Optimization of Nested Array-based LDPC Codes Via Spatial Coupling

Salman Habib[†], David G. M. Mitchell[*], and Jörg Kliewer[†]
[†]Helen and John C. Hartmann Dept. of Electrical and Computer Engineering,
New Jersey Institute of Technology, Newark, NJ 07102
[*]Klipsch School of Electrical and Computer Engineering, New Mexico State University, Las Cruces, NM 88011

***Abstract*— Linear nested codes, where two or more sub-codes are nested in a global code, have been proposed as candidates for reliable multi-terminal communication. In this paper, we consider nested array-based spatially coupled LDPC codes and propose a line-counting based optimization scheme for minimizing the number of dominant absorbing sets in order to improve its performance in the high signal-to-noise ratio regime. The presented multi-step optimization process is applied first to one of the nested codes, then an optimization of the remaining nested codes is carried out based on these code constraints. We also show that for certain code parameters, dominant absorbing sets in the Tanner graphs of all nested codes can be completely removed using our proposed optimization strategy.**

I. Introduction

One application for linear nested codes is given by error correction in wireless multi-terminal networks, where M message sequences $\mathbf{u}_i$ of length k_i, $i = 1, 2, \ldots, M$, are encoded separately and then are algebraically superimposed via bitwise XOR at the physical layer prior to transmission [1], [2]. A codeword corresponding to each message vector, $\mathbf{x}_i \in \mathbb{F}_2^n$, belongs to a different *nested* sub-code $\mathscr{C}_i$ generated via a generator matrix $\mathbf{G}_i$. The transmitted codeword $\mathbf{x}$ is an element of the *global* code $\mathscr{C}$. The generator matrix $\mathbf{G}$ of the global code is obtained by stacking all individual $\mathbf{G}_i$ matrices [3].

Spatially coupled low density parity-check (SC-LDPC) codes [4] are sparse graph codes that are known to approach the capacity of binary input memoryless channels under belief propagation (BP) decoding [5]. SC-LDPC codes are obtained by coupling, or connecting, multiple Tanner graphs corresponding to an LDPC block code (LDPC-BC). The Tanner graph of LDPC-BCs contain small sub-structures called absorbing sets (ABSs) [6] that are known to cause the BP decoder to fail. These failures are responsible for the error-floor phenomenon, seen by a flattening of the bit error rate (BER) performance curve in the high signal-to-noise ratio (SNR) region. SC-LDPC codes have superior error floor performance when compared to their BC counterpart, mainly because spatial coupling breaks these harmful ABSs [7]. It is well known that 6-cycles exist in the dominant ABSs of an array-based (AB) SC-LDPC codes [8], [9], [10]. In our earlier work [10], a line-counting based algorithm was proposed to enumerate 6-cycles in AB-SC-LDPC codes, and thus can be used to minimize harmful ABSs in the code's Tanner graph.

In this paper, we propose an adapted line-counting (ALC) technique to optimize the design of nested AB-SC-LDPC codes. The objective of a finite length nested code design is to ensure that each nested sub-code and the global code have a small number of dominant ABSs in the Tanner graph when compared to the underlying LDPC-BC. Since the parity-check matrices of different nested sub-codes partially overlap, the null-spaces for these sub-codes intersect. Consequently, an optimization of one nested sub-code affects other nested sub-codes, and thus impose constraints on the optimization. Since multiple design constraints must now be jointly satisfied, the construction of nested LDPC-BCs for multi-terminal setting is more challenging than the one for point-to-point LDPC-BCs. In contrast to the line counting approach proposed in [10], the presented ALC technique allows the enumeration of 6-cycles in arbitrary column weight 3 sub-matrices in polynomial time, facilitating a tractable nested code optimization. Note that the construction of nested regular and irregular LDPC-BCs have been considered in [3] and [11], respectively. However, to the best of our knowledge, an extension to nested AB-SC-LDPC codes has not been addressed in the open literature so far.

This material is based on work supported by the National Science Foundation under Grant Nos. ECCS-1710920, ECCS-1711056, and OIA-1757207.

II. Preliminaries

A. Protograph and Array-Based LDPC Codes

Let $G_{\mathbf{H}} = (V \cup C, E)$ denote the bipartite Tanner graph corresponding to a parity-check matrix $\mathbf{H} \in \mathbb{F}_2^{m \times n}$, where $V = \{v_1, v_2, \ldots, v_n\}$ is a set of n variable nodes (VNs), $C = \{c_1, c_2, \ldots, c_m\}$ is a set of m check nodes (CNs), and E is the set of edges connecting V to C. LDPC-BCs can be designed based on a protograph [12], which is a small Tanner graph consisting of p VNs and γ CNs, $p \geq \gamma$, with design rate $R = 1 - \frac{\gamma}{p}$. Let $\mathbf{B} = [B_{i,j}]$ represent the *base* matrix corresponding to the Tanner graph of protograph $G_{\mathbf{B}}$. By applying a graph lifting procedure with lifting factor p, that is by replacing each non-zero entry of $\mathbf{B}$ with a sum of $B_{i,j}$ $p \times p$ non-overlapping permutation matrices, and each zero entry with an all-zero matrix $B_{i,j}$ of size $p \times p$, we can construct a LDPC-BC parity-check matrix $\mathbf{H} \in \mathbb{F}_2^{\gamma p \times p^2}$. In the case of an AB-LDPC-BC, $\mathbf{B}$ is a $\gamma \times p$ all-ones matrix, p is prime, and the permutations used in edge-spreading are specific shifted identity matrices, also known as circulant matrices [13]. The resulting block matrix structure of an (γ, p) AB-LDPC-BC parity-check matrix is represented as

$$\mathbf{H}(\gamma, p) = \begin{bmatrix} \mathbf{I} & \mathbf{I} & \mathbf{I} & \cdots & \mathbf{I} \\ \mathbf{I} & \sigma & \sigma^2 & \cdots & \sigma^{p-1} \\ \vdots & \vdots & \vdots & \cdots & \vdots \\ \mathbf{I} & \sigma^{\gamma-1} & \sigma^{2(\gamma-1)} & \cdots & \sigma^{(\gamma-1)(p-1)} \end{bmatrix},$$

where the circulant σ^z is obtained by circularly left-shifting the non-zero entries of the identity matrix $\mathbf{I}$ by an amount $z \bmod p$. Here, $\sigma^0 = \mathbf{I}$, and each circulant has dimension $p \times p$. Each row (resp. column) of sub-matrices form a row (resp. column) group. There are a total of p (resp. p^2) column groups

