An introduction to coding theory

Adrish Banerjee

Department of Electrical Engineering
Indian Institute of Technology Kanpur
Kanpur, Uttar Pradesh
India

Feb. 27, 2017

Lecture #13: Low density parity check codes

Outline of the lecture

- Introduction

Outline of the lecture

- Introduction
- Tanner graphs

Outline of the lecture

- Introduction
- Tanner graphs
- Construction of regular LDPC codes

Outline of the lecture

- Introduction
- Tanner graphs
- Construction of regular LDPC codes
  - Gallager's construction

Outline of the lecture

- Introduction
- Tanner graphs
- Construction of regular LDPC codes
  - Gallager's construction
  - MacKay's construction

Outline of the lecture

- Introduction
- Tanner graphs
- Construction of regular LDPC codes
  - Gallager's construction
  - MacKay's construction
- Irregular LDPC codes

Outline of the lecture

- Introduction
- Tanner graphs
- Construction of regular LDPC codes
  - Gallager's construction
  - MacKay's construction
- Irregular LDPC codes
- Random construction of irregular LDPC codes.

Definitions

- The **density** of a source of random bits is the expected fraction of 1 bits.

Definitions

- The **density** of a source of random bits is the expected fraction of 1 bits.
- A source is **sparse** if its density is less than 0.5.

Definitions

- The **density** of a source of random bits is the expected fraction of 1 bits.
- A source is **sparse** if its density is less than 0.5.
- A vector **v** is **very sparse** if its density vanishes as its length increases.

Definitions

- The **density** of a source of random bits is the expected fraction of 1 bits.
- A source is **sparse** if its density is less than 0.5.
- A vector $\mathbf{v}$ is **very sparse** if its density vanishes as its length increases.
- The **overlap** between two vectors is the number of 1's in common between them.

Low-density parity check codes

- Low-density parity-check (LDPC) codes are codes specified by a parity check matrix $\mathbf{H}$ containing mostly 0's and only a small number of 1's.

Low-density parity check codes

- Low-density parity-check (LDPC) codes are codes specified by a parity check matrix $\mathbf{H}$ containing mostly 0's and only a small number of 1's.
- A *regular* (n, w_c, w_r) LDPC code is a code of blocklength n with a $m \times n$ parity check matrix where each column contains a small fixed number, $w_c \geq 3$, of 1's and each row contains a small fixed number, $w_r \geq w_c$, of 1's.

Low-density parity check codes

- Low-density parity-check (LDPC) codes are codes specified by a parity check matrix $\mathbf{H}$ containing mostly 0's and only a small number of 1's.
- A *regular* (n, w_c, w_r) LDPC code is a code of blocklength n with a $m \times n$ parity check matrix where each column contains a small fixed number, $w_c \geq 3$, of 1's and each row contains a small fixed number, $w_r \geq w_c$, of 1's.
- In other words,

Low-density parity check codes

- Low-density parity-check (LDPC) codes are codes specified by a parity check matrix $\mathbf{H}$ containing mostly 0's and only a small number of 1's.
- A *regular* (n, w_c, w_r) LDPC code is a code of blocklength n with a $m \times n$ parity check matrix where each column contains a small fixed number, $w_c \geq 3$, of 1's and each row contains a small fixed number, $w_r \geq w_c$, of 1's.
- In other words,
  - Each parity check constraint involves w_r codebits, and each codebit is involved in w_c constraints.

Low-density parity check codes

- Low-density parity-check (LDPC) codes are codes specified by a parity check matrix $\mathbf{H}$ containing mostly 0's and only a small number of 1's.
- A *regular* (n, w_c, w_r) LDPC code is a code of blocklength n with a $m \times n$ parity check matrix where each column contains a small fixed number, $w_c \geq 3$, of 1's and each row contains a small fixed number, $w_r \geq w_c$, of 1's.
- In other words,
  - Each parity check constraint involves w_r codebits, and each codebit is involved in w_c constraints.
  - Low-density implies that $w_c << m$ and $w_r << n$.

