Spatially Coupled LDPC Codes Constructed From Protographs

Mitchell, David G.M.; Lentmaier, Michael; Costello Jr., Daniel J.

Published in:
IEEE Transactions on Information Theory

DOI:
10.1109/TIT.2015.2453267

2015

Spatially Coupled LDPC Codes Constructed From Protographs

David G. M. Mitchell, *Member, IEEE*, Michael Lentmaier, *Senior Member, IEEE*,
and Daniel J. Costello, Jr., *Life Fellow, IEEE*

***Abstract*— In this paper, we construct protograph-based spatially coupled low-density parity-check (LDPC) codes by coupling together a series of L disjoint, or uncoupled, LDPC code Tanner graphs into a single coupled chain. By varying L, we obtain a flexible family of code ensembles with varying rates and frame lengths that can share the same encoding and decoding architecture for arbitrary L. We demonstrate that the resulting codes combine the best features of optimized irregular and regular codes in one design: capacity approaching iterative belief propagation (BP) decoding thresholds and linear growth of minimum distance with block length. In particular, we show that, for sufficiently large L, the BP thresholds on both the binary erasure channel and the binary-input additive white Gaussian noise channel saturate to a particular value significantly better than the BP decoding threshold and numerically indistinguishable from the optimal maximum *a posteriori* decoding threshold of the uncoupled LDPC code. When all variable nodes in the coupled chain have degree greater than two, asymptotically the error probability converges at least doubly exponentially with decoding iterations and we obtain sequences of asymptotically good LDPC codes with fast convergence rates and BP thresholds close to the Shannon limit. Further, the gap to capacity decreases as the density of the graph increases, opening up a new way to construct capacity achieving codes on memoryless binary-input symmetric-output channels with low-complexity BP decoding.**

***Index Terms*— Low-density parity-check (LDPC) codes, LDPC convolutional codes, spatially coupled codes, iterative decoding, belief propagation, density evolution, decoding thresholds, minimum distance, capacity achieving codes.**

I. Introduction

THE performance of an iterative belief propagation (BP) decoder for low-density parity-check (LDPC) codes is strongly influenced by the degrees of the different variable nodes and check nodes in the associated Tanner graph code representation [1]. (J, K)-*regular* LDPC codes, with constant variable node degree J and check node degree K, as originally proposed by Gallager [2] in 1962, are *asymptotically good* in the sense that their minimum distance grows linearly with block length for $J > 2$; however, the iterative decoding behavior of regular codes in the so-called *waterfall*, or moderate bit error rate (BER), region of the performance curve falls short of capacity, making them unsuitable for severely power-constrained applications, such as uplink cellular data transmission or digital satellite broadcasting systems.

On the other hand, optimized *irregular* LDPC codes [3], with a variety of different node degrees, exhibit capacity approaching performance in the waterfall but, unlike (J, K)-regular codes, are normally subject to an *error floor*, a flattening of the BER curve that results in poor performance at high signal-to-noise ratios (SNRs), as a result of a large number of degree two variable nodes; this makes such codes undesirable for applications that require very low decoded BERs, such as data storage and optical communication. For irregular LDPC code ensembles, the degrees of the variable and check nodes are often modeled as random variables that are characterized by their *degree distributions* $\lambda(x)$ and $\rho(x)$, respectively [3]. Each coefficient in the polynomials $\lambda(x)$ and $\rho(x)$ corresponds to the fraction of edges in the graph connected to nodes of a certain degree. Gallager's (J, K)-regular LDPC code ensembles correspond to the special case $\lambda(x) = x^{J-1}$ and $\rho(x) = x^{K-1}$, *i.e.*, the degrees of each node type are constant. Using an algorithm called *density evolution* (DE) [4], a BP decoding threshold can be calculated for a randomly constructed LDPC code ensemble with degree distribution pair $(\lambda(x), \rho(x))$ that determines the limit of the error-free region asymptotically as the block length tends to infinity. Using DE, irregular code ensembles with thresholds very close to the Shannon limit on the binary-input additive white Gaussian noise channel (AWGNC) were designed in [5]. Moreover, in [6], capacity achieving sequences of degree distribution pairs for a given rate R with a vanishing gap between the threshold and the Shannon limit $\varepsilon_{Sh} = 1 - R$ were presented for the binary erasure channel (BEC).

LDPC convolutional codes (LDPC-CCs) [7], the convolutional counterparts of LDPC block codes (LDPC-BCs), have been shown to have certain advantages compared to LDPC-BCs [8], [9]. Variations in the check and variable node

Manuscript received June 23, 2014; revised February 24, 2015; accepted May 29, 2015. Date of publication July 8, 2015; date of current version August 14, 2015. This work was supported by the National Science Foundation under Grant CCF-1161754. Parts of this paper were presented at the 2009 Information Theory and Applications Workshop, the 2010 Information Theory and Applications Workshop, the 2010 IEEE International Symposium on Information Theory, the 2010 International Symposium on Turbo Coding and Iterative Information Processing, the 2011 Information Theory and Applications Workshop, and the 2011 IEEE International Symposium on Information Theory.

D. G. M. Mitchell was with the Department of Electrical Engineering, University of Notre Dame, Notre Dame, IN 46556 USA. He is now with the Klipsch School of Electrical and Computer Engineering, New Mexico State University, Las Cruces, NM 88003 USA (e-mail: david.mitchell@nd.edu).

M. Lentmaier is with the Department of Electrical and Information Technology, Lund University, Lund 221 00, Sweden (e-mail: michael.lentmaier@eit.lth.se).

D. J. Costello, Jr., is with the Department of Electrical Engineering, University of Notre Dame, Notre Dame, IN 46556 USA (e-mail: costello.2@nd.edu).

Communicated by H. D. Pfister, Associate Editor for Coding Theory.

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TIT.2015.2453267

degrees of LDPC-CC Tanner graphs are also characterized by a degree distribution pair, where the connections between nodes in the bi-infinite Tanner graph occur within a *constraint length*. The performance of LDPC-CCs under iterative BP decoding has been well studied. Extensive computer simulation results (see [7], [10]–[12]) have verified that, for practical code lengths, LDPC-CCs obtained by *unwrapping* an LDPC-BC achieve a substantial *convolutional gain* compared to the underlying LDPC-BC, where both codes have the same computational complexity with iterative decoding and the block length of the LDPC-BC equals the constraint length of the LDPC-CC. Moreover, various code and graph properties, such as iterative decoding thresholds [13]–[15], girth [12], [16], minimum (free) distance [10], [17]–[19], minimum (free) pseudo-distance [20], and minimum trapping set size [19], of the unwrapped LDPC-CC have been shown to be at least as good as the corresponding values of the underlying LDPC-BC.

Spatially coupled LDPC (SC-LDPC) codes are constructed by coupling together a series of L disjoint, or uncoupled, LDPC code Tanner graphs into a single coupled *chain*. They can be viewed as a type of LDPC-CC, since spatial coupling is equivalent to introducing memory into the encoding process. If the coupled chain is *unterminated* ($L \to \infty$), a SC-LDPC convolutional code (SC-LDPC-CC) is formed, and if the chain is terminated (finite L), a SC-LDPC block code (SC-LDPC-BC) results. Recently, it has been proven by Kudekar *et al.* that SC-LDPC-BC ensembles are capacity achieving on memoryless binary-input symmetric-output (MBS) channels under BP decoding [21], [22]. Consequently, the principle of spatial graph coupling has attracted significant attention and has been successfully applied in many other areas of communications and signal processing, such as, for example, compressed sensing [23], [24], relay channels [25]–[28], wiretap channels [29], multiple access channels [30]–[33], broadcast channels [34], intersymbol-interference channels [35], [36], multiuser detection [37], random access [38], source coding [39], quantum codes [40], [41], and models in statistical physics [42]. Also, studies of the finite length scaling properties of SC-LDPC-BCs were performed in [43] and [44] and block erasure channel performance bounds were given in [45].

LDPC code ensembles with a certain predefined *structure* can be constructed by means of *protographs* [46]. By applying a graph lifting operation, Tanner graphs of various sizes can be constructed that preserve the rate, degree distribution, and *computation graphs* (see [47]) of the protograph. It has been observed that irregular protograph-based LDPC-BC ensembles often have better thresholds than *unstructured* irregular ensembles with the same degree distributions [48]. An extreme example of this behavior is that the thresholds of carefully designed protograph-based LDPC code ensembles containing variable nodes of degree one can have good thresholds [48]; whereas an unstructured LDPC code ensemble with degree one variable nodes will not even have a threshold. Moreover, the inherent structure in protograph-based ensembles can improve distance properties. For example, irregular protograph-based LDPC-BC ensembles that contain degree two variable nodes can be asymptotically good, and ensembles with minimum variable node degree three can provide a good trade-off between distance and threshold [48]. As a result of their good properties and implementation advantages, many LDPC codes have been adopted in recent industry standards, such as wireless LANs (IEEE 802.11n), WiMax (IEEE 802.16e), digital video broadcasting (DVB-S2), and the ITU-T standard for networking over power lines, phone lines, and coaxial cable (G.hn/G.9960), and each of these standard codes can be viewed as protograph-based LDPC-BCs.

In this paper, we analyze ensembles of SC-LDPC-BCs constructed from protographs. We present an *edge spreading* procedure that is used to couple together L block protographs to form a convolutional protograph. The protograph framework enables us to extend previous DE analysis [13]–[15] and codeword weight enumerator analysis [17] that were restricted to certain (J, K)-regular SC-LDPC-CC ensembles to general (J, K)-regular and irregular ensembles. We use this analysis to show that, for protograph-based SC-LDPC-BC ensembles with sufficiently large L, the iterative BP decoding thresholds on both the BEC and the AWGNC *saturate* to a particular value significantly larger than the BP decoding threshold and numerically indistinguishable from the maximum a-posteriori (MAP) decoding threshold of the underlying LDPC-BC ensemble.[1] Further, we show that both the (J, K)-regular SC-LDPC-BC ensembles with $J > 2$ and the irregular SC-LDPC-BC ensembles considered in this paper are asymptotically good, *i.e.*, their minimum distance grows linearly with block length. Thus, since the MAP thresholds of (J, K)-regular LDPC-BC ensembles approach capacity as the graph density increases, protograph-based SC-LDPC-BC ensembles combine the best features of optimized irregular and regular codes in one design: capacity approaching BP decoding thresholds *and* linear minimum distance growth. Finally, we study the relationship between the minimum distance growth rate of the SC-LDPC-BC ensemble and the free distance growth rate of the associated SC-LDPC-CC ensemble.

