Intuitive Understanding of Iterative Decoding

Fangning Hu

JACOBS UNIVERSITY

School of Engineering and Science
Jacobs University Bremen, Germany

Summer Academy 2007, Jacobs University Bremen, Germany

Outline

- Basic Concept of Linear Codes
- Principle of the Iterative Decoding
- Analysis in the Continuous Real Field
- More Insight of Analog Product Codes
- Summary

Outline

- Basic Concept of Linear Codes
- Principle of the Iterative Decoding
- Analysis in the Continuous Real Field
- More Insight of Analog Product Codes
- Summary

Outline

- Basic Concept of Linear Codes
- Principle of the Iterative Decoding
- Analysis in the Continuous Real Field
- More Insight of Analog Product Codes
- Summary

Outline

- Basic Concept of Linear Codes
- Principle of the Iterative Decoding
- Analysis in the Continuous Real Field
- More Insight of Analog Product Codes
- Summary

Outline

- Basic Concept of Linear Codes
- Principle of the Iterative Decoding
- Analysis in the Continuous Real Field
- More Insight of Analog Product Codes
- Summary

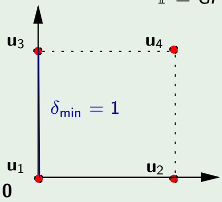
Definition

A **linear code** $\mathcal{X}(N, K)$ is a K-dimensional subspace of $\mathbb{F}^N$.

$\mathcal{X} = \{\mathbf{x} | \mathbf{x} = \mathbf{G}^T\mathbf{u}, \mathbf{u} \in \mathbb{F}^K, \mathbf{x} \in \mathbb{F}^N\}, \quad \mathcal{X} = \{\mathbf{x} | \mathbf{H}\mathbf{x} = \mathbf{0}\}$

Example

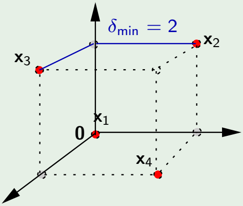
$\mathbb{F} = GF(2) = \{0, 1\}$



$\mathbf{u} \in \mathbb{F}^2$

$\mathbf{x} = \mathbf{G}^T\mathbf{u}$

$$\mathbf{G}^T = \begin{matrix} 1 & 0 \\ 0 & 1 \\ 1 & 1 \end{matrix}$$

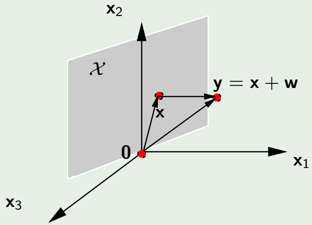


$\mathbf{x} \in \mathcal{X}(3, 2) \subset \mathbb{F}^3$

Analog Codes

$$\mathbb{F} = \mathbb{R}$$

Example



$\mathbf{x} \in \mathcal{X} \subset \mathbb{R}^3$

$$\begin{aligned} \mathcal{X}(3,2) &= \{\mathbf{x} | \mathbf{H}\mathbf{x} = \mathbf{0}\} \\ \mathbf{H} = [1,0,1] &\Leftrightarrow x_1 + x_3 = 0 \end{aligned}$$

Iterative Decoding

- MAP: $\hat{\mathbf{x}} = \arg\max_{\mathbf{x}\in\mathcal{X}} p(\mathbf{x}|\mathbf{y}) \propto \arg\max_{\mathbf{x}\in\mathcal{X}} p(\mathbf{y}|\mathbf{x})p(\mathbf{x})$
- ML Decoding: $\hat{\mathbf{x}} = \arg\max_{\mathbf{x}\in\mathcal{X}} p(\mathbf{y}|\mathbf{x})$
- For AWGN channel: $\hat{\mathbf{x}} = \arg\min_{\mathbf{x}\in\mathcal{X}} ||\mathbf{x}-\mathbf{y}||^2$

$\mathbb{F} = \mathbb{R}$

Least-Squares Solution: $\hat{\mathbf{x}} = \arg\max_{\mathbf{x}\in\mathcal{X}} p(\mathbf{x}|\mathbf{y}) = \Pi_{\mathcal{X}} \cdot \mathbf{y}$

$\mathbb{F} = GF(q)$

- $p(\mathbf{y}|\mathbf{x})p(\mathbf{x}) = C \prod_n p(y_n|x_n) \prod_m [\mathbf{h}_m \mathbf{x} = 0]$
- Iterative decoding: $p(x_n|\mathbf{y}) = \sum_{\sim x_n} p(\mathbf{x}|\mathbf{y})$

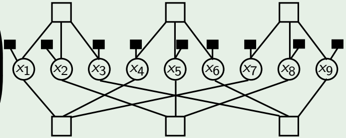
An Example: The Factor Graph Representation

