Threshold Upper Bounds and Optimized Design of Protograph LDPC Codes for the Binary Erasure Channel

Surajkumar HARIKUMAR†, Jayanth RAMESH†, Manikandan SRINIVASAN†, Andrew THANGARAJ†

† Department of Electrical Engineering, Indian Institute of Technology Madras, Chennai 600036, India
E-mail: †andrew@ee.iitm.ac.in

Abstract Exact density evolution of protograph Low Density Parity Check (LDPC) codes over the Binary Erasure Channel (BEC) is considered. Upper bounds on the threshold are derived and expressed as single-variable minimizations. A simplified version of the upper bound is expressed in closed form in terms of the degrees of the nodes in the protograph. By maximizing the closed-form upper bound, useful conditions are derived for optimizing protographs to get thresholds close to capacity bounds. Using these conditions, a randomized construction method for good small-sized protographs is presented.

key words: *Protograph LDPC codes, Upper bound on threshold, Protograph optimization*

1. Introduction

Protograph Low Density Parity Check (LDPC) codes [1] are popular in theory in the form of spatially-coupled codes [2] [3], and widely adopted in many practical standards such as WiFi and DVB-S2. Therefore, the analysis and design of protograph LDPC codes is of interest and importance to researchers and practitioners. While asymptotic analysis in the form of spatial coupling is very popular, finite length analysis of protograph codes has not received significant attention beyond a few works such as [4] [5].

We consider the exact density evolution of protograph LDPC codes over the Binary Erasure Channel (BEC) [6], and derive upper bounds on the thresholds by bounding multivariate evolution functions with a single variable function. The threshold of the single variable evolution is bounded, and the upper bound is expressed in closed form in terms of the degrees of the nodes in the protograph. By maximizing the closed-form upper bound analytically, useful conditions on the entries of the protograph are derived for optimizing protographs and getting thresholds close to capacity bounds. Finally, we present a randomized construction method that exploits these conditions to generate good small-sized protographs.

To the best of our knowledge, bounds on thresholds of finite-sized protograph density evolution in terms of node degrees and design conditions for good thresholds in small protographs have not been studied or presented in earlier work.

1.1 Protograph LDPC codes

A protograph $G = (V \cup C, E)$ is a bipartite graph consisting of a set of N variable (or bit) nodes $V = \{v_1, v_2, \ldots, v_N\}$, a set of M check nodes $C = \{c_1, c_2, \ldots, c_M\}$, and a set of edges E. Multiple edges are possible between a variable node and a check node. The design rate of the protograph is $1 - M/N$. Let $N(v_i)$ and $N(c_j)$ denote the neighbors of the nodes v_i and c_j, respectively, in the protograph.

A protograph $G = (V \cup C, E)$ can be represented conveniently by an $M \times N$ base matrix H whose (j, i)-th entry $H(j, i)$ is the number of edges d_{ij} between v_i and c_j. For example, consider a base matrix

$$H_1 = \begin{bmatrix} 5 & 2 & 1 \\ 4 & 1 & 2 \end{bmatrix}. \tag{1}$$

It has $M = 2$ check nodes, and $N = 3$ variable nodes. The protograph corresponding to the above base matrix is shown in Fig. 1. We can obtain a larger protograph LDPC code by

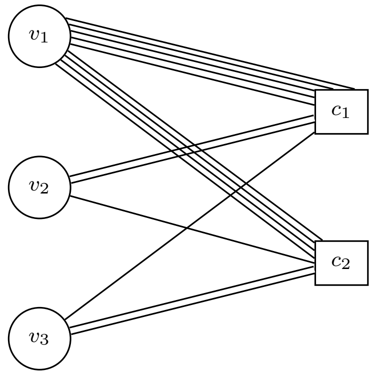

Fig. 1: Protograph for base matrix in (1).

the "copy-permute" operation [1]. The protograph is copied, say T times, and edges of the same type are permuted among the T copies. The *lifted graph* obtained by the copy-permute operation will have NT variable nodes, MT check nodes and $|E|T$ edges. For every edge $e \in E$ in the protograph, there are T edges of type e in the lifted Tanner graph.