The paper is structured as follows. In Section II, we give a brief review of LDPC-BCs and the protograph construction method. We then describe the construction and structural properties of protograph-based SC-LDPC-CC and SC-LDPC-BC ensembles. In Section III, we begin with an asymptotic weight enumerator analysis of protograph-based SC-LDPC-BC ensembles. We then proceed to study their iterative decoding properties by means of DE, first for the BEC and then for the AWGNC. As the coupling length L increases, we obtain a family of asymptotically good code ensembles with increasing rates that feature a trade-off between capacity approaching iterative decoding thresholds and declining minimum distance growth rates. Then, in Section IV, we show that the minimum distance growth rates, while declining with L, converge to a bound on the free distance growth rate of the unterminated SC-LDPC-CC

[1] It should be noted that the SC-LDPC-BC ensembles used to prove threshold saturation in [21] and [22] had to be suitably randomized, and thus those results do not apply directly to the protograph-based SC-LDPC-BC ensembles considered in this paper.

ensemble that is independent of L and significantly larger than the minimum distance growth rate of the underlying LDPC-BC ensemble. We then argue that an appropriate distance measure for terminated SC-LDPC-CC (*i.e.*, SC-LDPC-BC) ensembles should also behave independently of L. Some concluding remarks are given in Section V.

II. Spatially Coupled LDPC Code Ensembles

In this section, we will describe the construction of protograph-based SC-LDPC code ensembles. We begin with a brief introduction to LDPC codes in Section II-A and review the construction of LDPC code ensembles based on protographs in Section II-B. In Section II-C we discuss the construction of a *convolutional protograph* and the associated ensemble of protograph-based SC-LDPC-CCs by applying an edge spreading operation to (spatially) couple together a sequence of uncoupled LDPC-BC protographs. In Section II-D, we present two closely related ways to construct SC-LDPC-BCs from protograph-based SC-LDPC-CCs: termination and tail-biting. We conclude with a discussion of variations to the edge spreading rule and different ways of constructing SC-LDPC-BC ensembles in Section II-E.

A. LDPC Block Codes

We begin with a brief introduction to LDPC-BCs (see also [47], [49]). A (J, K)*-regular* LDPC-BC is defined as the null space of a *sparse* binary parity-check matrix $\mathbf{H}$, where each row of $\mathbf{H}$ contains exactly K ones, each column of $\mathbf{H}$ contains exactly J ones, and both J and K are small compared with the number of rows in $\mathbf{H}$. An LDPC-BC code is called *irregular* if the row and column weights are not constant. The code has *block length* n, where n is the number of columns of $\mathbf{H}$, and *rate* $R = k/n$, where $(n - k)$ is the rank of $\mathbf{H}$. For (J, K)-regular codes, the code rate is given as $R \geq 1 - J/K$, with equality when $\mathbf{H}$ has full rank. It is often useful to represent the parity-check matrix $\mathbf{H}$ using a bipartite graph called the *Tanner graph* [1]. In the Tanner graph representation, each column of $\mathbf{H}$ corresponds to a *code bit* or *variable node* and each row corresponds to a *parity-check* or *check node*. If position (i, j) of $\mathbf{H}$ is equal to one, then check node i is connected by an *edge* to variable node j in the Tanner graph; otherwise, there is no edge connecting these nodes. The notion of *degree distribution* is used to characterize the variations of check and variable node degrees (see [3]).

B. Protograph-Based Code Construction

A *protograph* [46] with *design rate* $R = 1 - n_c/n_v$ is a small bipartite graph (V, C, E) that connects a set of n_v variable nodes $V = \{v_0, \ldots, v_{n_v-1}\}$ to a set of n_c check nodes $C = \{c_0, \ldots, c_{n_c-1}\}$ by a set of edges E.[2] We assume $n_v > n_c$ so that the protograph has a strictly positive design rate.

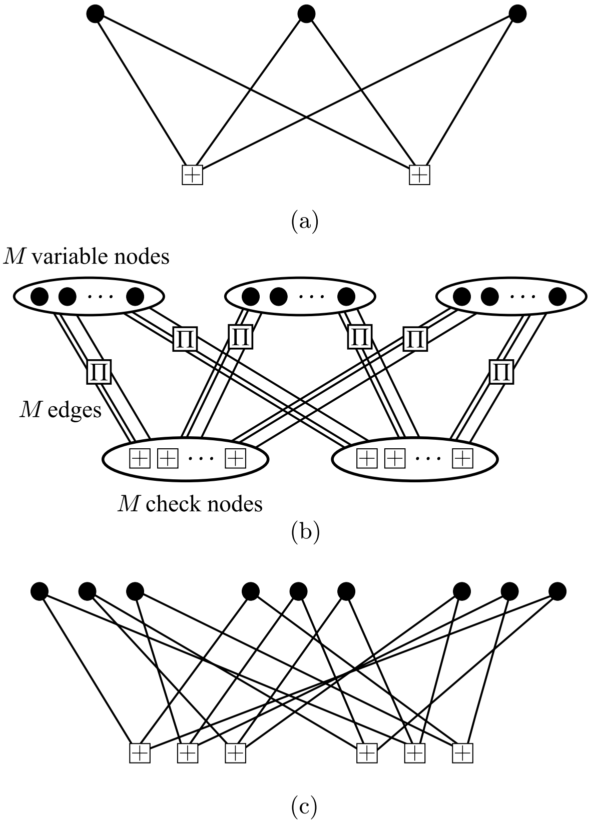


Fig. 1. (a) Protograph of a $(2, 3)$-regular LDPC-BC ensemble, (b) Tanner graph of a $(2, 3)$-regular LDPC-BC lifted from the protograph with lifting factor M, where Π denotes a random interleaver (permutation) of size M, and (c) example $(2, 3)$-regular Tanner graph lifted from the protograph with $M = 3$.

Fig. 1(a) shows an example protograph with $n_v = 3$ variable nodes and $n_c = 2$ check nodes. The Tanner graph representing a *protograph-based* LDPC-BC with block length $n = Mn_v$ is obtained by taking an M*-fold graph cover* (see [12]) or "M*-lifting*" of the protograph. Graph lifting can be informally described as follows: each edge in the protograph becomes a bundle of M edges, connecting M copies of a variable node to M copies of a check node. The connections within each bundle are then permuted between the variable and check node pairings. The resulting covering graph is M times larger than the protograph and has the same rate, degree distribution, and computation graphs as the protograph.[3]

Example 1: Fig. 1(b) shows a general M-lifting of the $(2, 3)$-regular protograph given in Fig. 1(a), and Fig. 1(c) shows a particular M-lifting with $M = 3$. □

The protograph can be represented by its $n_c \times n_v$ *base* biadjacency matrix $\mathbf{B}$, where $B_{i,j}$ is taken to be the number of edges connecting variable node v_j to check node c_i. In general, a protograph can have multiple edges connecting a variable node to a check node, which corresponds to entries in $\mathbf{B}$ greater than 1. The $Mn_c \times Mn_v$ parity-check matrix $\mathbf{H}$ of

[2] The design rate is determined by the size of the protograph and is a lower bound on the code rate of each member of the protograph-based code ensemble.

[3] The computation graphs are preserved since the computation graph of each of the M copies of a variable node in the lifted graph is identical to the computation graph of the original variable node in the protograph (see [46]).

a protograph-based LDPC-BC with block length $n = Mn_v$ and rate $R \geq 1 - Mn_c/Mn_v = 1 - n_c/n_v$ is created (*M-lifted*) by replacing each non-zero entry in $\mathbf{B}$ by a sum of $B_{i,j}$ permutation matrices of size $M \times M$ and each zero entry by the $M \times M$ all-zero matrix.[4]

Example 1 (cont.): The base matrix of the protograph shown in Fig. 1(a) is

$$\mathbf{B} = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}, \tag{1}$$

and the parity-check matrix corresponding to an M-lifting of the base matrix in (1) is

$$\mathbf{H} = \begin{bmatrix} \Pi_{1,1} & \Pi_{1,2} & \Pi_{1,3} \\ \Pi_{2,1} & \Pi_{2,2} & \Pi_{2,3} \end{bmatrix}, \tag{2}$$

where $\Pi_{i,j}$ is an $M \times M$ permutation matrix. □

Since an LDPC-BC is defined as the null space of a sparse parity-check matrix $\mathbf{H}$, we define the *ensemble* of protograph-based LDPC-BCs with block length $n = Mn_v$ and design rate $R = 1 - n_c/n_v$ as the set of all parity-check matrices $\mathbf{H}$ that can be lifted from a given base matrix $\mathbf{B}$, or equivalently as the collection of all M-fold graph covers of the protograph. It is an important feature of this construction that each lifted code inherits the design rate, degree distribution, and computation graphs of the protograph. As a consequence, ensemble DE and weight enumerator analysis can be performed *within* the protograph [48]. Using these tools, properly designed protograph-based LDPC-BC ensembles have been shown in the literature to have many desirable features, such as good iterative decoding thresholds and linear minimum distance growth (see [48]).

A particularly interesting example of such a code design that incorporates both of these desirable features is the accumulate-repeat-jagged-accumulate (ARJA) and accumulate-repeat-by-4-jagged-accumulate (AR4JA) family of irregular protograph-based LDPC-BC ensembles [48]. These practically interesting codes were proposed as a CCSDS standard for near-earth and deep space communication [50] and serve in this paper as an example and template for the successful application of spatial coupling to irregular graphs. The protographs of these ensembles are depicted in Fig. 2. (Note that setting $e = 0$ in the AR4JA protograph results in the ARJA protograph.) The white circles in these protographs represent *punctured* variable nodes, *i.e.*, no code bits are transmitted in these positions. In a Tanner graph M-lifted from the protograph, the M copies of a punctured variable node are also punctured. The design rate of a protograph-based LDPC-BC ensemble with n_t transmitted variable nodes in the protograph is

$$R = \frac{n_v - n_c}{n_t}. \tag{3}$$

Note that, in the case $n_t = n_v$, there is no puncturing and consequently the design rate is $R = 1 - n_c/n_v$.

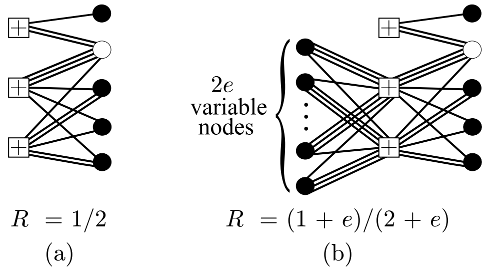


Fig. 2. Protographs representing good irregular LDPC-BC ensembles: (a) the ARJA protograph with design rate $R = 1/2$, and (b) the family of AR4JA protographs with extension parameter e and design rates $R = (1+e)/(2+e)$. White circles represent punctured variable nodes.

C. Convolutional Protographs and Spatial Coupling

In this section, we introduce the notion of a *convolutional protograph*, which represents an ensemble of SC-LDPC-CCs. A convolutional protograph is obtained by connecting, or *spatially coupling*, a sequence of disjoint block protographs together in a chain. Spatial coupling introduces *memory* into the code design, *i.e.*, transitioning from a block code to a convolutional code, and is achieved by applying an *edge spreading* operation to the sequence of disjoint block protographs.