978-1-5386-6900-6/19/$31.00 ©2019 IEEE

(resp. columns) and γ (resp. γp) row groups (resp. rows) in $\mathbf{H}(\gamma, p)$. Note that γ is also the column weight of $\mathbf{H}(\gamma, p)$.

B. Array-Based Spatially Coupled LDPC Codes

A protograph approach to the construction of SC-LDPC codes involves coupling L copies of the base Tanner graph $G_{\mathbf{B}}$ via edge-spreading [14]. In terms of matrices, edge-spreading is equivalent to splitting $\mathbf{B}$ into a sum of $m+1$ matrices of the same dimension as $\mathbf{B}$, such that $\mathbf{B} = \mathbf{B}_0 + \mathbf{B}_1 + \cdots + \mathbf{B}_m$, where m denotes the memory of the code. Similar to the case of LDPC-BCs, the parity-check matrix of a terminated SC-LDPC code $\mathbf{H}(\gamma, p, L) \in \mathbb{F}_2^{\gamma p(L+m)\times Lp^2}$ is obtained by graph lifting the terminated base matrix $\mathbf{B}_{sc}$. In terms of AB-SC-LDPC codes, the edge-spreading process splits $\mathbf{H}(\gamma, p)$ into a sum of $m+1$ matrices of the same dimension as $\mathbf{H}(\gamma, p)$, yielding $\mathbf{H}(\gamma, p) = \mathbf{H}_0 + \mathbf{H}_1 + \cdots + \mathbf{H}_m$ [8]. These matrices are arranged as

$$\mathbf{H}(\gamma, p, L) = \begin{bmatrix} \mathbf{H}_0 & & \\ \mathbf{H}_1 & \ddots & \\ \vdots & \ddots & \mathbf{H}_0 \\ \mathbf{H}_m & & \mathbf{H}_1 \\ & \ddots & \vdots \\ & & \mathbf{H}_m \end{bmatrix}, \tag{1}$$

where L is the number of column blocks of $\mathbf{H}(\gamma, p, L)$ and the code has a *constraint length* of $(m+1)p^2$. AB-SC-LDPC matrices can be obtained from AB-LDPC-BC matrices via a spreading matrix $\mathbf{B}_m \in \mathbb{F}_{m+1}^{\gamma\times p}$, where an entry $f \in \{0, \ldots, m\}$ in position (i, j) of this matrix indicates that the circulant block (i, j) of $\mathbf{H}(\gamma, p)$ is copied to its corresponding position in $\mathbf{H}_f$ [8]. In the remainder of this paper, LDPC codes are considered as AB only.

C. Nested Codes

A nested code consists of a group of M sub-codes $\mathscr{C}_i$, $i = 1, 2, \ldots, M$, $M \geq 2$, nested in a global code $\mathscr{C}$ of rate k/n with the property $\mathscr{C}_i \subset \mathscr{C}$. Nested codes are used to jointly encode M different information vectors $\mathbf{u}_i \in \mathbb{F}_2^{k_i}$, $k_i < k$, to generate an overall codeword $\mathbf{x} \in \mathbb{F}_2^n$, which is a linear combination of all the codewords $\mathbf{x}_i$ obtained from each of the sub-codes. Nested codes can also be interpreted via the parity check matrices $\mathbf{H}_i \in \mathbb{F}_2^{(n-k_i)\times n}$ and $\mathbf{H} \in \mathbb{F}_2^{(n-k)\times n}$ corresponding to the codes $\mathscr{C}_i$ and $\mathscr{C}$, respectively, which form the null spaces of matrices $\mathbf{H}_i$ and $\mathbf{H}$, respectively. Note that the matrices $\mathbf{H}_i$, for all i, and $\mathbf{H}$ can be considered as (potentially overlapping) sub-matrices of a larger $\hat{\mathbf{H}} \in \mathbb{F}_2^{m\times n}$ matrix, where $n - k_i < m \leq n\ \forall i$ and $n - k < m \leq n$, respectively. The construction of $\mathbf{H}_i$ and $\mathbf{H}$ from $\hat{\mathbf{H}}$ is discussed in detail in [3]. We refer to the column weight ω_i (resp. ω) of sub-code $\mathscr{C}_i$ (resp. global code $\mathscr{C}$) as the column weight of its corresponding (regular) parity-check matrix $\mathbf{H}_i$ (resp. $\mathbf{H}$).

D. Absorbing Sets

In the Tanner graph G of an LDPC code, let $X \subseteq V$ and let $O(X)$ be the set of neighboring CNs of X with odd degree.