- A linear code: $\mathcal{X}(9,3) = \{\mathbf{x} | \mathbf{Hx} = \mathbf{0}, \mathbf{x} \in \mathbb{F}^n\}, h_{ij} \in \mathbb{F}$
- $y_n = x_n + w_n, \quad \hat{x}_n = \max_{x_n \in \mathbb{F}} p(x_n|\mathbf{y})$
- Initialize: $f_{y_n}(x_n) = p(y_n|x_n) \sim p(x_n|y_n) = \mathcal{N}(y_n, \sigma^2)$

  Check Equation: $f_1^{(1)} = [h_{11}x_1 + h_{12}x_2 + h_{13}x_3 = 0]$

Example

$$\begin{pmatrix} h_{11} & h_{12} & h_{13} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & h_{24} & h_{25} & h_{26} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & h_{37} & h_{38} & h_{39} \\ h_{41} & 0 & 0 & h_{44} & 0 & 0 & h_{47} & 0 & 0 \\ 0 & h_{52} & 0 & 0 & h_{55} & 0 & 0 & h_{58} & 0 \\ 0 & 0 & h_{63} & 0 & 0 & h_{66} & 0 & 0 & h_{69} \end{pmatrix}$$



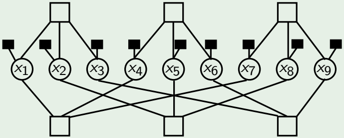
An Example: The Factor Graph Representation

- A linear code: $\mathcal{X}(9,3) = \{\mathbf{x} | \mathbf{Hx} = \mathbf{0}, \mathbf{x} \in \mathbb{F}^n\}, h_{ij} \in \mathbb{F}$
- $y_n = x_n + w_n, \quad \hat{x}_n = \max_{x_n \in \mathbb{F}} p(x_n | \mathbf{y})$
- Initialize: $f_{y_n}(x_n) = p(y_n|x_n) \sim p(x_n|y_n) = \mathcal{N}(y_n, \sigma^2)$

  Check Equation: $f_1^{(1)} = [h_{11}x_1 + h_{12}x_2 + h_{13}x_3 = 0]$

Example

$$\begin{pmatrix} h_{11} & h_{12} & h_{13} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & h_{24} & h_{25} & h_{26} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & h_{37} & h_{38} & h_{39} \\ h_{41} & 0 & 0 & h_{44} & 0 & 0 & h_{47} & 0 & 0 \\ 0 & h_{52} & 0 & 0 & h_{55} & 0 & 0 & h_{58} & 0 \\ 0 & 0 & h_{63} & 0 & 0 & h_{66} & 0 & 0 & h_{69} \end{pmatrix}$$



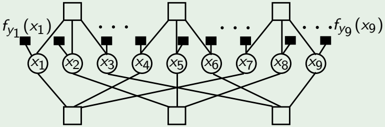
An Example: The Factor Graph Representation

- A linear code: $\mathcal{X}(9,3) = \{\mathbf{x}|\mathbf{H}\mathbf{x} = \mathbf{0}, \mathbf{x} \in \mathbb{F}^n\}, h_{ij} \in \mathbb{F}$
- $y_n = x_n + w_n, \quad \hat{x}_n = \max_{x_n \in \mathbb{F}} p(x_n|\mathbf{y})$
- Initialize: $f_{y_n}(x_n) = p(y_n|x_n) \sim p(x_n|y_n) = \mathcal{N}(y_n, \sigma^2)$

  Check Equation: $f_1^{(1)} = [h_{11}x_1 + h_{12}x_2 + h_{13}x_3 = 0]$

Example

$$\begin{pmatrix} h_{11} & h_{12} & h_{13} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & h_{24} & h_{25} & h_{26} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & h_{37} & h_{38} & h_{39} \\ h_{41} & 0 & 0 & h_{44} & 0 & 0 & h_{47} & 0 & 0 \\ 0 & h_{52} & 0 & 0 & h_{55} & 0 & 0 & h_{58} & 0 \\ 0 & 0 & h_{63} & 0 & 0 & h_{66} & 0 & 0 & h_{69} \end{pmatrix}$$



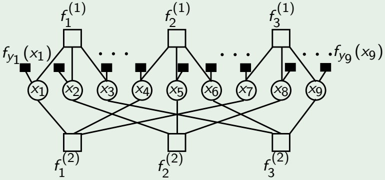
An Example: The Factor Graph Representation

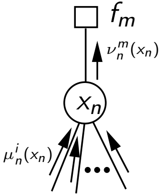
- A linear code: $\mathcal{X}(9,3) = \{\mathbf{x} | \mathbf{Hx} = \mathbf{0}, \mathbf{x} \in \mathbb{F}^n\}, h_{ij} \in \mathbb{F}$
- $y_n = x_n + w_n, \quad \hat{x}_n = \max_{x_n \in \mathbb{F}} p(x_n | \mathbf{y})$
- Initialize: $f_{y_n}(x_n) = p(y_n | x_n) \sim p(x_n | y_n) = \mathcal{N}(y_n, \sigma^2)$

