Code Ensembles
Ruediger Urbanke, EPFL
February 15th, 2010
~ 1 hours

For the most recent version of these slides visit

Modern Coding Theory
Recall

\[ H = \begin{pmatrix} x_1 & x_2 & x_3 & x_4 & x_5 & x_6 & x_7 \\ 1 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 1 \end{pmatrix} \]
Recall

\[ H = \begin{pmatrix} x_1 & x_2 & x_3 & x_4 & x_5 & x_6 & x_7 \\ 1 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 1 \end{pmatrix} \]

\[ f(x_1, \ldots, x_7) = 1_{\{x \in C(H)\}} = \begin{cases} 1, & \text{if } Hx^T = 0^T, \\ 0, & \text{otherwise}. \end{cases} \]
Recall

$$H = \begin{pmatrix} x_1 & x_2 & x_3 & x_4 & x_5 & x_6 & x_7 \\ 1 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 1 \end{pmatrix}$$

$$f(x_1, \ldots, x_7) = \mathbb{1}_{\{x \in C(H)\}} = \begin{cases} 1, & \text{if } Hx^T = 0^T, \\ 0, & \text{otherwise}. \end{cases}$$

$$f(x_1, \ldots, x_7) = \mathbb{1}_{\{x_1+x_2+x_4=0\}} \mathbb{1}_{\{x_3+x_4+x_5=0\}} \mathbb{1}_{\{x_4+x_5+x_7=0\}}$$
Recall

\[ H = \begin{pmatrix} \ x_1 & x_2 & x_3 & x_4 & x_5 & x_6 & x_7 \\ 1 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 1 \end{pmatrix} \]

\[ f(x_1, \ldots, x_7) = \mathbb{1}_{\{x \in C(H)\}} = \begin{cases} 1, & \text{if } Hx^T = 0^T; \\ 0, & \text{otherwise}. \end{cases} \]

\[ f(x_1, \ldots, x_7) = \mathbb{1}_{\{x_1+x_2+x_4=0\}} \mathbb{1}_{\{x_3+x_4+x_6=0\}} \mathbb{1}_{\{x_4+x_5+x_7=0\}} \]
Recall in order that the indicator functions only involve a small number of variables each row of $H$ should have only few ones.

$$H = \begin{pmatrix} x_1 & x_2 & x_3 & x_4 & x_5 & x_6 & x_7 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 1 \end{pmatrix}$$

$$f(x_1, \ldots, x_7) = \mathbb{1}_{\{x \in C(H)\}} = \begin{cases} 1, & \text{if } Hx^T = 0^T, \\ 0, & \text{otherwise}. \end{cases}$$

$$f(x_1, \ldots, x_7) = \mathbb{1}_{\{x_1 + x_2 + x_4 = 0\}} \mathbb{1}_{\{x_3 + x_4 + x_6 = 0\}} \mathbb{1}_{\{x_4 + x_5 + x_7 = 0\}}$$

in order that the indicator functions only involve a small number of variables each row of $H$ should have only few ones.
Low-Density Parity-Check Codes
Low-Density Parity-Check Codes

(3, 4)-regular code

Gallager ‘60
Low-Density Parity-Check Codes

(3, 4)-regular code

Gallager ‘60
Low-Density Parity-Check Codes

(3, 4)-regular code

Gallager ‘60
Low-Density Parity-Check Codes

(3, 4)-regular code

Gallager '60
Low-Density Parity-Check Codes

(3, 4)-regular code

Gallager ‘60

number of edges is linear in n
Ensemble of Codes
Ensemble of Codes

edges are chosen “randomly”
Ensemble of Codes

edges are chosen “randomly”
Ensemble of Codes

edges are chosen “randomly”
Ensemble of Codes

edges are chosen “randomly”
Ensemble of Codes

edges are chosen “randomly”
Ensemble of Codes

edges are chosen “randomly”

Ensemble parameters: variable and check degree and n
Configuration Model

(3, 6)-regular ensemble
Configuration Model

(3, 6)-regular ensemble
Configuration Model

(3, 6)-regular ensemble

sockets

Tuesday, February 16, 2010
Configuration Model

(3, 6)-regular ensemble

label the sockets

Tuesday, February 16, 2010
Configuration Model

there are 3 n labeled sockets in total

(3, 6)-regular ensemble
Configuration Model

(3, 6)-regular ensemble

there are 3 \( n \) labeled sockets in total

there are 3 \( n \) labeled sockets in total
Configuration Model

There are $3n$ labeled sockets in total.

