Networks with point-to-point codes

Abbas El Gamal

Stanford University

Shannon Memorial Lecture, UCSD 2013

Based on joint work with B. Bandemer, D. Tse, F. Bacelli, and Y.-H. Kim
“The fundamental problem of communication is that of reproducing at one point, either exactly or approximately, a message selected at another point.”

C.E. Shannon (1948)
Shannon’s point-to-point communication system

- **Discrete memoryless channel (DMC)** $p(y|x)$
- $(2^{nR}, n)$ block code:
  - **Message**: $M \sim \text{Unif}[1 : 2^{nR}]$
  - **Encoder**: $x^n(m), m \in [1 : 2^{nR}]$
  - **Decoder**: $\hat{m}(y^n) \in [1 : 2^{nR}] \cup \{e\}$
Shannon’s point-to-point communication system

- **Discrete memoryless channel (DMC)** $p(y|x)$
- $(2^{nR}, n)$ block code:
  - **Message**: $M \sim \text{Unif}[1 : 2^{nR}]$
  - **Encoder**: $x^n(m)$, $m \in [1 : 2^{nR}]$
  - **Decoder**: $\hat{m}(y^n) \in [1 : 2^{nR}] \cup \{e\}$
- **Probability of error** $P_e^{(n)} = P\{\hat{M} \neq M\}$
Shannon’s point-to-point communication system

- Discrete memoryless channel (DMC) $p(y|x)$
- $(2^{nR}, n)$ block code:
  - Message: $M \sim \text{Unif}[1 : 2^{nR}]$
  - Encoder: $x^n(m), m \in [1 : 2^{nR}]$
  - Decoder: $\hat{m}(y^n) \in [1 : 2^{nR}] \cup \{e\}$
- Probability of error $P_{e}^{(n)} = P\{\hat{M} \neq M\}$
- $R$ achievable if $\exists (2^{nR}, n)$ codes with $\lim_{n \to \infty} P_{e}^{(n)} = 0$
- Capacity $C$ is supremum of all achievable rates
Shannon’s channel coding theorem

Theorem (Shannon 1948)

\[ C = \max_{p(x)} I(X; Y) \]
Shannon’s channel coding theorem

\[ C = \max_{p(x)} I(X; Y) \]

- Shannon gave an existential proof of achievability via random coding.
Achievability using random point-to-point (ptp) codes

- **Codebook generation**: Fix \( p(x) \)
  - Randomly generate \( 2^{nR} \) sequences \( x^n(m) \sim \prod_{i=1}^n p_X(x_i) \), \( m \in [1 : 2^{nR}] \)
  - Random codebook: \( C_n(p) = \{X^n(m) : m \in [1 : 2^{nR}]\} \)
Achievability using random point-to-point (ptp) codes

- **Codebook generation:** Fix $p(x)$
  - Randomly generate $2^{nR}$ sequences $x^n(m) \sim \prod_{i=1}^{n} p_X(x_i)$, $m \in [1 : 2^{nR}]$
  - Random codebook: $C_n(p) = \{X^n(m) : m \in [1 : 2^{nR}]\}$

- **Encoding:**
  - To send $m$, the encoder transmits $x^n(m)$
Achievability using random point-to-point (ptp) codes

- **Codebook generation:** Fix $p(x)$
  - Randomly generate $2^{nR}$ sequences $x^n(m) \sim \prod_{i=1}^{n} p_X(x_i)$, $m \in [1 : 2^{nR}]$
  - Random codebook: $C_n(p) = \{X^n(m) : m \in [1 : 2^{nR}]\}$

- **Encoding:**
  - To send $m$, the encoder transmits $x^n(m)$

- **Decoding:**
  - Optimal decoding rule is maximum likelihood (MLD):
    $$\hat{m}(y^n) = \arg \max_{m \in [1:2^{nR}]} \prod_{i=1}^{n} p_{Y|X}(y_i | x_i(m))$$
Achievability using random point-to-point (ptp) codes

- **Codebook generation**: Fix $p(x)$
  - Randomly generate $2^{nR}$ sequences $x^n(m) \sim \prod_{i=1}^{n} p_X(x_i)$, $m \in [1 : 2^{nR}]$
  - Random codebook: $C_n(p) = \{X^n(m) : m \in [1 : 2^{nR}]\}$

- **Encoding**:
  - To send $m$, the encoder transmits $x^n(m)$

- **Decoding**:
  - Optimal decoding rule is maximum likelihood (MLD):
    \[
    \hat{m}(y^n) = \arg \max_{m \in [1:2^{nR}]} \prod_{i=1}^{n} p_{Y|X}(y_i|x_i(m))
    \]

- **Analysis of the probability of error**, e.g., Gallager (1965):
  - Show that $E[P_e^{(n)}(C_n(p))] \to 0$ if $R < I(X; Y)$; hence good codes exist
  - Tight error exponents, but difficult to extend analysis to networks
Shannon’s joint typicality proof

- **Decoding:**
  - Find *unique message* \( \hat{m} \) such that \( (x^n(\hat{m}), y^n) \in T^{(n)}_\epsilon \)
  - Otherwise declare an error
Shannon’s joint typicality proof

- **Decoding:**
  - Find unique message $\hat{m}$ such that $(x^n(\hat{m}), y^n) \in \mathcal{T}_\epsilon^{(n)}$
  - Otherwise declare an error

- Several ways to define typicality, e.g., Orlitsky–Roche (2001):
  
  $$\mathcal{T}_\epsilon^{(n)} = \{(x^n, y^n) : |\pi(x, y|x^n, y^n) - p(x, y)| \leq \epsilon \cdot p(x, y) \text{ for all (x, y)}\},$$

  where $\pi(x, y|x^n, y^n)$ is joint type of $(x^n, y^n)$
Shannon’s joint typicality proof

