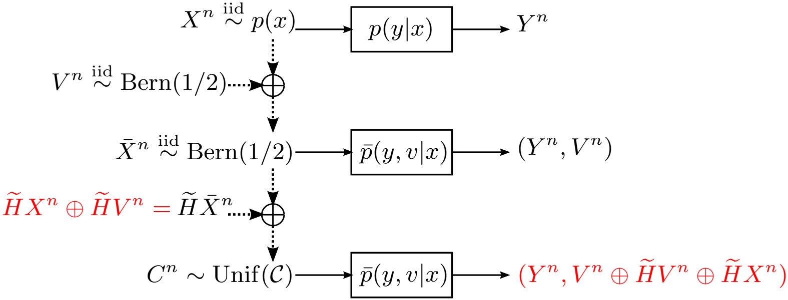
Example: P2P code → Slepian–Wolf code [Wang–Kim 2015]

- Slepian–Wolf problem $p(x, y)$



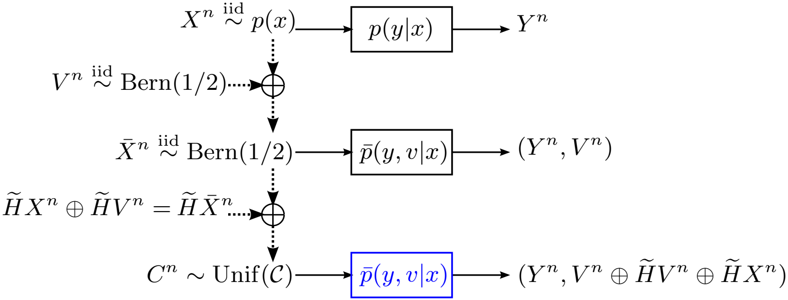
Example: P2P code → Slepian–Wolf code [Wang–Kim 2015]

- Slepian–Wolf problem $p(x, y)$

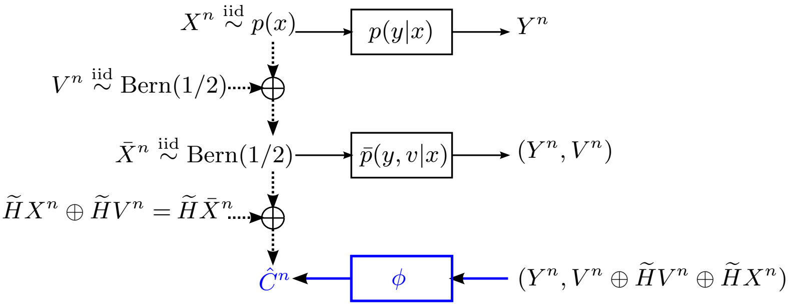


Example: P2P code → Slepian–Wolf code [Wang–Kim 2015]

- Slepian–Wolf problem $p(x, y)$

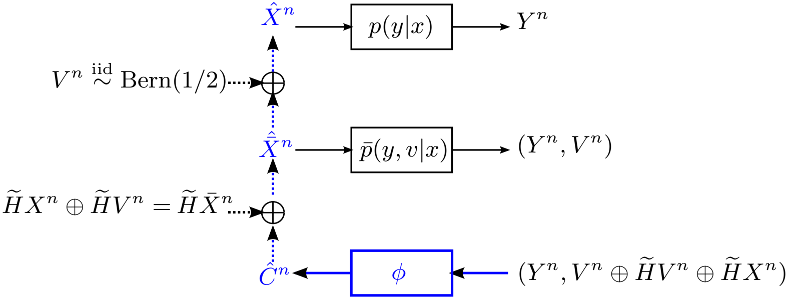


- Slepian–Wolf problem $p(x, y)$

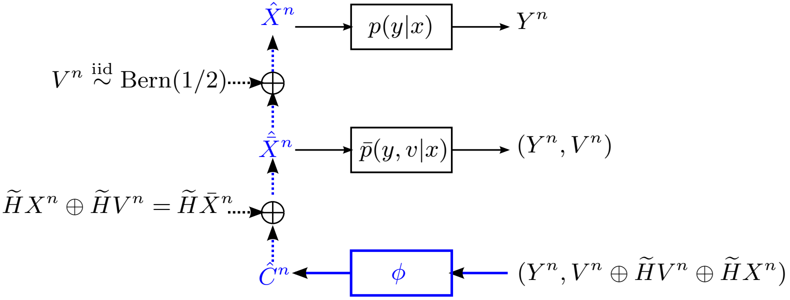


Example: P2P code → Slepian–Wolf code [Wang–Kim 2015]

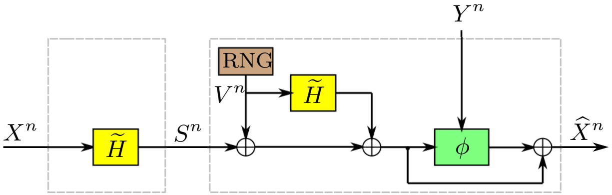
- Slepian–Wolf problem $p(x, y)$



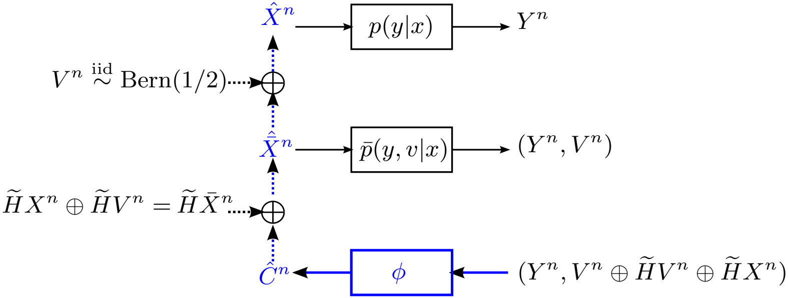
- Slepian–Wolf problem $p(x, y)$



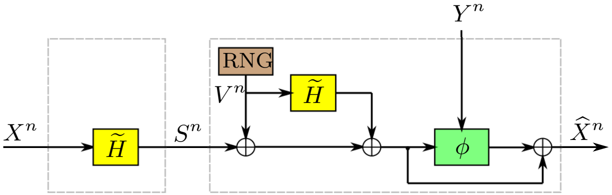
- Coding scheme: (H, ϕ) is a (k, n) code for $\bar{p}$ with error probability ϵ



- Slepian–Wolf problem $p(x,y)$



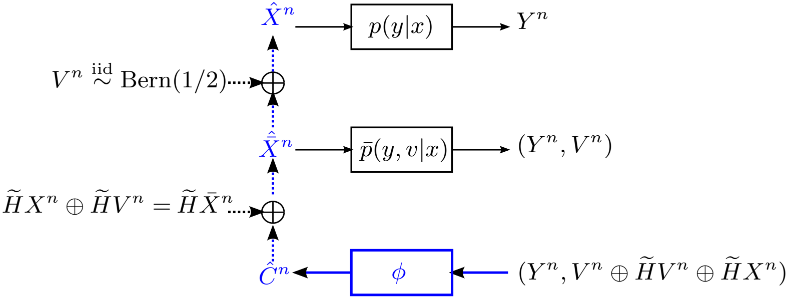
- Coding scheme: (H,ϕ) is a (k,n) code for $\bar{p}$ with error probability ϵ



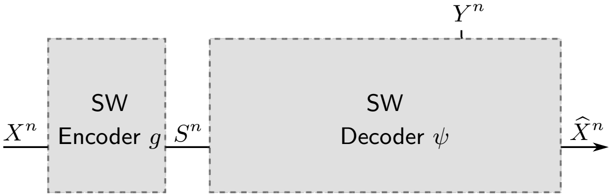
Theorem

$$R^{\text{SW}} = \frac{n-k}{n}$$
$$P_e^{\text{SW}} = \epsilon$$

- Slepian–Wolf problem $p(x,y)$



- Coding scheme: (H,ϕ) is a (k,n) code for $\bar{p}$ with error probability ϵ



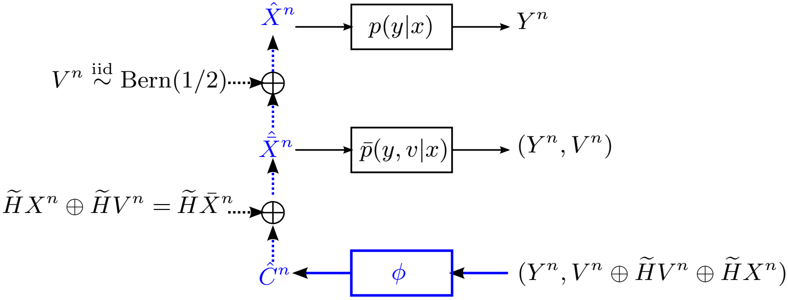
Theorem

$$R^{\text{SW}} = \frac{n-k}{n}$$
$$P_e^{\text{SW}} = \epsilon$$

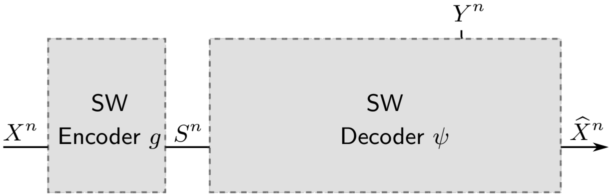
$(n-k,n)$ **SW code** (g,ψ)

Example: P2P code $\to$ Slepian–Wolf code [Wang–Kim 2015]

- Slepian–Wolf problem $p(x,y)$



- Coding scheme: (H,ϕ) is a (k,n) code for $\bar{p}$ with error probability ϵ



Theorem

$$R^{\text{SW}} = \frac{n-k}{n}$$
$$P_e^{\text{SW}} = \epsilon$$

$(n-k,n)$ **SW code** (H,ψ)

- This talk:

From Slepian–Wolf to Coding over Networks

- This talk:

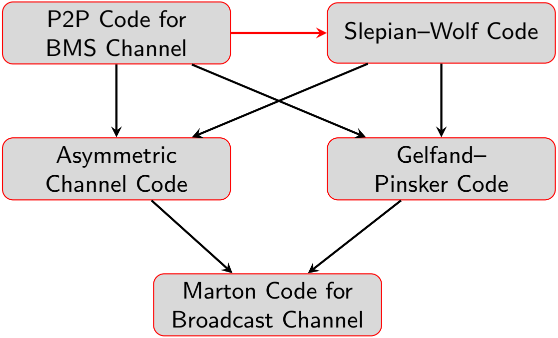
P2P Code for BMS Channel → Slepian–Wolf Code

Properties

Linearity
Error Probability

From Slepian–Wolf to Coding over Networks

- This talk:

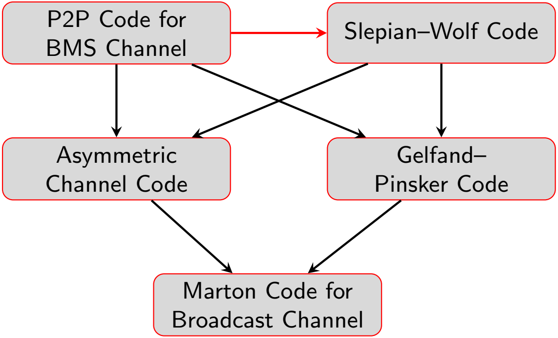


Properties

Linearity
Error Probability

From Slepian–Wolf to Coding over Networks

- This talk:



Properties

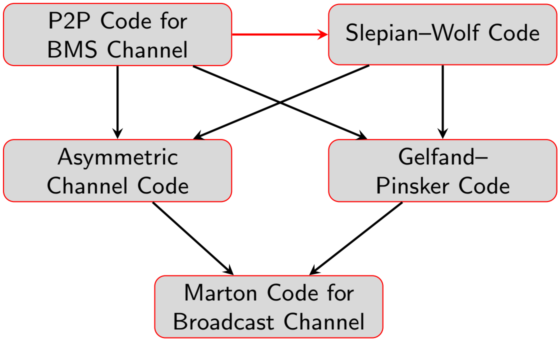
Linearity
Error Probability

Properties

Nested Linearity
Error Probability

From Slepian–Wolf to Coding over Networks

- This talk:



Properties

Linearity
Error Probability

Properties

Nested Linearity
Error Probability
Decoding Distance

Decoding Distance

- "Shaping property" of a decoding function

Decoding Distance

- "Shaping property" of a decoding function
- P2P code (H, ϕ) for BMS channel $p(y\,|\,x)$

$$Y^n \overset{\text{iid}}{\sim} p(y)$$

$$p(y) := \frac{1}{2}\sum_x p(y\,|\,x)$$

Decoding Distance

- "Shaping property" of a decoding function
- P2P code (H, ϕ) for BMS channel $p(y \,|\, x)$

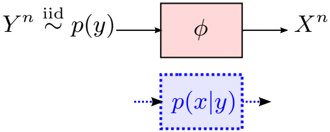
$$Y^n \overset{\text{iid}}{\sim} p(y) \longrightarrow \boxed{\phi} \longrightarrow X^n$$

$$p(y) := \frac{1}{2} \sum_x p(y \,|\, x)$$

$$X^n = \phi(Y^n)$$

Decoding Distance

- "Shaping property" of a decoding function
- P2P code (H, ϕ) for BMS channel $p(y\,|\,x)$



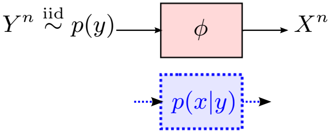
$$p(y) := \frac{1}{2}\sum_{x} p(y\,|\,x)$$

$$X^n = \phi(Y^n)$$

Question: How "far" is ϕ from the memoryless channel $p(x\,|\,y)$?

Decoding Distance

- "Shaping property" of a decoding function
- P2P code (H, ϕ) for BMS channel $p(y\,|\,x)$



$$p(y) := \frac{1}{2}\sum_x p(y\,|\,x)$$

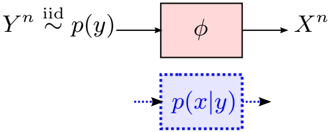
$$X^n = \phi(Y^n)$$

Question: How "far" is ϕ from the memoryless channel $p(x\,|\,y)$?

- $q(x^n, y^n)$: distribution of (X^n, Y^n)
  $p(x^n, y^n)$: i.i.d. distribution according to $\frac{1}{2}p(y\,|\,x)$

Decoding Distance

- "Shaping property" of a decoding function
- P2P code (H, ϕ) for BMS channel $p(y \,|\, x)$



$$p(y) := \frac{1}{2} \sum_x p(y \,|\, x)$$

$$X^n = \phi(Y^n)$$

Question: How "far" is ϕ from the memoryless channel $p(x \,|\, y)$?

- $q(x^n, y^n)$: distribution of (X^n, Y^n)
  $p(x^n, y^n)$: i.i.d. distribution according to $\frac{1}{2} p(y \,|\, x)$

$$\text{Decoding distance } \delta \triangleq \underbrace{\frac{1}{2} \sum_{x^n, y^n} \Big| q(x^n, y^n) - p(x^n, y^n) \Big|}_{\text{Total variation distance } d_{\mathrm{TV}}(p,q)}$$

Decoding Distance

- "Shaping property" of a decoding function
- P2P code (H, ϕ) for BMS channel $p(y\,|\,x)$

$$Y^n \overset{\text{iid}}{\sim} p(y) \longrightarrow \boxed{\phi} \longrightarrow X^n$$

$$\dashrightarrow \boxed{p(x|y)} \dashrightarrow$$

$$p(y) := \frac{1}{2}\sum_{x} p(y\,|\,x)$$

$$X^n = \phi(Y^n)$$

Question: How "far" is ϕ from the memoryless channel $p(x\,|\,y)$?

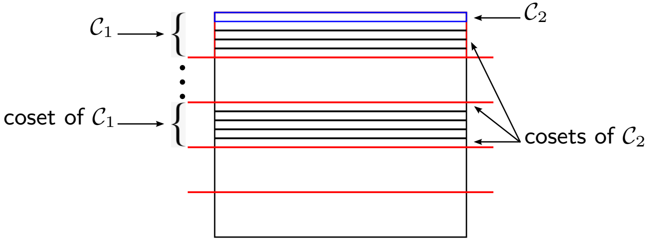
- $q(x^n, y^n)$: distribution of (X^n, Y^n)
  $p(x^n, y^n)$: i.i.d. distribution according to $\frac{1}{2}p(y\,|\,x)$

$$\text{Decoding distance } \delta \triangleq \underbrace{\frac{1}{2}\sum_{x^n, y^n}\left|q(x^n, y^n) - p(x^n, y^n)\right|}_{\text{Total variation distance } d_{\mathrm{TV}}(p,q)}$$

- Necessary & sufficient condition for vanishing δ: $R > 1 - H(X\,|\,Y)$
  - E.g., random codes [Bennett et. al 2002], polar codes [Korada–Urbanke 2010]

Nested Linear Codes

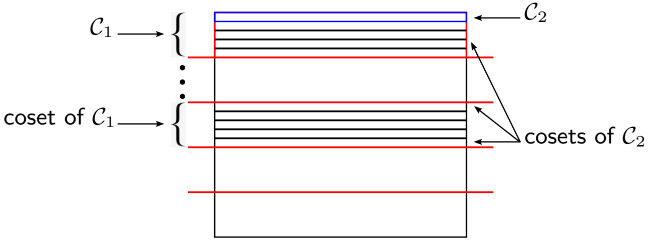
- Linear codes $\mathcal{C}_1$, $\mathcal{C}_2$ s.t. $\mathcal{C}_2 \subseteq \mathcal{C}_1$



- $2^{k_1 - k_2}$ cosets of $\mathcal{C}_2$ within a coset of $\mathcal{C}_1$

Nested Linear Codes

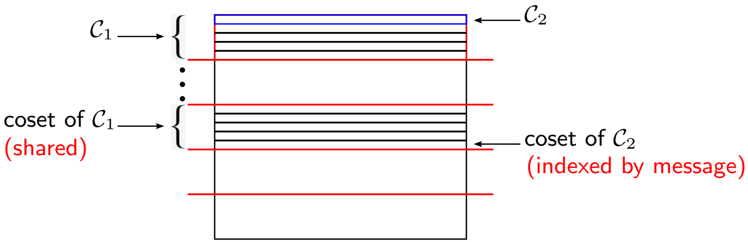
- Linear codes $\mathcal{C}_1$, $\mathcal{C}_2$ s.t. $\mathcal{C}_2 \subseteq \mathcal{C}_1$



- $2^{k_1-k_2}$ cosets of $\mathcal{C}_2$ within a coset of $\mathcal{C}_1$
- Applied to Gelfand–Pinsker & Marton coding [Padakandla–Pradhan 2011]
  - Coset of $\mathcal{C}_1$: uniformly chosen and shared between encoder and decoder
  - Coset shift of $\mathcal{C}_2$ within $\mathcal{C}_1$: indexed by message ($k_1 - k_2$ bits)

Nested Linear Codes

- Linear codes $\mathcal{C}_1$, $\mathcal{C}_2$ s.t. $\mathcal{C}_2 \subseteq \mathcal{C}_1$

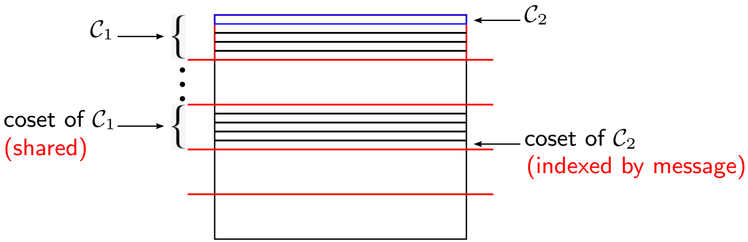


- $2^{k_1-k_2}$ cosets of $\mathcal{C}_2$ within a coset of $\mathcal{C}_1$
- Applied to Gelfand–Pinsker & Marton coding [Padakandla–Pradhan 2011]
  - Coset of $\mathcal{C}_1$: uniformly chosen and shared between encoder and decoder
  - Coset shift of $\mathcal{C}_2$ within $\mathcal{C}_1$: indexed by message ($k_1 - k_2$ bits)
  - Encoder: Finds a sequence in coset of $\mathcal{C}_2$ that has a desired distribution
  - Decoder: Finds a coset shift that includes a sequence having a desired distribution

  $\equiv$ Joint typicality encoding and decoding

Nested Linear Codes

- Linear codes $\mathcal{C}_1$, $\mathcal{C}_2$ s.t. $\mathcal{C}_2 \subseteq \mathcal{C}_1$



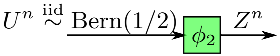
- $2^{k_1-k_2}$ cosets of $\mathcal{C}_2$ within a coset of $\mathcal{C}_1$
- Applied to Gelfand–Pinsker & Marton coding [Padakandla–Pradhan 2011]
  - Coset of $\mathcal{C}_1$: uniformly chosen and shared between encoder and decoder
  - Coset shift of $\mathcal{C}_2$ within $\mathcal{C}_1$: indexed by message ($k_1 - k_2$ bits)
  - Encoder: Finds a sequence in coset of $\mathcal{C}_2$ that has a desired distribution
  - Decoder: Finds a coset shift that includes a sequence having a desired distribution

  $\equiv$ Joint typicality encoding and decoding

- Note: We can choose H_1, H_2 s.t. $H_2 = \begin{bmatrix} H_1 \\ Q \end{bmatrix}$ for some matrix Q

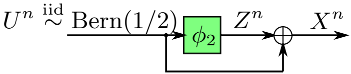
P2P Code + Slepian–Wolf Code $\longrightarrow$ Asymmetric Channel Code

- Goal: Code for asymmetric channel $p(y|x)$
- Approach: Target $p(x) \sim \mathrm{Bern}(\alpha)$ for some given α.
- Lego bricks:
  1. (k_2, n) P2P code (H_2, ϕ_2) for $\mathrm{BSC}(\alpha)$ with decoding distance δ



P2P Code + Slepian–Wolf Code $\longrightarrow$ Asymmetric Channel Code

- Goal: Code for asymmetric channel $p(y|x)$
- Approach: Target $p(x) \sim \mathrm{Bern}(\alpha)$ for some given α.
- Lego bricks:
  1. (k_2, n) P2P code (H_2, ϕ_2) for $\mathrm{BSC}(\alpha)$ with decoding distance δ

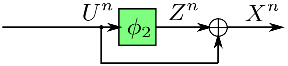


$$X^n = \phi_2(U^n) \oplus U^n$$

$$d_{\mathrm{TV}}\left(q_{U^n,Z^n}, \prod \mathrm{DSBS}(\alpha)\right) = \delta \quad \Longrightarrow \quad d_{\mathrm{TV}}\left(q_{X^n}, \prod \mathrm{Bern}(\alpha)\right) \le \delta$$

P2P Code + Slepian–Wolf Code $\longrightarrow$ Asymmetric Channel Code

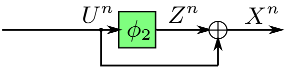
- Goal: Code for asymmetric channel $p(y|x)$
- Approach: Target $p(x) \sim \mathrm{Bern}(\alpha)$ for some given α.
- Lego bricks:
  1. (k_2, n) P2P code (H_2, ϕ_2) for $\mathrm{BSC}(\alpha)$ with decoding distance δ



P2P Code + Slepian–Wolf Code $\longrightarrow$ Asymmetric Channel Code

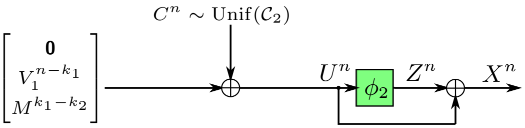
- Goal: Code for asymmetric channel $p(y|x)$
- Approach: Target $p(x) \sim \mathrm{Bern}(\alpha)$ for some given α.
- Lego bricks:
  1. (k_2, n) P2P code (H_2, ϕ_2) for $\mathrm{BSC}(\alpha)$ with decoding distance δ
  2. $(n - k_1, n)$ Slepian–Wolf code (H_1, ϕ_1) for $p(x, y)$ with error probability ϵ
  3. Two codes are nested s.t. $H_2 = \begin{bmatrix} H_1 \\ Q \end{bmatrix}$.

$$\begin{bmatrix} \mathbf{0} \\ V_1^{n-k_1} \\ M^{k_1-k_2} \end{bmatrix}$$



- $V_1^{n-k_1} \overset{\mathrm{iid}}{\sim} \mathrm{Bern}(1/2)$ is shared between encoder and decoder
- $M^{k_1-k_2}$ is the message

- Goal: Code for asymmetric channel $p(y \mid x)$
- Approach: Target $p(x) \sim \mathrm{Bern}(\alpha)$ for some given α.
- Lego bricks:
  1. (k_2, n) P2P code (H_2, ϕ_2) for $\mathrm{BSC}(\alpha)$ with decoding distance δ
  2. $(n - k_1, n)$ Slepian–Wolf code (H_1, ϕ_1) for $p(x, y)$ with error probability ϵ
  3. Two codes are nested s.t. $H_2 = \begin{bmatrix} H_1 \\ Q \end{bmatrix}$.