1.2 Protograph density evolution over BEC

Consider the standard message passing decoder over a Binary Erasure Channel BEC(ϵ). Since every edge between variable node v_i and check node c_j in the protograph has the same neighbourhood, erasure probabilities of all parallel edge types connecting v_i to c_j evolve the same way. Let (i, j) denote a representative edge between v_i and c_j. Let $\mathcal{E} = \{(i, j) : d_{ij} > 0\}$ denote the set of edge types in the protograph.

Let $x_{ij}^{(l)}$ and $y_{ij}^{(l)}$ be the probabilities that an erasure is sent from variable to check node and check to variable node, respectively, along an edge of type $(i, j) \in \mathcal{E}$ in iteration l. In the first iteration, the respective received values from the channel are sent from the variable node to all connected check nodes. At a check node, an outgoing message is an erasure if at least one of the incoming messages is an erasure. At a variable node, an outgoing message is an erasure if the received value from the channel and all other incoming messages are erasures. Using these rules, the protograph density evolution recursion [6] over BEC(ϵ) is given by

$$\begin{aligned} x_{ij}^{(0)} &= \epsilon, \\ y_{ij}^{(l+1)} &= 1-(1-x_{ij}^{(l)})^{d_{ij}-1}\prod_{k\in N(c_j)\backslash v_i}(1-x_{kj}^{(l)})^{d_{kj}}, \quad (2) \\ x_{ij}^{(l+1)} &= \epsilon\ (y_{ij}^{(l+1)})^{d_{ij}-1}\prod_{k\in N(v_i)\backslash c_j}(y_{ik}^{(l+1)})^{d_{ik}}, \end{aligned}$$

for every edge type $(i,j)\in\mathcal{E}$. From the above equations, we can obtain a recursion of the form

$$\begin{aligned} x_{ij}^{(l+1)} &= \left(1-(1-x_{ij}^{(l)})^{d_{ij}-1}\prod_{k\in N(c_j)\backslash v_i}(1-x_{kj}^{(l)})^{d_{kj}}\right)^{d_{ij}-1} \\ &\prod_{k\in N(v_i)\backslash c_j}\left(1-(1-x_{ik}^{(l)})^{d_{ik}-1}\prod_{k'\in N(c_k)\backslash v_i}(1-x_{k'k}^{(l)})^{d_{k'k}}\right)^{d_{ik}}, \quad (3) \\ &\stackrel{\text{def}}{=} f_{ij}(\overline{x}_{ij}), \end{aligned}$$

where expressions for $y_{ij}^{(l+1)}$ and $y_{ik}^{(l+1)}$ from (2) have been used, and f_{ij} is defined as the multivariate polynomial in the RHS of (3) with $\overline{x}_{ij}$ being the vector of variables occurring in f_{ij}. We call f_{ij} the *update function* for the edge (i,j).

1.3 Properties of density evolution

The following three properties of the update functions $f_{ij}(\overline{x}_{ij})$ and the sequence $\{x_{ij}^{(l)}\}$ in the density evolution recursion for $(i,j)\in\mathcal{E}$ are standard and easy to establish:

1. $f_{ij}(\overline{x}_{ij})$ is increasing for each variable in $\overline{x}_{ij}$.
2. $x_{ij}^{(l+1)}\leqslant x_{ij}^{(l)}$, i.e the sequence $\left\{x_{ij}^{(l)}\right\}$ is decreasing.
3. For any $0\leqslant\epsilon\leqslant 1$, the sequence $\left\{x_{ij}^{(l)}\right\}$ converges.

Using the above three properties, it is easy to establish the *threshold phenomenon* of density evolution. The threshold of density evolution, denoted by ϵ^*, is defined as the largest value of ϵ below which all message error probabilities $x_{ij}^{(l)}$ tend to zero as l tends to infinity. Formally,

$$\epsilon^* = \sup\{\epsilon : x_{ij}^{(l)}\to 0 \text{ for all } e_{ij}\}. \quad (4)$$

As an example, the threshold ϵ^* of the protograph represented by the matrix H_1 in (1) is numerically computed using the density evolution recursions to be 0.4794.