- **Decoding:**
  - Find unique message \( \hat{m} \) such that \((x^n(\hat{m}), y^n) \in \mathcal{T}^{(n)}_\epsilon\)
  - Otherwise declare an error

- **Analysis of the probability of error:**
  - Straightforward to show that \( \lim_{n \to \infty} E[P_e^n(C_n(p))] = 0 \) if \( R < I(X; Y) \)
Shannon’s joint typicality proof

**Decoding:**
- Find **unique** message \( \hat{m} \) such that \( (x^n(\hat{m}), y^n) \in T_{\epsilon}^{(n)} \)
- Otherwise declare an error

**Analysis of the probability of error:**
- Straightforward to show that \( \lim_{n \to \infty} E[P_e^{(n)}(C_n(p))] = 0 \) if \( R < I(X; Y) \)
- Random ptp codes achieve capacity (use \( \arg \max_{p(x)} I(X; Y) \))
Shannon’s joint typicality proof

- **Decoding:**
  - Find unique message $\hat{m}$ such that $(x^n(\hat{m}), y^n) \in \mathcal{T}_\varepsilon^{(n)}$
  - Otherwise declare an error

- **Analysis of the probability of error:**
  - Straightforward to show that $\lim_{n \to \infty} \mathbb{E}[P_e^{(n)}(C_n(p))] = 0$ if $R < I(X; Y)$
  - Random ptp codes achieve capacity (use $\arg \max_{p(x)} I(X; Y)$)

- **Converse for random ptp codes:** Given $p(x)$ and a decoding rule
  - If $\lim_{n \to \infty} \mathbb{E}[P_e^{(n)}(C_n(p))] = 0$, then $R \leq I(X; Y)$
Shannon’s joint typicality proof

- **Decoding:**
  - Find unique message $\hat{m}$ such that $(x^n(\hat{m}), y^n) \in \mathcal{T}_\epsilon^{(n)}$
  - Otherwise declare an error

- **Analysis of the probability of error:**
  - Straightforward to show that $\lim_{n \to \infty} E[P_e^{(n)}(C_n(p))] = 0$ if $R < I(X; Y)$
  - $\Rightarrow$ Random ptp codes achieve capacity (use $\arg \max_{p(x)} I(X; Y)$)

- **Converse for random ptp codes:** Given $p(x)$ and a decoding rule
  - If $\lim_{n \to \infty} E[P_e^{(n)}(C_n(p))] = 0$, then $R \leq I(X; Y)$
  - $\Rightarrow$ Joint typicality decoding achieves same rate as MLD
  - Extensions are useful for networks where MLD is difficult to analyze
Practical point-to-point codes

- Most randomly generated ptp codes with rate $R < I(X; Y)$ are good
- But encoding/decoding them is computationally prohibitive
Practical point-to-point codes

- Most randomly generated ptp codes with rate $R < I(X; Y)$ are good
- But encoding/decoding them is computationally prohibitive
- Coding theorists spent over 60 years looking for practical ptp codes:
  - That approach capacity
  - Have computationally tractable encoding/decoding
Practical point-to-point codes

- Most randomly generated ptp codes with rate $R < I(X; Y)$ are good
- But encoding/decoding them is computationally prohibitive
- Coding theorists spent over 60 years looking for practical ptp codes:
  - That approach capacity
  - Have computationally tractable encoding/decoding
- Several such codes have been found:
  - Algebraic codes (e.g., BCH, Reed–Solomon, polar codes)
  - Random codes with structure (turbo, LDPC, fountain, spatially coupled)
- These codes are widely used in communication networks and storage
Practical point-to-point codes

- Most randomly generated ptp codes with rate $R < I(X; Y)$ are good
- But encoding/decoding them is computationally prohibitive
- Coding theorists spent over 60 years looking for practical ptp codes:
  - That approach capacity
  - Have computationally tractable encoding/decoding
- Several such codes have been found:
  - Algebraic codes (e.g., BCH, Reed–Solomon, polar codes)
  - Random codes with structure (turbo, LDPC, fountain, spatially coupled)
- These codes are widely used in communication networks and storage
- Results in network information theory suggest we need network codes
This lecture

- How well do random ptp codes perform over networks?
  - Do we need to spend another 60+ years looking for practical network codes?
This lecture

- How well do **random ptp codes** perform over networks?
  - Do we need to spend another 60+ years looking for practical network codes?

- Outline:
  - Brief introduction to network information theory
  - Discuss various network models with random ptp codes
This lecture

• How well do random ptp codes perform over networks?
  ▶ Do we need to spend another 60+ years looking for practical network codes?

• Outline:
  ▶ Brief introduction to network information theory
  ▶ Discuss various network models with random ptp codes

• Preview of the results:
  ▶ Random ptp codes with more sophisticated decoding can perform very well
  ▶ Joint-typicality-based decoding continues to achieve same rates as MLD
  ▶ There are settings where we may need to develop network codes
Network information theory

- Extends Shannon’s point-to-point information theory to networks with:
  - Multiple sources and destinations
  - Multiple access, broadcast, and interference
  - Feedback and interactive communication
  - Cooperation (multihop)
Network information theory

- Extends Shannon’s point-to-point information theory to networks
- First paper was by Shannon (1961) on the two-way channel
Network information theory

- Extends Shannon’s point-to-point information theory to networks
- First paper was by Shannon (1961) on the two-way channel
- Significant progress in 70s, early 80s (Cover 1972, Slepian–Wolf 1973)
Network information theory

- Extends Shannon’s point-to-point information theory to networks
- First paper was by Shannon (1961) on the two-way channel
- Significant progress in 70s, early 80s (Cover 1972, Slepian–Wolf 1973)
- Wireless communication and Internet revived interest in late 90s
Network information theory

