Joint Channel Coding and Physical-Layer Network Coding Design for Gaussian Two-Way Relay Channels

Jinhong Yuan

University of New South Wales
Sydney, Australia

7 September, 2011

Outline

Introduction

Network Coding (NC) and Physical Layer Network Coding (PNC)

- **NC**: Coding better than suboptimal routing [1]
- **PNC** can enhance the throughput of a multi-user wireless network [2].
- **Channel coded PNC (CPNC)** can approach the capacity (upper bound) of a Gaussian two-way relay channel (TWRC) within 1/2 bit [3].
- The pioneering work on designing **practical CPNC** schemes was reported in [4].

Motivation

- To date, both convolutional coded or repeat-accumulate (RA) coded PNC schemes have been investigated by simulation.
- Some open questions:
  - Whether the conventional good channel codes remain good for PNC?
  - How to design capacity approaching channel codes for PNC schemes?

[1] R. Ahlswede, N. Cai, S.-Y. R. Li, and R. W. Yeung, "Network Information Flow, " *IEEE Transactions on Information Theory*, IT-46, pp. 1204-1216, 2000.

[2] S. Zhang, S.-C. Liew, and P. P. Lam, "Hot topic: physical layer network coding," *Proc. 12th Annual International Conference on Mobile Computing and Networking (MobiCom)*, pp. 358-365, Los Angeles, California, USA, Sept. 2006..

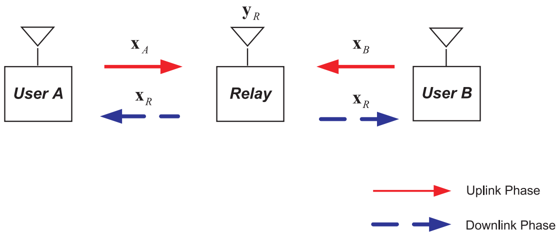
[3] W. Nam, S.-Y. Chung, and Y. H. Lee, "Capacity of the Gaussian twoway relay channel to within 1/2 bit," *IEEE Trans. Inform. Theory*, vol. 56, no. 11, pp. 5488-5494, Nov. 2010.

[4] S. Zhang and S.-C. Liew, "Channel coding and decoding in a relay system operated with physical-layer network coding," IEEE Jour. Select Area. Commun., vol. 27, pp. 788-796, June 2009.

[5] S. Liew, S. Zhang, and L. Lu, "Physical-layer network coding: tutorial, survey, and beyond," 2011.

System Model

Gaussian Two-way Relay Channel (TWRC)

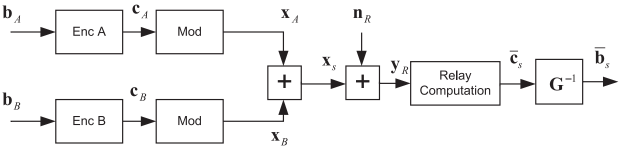


- Consider a Gaussian TWRC where user A and user B exchange information via an intermediate relay R.
- Two phases: **uplink phase**, the users transmit simultaneously to the relay; **downlink phase**, the relay broadcasts to the users.
- **No direct link** between A and B, single antenna nodes.
- At each node, the received signal is corrupted by AWGN.

System Model

UNSW THE UNIVERSITY OF NEW SOUTH WALES SYDNEY • AUSTRALIA

Uplink Phase of CPNC



- Messages: Binary message sequences $\mathbf{b}_A \in \{0,1\}^k$ and $\mathbf{b}_B \in \{0,1\}^k$.
- Encoding: The messages of users are encoded with the **same binary linear codes**, generating the coded sequences $\mathbf{c}_A \in \{0,1\}^n$ and $\mathbf{c}_B \in \{0,1\}^n$.

$$\mathbf{c}_A = \mathbf{b}_A \mathbf{G}, \quad \mathbf{c}_B = \mathbf{b}_B \mathbf{G}$$

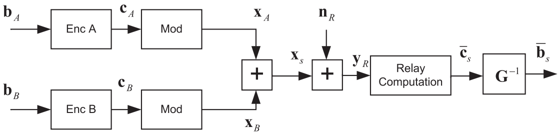
  Generator matrix: $\mathbf{G}$, and **codebook**: $\mathcal{C}$.

  Code rate of each user: $R = k/n$.

- Air interface: The coded sequences are BPSK modulated, obtaining the signal sequence $\mathbf{x}_m = 2\mathbf{c}_m - 1 \in \{-1,1\}^n$, $m \in \{A,B\}$.

System Model

Uplink Phase (Conti.)



- The signal received by the relay

$$\mathbf{y}_R = \sqrt{E_s}\mathbf{x}_A + \sqrt{E_s}\mathbf{x}_B + \mathbf{n}_R$$

- The relay decodes the **network codeword**

$$\mathbf{c}_s \triangleq \mathbf{c}_A \oplus \mathbf{c}_B$$

and computes **network codeword's message**

$$\mathbf{b}_s = \mathbf{b}_A \oplus \mathbf{b}_B$$

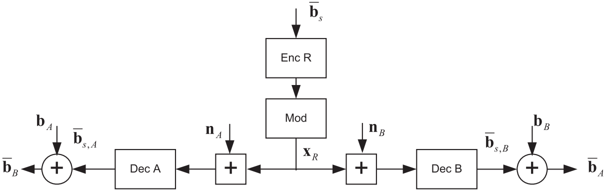
- Since the same channel code was used by the two users

$$\mathbf{c}_s = \mathbf{b}_s\mathbf{G}$$

- If the computed NC message $\bar{\mathbf{b}}_s \neq \mathbf{b}_s$, a **computation error** is declared.

System Model

Downlink Phase of CPNC



- The recovered NC message $\bar{\mathbf{b}}_s$ is **re-encoded**, BPSK-modulated, generating $\mathbf{x}_R$, then broadcasted to the two users.
- User m, $m \in \{A, B\}$, receives signal

$$\mathbf{y}_m = \sqrt{E_R}\mathbf{x}_R + \mathbf{n}_m$$

- Each user decodes the NC message sequence $\mathbf{b}_s = \mathbf{b}_A \oplus \mathbf{b}_B$, and then recovers the other user's message by XOR-ing $\mathbf{b}_s$ with its own message.

System Model

Remarks

- The **operation in the downlink is** a standard single-user decoding, followed by a simple XOR operation.
- Focus: uplink
- **Key problem: how to efficiently recover the NC message sequence $\mathbf{b}_s$** (or the Network codeword $\mathbf{c}_s$) at the relay in the uplink.