2. Upper Bounds on Threshold

In this section, we describe protograph threshold upper bounds, whose evaluation is simple and requires a single-variable function minimization. First, we describe the all-edge bound, which applies to all protographs. This is followed by the single-edge bound, which requires certain conditions on the protograph structure for validity.

2.1 All-edge upper bound

Let $f_{ij}(x)=f_{ij}(x,\ldots,x)$ denote the single-variable function obtained by setting all variables in $\overline{x}_{ij}$ as x.

Lemma 1. *Let $g(x)$ be a monotone, increasing function for $x\in[0,1]$ such that $0\leqslant g(x)\leqslant f_{ij}(x)$ for all $(i,j)\in\mathcal{E}$. Let $\overline{\epsilon}^*$ be the threshold for the iteration $p^{(l+1)}=\epsilon g(p^{(t)})$ with $p^{(0)}=\epsilon$. Then, the following upper bound holds:*

$$\epsilon^*\leqslant\overline{\epsilon}^*=\min_{x\neq 0}\frac{x}{g(x)}. \quad (5)$$

Proof. We use induction to show that at every iteration l, $p^{(l)}\leqslant x_{ij}^{(l)}$ for all $(i,j)\in\mathcal{E}$. For $l=0$, we have the base case $p^{(0)}=x_{ij}^{(0)}=\epsilon$. As the induction hypothesis, suppose $p^{(l)}\leqslant x_{ij}^{(l)}$ for all $(i,j)\in\mathcal{E}$ at some iteration l. Then, for every $(i,j)\in\mathcal{E}$, we have

$$p^{(l+1)}=\epsilon g(p^{(l)})\stackrel{(a)}{\leqslant}\epsilon f_{ij}(p^{(l)})\stackrel{(b)}{\leqslant}\epsilon f_{ij}(\overline{x}_{ij}^{(l)})=x_{ij}^{(l+1)}, \quad (6)$$

where (a) follows from $g(x)\leqslant f_{ij}(x)$, and (b) follows because f_{ij} is increasing in each variable and by the induction assumption.

Now, since $p^{(l)}\leqslant x_{ij}^{(l)}$, we see that, for $\epsilon\leqslant\epsilon^*$, $p^{(l)}\to 0$ because $x_{ij}^{(l)}\to 0$. Therefore, we have $\epsilon^*\leqslant\overline{\epsilon}^*$.

Finally, the fixed point characterization of threshold [7] readily results in $\overline{\epsilon}^*=\min_{x\neq 0}\frac{x}{g(x)}$. ☐

To obtain a bound from Lemma 1, one choice of the function $g(x)$ is $g(x)\stackrel{\text{def}}{=}\prod_{i,j}f_{ij}(x)$. Since $0\leqslant f_{ij}(x)\leqslant 1$ and $f_{ij}(x)$ is increasing for $x\in[0,1]$, $g(x)$ satisfies the requirements of Lemma 1. Using (3) and expanding, we get

$$\begin{aligned}\overline{\epsilon}^* &= \min_{x\neq 0}\frac{x}{\prod_{i,j}f_{ij}(x)} \\ &= \min_{x\neq 0}\frac{x}{\prod_{j\in C}\left(1-(1-x)^{d(c_j)-1}\right)^{Md(c_j)-N}}, \quad (7)\end{aligned}$$

where $d(c_j)$ is the degree of the check node c_j in the protograph. Note that $\overline{\epsilon}^*$ depends only on M, N and the check node degrees. For our running example protograph represented by H_1, the upper bound is computed to be $\overline{\epsilon}^*=0.6324$.

2.2 Single-edge upper bound

While the all-edge bound is general, it can be improved in specific cases. One such case is the following.