Choose permutations on $3n$ letters uniformly at random.

There are $3n$ labeled sockets in total.
Design Rate
Design Rate

(d_v, d_c)-regular code; e.g., (3, 6)-regular code
Design Rate

\[(d_v \ d_c)\text{-regular code; e.g., (3, 6)\text{-regular code}}\]

matching edge counts: \(n \ d_v = m \ d_c\)
(d_v d_c)-regular code; e.g., (3, 6)-regular code

matching edge counts: n d_v = m d_c

m = n d_v / d_c
Design Rate

\[(d_v d_c)\text{-regular code; e.g., (3, 6)-regular code}\]

matching edge counts: \( n d_v = m d_c \)

\( m = n d_v/d_c \)

assume all equations are linearly independent
Design Rate

(d_v \text{ v } d_c)-regular code; e.g., (3, 6)-regular code

matching edge counts: n d_v = m d_c

m = n d_v/d_c

assume all equations are linearly independent

r = \frac{n-m}{n} = 1 - \frac{m}{n} = 1 - \frac{d_v}{d_c}
Design Rate

\[ (d_v \ d_c) \text{-regular code; e.g., (3, 6)-regular code} \]

matching edge counts: \( n \ d_v = m \ d_c \)

\( m = n \ \frac{d_v}{d_c} \)

assume all equations are linearly independent

\[ r = \frac{(n-m)}{n} = 1 - \frac{m}{n} = 1 - \frac{d_v}{d_c} \]
Design Rate

assume all equations are linearly independent

\[ (d_v \, d_c) \text{-regular code; e.g., (3, 6)-regular code} \]

matching edge counts: \( n \, d_v = m \, d_c \)

\[ m = n \, d_v / d_c \]

assume all equations are linearly independent

\[ r = (n-m)/n = 1 - m/n = 1 - d_v/d_c \]

Question: Are the equations indeed typically independent?
Rate versus Design Rate
Rate versus Design Rate

**Lemma 3.27 (Design Rate Equals Rate for Regular Ensembles).** Consider the regular ensemble LDPC \((nx^1, n^1_\frac{1}{x^r})\) with \(2 \leq 1 < r\). Let \(r(1, r) = 1 - \frac{1}{r}\) denote the design rate of the ensemble and let \(r(G)\) denote the actual rate of a code \(G\), \(G \in \text{LDPC} \left( nx^1, n^1_\frac{1}{x^r} \right) \). Then

\[
\mathbb{P}\{r(G)n = r(1, r)n + v\} = 1 - o_n(1),
\]

where \(v = 1\) if \(1\) is even, and \(v = 0\) otherwise.
Lemma 3.27 (Design Rate Equals Rate for Regular Ensembles). Consider the regular ensemble $\text{LDPC}(nx^1, n^{\frac{1}{r}}x^r)$ with $2 \leq 1 < r$. Let $r(1, r) = 1 - \frac{1}{r}$ denote the design rate of the ensemble and let $r(G)$ denote the actual rate of a code $G$, $G \in \text{LDPC}(nx^1, n^{\frac{1}{r}}x^r)$. Then

$$\mathbb{P}\{r(G)n = r(1, r)n + \nu\} = 1 - o_n(1),$$

where $\nu = 1$ if $l$ is even, and $\nu = 0$ otherwise.