Optimal Decoding of the Network Codeword

UNSW THE UNIVERSITY OF NEW SOUTH WALES SYDNEY • AUSTRALIA

Preliminaries

- Recall the received signal at the relay:

$$\mathbf{y}_R = \sqrt{E_s}(\mathbf{x}_A + \mathbf{x}_B) + \mathbf{n}_R = \sqrt{E_s}\mathbf{x}_s + \mathbf{n}_R$$

- The relay receives a "ternary superimposed (SI) signal".

$$\mathbf{x}_s \triangleq \mathbf{x}_A + \mathbf{x}_B \in \{-2, 0, 2\}$$

- Then, recovers the binary network codeword $\mathbf{c}_s = \mathbf{c}_A \oplus \mathbf{c}_B$.

- The maximum likelihood (ML) decoding of the network codeword $\mathbf{c}_s$

$$\bar{\mathbf{c}}_s = \underset{\mathbf{c}_s \in \mathcal{C}}{\arg\max}\ p\left(\mathbf{y}_R | \mathbf{c}_s\right)$$

- Given each network codeword $\mathbf{c}_s$, there is a set of superimposed signals $\mathbf{x}_s$ associated with it

$$\mathcal{X}_s(\mathbf{c}_s) \triangleq \{\mathbf{x}_s = \mathbf{x}_A + \mathbf{x}_B : \mathbf{c}_A \oplus \mathbf{c}_B = \mathbf{c}_s, \mathbf{c}_A, \mathbf{c}_B \in \mathcal{C}\}$$

Optimal Decoding of the Network Codeword

- The ML rule:

$$\begin{aligned}\overline{\mathbf{c}}_s &= \arg\max_{\mathbf{c}_s \in \mathcal{C}} p\left(\mathbf{y}_R | \mathbf{c}_s\right) \\ &= \arg\max_{\mathbf{c}_s \in \mathcal{C}} \sum_{\mathbf{x}_s \in \mathcal{X}_s(\mathbf{c}_s)} p\left(\mathbf{y}_R | \mathbf{x}_s, \mathbf{c}_s\right) p\left(\mathbf{x}_s | \mathbf{c}_s\right) \\ &= \arg\max_{\mathbf{c}_s \in \mathcal{C}} \sum_{\mathbf{x}_s \in \mathcal{X}_s(\mathbf{c}_s)} p\left(\mathbf{y}_R | \mathbf{x}_s, \mathbf{c}_s\right) \frac{1}{|\mathcal{X}_s(\mathbf{c}_s)|}\end{aligned}$$

- Minimum Euclidean distance decoding
  - The most likely superimposed signal sequence $\overline{\mathbf{x}}_s$ is found by

$$\begin{aligned}\overline{\mathbf{x}}_s &= \arg\max_{\mathbf{x}_s \in \mathbb{X}_s} p\left(\mathbf{y}_R | \mathbf{x}_s\right) \\ &= \arg\min_{\mathbf{x}_s \in \mathbb{X}_s} \left|\mathbf{y}_R - \mathbf{x}_s\right|^2\end{aligned}$$

  - Mapping the most likely superimposed signal sequence to the network codeword

$$\overline{\mathbf{x}}_s \mapsto \overline{\mathbf{c}}_s$$

Optimal Decoding of the Network Codeword

- Our goal is to find the **error probabilities** of the above ML computation.
- To do this, we need
  - find the cardinality of set $\mathcal{X}_s(\mathbf{c}_s)$
  - obtain the **distance spectrum** of the CPNC scheme.
  - the error probability not only depends on **Hamming Distance**, but also the **Euclidean Distances** from the superimposed signals, even for binary modulation

Theorem 1. The cardinality of the set $\mathcal{X}_s(\mathbf{c}_s)$ is given by

$$|\mathcal{X}_s(\mathbf{c}_s)| = 2^{\mathrm{Rank}\left(\mathbf{G}^{\mathcal{S}^c(\mathbf{c}_s)}\right)}$$

$\mathbf{G}^{\mathcal{S}^c(\mathbf{c}_s)}$ is obtained by removing the columns indexed by t where $c_s(t) = 1$ from the original generator matrix $\mathbf{G}$.

Punctured Codebook Approach

Example 1. Consider a $(7, 4)$ Hamming code with

$$\mathbf{G} = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 0 & 1 \end{bmatrix}.$$

Let $\mathbf{c}_s$ be a certain codeword in $\mathcal{C}$, e.g., $\mathbf{c}_s = [0\ 0\ 1\ 1\ 0\ 1\ 0]$. Then, we have $\mathcal{S}(\mathbf{c}_s) = \{3, 4, 6\}$. Deleting Column 3, 4 and 6 of $\mathbf{G}$, we obtain

$$\mathbf{G}^{\mathcal{S}^c(\mathbf{c}_s)} = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 \end{bmatrix}.$$

The cardinality of the set $\mathcal{X}_s(\mathbf{c}_s)$ is given by

$$|\mathcal{X}_s(\mathbf{c}_s)| = 2^{\text{Rank}\left(\mathbf{G}^{\mathcal{S}^c(\mathbf{c}_s)}\right)} \tag{1}$$

Why Hamming and Euclidean Distances?

Example with (7, 4) Hamming Code

- Recall $\mathbf{c}_s = \mathbf{c}_A \oplus \mathbf{c}_B$, $\mathbf{x}_s = \mathbf{x}_A + \mathbf{x}_B$;
- Transmitted Codewords
  $\mathbf{c}_A = 0000000$, $\mathbf{x}_A = -1 -1 -1 -1 -1 -1 -1$;
  $\mathbf{c}_B = 0111001$, $\mathbf{x}_B = -1 +1 +1 +1 -1 -1 +1$;
  $\mathbf{c}_s = 0111001$, $\mathbf{x}_s = -2 \quad 0 \quad 0 \quad 0 -2 -2 \quad 0$;
- If decodes to $\mathbf{c}_s^* = 1110000$, $d_H(\mathbf{c}_s, \mathbf{c}_s^*) = 3$.
  - $\mathbf{c}_A^* = 0000000$, $\mathbf{x}_A = -1 -1 -1 -1 -1 -1 -1$;
    $\mathbf{c}_B^* = 1110000$, $\mathbf{x}_A = +1 +1 +1 -1 -1 -1 -1$;
    $\mathbf{x}_s^* = 0 \quad 0 \quad 0 -2 -2 -2 -2$;
    $d_E^2(\mathbf{x}_s, \mathbf{x}_s^*) = (-2)^2 + (2)^2 + (2)^2 = 12$.
  - $\mathbf{c}_A^* = 0001111$, $\mathbf{x}_A = -1 -1 -1 +1 +1 +1 +1$;
    $\mathbf{c}_B^* = 1111111$, $\mathbf{x}_A = +1 +1 +1 +1 +1 +1 +1$;
    $\mathbf{x}_s^* = 0 \quad 0 \quad 0 +2 +2 +2 +2$;
    $d_E^2(\mathbf{x}_s, \mathbf{x}_s^*) = (-2)^2 + (-2)^2 + (-4)^2 + (-4)^2 + (-2)^2 = 44$.
  - $\mathbf{c}_A^* = 0010011$, $\mathbf{x}_A = -1 -1 +1 -1 -1 +1 +1$;
    $\mathbf{c}_B^* = 1100011$, $\mathbf{x}_A = +1 +1 -1 -1 -1 +1 +1$;
    $\mathbf{x}_s^* = 0 \quad 0 \quad 0 -2 -2 +2 +2$;
    $d_E^2(\mathbf{x}_s, \mathbf{x}_s^*) = (-2)^2 + (2)^2 + (-4)^2 + (-2)^2 = 28$.