- Extends Shannon’s point-to-point information theory to networks
- First paper was by Shannon (1961) on the two-way channel
- Significant progress in 70s, early 80s (Cover 1972, Slepian–Wolf 1973)
- Wireless communication and Internet revived interest in late 90s
- State of the theory:
Network models with ptp codes

- Multiple access channel
- Interference channel
- Broadcast channel
- Relay channel
- Multicast networks
- Multimessage networks
Multiple access channel (MAC)

- $(2^{nR_1}, 2^{nR_2}, n)$ block code:
  - **Message pair**: $(M_1, M_2) \sim \text{Unif}([1 : 2^{nR_1}] \times [1 : 2^{nR_2}])$
  - **Encoder $j = 1, 2$**: $x_j^n(m_j), \ m_j \in [1 : 2^{nR_j}]$
  - **Decoder**: $(\hat{m}_1, \hat{m}_2) \in [1 : 2^{nR_1}] \times [1 : 2^{nR_2}]$

- **Probability of error**: $P_e^{(n)} = P\{ (\hat{M}_1, \hat{M}_2) \neq (M_1, M_2) \}$
Multiple access channel (MAC)

- $(2^{nR_1}, 2^{nR_2}, n)$ block code:
  - **Message pair**: $(M_1, M_2) \sim \text{Unif}([1 : 2^{nR_1}] \times [1 : 2^{nR_2}])$
  - **Encoder $j = 1, 2$**: $x_j^n(m_j), m_j \in [1 : 2^{nR_j}]$
  - **Decoder**: $(\hat{m}_1, \hat{m}_2) \in [1 : 2^{nR_1}] \times [1 : 2^{nR_2}]$

- **Probability of error**: $P_e^{(n)} = P\{(\hat{M}_1, \hat{M}_2) \neq (M_1, M_2)\}$

- $(R_1, R_2)$ achievable if $\exists (2^{nR_1}, 2^{nR_2}, n)$ codes with $\lim_{n \to \infty} P_e^{(n)} = 0$

- **Capacity region $C$**: Closure of the set of achievable $(R_1, R_2)$

---

El Gamal (Stanford University)
Theorem (Ahlswede 1971, Liao 1972)

The capacity region is the convex hull of $\bigcup_{p(x_1)p(x_2)} R(X_1, X_2)$
MAC with random ptp codes

- **Codebook generation**: Fix $p(x_1)p(x_2)$
  - Randomly generate $2^{nR_1}$ sequences $x_1^n(m) \sim \prod_{i=1}^{n} p_{X_1}(x_{1i})$, $m_1 \in [1 : 2^{nR_1}]$
  - Randomly generate $2^{nR_2}$ sequences $x_2^n(m) \sim \prod_{i=1}^{n} p_{X_2}(x_{i2})$, $m_2 \in [1 : 2^{nR_2}]$

- **Encoding**: To send $(m_1, m_2)$, transmit $x_1^n(m_1)$ and $x_2^n(m_2)$
MAC with random ptp codes

- **Codebook generation**: Fix \( p(x_1)p(x_2) \)
  - Randomly generate \( 2^{nR_1} \) sequences \( x_1^n(m) \sim \prod_{i=1}^{n} p_{X_1}(x_{1i}), m_1 \in [1 : 2^{nR_1}] \)
  - Randomly generate \( 2^{nR_2} \) sequences \( x_2^n(m) \sim \prod_{i=1}^{n} p_{X_2}(x_{i2}), m_2 \in [1 : 2^{nR_2}] \)

- **Encoding**: To send \((m_1, m_2)\), transmit \( x_1^n(m_1) \) and \( x_2^n(m_2) \)

- **Decoding**: Use joint typicality decoding, treating the other codeword as noise:

\[
R_2 \\
\text{Other codeword as noise}
\]

\[
\begin{align*}
I(X_2; Y) \\
I(X_1; Y) \\
R_1
\end{align*}
\]
MAC with random ptp codes

- **Successive cancellation decoding (SCD):**
  - Find unique $\hat{m}_1$: $(x_1^n(\hat{m}_1), y^n) \in T_\epsilon^{(n)}$
  - If such $\hat{m}_1$ is found, find unique $\hat{m}_2$: $(x_1^n(\hat{m}_1), x_2^n(\hat{m}_2), y^n) \in T_\epsilon^{(n)}$
MAC with random ptp codes

- **Successive cancellation decoding (SCD):**
  - Find unique $\hat{m}_1$: $(x_1^n(\hat{m}_1), y^n) \in \mathcal{T}_e^{(n)}$
  - If such $\hat{m}_1$ is found, find unique $\hat{m}_2$: $(x_1^n(\hat{m}_1), x_2^n(\hat{m}_2), y^n) \in \mathcal{T}_e^{(n)}$

- Can achieve other corner point by reversing decoding order

![Diagram](image)
MAC with random ptp codes

- **Simultaneous decoding (SD):**
  - Find unique pair \((\hat{m}_1, \hat{m}_2) : (x_1^n(\hat{m}_1), x_2^n(\hat{m}_2), y^n) \in \mathcal{T}_e^n\)
MAC with random ptp codes

- **Simultaneous decoding (SD):**
  - Find unique pair \((\hat{m}_1, \hat{m}_2) : (x_1^n(\hat{m}_1), x_2^n(\hat{m}_2), y^n) \in \mathcal{T}_e^n\)

- **Converse for ptp codes:** Given \(p = p(x_1)p(x_2)\) and decoding rule
  - If \(\lim_{n \to \infty} \mathbb{E}[P_e(n)(C_n(p))] = 0\), then \((R_1, R_2) \in \mathcal{R}(X_1, X_2)\)
  - \(\Rightarrow\) SD achieves same rates as MLD

\[
R_2 \\
I(X_2; Y | X_1) \\
I(X_2; Y) \\
I(X_1; Y | X_2) \quad I(X_1; Y) \\
R_1
\]