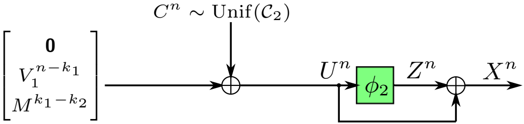


$U^n \overset{\text{iid}}{\sim} \mathrm{Bern}(1/2)$

- $V_1^{n-k_1} \overset{\text{iid}}{\sim} \mathrm{Bern}(1/2)$ is shared between encoder and decoder
- $M^{k_1-k_2}$ is the message
- $U^n \overset{\text{iid}}{\sim} \mathrm{Bern}(1/2)$

P2P Code + Slepian–Wolf Code $\longrightarrow$ Asymmetric Channel Code

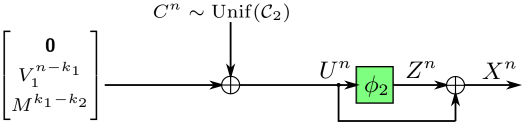
- Goal: Code for asymmetric channel $p(y|x)$
- Approach: Target $p(x) \sim \mathrm{Bern}(\alpha)$ for some given α.
- Lego bricks:
  1. (k_2, n) P2P code (H_2, ϕ_2) for $\mathrm{BSC}(\alpha)$ with decoding distance δ
  2. $(n-k_1, n)$ Slepian–Wolf code (H_1, ϕ_1) for $p(x,y)$ with error probability ϵ
  3. Two codes are nested s.t. $H_2 = \begin{bmatrix} H_1 \\ Q \end{bmatrix}$.



- $V_1^{n-k_1} \overset{\text{iid}}{\sim} \mathrm{Bern}(1/2)$ is shared between encoder and decoder
- $M^{k_1-k_2}$ is the message
- $U^n \overset{\text{iid}}{\sim} \mathrm{Bern}(1/2) \quad \Longrightarrow \quad d_{\mathrm{TV}}\left(q_{X^n}, \prod \mathrm{Bern}(\alpha)\right) \le \delta$

P2P Code + Slepian–Wolf Code $\longrightarrow$ Asymmetric Channel Code

- Goal: Code for asymmetric channel $p(y|x)$
- Approach: Target $p(x) \sim \mathrm{Bern}(\alpha)$ for some given α.
- Lego bricks:
  1. (k_2, n) P2P code (H_2, ϕ_2) for $\mathrm{BSC}(\alpha)$ with decoding distance δ
  2. $(n - k_1, n)$ Slepian–Wolf code (H_1, ϕ_1) for $p(x, y)$ with error probability ϵ
  3. Two codes are nested s.t. $H_2 = \begin{bmatrix} H_1 \\ Q \end{bmatrix}$.

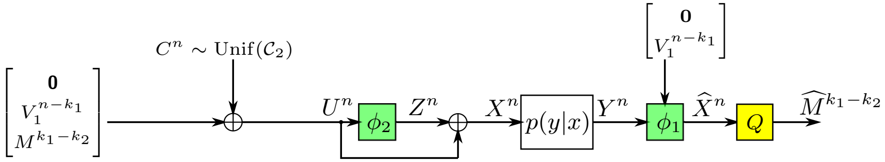


$H_1 X^n = V_1^{n-k_1}$

- $V_1^{n-k_1} \overset{\text{iid}}{\sim} \mathrm{Bern}(1/2)$ is shared between encoder and decoder
- $M^{k_1-k_2}$ is the message
- $U^n \overset{\text{iid}}{\sim} \mathrm{Bern}(1/2) \quad \Longrightarrow \quad d_{\mathrm{TV}}\left(q_{X^n}, \prod \mathrm{Bern}(\alpha)\right) \le \delta$
- $H_1 X^n = V_1^{n-k_1}$

P2P Code + Slepian–Wolf Code $\longrightarrow$ Asymmetric Channel Code

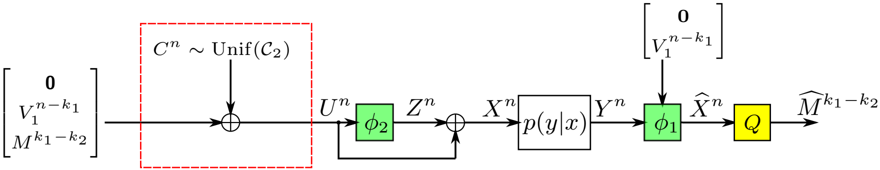
- Goal: Code for asymmetric channel $p(y\,|\,x)$
- Approach: Target $p(x) \sim \mathrm{Bern}(\alpha)$ for some given α.
- Lego bricks:
  1. (k_2, n) P2P code (H_2, ϕ_2) for $\mathrm{BSC}(\alpha)$ with decoding distance δ
  2. $(n-k_1, n)$ Slepian–Wolf code (H_1, ϕ_1) for $p(x, y)$ with error probability ϵ
  3. Two codes are nested s.t. $H_2 = \begin{bmatrix} H_1 \\ Q \end{bmatrix}$.



- $V_1^{n-k_1} \overset{\text{iid}}{\sim} \mathrm{Bern}(1/2)$ is shared between encoder and decoder
- $M^{k_1-k_2}$ is the message
- $U^n \overset{\text{iid}}{\sim} \mathrm{Bern}(1/2) \quad \Longrightarrow \quad d_{\mathrm{TV}}\left(q_{X^n}, \prod \mathrm{Bern}(\alpha)\right) \le \delta$
- $H_1 X^n = V_1^{n-k_1}$

P2P Code + Slepian–Wolf Code $\longrightarrow$ Asymmetric Channel Code

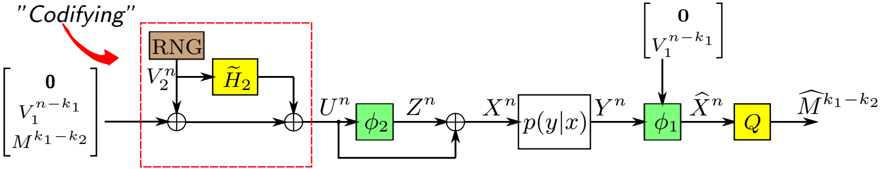
- Goal: Code for asymmetric channel $p(y|x)$
- Approach: Target $p(x) \sim \mathrm{Bern}(\alpha)$ for some given α.
- Lego bricks:
  1. (k_2, n) P2P code (H_2, ϕ_2) for $\mathrm{BSC}(\alpha)$ with decoding distance δ
  2. $(n-k_1, n)$ Slepian–Wolf code (H_1, ϕ_1) for $p(x,y)$ with error probability ϵ
  3. Two codes are nested s.t. $H_2 = \begin{bmatrix} H_1 \\ Q \end{bmatrix}$.



- $V_1^{n-k_1} \overset{\text{iid}}{\sim} \mathrm{Bern}(1/2)$ is shared between encoder and decoder
- $M^{k_1-k_2}$ is the message
- $U^n \overset{\text{iid}}{\sim} \mathrm{Bern}(1/2) \quad \Longrightarrow \quad d_{\mathrm{TV}}\left(q_{X^n}, \prod \mathrm{Bern}(\alpha)\right) \leq \delta$
- $H_1 X^n = V_1^{n-k_1}$

P2P Code + Slepian–Wolf Code $\longrightarrow$ Asymmetric Channel Code

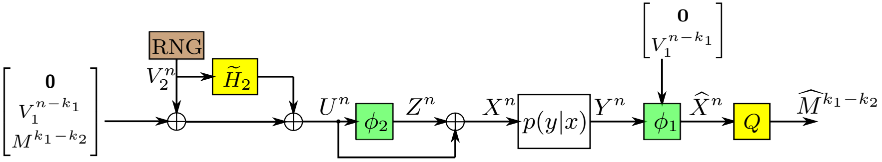
- Goal: Code for asymmetric channel $p(y|x)$
- Approach: Target $p(x) \sim \mathrm{Bern}(\alpha)$ for some given α.
- Lego bricks:
  1. (k_2, n) P2P code (H_2, ϕ_2) for $\mathrm{BSC}(\alpha)$ with decoding distance δ
  2. $(n - k_1, n)$ Slepian–Wolf code (H_1, ϕ_1) for $p(x, y)$ with error probability ϵ
  3. Two codes are nested s.t. $H_2 = \begin{bmatrix} H_1 \\ Q \end{bmatrix}$.



- $V_1^{n-k_1} \overset{\text{iid}}{\sim} \mathrm{Bern}(1/2)$ is shared between encoder and decoder
- $M^{k_1-k_2}$ is the message
- $U^n \overset{\text{iid}}{\sim} \mathrm{Bern}(1/2) \quad \Longrightarrow \quad d_{\mathrm{TV}}\left(q_{X^n}, \prod \mathrm{Bern}(\alpha)\right) \le \delta$
- $H_1 X^n = V_1^{n-k_1}$

P2P Code + Slepian–Wolf Code $\longrightarrow$ Asymmetric Channel Code

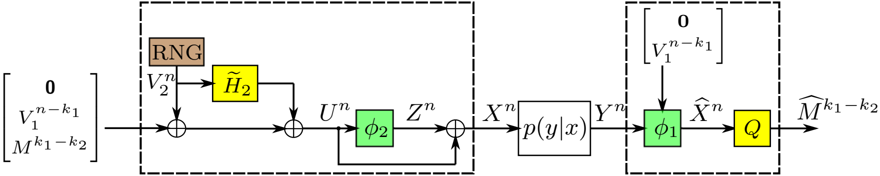
- Goal: Code for asymmetric channel $p(y|x)$
- Approach: Target $p(x) \sim \mathrm{Bern}(\alpha)$ for some given α.
- Lego bricks:
  1. (k_2, n) P2P code (H_2, ϕ_2) for $\mathrm{BSC}(\alpha)$ with decoding distance δ
  2. $(n - k_1, n)$ Slepian–Wolf code (H_1, ϕ_1) for $p(x, y)$ with error probability ϵ
  3. Two codes are nested s.t. $H_2 = \begin{bmatrix} H_1 \\ Q \end{bmatrix}$.