Error Performance Analysis

Pair-wise error probability

- Consider the genuine transmitted signal sequences $\mathbf{x}_A$ and $\mathbf{x}_B$, $\mathbf{x}_s = \mathbf{x}_A + \mathbf{x}_B$ and its network codeword $\mathbf{c}_s$.
- Let $\mathbf{c}_s^*$ be the wrong network codeword been detected .
- The pair-wise error probability (PEP)

$$P(\mathbf{x}_s \to \mathbf{c}_s^*)$$

  is determined by the distance between two network codewords.
- Each competing network codeword $\mathbf{c}_s^*$ has a set of superimposed signals $\mathcal{X}_s(\mathbf{c}_s^*)$

$$P(\mathbf{x}_s \to \mathbf{c}_s^*) = P(\mathbf{x}_s \to \mathcal{X}_s(\mathbf{c}_s^*))$$

- We partition the competing set $\mathcal{X}_s(\mathbf{c}_s^*)$ into *subsets* according to its Euclidean distance.

$$\mathcal{X}_s^d\left(\mathbf{x}_s^*, \mathbf{c}_s\right) \triangleq \{\mathbf{x}_s \in \mathcal{X}_s(\mathbf{c}_s) \ : \ \|\mathbf{x}_s - \mathbf{x}_s^*\|^2 = d^2\}$$

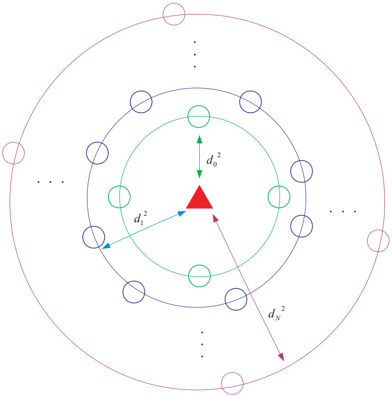
- We define the *pair-wise distance spectrum (PDS)* between $\mathbf{x}_s^*$ and $\mathcal{X}_s(\mathbf{c}_s)$ as

$$\mathbb{J}\left(\mathbf{x}_s^*, \mathbf{c}_s\right) \triangleq \left\{ \left(d_0, |\mathcal{X}_s^{d_0}\left(\mathbf{x}_s^*, \mathbf{c}_s\right)|\right), \ldots, \left(d_N, |\mathcal{X}_s^{d_N}\left(\mathbf{x}_s^*, \mathbf{c}_s\right)|\right) \right\}.$$

Error Performance Analysis

Pair-wise Distance Spectrum (PDS)

An illustration of PDS



△: Genuine transmitted superimposed sequence $\mathbf{x}_s$

◯: The superimposed sequences of a competing set $\mathcal{X}_s(\mathbf{c}_s^*)$

The points on the inner-most circle is called the "minimum distance subset".

Error Performance Analysis

UNSW THE UNIVERSITY OF NEW SOUTH WALES SYDNEY • AUSTRALIA

Pair-wise Distance Spectrum

- **Theorem 2**. The PDS w.r.t. $\mathbf{x}_s$ and $\mathcal{X}_s(\mathbf{c}_s^*)$ is given by

$$\mathbb{J}\left(\mathbf{x}_s, \mathbf{c}_s^*\right) = \frac{\mathbb{A}\left(\mathbf{C}^{\mathcal{S}^c(\mathbf{c}_s)\cap\mathcal{S}^c(\mathbf{c}_s^*)}\right)}{O\left(\mathbf{c}_s^*\right)}.$$

where $\mathbb{A}(\cdot)$ is the weight distribution of $\mathbf{C}^{\mathcal{S}^c(\mathbf{c}_s)\cap\mathcal{S}^c(\mathbf{c}_s^*)}$ and

$$O\left(\mathbf{c}_s\right) = 2^{nR-\mathrm{Rank}\left(\mathbf{G}^{\mathcal{S}^c(\mathbf{c}_s)}\right)}.$$

- **Corollary 1**. The cardinality of the "minimum distance subset" is upper-bounded by

$$\begin{aligned}\left|\mathcal{X}_s^{d_0}\left(\mathbf{x}_s^*, \mathbf{c}_s\right)\right| &= 2^{\mathrm{Rank}\left(\mathbf{G}^{\mathcal{S}^c(\mathbf{c}_s)}\right)-\mathrm{Rank}\left(\mathbf{G}^{\mathcal{S}^c(\mathbf{c}_s^*)\cap\mathcal{S}^c(\mathbf{c}_s)}\right)} \\ &\leq 2^{\left|\mathcal{S}(\mathbf{c}_s^*)\cap\mathcal{S}^c(\mathbf{c}_s)\right|} = 2^{d_{10}\left(\mathbf{c}_s^*,\mathbf{c}_s\right)} \end{aligned} \tag{2}$$

where $d_{10}\left(\mathbf{c}_s^*, \mathbf{c}_s\right) \triangleq \left|\mathcal{S}\left(\mathbf{c}_s^*\right)\cap\mathcal{S}^c\left(\mathbf{c}_s\right)\right|$.

[4] T. Yang, I. Land, T. Huang, J. Yuan, and Z. Chen, "Distance properties and performance of physical layer network coding with binary linear codes for Gaussian two-way relay channels," *Proc. IEEE ISIT*, Aug. 2011.