\(\mathcal{R}(X_1, X_2)\)
Random ptp codes perform extremely well over MAC

**Theorem (Ahlswede 1971, Liao 1972)**

The capacity region is the convex hull of $\bigcup_{p(x_1)p(x_2)} R(X_1, X_2)$

- $R(X_1, X_2)$ achieved using random ptp codes + simultaneous decoding
- Rest of the capacity region is achieved using time-sharing
Random ptp codes perform extremely well over MAC

**Theorem (Ahlswede 1971, Liao 1972)**

The capacity region is the convex hull of \( \bigcup_{p(x_1)p(x_2)} R(X_1, X_2) \)

- \( R(X_1, X_2) \) achieved using random ptp codes + simultaneous decoding
- Rest of the capacity region is achieved using time-sharing
- Results generalize to more than two senders
Interference channel

- First studied by Ahlswede (1974)
Interference channel

- First studied by Ahlswede (1974)
- $(2^{nR_1}, 2^{nR_2}, n)$ code, $P_e^{(n)}$, achievability, capacity region $\mathcal{C}$: Same as MAC
- Capacity region is not known in general
- Coding schemes that are optimal in some cases
Interference channel with random ptp codes

- **Codebook generation:** Fix $p = p(x_1)p(x_2)$
  - Randomly generate $2^{nR_1}$ sequences $x_1^n(m_1) \sim \prod_{i=1}^n p_{X_1}(x_{1i})$, $m_1 \in [1 : 2^{nR_1}]$
  - Randomly generate $2^{nR_2}$ sequences $x_2^n(m_2) \sim \prod_{i=1}^n p_{X_2}(x_{2i})$, $m_2 \in [1 : 2^{nR_2}]$

- **Encoding:**
  - To send $(m_1, m_2)$, transmit $x_1^n(m_1)$ and $x_2^n(m_2)$
Interference channel with random ptp codes

- **Codebook generation**: Fix $p = p(x_1)p(x_2)$
  - Randomly generate $2^{nR_1}$ sequences $x_1^n(m_1) \sim \prod_{i=1}^n p_{X_1}(x_{1i})$, $m_1 \in [1 : 2^{nR_1}]$  
  - Randomly generate $2^{nR_2}$ sequences $x_2^n(m_2) \sim \prod_{i=1}^n p_{X_2}(x_{2i})$, $m_2 \in [1 : 2^{nR_2}]$

- **Encoding**:  
  - To send $(m_1, m_2)$, transmit $x_1^n(m_1)$ and $x_2^n(m_2)$

- **Decoding schemes we used for the MAC (receiver 1)**:

  - Treating interference as noise (IAN)
  - Simultaneous decoding (SD)
Neither scheme uniformly outperforms the other
Interference channel with random ptp codes

- Neither scheme uniformly outperforms the other
- IAN: Receiver 1 knows the codebook for $M_2$
  - If $R_2$ is not too high, it may be better to recover $M_2$ and achieve higher $R_1$

### Treating interference as noise (IAN)

$$R_2$$

$$I(X_1; Y_1)$$

### Simultaneous decoding (SD)

$$R_2$$

$$I(X_2; Y_1)$$

$$I(X_1; Y_1)$$

$$I(X_1; Y_1 | X_2)$$

$$I(X_1; Y_1)$$

$$I(X_2; Y_1)$$

$$I(X_1; Y_1)$$

$$I(X_1; Y_1 | X_2)$$

El Gamal  (Stanford University)
Interference channel with random ptp codes

- Neither scheme uniformly outperforms the other
- IAN: Receiver 1 knows the codebook for $M_2$
  - If $R_2$ is not too high, it may be better to recover $M_2$ and achieve higher $R_1$
- SD: Receiver 1 doesn’t really want to recover $M_2$
  - Insisting on recovering $M_2$ when $R_2$ is high artificially limits $R_1$

Treating interference as noise (IAN)  Simultaneous decoding (SD)
Interference channel with random ptp codes

- **Simultaneous nonunique decoding:**
  - Receiver 1 finds unique $\hat{m}_1$: $(X^n_1(\hat{m}_1), X^n_2(m_2), Y^n_1) \in \mathcal{T}^{(n)}_{\epsilon}$ for some $m_2$
Interference channel with random ptp codes

- **Simultaneous nonunique decoding:**
  - Receiver 1 finds unique \( \hat{m}_1: (X^n_1(\hat{m}_1), X^n_2(m_2), Y^n_1) \in T_e^{(n)} \) for some \( m_2 \)
- \( (R_1, R_2) \) achievable using SD \( \Rightarrow \) achievable using SND
Interference channel with random ptp codes

- **Simultaneous nonunique decoding:**
  - Receiver 1 finds unique $\hat{m}_1$: $(X_1^n(\hat{m}_1), X_2^n(m_2), Y_1^n) \in T_e^n$ for some $m_2$

- $(R_1, R_2)$ achievable using SD $\Rightarrow$ achievable using SND

- $(R_1, R_2)$ achievable using IAN $\Rightarrow$ achievable using SND

```
I(X_2; Y_1 | X_1)
```

```
I(X_2; Y_1)
```

```
I(X_1; Y_1) I(X_1; Y_1 | X_2)
```

```
R_2
```

```
R_1
```

El Gamal (Stanford University)
Interference channel with random ptp codes

- Simultaneous nonunique decoding:
  - Receiver 1 finds unique $\hat{m}_1$: $(X_1^n(\hat{m}_1), X_2^n(m_2), Y_1^n) \in \mathcal{T}_e^{(n)}$ for some $m_2$

- $(R_1, R_2)$ achievable using SD $\Rightarrow$ achievable using SND
- $(R_1, R_2)$ achievable using IAN $\Rightarrow$ achievable using SND

$\Rightarrow$ SND achieves union of regions for SD and IAN
Similar analysis can be performed for receiver 2, yielding $R_2(p)$.
Interference channel with random ptp codes