Definition 1 ([6]). *For $a \geq 0, b \geq 0$, an (a, b) ABS set X is a set of VNs with $|X| = a$, $|O(X)| = b$, and the property that each VN in X has strictly fewer neighbors in $O(X)$ than in $C \setminus O(X)$. An (a, b) ABS is a fully ABS if, additionally, all nodes in $V \setminus X$ have strictly more neighbors in $C \setminus O(X)$ than in $O(X)$. A minimal ABS refers to an ABS which has the smallest possible existing value for a, in a given LDPC Tanner graph, and where b is the smallest possible value for the given a.*

Remark 1. *For $\gamma = 3$, an $(3, 3)$ ABS is the smallest ABS existing in a column weight 3 AB-LDPC-BC [6] or AB-SC-LDPC code [8]. The ABS contains a 6-cycle; therefore by eliminating these cycles via spatial coupling, we also eliminate all $(3, 3)$ ABSs from the Tanner graph of $\gamma = 3$ AB-SC-LDPC codes [8], [10]. For $\gamma = 4$, a $(4, 4)$ ABS is a minimal ABS in an AB-LDPC code for $p = 5, 7$; on the other hand, a $(5, 4)$ ABS is a minimal ABS in an AB-LDPC-BC for the case $7 < p < 19$; a $(6, 4)$ ABS exists in an AB-LDPC-BC for $p > 5$, and it is a minimal ABS for the case $p > 19$ [6].*

E. Line-counting

We briefly review the line-counting method of [10] in the following. Let $q_t \in \{0, \ldots, \gamma - 1\}$ and $s_\ell \in \{0, \ldots, p - 1\}$ denote the row group number, and the row number within a particular row group of an AB-LDPC matrix, respectively. Also, let $j_\ell \in \{0, \ldots, p - 1\}$ and $k_u \in \{0, \ldots, p - 1\}$ represent the column group number and the column number inside a particular column group of an AB-LDPC matrix $\mathbf{H}(\gamma, p)$. Therefore, each row (resp. column) of an AB matrix is given by $r_t = q_t p + s_\ell$ (resp. $c_\ell = j_\ell p + k_u$). In this way, the location of an entry of an AB matrix (r_t, c_ℓ) may be written as $(q_t, s_v; j_\ell, k_u)$, $t, \ell, u, v \in \{1, 2, 3\}$. A 6-cycle spans three distinct row and column groups of an AB matrix [10]. The columns of a 6-cycle have indices c_1, c_2, c_3 and they exist in distinct column groups j_1, j_2, j_3, respectively. Similarly, the rows of a 6-cycle have indices r_1, r_2, r_3 and they exist in distinct row groups q_1, q_2, q_3, respectively.

In an column weight 3 AB-LDPC (sub) matrix, a region $\mathcal{R}$ (or block cycle) consists of at least six (not necessarily contiguous) circulant matrices spread across three row groups, with one row group being a row group of $\mathbf{I}$ matrices only. W.l.o.g, for any 6-cycle in $\mathcal{R}$, we have the following [10]:

- $c_2 > c_1$ and $w_1 p \leq c_1 < w_2 p$, $w_3 p \leq c_2 < w_4 p$, where w_1, w_2, w_3, w_4 are integers satisfying $0 \leq w_1 \leq p - 1$, $1 \leq w_2 \leq p - 1$, $w_1 + 1 \leq w_3 \leq p$, and $w_2 + 1 \leq w_4 \leq p$. In the case where c_1 and c_2 are associated with a row group of $\mathbf{I}$ matrices, we obtain $c_2 - c_1 = np$, where $n = \{1, \ldots, w_4 - w_1 - 1\}$;
- $\alpha p \leq c_3 < \beta p$, where α and β are integers satisfying $0 \leq \alpha \leq p - 1$, $1 \leq \beta \leq p$, and $\alpha < \beta$.

Note that even after spatial coupling, the location of circulant matrices in $\mathcal{R}$ can be described by their corresponding positions in the block matrix $\mathbf{H}(\gamma, p, L)$. The range of c_3 in $\mathcal{R}$ can also be expressed using c_1, c_2 as [10]

$$\begin{gathered} \frac{\alpha p}{2} \leq c_2 - \frac{1}{2}c_1 < \frac{\beta p}{2}, \quad \frac{p^2 + \alpha p}{2} \leq c_2 - \frac{1}{2}c_1 < \frac{p^2 + \beta p}{2}, \\ p^2 - \beta p < c_2 - 2c_1 \leq p^2 - \alpha p, \text{ or } \quad -\beta p < c_2 - 2c_1 \leq -\alpha p. \end{gathered} \tag{2}$$

A 6-cycle with columns c_1 and c_2 exists in $\mathcal{R}$ if an integer point on the line $c_2 - c_1 = np$ in the (c_1, c_2) plane lies within the boundaries described by the line inequalities in (2). Hence, the number of 6-cycles in $\mathcal{R}$ is determined by the number of integer pairs (c_1, c_2) on the line $c_2 - c_1 = np$ enclosed within these boundaries [10].

III. Nested AB-SC-LDPC Codes

A. Example Construction with $M = 2$

We begin with an example of an $M = 2$ nested AB code construction to demonstrate our approach, although generalization to larger M can be achieved in a similar fashion. Consider $\hat{\mathbf{H}} = \mathbf{H}(5, p)$ with corresponding parity-check matrix

$$\mathbf{H}(5,p) = \begin{bmatrix} \mathbf{I} & \mathbf{I} & \mathbf{I} & \cdots & \mathbf{I} \\ \mathbf{I} & \sigma & \sigma^2 & \cdots & \sigma^{p-1} \\ \mathbf{I} & \sigma^2 & \sigma^4 & \cdots & \sigma^{2(p-1)} \\ \mathbf{I} & \sigma^3 & \sigma & \cdots & \sigma^{3(p-1)} \\ \mathbf{I} & \sigma^4 & \sigma^3 & \cdots & \sigma^{4(p-1)} \end{bmatrix}.$$