  Check Equation: $f_1^{(1)} = [h_{11}x_1 + h_{12}x_2 + h_{13}x_3 = 0]$

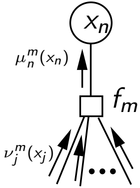
Example

$$\begin{pmatrix} h_{11} & h_{12} & h_{13} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & h_{24} & h_{25} & h_{26} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & h_{37} & h_{38} & h_{39} \\ h_{41} & 0 & 0 & h_{44} & 0 & 0 & h_{47} & 0 & 0 \\ 0 & h_{52} & 0 & 0 & h_{55} & 0 & 0 & h_{58} & 0 \\ 0 & 0 & h_{63} & 0 & 0 & h_{66} & 0 & 0 & h_{69} \end{pmatrix}$$





$$\nu_n^m(x_n) = \prod_{i \neq m} \mu_n^i(x_n)$$



$$\mu_n^m(x_n) = \sum_{\sim\{x_n\}} f_m \prod_{j \neq n} \nu_j^m(x_j)$$

Binary case ($L = \log \frac{p_0}{p_1}$):

$$\begin{aligned} \mathrm{VAR}(L_1, L_2) &= L_1 + L_2 \\ \mathrm{CHK}(L_1, L_2) &= 2\tanh^{-1}(\tanh(L_1/2)\tanh(L_2/2)) \,. \end{aligned}$$

GF field:

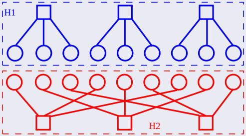
- When the graph contains no cycle:
  $\hat{x}_n = \max_{x_n \in \mathbb{F}} p(x_n|\mathbf{y}) \Leftrightarrow \hat{\mathbf{x}} = \max_{\mathbf{x} \in \mathcal{X}} p(\mathbf{x}|\mathbf{y})$ (ML Decoding).
- When a graph contains cycles: No guarantee regarding the decoding performance; Short cycles (especially cycles of length 4) limit the decoding performance.

Continuous Real Field:

- Always converges to the least-squares solution $\hat{\mathbf{x}} = \max_{\mathbf{x} \in \mathcal{X}} p(\mathbf{x}|\mathbf{y})$ even there are short cycles.
- Can be intuitively illustrated in Euclidean space.

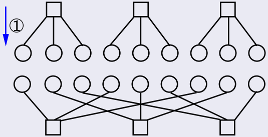
Two Concatenation Constituent Codes

$$\begin{pmatrix} 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \\ - & - & - & - & - & - & - & - & - \\ 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 \end{pmatrix}$$

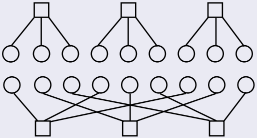


The Scheduling

Serial

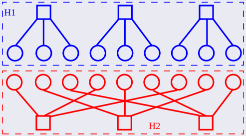


Parallel



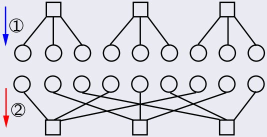
Two Concatenation Constituent Codes

$$\begin{pmatrix} 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \\ - & - & - & - & - & - & - & - & - \\ 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 \end{pmatrix}$$

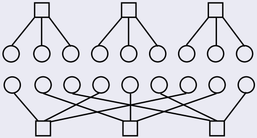


The Scheduling

Serial

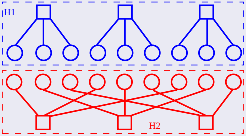


Parallel



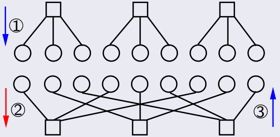
Two Concatenation Constituent Codes

$$\begin{pmatrix} 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \\ \hline 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 \end{pmatrix}$$

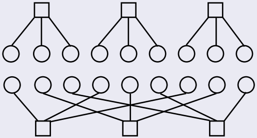


The Scheduling

Serial

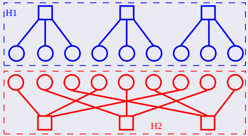


Parallel



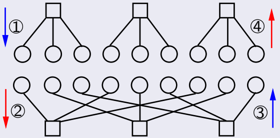
Two Concatenation Constituent Codes

$$\begin{pmatrix} 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \\ - & - & - & - & - & - & - & - & - \\ 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 \end{pmatrix}$$

