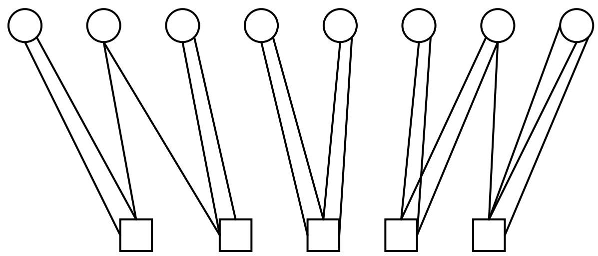
LDPC code construction example

Problem:

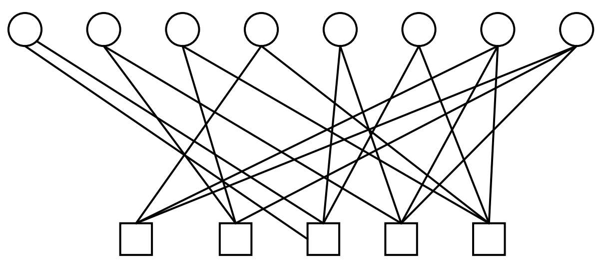
Let $L(x) = 0.7x^2 + 0.3x^3, R(x) = x^4$ be a normalized degree distribution (from node perspective).

- What is the design rate of the code?
  Solution:
  $$r = 1 - \frac{L'(1)}{R'(1)} = 1 - \frac{1.4 + 0.9}{4} = 0.425$$
- Create a random Tanner graph with 8 variable nodes according to the normalized degree distribution $(L(x), R(x))$.
  1. Create (non-normalized) degree distribution $\left(\Lambda(x), P(x)\right)$.
     Solution:
     $$\Lambda(x) = nL(x) = 5.6x^2 + 2.4x^3$$
     $$P(x) = (1 - r)nR(x) = 4.6x^4$$
     Round the coefficients to integers:
     $$\Lambda(x) = 6x^2 + 2x^3$$
     $$P(x) = 5x^4.$$
  2. Count the number of edges in the graph from $\Lambda(x)$ and $\mathrm{P}(x)$ and make sure they match.
     Solution: Number of edges going from the variable nodes is $\Lambda'^{(1)} = 6 * 2 + 2 * 3 = 18$. Number of edges going from the parity-check nodes is $\mathrm{P}'(1) = 5 * 4 = 20$. The numbers do not match, thus we need to adjust the degrees of the parity-check nodes. We do this by decreasing the degree in 2 parity-check nodes by 1, thus obtaining $P(x) = 2x^3 + 3x^4$. Always decrease or increase the number of edges by changing suitable number of parity-check nodes by at most one edge. Polynomial $\Lambda(x)$ stays the same. Now $\mathrm{P}'(1) = 2 * 3 + 3 * 4 = 18$ as we need. The Tanner graph will have 6 variable nodes of degree 2, 2 variable nodes of degree 3, 2 parity-check nodes of degree 3 and 3 parity-check nodes of degree 4.
  3. Let $m = \mathrm{P}'(1) = 18$ be the number of edges in the Tanner graph. Define the sockets at variable side as vector $v = (v_1, \dots, v_m)$, where v_i tells us that i-th edge in the graph is connected to v_i-th variable node. Similarly define sockets at parity-check side as vector $c = (c_1, \dots, c_m)$, where c_i tells us that i-th edge in the graph is connected to c_i-th parity-check node.
  4. Initialize both socket vectors so that the degrees in the graph match with $\Lambda(x)$, and $P(x)$.
     Solution: one simple and possible way is the following
     $$v = (1, 1, 2, 2, 3, 3, 4, 4, 5, 5, 6, 6, 7, 7, 7, 8, 8, 8)$$
     $$c = (1, 1, 1, 2, 2, 2, 3, 3, 3, 3, 4, 4, 4, 4, 5, 5, 5, 5)$$

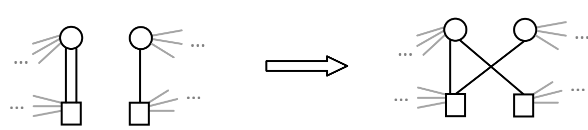
The corresponding Tanner graph to vectors v and c is as follows:



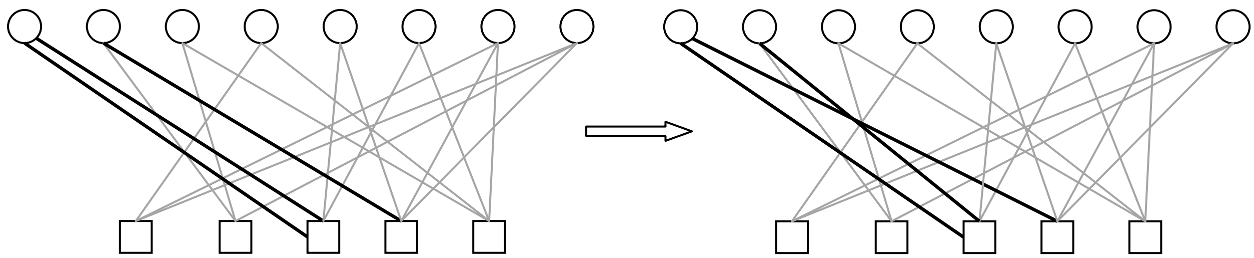
5. Take the random permutation σ on the set $\{1, \dots, P'(1)\}$. For example,
$$\sigma = (17, 15, 8, 16, 3, 5, 12, 2, 1, 9, 14, 4, 10, 18, 6, 11, 13, 7)$$
and permute the socket vector on the variable side to obtain the following vector
$$v' = (8, 7, 4, 8, 2, 3, 6, 1, 1, 5, 7, 2, 5, 8, 3, 6, 7, 4).$$
6. Construct the matrix H' (not necessarily parity-check) and Tanner graph using vectors v' and c.
Solution:
we have
$$v' = (8, 7, 4, 8, 2, 3, 6, 1, 1, 5, 7, 2, 5, 8, 3, 6, 7, 4)$$
$$c = (1, 1, 1, 2, 2, 2, 3, 3, 3, 3, 4, 4, 4, 4, 5, 5, 5, 5)$$



and in matrix form (number 2 means a double edge)
$$H' = \begin{pmatrix} 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 & 0 & 0 & 0 & 1 \\ 2 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 & 1 & 1 & 0 \end{pmatrix}$$
7. Resolve multi edges.
Solution:
Use the following diagram to resolve one double edge.



In our example, we have only one double edge between the first variable and the third check node. We need to find an arbitrary single edge between a different variable-check pair and reconnect the edges. For our example, as the single edge we take the edge between the second variable and the fourth check node.



and in the matrix representation

$$H' = \begin{pmatrix} 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 & 0 & 0 & 0 & 1 \\ \mathbf{2} & \mathbf{0} & 0 & 0 & 1 & 1 & 0 & 0 \\ \mathbf{0} & \mathbf{1} & 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 & 1 & 1 & 0 \end{pmatrix} \Rightarrow H = \begin{pmatrix} 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 & 0 & 0 & 0 & 1 \\ \mathbf{1} & \mathbf{1} & 0 & 0 & 1 & 1 & 0 & 0 \\ \mathbf{1} & \mathbf{0} & 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 & 1 & 1 & 0 \end{pmatrix}$$

Finally, the matrix

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

is the final, random parity-check matrix with the correct number of ones in its rows and columns.