We can form nested AB-LDPC-BC sub-codes $\mathscr{C}_1$, $\mathscr{C}_2$, and $\mathscr{C}$ from sub-matrices $\mathbf{H}_1(4, p)$, $\mathbf{H}_2(4, p)$, and $\mathbf{H}(3, p)$, respectively, where the matrices are comprised of the first four (black and blue), first three and fifth (black and red), and first three (black only) row groups of $\mathbf{H}(5, p)$, respectively. Given a global AB nested code of column weight 3 (e.g., $\mathbf{H}(3, p)$), it follows from the array structure [13] that there are $\gamma - \omega_i + 1$, $\gamma \geq 5$, possible sub-codes $\mathscr{C}_i$ with column weight $\omega_i \geq 4$. Note that the design rate $R_i = 1 - \frac{\omega_i}{p}$ of $\mathscr{C}_i$ decreases with ω_i. As a result, constructing high rate nested codes with large ω_i also requires a relatively large p compared to a nested code with smaller ω_i. In turn, this also makes each sub-code optimization more computationally intensive as M increases.

Nested AB-SC-LDPC matrices can be constructed in a similar way, with parity-check matrices denoted as $\mathbf{H}_i(4, p, L)$ and $\mathbf{H}(3, p, L)$ obtained by edge-spreading $\mathbf{H}_i(4, p)$ and $\mathbf{H}(3, p)$ via spreading matrices $\mathbf{B}_{i,m} \in \mathbb{F}_{m+1}^{4\times p}$, $i \in \{1, 2\}$, and $\mathbf{B}_m \in \mathbb{F}_{m+1}^{3\times p}$, respectively. Note that since the codes are nested, $\mathbf{B}_m$ is a sub-matrix of $\mathbf{B}_{i,m}$. For the case $M = 2$, the optimization of nested AB-SC-LDPC codes can be performed in two ways:

Method 1: First, optimize $\mathbf{B}_m$ to construct $\mathbf{H}(3, p, L)$ from $\mathbf{H}(3, p)$. Then optimize the last row of $\mathbf{B}_{1,m}$ (resp. $\mathbf{B}_{2,m}$) by incorporating the constraints given by $\mathbf{B}_m$ to construct $\mathbf{H}_1(4, p, L)$ from $\mathbf{H}_1(4, p)$ (resp. $\mathbf{H}_2(4, p, L)$ from $\mathbf{H}_2(4, p)$).

Method 2: First, optimize one of the $\mathbf{B}_{i,m}$ matrices to construct $\mathbf{H}_i(4, p, L)$ from $\mathbf{H}_i(4, p)$ and then by using the constraints, optimize the remaining parity-check matrix. For example, we may first optimize the $\mathbf{B}_{1,m}$ matrix to construct $\mathbf{H}_1(4, p, L)$ from $\mathbf{H}_1(4, p)$ and optimize the last row of $\mathbf{B}_{2,m}$ to construct $\mathbf{H}_2(4, p, L)$ from $\mathbf{H}_2(4, p)$.

B. Terminal Lift

To simplify our code search for good nested code designs, we choose to apply an (additional) circulant-based terminal lift with lifting factor J. This results in quasi-cyclic LDPC-BCs (QC-LDPC-BCs) and QC-SC-LDPC codes. QC codes are well known to facilitate hardware implementation [4]. The parity-check matrices obtained by applying a terminal lift to $\mathbf{H}_i(4, p, L)$ and $\mathbf{H}(3, p, L)$ are denoted as $\mathbf{H}_i(4, p, L, J) \in \mathbb{F}_2^{4pJ(L+m)\times JLp^2}$, $i = 1, 2$, and $\mathbf{H}(3, p, L, J) \in \mathbb{F}_2^{3pJ(L+m)\times JLp^2}$, respectively. A terminal lift serves two purposes: firstly it helps us generate sufficiently long nested codes to achieve good performance for typical applications, and secondly, we are able to further reduce the multiplicity of, or even eliminate, any residual ABSs in the nested Tanner graphs of $\mathbf{H}_i(4, p, L)$ and $\mathbf{H}(3, p, L)$ that remain after the edge-spreading construction.

C. 6-cycles and $(6, 4)$ ABSs in AB-LDPC Codes

We now state some useful results on dominant objects of AB-SC-LDPC codes.

Lemma 1. *For $a = 4, 5, 6$, an $(a, 4)$ ABS in an AB-LDPC code with parity-check matrix $\mathbf{H}(4, p)$ and an AB-SC-LDPC code with parity-check matrix $\mathbf{H}(4, p, L)$ contains at least one 6-cycle.*

The proof is omitted due to space constraints.

Remark 2. *From Lemma 1 we deduce that by eliminating all 6-cycles via spatial coupling, we can also eliminate all $(a, 4)$ ABSs from the Tanner graph of column weight 4 nested AB-SC-LDPC codes, where $a = 4, 5, 6$.*

Note that a 6-cycle in an AB parity-check matrix spans 3 row groups [8] and it spans at most $m+1$ contiguous column blocks of (1). Let μ_ℓ, $\ell = 1, \ldots, m + 1$, represent the total number of 6-cycles present in precisely ℓ contiguous column blocks of (1) and let μ be the total number of 6-cycles in $\mathbf{H}(\gamma, p, L)$. From the repeated structure of $\mathbf{H}(\gamma, p, L)$ it follows that the total number of 6-cycles in $\mathbf{H}(\gamma, p, L)$ is $\mu = \sum_{\ell=1}^{m+1}(L - \ell + 1)\mu_\ell$ [8].