Lemma 2. *(Single Edge Bound) Suppose there exists an edge $(a,b)\in\mathcal{E}$ satisfying the following conditions:*

- *Highest check-node degree: $d(c_b)\geqslant d(c_j)$ for $j\in C$.*
- *Fully-connected, dominant column: $d_{aj}>0$ and $d_{aj}\geqslant d_{ij}$ for $j\in C$, $i\in V$.*

Then, the following upper bound holds:

$$\epsilon^*\leqslant\overline{\epsilon}_{ab}^*=\min_{x\neq 0}\frac{x}{f_{ab}(x)}. \quad (8)$$

Proof. We will show that $f_{ab}(x) \leqslant f_{ij}(x)$ for all $(i,j) \in \mathcal{E}$, $x \in [0,1]$. Then, by the same proof as for Lemma 1, the bound is established.

First, note that the function $f_{ij}(x) = f_{ij}(x,\ldots,x)$ is of the form

$$f_{ij}(x) = \frac{\prod_{k\in C}\left(1-(1-x)^{d(c_k)-1}\right)^{d_{ik}}}{\left(1-(1-x)^{d(c_j)-1}\right)}. \tag{9}$$

For every $j \in C$, $(a,j) \in \mathcal{E}$ by the fully-connected column assumption. Since $d(c_b) \geqslant d(c_j)$ by the highest check-node degree assumption, we have

$$\left(1-(1-x)^{d(c_b)-1}\right) \geqslant \left(1-(1-x)^{d(c_j)-1}\right)$$

for $x \in [0,1]$. So, we get

$$\frac{\prod_k \left(1-(1-x)^{d(c_k)-1}\right)^{d_{ak}}}{\left(1-(1-x)^{d(c_b)-1}\right)} \leqslant \frac{\prod_k \left(1-(1-x)^{d(c_k)-1}\right)^{d_{ak}}}{\left(1-(1-x)^{d(c_j)-1}\right)},$$

which is rewritten, using (9), as

$$f_{ab}(x) \leqslant f_{aj}(x) \text{ for } j \in C. \tag{10}$$

Consider $i \in V$. We have

$$\prod_k \left(1-(1-x)^{d(c_k)-1}\right)^{d_{ak}} \leqslant \prod_k \left(1-(1-x)^{d(c_k)-1}\right)^{d_{ik}}, \tag{11}$$

because $1-(1-x)^{d(c_k)-1} \leqslant 1$ and $d_{ak} \geqslant d_{ik}$ by the dominant column assumption. Dividing both sides of (11) by $(1-(1-x)^{d(c_j)-1})$ for $j \in C$ such that $(i,j) \in \mathcal{E}$, we get

$$f_{aj}(x) \leqslant f_{ij}(x) \text{ for } (i,j) \in \mathcal{E}. \tag{12}$$

Combining (10) and (12), we have that $f_{ab}(x) \leqslant f_{ij}(x)$ for all $(i,j) \in \mathcal{E}$. □

The edge $(a,b) \in \mathcal{E}$ satisfying the conditions of Lemma 2 will be referred to as the *fastest converging* edge type. The conditions of Lemma 2 are illustrated in the next section.

2.3 Examples illustrating bounds

In the running example H_1 of (1), the edge e_{11} is readily seen to be the fastest converging edge type. The threshold is $\epsilon = 0.4794$, while the single-edge bound is $\bar{\epsilon}^*_{11} = 0.5374$, and the all-edge bound is $\bar{\epsilon}^* = 0.6324$.

Consider the optimized 4×8 protograph with base matrix H_2 reported in [8] given below:

$$H_2 = \begin{bmatrix} 1 & 3 & 2 & 3 & 4 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 & 5 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 4 & 0 & 4 & 1 \\ 1 & 0 & 1 & 0 & 6 & 1 & 0 & 0 \end{bmatrix}. \tag{13}$$

The fastest converging edge type is e_{15}, and the corresponding single-edge bound is $\bar{\epsilon}^*_{15} = 0.5533$, while the exact threshold is 0.478. We remark that there are many examples of protographs that do not have any edge satisfying the conditions of Lemma 2.

2.4 Closed-form bounds

In the previous section, we derived the all-edge bound and the single-edge bound involving a minimization. To get better analytical insight, we now obtain a closed-form bound explicitly in terms of the protograph degrees.