Error Performance Analysis

Pair-wise Error Probability Upper Bound

- With the union bound technique, the pair-wise error probability (PEP) that the decoder recovers $\mathbf{c}_s^* \neq \mathbf{c}_s$ is upper bounded by

$$P_e\left(\mathbf{x}_s, \mathcal{X}_s(\mathbf{c}_s^*)\right) \leq \sum_{i=0}^{N} \left|\mathcal{X}_s^{d_i}\left(\mathbf{x}_s, \mathbf{c}_s^*\right)\right| Q\left(\sqrt{\frac{E_s d_H\left(\mathbf{c}_s^*, \mathbf{c}_s\right) + i \cdot 4E_s}{\sigma^2}}\right) \quad (3)$$

  where $N = \left|\mathcal{S}\left(\mathbf{c}_s\right) \cap \mathcal{S}^c\left(\mathbf{c}_s^*\right)\right|$.
- To compute the PEP union bound, we need to find the PDS $\mathbb{J}\left(\mathbf{c}_s^*, \mathbf{c}_s\right)$.
- For a short code, $\mathbb{J}\left(\mathbf{c}_s^*, \mathbf{c}_s\right)$ can be determined based on Theorem 2.
- However, as n increases, the number of distinct rows in $\mathbf{C}^{\mathcal{S}^c\left(\mathbf{c}_s^*\right)\cap\mathcal{S}^c\left(\mathbf{c}_s\right)}$ increases exponentially with n and the task quickly becomes prohibitive.
- To simplify this task, we now consider an upper bound for the **high SNR** case.

Error Performance Analysis

UNSW THE UNIVERSITY OF NEW SOUTH WALES SYDNEY • AUSTRALIA

Asymptotic Pair-wise Error Probability Bound

- **Lemma:** Asymptotically, the PEP upper bound is approximated as

$$\begin{aligned} P_e\left(\mathbf{x}_s, \mathcal{X}_s(\mathbf{c}_s^*)\right) &\lesssim \left|\mathcal{X}_s^{d_0}\left(\mathbf{x}_s, \mathbf{c}_s^*\right)\right| Q\left(\sqrt{\frac{E_s d_H\left(\mathbf{c}_s^*, \mathbf{c}_s\right)}{\sigma^2}}\right) \\ &\leq 2^{d_{10}\left(\mathbf{c}_s^*, \mathbf{c}_s\right)} Q\left(\sqrt{\frac{E_s d_H\left(\mathbf{c}_s^*, \mathbf{c}_s\right)}{\sigma^2}}\right) \end{aligned}$$

- This means that at a high SNR, the PEP is only determined by the minimum distance subset.
- For more insight, we consider a single-user one-way relay (OWRC) case. The PEP of this OWRC is

$$P_e^{SU}\left(\mathbf{c}_s^*, \mathbf{c}_s\right) \leq Q\left(\sqrt{\frac{E_s d_H\left(\mathbf{c}_s^*, \mathbf{c}_s\right)}{\sigma^2}}\right)$$

.

- At high SNRs, the PEP of the CPNC over TWRC is approximately **increased by a factor of (at most)** $2^{d_{10}\left(\mathbf{c}_s^*, \mathbf{c}_s\right)}$ relative to the single-user case.

Error Performance Analysis

Conditional Word Error Probability

- The word error probability (WEP) conditioned on $\mathbf{x}_s^* \in \mathcal{X}_s(\mathbf{c}_s^*)$ is

$$P_e\left(\mathbf{c}_s^*\right) \lesssim \sum_{\mathbf{c}_s \in \mathcal{C}, \mathbf{c}_s \neq \mathbf{c}_s^*} 2^{d_{10}\left(\mathbf{c}_s^*, \mathbf{c}_s\right)} Q\left(\sqrt{\frac{E_s d_H\left(\mathbf{c}_s^*, \mathbf{c}_s\right)}{\sigma^2}}\right).$$

- The parameter $d_{10}\left(\mathbf{c}_s^*, \mathbf{c}_s\right)$ is codeword-dependent.
- For random codes, we have $\Pr\left[\left|d_{10}\left(\mathbf{c}_s^*, \mathbf{c}_s\right) - \frac{d_H\left(\mathbf{c}_s^*, \mathbf{c}_s\right)}{2}\right| < \varepsilon\right] \overset{n\to\infty}{\rightarrow} 1$ for an arbitrarily small ε [5].
- For long linear codes, we may assume

$$d_{10}\left(\mathbf{c}_s^*, \mathbf{c}_s\right) \approx \frac{d_H\left(\mathbf{c}_s^*, \mathbf{c}_s\right)}{2}.$$

- The conditional WEP is

$$P_e\left(\mathbf{c}_s^*\right) \lesssim \sum_{d=d_{\min}(\mathbf{C})}^{d_{\max}(\mathbf{C})} A_d\left(\mathbf{C}\right) 2^{\frac{d}{2}} Q\left(\sqrt{\frac{E_s d}{\sigma^2}}\right)$$

[5] T. Cover and etc, "Elements of information theory", Wiley Science, 1991.

Error Performance Analysis

Averaged Word Error Probability and Bit Error Probability

- The averaged WEP of the CPNC is

$$P_e = \frac{1}{2^{nR}} \sum_{\mathbf{c}_s \in \mathcal{C}} P_e(\mathbf{c}_s) \lesssim \sum_{d=d_{\min}(\mathbf{C})}^{d_{\max}(\mathbf{C})} A_d(\mathbf{C})\, 2^{\frac{d}{2}} Q\left(\sqrt{\frac{E_s d}{\sigma^2}}\right).$$

- The average BEP of the CPNC is

$$\begin{aligned} P_b &\lesssim \sum_{d=d_{\min}(\mathbf{C})}^{d_{\max}(\mathbf{C})} B_d(\mathbf{C})\, 2^{\frac{d}{2}} Q\left(\sqrt{\frac{E_s d}{\sigma^2}}\right) \\ &\le \frac{1}{2} \sum_{d=d_{\min}(\mathbf{C})}^{d_{\max}(\mathbf{C})} B_d(\mathbf{C}) \exp\left[-\frac{d}{2}\left(\frac{E_s}{\sigma^2} - \ln 2\right)\right] \end{aligned}$$

  where $B_d(\mathbf{C})$ is the average information weight w.r.t. all codewords of weight d.