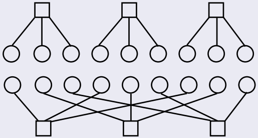


The Scheduling

Serial

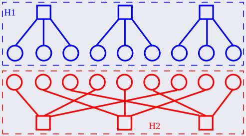


Parallel



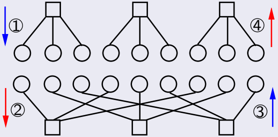
Two Concatenation Constituent Codes

$$\begin{pmatrix} 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \\ \hline 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 \end{pmatrix}$$

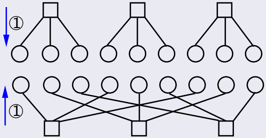


The Scheduling

Serial

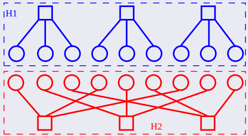


Parallel



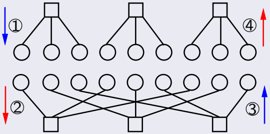
Two Concatenation Constituent Codes

$$\begin{pmatrix} 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \\ \hline 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 \end{pmatrix}$$

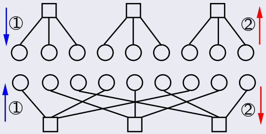


The Scheduling

Serial



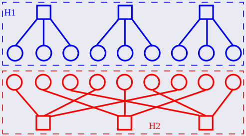
Parallel



Two Decompositions

The 1st decomposition

$$\begin{pmatrix} 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \\ - & - & - & - & - & - & - & - & - \\ 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 \end{pmatrix}$$



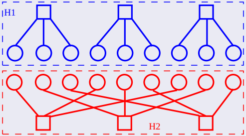
The 2nd decomposition

$$\begin{pmatrix} 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \\ - & - & - & - & - & - & - & - & - \\ 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \end{pmatrix}$$

Two Decompositions

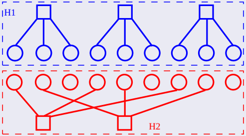
The 1st decomposition

$$\begin{pmatrix} 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \\ - & - & - & - & - & - & - & - & - \\ 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 \end{pmatrix}$$



The 2nd decomposition

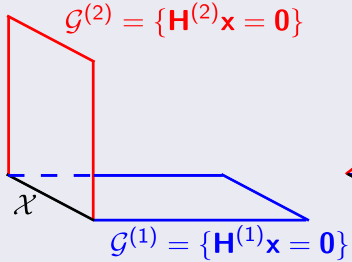
$$\begin{pmatrix} 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \\ - & - & - & - & - & - & - & - & - \\ 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \end{pmatrix}$$



The Geometric Properties of Two Decompositions

- $\mathcal{X} = \mathcal{G}^{(1)} \cap \mathcal{G}^{(2)}$
- $\sphericalangle(\mathbf{y}_1^{\perp}, \mathbf{y}_2^{\perp}) = \sphericalangle(\mathcal{G}^{(1)}, \mathcal{G}^{(2)})$
- $< \mathbf{y}_1^{\perp}, \mathbf{y}_2^{\perp} >= 0 \Leftrightarrow \mathcal{G}^{(1)} \perp \mathcal{G}^{(2)}$

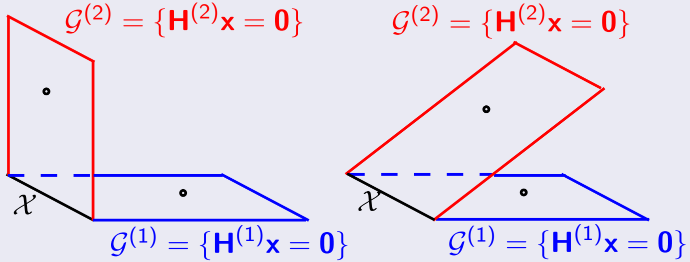
The geometric illustration of the decompositions



The Geometric Properties of Two Decompositions

- $\mathcal{X} = \mathcal{G}^{(1)} \cap \mathcal{G}^{(2)}$
- $\sphericalangle(\mathbf{y}_1^{\perp}, \mathbf{y}_2^{\perp}) = \sphericalangle(\mathcal{G}^{(1)}, \mathcal{G}^{(2)})$
- $< \mathbf{y}_1^{\perp}, \mathbf{y}_2^{\perp} >= 0 \Leftrightarrow \mathcal{G}^{(1)} \perp \mathcal{G}^{(2)}$

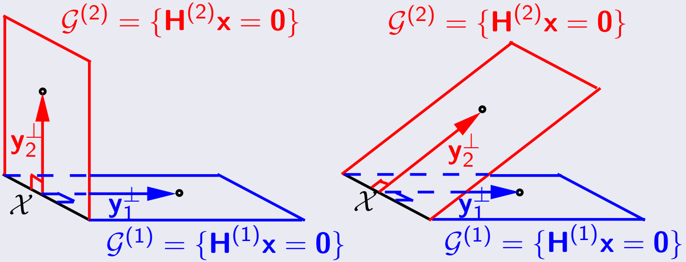
The geometric illustration of the decompositions