Low-density parity check codes

- Low-density parity-check (LDPC) codes are codes specified by a parity check matrix $\mathbf{H}$ containing mostly 0's and only a small number of 1's.
- A *regular* (n, w_c, w_r) LDPC code is a code of blocklength n with a $m \times n$ parity check matrix where each column contains a small fixed number, $w_c \geq 3$, of 1's and each row contains a small fixed number, $w_r \geq w_c$, of 1's.
- In other words,
  - Each parity check constraint involves w_r codebits, and each codebit is involved in w_c constraints.
  - Low-density implies that $w_c << m$ and $w_r << n$.
  - Number of ones in the parity check matrix $\mathbf{H} = w_c \cdot n = w_r \cdot m$.

Low-density parity check codes

- Low-density parity-check (LDPC) codes are codes specified by a parity check matrix $\mathbf{H}$ containing mostly 0's and only a small number of 1's.
- A *regular* (n, w_c, w_r) LDPC code is a code of blocklength n with a $m \times n$ parity check matrix where each column contains a small fixed number, $w_c \geq 3$, of 1's and each row contains a small fixed number, $w_r \geq w_c$, of 1's.
- In other words,
  - Each parity check constraint involves w_r codebits, and each codebit is involved in w_c constraints.
  - Low-density implies that $w_c << m$ and $w_r << n$.
  - Number of ones in the parity check matrix $\mathbf{H} = w_c \cdot n = w_r \cdot m$.
  - $m \geq n - k \implies R = k/n \geq 1 - (w_c/w_r)$, and thus $w_c < w_r$.

Regular low-density parity check code

1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	1	1	1	1	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1
1	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	0	0	0	0
0	1	0	0	0	1	0	0	0	1	0	0	0	0	0	0	1	0	0	0
0	0	1	0	0	0	1	0	0	0	0	0	0	1	0	0	0	1	0	0
0	0	0	1	0	0	0	0	0	0	1	0	0	0	1	0	0	0	1	0
0	0	0	0	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	1
1	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	1	0	0
0	1	0	0	0	0	1	0	0	0	1	0	0	0	0	1	0	0	0	0
0	0	1	0	0	0	0	1	0	0	0	0	1	0	0	0	0	0	1	0
0	0	0	1	0	0	0	0	1	0	0	0	0	1	0	0	1	0	0	0
0	0	0	0	1	0	0	0	0	1	0	0	0	0	1	0	0	0	0	1

Example of a regular low density code matrix; $n = 20$, $w_c = 3$, $w_r = 4$

Tanner Graphs

- A *bipartite graph* is one in which the nodes can be partitioned into two classes, and no edge can connect nodes from the same class.

Tanner Graphs

- A *bipartite graph* is one in which the nodes can be partitioned into two classes, and no edge can connect nodes from the same class.
- A *Tanner graph* for an LDPC code is an bipartite graph such that:

Tanner Graphs

- A *bipartite graph* is one in which the nodes can be partitioned into two classes, and no edge can connect nodes from the same class.
- A *Tanner graph* for an LDPC code is an bipartite graph such that:
  - One class of nodes is the “variable nodes” corresponding to n bits in the codeword.

Tanner Graphs

- A *bipartite graph* is one in which the nodes can be partitioned into two classes, and no edge can connect nodes from the same class.
- A *Tanner graph* for an LDPC code is an bipartite graph such that:
  - One class of nodes is the "variable nodes" corresponding to n bits in the codeword.
  - Second class of nodes is "check nodes" corresponding to m parity check equations.