Why do we loose one degree of freedom when $l$ is even?
Lemma 3.27 (Design Rate Equals Rate for Regular Ensembles). Consider the regular ensemble LDPC \((nx^1, n_{\frac{1}{r}}x^r)\) with \(2 \leq 1 < r\). Let \(r(1, r) = 1 - \frac{1}{r}\) denote the design rate of the ensemble and let \(r(G)\) denote the actual rate of a code \(G\), \(G \in \text{LDPC} \((nx^1, n_{\frac{1}{r}}x^r)\). Then

\[
P\{r(G) = r(1, r)n + \nu\} = 1 - o_n(1),
\]

where \(\nu = 1\) if \(1\) is even, and \(\nu = 0\) otherwise.

Why do we loose one degree of freedom when \(l\) is even?

Add up all equations. Since every variable appears times, and \(l\) is even, and since the equations are over GF(2) this gives the equation 0 = 0. In other words, we found a linearly dependent combination of the rows.
Irregular LDPC Codes

Figure 7.6: Tanner graph of a standard irregular LDPC code.
Parameterization of LDPC Ensembles

[7, 4, 3] Hamming Code

\[
H = \begin{pmatrix}
1 & 1 & 0 & 1 & 1 & 0 & 0 \\
1 & 0 & 1 & 1 & 0 & 1 & 0 \\
0 & 1 & 1 & 1 & 0 & 0 & 1
\end{pmatrix}
\]
Parameterization of LDPC Ensembles

\[ [7, 4, 3] \text{ Hamming Code} \]

\[ H = \begin{pmatrix} 1 & 1 & 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 & 1 \end{pmatrix} \]

\[ \Lambda(x) = \sum_{i=1}^{\Lambda_{\text{max}}} \Lambda_i x^i, \quad P(x) = \sum_{i=1}^{P_{\text{max}}} P_i x^i. \]

**Example 3.14 (Degree Distribution of [7, 4, 3] Hamming Code).** We have

\[ \Lambda(x) = 3x + 3x^2 + x^3, \quad P(x) = 3x^4, \]

\[ L(x) = \frac{3}{7}x + \frac{3}{7}x^2 + \frac{1}{7}x^3, \quad R(x) = x^4. \]
Parameterization of LDPC Ensembles

[7, 4, 3] Hamming Code

\[ H = \begin{pmatrix} 1 & 1 & 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 & 0 & 1 & 0 \end{pmatrix} \]

\[ \Lambda(x) = \sum_{i=1}^{\lambda_{\text{max}}} \Lambda_i x^i, \quad P(x) = \sum_{i=1}^{\mu_{\text{max}}} P_i x^i. \]

**Example 3.14 (Degree Distribution of [7, 4, 3] Hamming Code).** We have

\[ \Lambda(x) = 3x + 3x^2 + x^3, \quad P(x) = 3x^4, \]

\[ L(x) = \frac{3}{7} x + \frac{3}{7} x^2 + \frac{1}{7} x^3, \quad R(x) = x^4. \]

\[ \Lambda(1) = n, \quad P(1) = n\tilde{r}, \quad r(\Lambda, P) = 1 - \frac{P(1)}{\Lambda(1)}, \quad \Lambda'(1) = P'(1) \]

\[ L(x) = \frac{\Lambda(x)}{\Lambda(1)}, \quad R(x) = \frac{P(x)}{P(1)} \]
Node versus Edge Perspective
Node versus Edge Perspective

\[ \lambda(x) = \sum_i \lambda_i x^{i-1} = \frac{\Lambda'(x)}{\Lambda'(1)} = \frac{L'(x)}{L'(1)}, \quad \rho(x) = \sum_i \rho_i x^{i-1} = \frac{P'(x)}{P'(1)} = \frac{R'(x)}{R'(1)}. \]
Node versus Edge Perspective

\[ \lambda(x) = \sum_i \lambda_i x^{i-1} = \frac{\Lambda'(x)}{\Lambda'(1)} = \frac{L'(x)}{L'(1)}, \quad \rho(x) = \sum_i \rho_i x^{i-1} = \frac{P'(x)}{P'(1)} = \frac{R'(x)}{R'(1)}. \]

