Code-Based Cryptography

McEliece Cryptosystem

I. Márquez-Corbella

2. McEliece Cryptosystem

1. Formal Definition
2. Security-Reduction Proof
3. McEliece Assumptions
4. Notions of Security
5. Critical Attacks - Semantic Secure Conversions
6. Reducing the Key Size
7. **Reducing the Key Size - LDPC codes**
8. Reducing the Key Size - MDPC codes
9. Implementation

Low-density parity-check (LDPC) codes

1963: Gallager introduced LDPC codes

R. G. Gallager.
Low-Density Parity-Check Codees.
PhD thesis, MIT, 1963.

Low-density parity-check (LDPC) codes

1963: Gallager introduced LDPC codes

R. G. Gallager.
Low-Density Parity-Check Codees.
PhD thesis, MIT, 1963.

1981: Tanner introduced a graphical representation

R. M. Tanner.
A recursive approach to low complexity codes.
In IEEE Transaction on Information Theory, 27(5):533-547, 2006.

Low-density parity-check (LDPC) codes

1963: Gallager introduced LDPC codes

R. G. Gallager.
Low-Density Parity-Check Codees.
PhD thesis, MIT, 1963.

1981: Tanner introduced a graphical representation

R. M. Tanner.
A recursive approach to low complexity codes.
In IEEE Transaction on Information Theory, 27(5):533-547, 2006.

1996: MacKay and Neal (re)-discovered LDPC codes

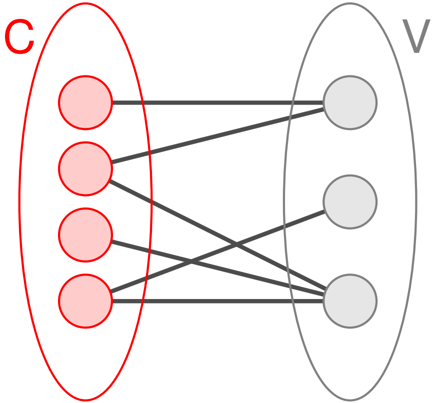
D. J.C. MacKay and R. M. Neal.
Near shannon limit performance of Low Density Parity Check codes.
Electronics Letters, 32:1645âĂŞ1646, 1996.

Representation for LDPC codes

- **Matrix Representation:** Sparse parity check matrix $H \in \mathbb{F}_2^{m \times n}$

Representation for LDPC codes

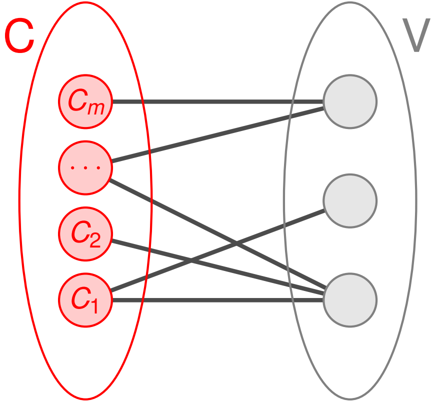
- **Matrix Representation:** Sparse parity check matrix $H \in \mathbb{F}_2^{m \times n}$
- **Graphical Representation**



Bipartite Graph

Representation for LDPC codes

- **Matrix Representation:** Sparse parity check matrix $H \in \mathbb{F}_2^{m \times n}$
- **Graphical Representation**



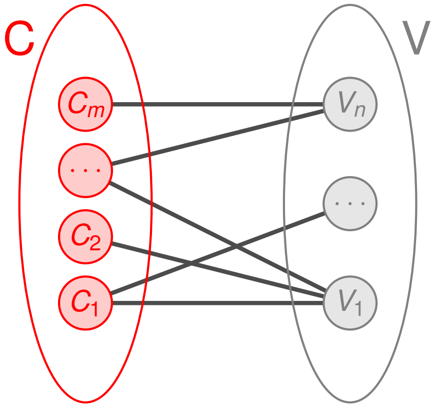
Tanner Graph

Variable nodes

$V_j \longleftrightarrow j$-th column of H

Representation for LDPC codes

- **Matrix Representation:** Sparse parity check matrix $H \in \mathbb{F}_2^{m \times n}$
- **Graphical Representation**



Tanner Graph

Variable nodes

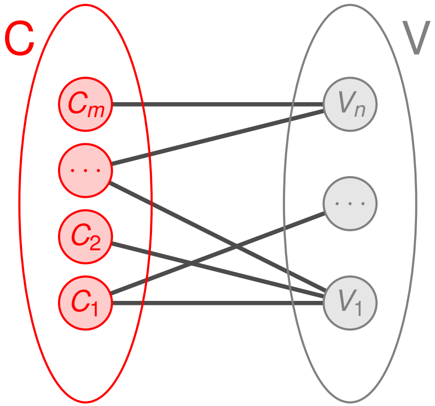
$V_j \longleftrightarrow j$-th column of H