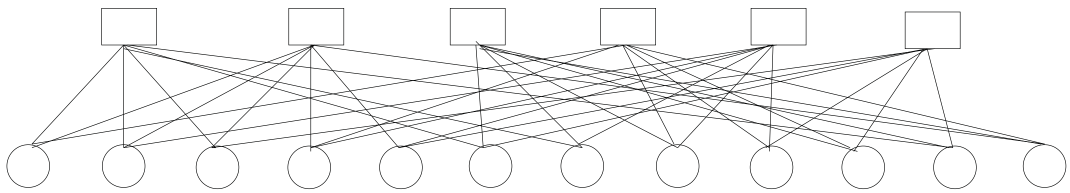
Tanner Graphs

- A *bipartite graph* is one in which the nodes can be partitioned into two classes, and no edge can connect nodes from the same class.
- A *Tanner graph* for an LDPC code is an bipartite graph such that:
  - One class of nodes is the "variable nodes" corresponding to n bits in the codeword.
  - Second class of nodes is "check nodes" corresponding to m parity check equations.
  - An edge connects a variable node to the check node if and only if that particular bit is included in the parity check equation.

Tanner Graphs

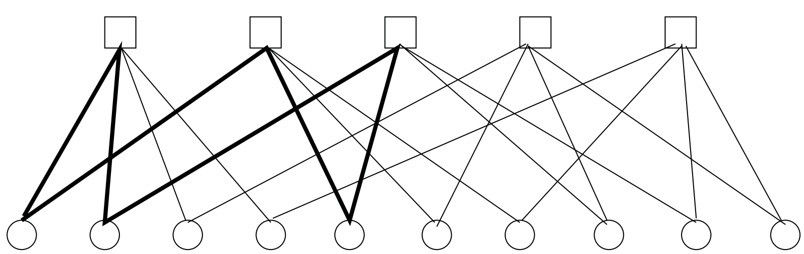
- Example of a regular low density code matrix; $n = 12$, $w_c = 3$, $w_r = 6$

$$H = \begin{pmatrix} 1 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 1 & 0 \end{pmatrix}$$



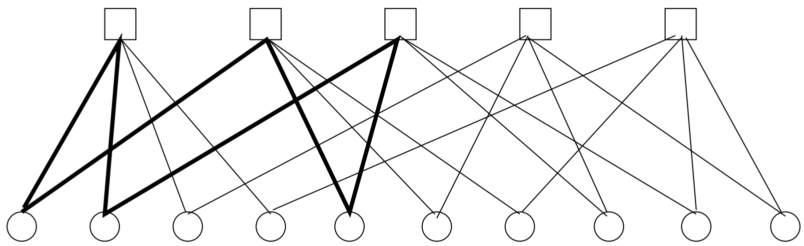
Tanner Graphs

- A cycle of length l in a Tanner graph is a path comprised of l edges from a node back to the same node.



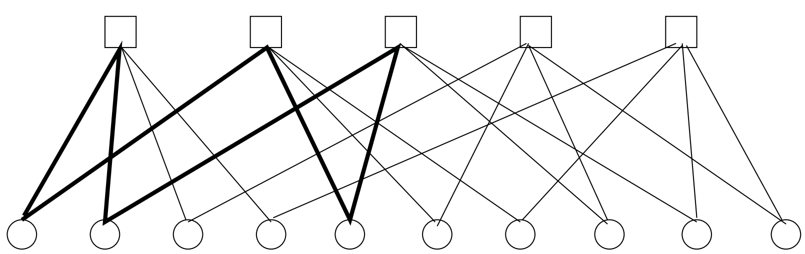
Tanner Graphs

- A cycle of length l in a Tanner graph is a path comprised of l edges from a node back to the same node.
- Example: The bipartite graph has a cycle of length six.



Tanner Graphs

$$\mathbf{H} = \begin{bmatrix} \boxed{1} & \boxed{1} & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ \boxed{1} & 0 & 0 & 0 & \boxed{1} & 1 & 1 & 0 & 0 & 0 \\ 0 & \boxed{1} & 0 & 0 & \boxed{1} & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 1 \end{bmatrix}$$

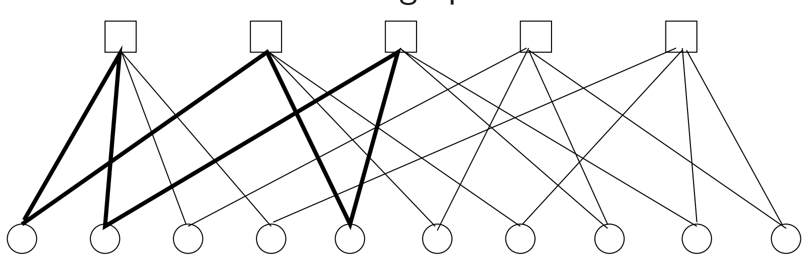


Tanner Graphs

- The length of smallest cycle in the graph is known as its girth.