- Similar analysis can be performed for receiver 2, yielding $\mathcal{R}_2(p)$

**Theorem (Bandemer–EG–Kim 2012)**

The optimal rate region with random ptp codes $p(x_1)p(x_2)$ is:

$$\mathcal{R}(p) = \mathcal{R}_1(p) \cap \mathcal{R}_2(p)$$
Interference channel with random ptp codes

- Similar analysis can be performed for receiver 2, yielding $\mathcal{R}_2(p)$

**Theorem (Bandemer–EG–Kim 2012)**

The optimal rate region with random ptp codes $p(x_1)p(x_2)$ is:

$$\mathcal{R}(p) = \mathcal{R}_1(p) \cap \mathcal{R}_2(p)$$

- SND cannot achieve more than union of SD and IAN regions
- SND achieves same rates as MLD
Similar analysis can be performed for receiver 2, yielding $R_2(p)$

**Theorem (Bandemer–EG–Kim 2012)**

The optimal rate region with random ptp codes $p(x_1)p(x_2)$ is:

$$R(p) = R_1(p) \cap R_2(p)$$

- Optimal when *interference is strong* (Costa–EG 1987)
- Generalizes result for *deterministic IC* (Bandemer–EG 2011)
- Generalizes result for *Gaussian IC* (Baccelli–EG–Tse 2011)
Interference channel with random ptp codes

• Similar analysis can be performed for receiver 2, yielding $R_2(p)$

Theorem (Bandemer–EG–Kim 2012)

The optimal rate region with random ptp codes $p(x_1)p(x_2)$ is:

$$R(p) = R_1(p) \cap R_2(p)$$

• Optimal when interference is strong (Costa–EG 1987)
• Generalizes result for deterministic IC (Bandemer–EG 2011)
• Generalizes result for Gaussian IC (Baccelli–EG–Tse 2011)
• But, random ptp codes can do better
Han–Kobayashi (1981) coding scheme

- Idea: **Decode part of interference and treat rest as noise:**
  - Split message $M_j$, $j = 1, 2$, into **public message** $M_{j0}$ and **private message** $M_{jj}$
Han–Kobayashi (1981) coding scheme

- Idea: **Decode part of interference and treat rest as noise:**
  - Split message $M_j$, $j = 1, 2$, into **public message** $M_{j0}$ and **private message** $M_{jj}$.
  - Use **superposition encoding** (Cover 1972).
  - Fix $p(u_1)p(u_2)p(v_1)p(v_2)$, functions $x_1(u_1, v_1)$, $x_2(u_2, v_2)$.
Han–Kobayashi (1981) coding scheme

- **Idea:** Decode part of interference and treat rest as noise:
  - Split message $M_j$, $j = 1, 2$, into public message $M_{j0}$ and private message $M_{jj}$
  - Use superposition encoding (Cover 1972)
  - Fix $p(u_1)p(u_2)p(v_1)p(v_2)$, functions $x_1(u_1, v_1)$, $x_2(u_2, v_2)$

- **Codebook generation:** For $j = 1, 2$
  - Randomly generate $2^{nR_{j0}}$ sequences $u_j^n(m_{j0}) \sim \prod_{i=1}^n p_{U_j}(u_{ji})$, $m_{j0} \in [1 : 2^{nR_{j0}}]$
  - Randomly generate $2^{nR_{jj}}$ sequences $v_j^n(m_{jj}) \sim \prod_{i=1}^n p_{V_j}(u_{ji})$, $m_{jj} \in [1 : 2^{nR_{jj}}]$
Han–Kobayashi (1981) coding scheme

- **Idea:** Decode part of interference and treat rest as noise:
  - Split message $M_j$, $j = 1, 2$, into public message $M_{j0}$ and private message $M_{jj}$
  - Use superposition encoding (Cover 1972)
  - Fix $p(u_1)p(u_2)p(v_1)p(v_2)$, functions $x_1(u_1, v_1)$, $x_2(u_2, v_2)$

- **Codebook generation:** For $j = 1, 2$
  - Randomly generate $2^{nR_{j0}}$ sequences $u_{j0}(m_{j0}) \sim \prod_{i=1}^{n} p_{U_j}(u_{ji})$, $m_{j0} \in [1 : 2^{nR_{j0}}]
  - Randomly generate $2^{nR_{jj}}$ sequences $v_{jj}(m_{jj}) \sim \prod_{i=1}^{n} p_{V_j}(u_{ji})$, $m_{jj} \in [1 : 2^{nR_{jj}}]

- **Encoding:** To send $m_j$, $j = 1, 2$, transmit $x_j(u_{ji}(m_{j0}), v_{ji}(m_{jj}))$, $i \in [1 : n]$
Han–Kobayashi (1981) coding scheme

- Idea: Decode part of interference and treat rest as noise:
  - Split message $M_j$, $j = 1, 2$, into public message $M_{j0}$ and private message $M_{jj}$
  - Use superposition encoding (Cover 1972)
  - Fix $p(u_1)p(u_2)p(v_1)p(v_2)$, functions $x_1(u_1, v_1)$, $x_2(u_2, v_2)$

  ![Diagram showing the coding scheme with boxes for $x_1(u_1, v_1)$ and $x_2(u_2, v_2)$, and a joint distribution $p(y_1, y_2|x_1, x_2)$]

- Codebook generation: For $j = 1, 2$
  - Randomly generate $2^{nR_{j0}}$ sequences $u_j^n(m_{j0}) \sim \prod_{i=1}^{n} p_{U_j}(u_{ji})$, $m_{j0} \in [1 : 2^{nR_{j0}}]$
  - Randomly generate $2^{nR_{jj}}$ sequences $v_j^n(m_{jj}) \sim \prod_{i=1}^{n} p_{V_j}(u_{ji})$, $m_{jj} \in [1 : 2^{nR_{jj}}]$