Consider the minimization

$$W_1 = \min_{x\neq 0} \frac{x}{\left(1-(1-x)^p\right)^q}, \tag{14}$$

which is involved in the upper bound on threshold. Using the inequality

$$\sqrt{x}\,\ln\frac{1}{x} \leqslant 1-x \leqslant e^{-x}, \quad x \in (0,1], \tag{15}$$

we get that

$$1-(1-x)^p \geqslant 1-e^{-px} \geqslant \sqrt{e^{-px}}\,\ln\frac{1}{e^{-px}}. \tag{16}$$

So, we can upper bound the minimization as

$$W_1 = \min_{x\neq 0} \frac{x}{\left(1-(1-x)^p\right)^q} \leqslant \min_{x\neq 0} \frac{1}{\left(p^q x^{q-1}\, e^{-pqx/2}\right)}. \tag{17}$$

Performing the minimization on the RHS above, we get

$$W_1 \leqslant \left(p^q\, e^{-(q-1)}\, \left[\frac{2(q-1)}{pq}\right]^{q-1}\right)^{-1}. \tag{18}$$

Using similar techniques, we can obtain an upper bound on the following minimization:

$$W_N = \min_{x\neq 0} \frac{x}{\prod_{i=1}^N \left(1-(1-x)^{p_i}\right)^{q_i}}, \tag{19}$$

where p_i, q_i are integers. This minimization is directly applicable for evaluating the single-edge upper bounds. Skipping the details for want of space, the final bound can be expressed as follows:

$$W_N \leqslant \left(e^{-q'}\, \left[\frac{2q'}{p'}\right]^{q'} \prod_{i=1}^N p_i^{q_i}\right)^{-1}, \tag{20}$$

where $q' = \sum_i q_i - 1$, and $p' = \sum_i p_i q_i$.

Finally, a very simple bound on W_N is the following:

$$W_N \leqslant \frac{x_0}{\prod_{i=1}^N \left(1-(1-x_0)^{p_i}\right)^{q_i}}, \tag{21}$$

where $x_0 \in (0,1)$ is some fixed value.

While the closed-form upper bound values are not very tight, they help us understand the properties of high threshold protographs, and guide our heuristic for improving protograph thresholds.

3. Design of Small Protographs

Characterizing the threshold exactly from density evolution is very difficult, and, instead, we work with the more tractable upper bounds derived in the previous section. Our goal is to derive heuristics for design of good protographs by trying to maximize the upper bounds. Interestingly, we find that maximizing the upper bound has the effect of improving the threshold in a wide variety of cases.

Since the single-edge bound is the tightest one we have derived, we consider only those protographs for which the single-edge bound holds in our design. That is, we consider protographs that have a dominant column. We begin with designing small-sized protographs and later develop heuristics from these designs for larger protographs.

3.1 Designing a 1×2 protograph

Consider the general 1×2 protograph base matrix given by $H_{G1} = [a, b]$. In order for this to be a valid protograph (with non-trivial threshold), the protograph entries must satisfy $a \geqslant 2, b \geqslant 2$. Without loss of generality, we can assume $a \geqslant b$, which implies that the first column is dominant. The single-edge upper bound, as given by Lemma 2 is

$$\epsilon^* \leqslant \bar{\epsilon}^*_{11} = \min_{x \neq 0} \frac{x}{\left(1-(1-x)^{a+b-1}\right)^{a-1}}. \tag{22}$$

Using the closed-form expression from (21),

$$\epsilon^* \leqslant \bar{\epsilon}^*_{\hat{1}1} \leqslant \frac{x_0}{\left(1-(1-x_0)^{p}\right)^{q}}, \tag{23}$$

where $p = a + b - 1$, $q = a - 1$. For a fixed value of a, it is easily seen that as b increases the upper bound increases. So for any value of a, the best choice of b is $b = 2$. This gives $p = a + b - 1 = q + 2$, and we find the value of q which maximizes the bound

$$\frac{x_0}{\left(1-(1-x_0)^{q+2}\right)^{q}}. \tag{24}$$

Assuming x_0 is a constant, we differentiate the above expression with respect to q and show that the derivative is negative for a suitably large choice of x_0. Therefore, the bound with that choice of x_0 is maximised by choosing q to be as small as possible. This results in $q = 2$ and, in turn, $a = 3$. Therefore, the best 1×2 protograph with a dominant column is $H_{B1} = [3, 2]$. The threshold of this protograph is $\epsilon^* = 0.4448$, while the capacity upper bound is 0.5.