- The BEP of the single-user case is

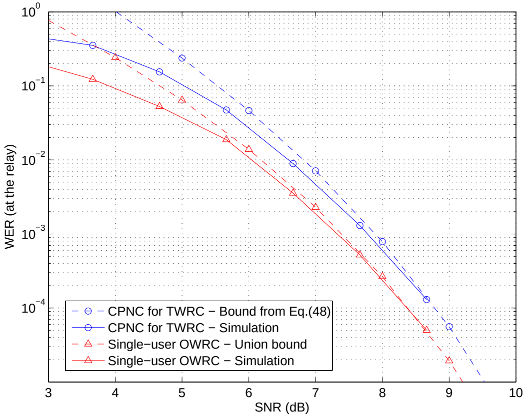
$$P_b^{SU} \le \frac{1}{2} \sum_d B_d(\mathbf{C}) \exp\left[-\frac{d}{2}\frac{E_s}{\sigma^2}\right].$$

- At high SNRs, the CPNC scheme relative to the single-user case has a performance degradation of approximately $\ln 2$ **(in linear scale)**.

Numerical Results

Hamming Coded PNC

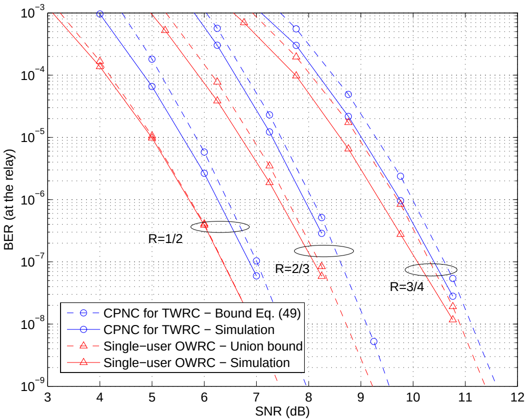
- We consider (7,4) Hamming coded PNC.
- Our analytical results match well with the numerical results.
- The SNR gap between the single-user performance and the two-user CPNC scheme is just under ln2 (in linear scale.)



Numerical Results

Convolutional Coded PNC

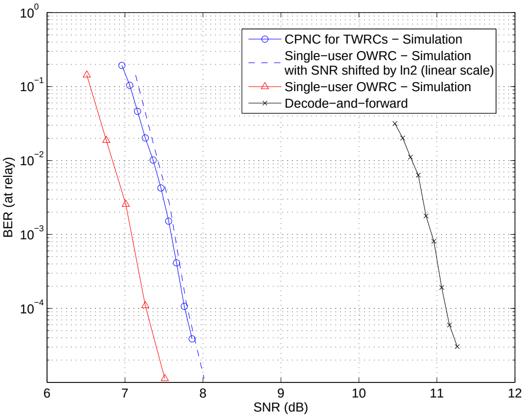
- We consider convolutional coded PNC with various coding rates.
- Our analytical results match well with the numerical results.
- The SNR gap between the single-user performance and the two-user CPNC scheme is just under ln2 (in linear scale.)



Numerical Results

Repeat-Accumulate (RA) Coded PNC

- We consider a RA coded PNC with code rate $R = 3/4$.
- The performance difference of ln2 is very clear.
- The CPNC significantly outperforms the complete-decoding based scheme by a few dBs.



Summary of Performance Analysis

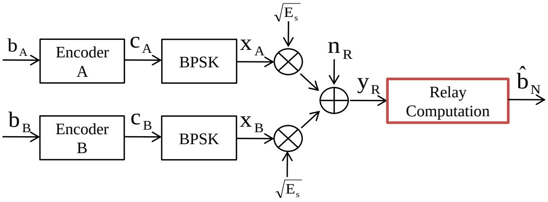
- Interesting to know that the SNR loss of the CPNC scheme relative to the single-user case is about $\ln 2$ (in linear scale) at a high SNR, regardless of the code rate.
- This means a conventional good code tends to perform well asymptotically in a CPNC scheme.

Further Questions

- How to design capacity achieving codes for CPNC schemes in the low SNR region?

Revisit System Model

Uplink Phase and Its Signals



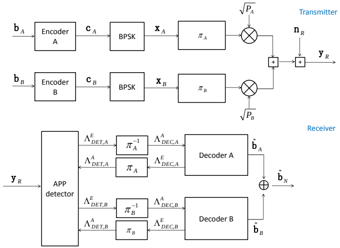
- $\mathbf{b}_A, \mathbf{b}_B \in \{0,1\}^k$, $\mathbf{c}_A, \mathbf{c}_B \in \{0,1\}^k$.
- *Superimposed codeword* $\mathbf{c}_s \triangleq \mathbf{c}_A + \mathbf{c}_B \in \{0,1,2\}^n$, $\mathbf{x}_s \triangleq \mathbf{x}_A + \mathbf{x}_B \in \{-2,0,2\}^n$.
- $\mathbf{y}_R$ is a noisy observation of $\mathbf{c}_s$.
- Relay needs to compute the network-coded (NC) message sequence

$$\mathbf{b}_N = \mathbf{b}_A \oplus \mathbf{b}_B \in \{0,1\}^k.$$

Revisit System Model

Relay Operations

- Relay computes $\mathbf{b}_N \in \{0,1\}^k$ from a noisy observation of $\mathbf{c}_s \in \{0,1,2\}^n$.
- Various network decoding approaches in [1]:
- Method 1: Complete-decode and forward.
  Relay decodes $\mathbf{b}_A$ and $\mathbf{b}_B$ first, then computes $\mathbf{b}_N = \mathbf{b}_A \oplus \mathbf{b}_B$



- Similar to CNC1 in [1], but iterative MUD/decoding brings a large gain.
- Multiplexing gain loss at high SNR.

[1] S. Zhang and S.-C. Liew, "Channel coding and decoding in a relay system operated with physical-layer network coding," *IEEE Jour. Select Area. Commun.*, vol. 27, pp. 788-796, June 2009.

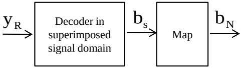
Revisit System Model

Relay Operation

- Method 2: Compute and forward.

  Relay decodes *superimposed message sequence* $\mathbf{b}_s \triangleq \mathbf{b}_A + \mathbf{b}_B$, from the noisy observation of the superimposed codeword $\mathbf{c}_s \triangleq \mathbf{c}_A + \mathbf{c}_B$.

  Then, map $\mathbf{b}_s = \mathbf{b}_A + \mathbf{b}_B \mapsto \mathbf{b}_N = \mathbf{b}_A \oplus \mathbf{b}_B$.

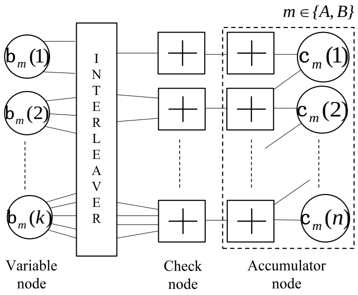


  - Similar to ACNC in [1], only forwards sufficient information.
  - Virtual encoder with ternary inputs and outputs needs to be defined.
  - For convolutional code, a super trellis or the product of the component code trellis will be useful [2].
  - For LDPC or RA code, an equivalent Tanner graph (ETG) defined over the superimposed messages.