- Encoding: To send $m_j$, $j = 1, 2$, transmit $x_j(u_{ji}(m_{j0}), v_{ji}(m_{jj}))$, $i \in [1 : n]$

- Decoding: SND is rate optimal (Bandemer–EG–Kim 2012)
Summary

- Multiple access channel:

- Interference channel:
  - Random ptp codes + superposition + SND achieve H–K bound
  - SND achieves the same rates as MLD
  - We don’t know how to do better than H–K

- Broadcast channel

- Multihop networks

- Multimessage networks
Broadcast channel

- First studied by Cover (1972)
Broadcast channel

- First studied by Cover (1972)
- \((2^{nR_1}, 2^{nR_2}, n)\) code, \(P_e^{(n)}\), achievability, capacity region \(C\): Same as MAC
- Capacity region is not known in general
- Coding schemes that are optimal in some cases
Broadcast channel with random ptp codes

- Use superposition coding: Fix $p(u_1)p(u_2)$, function $x(u_1, u_2)$
Broadcast channel with random ptp codes

- Use superposition coding: Fix $p(u_1)p(u_2)$, function $x(u_1, u_2)$

\[\begin{align*}
  U_1 &\rightarrow x(u_1, u_2) \rightarrow X \\
  U_2 &\rightarrow x(u_1, u_2) \rightarrow X \\
\end{align*}\]

\[\begin{align*}
  X &\rightarrow p(y_1, y_2|x) \\
  Y_1 &\rightarrow p(y_1, y_2|x) \\
  Y_2 &\rightarrow p(y_1, y_2|x)
\end{align*}\]

- Codebook generation:
  - Randomly generate $2^{nR_1}$ sequences $u_1^n(m_1) \sim \prod_{i=1}^{n} p_{U_1}(u_{1i}), m_1 \in [1 : 2^{nR_1}]$
  - Randomly generate $2^{nR_2}$ sequences $u_2^n(m_2) \sim \prod_{i=1}^{n} p_{U_2}(u_{2i}), m_2 \in [1 : 2^{nR_2}]$

- Encoding:
  - To send $(m_1, m_2)$, transmit $x(u_{1i}(m_1), u_{2i}(m_2)), i \in [1 : n]$
Broadcast channel with random ptp codes

- Use superposition coding: Fix $p(u_1)p(u_2)$, function $x(u_1, u_2)$

- Codebook generation:
  - Randomly generate $2^{nR_1}$ sequences $u_1^n(m_1) \sim \prod_{i=1}^{n} p_{U_1}(u_{1i})$, $m_1 \in [1 : 2^{nR_1}]$
  - Randomly generate $2^{nR_2}$ sequences $u_2^n(m_2) \sim \prod_{i=1}^{n} p_{U_2}(u_{2i})$, $m_2 \in [1 : 2^{nR_2}]$

- Encoding:
  - To send $(m_1, m_2)$, transmit $x(u_{1i}(m_1), u_{2i}(m_2))$, $i \in [1 : n]$

- Exactly the same as scheme for IC without superposition ($X_j \leftarrow U_j$)!
Broadcast channel with random ptp codes

- Use superposition coding: Fix $p(u_1)p(u_2)$, function $x(u_1, u_2)$

  ![Diagram](image)

  - $U_1$  \[ \rightarrow \]
  - $U_2$  \[ \rightarrow \]
  - $p(y_1, y_2|u_1, u_2)$
  - $Y_1$  \[ \rightarrow \]
  - $Y_2$  \[ \rightarrow \]

- Codebook generation:
  - Randomly generate $2^{nR_1}$ sequences $u_1^n(m_1) \sim \prod_{i=1}^n p_{U_1}(u_{1i})$, $m_1 \in [1 : 2^{nR_1}]$
  - Randomly generate $2^{nR_2}$ sequences $u_2^n(m_2) \sim \prod_{i=1}^n p_{U_2}(u_{2i})$, $m_2 \in [1 : 2^{nR_2}]$

- Encoding:
  - To send $(m_1, m_2)$, transmit $x(u_{1i}(m_1), u_{2i}(m_2))$, $i \in [1 : n]$

- Exactly the same as scheme for IC without superposition ($X_j \leftarrow U_j$)!

$\Rightarrow$ SND is rate optimal
Theorem (Bandemer–EG–Kim 2012)

The optimal rate region with random ptp codes \( p(u_1)p(u_2) \), \( x(u_1, u_2) \) is:

\[
\mathcal{R}(p) = \mathcal{R}_1(p) \cap \mathcal{R}_2(p)
\]
Broadcast channel with random ptp codes

**Theorem (Bandemer–EG–Kim 2012)**

The optimal rate region with random ptp codes $p(u_1)p(u_2)$, $x(u_1, u_2)$ is:

$$\mathcal{R}(p) = \mathcal{R}_1(p) \cap \mathcal{R}_2(p)$$

- Includes superposition coding region (Cover 1972, Bergmans 1973):
  - Optimal for degraded, less noisy, more capable BCs
- Also includes Cover–van der Meulen (1975) region
Theorem (Bandemer–EG–Kim 2012)

The optimal rate region with random ptp codes $p(u_1)p(u_2)$, $x(u_1, u_2)$ is:

$$R(p) = R_1(p) \cap R_2(p)$$

- Includes superposition coding region (Cover 1972, Bergmans 1973):
  - Optimal for degraded, less noisy, more capable BCs
- Also includes Cover–van der Meulen (1975) region
- We can do better using schemes beyond ptp codes
Marton (1979) coding scheme

- **Motivation:** Since \((M_1, M_2)\) is available at sender, *jointly* code them.