Definition 1 (Edge Spreading Rule for Spatial Coupling): Consider replicating a block protograph with b_v variable nodes and b_c check nodes as an infinite sequence of disjoint graphs. We associate each graph in the sequence with a time index t. Suppose variable node v_j is connected to check node c_i by $B_{i,j}$ edges in each protograph, where $i \in \{0, 1, \ldots, b_c - 1\}$ and $j \in \{0, 1, \ldots, b_v - 1\}$. We now *spread* (connect) the $B_{i,j}$ edges emanating from node v_j at time t arbitrarily over the $w + 1$ check nodes of type c_i at times $t, t + 1, \ldots, t + w$, where $w > 0$ is the *coupling width* of the graph, or *memory* of the code.[5] This operation is repeated (independently) for each of the b_v variable nodes at time t. Applying this edge spreading identically to the variable nodes at all time instants results in a *convolutional protograph.*

Definition 2: An ensemble of protograph-based spatially coupled LDPC-CCs (SC-LDPC-CCs) with coupling width w, design rate $R = 1 - b_c/b_v$, and *constraint length* $\nu = (w + 1)Mb_v$ is the collection of all M-fold graph covers of a convolutional protograph.

In this paper we are primarily interested in asymptotic results in the code block or constraint length, *i.e.*, in the regime where the lifting factor M tends to infinity. A block/convolutional protograph represents a finite block/constraint length LDPC-BC/LDPC-CC ensemble for each lifting factor M; however, in the sequel, unless stated otherwise, we consider an infinite lifting factor. Thus, if we refer to a code ensemble represented by a protograph in the singular, we implicitly assume infinite M.

Note that the convolutional protograph constructed using the Edge Spreading Rule has the same design rate, degree

[4] The lifted parity-check matrix $\mathbf{H}$ may have linearly dependent rows; this simply means that the lifted code has a slightly higher rate than the design rate $R = 1 - n_c/n_v$.

[5] The coupling width w is referred to in convolutional coding parlance as the *syndrome former memory* (see [7]), or, in the recent series of papers by Kudekar *et al.*, the *smoothing parameter* [21], [22].

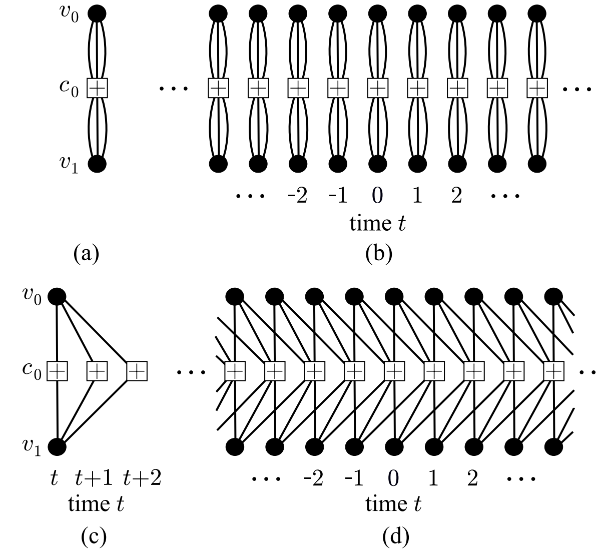


Fig. 3. (a) Protograph representing a (3, 6)-regular LDPC-BC ensemble, (b) sequence of (3, 6)-regular LDPC-BC protographs, (c) illustration of edge spreading for one segment of the graph at time t with coupling width $w = 2$, and (d) protograph representing a (spatially coupled) (3, 6)-regular LDPC-CC ensemble with $w = 2$.

distribution, and computation graphs as the original block protograph. This ensures that the computation graphs of the LDPC-CC ensembles defined by the convolutional protograph are the same as those of the LDPC-BC ensembles defined by the original block protograph. There are many ways to apply the Edge Spreading Rule for a given w to a sequence of disjoint block protographs that give an extra degree of freedom in the protograph-based construction. It will be shown later that different edge spreadings affect both the iterative BP decoding performance and distance properties of the resulting code ensemble. Generalizations of the edge spreading operation and a discussion of the degrees of freedom in the design are presented in Section II-E.

Example 2: Fig. 3 illustrates the edge spreading operation applied to a $(3, 6)$-regular (block) protograph with design rate $R = 1/2$. First, the protograph is replicated as an infinite sequence of disjoint graphs, shown in Fig. 3(b). (This can be considered as block code transmission over time.) An edge spreading with coupling width $w = 2$ applied to the variable nodes at time t is shown in Fig. 3(c). The three edges emanating from each variable node v_0 and v_1 are spread such that exactly one edge connects the variable node to check node c_0 at times t, $t{+}1$, and $t{+}2$. Applying this edge spreading to variable nodes at all time instants results in the $(3, 6)$-regular convolutional protograph with design rate $R = 1/2$ shown in Fig. 3(d). □

The (convolutional) base matrix corresponding to the convolutional protograph is

$$\mathbf{B}_{[-\infty,\infty]} = \begin{bmatrix} \ddots & \ddots & & & \ddots & & \\ & \mathbf{B}_w & \mathbf{B}_{w-1} & \cdots & \mathbf{B}_0 & & \\ & & \mathbf{B}_w & & \mathbf{B}_{w-1} & \cdots & \mathbf{B}_0 \\ & & & \ddots & & \ddots & \ddots \end{bmatrix}, \tag{4}$$

where the $b_c \times b_v$ *component base matrices* $\mathbf{B}_i$, $i = 0, 1, \ldots, w$, represent the edge connections from the b_v variable nodes at time t to the b_c check nodes at time $t + i$. Starting from the base matrix $\mathbf{B} = [B_{i,j}]_{0 \le i \le n_c - 1, 0 \le j \le n_v - 1}$ of an LDPC-BC ensemble, the Edge Spreading Rule divides the edges associated with each variable node in $\mathbf{B}$ among $w + 1$ component base matrices $\mathbf{B}_i$, $i = 0, 1, \ldots, w$, such that the condition

$$\sum_{i=0}^{w} \mathbf{B}_i = \mathbf{B} \tag{5}$$

is satisfied, where each $\mathbf{B}_i$ contains non-negative integer entries.

Example 2 (cont.): The $(3, 6)$-regular protograph shown in Fig. 3(a) has base matrix $\mathbf{B} = [\,3 \quad 3\,]$. The edge spreading depicted in Fig. 3(c) with $w = 2$ corresponds to component base matrices

$$\mathbf{B}_0 = [\,1 \quad 1\,] = \mathbf{B}_1 = \mathbf{B}_2. \tag{6}$$

(Note that this is a valid edge spreading since the component base matrices conform to condition (5).) Then the $(3, 6)$-regular convolutional base matrix corresponding to the convolutional protograph of Fig. 3(d) is obtained in the form of (4) as

$$\mathbf{B}_{[-\infty,\infty]} = \begin{bmatrix} \ddots & \ddots & \ddots & & & \\ \boxed{1\ 1} & \boxed{1\ 1} & \boxed{1\ 1} & & & \\ & \boxed{1\ 1} & \boxed{1\ 1} & \boxed{1\ 1} & & \\ & & \boxed{1\ 1} & \boxed{1\ 1} & \boxed{1\ 1} & \\ & & & \ddots & \ddots & \ddots \end{bmatrix}.$$

□

A similar edge spreading to that used in Example 2 can be applied to construct (J, K)-regular convolutional protographs from (J, K)-regular block protographs where the greatest common divisor of J and K is greater than one.

Definition 3 (The $\mathcal{C}(J, K)$ SC-LDPC-CC Ensemble): Let $a = \gcd(J, K)$ denote the greatest common divisor of J and K. Then there exist positive integers J' and K' such that $J = aJ'$ and $K = aK'$ with $\gcd(J', K') = 1$. It follows that the base matrix of a (J, K)-regular protograph-based SC-LDPC-CC ensemble with coupling width $w = a - 1$ can be defined as in (4), where the submatrices $\mathbf{B}_i$, $i = 0, \ldots, w$, are identical $J' \times K'$ matrices with all entries equal to one. We denote the SC-LDPC-CC ensembles constructed using this edge spreading as $\mathcal{C}(J, K)$.

Note that, if $a = 1$, the coupling width is equal to zero and the convolutional protograph is not fully connected. In this case, we can simply choose a different edge spreading of a (J, K)-regular block protograph following the Edge Spreading Rule.

Example 3: Consider the $(3, 4)$-regular protograph defined by the all-ones base matrix $\mathbf{B}$ of size 3×4. We can spread the edges of $\mathbf{B}$ as

$$\mathbf{B}_0 = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix} \text{ and } \mathbf{B}_1 = \begin{bmatrix} 0 & 0 & 1 & 1 \\ 1 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 \end{bmatrix}.$$

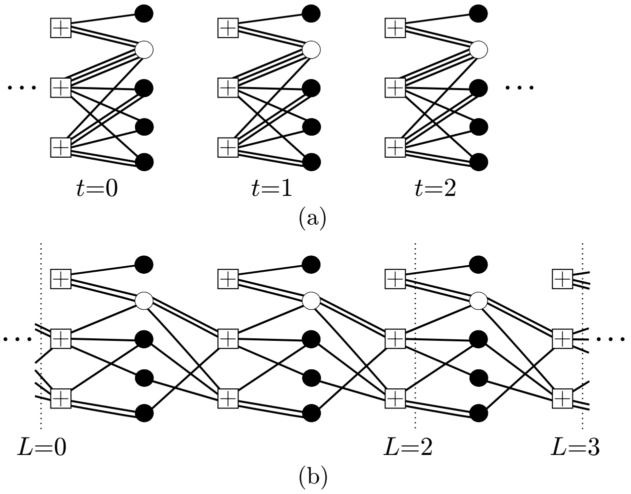


Fig. 4. (a) Sequence of ARJA block protographs with design rate $R = 1/2$, and (b) spatially coupled ARJA convolutional protograph with $w = 1$ and design rate $R = 1/2$. Also shown are the termination markings for the related spatially coupled ARJA block protograph.

These component base matrices satisfy condition (5) and can be used to construct a $(3, 4)$-regular convolutional base matrix $\mathbf{B}_{[-\infty,\infty]}$ with coupling width $w = 1$ (see (4)). □

Next, we demonstrate the application of the Edge Spreading Rule to irregular protographs.

Example 4: Fig. 4 shows an example of the Edge Spreading Rule applied to the irregular ARJA protograph with base matrix

$$\mathbf{B} = \begin{bmatrix} 1 & 2 & 0 & 0 & 0 \\ 0 & 3 & 1 & 1 & 1 \\ 0 & 1 & 2 & 1 & 2 \end{bmatrix}. \tag{7}$$

A sequence of disjoint ARJA protographs with design rate $R = 1/2$ is shown in Fig. 4(a). An irregular (spatially coupled) ARJA convolutional protograph with design rate $R = 1/2$ and coupling width $w = 1$ is shown in Fig. 4(b). The component base matrices corresponding to this edge spreading are

$$\mathbf{B}_0 = \begin{bmatrix} 1 & 2 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 2 \end{bmatrix}, \quad \mathbf{B}_1 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 & 1 \\ 0 & 1 & 1 & 1 & 0 \end{bmatrix},$$

where $\mathbf{B}_0 + \mathbf{B}_1 = \mathbf{B}$. Note that there is one punctured variable node at each time instant of the convolutional protograph. In the sequel, we refer to the SC-LDPC-CC ensemble represented by this ARJA convolutional protograph by $\mathcal{C}_{\text{ARJA}}$.