The Geometric Properties of Two Decompositions

- $\mathcal{X} = \mathcal{G}^{(1)} \cap \mathcal{G}^{(2)}$
- $\sphericalangle(\mathbf{y}_1^{\perp}, \mathbf{y}_2^{\perp}) = \sphericalangle(\mathcal{G}^{(1)}, \mathcal{G}^{(2)})$
- $< \mathbf{y}_1^{\perp}, \mathbf{y}_2^{\perp} >= 0 \Leftrightarrow \mathcal{G}^{(1)} \perp \mathcal{G}^{(2)}$

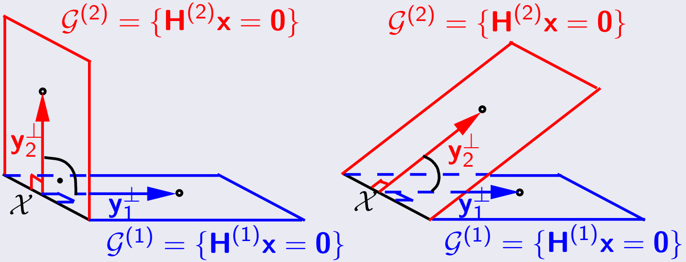
The geometric illustration of the decompositions



The Geometric Properties of Two Decompositions

- $\mathcal{X} = \mathcal{G}^{(1)} \cap \mathcal{G}^{(2)}$
- $\sphericalangle(\mathbf{y}_1^{\perp}, \mathbf{y}_2^{\perp}) = \sphericalangle(\mathcal{G}^{(1)}, \mathcal{G}^{(2)})$
- $< \mathbf{y}_1^{\perp}, \mathbf{y}_2^{\perp} >= 0 \Leftrightarrow \mathcal{G}^{(1)} \perp \mathcal{G}^{(2)}$

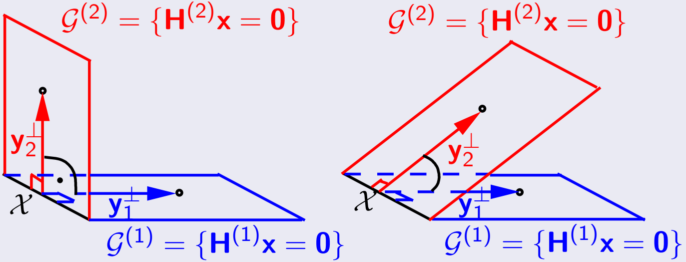
The geometric illustration of the decompositions



The Geometric Properties of Two Decompositions

- $\mathcal{X} = \mathcal{G}^{(1)} \cap \mathcal{G}^{(2)}$
- $\sphericalangle(\mathbf{y}_1^{\perp}, \mathbf{y}_2^{\perp}) = \sphericalangle(\mathcal{G}^{(1)}, \mathcal{G}^{(2)})$
- $< \mathbf{y}_1^{\perp}, \mathbf{y}_2^{\perp} >= 0 \Leftrightarrow \mathcal{G}^{(1)} \perp \mathcal{G}^{(2)}$

The geometric illustration of the decompositions



The Algorithm Derivation

Facts

- Check-to-variable: $g_1(x_1') = (g_2 \circledast g_3)(-x_1'), x_1' = h_1 x_1$
- Variable-to-check: $g_1(x_1') = g_2(x_1') \cdot g_3(x_1')$
- All the densities are preserved to be Gaussian
- Variance converges to a fixed value (by simulation)
- Tracing the means is sufficient

Check-to-variable

$\mu_1 = -\frac{1}{h_1}(h_2\mu_2 + h_3\mu_3)$
$\sigma_2^2 = \frac{1}{h_1^2}(h_2^2\sigma_2^2 + h_3^2\sigma_3^2)$

Variable-to-check

$\sigma_1^{-2} = \sigma_2^{-2} + \sigma_3^{-2}$
$\mu_1 = \frac{\mu_2 + w\mu_3}{1+w}, w = \sigma_2^2/\sigma_3^2$

The Algorithm Derivation

Facts

- Check-to-variable: $g_1(x_1') = (g_2 \circledast g_3)(-x_1'), x_1' = h_1 x_1$
- Variable-to-check: $g_1(x_1') = g_2(x_1') \cdot g_3(x_1')$
- All the densities are preserved to be Gaussian
- Variance converges to a fixed value (by simulation)
- Tracing the means is sufficient