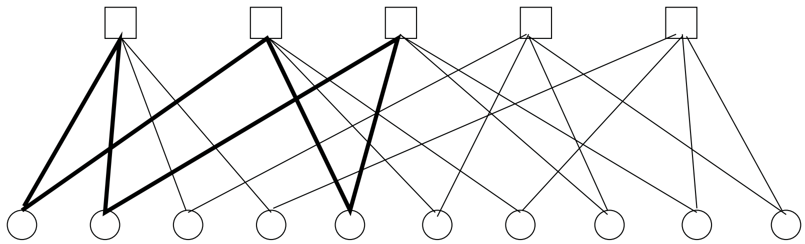
Tanner Graphs

- The length of smallest cycle in the graph is known as its girth.
- When decoding LDPC codes using sum-product algorithm, the number of independent iterations of the algorithm is proportional to the girth of its associated Tanner graph.



Tanner Graphs

- The length of smallest cycle in the graph is known as its girth.
- When decoding LDPC codes using sum-product algorithm, the number of independent iterations of the algorithm is proportional to the girth of its associated Tanner graph.



- The girth of this Tanner graph is six.

Gallager's construction for regular (n, w_c, w_r) code

- Let, n be the transmitted block-length of an information sequence of length k. m is the number of parity check equations.

Gallager's construction for regular (n, w_c, w_r) code

- Let, n be the transmitted block-length of an information sequence of length k. m is the number of parity check equations.
- Construct a $m \times n$ matrix with w_c 1's per column and w_r 1's per row. (An (n, w_c, w_r) code)

Gallager's construction for regular (n, w_c, w_r) code

- Let, n be the transmitted block-length of an information sequence of length k. m is the number of parity check equations.
- Construct a $m \times n$ matrix with w_c 1's per column and w_r 1's per row. (An (n, w_c, w_r) code)
- Divide a $m \times n$ matrix into w_c $m/w_c \times n$ sub-matrices, each containing a single 1 in each column.

Gallager's construction for regular (n, w_c, w_r) code

- Let, n be the transmitted block-length of an information sequence of length k. m is the number of parity check equations.
- Construct a $m \times n$ matrix with w_c 1's per column and w_r 1's per row. (An (n, w_c, w_r) code)
- Divide a $m \times n$ matrix into w_c $m/w_c \times n$ sub-matrices, each containing a single 1 in each column.
- The first of these sub-matrices contains all 1's in descending order, i.e. the i'th row contains 1's in columns $(i-1) \cdot w_r + 1$ to $i \cdot w_r$.

Gallager's construction for regular (n, w_c, w_r) code

- Let, n be the transmitted block-length of an information sequence of length k. m is the number of parity check equations.
- Construct a $m \times n$ matrix with w_c 1's per column and w_r 1's per row. (An (n, w_c, w_r) code)
- Divide a $m \times n$ matrix into w_c $m/w_c \times n$ sub-matrices, each containing a single 1 in each column.
- The first of these sub-matrices contains all 1's in descending order, i.e. the i'th row contains 1's in columns $(i-1) \cdot w_r + 1$ to $i \cdot w_r$.
- The other sub-matrices are merely column permutations of the first sub-matrix.