3.2 Designing a 2×3 protograph

Consider a general 2×3 protograph base matrix given by

$$H_{G2} = \begin{pmatrix} a & b & c \\ d & e & f \end{pmatrix}. \tag{25}$$

Setting the first column of H_{G2} as dominant, we have $a \geqslant b, c$ and $d \geqslant e, f$. Without loss of generality, we can assume $a \geqslant d$. The single-edge upper bound from Lemma 2 is

$$\epsilon^* \leqslant \min_{x \neq 0} \frac{x}{\left(1-(1-x)^{a+b+c-1}\right)^{a-1}\left(1-(1-x)^{d+e+f-1}\right)^{d}}. \tag{26}$$

Summarizing all the requirements, we get the following conditions on the protograph:

- $a \geqslant b, c$ and $d \geqslant e, f$, for a dominant first column.
- $b + c + e + f \geqslant 5$, for avoiding degree-2 cycles [8].
- Minimize $e + f$ and $b + c$, since the bound in (26) decreases when $e + f$ or $b + c$ increase.

These conditions are satisfied for $b = 2, c = e = f = 1$. So, the class of "good" protographs has the form

$$H = \begin{pmatrix} a & 2 & 1 \\ d & 1 & 1 \end{pmatrix} \tag{27}$$

Now, the bound in (21) for the above protograph is given by

$$\frac{x_0}{\left(1-(1-x_0)^{a+2}\right)^{a-1}\left(1-(1-x_0)^{d+1}\right)^{d}}, \tag{28}$$

which is readily shown to be maximized for a suitably chosen x_0 by choosing $a = 2$ and $d = 1$. Therefore, the bound-maximizing 2×3 protograph is seen to be

$$H_{B2} = \begin{pmatrix} 2 & 2 & 1 \\ 1 & 1 & 1 \end{pmatrix}, \tag{29}$$

which has a numerically computed threshold $\epsilon^* = 0.5947$ in comparison to a capacity bound of 0.667.

4. Protograph Optimization

Generalizing the ideas from the previous section, we present conditions on the choice of protograph matrix entries that maximize the closed-form bound. These conditions are then enforced in a partially-random construction of protographs.

4.1 Maximizing the closed-form bound

Technically, maximizing the upper bound need not always result in better thresholds. However, in several examples, we find that maximizing the upper bound improves threshold. The conditions required for maximizing the closed-form upper bound of (20) are described next.

4.1.1 Dominant column

For the single-edge bound to apply, we require the dominant column condition. So, we consider only those protographs that have a dominant column in our design. This observation is supported by the fact that optimized protographs reported in the literature have a dominant column [8].

4.1.2 Sparsity of protograph

From the small protograph examples, maximizing the closed-form bound required that the sum of entries of the dominant column (or degree of the dominant bit node) is to be minimized. This can be readily extended to show that to maximize the bound in (20), we require $\sum_{i=1}^{N} q_i$ to be minimized. This implies that the dominant column entries in the protograph should be as low as possible, which in-turn forces the other entries of the protograph to be as small as possible. Overall, we observe that the rest of the protograph, other than the dominant column, is to be made as sparse as possible.

4.1.3 Other requirements

Additionally, we enforce the following three requirements:

1. *Block error threshold*: As mentioned before, degree-2 variable nodes should be cycle-free for block-error threshold in large-girth constructions as per [8].
2. *Connectivity*: Clearly, the protograph is required to be connected.
3. *Non-repetitive columns*: This is a heuristic condition observed from constructions.