Check-to-variable

$\mu_1 = -\frac{1}{h_1}(h_2\mu_2 + h_3\mu_3)$

$\sigma_2^2 = \frac{1}{h_1^2}(h_2^2\sigma_2^2 + h_3^2\sigma_3^2)$

Variable-to-check

$\sigma_1^{-2} = \sigma_2^{-2} + \sigma_3^{-2}$

$\mu_1 = \frac{\mu_2 + w\mu_3}{1+w}, w = \sigma_2^2/\sigma_3^2$

The Algorithm Derivation

Facts

- Check-to-variable: $g_1(x_1') = (g_2 \circledast g_3)(-x_1'), x_1' = h_1 x_1$
- Variable-to-check: $g_1(x_1') = g_2(x_1') \cdot g_3(x_1')$
- All the densities are preserved to be Gaussian
- Variance converges to a fixed value (by simulation)
- Tracing the means is sufficient

Check-to-variable

$$\mu_1 = -\frac{1}{h_1}(h_2\mu_2 + h_3\mu_3)$$
$$\sigma_2^2 = \frac{1}{h_1^2}(h_2^2\sigma_2^2 + h_3^2\sigma_3^2)$$

Variable-to-check

$$\sigma_1^{-2} = \sigma_2^{-2} + \sigma_3^{-2}$$
$$\mu_1 = \frac{\mu_2 + w\mu_3}{1+w}, w = \sigma_2^2/\sigma_3^2$$

The Algorithm Derivation

Facts

- Check-to-variable: $g_1(x_1') = (g_2 \circledast g_3)(-x_1'), x_1' = h_1 x_1$
- Variable-to-check: $g_1(x_1') = g_2(x_1') \cdot g_3(x_1')$
- All the densities are preserved to be Gaussian
- Variance converges to a fixed value (by simulation)
- Tracing the means is sufficient

Check-to-variable

$$\mu_1 = -\frac{1}{h_1}(h_2\mu_2 + h_3\mu_3)$$
$$\sigma_2^2 = \frac{1}{h_1^2}(h_2^2\sigma_2^2 + h_3^2\sigma_3^2)$$

Variable-to-check

$$\sigma_1^{-2} = \sigma_2^{-2} + \sigma_3^{-2}$$
$$\mu_1 = \frac{\mu_2 + w\mu_3}{1+w}, w = \sigma_2^2/\sigma_3^2$$

The Algorithm Derivation

Facts

- Check-to-variable: $g_1(x_1') = (g_2 \circledast g_3)(-x_1'), x_1' = h_1 x_1$
- Variable-to-check: $g_1(x_1') = g_2(x_1') \cdot g_3(x_1')$
- All the densities are preserved to be Gaussian
- Variance converges to a fixed value (by simulation)
- Tracing the means is sufficient

Check-to-variable

$$\mu_1 = -\frac{1}{h_1}(h_2\mu_2 + h_3\mu_3)$$
$$\sigma_2^2 = \frac{1}{h_1^2}(h_2^2\sigma_2^2 + h_3^2\sigma_3^2)$$

Variable-to-check

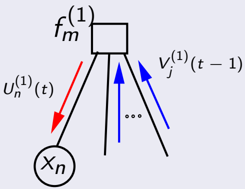
$$\sigma_1^{-2} = \sigma_2^{-2} + \sigma_3^{-2}$$
$$\mu_1 = \frac{\mu_2 + w\mu_3}{1+w}, w = \sigma_2^2/\sigma_3^2$$

The Overall Algorithm

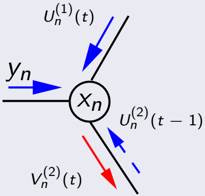
Initialization: $V_n^{(1)}(0) = V_n^{(2)}(0) = y_n, v_n^{(1)}(0) = v_n^{(2)}(0) = \sigma^2$

MVE for the serial schedule

1) Update $U_n^{(1)}(t)$



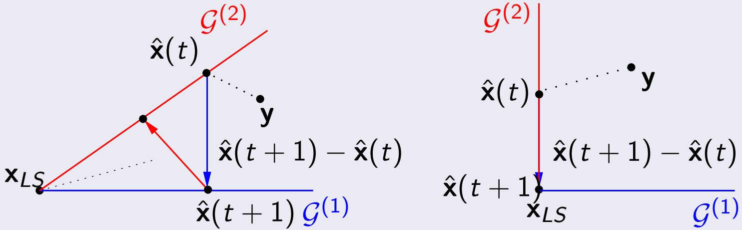
2) Update $V_n^{(2)}(t)$



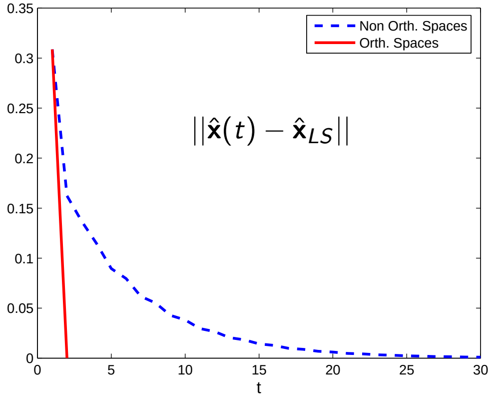
The estimate: $\hat{x}_n(t) = \frac{y_n + w_n^{(1)}(t) U_n^{(1)}(t) + w_n^{(2)}(t-1) U_n^{(2)}(t-1)}{1 + w_n^{(1)}(t) + w_n^{(2)}(t-1)}$.