Gallager's construction for regular (n, w_c, w_r) code

1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	1	1	1	1	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1
1	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	0	0	0	0
0	1	0	0	0	1	0	0	0	1	0	0	0	0	0	0	1	0	0	0
0	0	1	0	0	0	1	0	0	0	0	0	0	1	0	0	0	1	0	0
0	0	0	1	0	0	0	0	0	0	1	0	0	0	1	0	0	0	1	0
0	0	0	0	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	1
1	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	1	0	0
0	1	0	0	0	0	1	0	0	0	1	0	0	0	0	1	0	0	0	0
0	0	1	0	0	0	0	1	0	0	0	0	1	0	0	0	0	0	1	0
0	0	0	1	0	0	0	0	1	0	0	0	0	1	0	0	1	0	0	0
0	0	0	0	1	0	0	0	0	1	0	0	0	0	1	0	0	0	0	1

Example of a regular low density code matrix; $n = 20$**,** $w_c = 3$**,** $w_r = 4$

MacKay's construction

- An m by n matrix (m rows, n columns) is created at random with weight per column w_c, and weight per row w_r, and overlap between any two columns no greater than 1.

Reference:

- "Good Error-Correcting Codes Based on Very Sparse Matrices" by David J. C. MacKay in *IEEE Transactions on Information Theory*, pp. 399-431, Mar. 1999.

MacKay's construction

- An m by n matrix (m rows, n columns) is created at random with weight per column w_c, and weight per row w_r, and overlap between any two columns no greater than 1.
- Another way of constructing regular LDPC codes is to build the parity check matrix from non-overlapping random permutation matrices.

Reference:

- "Good Error-Correcting Codes Based on Very Sparse Matrices" by David J. C. MacKay in *IEEE Transactions on Information Theory*, pp. 399-431, Mar. 1999.

Construction of low density parity check codes

- A permutation matrix is just the identity matrix with its row re-ordered, e.g.

$$P = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 \end{bmatrix}$$

Construction of low density parity check codes

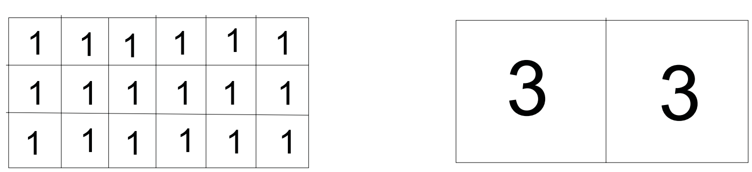
- A permutation matrix is just the identity matrix with its row re-ordered, e.g.

$$P = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 \end{bmatrix}$$

- A circulant matrix is defined by the property that each row is a cyclic shift of the previous row to the right by one position.

$$C = \begin{bmatrix} 0 & 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 \end{bmatrix}$$

MacKay's construction



Schematic Illustration of Regular Gallager Codes

Notation: An integer represents a number of permutation matrices superposed on the surrounding square.

Column Weight	Fraction of columns	Row weight	Fraction
3	1	6	1

Irregular low-density parity check codes

- For an **irregular** low-density parity-check code the degrees of each set of nodes are chosen according to some distribution.

Irregular low-density parity check codes

- For an **irregular** low-density parity-check code the degrees of each set of nodes are chosen according to some distribution.
- A degree distribution $\gamma(x) = \sum_i \gamma_i x^{i-1}$ is simply a polynomial with nonnegative real coefficients satisfying $\gamma(1) = 1$.

Irregular low-density parity check codes

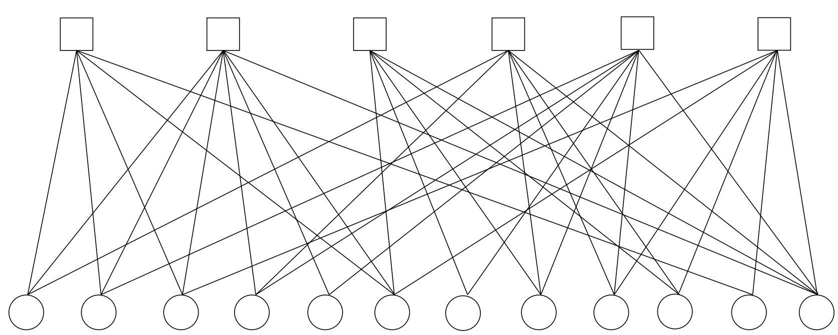
- For an **irregular** low-density parity-check code the degrees of each set of nodes are chosen according to some distribution.
- A degree distribution $\gamma(x) = \sum_i \gamma_i x^{i-1}$ is simply a polynomial with nonnegative real coefficients satisfying $\gamma(1) = 1$.
- An irregular low-density code is a code of block-length N with a sparse parity check matrix where column distribution $\lambda(x)$ and row distribution $\rho(x)$ is respectively given by

$$\lambda(x) = \sum_{i>1} \lambda_i x^{i-1}$$

$$\rho(x) = \sum_{i>1} \rho_i x^{i-1}$$

where λ_i and ρ_i denote the fraction of edges incident to variable and check nodes with degree i, respectively.