Similarly, we can construct a series of AR4JA convolutional protographs with coupling width $w = 1$ and design rate $R = (1 + e)/(2 + e)$ using the edge spreading shown in Fig. 5. The resulting SC-LDPC-CC ensembles are denoted $\mathcal{C}_{\text{AR4JA}}(e)$. □

Remark 4: The convolutional protograph constructed using the Edge Spreading Rule with coupling width w can be viewed as an infinite graph lifting of the block protograph. Consequently, a protograph-based SC-LDPC-CC can be viewed as a double graph cover of a block code protograph. As the local connectivity is maintained by graph lifting, the computation graph is identical and the BP decoder cannot distinguish if it is operating on the original protograph or a covering graph of the protograph. As a result, the BP decoding threshold of the SC-LDPC-CC is *equal* to the BP decoding threshold of the uncoupled LDPC-BC ensemble. For further discussion of SC-LDPC-CCs as graph covers of LDPC-BCs, see [10], [12].

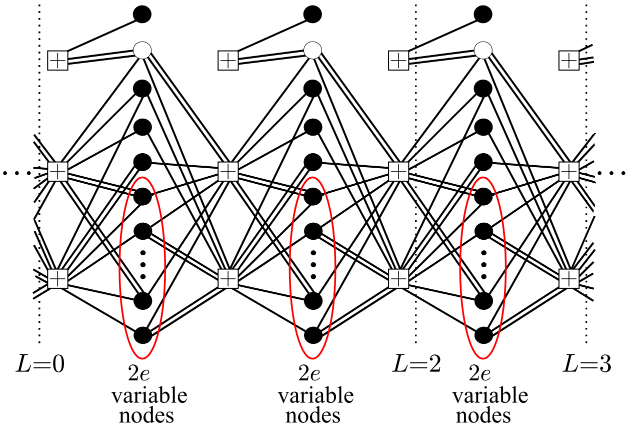


Fig. 5. Spatially coupled AR4JA convolutional protographs with coupling width $w = 1$ and design rate $R = (1 + e)/(2 + e)$. Also shown are the termination markings for the related spatially coupled AR4JA block protograph.

D. Spatially Coupled LDPC Block Codes

Even though the convolutional protograph and lifted SC-LDPC-CC Tanner graphs extend infinitely forward and backward in time, in practice there is always some finite starting and ending time, *i.e.*, the protograph is *terminated.* As a consequence, 'convolutional-like' block codes of flexible frame length can be obtained by termination, and we will see later that the iterative BP threshold of SC-LDPC-CCs is significantly improved by termination.

1) Terminated SC-LDPC-CC Ensembles:

Definition 5: A terminated convolutional protograph with coupling width w and *coupling length* $L > 0$ can be obtained as the subgraph of the convolutional protograph induced by the variable nodes over time instants $t = 0, 1, \ldots, L - 1$. An ensemble of protograph-based *spatially coupled LDPC-BCs (SC-LDPC-BCs)* with coupling width w, coupling length L, and block length $n = Mn_v = MLb_v$ is obtained as the collection of all M-fold graph covers of the terminated convolutional protograph, where $n_v = Lb_v$ is the total number of variable nodes in the terminated convolutional protograph.

Terminating the convolutional protograph is equivalent to applying the Edge Spreading Rule to spatially couple L disjoint copies of a block protograph, where connections are allowed at the right hand boundary to wb_c additional check nodes in sections $t = L, L+1, \ldots, L+w-1$. Consequently, there are b_v variable nodes and b_c check nodes at each time instant $t = 0, 1, \ldots, L - 1$ and b_c additional check nodes at each time instant $t = L, L + 1, \ldots, L + w - 1$. We now use n_v and n_c to denote the total number of variable nodes and check nodes, respectively, in the terminated convolutional protograph. The $n_c \times n_v = (L + w)b_c \times Lb_v$ base matrix $\mathbf{B}_{[0,L-1]}$ corresponding to the terminated convolutional protograph is

$$\mathbf{B}_{[0,L-1]} = \begin{bmatrix} \mathbf{B}_0 & & & \\ \mathbf{B}_1 & \mathbf{B}_0 & & \\ \vdots & \mathbf{B}_1 & \ddots & \\ \mathbf{B}_w & \vdots & \ddots & \mathbf{B}_0 \\ & \mathbf{B}_w & & \mathbf{B}_1 \\ & & \ddots & \vdots \\ & & & \mathbf{B}_w \end{bmatrix}_{(L+w)b_c \times Lb_v}, \tag{8}$$

which can be obtained by truncating the convolutional base matrix $\mathbf{B}_{[-\infty,\infty]}$ from (4).

Terminating a convolutional protograph has two effects on the resulting subgraph:

- A *structured irregularity* is introduced to the graph: the variable nodes at each time index have the same number and type of edge connections as the original block protograph; however, the first wb_c and last wb_c check nodes have, in general, a reduced number of edge connections;
- For finite L, a *rate loss* is incurred due to the check nodes at the right hand boundary of the subgraph (connections to check nodes at time instants $t = L, L + 1, \ldots, L + w - 1$).

The design rate of the terminated convolutional protograph (without puncturing) is

$$R_L = 1 - \frac{n_c}{n_v} = 1 - \frac{(L + w)b_c}{Lb_v} = 1 - \left(\frac{L + w}{L}\right)(1 - R), \tag{9}$$

where $R = 1 - b_c/b_v$ is the design rate of the unterminated convolutional protograph (and the uncoupled block protograph). We assume that a sufficiently large L is chosen such that the SC-LDPC-BC ensemble has a strictly positive design rate, *i.e.*, $L > wb_c/(b_v - b_c)$. Assuming $w > 0$ and finite L, we see from (9) that the terminated convolutional protograph has reduced design rate $R_L < R$. Note that the rate loss and structured irregularity in the check node degree distribution introduced by termination become vanishingly small as the coupling length L increases: the rate R_L increases monotonically and approaches the design rate R of the unterminated convolutional protograph ($\lim_{L\to\infty} R_L = R$), and the check node degree distribution approaches that of the unterminated convolutional protograph.

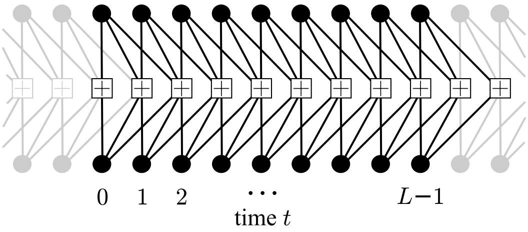


Fig. 6. Protograph of a SC-LDPC-BC ensemble (highlighted in black) with coupling length L and coupling width $w = 2$ obtained by terminating a $(3, 6)$-regular convolutional protograph.

Example 2 (cont.): Fig. 6 shows (highlighted in black) the terminated $(3, 6)$-regular convolutional protograph induced by the variable nodes over time instants $t = 0, 1, \ldots, L - 1$. The corresponding terminated convolutional base matrix, obtained using (8), is

$$\mathbf{B}_{[0,L-1]} = \begin{bmatrix} \boxed{1\ 1} & & & \\ \boxed{1\ 1} & \boxed{1\ 1} & & \\ \boxed{1\ 1} & \boxed{1\ 1} & \ddots & \\ & \boxed{1\ 1} & \ddots & \\ & & \ddots & \boxed{1\ 1} \\ & & & \boxed{1\ 1} \\ & & & \boxed{1\ 1} \end{bmatrix}_{(L+2)\times 2L}.$$

Note the structured irregularity in the resulting chain of coupled protographs: each variable node is connected to 3 check nodes, while the check nodes in the middle are connected to 6 variable nodes. The $wb_c = 2$ check nodes located at the beginning and at the end of the chain, however, are only connected to either 2 or 4 variable nodes. The ensemble design rate for $L > 2$ is obtained using (9) as

$$R_L = 1 - \frac{L + 2}{2L} = \frac{L - 2}{2L}.$$

We will denote the SC-LDPC-BC ensemble obtained using this edge spreading and coupling length L as $\mathcal{C}(3, 6, L)$. With the exception of the slight structured irregularities at the ends of the graph, the $\mathcal{C}(3, 6, L)$ ensembles retain essentially all of the beneficial structural properties of $(3, 6)$-regular ensembles with, as will be demonstrated later, dramatically improved iterative BP thresholds. □

In a similar fashion, we obtain $\mathcal{C}(J, K, L)$ SC-LDPC-BC ensembles by terminating the associated $\mathcal{C}(J, K)$ SC-LDPC-CC ensemble with coupling length L.

Definition 6 (The $\mathcal{C}(J, K, L)$ SC-LDPC-BC Ensemble): Let $a = \gcd(J, K)$ denote the greatest common divisor of J and K. Then there exist positive integers J' and K' such that $J = aJ'$ and $K = aK'$ with $\gcd(J', K') = 1$. It follows that the base matrix of a protograph-based SC-LDPC-BC ensemble with coupling width $w = a - 1$ can be defined as in (8), where the submatrices $\mathbf{B}_i$, $i = 0, \ldots, w$, are identical $J' \times K'$ matrices with all entries equal to one. We denote the SC-LDPC-BC ensembles constructed using this edge spreading as $\mathcal{C}(J, K, L)$.

It follows that $\mathcal{C}_{\mathrm{ARJA}}(L)$ and $\mathcal{C}_{\mathrm{AR4JA}}(e, L)$ SC-LDPC-BC ensembles can be defined in a similar fashion.