- $V_1^{n-k_1} \overset{\text{iid}}{\sim} \mathrm{Bern}(1/2)$ is shared between encoder and decoder
- $M^{k_1-k_2}$ is the message
- $U^n \overset{\text{iid}}{\sim} \mathrm{Bern}(1/2) \quad \Longrightarrow \quad d_{\mathrm{TV}}\left(q_{X^n}, \prod \mathrm{Bern}(\alpha)\right) \leq \delta$
- $H_1 X^n = V_1^{n-k_1}$

P2P Code + Slepian–Wolf Code $\longrightarrow$ Asymmetric Channel Code

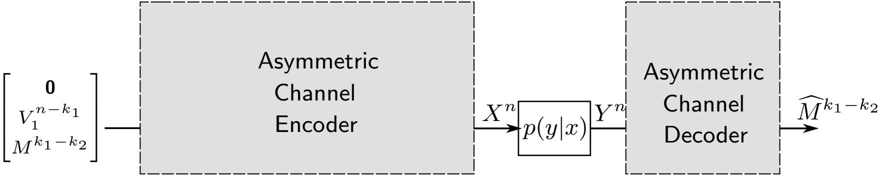
- Goal: Code for asymmetric channel $p(y|x)$
- Approach: Target $p(x) \sim \mathrm{Bern}(\alpha)$ for some given α.
- Lego bricks:
  1. (k_2, n) P2P code (H_2, ϕ_2) for $\mathrm{BSC}(\alpha)$ with decoding distance δ
  2. $(n-k_1, n)$ Slepian–Wolf code (H_1, ϕ_1) for $p(x, y)$ with error probability ϵ
  3. Two codes are nested s.t. $H_2 = \begin{bmatrix} H_1 \\ Q \end{bmatrix}$.



- $V_1^{n-k_1} \overset{\text{iid}}{\sim} \mathrm{Bern}(1/2)$ is shared between encoder and decoder
- $M^{k_1-k_2}$ is the message
- $U^n \overset{\text{iid}}{\sim} \mathrm{Bern}(1/2) \quad \Longrightarrow \quad d_{\mathrm{TV}}\left(q_{X^n}, \prod \mathrm{Bern}(\alpha)\right) \leq \delta$
- $H_1 X^n = V_1^{n-k_1}$

P2P Code + Slepian–Wolf Code $\longrightarrow$ Asymmetric Channel Code

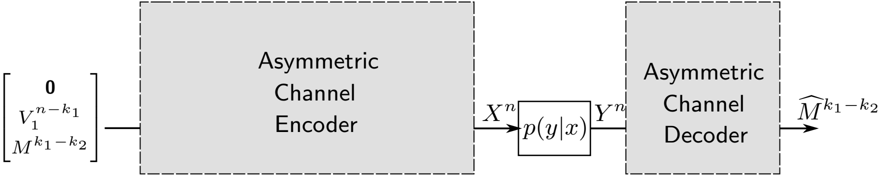
- Goal: Code for asymmetric channel $p(y\,|\,x)$
- Approach: Target $p(x) \sim \mathrm{Bern}(\alpha)$ for some given α.
- Lego bricks:
  1. (k_2, n) P2P code (H_2, ϕ_2) for $\mathrm{BSC}(\alpha)$ with decoding distance δ
  2. $(n - k_1, n)$ Slepian–Wolf code (H_1, ϕ_1) for $p(x, y)$ with error probability ϵ
  3. Two codes are nested s.t. $H_2 = \begin{bmatrix} H_1 \\ Q \end{bmatrix}$.



- $V_1^{n-k_1} \overset{\text{iid}}{\sim} \mathrm{Bern}(1/2)$ is shared between encoder and decoder
- $M^{k_1-k_2}$ is the message
- $U^n \overset{\text{iid}}{\sim} \mathrm{Bern}(1/2) \quad \Longrightarrow \quad d_{\mathrm{TV}}\left(q_{X^n}, \prod \mathrm{Bern}(\alpha)\right) \leq \delta$
- $H_1 X^n = V_1^{n-k_1}$

P2P Code + Slepian–Wolf Code $\longrightarrow$ Asymmetric Channel Code

- Goal: Code for asymmetric channel $p(y\,|\,x)$
- Approach: Target $p(x) \sim \mathrm{Bern}(\alpha)$ for some given α.
- Lego bricks:
  1. (k_2, n) P2P code (H_2, ϕ_2) for $\mathrm{BSC}(\alpha)$ with decoding distance δ
  2. $(n - k_1, n)$ Slepian–Wolf code (H_1, ϕ_1) for $p(x, y)$ with error probability ϵ
  3. Two codes are nested s.t. $H_2 = \begin{bmatrix} H_1 \\ Q \end{bmatrix}$.



- $V_1^{n-k_1} \overset{\text{iid}}{\sim} \mathrm{Bern}(1/2)$ is shared between encoder and decoder
- $M^{k_1-k_2}$ is the message
- $U^n \overset{\text{iid}}{\sim} \mathrm{Bern}(1/2) \quad \Longrightarrow \quad d_{\mathrm{TV}}\left(q_{X^n}, \prod \mathrm{Bern}(\alpha)\right) \le \delta$
- $H_1 X^n = V_1^{n-k_1}$

Theorem

$R = \frac{k_1 - k_2}{n}$

$P_e \le \epsilon + \delta$

Rate Achievability

- $\exists$ a sequence of Slepian–Wolf codes for $p(x, y)$ s.t. $\epsilon \to 0$ if and only if

$$\frac{n - k_1}{n} > H(X \mid Y)$$

Rate Achievability

- $\exists$ a sequence of Slepian–Wolf codes for $p(x, y)$ s.t. $\epsilon \to 0$ if and only if

$$\frac{n - k_1}{n} > H(X \,|\, Y)$$

- $\exists$ a sequence of P2P codes for $\mathrm{BSC}(\alpha)$ s.t. $\delta \to 0$ if and only if

$$\frac{k_2}{n} > 1 - H(\alpha) = 1 - H(X)$$

Rate Achievability

- $\exists$ a sequence of Slepian–Wolf codes for $p(x,y)$ s.t. $\epsilon \to 0$ if and only if

$$\frac{n-k_1}{n} > H(X\,|\,Y)$$

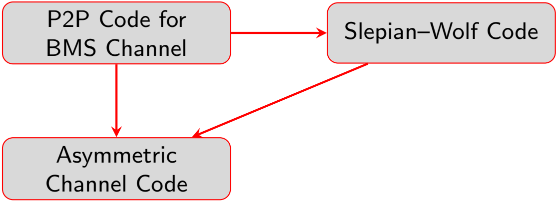
- $\exists$ a sequence of P2P codes for $\mathrm{BSC}(\alpha)$ s.t. $\delta \to 0$ if and only if

$$\frac{k_2}{n} > 1 - H(\alpha) = 1 - H(X)$$

- Rate $R = \frac{k_1 - k_2}{n}$ can be made arbitrarily close to $I(X;Y) = H(X) - H(X\,|\,Y)$.

Outline

- This talk:

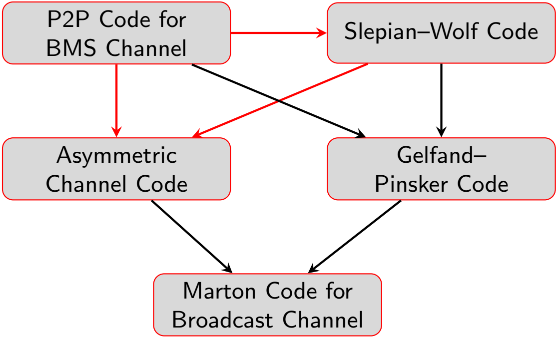


Properties

Nested Linearity
Error Probability
Decoding Distance

Outline

- This talk:

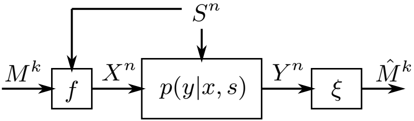


Properties

Nested Linearity
Error Probability
Decoding Distance

Gelfand–Pinsker Coding

- Channel with state S^n available noncausally only at the encoder



$S^n \overset{\text{iid}}{\sim} p(s)$

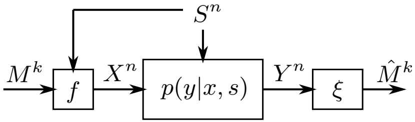
Encoder f, Decoder ξ

Rate $R = k/n$

$P_e = \mathsf{P}\{M^k \neq \hat{M}^k\}$

Gelfand–Pinsker Coding

- Channel with state S^n available noncausally only at the encoder



$S^n \overset{\text{iid}}{\sim} p(s)$

Encoder f, Decoder ξ

Rate $R = k/n$

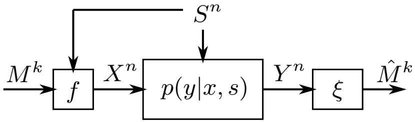
$P_e = \mathsf{P}\{M^k \neq \hat{M}^k\}$

- [Gelfand–Pinsker 1980]: $\exists$ a code (f, ξ) with vanishing P_e if

$$R < \max_{p(x|s)} \big(I(X;Y) - I(X;S) \big)$$

Gelfand–Pinsker Coding

- Channel with state S^n available noncausally only at the encoder



$S^n \overset{\text{iid}}{\sim} p(s)$

Encoder f, Decoder ξ

Rate $R = k/n$

$P_e = \mathsf{P}\{M^k \neq \hat{M}^k\}$

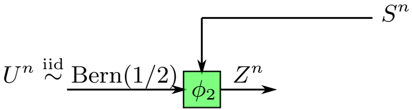
- [Gelfand–Pinsker 1980]: $\exists$ a code (f, ξ) with vanishing P_e if

$$R < \max_{p(x|s)} \big(I(X;Y) - I(X;S)\big)$$

- Approach: Target a conditional distribution $p(x\,|\,s)$
- This completely defines $p(x, s) = p(s)p(x\,|\,s)$

- Lego bricks:
  1. (k_2, n) P2P code (H_2, ϕ_2) with decoding distance δ for

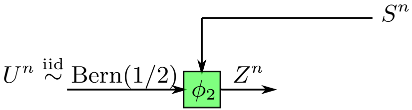
$$\bar{p}(s, v \mid x) \triangleq p_{X,S}(x \oplus v, s)$$



P2P Code + Slepian–Wolf Code $\longrightarrow$ Gelfand–Pinsker Code

- Lego bricks:
  1. (k_2, n) P2P code (H_2, ϕ_2) with decoding distance δ for
     $$\bar{p}(s, v \mid x) \triangleq p_{X,S}(x \oplus v, s)$$

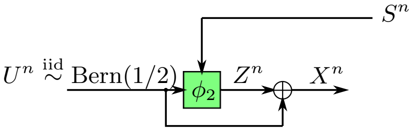
$\bar{p}$ is the *"symmetrized channel"* of $p(x, s)$



- Lego bricks:
  1. (k_2, n) P2P code (H_2, ϕ_2) with decoding distance δ for

$$\bar{p}(s, v \,|\, x) \triangleq p_{X,S}(x \oplus v, s)$$

$\bar{p}$ is the *"symmetrized channel"* of $p(x, s)$



$$X^n = \phi_2(S^n, U^n) \oplus U^n$$

Lemma

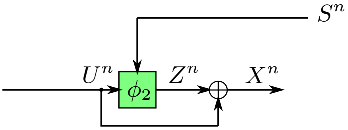
$$d_{\text{TV}}\left(q_{X^n,S^n}, \prod p(x, s)\right) \le \delta$$

P2P Code + Slepian–Wolf Code $\longrightarrow$ Gelfand–Pinsker Code

- Lego bricks:
  1. (k_2, n) P2P code (H_2, ϕ_2) with decoding distance δ for

$$\bar{p}(s, v \mid x) \triangleq p_{X,S}(x \oplus v, s)$$

$\bar{p}$ is the *"symmetrized channel"* of $p(x, s)$

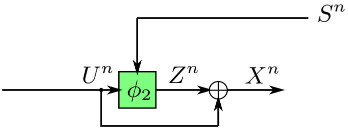


P2P Code + Slepian–Wolf Code $\longrightarrow$ Gelfand–Pinsker Code

- Lego bricks:
  1. (k_2, n) P2P code (H_2, ϕ_2) with decoding distance δ for
  $$\bar{p}(s, v \mid x) \triangleq p_{X,S}(x \oplus v, s)$$
  2. $(n - k_1, n)$ Slepian–Wolf code (H_1, ϕ_1) with error probability ϵ
  $$p(x, y) = \sum_s p(x, s) p(y \mid x, s)$$
  3. Two codes are nested s.t. $H_2 = \begin{bmatrix} H_1 \\ Q \end{bmatrix}$.