Check nodes

$C_i \longleftrightarrow i$-th row of H

Representation for LDPC codes

- **Matrix Representation:** Sparse parity check matrix $H \in \mathbb{F}_2^{m \times n}$
- **Graphical Representation**



Tanner Graph

Variable nodes

$V_j \longleftrightarrow j$-th column of H

Check nodes

$C_i \longleftrightarrow i$-th row of H

Edges

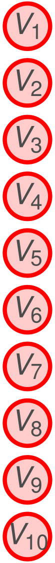
$e_{i,j} = \{C_i, V_j\} \longleftrightarrow h_{i,j} = 1$ in H

Example

Let $\mathcal{C}$ be an $[10, 7]$ binary LDPC code with parity-check matrix:

$$H = \begin{pmatrix} 0 & 1 & 1 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \end{pmatrix} \in \mathbb{F}_2^{3\times 10}$$

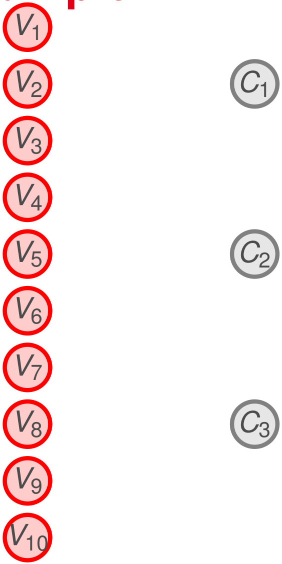
Example



Let $\mathcal{C}$ be an $[10, 7]$ binary LDPC code with parity-check matrix:

$$H = \begin{pmatrix} 0 & 1 & 1 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \end{pmatrix} \in \mathbb{F}_2^{3\times 10}$$

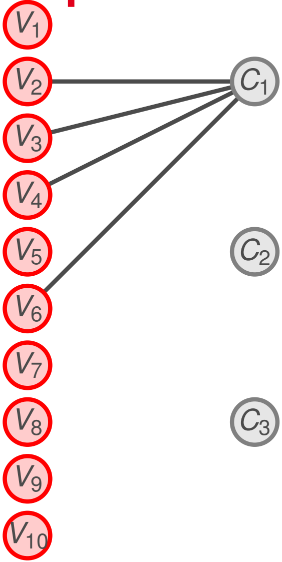
Example



Let $\mathcal{C}$ be an $[10, 7]$ binary LDPC code with parity-check matrix:

$$H = \begin{pmatrix} 0 & 1 & 1 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \end{pmatrix} \in \mathbb{F}_2^{3\times 10}$$

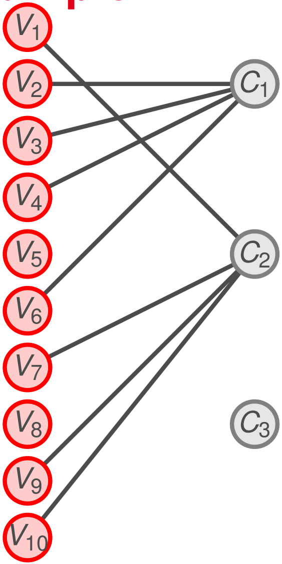
Example



Let $\mathcal{C}$ be an $[10, 7]$ binary LDPC code with parity-check matrix:

$$H = \begin{pmatrix} 0 & 1 & 1 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \end{pmatrix} \in \mathbb{F}_2^{3\times 10}$$

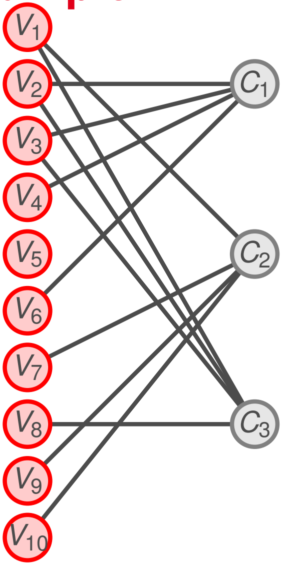
Example



Let $\mathcal{C}$ be an $[10, 7]$ binary LDPC code with parity-check matrix:

$$H = \begin{pmatrix} 0 & 1 & 1 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \end{pmatrix} \in \mathbb{F}_2^{3\times 10}$$