[1] S. Zhang and S.-C. Liew, "Channel coding and decoding in a relay system operated with physical-layer network coding," *IEEE Jour. Select Area. Commun.*, vol. 27, pp. 788-796, June 2009.
[2] D. To and J. Choi, "Convolutional codes in two-way relay networks with physical-layer network coding," *IEEE Trans. Wireless Commun.*, vol. 9, no.9, pp. 2724-2729, Sept. 2010.

Equivalent Tanner Graph

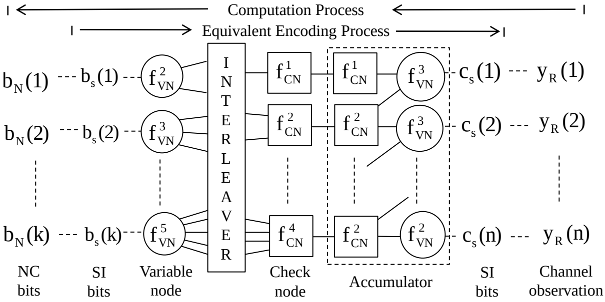
Tanner Graph for single user: Irregular Repeat-Accumulate (IRA) Code



- Message bits $b_m(t)$, $t = 1, \cdots, k$, are repeated η times, for $\eta = 2, 3, \cdots, d_v$.
- Variable node degree distribution is $\lambda_\eta : \lambda_\eta \geq 0, \sum_{\eta=2}^{d_v} \lambda_\eta = 1$.
- Interleaved sequence is encoded by a series of parity-check codes of degrees ψ, for $\psi = 1, 2, \cdots, d_c$.
- Check node degree distribution is $\rho_\psi : \rho_\psi \geq 0, \sum_{\psi=1}^{d_c} \rho_\psi = 1$.

Equivalent Tanner Graph

ETG for two users



- Input $\mathbf{b}_s = \mathbf{b}_A + \mathbf{b}_B$: ternary
- Output $\mathbf{c}_s = \mathbf{c}_A + \mathbf{c}_B$: ternary
- How to define/exchange/update extrinsic information or log-likelihood ratios?
- For code design, how to model the distribution of the *a priori* information for density evolution or EXIT chart functions?

Message Updates

- Intrinsic information from $\mathbf{y}_R$
  - For jth superimposed coded symbol

$$p_0(j) = \mathrm{P}(c_s(j) = 0 | y_R(j))$$
$$p_1(j) = \mathrm{P}(c_s(j) = 1 | y_R(j))$$
$$p_2(j) = \mathrm{P}(c_s(j) = 2 | y_R(j))$$

  - Represented in log-likelihood ratio (LLR) form

$$\mathrm{LLR}(c_s(j)|y_R(j)) = [\Lambda(j), \Omega(j)]$$

Primary LLR:

$$\Lambda(j) \triangleq \log\left(\frac{p_0(j)+p_2(j)}{p_1(j)}\right)$$

Secondary LLR:

$$\Omega(j) \triangleq \log\left(\frac{p_0(j)}{p_2(j)}\right)$$

  - Primary LLR is the LLR of the network coded bits.

$$\Lambda(j) \triangleq \log\left(\frac{p_0(j)+p_2(j)}{p_1(j)}\right) = \log\left(\frac{\mathrm{P}(c_N(j)=0|y_R(j))}{\mathrm{P}(c_N(j)=1|y_R(j))}\right)$$

Message Updates

Check Node Update Rule

- Update function f_{CN}^2 for degree $\psi = 2$.

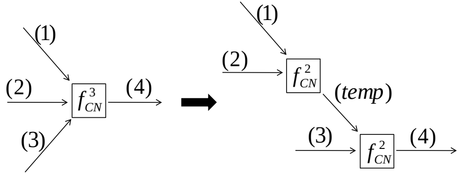
$$\Lambda_Q^{(3)} = \log\left(\frac{1+\exp\left(\Lambda_P^{(1)}\right)\exp\left(\Lambda_P^{(2)}\right)}{\exp\left(\Lambda_P^{(1)}\right)+\exp\left(\Lambda_P^{(2)}\right)}\right)$$

$$\Omega_Q^{(3)} = \log\left(\frac{1+\exp\left(\Omega_P^{(1)}\right)\exp\left(\Omega_P^{(2)}\right)+K_{\text{CN}}}{\exp\left(\Omega_P^{(2)}\right)+\exp\left(\Omega_P^{(1)}\right)+K_{\text{CN}}}\right)$$

P: *a priori*, Q: extrinsic

$$K_{\text{CN}} = \frac{\left[1+\exp\left(\Omega_P^{(1)}\right)\right]\left[1+\exp\left(\Omega_P^{(2)}\right)\right]}{2\exp\left(\Lambda_P^{(1)}\right)\exp\left(\Lambda_P^{(2)}\right)}.$$

- For CN degree $\psi > 2$, successively using f_{CN}^2 to update the extrinsic



Message Updates

Variable Node Updating Rule

- Variable node (VN) updating rule for degree η is

$$\Lambda_Q^{(l)} = (\eta - 2)\log 2 + \sum_{l'=1, l'\neq l}^{\eta} \Lambda_P^{(l')} + K_{\mathrm{VN}}$$

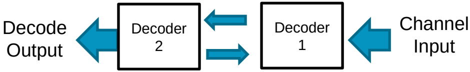
$$\Omega_Q^{(l)} = \sum_{l'=1, l'\neq l}^{\eta} \Omega_P^{(l')}$$

where

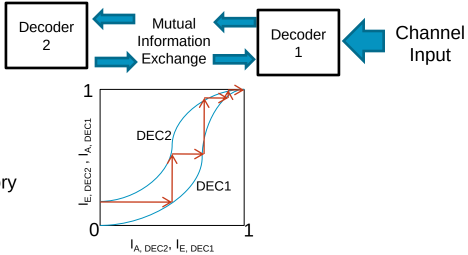
$$K_{\mathrm{VN}} = \log\left(\frac{1+\prod\limits_{l'=1, l'\neq l}^{\eta} \exp\left(\Omega_P^{(l')}\right)}{\prod\limits_{l'=1, l'\neq l}^{\eta}\left(1+\exp\left(\Omega_P^{(l')}\right)\right)}\right)$$

EXIT Chart

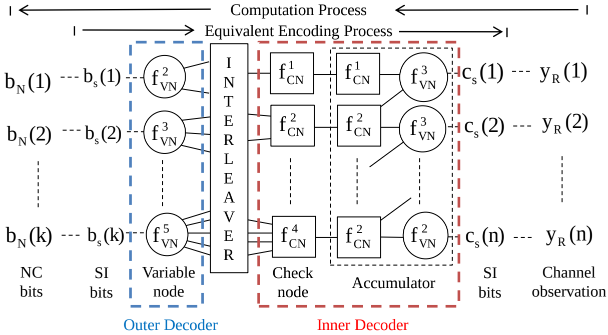
- Extrinsic information transfer (EXIT) chart (S. ten Brink 1999, 2003, 2004).
- To illustrate the iteration decoding (mutual information) trajectory.
- To visualize the convergence of the iterative decoding.
- With curve fitting technique, EXIT chart can be used for code design and threshold analysis.