$\bar{p}$ is the *"symmetrized channel"* of $p(x, s)$

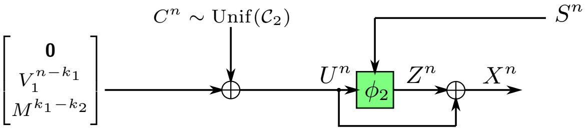
$$\begin{bmatrix} \mathbf{0} \\ V_1^{n-k_1} \\ M^{k_1-k_2} \end{bmatrix}$$



- $V_1^{n-k_1} \overset{\text{iid}}{\sim} \mathrm{Bern}(1/2)$ is shared between encoder and decoder
- $M^{k_1-k_2}$ is the message

- Lego bricks:
  1. (k_2, n) P2P code (H_2, ϕ_2) with decoding distance δ for
  $$\bar{p}(s, v \mid x) \triangleq p_{X,S}(x \oplus v, s)$$
  2. $(n - k_1, n)$ Slepian–Wolf code (H_1, ϕ_1) with error probability ϵ
  $$p(x, y) = \sum_s p(x, s) p(y \mid x, s)$$
  3. Two codes are nested s.t. $H_2 = \begin{bmatrix} H_1 \\ Q \end{bmatrix}$.

$\bar{p}$ is the *"symmetrized channel"* of $p(x, s)$



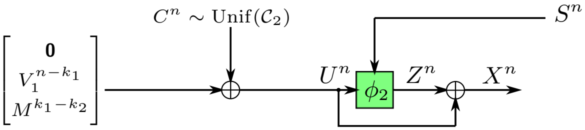
$U^n \overset{\text{iid}}{\sim} \mathrm{Bern}(1/2)$

- $V_1^{n-k_1} \overset{\text{iid}}{\sim} \mathrm{Bern}(1/2)$ is shared between encoder and decoder
- $M^{k_1-k_2}$ is the message
- $U^n \overset{\text{iid}}{\sim} \mathrm{Bern}(1/2) \quad \Longrightarrow \quad d_{\mathrm{TV}}\left(q_{X^n,S^n}, \prod p(x,s)\right) \le \delta$

P2P Code + Slepian–Wolf Code $\longrightarrow$ Gelfand–Pinsker Code

- Lego bricks:
  1. (k_2, n) P2P code (H_2, ϕ_2) with decoding distance δ for
  $$\bar{p}(s, v \mid x) \triangleq p_{X,S}(x \oplus v, s)$$
  2. $(n - k_1, n)$ Slepian–Wolf code (H_1, ϕ_1) with error probability ϵ
  $$p(x, y) = \sum_s p(x, s) p(y \mid x, s)$$
  3. Two codes are nested s.t. $H_2 = \begin{bmatrix} H_1 \\ Q \end{bmatrix}$.

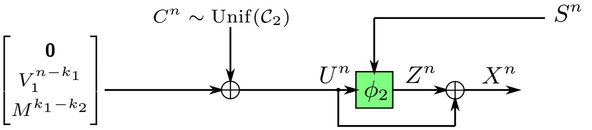
$\bar{p}$ is the *"symmetrized channel"* of $p(x, s)$



- $V_1^{n-k_1} \overset{\text{iid}}{\sim} \mathrm{Bern}(1/2)$ is shared between encoder and decoder
- $M^{k_1-k_2}$ is the message
- $U^n \overset{\text{iid}}{\sim} \mathrm{Bern}(1/2) \quad \Longrightarrow \quad d_{\mathrm{TV}}\left(q_{X^n,S^n}, \prod p(x, s)\right) \leq \delta$

- Lego bricks:
  1. (k_2, n) P2P code (H_2, ϕ_2) with decoding distance δ for
     $$\bar{p}(s, v \mid x) \triangleq p_{X,S}(x \oplus v, s)$$
  2. $(n - k_1, n)$ Slepian–Wolf code (H_1, ϕ_1) with error probability ϵ
     $$p(x, y) = \sum_s p(x, s) p(y \mid x, s)$$
  3. Two codes are nested s.t. $H_2 = \begin{bmatrix} H_1 \\ Q \end{bmatrix}$.

$\bar{p}$ is the *"symmetrized channel"* of $p(x, s)$



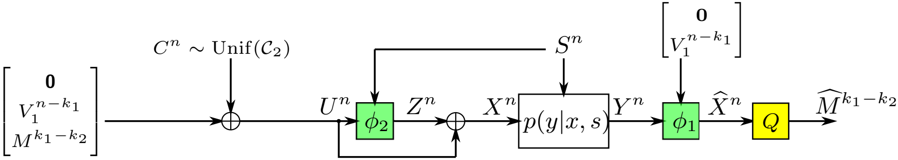
$H_1 X^n = V_1^{n-k_1}$

- $V_1^{n-k_1} \overset{\text{iid}}{\sim} \mathrm{Bern}(1/2)$ is shared between encoder and decoder
- $M^{k_1-k_2}$ is the message
- $U^n \overset{\text{iid}}{\sim} \mathrm{Bern}(1/2) \quad \Longrightarrow \quad d_{\mathrm{TV}}\left(q_{X^n,S^n}, \prod p(x,s)\right) \le \delta$
- $H_1 X^n = V_1^{n-k_1}$

P2P Code + Slepian–Wolf Code $\longrightarrow$ Gelfand–Pinsker Code

- Lego bricks:
  1. (k_2, n) P2P code (H_2, ϕ_2) with decoding distance δ for
  $$\bar{p}(s, v \,|\, x) \triangleq p_{X,S}(x \oplus v, s)$$
  2. $(n - k_1, n)$ Slepian–Wolf code (H_1, ϕ_1) with error probability ϵ
  $$p(x, y) = \sum_s p(x, s) p(y \,|\, x, s)$$
  3. Two codes are nested s.t. $H_2 = \begin{bmatrix} H_1 \\ Q \end{bmatrix}$.

$\bar{p}$ is the *"symmetrized channel"* of $p(x, s)$

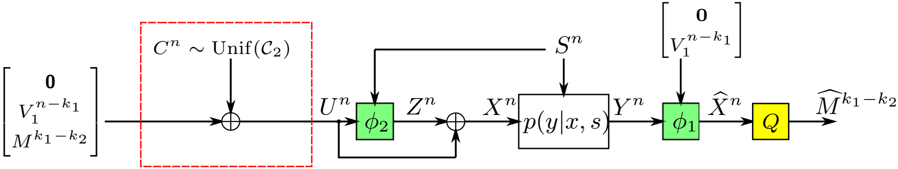


- $V_1^{n-k_1} \overset{\text{iid}}{\sim} \mathrm{Bern}(1/2)$ is shared between encoder and decoder
- $M^{k_1-k_2}$ is the message
- $U^n \overset{\text{iid}}{\sim} \mathrm{Bern}(1/2) \quad \Longrightarrow \quad d_{\mathrm{TV}}\left(q_{X^n,S^n}, \prod p(x, s)\right) \leq \delta$
- $H_1 X^n = V_1^{n-k_1}$

P2P Code + Slepian–Wolf Code $\longrightarrow$ Gelfand–Pinsker Code

- Lego bricks:
  1. (k_2, n) P2P code (H_2, ϕ_2) with decoding distance δ for
  $$\bar{p}(s, v \mid x) \triangleq p_{X,S}(x \oplus v, s)$$
  2. $(n - k_1, n)$ Slepian–Wolf code (H_1, ϕ_1) with error probability ϵ
  $$p(x, y) = \sum_s p(x, s) p(y \mid x, s)$$
  3. Two codes are nested s.t. $H_2 = \begin{bmatrix} H_1 \\ Q \end{bmatrix}$.

$\bar{p}$ is the *"symmetrized channel"* of $p(x, s)$



- $V_1^{n-k_1} \overset{\text{iid}}{\sim} \mathrm{Bern}(1/2)$ is shared between encoder and decoder
- $M^{k_1-k_2}$ is the message
- $U^n \overset{\text{iid}}{\sim} \mathrm{Bern}(1/2) \quad \Longrightarrow \quad d_{\mathrm{TV}}\left(q_{X^n,S^n}, \prod p(x, s)\right) \le \delta$
- $H_1 X^n = V_1^{n-k_1}$

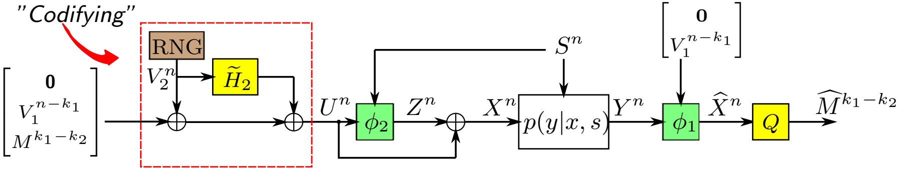
P2P Code + Slepian–Wolf Code $\longrightarrow$ Gelfand–Pinsker Code

- Lego bricks:
  1. (k_2, n) P2P code (H_2, ϕ_2) with decoding distance δ for

     $$\bar{p}(s, v \mid x) \triangleq p_{X,S}(x \oplus v, s)$$

     $\bar{p}$ is the *"symmetrized channel"* of $p(x, s)$
  2. $(n - k_1, n)$ Slepian–Wolf code (H_1, ϕ_1) with error probability ϵ

     $$p(x, y) = \sum_s p(x, s) p(y \mid x, s)$$
  3. Two codes are nested s.t. $H_2 = \begin{bmatrix} H_1 \\ Q \end{bmatrix}$.

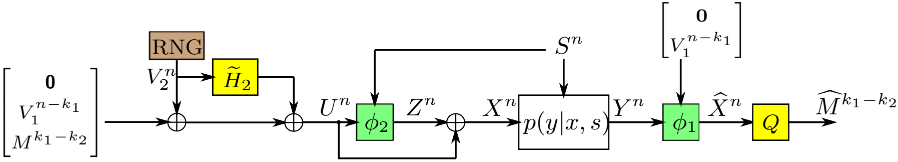


- $V_1^{n-k_1} \overset{\text{iid}}{\sim} \text{Bern}(1/2)$ is shared between encoder and decoder
- $M^{k_1-k_2}$ is the message
- $U^n \overset{\text{iid}}{\sim} \text{Bern}(1/2) \quad \Longrightarrow \quad d_{\text{TV}}\left(q_{X^n,S^n}, \prod p(x,s)\right) \leq \delta$
- $H_1 X^n = V_1^{n-k_1}$

P2P Code + Slepian–Wolf Code $\longrightarrow$ Gelfand–Pinsker Code

- Lego bricks:
  1. (k_2, n) P2P code (H_2, ϕ_2) with decoding distance δ for
  $$\bar{p}(s, v \mid x) \triangleq p_{X,S}(x \oplus v, s)$$
  2. $(n - k_1, n)$ Slepian–Wolf code (H_1, ϕ_1) with error probability ϵ
  $$p(x, y) = \sum_s p(x, s) p(y \mid x, s)$$
  3. Two codes are nested s.t. $H_2 = \begin{bmatrix} H_1 \\ Q \end{bmatrix}$.