- **Fix** \(p(u_1, u_2)\) (instead of \(p(u_1)p(u_2)\)) and function \(x(u_1, u_2)\)

![Diagram of the coding scheme](image)
Marton (1979) coding scheme

- **Codebook generation**: Fix \( p(u_1, u_2) \) and function \( x(u_1, u_2) \)
  - Randomly generate ptp codebooks \( u^n_j(l_j), l_j \in [1 : 2^{n\tilde{R}_j}], \tilde{R}_j > R_j, j = 1, 2 \)
  - Partition each into subcodebooks \( C_j(m_j), m_j \in [1 : 2^{nR_j}], j = 1, 2 \)
Marton (1979) coding scheme

- **Encoding**: To send $(m_1, m_2)$:
  - Find pair $(u_1^n(l_1), u_2^n(l_2)) \in T_\epsilon^{(n)}, l_j \in C_j(m_j), j = 1, 2$
  - Transmit $x(u_1^i(l_1), u_2^i(l_2)), i \in [1:n]$
Summary

- Multiple access channel:
  - Random ptp codes achieve capacity region

- Interference channel
  - Random ptp codes achieve best known inner bound

- Broadcast channel:
  - Can do better than ptp codes using Marton coding

- Relay channel

- Multicast networks

- Multimessage networks
First studied by van der Meulen (1971)
First studied by van der Meulen (1971)

**Capacity** $C$ is not known in general

Coding schemes that are optimal in some cases
RC with random ptp codes: Multihop

- Send \( b - 1 \) messages over \( b \) \( n \)-transmission blocks

<p>| | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>( M_1 )</td>
<td>( M_2 )</td>
<td>( M_3 )</td>
<td>( \cdots )</td>
<td>( M_{b-1} )</td>
</tr>
<tr>
<td>Block 1</td>
<td>Block 2</td>
<td>Block 3</td>
<td>( \cdots )</td>
<td>Block ( b - 1 )</td>
</tr>
</tbody>
</table>

- Fix \( p(x_1)p(x_2) \), generate (for each block) ptp codes for sender and relay
RC with random ptp codes: Multihop

- Send $b - 1$ messages over $b$ $n$-transmission blocks

\[ \tilde{M}_{j-1} \]

\[ Y_2 : X_2 \]

\[ M_j \rightarrow X_1 \rightarrow Y_3 \]
RC with random ptp codes: Multihop

- Send $b - 1$ messages over $b$ $n$-transmission blocks

![Diagram of communication process]

$M_j \rightarrow X_1 \rightarrow Y_2 : X_2 \rightarrow Y_3 \rightarrow \tilde{M}_{j-1}$
RC with random ptp codes: Multihop

- Send \( b - 1 \) messages over \( b \) \( n \)-transmission blocks

\[ R = \min \{ I(X_1; Y_2|X_2), \ I(X_2; Y_3) \} \]
RC with random ptp codes: Multihop

- Send \( b - 1 \) messages over \( b \) \( n \)-transmission blocks

\[
\begin{align*}
\tilde{M}_j \tilde{M}_{j-1} \\
\uparrow \downarrow \\
Y_2 : X_2 \\
M_j \rightarrow X_1 \\
\rightarrow Y_3 \rightarrow \tilde{M}_{j-1}
\end{align*}
\]

- This achieves:

\[
R = \min \{ I(X_1; Y_2 | X_2), \ I(X_2; Y_3) \}
\]

- Optimal for a cascade of two DMCs: \( p(y_2, y_3 | x_1, x_2) = p(y_2 | x_1) p(y_3 | x_2) \)

\[
C = \min \left\{ I(X_1; Y_2), \ I(X_2; Y_3) \right\}
\]
RC with random ptp codes: Multihop

- Send $b - 1$ messages over $b$ $n$-transmission blocks

This achieves:
\[
R = \min \{ I(X_1; Y_2 | X_2), I(X_2; Y_3) \}
\]

Optimal for a cascade of two DMCs: $p(y_2, y_3 | x_1, x_2) = p(y_2 | x_1)p(y_3 | x_2)$

\[
C = \min \left\{ I(X_1; Y_2), I(X_2; Y_3) \right\}
\]

- But we can do better via superposition and more sophisticated decoding
RC with random ptp codes: Decode–forward (DF)

- The sender knows what the relay knows:
  - They can coherently cooperate via superposition coding at the sender
  - Fix $p(x_2)p(u)$ and $x_1(u, x_2)$, generate ptp codes for relay and sender
- Receiver uses backward decoding (Zeng–Kuhlmann–Buzo 1989)

\[\begin{align*}
\tilde{M}_j &\quad \tilde{M}_{j-1} \\
Y_2 &\quad X_2
\end{align*}\]
RC with random ptp codes: Decode–forward (DF)

- The sender knows what the relay knows:
  - They can **coherently cooperate** via superposition coding at the sender
  - Fix $p(x_2)p(u)$ and $x_1(u, x_2)$, generate ptp codes for relay and sender

- Receiver uses **backward decoding** (Zeng–Kuhlmann–Buzo 1989)

- Achieves (Cover–EG 1979):

  $$ R = \min \{ I(X_1; Y_2 | X_2), \ I(X_1, X_2; Y_3) \} $$
RC with random ptp codes: Decode–forward (DF)

• The sender knows what the relay knows:
  ▶ They can coherently cooperate via superposition coding at the sender
  ▶ Fix $p(x_2)p(u)$ and $x_1(u, x_2)$, generate ptp codes for relay and sender

• Receiver uses backward decoding (Zeng–Kuhlmann–Buzo 1989)

\[
\begin{align*}
&\tilde{M}_j \quad \tilde{M}_{j-1} \\
&\uparrow \quad \downarrow \\
&Y_2 : X_2 \\
&(M_{j-1}, M_j) \rightarrow X_1 \rightarrow Y_3 \rightarrow \hat{M}_{j-1}
\end{align*}
\]

• Achieves (Cover–EG 1979):

\[
R = \min\{I(X_1; Y_2|X_2), I(X_1, X_2; Y_3)\}
\]