EXIT chart and decoding trajectory



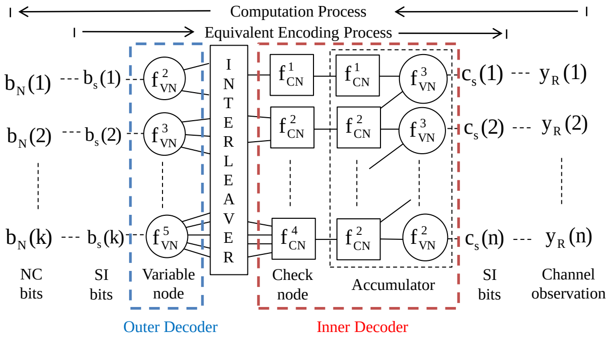
EXIT Chart



- Inner decoder: CN-ACC decoder; Outer decoder: VN decoder.
- Exchange both primary and secondary LLRs.
- Recall: Primary LLR Λ linked with NC message b_N.
- Tracking only primary LLR: $I_A = I\left(\mathbf{b}_N; \Lambda_P\right)$, $I_E = I\left(\mathbf{b}_N; \Lambda_Q\right)$.

EXIT Chart

EXIT Functions



- Inner decoder:

$$I_E = T_{\text{Inner}}\left(I_A, \mathbb{P}(\Omega_P), \rho_\psi, E_s\right)$$

- Outer decoder:

$$I_E = T_{\text{Outer}}\left(I_A, \mathbb{P}(\Omega_P), \lambda_\eta\right)$$

where $I_A = I\left(\mathbf{b}_N; \Lambda_P\right)$.

EXIT Chart

- EXIT functions contains both primary LLR Λ_P and secondary LLR Ω_P.
- Primary LLR approaches a consistent Gaussian-like distribution with its mean equal to half of its variance.

$$\Lambda_P = (\sigma_\Lambda^2/2)x_N + n_\Lambda$$

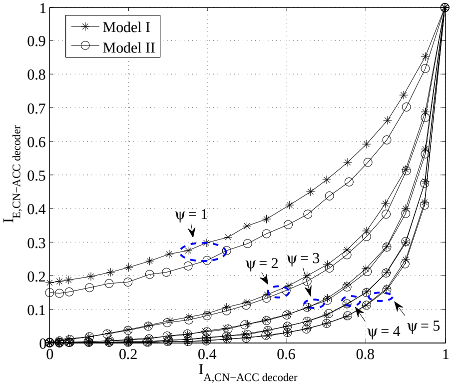
- Secondary LLR $\Omega(j) \triangleq \log\left(p_0(j)/p_2(j)\right)$ resembles a combination of a Gaussian-like distribution and an impulse at zero.
- Two models to bound the EXIT functions.
  - Model I: Assume perfect secondary LLR information

$$\dot{\Omega}_P = \begin{cases} +\infty & \text{if } b_s = 0 \\ 0 & \text{if } b_s = 1 \\ -\infty & \text{if } b_s = 2 \end{cases}$$

  - Model II: Assume no *a priori* information on the secondary LLR Ω_P, i.e., $\ddot{\Omega}_P = 0$.

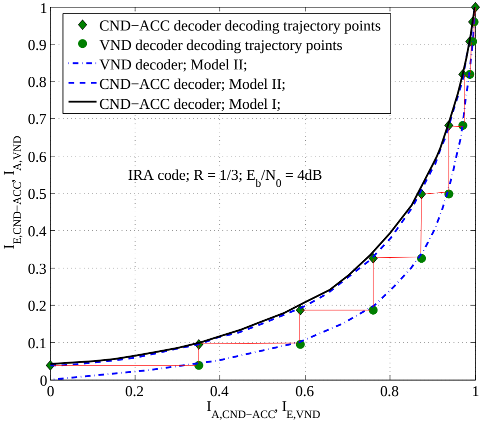
$$T_{\text{Inner}}\left(I_A, \mathbb{P}(\dot{\Omega}_P), \rho_\psi, E_s\right) \geq I_E \geq T_{\text{Inner}}\left(I_A, \mathbb{P}(\ddot{\Omega}_P), \rho_\psi, E_s\right)$$

EXIT Chart



- Model I always gives higher mutual information than Model II.
- For CN degrees higher than 2, no much performance difference.

EXIT Chart

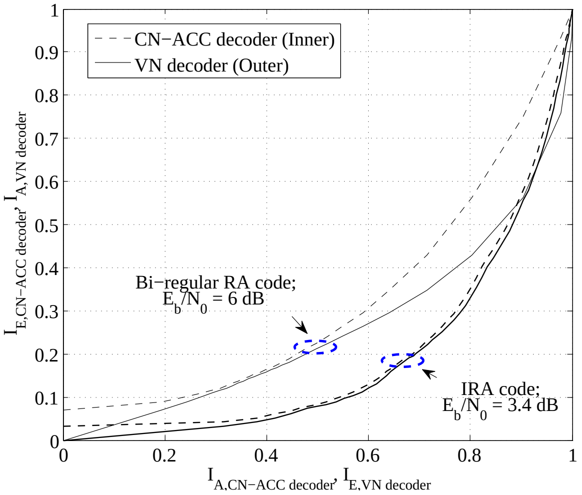


- Capacity approaching codes usually have higher degree CNs.
- Using Model II will be sufficient in our code design.
- Model II is a lower bound and it guarantees the convergence.