4.2 Construction steps

An $M \times N$ protograph is represented by a base matrix $H = [h_1 \ H']$, where h_1 is the dominant first column and H' is sparse and of size $M \times (N-1)$. The following randomized algorithm generates protographs according to the conditions described above:

1. Random construction of H'
   - Each entry of H' is chosen from the set $\{0, 1, 2, 3\}$.
   - Each column of H' sums to 3.
   - H' should correspond to a connected graph.
   - H' has no degree-2 variable node cycle.
2. Search for dominant column h_1
   - Each entry of h_1 is chosen from the set $\{1, 2, 3, 4\}$.
   - In each row, the entry of h_1 should be greater than those of H'.

The above construction is repeated multiple times, and the protograph with the best threshold is retained.

5. Examples

Table 1 shows the sizes of protographs and their thresholds obtained by the construction ideas described in the previous section. We observe that the thresholds at even small sizes

Size of protograph	BEC threshold
2×4	0.43
3×6	0.4737
4×8	0.4795
10×20	0.4852

Table 1: Constructed protographs and thresholds.

are quite close to the capacity upper bound of 0.5. The previous reported methods such as [8] use genetic optimization algorithms, while the proposed method uses much simpler methods justified theoretically using upper bounds.

The optimized 2×4 and 3×6 protographs in Table 1 are as follows:

$$\begin{bmatrix} 3 & 3 & 1 & 1 \\ 3 & 0 & 2 & 1 \end{bmatrix}, \begin{bmatrix} 3 & 1 & 0 & 1 & 2 & 3 \\ 4 & 1 & 1 & 1 & 1 & 0 \\ 4 & 0 & 1 & 1 & 0 & 0 \end{bmatrix}$$

For 4×8 and larger protographs, a few further choices are made for the sparse part H'. About $M - 1$ degree-2 variable nodes are added in a tree structure without cycles. The 4×8 optimized protographs from Table 1 is as follows:

$$\begin{bmatrix} 4 & 1 & 0 & 0 & 1 & 0 & 0 & 1 \\ 4 & 1 & 1 & 0 & 0 & 0 & 1 & 1 \\ 3 & 0 & 1 & 1 & 2 & 2 & 2 & 0 \\ 4 & 0 & 0 & 1 & 0 & 1 & 0 & 1 \end{bmatrix}.$$

Additionally, the matrix $3I_{M\times M}$ is included as part of H' in the 10×20 case.

6. Conclusion

Using a closed-form upper bound on threshold, protograph LDPC codes were optimized for good thresholds. A simple randomized construction resulted in small-sized protographs with threshold close to capacity upper bounds. Analytical characterization of the gap between capacity and the upper bounds is an interesting subject for future work.

References

[1] J. Thrope, "Low-density parity-check (LDPC) codes constructed from protographs," *INP progress report*, vol. 53, no. 8, pp. 42 –154, Aug. 2005.

[2] S. Kudekar, T. Richardson, and R. Urbanke, "Threshold saturation via spatial coupling: Why convolutional LDPC ensembles perform so well over the BEC," *IEEE Trans. Inf. Theory*, vol. 57, no. 2, pp. 803 –834, Feb. 2011.

[3] D. G. Mitchell, M. Lentmaier, and D. J. Costello Jr, "Spatially coupled LDPC codes constructed from protographs," *arXiv preprint arXiv:1407.5366*, 2014.

[4] D. Divsalar, S. Dolinar, C. Jones, and K. Andrews, "Capacity-approaching protograph codes," *IEEE J. Sel. Areas Commun.*, vol. 27, no. 6, pp. 876 –888, Aug. 2009.

[5] G. Liva and M. Chiani, "Protograph LDPC code design based on EXIT chart analysis," in *IEEE GLOBECOM ,2007. Proceedings.*, 2007, p. 6.

[6] M. Lentmaier, M. Tavares, and G. Fettweis, "Exact erasure channel density evolution for protograph-based generalized LDPC codes," in *Information Theory, 2009. ISIT 2009. IEEE International Symposium on*, June 2009, pp. 566–570.

[7] T. Richardson and R. Urbanke, *Modern Coding Theory*. Cambridge University Press, 2008.

[8] A. Pradhan, A. Subramanian, and A. Thangaraj, "Deterministic constructions for large girth protograph LDPC codes," in *Information Theory Proceedings (ISIT), 2013 IEEE International Symposium on*, July 2013, pp. 1680–1684.