Design of LDPC codes

- Codes from finite geometries
- Random codes: Determine the connections of the bipartite Tanner graph by using a (pseudo)random algorithm observing the degree distribution of the code bit vertices and the parity check vertices
  - Regular
  - Irregular
- Graph theoretic codes
- Combinatorial codes
- Other algebraic constructions
Column splitting

- Can be applied to any code; also those designed by use of finite geometries
- Effects:
  - The variable nodes in the Tanner graph are split into several nodes
  - The new *extended* code $C_{\text{ext}}$ will have the following properties:
    - More code symbols (higher $n$)
    - Higher code rate ($J$ is constant; rank of $H$ may increase, but usually not by much)
    - Row weight is unchanged. Any two columns will still have at most one 1 in common
    - The column weight is reduced from its original value $\gamma$ to $\gamma_{\text{ext}}$
    - The minimum distance is reduced
    - Cycles in the Tanner graph are broken
Column splitting: Example

- (4095,3367) type-I cyclic 2-dimensional (0,6)th order EG code
- Split each column of $H$ into 16 new columns
- $C_{\text{ext}}$ is a (65520,61425) code with $\rho = 64$, $\gamma_{\text{ext}} = 4$, $R = 0.9375$, $r = 0.00098$

![Graph showing bit/block-error probability vs. $E_b/N_0$ (dB)]
Column splitting: Example

- (16383,14197) type-I cyclic 2-dimensional (0,7)th order EG code
- Split each column of $H$ into 32 new columns
- $C_{\text{ext}}$ is a code with $n = 524256$, $\rho = 128$, $\gamma_{\text{ext}} = 4$, $R = 0.97$, $r = 0.00024$

![Graph showing bit-error probability against $E_b/N_0$ (dB)](image)
Column splitting

- Column splitting of cyclic code -> extended code is usually not cyclic
- By starting with a parity check matrix consisting of $K \times n$ circulant submatrices, and splitting each column in a "rotating and circular" fashion into a fixed number of new columns, the extended code will be quasi-cyclic
- PG codes: $J$ may not be a multiple of $n$, so a modification of the above procedure is necessary
Row splitting

- Can be applied to any code
- Effects:
  - The parity check nodes in the Tanner graph are split into several nodes
  - The new code will have the following properties:
    - Same length as original code
    - More parity checks (higher $J$) and lower code rate
    - Column weight is unchanged. Any two columns will still have at most one 1 in common. The minimum distance ought to increase
    - The row weight is reduced
    - Cycles in the Tanner graph are broken
  - Can be combined with column splitting
Column + row splitting: Example

- (255,175) type-I cyclic 2-dimensional (0,4)th order EG code
- Split each column of $H$ into 5 new columns and each row into 2 rows
- $C_{\text{ext}}$ is a (1275,765) code with $\rho = 8$, column weights 3 and 4, $R = 0.6$, $r = 0.00627$

1.8 dB
Column + row splitting: Example

- (4095,3367) type-I cyclic 2-dimensional (0,6)th order EG code
- Split each column of $H$ into 16 new columns and each row into 3 rows
- $C_{ext}$ is a (65520,53235) code with row weights 21 and 22, and $\gamma = 4$
Cycle breaking

• Splitting rows and columns changes the Tanner graph

Column splitting
Row splitting
Cycle breaking

- Splitting columns to break a cycle of length 4

- Splitting rows to break a cycle of length 4
Cycle breaking

- Splitting columns to break a cycle of length 6
Effects of cycle breaking

- Increases the number of nodes and hence the complexity of the message passing algorithms (SPA/BF algorithm)
- Reduces the number of cycles and so improves decoding performance
Example of cycle breaking

- (7,4) Hamming code. Here: Cyclic version
Example of cycle breaking

- (7,4) Hamming code. Tanner graph
Example of cycle breaking

- (14,8) extended Hamming code. Performance
Example 2 of cycle breaking

- (23,12) Golay code. Here: Cyclic version

1748 4-cycles
Example 2 of cycle breaking

- (46,24) extended Golay code obtained by random splitting of columns. 106 cycles of length 4. The weight distribution is