Example 4 (cont.): The irregular ARJA convolutional protograph can be terminated as shown in Fig. 4(b). Note the additional (structured) irregularity introduced when the convolutional protograph is terminated: the variable nodes

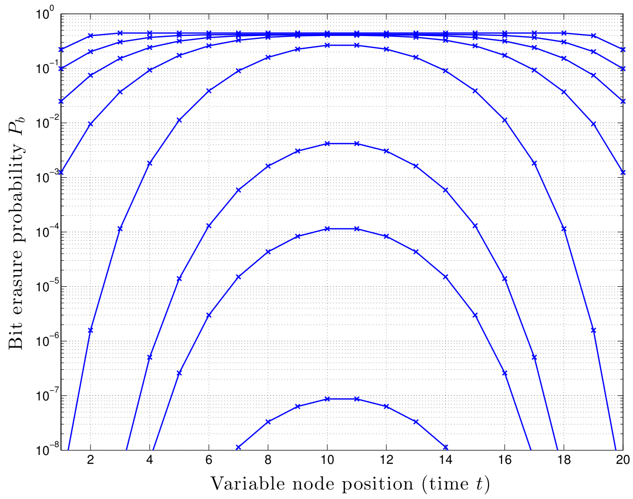


Fig. 7. Evolution of the average bit erasure probability P_b of the variable nodes at time t for the $\mathcal{C}(3, 6, 20)$ SC-LDPC-BC ensemble transmitted over a BEC with erasure probability $\varepsilon = 0.48$ for iterations $i = 1, 5, 20, 50, 90, 98, 99, 100$ (from top to bottom).

at each time instant and the check nodes in the center of the chain have the same number and type of connections as the block/convolutional ARJA protograph; however, the check nodes at the ends of the chain have a reduced number of connections. Note also that there are exactly L punctured nodes in the terminated protograph with base matrix $\mathbf{B}_{[0,L-1]}$. As a result of the all-zero row in component matrix $\mathbf{B}_1$ (the disconnected check node in Fig. 4(b)), the terminated protograph associated with $\mathbf{B}_{[0,L-1]}$ has $n_c = (L+w)b_c - 1 = 3L + 2$ check nodes and $n_v = Lb_v = 5L$ variable nodes. After puncturing, the number of transmitted variable nodes is $n_t = 5L - L = 4L$ (see Fig. 4(b)) and the design rate of the SC-LDPC-BC ensemble with coupling length $L \geq 2$ is

$$R_L = \frac{n_v - n_c}{n_t} = \frac{5L - (3L+2)}{4L} = \frac{L-1}{2L}.$$

In the sequel we will denote the SC-LDPC-BC ensemble obtained using this edge spreading and coupling length L as $\mathcal{C}_{\text{ARJA}}(L)$.

In an identical way, the irregular AR4JA convolutional protograph can be terminated as shown in Fig. 5. The design rate of the terminated convolutional protograph with extension parameter e and coupling length $L \geq 2$ is given by

$$R_L = \frac{(5+2e)L - (3L+2)}{(4+2e)L} = \frac{(1+e)L - 1}{(2+e)L}.$$

In the sequel we will denote the SC-LDPC-BC ensemble obtained using this edge spreading and coupling length L as $\mathcal{C}_{\text{AR4JA}}(e, L)$. □

In the context of iterative BP decoding, the smaller degree check nodes at the ends of the graph pass more reliable messages to their neighboring variable nodes, and this effect propagates throughout the graph as iterations increase. This effect is demonstrated in Fig. 7, where we plot the evolution of the average bit erasure probability P_b, obtained using DE on a BEC, of the variable nodes at times $t = 1, 2, \ldots, 20$ for the $\mathcal{C}(3, 6, 20)$ SC-LDPC-BC ensemble with an increasing number of iterations of the BP decoder with a 'flooding' update schedule (all check nodes in the graph are updated followed by all variable nodes in each iteration). We observe that P_b for variable nodes close to the ends of the spatially coupled chain, which are connected to the lower degree check nodes, quickly decreases with iterations, and that this 'wave' moves through the chain from either end towards the variable nodes in the center. In Section III, we will see that this phenomenon results in excellent iterative decoding thresholds for SC-LDPC-BC ensembles. Note that, after termination, the SC-LDPC-CC ensemble can be viewed as an LDPC-BC ensemble with block length $n = MLb_v$. However, compared to typical LDPC-BC designs that have no restrictions on the location of the ones in the parity-check matrix and hence allow connections across the entire graph, the SC-LDPC-BC ensemble has a highly *localized* graph structure, since the non-zero portion of the parity-check matrix is restricted to a diagonal band of width ν. In addition to the good asymptotic ensemble properties such as excellent BP thresholds and linear minimum distance growth rates that will be demonstrated in Section III, this localized graph structure also gives rise to efficient decoder implementations such as the high-throughput *pipeline decoder* [7], [11] and low-latency *sliding window* decoding strategies [15], [51], [52]. Such strategies can significantly reduce the complexity, memory, and latency requirements; see Section II-E.6 for further details.

2) Tail-Biting LDPC-CC Ensembles: The convolutional protograph can also be terminated using *tail-biting* [53], [54].

Definition 7: A tail-biting convolutional protograph is obtained from the terminated convolutional protograph with coupling length $L > w$ by combining the check nodes at times $t = L, L+1, \ldots, L+w-1$ with the corresponding check nodes of the same type at times $t = 0, 1, \ldots, w-1$, respectively. An ensemble of protograph-based *tail-biting spatially coupled LDPC-BCs (TB-SC-LDPC-BCs)* with block length $n = MLb_v$ is then obtained as the collection of all M-fold graph covers of the tail-biting convolutional protograph.

The $Lb_c \times Lb_v$ base matrix $\mathbf{B}^{tb}_{[0,L-1]}$ corresponding to the tail-biting convolutional protograph is

$$\mathbf{B}^{tb}_{[0,L-1]} = \left[\begin{array}{ccccccc}
\mathbf{B}_0 & & & & \mathbf{B}_w & \cdots & \mathbf{B}_1 \\
\vdots & \mathbf{B}_0 & & & & \ddots & \vdots \\
\mathbf{B}_{w-1} & \vdots & & & & & \mathbf{B}_w \\
\mathbf{B}_w & \mathbf{B}_{w-1} & & & & & \\
 & \mathbf{B}_w & \ddots & & & & \\
 & & & \mathbf{B}_0 & & & \\
 & & \ddots & \vdots & \mathbf{B}_0 & & \\
 & & \ddots & \mathbf{B}_{w-1} & \vdots & \ddots & \\
 & & & \mathbf{B}_w & \mathbf{B}_{w-1} & \cdots & \mathbf{B}_0
\end{array}\right], \quad (10)$$

which can be obtained from the terminated base matrix $\mathbf{B}_{[0,L-1]}$ in (8) by adding the last wb_c rows to the first wb_c rows.

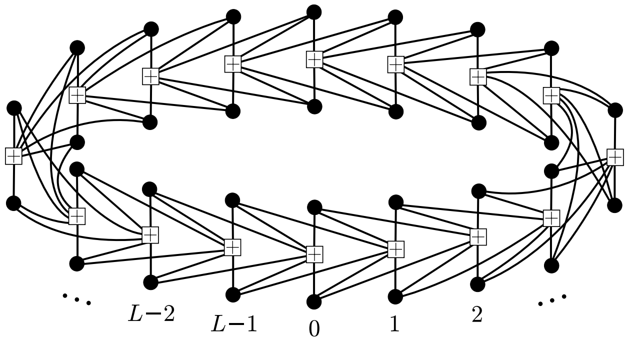


Fig. 8. Protograph of a $(3, 6)$-regular TB-SC-LDPC-BC ensemble with coupling length L and coupling width $w = 2$.

Note that the tail-biting protograph has the same design rate

$$R_L^{tb} = 1 - Lb_c/Lb_v = 1 - b_c/b_v = R, \tag{11}$$

and degree distribution as the convolutional protograph, *i.e.*, there is no structured irregularity introduced to the graph or rate loss after termination. Consequently, tail-biting is a useful way to terminate a convolutional protograph to a block protograph of desired length such that the properties of the convolutional protograph are retained. We denote the TB-SC-LDPC-BC ensemble obtained from the SC-LDPC-BC ensemble $\mathcal{C}(J, K, L)$ with coupling length L as $\mathcal{C}_{\text{tb}}(J, K, L)$.

Example 2 (cont.): Fig. 8 shows the tail-biting (3, 6)-regular convolutional protograph with coupling length L and coupling width $w = 2$. The corresponding tail-biting base matrix, obtained using (10), is

$$\mathbf{B}^{tb}_{[0,L-1]} = \begin{bmatrix} \boxed{1\ 1} & & & & \boxed{1\ 1} & \boxed{1\ 1} \\ \boxed{1\ 1} & \boxed{1\ 1} & & & & \boxed{1\ 1} \\ \boxed{1\ 1} & \boxed{1\ 1} & \ddots & & & \\ & \boxed{1\ 1} & \ddots & & & \\ & & \ddots & \boxed{1\ 1} & & \\ & & & \boxed{1\ 1} & \boxed{1\ 1} & \\ & & & \boxed{1\ 1} & \boxed{1\ 1} & \boxed{1\ 1} \end{bmatrix}_{L\times 2L}.$$

Note that each variable node has degree 3 and each check node has degree 6 in the tail-biting protograph, *i.e.*, the graph is $(3, 6)$-regular, the degree distribution is unchanged, and the ensemble design rate is $R_L^{tb} = 1 - L/2L = 1/2$. □

Protograph-based TB-SC-LDPC-BCs have been used to obtain lower bounds on important parameters of protograph-based SC-LDPC-CC ensembles, such as the free distance [18], [19] and minimum trapping set size [19].

E. Discussion

1) Edge Spreading Variations: Given a coupling width w, one may construct a convolutional protograph based on a *time-varying edge spreading*, where a different edge spreading is applied at each time instant, such that

$$\sum_{i=0}^{w} \mathbf{B}_i(t) = \mathbf{B}, \quad \forall t. \tag{12}$$

(Note that the degree distribution and computation graphs are not necessarily preserved under this generalization of the Edge Spreading Rule.) Moreover, the construction of a convolutional protograph can be further generalized by coupling a sequence of time-varying protographs, *i.e.*, the base matrix $\mathbf{B}(t)$ at each time instant depends on t.

2) Non-Protograph-Based SC-LDPC-CC Construction: Edge spreading can be applied directly to the Tanner graph or parity-check matrix $\mathbf{H}$ of an LDPC-BC to construct an SC-LDPC-CC, without first constructing a block protograph. The two major approaches that have been detailed in the literature can be categorized in this way:

- Tanner first developed the connection between *quasi-cyclic (QC)* block codes and time-invariant convolutional codes [55]. This approach was extended to construct time-invariant SC-LDPC-CCs in [10], where the construction can be viewed as a particular infinite graph cover of the Tanner graph of a QC-LDPC-BC [12];
- SC-LDPC-CCs were first introduced in the open literature by Jimenez-Felström and Zigangirov in 1999 [7]. Here, time-varying LDPC-CCs were constructed using a cut-and-paste technique termed *unwrapping*, which is equivalent to applying the Edge Spreading Rule to a sequence of disjoint LDPC-BC Tanner graphs. As a result of the unwrapping/edge spreading procedure, the computation graph of the underlying LDPC-BC is preserved.

For both construction methods, the SC-LDPC-CCs were shown to have improved BER performance compared to their underlying LDPC-BC counterparts [7], [10], [12].

3) Kudekar's Randomized SC-LDPC-BC Ensemble: A construction of SC-LDPC-BC ensembles, closely related to the $\mathcal{C}(J, K, L)$ ensemble studied in this paper, was presented recently by Kudekar *et al.* [21]. Here, M degree J variable nodes and $\frac{J}{K}M$ degree K check nodes are placed at $2L + 1$ index positions $[-L, L]$, $L \in \mathbb{N}$. In a similar way to the construction presented here, the graphs are coupled together, where the J connections from the M variable nodes at position t are allowed only to check nodes at positions $[t, t + w - 1]$; however, here the connections are *randomized*, such that different edge spreadings are applied to specific fractions of the M variable nodes at each time instant (see [21] for a precise definition of the ensemble). This randomized ensemble construction results in a similar structured irregularity and rate loss due to the boundary (termination) effects as the protograph-based SC-LDPC-BC ensembles presented here. It should be noted, however, that the randomized code ensemble does not contain any particular protograph-based code ensemble since, given a coupling width w, the randomized ensemble has a non-zero fraction of variable nodes at each position in the chain of every possible edge spreading type, whereas a protograph-based ensemble contains a specific (small) number of edge spreading types at each position.