D. Adapted Line Counting

The line-counting method discussed in Section II-E is restricted to the enumeration of 6-cycles in $\mathbf{H}(3, p, L)$ only, i.e., row groups 0, 1, and 2. To overcome this shortcoming, we propose an adapted line-counting (ALC) method that is generalized to enumerate 6-cycles from other row group triples of the larger parity-check matrix $\hat{\mathbf{H}}$. By constrained optimization of the entries of $\mathbf{B}_{i,m}$ and $\mathbf{B}_m$ matrices via ALC, we can obtain $\mathbf{H}_i(4, p, L)$ and $\mathbf{H}(3, p, L)$ matrices, respectively, whose Tanner graphs contain as few $(6, 4)$ and $(3, 3)$ ABSs as possible, respectively. The conventional cycle counting method discussed in [15] has complexity $O(gE^2/p) = O(g\gamma^2 p^3)$, where g is the girth and $E = \gamma p^2$ is the number of edges in the graph, respectively. In comparison, ALC has complexity $O(p^2)$ and hence is more desirable for the optimization of nested codes.

1) ALC Based Optimization for $M = 2$: We consider the three column weight 3 sub-matrices of $\mathbf{H}_i(4, p)$, denoted as $\mathbf{H}_i^{(z)}(3, p)$, $z = 1, 2, 3$, where $\mathbf{H}_i^{(1)}(3, p)$, $\mathbf{H}_i^{(2)}(3, p)$, and $\mathbf{H}_i^{(3)}(3, p)$ consist of row groups $(0, 1, 2)$, $(0, 2, 3)$, and $(0, 1, 3)$ of $\mathbf{H}_i(4, p)$, respectively.[1] Let $\mathbf{H}_i^{(z)}(3, p, L) \in \mathbb{F}_2^{3p(L+m)\times Lp^2}$ be a SC-LDPC matrix obtained by edge-spreading $\mathbf{H}_i^{(z)}(3, p)$ via a sub-matrix $\mathbf{B}'_{i,m} \in \mathbb{F}_{m+1}^{3\times p}$ of the spreading matrix $\mathbf{B}_{i,m}$, whose row indices correspond to the row group indices of $\mathbf{H}_i^{(z)}(3, p)$. For example, if $z = 2$ then $\mathbf{B}'_{i,m}$ is comprised of rows $(0, 2, 3)$ of $\mathbf{B}_{i,m}$.

With ALC (details to follow in Section III-D2), 6-cycles can be enumerated for any of the $\mathbf{H}_i^{(z)}(3, p, L)$ matrices. The global code can be optimized by searching over the set of edge-spreading matrices $\mathbf{B}'_{i,m}$ for $\mathbf{H}_i^{(1)}(3, p, L)$ (any i) via numerical optimization [10], where the number of 6-cycles is the optimization criteria, evaluated by ALC. To optimize $\mathbf{H}_i(4, p, L)$, we pick $z_1, z_2 \in \{0, 1, 2\}$, $z_1 \neq z_2$, and first optimize $\mathbf{H}_i^{(z_1)}(3, p, L)$. Two row groups of $\mathbf{H}_i^{(z_2)}(3, p, L)$ are now fixed

[1] Note that there is one more row group, namely $(1, 2, 3)$. However it is sufficient to examine only the three groups described above since they are sufficient to form the parity-check matrices of all the nested codes.

and we minimize the number of 6-cycles in $\mathbf{H}_i^{(z_2)}(3,p,L)$ by determining the edge-spreading for the remaining (unfixed) row group of z_2. In general, this approach can be repeated as many times as necessary to sequentially optimize the edge-spreading matrices row group by row group.

2) Description for ALC: Recall from Section II-E that $\mathcal{R}$ is contained in a column weight 3 sub-matrix of an AB parity-check matrix, and c_1, c_2, c_3 are the column indices of a 6-cycle in $\mathcal{R}$.

Lemma 2. *For ALC, the range of c_3 in $\mathcal{R}$ can be expressed via c_1 and c_2 as*

$$\left(1-\frac{q_3}{q_2}\right)\alpha p+\frac{\lambda p^2}{q_2} \le c_2-\frac{q_3}{q_2}c_1 < \left(1-\frac{q_3}{q_2}\right)\beta p+\frac{\lambda p^2}{q_2}, \qquad (3)$$

where $c_2 > c_1$, $\lambda \in \{2-2p,\ldots,2p-2\}$, $q_2, q_3 \in \{1,\ldots,p-1\}$ and $q_2 \neq q_3$.

Sketch of Proof. Recall that a circulant matrix $\sigma^{q_t j_\ell}$ in an AB code has its non-zero elements located at (s_t, k_u), where $s_t \equiv q_t j_\ell + k_u \mod p$. W.l.o.g., let row groups q_2 and q_3 of an AB matrix consist of $\sigma^{q_2 j_\ell}$ and $\sigma^{q_3 j_\ell}$ circulant matrices, respectively, where $q_2 \neq q_3$. The edges corresponding to (r_2, c_3) and (r_2, c_2) both exist in row s_2. Hence, from the value of s_2 we obtain the relation $q_2 j_2 + k_2 \equiv q_2 j_3 + k_3 \mod p$. Similarly, from the value of s_3 we may obtain $q_3 j_1 + k_2 \equiv q_3 j_3 + k_3 \mod p$. As a result, from the values of s_2 and s_3 we obtain relations $k_3 - k_2 = q_2 j_2 - q_2 j_3 - \lambda_1 p$ and $k_3 - k_2 = q_3 j_1 - q_3 j_3 - \lambda_2 p$, respectively, where $\lambda_1, \lambda_2 \in \{1-p,\ldots,p-1\}$. Since the left hand sides of both of these relations are identical, we can equate the right hand side expressions, and after rearranging, we get $j_3 = \frac{q_2 j_2 - q_3 j_1 - \lambda p}{q_2 - q_3}$, where $\lambda = \lambda_1 - \lambda_2$, $\lambda \in \{2-2p,\ldots,2p-2\}$. Finally, by invoking the inequality $\alpha \le j_3 \le \beta - 1$ and by taking into account the corresponding 6-cycle column value c_3 as a function of c_1 and c_2, we obtain (3). □