Irregular Low-density parity check code



$$\lambda(x) = \tfrac{1}{4}x + \tfrac{7}{12}x^2 + \tfrac{1}{12}x^3 + \tfrac{1}{12}x^4$$
$$\rho(x) = \tfrac{1}{3}x^4 + \tfrac{1}{3}x^5 + \tfrac{1}{3}x^6$$

Irregular low density parity check codes

Construction of irregular LDPC code

Step 1 : Selecting a profile that describes the desired number of columns of each weight and the desired number of rows of each weight.

Step 2 : Construction method, i.e. algorithm for putting edges between the vertices in a way that satisfies the constraints.

Random construction of Irregular LDPC Codes

The edges are placed “completely at random” subject to the profile constraints. One way of implementing it is shown below.

- Make a list of all columns in the matrix, with each column appearing in the list a number of times equal to its weight.

Random construction of Irregular LDPC Codes

The edges are placed “completely at random” subject to the profile constraints. One way of implementing it is shown below.

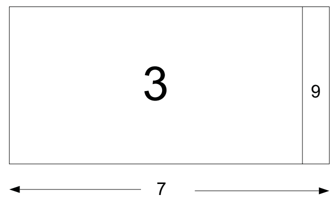
- Make a list of all columns in the matrix, with each column appearing in the list a number of times equal to its weight.
- Make a similar list of all rows in the matrix, with each row appearing in the list a number of times equal to its weight.

Random construction of Irregular LDPC Codes

The edges are placed “completely at random” subject to the profile constraints. One way of implementing it is shown below.

- Make a list of all columns in the matrix, with each column appearing in the list a number of times equal to its weight.
- Make a similar list of all rows in the matrix, with each row appearing in the list a number of times equal to its weight.
- Map one list onto the other by a random permutation, taking care not to create duplicate entries.

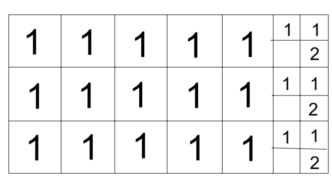
Construction of Irregular LDPC Codes



Notation: integers "3" and "9" represent the column weights.

Column Weight	Fraction of columns	Row weight	Fraction
3 9	11/12 1/12	7	1

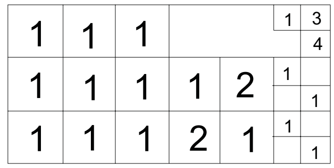
Construction of Irregular LDPC Codes



Notation: An integer represents a number of permutation matrices superposed on the surrounding square. Horizontal and vertical lines indicate the boundaries of the permutation blocks.

Column Weight	Fraction of columns	Row weight	Fraction
3 9	11/12 1/12	7	1

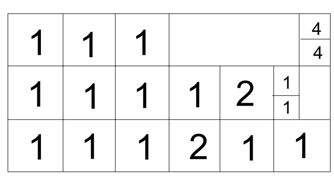
Construction of Irregular LDPC Codes



Notation: An integer represents a number of permutation matrices superposed on the surrounding square. Horizontal and vertical lines indicate the boundaries of the permutation blocks.

Column Weight	Fraction of columns	Row weight	Fraction
3	11/12	7	1
9	1/12		

Construction of Irregular LDPC Codes



Notation: An integer represents a number of permutation matrices superposed on the surrounding square. Horizontal and vertical lines indicate the boundaries of the permutation blocks.

Column Weight	Fraction of columns	Row weight	Fraction
3	11/12	7	1
9	1/12		