In general, the randomized ensemble does not enjoy as favorable a tradeoff between rate, threshold, and block length as the protograph-based ensemble [21], and it lacks the inherent implementation advantages of a structured ensemble; however, it is a useful ensemble for analytical purposes. In particular, it was shown analytically in [21] that the BP threshold for the randomized SC-LDPC-BC ensemble improves all the way to the optimal maximum *a posteriori* (MAP) threshold of the underlying (J, K)-regular LDPC-BC ensemble (a fact previously demonstrated numerically for a permutation matrix-based SC-LDPC-CC ensemble in [15]), a phenomenon termed *threshold saturation*. In other words, the randomized ensemble achieves globally optimal decoding performance with low-complexity, locally optimal, iterative BP decoding. In this paper, following the approach applied in [15], we show numerically that the threshold saturation effect also occurs for the considered protograph-based SC-LDPC-BC ensembles. In addition, we show that such ensembles have minimum distance growing linearly with block length, promising excellent performance in both the waterfall *and* error-floor regions of the BER curve.

4) Quasi-Cyclic Protograph-Based Codes: In general, highly structured codes and code ensembles are attractive from an implementation standpoint. In particular, members of a protograph-based LDPC code ensemble that are QC are of great interest to code designers, since they can be encoded with low complexity using simple feedback shift-registers [56], [57] and their structure leads to efficiencies in decoder design [58], [59]. Moreover, QC-LDPC codes can be shown to perform well compared to randomly constructed LDPC codes for moderate block lengths [10], [60]–[62]. The construction of QC-LDPC codes can be seen as a special case of the protograph-based construction in which the M-fold graph cover is obtained by restricting the edge permutations to be cyclic, and it can be described by an $Mn_c \times Mn_v$ parity-check matrix formed as an $n_c \times n_v$ array of $M \times M$ circulant matrices. However, unlike typical members of a protograph-based LDPC code ensemble, asymptotic ensemble average results such as iterative decoding thresholds and minimum distance growth rates cannot be used to describe the behavior of the QC sub-ensemble, since the probability of picking such a code vanishes in the limit of large M. For example, if the protograph base matrix consists of only ones and zeros, then the minimum Hamming distance is bounded above by $(n_c+1)!$, where n_c is the number of check nodes in the protograph, regardless of the lifting factor M [63], [64].

5) Code Design Flexibility: A nice feature of SC-LDPC-CCs is that, by varying the termination (or coupling) length L, we obtain a flexible family of SC-LDPC-BCs with varying rates and frame lengths that display little variation in performance [11], *i.e.*, the beneficial properties of spatial coupling are preserved over a range of termination lengths. This is particularly useful in applications or standards that require varying frame lengths, because one would typically have to design a separate LDPC-BC for each required length. Moreover, if the SC-LDPC-CC is *periodic* (defined formally in Section IV), it is possible to obtain a family of periodically time-varying SC-LDPC-BCs that share the same encoding and decoding architecture for arbitrary L.

6) Windowed Decoding and Finite-Length Performance: Practical code design of SC-LDPC-BCs is beyond the scope of this contribution; however we conclude this section with a brief summary of some of the design issues concerning the implementation of windowed decoding of such codes in modern communication and storage systems. The interested reader is directed to [15], [51], [52], and [65] for a more in-depth discussion of the performance, latency, and complexity trade-offs of windowed decoding of SC-LDPC-BCs.

- Since the decoding latency of windowed decoding is determined by the size of the decoding window W [51], [52], [66], a windowed decoding strategy significantly reduces the latency, which is crucial in time-dependent applications such as personal wireless communication, real-time audio and video, and command-and-control military communication.
- In addition to the density of the Tanner graph, which determines the complexity for a single iteration of BP decoding, the overall decoding complexity also depends on the allowed/required number of iterations of message passing. When L is large, applying a standard 'flooding' message passing scheme, where all the check nodes in the graph are updated followed by all the variable nodes in each iteration, will typically not be an efficient way to decode SC-LDPC-BCs since a large number of iterations may be required for the decoding wave to reach the center of the chain. A natural approach to decode SC-LDPC-BCs is to schedule node updates in a greedy way determined by the progress of the decoding wave, *i.e.*, windowed decoding with an appropriate stopping rule. In this way, the required number of iterations required per symbol can be significantly reduced and is similar to LDPC-BCs with the same decoding latency [51], [52], [67].
- The finite-length scaling of SC-LDPC-BCs has been discussed in [44]. As discussed further in Section IV in the context of free distance, the potential strength of SC-LDPC-BC ensembles for large L scales with the constraint length $\nu = M(w + 1)b_v$, which increases with M but is independent of L. Note that the code performance is determined primarily by the coupling width w and the lifting factor M; whereas the decoder performance is determined by W. This implies that, when sliding window decoding strategies are employed, the coupling length L is not a crucial design parameter.

III. Minimum Distance and Threshold Trade-Offs for SC-LDPC-BC Ensembles

In this section, we begin with an asymptotic weight enumerator analysis of protograph-based SC-LDPC-BC ensembles, then proceed by means of a DE analysis to obtain iterative decoding thresholds for both the BEC and AWGNC, demonstrating that the ensembles are both asymptotically good in terms of minimum distance and exhibit the threshold saturation effect with iterative decoding.

A. Weight Enumerators

We begin by summarizing the procedure presented in [48] to obtain the average distance spectrum for a protograph-based ensemble and then apply it to some example SC-LDPC-BC ensembles to test if they are asymptotically good, taking advantage of the fact that the inherent structure of members of a protograph-based LDPC code ensemble facilitates the calculation of average weight enumerators.

Suppose that all n_v variable nodes of the protograph are to be transmitted over the channel and that each of the n_v transmitted variable nodes has an associated weight d_i, where $0 \leq d_i \leq M$ for all i.[6] Let $S_d = \{(d_0, d_1, \ldots, d_{n_v-1})\}$ be the set of all possible weight distributions such that $d_0 + d_1 + \ldots + d_{n_v-1} = d$. The ensemble average weight enumerator for the protograph is then given by

$$A_d = \sum_{(d_0, d_1, \ldots, d_{n_v-1}) \in S_d} A_{\mathbf{d}}, \tag{13}$$

where $A_{\mathbf{d}}$ is the average number of codewords in the ensemble with a particular weight distribution

$$\mathbf{d} = (d_0, d_1, \ldots, d_{n_v-1}).$$

Combinatorial expressions for $A_{\mathbf{d}}$ have been derived in [48] and [68]. Note that if $n_t < n_v$ variable nodes are to be transmitted over the channel, then the weight enumerator A_d is a double summation over all possible partial weight patterns S_p and S_d of the punctured and transmitted variable node weights, respectively, where the codeword weight d is the sum of the partial weights associated with the transmitted nodes (see [48] for details).

The *asymptotic spectral shape function* of a code ensemble can be written as $r(\delta) = \lim_{n\to\infty} \sup r_n(\delta)$, where $r_n(\delta) = \ln(A_d)/n$, $\delta = d/n$, d is the Hamming weight, n is the block length, and A_d is the ensemble average weight distribution. Suppose that the first positive zero crossing of $r(\delta)$ occurs at $\delta = \delta_{\min}$. If $r(\delta)$ is negative in the range $0 < \delta < \delta_{\min}$, then $\delta_{\min}$ is called the *minimum distance growth rate* of the code ensemble. By considering the probability

$$\mathbb{P}(d < n\delta_{\min}) \leq \sum_{d=1}^{n\delta_{\min}-1} A_d,$$

it is clear that, as the block length n becomes sufficiently large, if $\mathbb{P}(d < n\delta_{\min}) \ll 1$, then we can say with high probability that a randomly chosen code from the ensemble has a minimum distance that is at least as large as $n\delta_{\min}$ [48], i.e., the minimum distance increases linearly with block length n. We refer to such an ensemble of codes as *asymptotically good.*

Example 2 (cont.): Examining the asymptotic weight enumerators of the $\mathcal{C}(3, 6, L)$ ensembles for various coupling lengths L, we find that the ensembles are asymptotically good. The calculated minimum distance growth rates are given in Table I. As the coupling length L tends to infinity, we observe that the minimum distance growth rate $\delta_{\min}^{(L)}$ decreases. This is consistent with similar results obtained for TB-SC-LDPC-BC ensembles in [19]. We also observe from Table I that the scaled growth rates $\delta_{\min}^{(L)} L/(w+1)$ converge to a fixed value as L increases. A similar result was first observed in [17] for an ensemble of $(3, 6)$-regular SC-LDPC-CCs constructed from $M \times M$ permutation matrices, where it was shown that the scaled growth rates of the terminated SC-LDPC-BC ensembles converged to a bound on the *free distance* growth rate of the unterminated SC-LDPC-CC ensemble. □

TABLE I
MINIMUM DISTANCE GROWTH RATES FOR THE $\mathcal{C}(3, 6, L)$ SC-LDPC-BC ENSEMBLES

L	Design Rate R_L	Growth rate $\delta_{\min}^{(L)}$	$\delta_{\min}^{(L)} L/(w+1)$
3	$1/6$	0.1419	0.142
4	$1/4$	0.0814	0.109
5	$3/10$	0.0573	0.096
6	$1/3$	0.0449	0.090
7	$5/14$	0.0374	0.087
8	$3/8$	0.0324	0.086
9	$7/18$	0.0287	0.086
10	$2/5$	0.0258	0.086
20	$9/20$	0.0129	0.086
∞	$1/2$	0	

TABLE II
COMPLEXITY OF THE $\mathcal{C}(J, 2J, L)$ SC-LDPC-BC ENSEMBLES

Ensemble	Design Rate R_L	Variable node degree	Avg. check node degree
$\mathcal{C}(3,6,L)$	$(L-2)/2L$	3	$6L/(L+2)$
$\mathcal{C}(4,8,L)$	$(L-3)/2L$	4	$8L/(L+3)$
$\mathcal{C}(5,10,L)$	$(L-4)/2L$	5	$10L/(L+4)$
$\mathcal{C}(J,2J,L)$	$(L-J+1)/2L$	J	$2JL/(L+J-1)$

In the following example, we consider how the distance growth rates of SC-LDPC-BC ensembles are affected by increasing the density of the graph.