Note that (2) is a special case of (3) because, for parameters (λ, q_2, q_3) taking on the values $(0,2,1)$, $(1,2,1)$, $(1,1,2)$, and $(0,1,2)$, we recover the first, second, third, and fourth inequality in (2), respectively. Based on the principles of Cartesian geometry, the total number of 6-cycles in $\mathcal{R}$ for a given λ, $\mathcal{N}_{\lambda,\mathcal{R}}$, can be derived using the method discussed in [10]. Eventually, the total number of 6-cycles obtained via ALC in a AB parity-check matrix region $\mathcal{R}$ is given by $\sum_\lambda \mathcal{N}_{\lambda,\mathcal{R}}$ with computational complexity $O(p^2)$.

IV. AB-SC-LDPC Nested Code Optimization Procedure

In this section, we outline the strategy for optimizing the AB-SC-LDPC nested codes for $M = 2$. First, a numerical optimization scheme is employed to determine the entries of the edge-spreading matrices $\mathbf{B}_{i,m}$ and $\mathbf{B}_m$ to construct parity-check matrices $\mathbf{H}_i(4,p,L)$ and $\mathbf{H}(3,p,L)$ from $\mathbf{H}_i(4,p)$ and $\mathbf{H}(3,p)$, respectively. ALC is used in each optimization step for computing the number of 6-cycles in the underlying AB-SC-LDPC parity-check matrices. Finally, a terminal lift is applied to further reduce residual ABSs in the Tanner graphs of optimized $\mathbf{H}_i(4,p,L)$ and $\mathbf{H}(3,p,L)$ matrices.

A. Step 1: ALC Based Optimization

We outline the approach of Method 1 to first optimize $\mathbf{H}(3,p,L)$, and then optimize $\mathbf{H}_i(4,p,L)$ based on the constraints given by $\mathbf{H}(3,p,L)$.

- Inputs to the ALC based optimization algorithm are the $\mathbf{H}(3,p)$ matrix and the empty $\mathbf{B}_m$ matrix. The optimization algorithm is allowed to run until either the number of 6-cycles in $\mathbf{H}(3,p,L)$ obtained in an optimization iteration is 0, or until the number of optimization iterations exceed a predetermined threshold $I_{\max}$, whichever occurs first. At the end of this step, we obtain $\mathbf{B}_m$, and the spreading for row groups $(0,1,2)$ are fixed.
- For a given i, inputs to the ALC based optimization algorithm are $\mathbf{H}_i^{(2)}(3,p)$ and matrix $\mathbf{B}'_{i,m}$ which will form rows $(0,2,3)$ of $\mathbf{B}_{i,m}$. Note that the first two rows of $\mathbf{B}'_{i,m}$ are initialized with rows $\{0,2\}$ of $\mathbf{B}_m$, whereas the last row of $\mathbf{B}'_{i,m}$ is empty and will be determined in this step. The optimization algorithm is now allowed to run until either all the 6-cycles in $\mathbf{H}_i^{(2)}(3,p,L)$, obtained by edge-spreading $\mathbf{H}_i^{(2)}(3,p)$ using $\mathbf{B}'_{i,m}$, are eliminated or the number of optimization iteration exceed a threshold $I_{\max}^{(i)}$, whichever occurs first. At the end of this step, we obtain the complete $\mathbf{B}_{i,m}$ matrix. The procedure is then repeated for the other sub-code i.
- The $\mathbf{H}_i(4,p,L)$ and $\mathbf{H}(3,p,L)$ matrices are now obtained by edge-spreading according to $\mathbf{B}_{i,m}$.

Method 2 largely follows the approach in Method 1, but here $\mathbf{H}_i(4,p,L)$ is optimized first (as described in Section III-A). The details are omitted due to space constraints.

B. Step 2: Terminal Lift

The terminal lifted matrices $\mathbf{H}_i(4,p,L,J)$, $i = 1,2$, and $\mathbf{H}(3,p,L,J)$ may now be obtained by lifting the non-zero (resp. zero) entries of $\mathbf{H}_i(4,p,L)$ and $\mathbf{H}(3,p,L)$ via randomly generated circulant (resp. all-zero) matrices of size $J \times J$, respectively. The goal of the terminal lift is to break any remaining 6-cycles in the Tanner graphs of $\mathbf{H}_i(4,p,L)$ and $\mathbf{H}(3,p,L)$. In this paper, we used a straightforward approach to randomly select circulants for $m+1$ contiguous column blocks of (1) for $\mathbf{H}_1(4,p,L,J)$ until either all 6-cycles were eliminated or a maximum time duration elapsed. These permutations were repeated periodically to form the complete $\mathbf{H}_1(4,p,L,J)$ matrix (where four of the five row groups are fixed). Note that the global code is now fixed. We complete the construction by selecting circulants to optimize the fifth row group. The optimized QC-SC-LDPC nested codes $\mathscr{C}_1$, $\mathscr{C}_2$, and $\mathscr{C}$ are now obtained as the null spaces of $\mathbf{H}_1(4,p,L,J)$, $\mathbf{H}_2(4,p,L,J)$, and $\mathbf{H}(3,p,L,J)$, respectively.