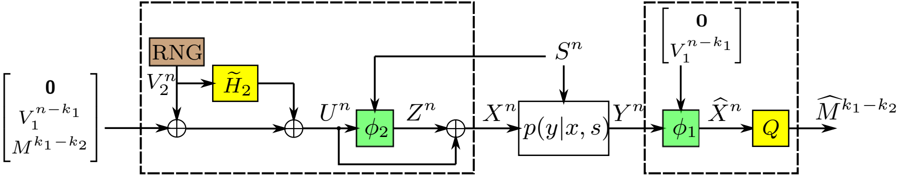
$\bar{p}$ is the *"symmetrized channel"* of $p(x, s)$



- $V_1^{n-k_1} \overset{\text{iid}}{\sim} \text{Bern}(1/2)$ is shared between encoder and decoder
- $M^{k_1-k_2}$ is the message
- $U^n \overset{\text{iid}}{\sim} \text{Bern}(1/2) \quad \Longrightarrow \quad d_{\text{TV}}\left(q_{X^n,S^n}, \prod p(x, s)\right) \leq \delta$
- $H_1 X^n = V_1^{n-k_1}$

P2P Code + Slepian–Wolf Code $\longrightarrow$ Gelfand–Pinsker Code

- Lego bricks:
  1. (k_2, n) P2P code (H_2, ϕ_2) with decoding distance δ for
  $$\bar{p}(s, v \mid x) \triangleq p_{X,S}(x \oplus v, s)$$
  $\bar{p}$ is the *"symmetrized channel"* of $p(x, s)$
  2. $(n - k_1, n)$ Slepian–Wolf code (H_1, ϕ_1) with error probability ϵ
  $$p(x, y) = \sum_s p(x, s) p(y \mid x, s)$$
  3. Two codes are nested s.t. $H_2 = \begin{bmatrix} H_1 \\ Q \end{bmatrix}$.



- $V_1^{n-k_1} \overset{\text{iid}}{\sim} \text{Bern}(1/2)$ is shared between encoder and decoder
- $M^{k_1-k_2}$ is the message
- $U^n \overset{\text{iid}}{\sim} \text{Bern}(1/2) \quad \Longrightarrow \quad d_{\text{TV}}\left(q_{X^n,S^n}, \prod p(x, s)\right) \leq \delta$
- $H_1 X^n = V_1^{n-k_1}$

P2P Code + Slepian–Wolf Code $\longrightarrow$ Gelfand–Pinsker Code

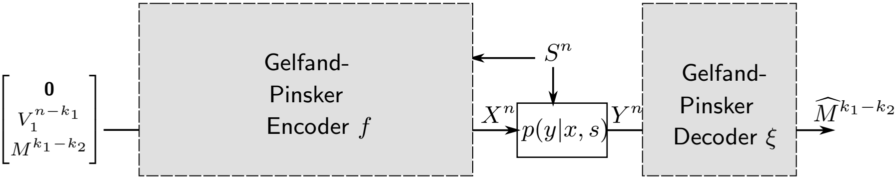
- Lego bricks:
  1. (k_2, n) P2P code (H_2, ϕ_2) with decoding distance δ for

     $$\bar{p}(s, v \mid x) \triangleq p_{X,S}(x \oplus v, s)$$

     $\bar{p}$ is the *"symmetrized channel"* of $p(x, s)$
  2. $(n - k_1, n)$ Slepian–Wolf code (H_1, ϕ_1) with error probability ϵ

     $$p(x, y) = \sum_s p(x, s) p(y \mid x, s)$$

  3. Two codes are nested s.t. $H_2 = \begin{bmatrix} H_1 \\ Q \end{bmatrix}$.

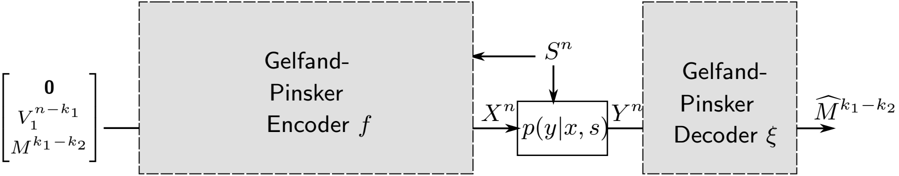


- $V_1^{n-k_1} \overset{\text{iid}}{\sim} \text{Bern}(1/2)$ is shared between encoder and decoder
- $M^{k_1-k_2}$ is the message
- $U^n \overset{\text{iid}}{\sim} \text{Bern}(1/2) \quad \Longrightarrow \quad d_{\text{TV}}\left(q_{X^n,S^n}, \prod p(x,s)\right) \leq \delta$
- $H_1 X^n = V_1^{n-k_1}$

P2P Code + Slepian–Wolf Code $\longrightarrow$ Gelfand–Pinsker Code

- Lego bricks:
  1. (k_2, n) P2P code (H_2, ϕ_2) with decoding distance δ for
  $$\bar{p}(s, v \mid x) \triangleq p_{X,S}(x \oplus v, s)$$
  2. $(n - k_1, n)$ Slepian–Wolf code (H_1, ϕ_1) with error probability ϵ
  $$p(x, y) = \sum_s p(x, s) p(y \mid x, s)$$
  3. Two codes are nested s.t. $H_2 = \begin{bmatrix} H_1 \\ Q \end{bmatrix}$.

$\bar{p}$ is the *"symmetrized channel"* of $p(x, s)$



- $V_1^{n-k_1} \overset{\text{iid}}{\sim} \mathrm{Bern}(1/2)$ is shared between encoder and decoder
- $M^{k_1-k_2}$ is the message
- $U^n \overset{\text{iid}}{\sim} \mathrm{Bern}(1/2) \quad \Longrightarrow \quad d_{\mathrm{TV}}\left(q_{X^n, S^n}, \prod p(x, s)\right) \le \delta$
- $H_1 X^n = V_1^{n-k_1}$

Theorem

$R = \frac{k_1 - k_2}{n}$
$P_e \le \epsilon + \delta$

Rate Achievability

- $\exists$ a sequence of Slepian–Wolf codes for $p(x, y)$ s.t. $\epsilon \to 0$ if and only if

$$\frac{n - k_1}{n} > H(X \mid Y)$$

Rate Achievability

- $\exists$ a sequence of Slepian–Wolf codes for $p(x, y)$ s.t. $\epsilon \to 0$ if and only if

$$\frac{n - k_1}{n} > H(X \,|\, Y)$$

- $\exists$ a sequence of P2P codes for the symmetrized channel $\bar{p}$ s.t. $\delta \to 0$ if and only if

$$\frac{k_2}{n} > 1 - H(X \,|\, S)$$

Rate Achievability

- $\exists$ a sequence of Slepian–Wolf codes for $p(x, y)$ s.t. $\epsilon \to 0$ if and only if

$$\frac{n - k_1}{n} > H(X \mid Y)$$

- $\exists$ a sequence of P2P codes for the symmetrized channel $\bar{p}$ s.t. $\delta \to 0$ if and only if

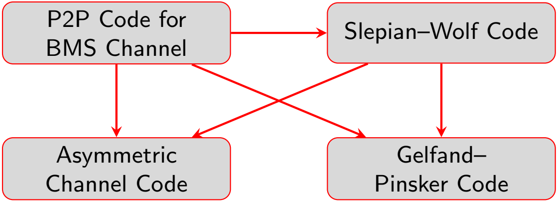
$$\frac{k_2}{n} > 1 - H(X \mid S)$$

- Rate $R = \frac{k_1 - k_2}{n}$ can be made arbitrarily close to

$$H(X \mid S) - H(X \mid Y) = I(X; Y) - I(X; S)$$

Outline

- This talk:

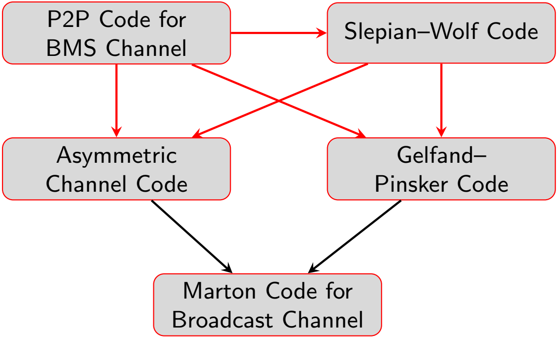


Properties

Nested Linearity
Error Probability
Decoding Distance

Outline

- This talk:

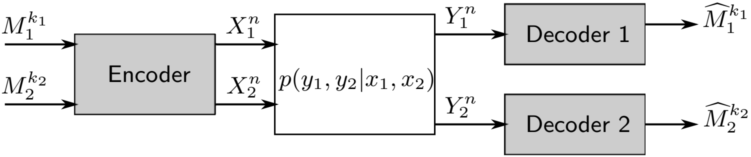


Properties

Nested Linearity
Error Probability
Decoding Distance

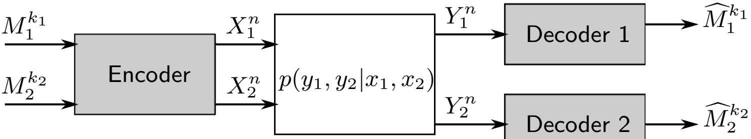
Marton Coding for Broadcast Channels

- Goal: Code for 2-user broadcast channel $p(y_1, y_2 \,|\, x_1, x_2)$ with two transmit antennas



Marton Coding for Broadcast Channels

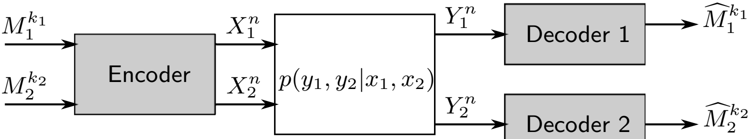
- Goal: Code for 2-user broadcast channel $p(y_1, y_2 \,|\, x_1, x_2)$ with two transmit antennas



$R_1 = k_1/n$
$R_2 = k_2/n$
$P_e = \mathsf{P}\{M_1^{k_1} \neq \hat{M}_1^{k_1} \cup M_2^{k_2} \neq \hat{M}_2^{k_2}\}$

Marton Coding for Broadcast Channels

- Goal: Code for 2-user broadcast channel $p(y_1, y_2 \,|\, x_1, x_2)$ with two transmit antennas



$$R_1 = k_1/n$$
$$R_2 = k_2/n$$
$$P_e = \mathsf{P}\{M_1^{k_1} \neq \hat{M}_1^{k_1} \cup M_2^{k_2} \neq \hat{M}_2^{k_2}\}$$

- [Marton 1979]: $\exists$ encoder and decoders with vanishing P_e for rates (R_1, R_2) if

$$\begin{aligned} R_1 &< I(X_1; Y_1) \\ R_2 &< I(X_2; Y_2) \\ R_1 + R_2 &< I(X_1; Y_1) + I(X_2; Y_2) - I(X_1; X_2) \end{aligned}$$

for some input distribution $p(x_1, x_2)$

Marton Coding for Broadcast Channels

- Approach: Target a channel input distribution $p(x_1, x_2)$.
- This completely defines

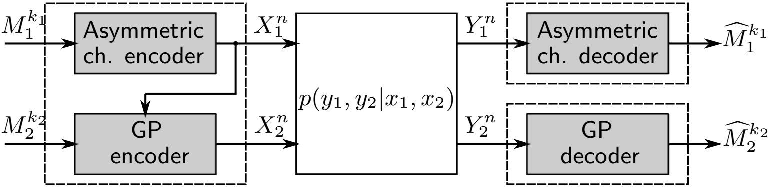
$$p(x_1, x_2, y_1, y_2) = p(x_1, x_2)p(y_1, y_2 \mid x_1, x_2)$$