Example



Let $\mathcal{C}$ be an $[10, 7]$ binary LDPC code with parity-check matrix:

$$H = \begin{pmatrix} 0 & 1 & 1 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \end{pmatrix} \in \mathbb{F}_2^{3\times 10}$$

Bit-Flipping decoding algorithm

Step 1 - Iteration I Compute:

$f_j :=$ Number of unsatisfied parity-check equations of V_j with $j = 1, \ldots, n$
$f := \max(f_1, \ldots, f_n)$

Bit-Flipping decoding algorithm

Step 1 - Iteration I Compute:

$f_j :=$ Number of unsatisfied parity-check equations of V_j with $j = 1, \ldots, n$
$f := \max(f_1, \ldots, f_n)$

Step 2 - Iteration I **Bit-Flipping**

$$V_j = \begin{cases} 1 - V_j & \text{, if } f_j = f \\ V_j & \text{, otherwise} \end{cases}$$

Bit-Flipping decoding algorithm

Step 1 - Iteration I Compute:

$f_j :=$ Number of unsatisfied parity-check equations of V_j with $j = 1, \ldots, n$
$f := \max(f_1, \ldots, f_n)$

Step 2 - Iteration I **Bit-Flipping**

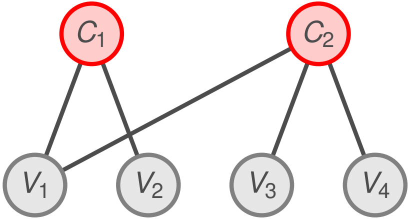
$$V_j = \begin{cases} 1 - V_j & \text{, if } f_j = f \\ V_j & \text{, otherwise} \end{cases}$$

Step 3 - Iteration I **Stop Criterion**

- Success: If $f = 0$ and I $< I_{max}$
- Failure: If $f \neq 0$ and I $= I_{max}$

Bit-Flipping Decoding - Example

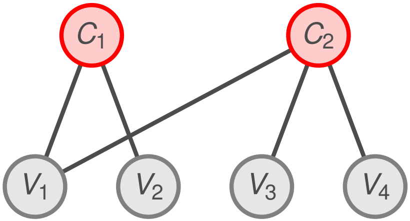
Received Data: $(0, 1, 1, 1)$



Bit-Flipping Decoding - Example

Received Data: $(0, 1, 1, 1)$

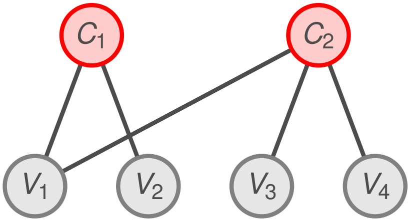
	V_1	V_2	V_3	V_4
Current	0	1	1	1
C_0	✗	✗	–	–
C_1	✓	–	✓	✓
f_j	1	1	0	0



Bit-Flipping Decoding - Example

	V_1	V_2	V_3	V_4
Current	0	1	1	1
C_0	✗	✗	–	–
C_1	✓	–	✓	✓
f_j	1	1	0	0
Updated	1	0	1	1
C_0	✗	✗	–	–
C_1	✗	–	✗	✗
f_j	2	1	1	1

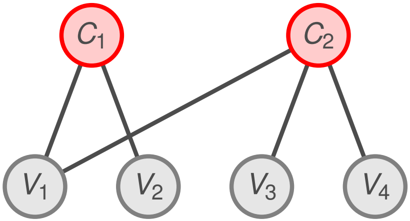
Received Data: $(0, 1, 1, 1)$



Bit-Flipping Decoding - Example

Received Data: $(0, 1, 1, 1)$

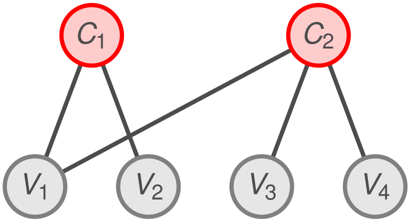
	V_1	V_2	V_3	V_4
Current	0	1	1	1
C_0	✗	✗	–	–
C_1	✓	–	✓	✓
f_j	1	1	0	0
Updated	1	0	1	1
C_0	✗	✗	–	–
C_1	✗	–	✗	✗
f_j	2	1	1	1
Updated	0	0	1	1
C_0	✓	✓	–	–
C_1	✓	–	✓	✓
f_j	0	0	0	0



Bit-Flipping Decoding - Example

	V_1	V_2	V_3	V_4
Current	0	1	1	1
C_0	✗	✗	–	–
C_1	✓	–	✓	✓
f_j	1	1	0	0
Updated	1	0	1	1
C_0	✗	✗	–	–
C_1	✗	–	✗	✗
f_j	2	1	1	1
Updated	0	0	1	1
C_0	✓	✓	–	–
C_1	✓	–	✓	✓
f_j	0	0	0	0

Received Data: (0, 1, 1, 1)



Decoding Result: (0, 0, 1, 1)

Variants based on LDPC codes

Using pure LDPC codes

C. Monico, J. Rosenthal, A. Shokrollahi.
Using low density parity check codes in the McEliece cryptosystem.
In ISIT 2000, pp. 215.