V. Numerical Results

We now demonstrate the effectiveness of the procedure outlined in Section IV by providing 6-cycle enumeration results for $\mathbf{H}(3,p,L)$ and $\mathbf{H}_i(4,p,L)$, $i = 1,2$, for syndrome former memories $m = 1,2$. We begin with the ALC based optimization described in Section IV-A, i.e., the following results are obtained before applying a terminal lift. As a benchmark for comparing our optimization results we also enumerate 6-cycles in the matrix $\mathbf{H}_{unc}(3,p,L)$ (resp. $\mathbf{H}_{i,unc}(4,p,L)$), which is a block diagonal matrix in the form of (1) with $m = 0$ (an uncoupled or non-optimized matrix) containing L diagonally placed blocks of $\mathbf{H}(3,p)$ (resp. $\mathbf{H}_i(4,p)$) AB-LDPC-BC parity-check matrices. Note from Section III-C that it is sufficient to enumerate the quantities μ_1, μ_2 (resp. μ_1, μ_2, μ_3) in order to determine the total number of 6-cycles μ in an entire AB-SC-LDPC matrix with $m = 1$ (resp. $m = 2$) for an arbitrary L. Also, note that for the uncoupled matrices, it is sufficient to only compute μ_1 since 6-cycles can only span a single column block

of (1), then $\mu = \mu_1 L$. For $\mathbf{H}_{unc}(3, p)$ (resp. $\mathbf{H}_{i,unc}(4, p)$, $i = 1, 2$), the values of μ_1 are 300, 882, and 3630 (resp. 1200, 3528, and 14520) for $p = 5, 7$, and 11, respectively. The enumeration results in Table I and II provide results for nested AB-SC-LDPC matrices constructed based on ALC based optimization Methods 1 and 2, respectively. For all tables, plain text corresponds to $m = 1$ and bold font corresponds to $m = 2$. Finally, the results in Table III present the average values of μ_1, μ_2, and μ_3 for randomly generated (non-optimized) AB-SC-LDPC nested matrices, where randomly generated $\mathbf{B}_m$ and $\mathbf{B}_{i,m}$ matrices were selected.

p	$\mathbf{H}(3, p, L)$ $(\mu_1, \mu_2), (\mu_1, \mu_2, \mu_3)$	$\mathbf{H}_1(4, p, L)$ $(\mu_1, \mu_2), (\mu_1, \mu_2, \mu_3)$	$\mathbf{H}_2(4, p, L)$ $(\mu_1, \mu_2), (\mu_1, \mu_2, \mu_3)$
5	(0, 0), (**0, 0**)	(30, 40), (**20, 0, 0**)	(40, 50), (**0, 0, 10**)
7	(14, 28), (**0, 0, 0**)	(70, 161), (**28, 63, 49**)	(133, 196), (**42, 42, 56**)
11	(22, 99), (**0, 0, 0**)	(407, 737), (**128, 187, 275**)	(374, 825), (**77, 143, 253**)

TABLE I: Values of μ_1, μ_2, μ_3 obtained for different nested SC-LDPC matrices using Method 1.

p	$\mathbf{H}(3, p, L)$ $(\mu_1, \mu_2), (\mu_1, \mu_2, \mu_3)$	$\mathbf{H}_1(4, p, L)$ $(\mu_1, \mu_2), (\mu_1, \mu_2, \mu_3)$	$\mathbf{H}_2(4, p, L)$ $(\mu_1, \mu_2), (\mu_1, \mu_2, \mu_3)$
5	(5, 10), (**0, 0, 0**)	(70, 50), (**0, 0, 0**)	(35, 55), (**0, 0, 0**)
7	(14, 7), (**0, 0, 28**)	(84, 84), (**0, 7, 63**)	(91, 126), (**21, 35, 91**)
11	(44, 176), (**0, 33, 55**)	(220, 770), (**55, 110, 143**)	(319, 792), (**44, 253, 275**)

TABLE II: Values of μ_1, μ_2, μ_3 obtained for different nested SC-LDPC matrices using Method 2.

p	$\mathbf{H}(3, p, L)$ $(\mu_1, \mu_2), (\mu_1, \mu_2, \mu_3)$	$\mathbf{H}_1(4, p, L)$ $(\mu_1, \mu_2), (\mu_1, \mu_2, \mu_3)$	$\mathbf{H}_2(4, p, L)$ $(\mu_1, \mu_2), (\mu_1, \mu_2, \mu_3)$
5	(6.67, 26.7), (**5, 10, 1.7**)	(33.3, 93.4), (**10, 41.7, 14.9**)	(35, 80), (**20, 36.7, 11.6**)
7	(23.3, 49.1), (**3.5, 21, 7**)	(127, 271), (**65, 117, 42**)	(135, 229), (**40, 125, 52**)
11	(132, 271), (**55, 66, 84**)	(587, 989), (**169, 370, 278**)	(513, 1046), (**191, 333, 297**)

TABLE III: Average values of μ_1, μ_2, μ_3 obtained for randomly generated nested AB-SC-LDPC matrices.

The results show that the uncoupled matrices contain a large number of 6-cycles (and hence dominant ABSs). By comparing the results in Tables I and II, we notice that, as expected, spatial coupling is able to significantly reduce the number of dominant ABSs. Further, Method 1 results in a lower multiplicity for the global SC-LDPC matrix $\mathbf{H}(3, p, L)$, but in a generally higher multiplicity for the nested SC-LDPC matrices. Method 2, on the other hand, shows the opposite behavior - a reduced multiplicity for the nested codes, but a relatively higher multiplicity for the global code. From Table II, we also note that for $p = 5$ there are no 6-cycles present in either column weight 3 or column weight 4 nested codes, which, by invoking Remarks 1 and 2, implies the absence of $(3, 3)$ and $(4, 4)$ ABSs, respectively. From Table III we observe that randomly generated nested SC-LDPC matrices contain a significantly larger number of 6-cycles (and hence dominant ABSs) when compared to nested AB-SC-LDPC matrices obtained via ALC based optimization.