Example 5: Consider the $\mathcal{C}(J, 2J, L)$ SC-LDPC-BC ensembles. The design rates R_L of these SC-LDPC-BC ensembles approach the design rates $R = 1/2$ of the associated unterminated $\mathcal{C}(J, 2J)$ SC-LDPC-CC ensembles as $L \to \infty$. As we increase the variable node degree J, the graph density, and hence the iterative decoding complexity (commonly measured as the average variable and check node degrees), grows. Table II describes the complexity of the $\mathcal{C}(J, 2J, L)$ ensembles. For finite L, the average check node degree of the $\mathcal{C}(J, 2J, L)$ ensemble is strictly less than $2J$ (the check node degree of the $\mathcal{C}(J, 2J)$ SC-LDPC-CC ensemble). The check node degree increases with L, tending to $2J$ as L tends to infinity. The variable node degree remains constant at J for all coupling lengths L.

Fig. 9 plots the minimum distance growth rates for $\mathcal{C}(J, 2J, L)$ code ensembles with $J = 3, 4$, and 5, some (J, K)-regular LDPC-BC ensembles, and the Gilbert-Varshamov bound [69], [70]. As with the $\mathcal{C}(3, 6, L)$ ensembles analyzed in Example 2, we find that the $\mathcal{C}(4, 8, L)$ and $\mathcal{C}(5, 10, L)$ ensembles are asymptotically good, with large minimum distance growth rates for the lower rate ensembles corresponding to small L; then, as the coupling length L is increased, we observe declining minimum distance growth rates as the code rates increase. We again observe that the scaled minimum distance growth rates $\delta_{\min}^{(L)} L/(w+1)$ converge as L increases, which allows us to estimate the growth rates

[6] In this context, the 'weight' d_i associated with a particular variable node v_i in the protograph refers to the portion of the overall Hamming weight d of a codeword that is distributed over the M variable nodes of type v_i in the M-fold graph cover. Since we use M copies of the protograph, the weight associated with a particular variable node in the protograph can be as large as M.

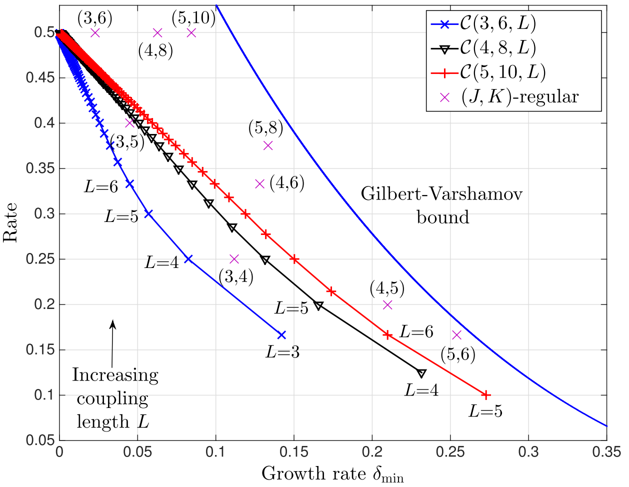


Fig. 9. Minimum distance growth rates for $\mathcal{C}(J, 2J, L)$ SC-LDPC-BC ensembles with design rate $R_L = (L - J + 1)/2L$ and some (J, K)-regular LDPC-BC ensembles with design rate $R = 1 - J/K$. Also shown is the Gilbert-Varshamov bound for random block code minimum distance growth rates.

for $L > 20$ (as explained further in Section IV). As expected, there is a significant increase observed in the growth rates of the $\mathcal{C}(4, 8, L)$ ensembles compared to the $\mathcal{C}(3, 6, L)$ ensembles of the same rate, and there is a smaller improvement for the $\mathcal{C}(5, 10, L)$ ensembles. We would expect this trend to continue as we further increase the variable node degree J. □

B. Thresholds for the BEC

In this section, we assume that BP decoding is performed after transmission over a BEC with erasure probability ε. In every decoding iteration, all of the check nodes are updated followed by all of the variable nodes. The messages that are passed between the nodes represent either an erasure or the correct symbol value (0 or 1). For the BEC, a DE analysis of the BP decoder can be performed for an *unstructured* LDPC-BC ensemble with degree distribution pairs $(\lambda(x), \rho(x))$ explicitly by means of the equation

$$p^{(i)} = \varepsilon\lambda\left(1 - \rho\left(1 - p^{(i-1)}\right)\right), \tag{14}$$

where $p^{(i)}$ denotes the probability that a variable to check node message in decoding iteration i corresponds to an erasure, averaged over all codes in the ensemble. Due to this averaging, the message probabilities are equal for all edges in the graph. The DE threshold of an ensemble, defined as the maximum value of the channel parameter ε for which $p^{(i)}$ converges to zero as i tends to infinity, directly follows from (14). Equation (14) is also the key to the design of degree distribution pairs $(\lambda(x), \rho(x))$ for capacity achieving sequences of codes with a vanishing gap between the threshold and the Shannon limit (capacity) $\varepsilon_{\mathrm{Sh}} = 1 - R$ [6]. Check-concentrated or even check-regular ensembles are known to provide a good trade-off between iterative decoding complexity (measured by the average variable and check node degrees) and gap to capacity.[7]

The lower bounds in [71] on the decoding complexity of general message passing decoders, obtained using sphere-packing arguments, predict a double exponential reduction of the error (erasure) probability with the number of iterations. A double exponential decrease of the decoding erasure probability with iterations implies that the probability of erased frames also converges to zero [72]. A Taylor expansion of (14) reveals that the erasure probability $p^{(i)}$ converges to zero at least doubly exponentially with i if all nodes have a variable node degree of at least three, while an analysis by means of the messages' Bhattacharyya parameter shows that this is also true for general MBS channels [72]. Consequently, protograph-based $\mathcal{C}(J, K, L)$ ensembles achieve this double exponential decay in error probability for $J \geq 3$. We note that the condition that all variable node degrees should be at least 3 is sufficient, but not necessary, for a double exponential decay. For example, it has been shown that structured protograph-based LDPC code ensembles containing degree two variable nodes can achieve the desired double exponential decay [73]; however, unstructured capacity approaching irregular LDPC-BC ensembles containing a large number of degree two variable nodes have frame error (erasure) probabilities bounded away from zero.

Since every member of a protograph-based ensemble preserves the structure of the base protograph, DE analysis for the resulting codes can be performed within the protograph. We now describe the application of DE to *structured* protograph-based ensembles. It is useful to label the edges in E from both a variable node and a check node perspective. Then $e^{v}_{y,l}$ indicates the lth edge emanating from variable node v_y. Similarly, $e^{c}_{x,m}$ denotes the mth edge emanating from check node c_x. Note that $l \in \{1, \ldots, \partial(v_y)\}$ and $m \in \{1, \ldots, \partial(c_x)\}$, where $\partial(v_y)$ and $\partial(c_x)$ denote the degree of variable node v_y and check node c_x, respectively. It follows that if $e^{v}_{y,l}$ and $e^{c}_{x,m}$ define the same edge, v_y is connected to c_x.

For a BEC, let $q^{(i)}(e^{c}_{x,m})$ denote the probability that the check to variable node message sent along edge $e^{c}_{x,m}$ in decoding iteration i is an erasure. (Note that this will be the case if at least one of the incoming messages from other neighboring variable nodes is erased.) Explicitly,

$$q^{(i)}\left(e^{c}_{x,m}\right) = 1 - \prod_{m' \neq m}\left(1 - p^{(i-1)}\left(e^{c}_{x,m'}\right)\right), \tag{15}$$

where $p^{(i-1)}(e^{c}_{x,m'})$ denotes the probability that the incoming message in the previous update of check node x is an erasure and $m, m' \in \{1, \ldots, \partial(c_x)\}$. In contrast, the variable to check node message sent along edge $e^{v}_{y,l}$ is an erasure if the incoming message from the channel and the messages from all the other neighboring check nodes are erasures. This happens with probability $p^{(i)}(e^{v}_{y,l})$, where

$$p^{(i)}\left(e^{v}_{y,l}\right) = \varepsilon \prod_{l' \neq l} q^{(i)}\left(e^{v}_{y,l'}\right) \tag{16}$$

and $l, l' \in \left\{1, \ldots, \partial(v_y)\right\}$. The *BP decoding threshold* ε^* of a protograph-based ensemble is defined as the maximum value of the channel parameter ε for which $p^{(i)}(e^{v}_{y,l})$ converges

[7] Check-concentrated ensembles have a degree distribution such that $\rho(x)$ has two non-zero terms. Check-regular ensembles have a degree distribution such that $\rho(x)$ has precisely one term.

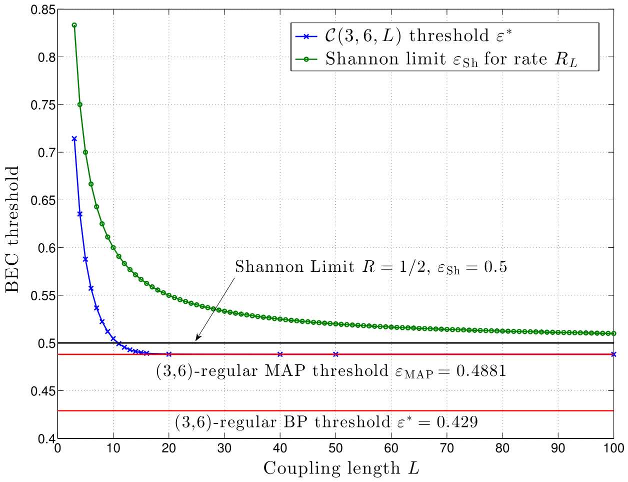


Fig. 10. BEC iterative BP decoding thresholds for $\mathcal{C}(3,6,L)$ SC-LDPC-BC ensembles with design rate $R_L = (L-2)/2L$ and the corresponding Shannon limit $\varepsilon_{\text{Sh}} = 1 - R_L$ for rate R_L. Also shown are the BP and MAP decoding thresholds for the underlying $(3,6)$-regular LDPC-BC ensemble, $\varepsilon^* = 0.429$ and $\varepsilon_{\text{MAP}} = 0.4881$, respectively, and the Shannon limit for $R = 1/2$ codes, $\varepsilon_{\text{Sh}} = 0.5$.

to zero as i tends to infinity for all edges $e^{v}_{y,l}$ emanating from variable node v_y and for all variable nodes v_y in the protograph.

Example 2 (cont.): Fig. 10 shows the calculated BP decoding thresholds ε^* obtained for the $\mathcal{C}(3,6,L)$ SC-LDPC-BC ensembles by recursive application of (15) and (16) for different channel parameters ε. Also shown is the corresponding Shannon limit $\varepsilon_{\text{Sh}} = 1 - R_L$. For small values of L, where the design rate is lower, we observe large thresholds (*e.g.*, $\varepsilon^* = 0.635$ for $L = 3$, where $R_3 = 1/4$ and $\varepsilon_{\text{Sh}} = 0.750$, resulting in a gap to capacity of 0.115). As we increase L, the rate increases and the thresholds decrease; however, the gap to capacity also decreases (*e.g.*, $\varepsilon^* = 0.505$ for $L = 10$, where $R_{10} = 2/5$ and $\varepsilon_{\text{Sh}} = 0.6$, resulting in a gap to capacity of 0.095). When L becomes sufficiently large (in this example around $L = 20$), the threshold converges, or *saturates*, to a constant value $\varepsilon^* = 0.488$. As L is further increased and the rate approaches $R_\infty = 1/2$, the threshold remains constant at $\varepsilon^* = 0.488$, *i.e.*, it displays the remarkable property that it does not continue to decay as the design rate of the ensembles increases and approaches $R_\infty = 1/2$. The Shannon limit is equal to $\varepsilon_{\text{Sh}} = 0.5$ for rate $R_\infty = 1/2$, and thus the gap to capacity decreases with increasing L to the constant value 0.012. Since the gap to capacity improves with increasing L, while the distance growth rate worsens with L (see Table I), this indicates the existence of a trade-off between distance growth rate and threshold.