<table>
<thead>
<tr>
<th>Weight</th>
<th>Number of codewords</th>
<th>Weight</th>
<th>Number of codewords</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
<td>23</td>
<td>1963152</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>24</td>
<td>1882241</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>25</td>
<td>1656075</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>26</td>
<td>1336148</td>
</tr>
<tr>
<td>4</td>
<td>1</td>
<td>27</td>
<td>989121</td>
</tr>
<tr>
<td>5</td>
<td>5</td>
<td>28</td>
<td>671322</td>
</tr>
<tr>
<td>6</td>
<td>18</td>
<td>29</td>
<td>417751</td>
</tr>
<tr>
<td>7</td>
<td>58</td>
<td>30</td>
<td>237105</td>
</tr>
<tr>
<td>8</td>
<td>280</td>
<td>31</td>
<td>122262</td>
</tr>
<tr>
<td>9</td>
<td>1000</td>
<td>32</td>
<td>57390</td>
</tr>
<tr>
<td>10</td>
<td>3151</td>
<td>33</td>
<td>24069</td>
</tr>
<tr>
<td>11</td>
<td>9034</td>
<td>34</td>
<td>9034</td>
</tr>
<tr>
<td>12</td>
<td>24069</td>
<td>35</td>
<td>3151</td>
</tr>
<tr>
<td>13</td>
<td>57390</td>
<td>36</td>
<td>1000</td>
</tr>
<tr>
<td>14</td>
<td>122262</td>
<td>37</td>
<td>280</td>
</tr>
<tr>
<td>15</td>
<td>237105</td>
<td>38</td>
<td>58</td>
</tr>
<tr>
<td>16</td>
<td>417751</td>
<td>39</td>
<td>18</td>
</tr>
<tr>
<td>17</td>
<td>671322</td>
<td>40</td>
<td>5</td>
</tr>
<tr>
<td>18</td>
<td>989121</td>
<td>41</td>
<td>1</td>
</tr>
<tr>
<td>19</td>
<td>1336148</td>
<td>42</td>
<td>0</td>
</tr>
<tr>
<td>20</td>
<td>1656075</td>
<td>43</td>
<td>0</td>
</tr>
<tr>
<td>21</td>
<td>1882241</td>
<td>44</td>
<td>0</td>
</tr>
<tr>
<td>22</td>
<td>46</td>
<td>45</td>
<td>1</td>
</tr>
</tbody>
</table>
Performance of (46,24) extended Golay code
Random LDPC codes

- LDPC codes reinvented by Mackay (1995) using random codes
- Create a matrix of $J$ rows and $n$ columns, with column weight $\gamma$ and row weight $\rho$
- $J$ is usually chosen equal to $n - k$
- In general, $\gamma n = \rho (n - k) + b$, where $b$ is the remainder
- Thus, we can choose $(n - k - b)$ rows of weight $\rho$ and $b$ rows of weight $\rho + 1$
Construction of random LDPC codes

- \( H_i = [h_1, \ldots, h_i] \) is a partial parity check matrix consisting of the first \( i \) columns

- *Initialization*: \( H_0 = \text{empty matrix}; \ i = 1; \ \text{Cand} = \text{set of all nonzero} \ (n - k)\)-dimensional column vectors of weight \( \gamma \)

- Let \( h \) be a random column from \( \text{Cand} \). Delete \( h \) from \( \text{Cand} \)

- Check whether \( h \) has more than one 1 in common with any column in \( H_{i-1} \) and if any (partial) row weight exceeds its maximum weight. If so, choose another \( h \) from \( \text{Cand} \) and repeat. Otherwise, proceed to the next step

- Set \( h_i = h \) and \( i = i + 1 \). If \( i \leq n \), then repeat

- Can use backtracking. Also, select parameters such that the number of weight-\( \gamma \) \((n - k)\)-tuples is \(>> n \). Also, it is possible to relax the requirements on row weights

- The actual rank of \( H \) may end up to be \((n - k') < (n - k)\)

- Efficient for small values of \( \rho \) and \( \gamma \), but the lower bound of \( \gamma + 1 \) on the minimum distance can be very poor
Irregular LDPC codes

- Use variable nodes of varying degrees and parity check nodes of varying degrees
- Degree distributions $\gamma(x) = \sum_i \gamma_i x^{i-1}$ and $\rho(x) = \sum_i \rho_i x^{i-1}$
- Approach: Density evolution / EXIT charts
- Create random LDPC codes according to distributions
- Optimum degree distributions often contain a large $\gamma_2$, which can lead to a poor minimum distance and high error floor
- But holds the world record: 0.0045 dB from the Shannon limit
- The approach assumes infinite block length and cycle free graphs
- These optimum degree distributions are not optimum for short codes in general
Improved irregular LDPC codes

- Extra design rules:
  - All degree 2 variable nodes are associated with parity symbols (if possible)
  - No length 4 cycles
  - No short cycles involving degree 2 variable nodes
  - Limit the number of degree 2 variable nodes
  - Degree redistribution
Graph theoretic LDPC codes

- Graph $G$
  - No self-loops
  - No multiple edges between a pair of vertices
- Let $P$ be a set of $n$ paths of length $\gamma$ that are pairwise disjoint or singularly crossing, and such that the union of nodes involved in the paths contains $J$ nodes
- Let $H$ be an incidence matrix of $P$, i.e., a $J \times n$ matrix such that $h_{i,j} = 1$ iff. node $i$ is on path $j$
- If each row of $H$ has constant weight $\rho$, then $H$ defines a $(\gamma, \rho)$-regular LDPC code
Graph theoretic LDPC codes: Example
Graph theoretic LDPC codes: Example
Graph theoretic LDPC codes: Example
We have skipped

- 17.5 PG LDPC codes
- 17.9 Shortened FG LDPC codes
- 17.10 Construction of Gallager LDPC codes
- 17.11 Masked EG-Gallager codes
- 17.12 Construction of QC LDPC codes
- 17.13 LDPC codes from FGs over GF($p^s$)
- 17.17 LDPC codes from BIBDs
- 17.18 LDPC codes from RS codes
- 17.19 Concatenations of LDPC and Turbo codes