For $m = 2$, Fig. 1 shows the number of 6-cycles versus the coupling length L for nested AB-SC-LDPC matrices $\mathbf{H}(3, p, L)$ and $\mathbf{H}_1(4, p, L)$, respectively, obtained via Method 1, Method 2, and a random generation. We again observe that randomly generated nested AB-SC-LDPC matrices contain more 6-cycles (and hence more dominant ABSs) in both column weight 3 and column weight 4 cases compared to nested matrices obtained via Method 1 or 2.

To conclude, we note that by applying the terminal lift described in Section IV-B, even for small J, it is possible to significantly reduce, or even eliminate dominant ABSs. For example, with $p = 5, 7$, $m = 1, 2$, and $p = 11, m = 2$, and a lifting factor of only $J = 5$, we are able to completely eliminate all 6-cycles. This means that any $(3, 3)$, $(4, 4)$, $(5, 4)$ and $(6, 4)$ ABSs are broken after a terminal lift of the AB-SC-LDPC nested codes given in Tables I and II.

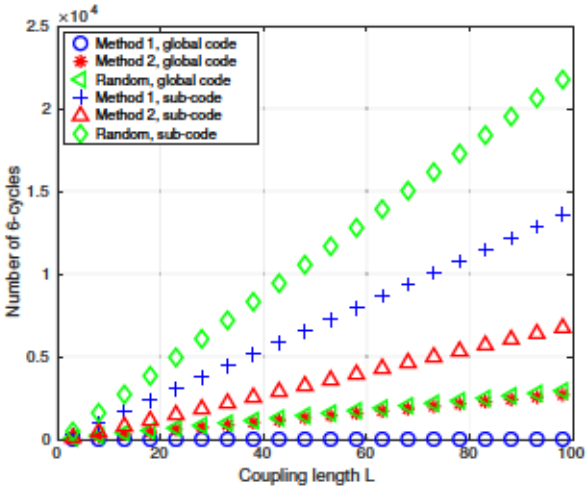

Fig. 1: Total number of 6-cycles versus L for global code corresponding to $\mathbf{H}(3, p, L)$ and sub-code corresponding to $\mathbf{H}_1(4, p, L)$ for $m = 2$ and $p = 7$. The number of 6-cycles in the global code obtained via Method 1 is zero.

References

[1] L. Xiao, T. E. Fuja, J. Kliewer, and D. J. Costello, Jr., "A network coding approach to cooperative diversity," *IEEE Trans. Inf. Theory*, vol. 53, no. 10, pp. 3714–3722, Oct 2007.

[2] J. Kliewer, T. Dikaliotis, and T. Ho, "On the performance of joint and separate channel and network coding in wireless fading networks," *Proc. IEEE Inf. Theory Workshop (ITW)*, pp. 1–5, 2007.

[3] C. A. Kelley and J. Kliewer, "Algebraic constructions of graph-based nested codes from protographs," *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, pp. 829–833, June 2010.

[4] D. J. Costello, Jr., L. Dolecek, T. Fuja, J. Kliewer, D. G. M. Mitchell, and R. Smarandache, "Spatially coupled sparse codes on graphs: theory and practice," *IEEE Comms. Mag.*, vol. 52, no. 7, pp. 168–176, 2014.

[5] S. Kudekar, T. Richardson, and R. L. Urbanke, "Spatially coupled ensembles universally achieve capacity under belief propagation," *IEEE Trans. Inf. Theory*, vol. 59, no. 12, pp. 7761–7813, Dec 2013.

[6] L. Dolecek, Z. Zhang, V. Anantharam, M. Wainright, and B. Nikolic, "Analysis of absorbing sets and fully absorbing sets of array-based LDPC codes," *IEEE Trans. Inf. Theory*, vol. 56, pp. 181–201, Jan. 2010.

[7] D. G. M. Mitchell, L. Dolecek, and D. J. Costello, Jr., "Absorbing set characterization of array-based spatially coupled LDPC codes," *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, pp. 886–890, June 2014.

[8] D. G. M. Mitchell and E. Rosnes, "Edge spreading design of high rate array-based SC-LDPC codes," *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, pp. 2940–2944, July 2017.

[9] H. Esfahanizadeh, A. Hareedy, and L. Dolecek, "Finite-length construction of high performance spatially-coupled codes via optimized partitioning and lifting," *IEEE Trans. on Comm.*, vol. 67, pp. 3–16, August 2018.

[10] A. Beemer, S. Habib, C. Kelley, and J. Kliewer, "A generalized algebraic approach to optimizing SC-LDPC codes," *Proc. 55th Allerton Conf. on Communication, Control, and Computing*, pp. 672–679, Oct. 2017.

[11] M. Grimes and D. G. M. Mitchell, "Design of nested protograph-based LDPC codes with low error-floors," *Proc. IEEE 8th Annual Comp. and Comm. Workshop and Conf. (CCWC)*, pp. 575–579, 2018.

[12] J. Thorpe, "Low-density parity-check (LDPC) codes constructed from protographs," *IPN Progress Report 42-154*, pp. 1–7, August 2003.

[13] J. L. Fan, "Array codes as low-density parity-check codes," *Proc. 2nd Intl. Symp. Turbo Codes and Rel. Topics*, pp. 543–546, Sep 2000.

[14] D. G. M. Mitchell, M. Lentmaier, and D. J. Costello, Jr., "Spatially coupled LDPC codes constructed from protographs," *IEEE Trans. Inf. Theory*, vol. 61, no. 9, pp. 4866–4889, Sep. 2015.

[15] J. Li, S. Lin, and K. Abdel-Ghaffar, "Improved message-passing algorithm for counting short cycles in bipartite graphs," *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, pp. 416–420, Jun. 2015.