Marton Coding for Broadcast Channels

- Approach: Target a channel input distribution $p(x_1, x_2)$.
- This completely defines

$$p(x_1, x_2, y_1, y_2) = p(x_1, x_2)p(y_1, y_2 \mid x_1, x_2)$$

- Lego bricks:
  - (R_1, n) asymmetric channel code for channel $X_1 \to Y_1$ with error probability ϵ_1
  - (R_2, n) Gelfand–Pinsker code for $X_2 \to Y_2$ with available "state" X_1 at the encoder with error probability ϵ_2

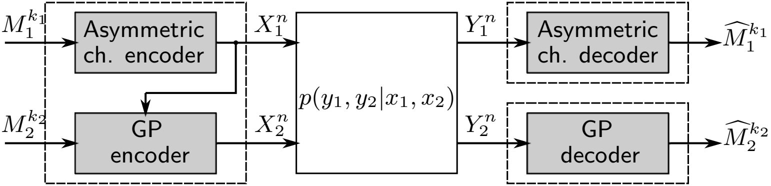


Marton Coding for Broadcast Channels

- Approach: Target a channel input distribution $p(x_1, x_2)$.
- This completely defines

$$p(x_1, x_2, y_1, y_2) = p(x_1, x_2)p(y_1, y_2 \mid x_1, x_2)$$

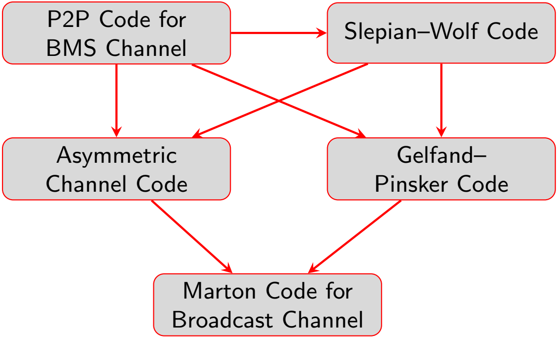
- Lego bricks:
  - (R_1, n) asymmetric channel code for channel $X_1 \to Y_1$ with error probability ϵ_1
  - (R_2, n) Gelfand–Pinsker code for $X_2 \to Y_2$ with available "state" X_1 at the encoder with error probability ϵ_2



- Error probability $P_e \leq \epsilon_1 + \epsilon_2$
- Can achieve a corner point in Marton's rate region

Outline

- This talk:



Properties

Nested Linearity
Error Probability
Decoding Distance

- Marton code can be implemented using **four** P2P codes for BMS channels

Simulation Results: Marton Coding

- Two-user broadcast channel

$$\begin{bmatrix} Y_1 \\ Y_2 \end{bmatrix} = H_{\text{ch}} W \begin{bmatrix} X_1 \\ X_2 \end{bmatrix} + \begin{bmatrix} Z_1 \\ Z_2 \end{bmatrix},$$

  where:
  - $(X_1, X_2) \in \{\pm 1\}^2$
  - $H_{\text{ch}} = \begin{bmatrix} 1 & g \\ g & 1 \end{bmatrix}$ is the channel gain matrix
  - W is a 2×2 precoding matrix used by the transmitter (if needed)
  - $Z_1, Z_2 \sim \mathcal{N}(0, 1)$ are independent
  - Encoder is subject to a sum-power constraint P such that

$$\mathsf{E}[\|W X_1^2\|^2] \leq P$$

- We use $g = 0.9$.
- Marton Coding: target a channel input distribution $p(x_1, x_2)$

Coding Strategies Over Broadcast Channel

- Maximum achievable sum-rate:

$$R_{\mathrm{sum,max}}\big(p(x_1, x_2), W\big) \triangleq I(X_1; Y_1) + I(X_2; Y_2) - I(X_1; X_2)$$

Coding Strategies Over Broadcast Channel

- Maximum achievable sum-rate:

$$R_{\mathrm{sum,max}}\big(p(x_1,x_2),W\big) \triangleq I(X_1;Y_1)+I(X_2;Y_2)-I(X_1;X_2)$$

- Coding strategies:
  1. Marton coding with optimal precoding
  2. Marton coding without precoding
  3. Symmetric coding with optimal precoding
  4. Symmetric coding with MMSE precoding
  5. Time division

Coding Strategies Over Broadcast Channel

- Maximum achievable sum-rate:

$$R_{\text{sum,max}}\big(p(x_1,x_2),W\big) \triangleq I(X_1;Y_1)+I(X_2;Y_2)-I(X_1;X_2)$$

- Coding strategies:
  1. Marton coding with optimal precoding
     $\Rightarrow$ target $p(x_1,x_2)$ and W that maximize $R_{\text{sum,max}}$
  2. Marton coding without precoding
  3. Symmetric coding with optimal precoding
  4. Symmetric coding with MMSE precoding
  5. Time division

Coding Strategies Over Broadcast Channel

- Maximum achievable sum-rate:

$$R_{\text{sum,max}}\big(p(x_1,x_2),W\big) \triangleq I(X_1;Y_1)+I(X_2;Y_2)-I(X_1;X_2)$$

- Coding strategies:
  1. Marton coding with optimal precoding
     $\Rightarrow$ target $p(x_1,x_2)$ and W that maximize $R_{\text{sum,max}}$
  2. Marton coding without precoding
     $\Rightarrow$ set $W=\sqrt{\frac{P}{2}}\mathbf{I}$ and target $p(x_1,x_2)$ that maximizes $R_{\text{sum,max}}$
  3. Symmetric coding with optimal precoding
  4. Symmetric coding with MMSE precoding
  5. Time division

Coding Strategies Over Broadcast Channel

- Maximum achievable sum-rate:

$$R_{\mathrm{sum,max}}\big(p(x_1,x_2),W\big) \triangleq I(X_1;Y_1) + I(X_2;Y_2) - I(X_1;X_2)$$

- Coding strategies:
  1. Marton coding with optimal precoding
     $\Rightarrow$ target $p(x_1,x_2)$ and W that maximize $R_{\mathrm{sum,max}}$
  2. Marton coding without precoding
     $\Rightarrow$ set $W = \sqrt{\frac{P}{2}}\mathbf{I}$ and target $p(x_1,x_2)$ that maximizes $R_{\mathrm{sum,max}}$
  3. Symmetric coding with optimal precoding
     $\Rightarrow$ Set $p(x_1,x_2) = 1/4$ for each (x_1,x_2), and target W that maximizes $R_{\mathrm{sum,max}}$
  4. Symmetric coding with MMSE precoding

  5. Time division

Coding Strategies Over Broadcast Channel

- Maximum achievable sum-rate:

$$R_{\text{sum,max}}\big(p(x_1, x_2), W\big) \triangleq I(X_1; Y_1) + I(X_2; Y_2) - I(X_1; X_2)$$

- Coding strategies:
  1. Marton coding with optimal precoding
     $\Rightarrow$ target $p(x_1, x_2)$ and W that maximize $R_{\text{sum,max}}$
  2. Marton coding without precoding
     $\Rightarrow$ set $W = \sqrt{\frac{P}{2}}\mathbf{I}$ and target $p(x_1, x_2)$ that maximizes $R_{\text{sum,max}}$
  3. Symmetric coding with optimal precoding
     $\Rightarrow$ Set $p(x_1, x_2) = 1/4$ for each (x_1, x_2), and target W that maximizes $R_{\text{sum,max}}$
  4. Symmetric coding with MMSE precoding
     $\Rightarrow$ Set $p(x_1, x_2) = 1/4$ for each (x_1, x_2), and $W = (H_{\text{ch}}^T H_{\text{ch}} + \frac{2}{P}\mathbf{I})^{-1} H_{\text{ch}}^T$
  5. Time division

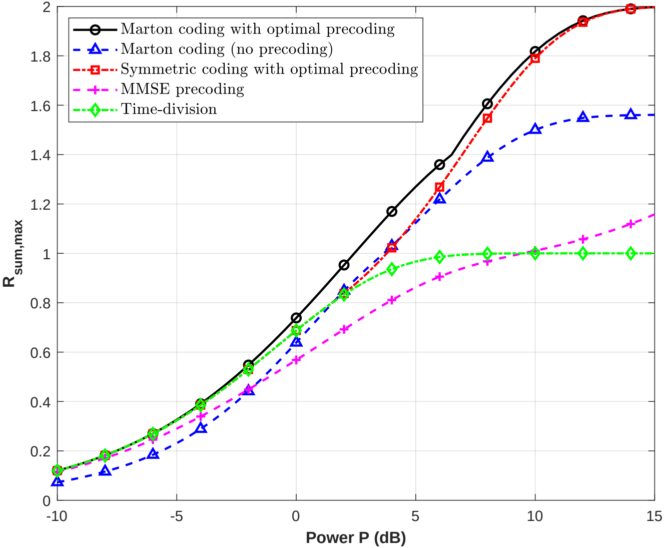
Coding Strategies Over Broadcast Channel

- Maximum achievable sum-rate:

$$R_{\text{sum,max}}\big(p(x_1,x_2),W\big) \triangleq I(X_1;Y_1)+I(X_2;Y_2)-I(X_1;X_2)$$

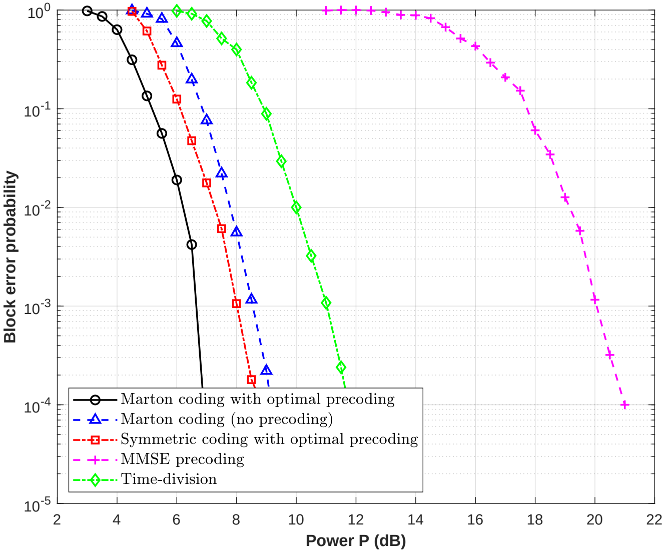
- Coding strategies:
  1. Marton coding with optimal precoding
     $\Rightarrow$ target $p(x_1,x_2)$ and W that maximize $R_{\text{sum,max}}$
  2. Marton coding without precoding
     $\Rightarrow$ set $W=\sqrt{\frac{P}{2}}\mathbf{I}$ and target $p(x_1,x_2)$ that maximizes $R_{\text{sum,max}}$
  3. Symmetric coding with optimal precoding
     $\Rightarrow$ Set $p(x_1,x_2)=1/4$ for each (x_1,x_2), and target W that maximizes $R_{\text{sum,max}}$
  4. Symmetric coding with MMSE precoding
     $\Rightarrow$ Set $p(x_1,x_2)=1/4$ for each (x_1,x_2), and $W=(H_{\text{ch}}^T H_{\text{ch}}+\frac{2}{P}\mathbf{I})^{-1}H_{\text{ch}}^T$
  5. Time division
     $\Rightarrow$ communicate to only one user in the channel