The Geometric Illustration of the MVE

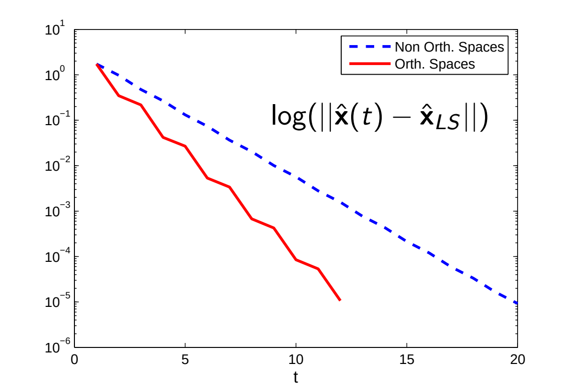
The Geometric Illustration of the MVE



Comparing convergence speed of a (25, 16) analog product code



Comparing convergence speed of a (7, 4) analog compound code



$$\mathbf{u} = [x_{1,1}, x_{1,2}, \ldots, x_{k,k}] \mapsto \mathbf{X}$$

$$\mathbf{X} = \left[\begin{array}{ccc|c} x_{1,1} & \cdots & x_{1,k} & x_{1,k+1} \\ \vdots & \ddots & \vdots & \vdots \\ x_{k,1} & \cdots & x_{k,k} & x_{k,k+1} \\ \hline x_{k+1,1} & \cdots & x_{k+1,k} & x_{k+1,k+1} \end{array}\right]$$

$$\mathbf{x} = \text{vec}\,(\mathbf{X}) = [x_{1,1}, x_{2,1}, \ldots, x_{k,1}, x_{k+1,1}, \ldots, \ldots, x_{k,k+1}, x_{k+1,k+1}]^T$$

$$\sum_i x_{ij} = 0 \rightarrow \mathbf{H}_1\mathbf{x} = \mathbf{0}, \quad \mathbf{H}_1 = (\mathbf{I} \otimes \mathbf{1}^T).$$

$$\sum_j x_{ij} = 0 \rightarrow \mathbf{H}_2\mathbf{x} = \mathbf{0}, \quad \mathbf{H}_2 = (\mathbf{1}^T \otimes \mathbf{I}).$$

A Slightly Modified Version

Lemma

- *The weight* $w_{i,j}^{(m)}(t)$ *is constant:*
  $w_{i,j}^{(m)}(t) = \frac{1}{n-1} = \frac{1}{k}, \quad m = 1, 2, \quad i, j = 1, \ldots, n\,.$
- $\mathbf{U}^{(1)}(t) = -\bar{\mathbf{I}}\mathbf{V}^{(1)}(t)$ *and* $\mathbf{U}^{(2)}(t) = -\mathbf{V}^{(2)}(t)\bar{\mathbf{I}}$

Modification in the Variable-to-Check Operation:

$$y_{i,j} \rightarrow V_{i,j}^{(2)}(t-1): \quad V_{i,j}^{(1)}(t) = \frac{V_{i,j}^{(2)}(t-1) + w_{i,j}^{(2)}(t-1)U_{i,j}^{(2)}(t-1)}{1 + w_{i,j}^{(2)}(t-1)}$$

Decoding Algorithm for Analog Product Codes

$$\mathbf{V}(0) = \mathbf{Y}, \quad \mathbf{V}(t) = \mathbf{Y}(t)$$

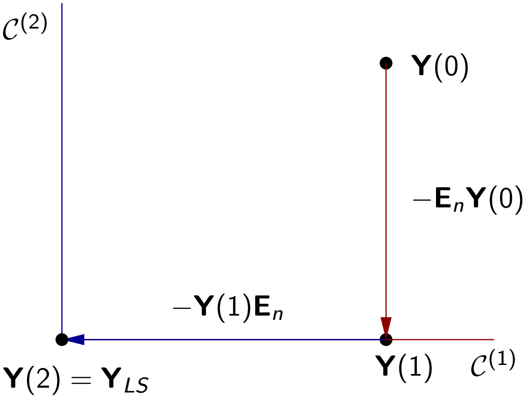
Serial Decoding

$$\begin{aligned} \mathbf{Y}(t) &= \mathbf{Y}(t-1) - \beta(t-1)\mathbf{Y}(t-1)\mathbf{E}_n \\ \mathbf{Y}(t+1) &= \mathbf{Y}(t) - \beta(t)\mathbf{E}_n\mathbf{Y}(t) \end{aligned}$$