Variants based on LDPC codes

➢ Using pure LDPC codes

C. Monico, J. Rosenthal, A. Shokrollahi.
Using low density parity check codes in the McEliece cryptosystem.
In ISIT 2000, pp. 215.

✗ **Weakness:** Search for low weight codewords in the dual of the public code

Variants using QC-LDPC codes

➢ First proposal

M. Baldi, F. Chiaraluce, and R. Garello.
On the usage of quasicyclic low-density parity-check codes in the McEliece cryptosystem..
In ICEE 2006, pp. 305-310.

Variants using QC-LDPC codes

➢ **First proposal**

M. Baldi, F. Chiaraluce, and R. Garello.
On the usage of quasicyclic low-density parity-check codes in the McEliece cryptosystem..
In ICEE 2006, pp. 305-310.

✗ **Weakness:** Same as the *pure* LDPC variants.

Variants using QC-LDPC codes

- **First proposal**

  M. Baldi, F. Chiaraluce, and R. Garello.
  On the usage of quasicyclic low-density parity-check codes in the McEliece cryptosystem..
  In ICEE 2006, pp. 305-310.

✗ **Weakness:** Same as the *pure* LDPC variants.

- **Using an auxiliary “dense” matrix**

  M. Baldi, F. Chiaraluce, R. Garello, and F. Mininni.
  Quasi-cyclic low-density parity-check codes in the McEliece cryptosystem.
  In ICC 2007, pp. 951-956.

  M. Baldi and F. Chiaraluce.
  Cryptanalysis of a new instance of McEliece cryptosystem based on QC-LDPC codes.
  In ISIT 2007, pp. 2591-2595.

Variants using QC-LDPC codes

➢ **First proposal**

M. Baldi, F. Chiaraluce, and R. Garello.
On the usage of quasicyclic low-density parity-check codes in the McEliece cryptosystem..
In ICEE 2006, pp. 305-310.

✗ **Weakness:** Same as the *pure* LDPC variants.

➢ **Using an auxiliary "dense" matrix**

M. Baldi, F. Chiaraluce, R. Garello, and F. Mininni.
Quasi-cyclic low-density parity-check codes in the McEliece cryptosystem.
In ICC 2007, pp. 951-956.

M. Baldi and F. Chiaraluce.
Cryptanalysis of a new instance of McEliece cryptosystem based on QC-LDPC codes.
In ISIT 2007, pp. 2591-2595.

✗ **Attack:**

A. Otmani, J.P. Tillich, and L. Dallot.
Cryptanalysis of two McEliece cryptosystems based on quasi-cyclic codes.
Special Issues of Mathematics in Computer Science, pp. 126-140, 2010.

Variants using QC-LDPC codes

➢ **First proposal**

M. Baldi, F. Chiaraluce, and R. Garello.
On the usage of quasicyclic low-density parity-check codes in the McEliece cryptosystem..
In ICEE 2006, pp. 305-310.

✗ **Weakness:** Same as the *pure* LDPC variants.

➢ **Using an auxiliary "dense" matrix**

M. Baldi, F. Chiaraluce, R. Garello, and F. Mininni.
Quasi-cyclic low-density parity-check codes in the McEliece cryptosystem.
In ICC 2007, pp. 951-956.

M. Baldi and F. Chiaraluce.
Cryptanalysis of a new instance of McEliece cryptosystem based on QC-LDPC codes.
In ISIT 2007, pp. 2591-2595.

✗ **Attack:**

A. Otmani, J.P. Tillich, and L. Dallot.
Cryptanalysis of two McEliece cryptosystems based on quasi-cyclic codes.
Special Issues of Mathematics in Computer Science, pp. 126-140, 2010.

➢ **New variant:**

M. Baldi, M. Bodrato, and F. Chiaraluce.
A new analysis of the McEliece cryptosystem based on QC-LDPC codes.
In SCN 2008, pp. 246-262.

2. McEliece Cryptosystem

1. Formal Definition
2. Security-Reduction Proof
3. McEliece Assumptions
4. Notions of Security
5. Critical Attacks - Semantic Secure Conversions
6. Reducing the Key Size
7. Reducing the Key Size - LDPC codes
8. **Reducing the Key Size - MDPC codes**
9. Implementation