Code Design

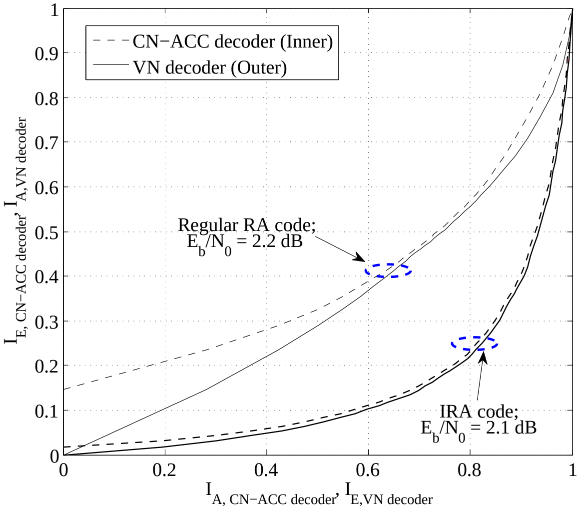
Scheme	Code Type	CN		VN	
		ψ	ρ	η	λ
Physical-layer network coded (PNC)	**Regular** $R = 1/3$	1	1	3	1
	Bi-regular $R = 3/4$	1	0.2288	4	1
		3	0.5424		
		5	0.2288		
	Irregular $R = 1/3$	1	0.30	2	0.1542
		3	0.70	3	0.3353
				7	0.1375
				8	0.2237
				21	0.1493
	Irregular $R = 3/4$	1	0.20	2	0.3221
		5	0.80	3	0.3297
				6	0.2272
				7	0.478
				31	0.732
Based on complete decoding	**Irregular** $R = 1/3$	1	0.20	3	0.4963
		3	0.80	4	0.1144
				9	0.829
				10	0.2004
				29	0.870
				30	0.190
	Irregular $R = 3/4$	1	0.10	2	0.2672
		3	0.90	3	0.5915
				7	0.493
				8	0.610
				19	0.310

Code Design



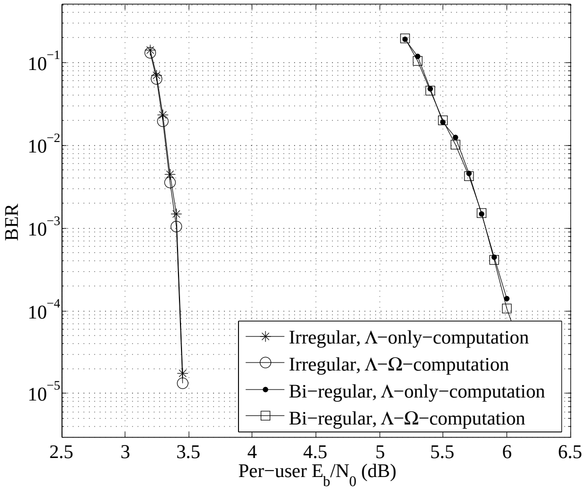
$R = 3/4$

Code Design



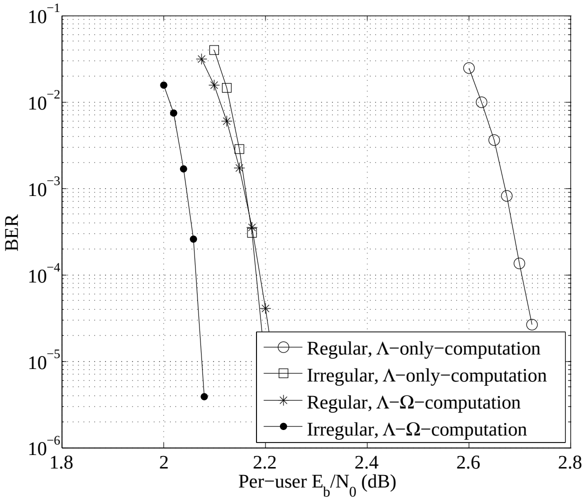
$R = 1/3$

Code Design



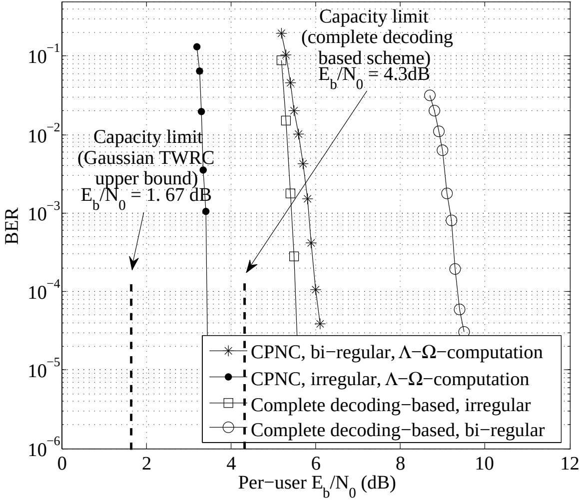
$$R = 3/4$$

Code Design



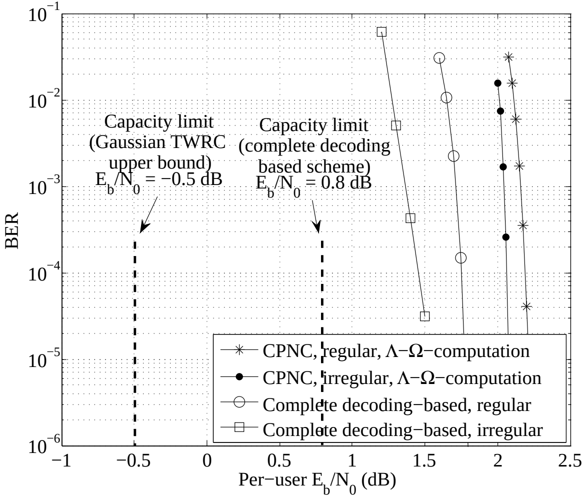
$R = 1/3$

Code Design



$R = 3/4$

Code Design



$R = 1/3$

Summary

- We investigated performance and design of channel coded PNC scheme.
- We proposed a method to compute the pairwise distance spectrum of a CPNC scheme, and asymptotically tight WEP and BEP bounds were derived.
- The SNR loss of the CPNC scheme relative to the single-user case is about $\ln 2$ (in linear scale) at a high SNR, regardless of the code rate.
- Proposed ETG and general message updating rules.
- Present two models to bound the EXIT functions for convergency analysis and code designs.
- Design capacity approaching IRA coded PNC schemes.

Further Work

- Information-theorectic issues: Capacity?
- Practical design issues: Synchronization, channel estimation, power/phase controls?
- Extentions: Higher level modulation (lattice coding), fading channels, multihop TWRC, MIMO TWRC, multiway, etc ?

Acknowledgement

Tao Huang, University of New South Wales

Tom Yang, CSIRO ICT Centre

Ingmar Land, ITR, University of South Australia