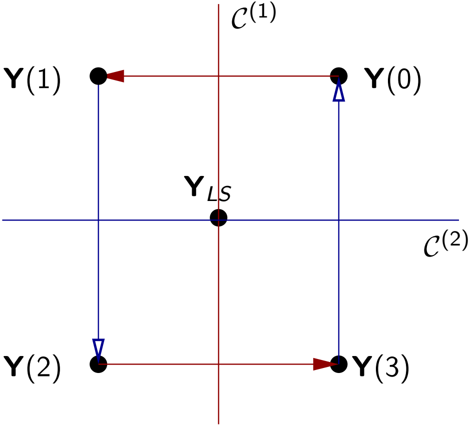
Parallel Decoding

$$\mathbf{Y}(t+1) = \mathbf{Y}(t) - \beta(t)\mathbf{Y}(t)\mathbf{E}_n - \beta(t)\mathbf{E}_n\mathbf{Y}(t)$$

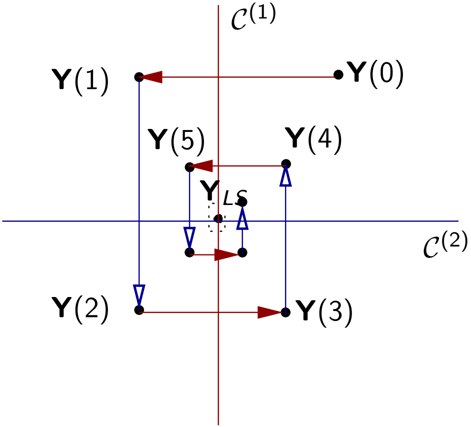
Exact Projection $w_1 = w_2 = 1/k, \beta = 1$



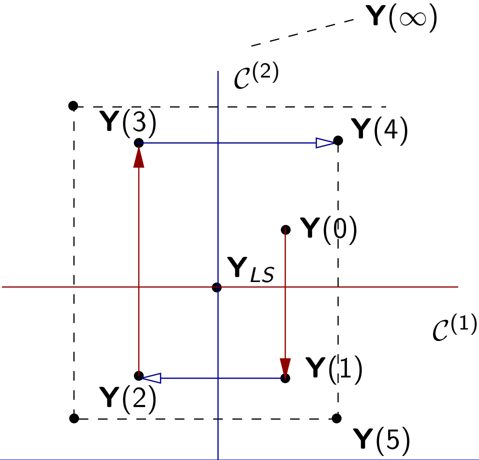
At the bound $w_1 = w_2 = 2/(k-1), \beta = 2$



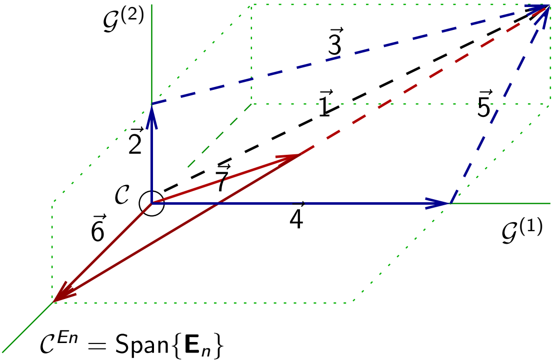
Converging Condition $w_1 = w_2 < 2/(k-1), \beta \le 2$



Diverging $w_1 = w_2 > 2/(k-1), \beta \geq 2$



Parallel Decoding



Summary

1. Iterative Decoding Algorithm for Analog Codes (MVE)
2. A novel geometric illustration of the iterative decoding for analog codes.
3. Orthogonal decomposition of the analog compound codes to achieve a faster convergence speed.
4. An intuitive tool to visualize the convergence behavior of the iterative decoding in the analog case.

Thanks!

Reference

- F.R. Kschishang and B.J. Frey, "Iterative Decoding of Compound Codes by Probability Propagation in Graphical Models," *IEEE JSAC*, vol.16, no.2, Feb. 1998.
- F.R. Kschishang, B.J. Frey, and H. Loeliger, "Factor Graphs and the Sum-Product Algorithm," *IEEE Trans. on Information Theory*, vol. 47, no. 2, Feb. 2001.
- H.A. Loeliger, "An introduction to factor graphs," *IEEE Signal Proc. Mag.*, pp. 28-41, Jan. 2004.
- F. Hu, W. Henkel, "A Geometric Description of the Iterative Least-Squares Decoding of Analog Block Codes", 4th International Symposium on Turbo Codes and Related Topics, Munich, April 3-7, 2006.

The trade off of the sparseness and the orthogonal decomposition of a $(7, 4)$ analog compound code