\[ r(\lambda, \rho) = 1 - \frac{1}{r_{avg}} = 1 - \frac{L'(1)}{R'(1)} = 1 - \frac{\int_0^1 \rho(x) \, dx}{\int_0^1 \lambda(x) \, dx}. \]
Node versus Edge Perspective

\[ \lambda(x) = \sum_i \lambda_i x^{i-1} = \frac{\Lambda'(x)}{\Lambda'(1)} = \frac{L'(x)}{L'(1)}, \quad \rho(x) = \sum_i \rho_i x^{i-1} = \frac{P'(x)}{P'(1)} = \frac{R'(x)}{R'(1)}. \]

\[ r(\lambda, \rho) = 1 - \frac{1_{\text{avg}}}{r_{\text{avg}}} = 1 - \frac{L'(1)}{R'(1)} = 1 - \frac{\int_0^1 \rho(x) \text{d}x}{\int_0^1 \lambda(x) \text{d}x}. \]

**Example 3.20 (Conversion of Degree Distributions: Hamming Code).** For the [7, 4, 3] Hamming code we have

\[ \lambda(x) = \frac{1}{4} + \frac{1}{2} x + \frac{1}{4} x^2, \quad \rho(x) = x^3. \]
Node versus Edge Perspective

\[ \lambda(x) = \sum_i \lambda_i x^{i-1} = \frac{\Lambda'(x)}{\Lambda'(1)} = \frac{L'(x)}{L'(1)}, \quad \rho(x) = \sum_i \rho_i x^{i-1} = \frac{P'(x)}{P'(1)} = \frac{R'(x)}{R'(1)}. \]

\[ r(\lambda, \rho) = 1 - \frac{1_{\text{avg}}}{r_{\text{avg}}} = 1 - \frac{L'(1)}{R'(1)} = 1 - \frac{\int_0^1 \rho(x) \, dx}{\int_0^1 \lambda(x) \, dx}. \]

**Example 3.20 (Conversion of Degree Distributions: Hamming Code).** For the [7, 4, 3] Hamming code we have

\[ \lambda(x) = \frac{1}{4} + \frac{1}{2}x + \frac{1}{4}x^2, \quad \rho(x) = x^3. \]

\[ \diamond \]

**Example 3.21 (Conversion of Degree Distributions: Second Example).** Consider the pair \((\Lambda, P)\):

\[ \Lambda(x) = 613x^2 + 202x^3 + 57x^4 + 84x^7 + 44x^8, \quad P(x) = 500x^6, \]

with

\[ \Lambda(1) = 1000, \quad P(1) = 500, \quad \Lambda'(1) = P'(1) = 3000. \]
Repeat Accumulate (RA) Codes

Figure 7.7: Encoder for an RA code. Each systematic bit is repeated 1 times; the resulting vector is permuted and fed into a filter with response $1/(1+D)$ (accumulate).

Figure 7.8: Tanner graph of an RA code with $1 = 3$. 
Irregular RA Codes

Figure 7.10: Tanner graph corresponding to an IRA code.
Accumulate Repeat Accumulate Codes

Figure 7.12: Tanner graph of an ARA code.
Low-Density Generator-Matrix (LDGM) Codes

Figure 7.15: Tanner graph of a simple LDGM code.

Figure 7.14: Tanner graph of an irregular LDGM code.
MN Codes

Figure 7.17: Tanner graph of an MN code.
Protographs

Figure 7.29: Base graph.
Protographs

Figure 7.29: Base graph.

Figure 7.30: Left: $m$ copies of base graph with $m = 5$. Right: Lifted graph resulting from applying permutations to the edge clusters.
Non-Binary Codes

Figure 7.34: FSFG of a simple code over $\mathbb{F}_4$ and its associated parity-check matrix $H$. The primitive polynomial generating $\mathbb{F}_4$ is $p(z) = 1 + z + z^2$. 
Turbo Codes
Turbo Codes
Questions

- Which ensemble is best?
- How do I choose the parameters?
- How can we determine the performance?
Select References