• Optimal for physically degraded RC: $(X_1, X_2) \rightarrow (Y_2, X_2) \rightarrow Y_3$
• DF does not do well when $X_1 \rightarrow Y_2$ is not better than $X_1 \rightarrow Y_3$
Compress–forward (Cover–EG 1979)

- DF does not do well when $X_1 \rightarrow Y_2$ is not better than $X_1 \rightarrow Y_3$
- Can do better by relaying compressed version of $Y_2^n$ instead of DF
Compress–forward (Cover–EG 1979)

- DF does not do well when $X_1 \rightarrow Y_2$ is not better than $X_1 \rightarrow Y_3$
- Can do better by relaying compressed version of $Y_2^n$ instead of DF

\[ \hat{Y}_{2,j} \hat{Y}_{2,j-1} \]

\[ Y_2 : X_2 \]

- Does not use ptp codes as defined
- Optimal for some RCs (Aleksic–Razaghi–Yu 2009)
Summary

- Multiple access channel:
- Interference channel:
- Broadcast channel
- Relay channel
  - Random ptp codes + superposition + backward decoding achieve DF bound
  - Can sometimes do better using schemes beyond ptp codes, e.g., CF
- Multicast networks
- Multimessage networks:
Multicast network

Capacity is not known in general (relay channel is special case)
Multicast network

- Capacity is not known in general (relay channel is special case)
- DF can be extended (Xie–Kumar 2005, Kramer–Gastpar–Gupta 2005)
- CF can be extended (noisy network coding (EG–Kim 2011))
Multicast network

- Capacity is not known in general (relay channel is special case)
- DF can be extended (Xie–Kumar 2005, Kramer–Gastpar–Gupta 2005)
- CF can be extended (noisy network coding (EG–Kim 2011))
- **Gaussian multicast network:**
  - CF achieves within $0.63N$ bit of capacity (Lim–Kim–EG–Chung 2011)
  - DF can have unbounded gap to capacity
Node $j$ wishes to send message $M_j$ to set of destination nodes $D_j$
Multimessage network

- Node $j$ wishes to send message $M_j$ to set of destination nodes $D_j$
- No general capacity gap result exists for Gaussian multimessage network
Node $j$ wishes to send message $M_j$ to set of destination nodes $D_j$

No general capacity gap result exists for Gaussian multimessage network

How well do ptp codes perform?
Capacity scaling laws

- Random wireless network (Gupta–Kumar 2000):
  - **Multi-unicast**: Send message $M_j$ at rate $R$ from node $j \in [1 : N]$ to $k = j + N$
  - **Gaussian network model**: Path loss exponent $\nu \geq 2$, average power constraint
  - What is the scaling law for symmetric capacity $C$ (highest achievable $R$)?
Random wireless network (Gupta–Kumar 2000):

- **Multi-unicast**: Send message $M_j$ at rate $R$ from node $j \in [1 : N]$ to $k = j + N$
- **Gaussian network model**: Path loss exponent $\nu \geq 2$, average power constraint
- What is the scaling law for symmetric capacity $C$ (highest achievable $R$)?

- **Ptp codes + cellular time division + multihop**: $C = \Omega(N^{-1/2})$
Capacity scaling laws

- **Random wireless network (Gupta–Kumar 2000):**
  - **Multi-unicast:** Send message $M_j$ at rate $R$ from node $j \in [1 : N]$ to $k = j + N$
  - **Gaussian network model:** Path loss exponent $\nu \geq 2$, average power constraint
  - What is the **scaling law** for symmetric capacity $C$ (highest achievable $R$)?

- **Ptp codes + cellular time division + multihop:** $C = \Omega(N^{-1/2})$

- **Upper bound (Lévêque–Telatar 2005):** $C = O(N^{-1/2 + 1/\nu} \log N)$
Random ptp codes with sophisticated decoding can perform well:

- Achieve capacity region of MAC
- Achieve the Han–Kobayashi bound for IC (best known)
- Achieve the superposition coding and Cover–van der Meulen bounds for BC
- Achieve decode–forward bound for multicast networks
- Achieve close to optimal scaling law for random multi-unicast networks
Conclusion and final remarks

- Random ptp codes with **sophisticated decoding** can perform well
- Joint-typicality-based decoding does as well as MLD
Random ptp codes with sophisticated decoding can perform well.

Joint-typicality-based decoding does as well as MLD.

There are scenarios in which we need new network codes:
- Marton coding for BC
- Compress–forward (noisy network coding) for relay (networks)

These network codes employ lossy compression and structured codes.
Conclusion and final remarks

- Random ptp codes with sophisticated decoding can perform well
- Joint-typicality-based decoding does as well as MLD
- There are scenarios in which we need new network codes:
  - Marton coding for BC
  - Compress–forward (noisy network coding) for relay (networks)
- These network codes employ lossy compression and structured codes
- What does all this say about practical coding and communication?
Conclusion and final remarks

- Random ptp codes with sophisticated decoding can perform well
- Joint-typicality-based decoding does as well as MLD
- There are scenarios in which we need new network codes:
  - Marton coding for BC
  - Compress–forward (noisy network coding) for relay (networks)
- These network codes employ lossy compression and structured codes
- What does all this say about practical coding and communication?
  - Work on constructing better decoders
  - Find better practical codes for broadcasting and relaying
Random ptp codes with sophisticated decoding can perform well

Joint-typicality-based decoding does as well as MLD

There are scenarios in which we need new network codes:
  - Marton coding for BC
  - Compress–forward (noisy network coding) for relay (networks)

These network codes employ lossy compression and structured codes

What does all this say about practical coding and communication?
  - Work on constructing better decoders
  - Find better practical codes for broadcasting and relaying

The best answer, to quote Tom Cover, is:

“Theory is the first term in the Taylor series expansion of practice.”
Thank You!