Maximum Achievable Sum-Rates

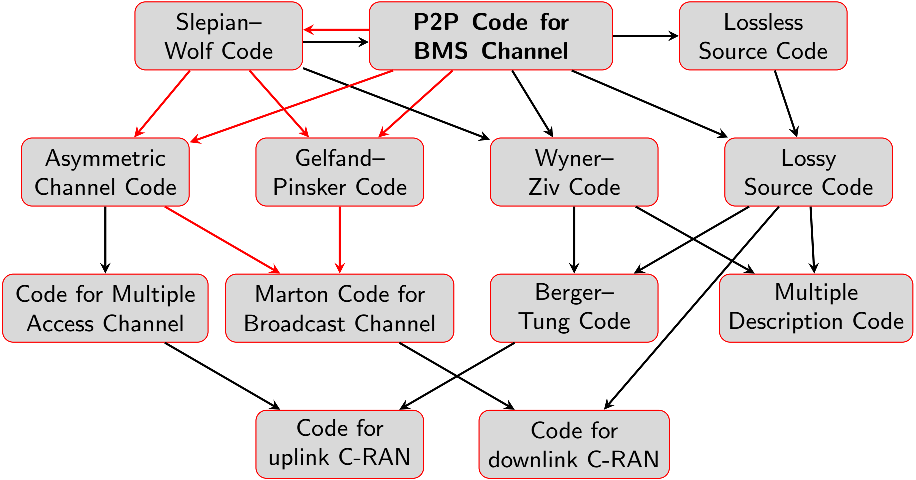


Simulations, $n = 1024$, $R_{\text{sum}} = 1$

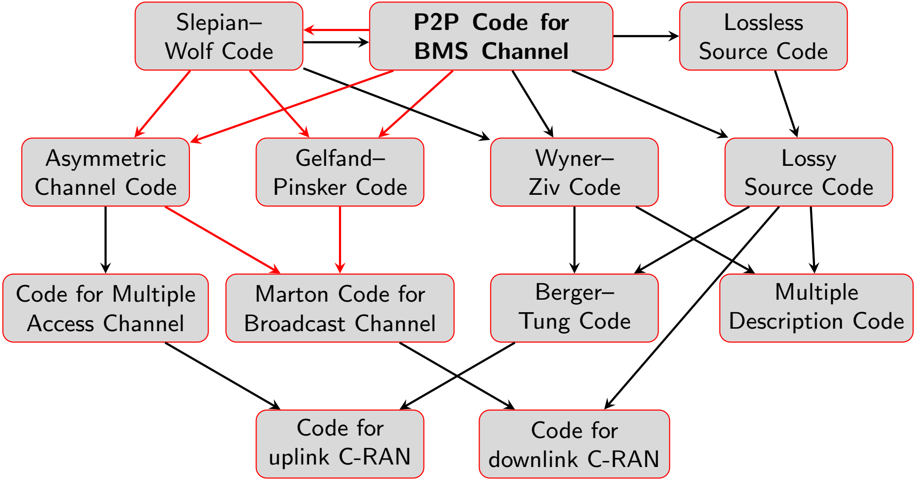
- Use polar codes with SC decoding (rates chosen "close" to theoretical limits)



Code Constructions



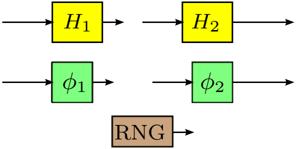
Code Constructions



- All coding schemes can be constructed starting from P2P codes for BMS channels.
- All constructions are rate-optimal if the constituent Lego bricks are rate-optimal.*

Concluding Remarks

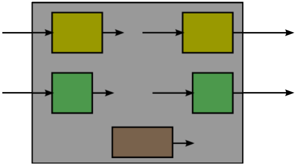
- Lego-brick code design:



Concluding Remarks

- Lego-brick code design:

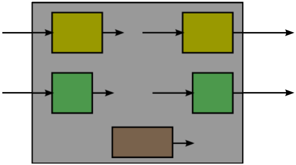
Code for a network



Concluding Remarks

- Lego-brick code design:

Code for a network



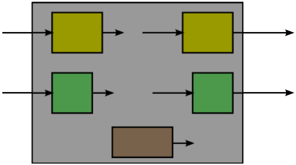
Properties

Nested Linearity
Error Probability
Decoding Distance

Concluding Remarks

- Lego-brick code design:

Code for a network



Properties

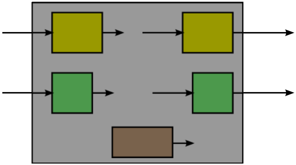
Nested Linearity
Error Probability
Decoding Distance

- Future directions
  1. Channels with non-binary inputs $\Longrightarrow$ coded modulation
  2. Networks with large (unknown) number of users: random access?
  3. Design of codes with a good decoding distance

Concluding Remarks

- Lego-brick code design:

Code for a network



Properties

Nested Linearity
Error Probability
Decoding Distance

- Future directions
  1. Channels with non-binary inputs $\Longrightarrow$ coded modulation
  2. Networks with large (unknown) number of users: random access?
  3. Design of codes with a good decoding distance

- ArXiv paper: https://arxiv.org/abs/2211.07208

- Simulation code: https://github.com/nadimgh/lego-brick

Acknowledgments

- Prof. Young-Han Kim

Acknowledgments

- Prof. Young-Han Kim
- Prof. Larry Milstein

Acknowledgments

- Prof. Young-Han Kim
- Prof. Larry Milstein
- Prof. Paul Siegel, Prof. Arya Mazumdar, Prof. Alireza Salehi Golsefidy

Acknowledgments

- Prof. Young-Han Kim
- Prof. Larry Milstein
- Prof. Paul Siegel, Prof. Arya Mazumdar, Prof. Alireza Salehi Golsefidy
- Prof. Alexander Vardy

Acknowledgments

- Prof. Young-Han Kim
- Prof. Larry Milstein
- Prof. Paul Siegel, Prof. Arya Mazumdar, Prof. Alireza Salehi Golsefidy
- Prof. Alexander Vardy
- Prof. Lele Wang

Acknowledgments

- Prof. Young-Han Kim
- Prof. Larry Milstein
- Prof. Paul Siegel, Prof. Arya Mazumdar, Prof. Alireza Salehi Golsefidy
- Prof. Alexander Vardy
- Prof. Lele Wang
- **Funding:** InterDigital (Dr. Liangping Ma), ETRI

Acknowledgments

- Prof. Young-Han Kim
- Prof. Larry Milstein
- Prof. Paul Siegel, Prof. Arya Mazumdar, Prof. Alireza Salehi Golsefidy
- Prof. Alexander Vardy
- Prof. Lele Wang
- **Funding:** InterDigital (Dr. Liangping Ma), ETRI
- **Internship:** Samsung (Dr. Hamid Saber, Dr. Jung Hyun Bae), Qualcomm (Dr. Ari Klein)

Acknowledgments

- Prof. Young-Han Kim
- Prof. Larry Milstein
- Prof. Paul Siegel, Prof. Arya Mazumdar, Prof. Alireza Salehi Golsefidy
- Prof. Alexander Vardy
- Prof. Lele Wang
- **Funding:** InterDigital (Dr. Liangping Ma), ETRI
- **Internship:** Samsung (Dr. Hamid Saber, Dr. Jung Hyun Bae), Qualcomm (Dr. Ari Klein)
- **Labmates:** Alankrita, Jiun-Ting, Jongha, Pinar, Shouvik

Acknowledgments

- Prof. Young-Han Kim
- Prof. Larry Milstein
- Prof. Paul Siegel, Prof. Arya Mazumdar, Prof. Alireza Salehi Golsefidy
- Prof. Alexander Vardy
- Prof. Lele Wang
- **Funding:** InterDigital (Dr. Liangping Ma), ETRI
- **Internship:** Samsung (Dr. Hamid Saber, Dr. Jung Hyun Bae), Qualcomm (Dr. Ari Klein)
- **Labmates:** Alankrita, Jiun-Ting, Jongha, Pinar, Shouvik
- **UCSD:** Rohan, Sukanya, Aditya, Anwesan, Shahar, Raghu, Yuhan, Muhammad, Mouna, Ahmed, Hanwen, Govind, Sheel, Pranav

Acknowledgments

- Prof. Young-Han Kim
- Prof. Larry Milstein
- Prof. Paul Siegel, Prof. Arya Mazumdar, Prof. Alireza Salehi Golsefidy
- Prof. Alexander Vardy
- Prof. Lele Wang
- **Funding:** InterDigital (Dr. Liangping Ma), ETRI
- **Internship:** Samsung (Dr. Hamid Saber, Dr. Jung Hyun Bae), Qualcomm (Dr. Ari Klein)
- **Labmates:** Alankrita, Jiun-Ting, Jongha, Pinar, Shouvik
- **UCSD:** Rohan, Sukanya, Aditya, Anwesan, Shahar, Raghu, Yuhan, Muhammad, Mouna, Ahmed, Hanwen, Govind, Sheel, Pranav
- **AUB:** Abed, Wajeb, Rami, Mounib, Razan, Kassir, Taha, and Natali

Acknowledgments

- Prof. Young-Han Kim
- Prof. Larry Milstein
- Prof. Paul Siegel, Prof. Arya Mazumdar, Prof. Alireza Salehi Golsefidy
- Prof. Alexander Vardy
- Prof. Lele Wang
- **Funding:** InterDigital (Dr. Liangping Ma), ETRI
- **Internship:** Samsung (Dr. Hamid Saber, Dr. Jung Hyun Bae), Qualcomm (Dr. Ari Klein)
- **Labmates:** Alankrita, Jiun-Ting, Jongha, Pinar, Shouvik
- **UCSD:** Rohan, Sukanya, Aditya, Anwesan, Shahar, Raghu, Yuhan, Muhammad, Mouna, Ahmed, Hanwen, Govind, Sheel, Pranav
- **AUB:** Abed, Wajeb, Rami, Mounib, Razan, Kassir, Taha, and Natali
- **EPFL:** Rakshita

Acknowledgments

- Prof. Young-Han Kim
- Prof. Larry Milstein
- Prof. Paul Siegel, Prof. Arya Mazumdar, Prof. Alireza Salehi Golsefidy
- Prof. Alexander Vardy
- Prof. Lele Wang
- **Funding:** InterDigital (Dr. Liangping Ma), ETRI
- **Internship:** Samsung (Dr. Hamid Saber, Dr. Jung Hyun Bae), Qualcomm (Dr. Ari Klein)
- **Labmates:** Alankrita, Jiun-Ting, Jongha, Pinar, Shouvik
- **UCSD:** Rohan, Sukanya, Aditya, Anwesan, Shahar, Raghu, Yuhan, Muhammad, Mouna, Ahmed, Hanwen, Govind, Sheel, Pranav
- **AUB:** Abed, Wajeb, Rami, Mounib, Razan, Kassir, Taha, and Natali
- **EPFL:** Rakshita
- **Cousins:** Manal, Ali, Leila

Acknowledgments

- Prof. Young-Han Kim
- Prof. Larry Milstein
- Prof. Paul Siegel, Prof. Arya Mazumdar, Prof. Alireza Salehi Golsefidy
- Prof. Alexander Vardy
- Prof. Lele Wang
- **Funding:** InterDigital (Dr. Liangping Ma), ETRI
- **Internship:** Samsung (Dr. Hamid Saber, Dr. Jung Hyun Bae), Qualcomm (Dr. Ari Klein)
- **Labmates:** Alankrita, Jiun-Ting, Jongha, Pinar, Shouvik
- **UCSD:** Rohan, Sukanya, Aditya, Anwesan, Shahar, Raghu, Yuhan, Muhammad, Mouna, Ahmed, Hanwen, Govind, Sheel, Pranav
- **AUB:** Abed, Wajeb, Rami, Mounib, Razan, Kassir, Taha, and Natali
- **EPFL:** Rakshita
- **Cousins:** Manal, Ali, Leila
- **My heart:** Baba, Mama, Fattouma, Hammoudi

The best