Also shown in Fig. 10 are the sub-optimal, low complexity BP threshold $\varepsilon^* = 0.429$ and the optimal, high complexity MAP decoding threshold $\varepsilon_{\text{MAP}} = 0.4881$ for the underlying $(3,6)$-regular LDPC-BC ensemble. (Note that even with optimal decoding, there is still a small gap to capacity for a $(3,6)$-regular LDPC-BC ensemble.) We observe that the BP thresholds of the $\mathcal{C}(3,6,L)$ SC-LDPC-BC ensembles are significantly larger than the BP threshold of a $(3,6)$-regular LDPC-BC ensemble for all L. Moreover, the BP thresholds of the $\mathcal{C}(3,6,L)$ ensembles converge to a value numerically indistinguishable from the MAP decoding threshold of a $(3,6)$-regular LDPC-BC ensemble, *i.e.*, threshold saturation is observed. Recall that as $L \to \infty$, $R_\infty = 1/2$ and the $\mathcal{C}(3,6,L)$ ensemble degree distribution approaches $(3,6)$-regular; consequently, the $\mathcal{C}(3,6,L)$ ensemble displays the remarkable property of achieving optimal decoding performance with low complexity BP decoding! As we will observe in the remainder of this section, this phenomenon occurs for all of the protograph-based $\mathcal{C}(J,K,L)$ ensembles. Indeed, it has recently been proven analytically that the BP thresholds of the randomized $\mathcal{C}(J,K,L)$ ensemble described in Section II-E.3 saturate precisely to the MAP decoding thresholds of their underlying (J,K)-regular LDPC-BC ensembles, both for the BEC [21] and for general MBS channels [22]. Finally, we note that, in conjunction with the excellent thresholds, all variable nodes in the $\mathcal{C}(3,6,L)$ ensembles have degree greater than two and thus, asymptotically, the error probability converges at least doubly exponentially with decoding iterations. □

The interesting phenomenon that the calculated thresholds do not decay as L increases beyond a certain value was first observed empirically in [13] for $\mathcal{C}(J,2J,L)$ SC-LDPC-BC ensembles constructed from $M \times M$ permutation matrices, and it was shown to be true for arbitrarily large L in [15]. To prove this result, a sliding window updating schedule can be considered, where the decoder updates the nodes only within a window of size $W \leq L$, starting at time $t = 0$. Once the variable-to-check node message probabilities $p(e^{v}_{y,l})$, $l = 1, 2, \ldots, \partial(v_y)$, are below some value ε_0 for all nodes v_y, $y = 0, 1, \ldots, b_v - 1$, at time t, the window is shifted one time unit further. Suppose that the message probabilities at times $t < 0$ are initialized by some value $\varepsilon_0 > 0$. If, under these conditions, the value ε_0 is reached at time $t = 0$ after some number of iterations, so that the window can be shifted one step further, then, for the actual initial probabilities $p(e^{v}_{y,l}) = 0$ of nodes at times $t < 0$, the value ε_0 can also be reached at all times t, $t = 0, \ldots, L-1$.

Intuitively, one can explain the result as follows: during the iterations, due to the lower check node degrees at the start of the graph, the messages along edges at time $t = 0$ will be the most reliable ones. Their erasure probabilities thus have the potential to converge to zero even for channel parameters ε beyond the threshold of the underlying LDPC-BC ensemble. But when the symbols at $t = 0$ are perfectly known, the connected edges can be removed from the protograph with base matrix $\mathbf{B}_{[0,L-1]}$, which results in a shortened protograph with base matrix $\mathbf{B}_{[1,L-1]}$. It follows now by induction that the messages eventually converge to zero at all times $t = 0, \ldots, L-1$ for an arbitrary coupling length L.

Example 5 (cont.): Fig. 11 shows the BEC iterative decoding thresholds for several $\mathcal{C}(J,2J,L)$ SC-LDPC-BC ensembles. In each case, we observe that the gap to capacity decreases as the coupling length L increases. Also, for a fixed rate and small values of L, we see that the thresholds worsen as we increase J, which is consistent with the behavior observed for fixed rate $(J,2J)$-regular LDPC-BC ensembles,

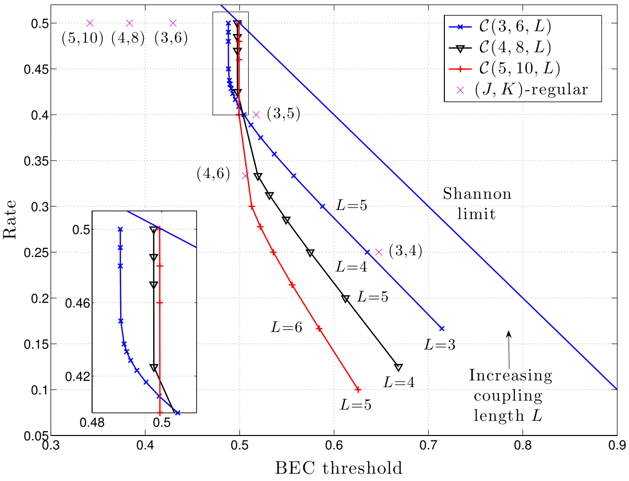


Fig. 11. BEC iterative BP decoding thresholds for $\mathcal{C}(J,2J,L)$ SC-LDPC-BC ensembles with design rate $R_L = (L-J+1)/2L$ and some (J,K)-regular LDPC-BC ensembles with design rate $R = 1 - J/K$.

e.g., the $(3,6)$-, $(4,8)$-, and $(5,10)$-regular ensembles shown in the figure. Recall from Fig. 9 that, for a fixed rate and small values of L, the minimum distance growth rates of the $\mathcal{C}(J,2J,L)$ ensembles improve as we increase J, which is also consistent with the behavior observed for fixed rate $(J,2J)$-regular LDPC-BC ensembles. Thus, in the small L regime, SC-LDPC-BC ensembles behave like LDPC-BC ensembles, *i.e.*, thresholds worsen and distance growth rates improve by increasing J (and hence iterative decoding complexity).

However, as L increases, the thresholds of the $\mathcal{C}(J,2J,L)$ ensembles each saturate to a value numerically indistinguishable from the MAP decoding threshold (and significantly larger than the BP threshold) of the underlying $(J,2J)$-regular LDPC-BC ensemble. These values *improve*, rather than worsen, as we increase J ($\varepsilon^* = 0.4881$, 0.4977, and 0.4994 for the $\mathcal{C}(3,6,L)$, $\mathcal{C}(4,8,L)$, and $\mathcal{C}(5,10,L)$ ensembles, respectively). This indicates that, unlike the underlying $(J,2J)$-regular LDPC-BC ensembles, for large L, both the distance growth rates *and* the BP thresholds improve with increasing complexity, and we would expect this trend to continue as we further increase the variable node degree J. Moreover, as we let $J \to \infty$, the MAP threshold (for an arbitrary MBS channel) of the underlying $(J,2J)$-regular LDPC-BC ensemble improves all the way to the Shannon limit [74]. This allows the construction of *capacity achieving* $\mathcal{C}(J,2J,L)$ SC-LDPC-BC ensembles with BP decoding as the graph density grows unbounded.[8] □

The results described so far are indicative of the general behavior of $\mathcal{C}(J,K,L)$ SC-LDPC-BC ensembles. In practice, the coupling length L adds an extra degree of freedom. We obtain a family of asymptotically good ensembles with varying iterative decoding thresholds and minimum distance growth rates covering a wide variety of design rates. Moreover, the desired range of achievable SC-LDPC-BC design rates can be extended by coupling higher or lower rate LDPC-BC protographs together using the Edge Spreading Rule.

Example 6: In this example, we consider the $\mathcal{C}(3,12,L)$, $\mathcal{C}(3,9,L)$, $\mathcal{C}(3,6,L)$, and $\mathcal{C}(4,6,L)$ SC-LDPC-BC ensembles. Each ensemble has design rate approaching $R_\infty = 1 - J/K$ (the rate of the underlying (J,K)-regular LDPC-BC ensemble), with the usual structured irregularity occurring as a result of the termination. Fig. 12 displays the BEC thresholds and distance growth rates of these $\mathcal{C}(J,K,L)$ ensembles and, for comparison, several uncoupled (J,K)-regular LDPC-BC ensembles, along with the Shannon limit and the Gilbert-Varshamov bound, respectively. For each family, when L is small and the design rate is low, the iterative decoding thresholds are further from capacity and the minimum distance growth rates are larger compared to the ensembles with larger L. Then, as L increases, the gap to capacity decreases and the BP threshold saturates to a value close to the Shannon limit (*i.e.*, the MAP decoding threshold of the underlying (J,K)-regular LDPC-BC ensemble) and significantly better than the BP threshold of the underlying (J,K)-regular LDPC-BC ensemble. It follows that, as in the previous examples, we observe a minimum distance vs. threshold trade-off for each of these $\mathcal{C}(J,K,L)$ ensembles, since both the minimum distance growth rates and the gap to capacity decrease with increasing L. Finally, we note that the design rates R_L of the $\mathcal{C}(J,K,L)$ ensembles included in Fig. 12, given by (9), cover a wide range of values. □

The choice of edge spreading affects the properties of the SC-LDPC-BC ensembles, in particular for small to moderate values of L. In the next example, we will see that it is possible to improve both the minimum distance growth rates and thresholds simultaneously by carefully selecting the edge spreading.

Example 7: In this example, we consider a different edge spreading than that chosen for the $\mathcal{C}(3,6,L)$ ensembles. Fig. 13(a) shows the $(3,6)$-regular protograph used to construct the $\mathcal{C}(3,6,L)$ ensembles. This protograph is copied L times and an edge spreading with coupling width $w = 1$ is applied as shown in Fig. 13(b). We will denote the SC-LDPC-BC ensembles obtained with this edge spreading as $\mathcal{C}_A(3,6,L)$. Comparing the $\mathcal{C}_A(3,6,L)$ ensembles to the $\mathcal{C}(3,6,L)$ ensembles, we notice two major differences. Structurally, the $\mathcal{C}_A(3,6,L)$ ensembles are more regular. The first and last check nodes have degree 3 and all the other check nodes have degree 6. Secondly, since $w = 1$, the ensemble design rate $R_L = (L-1)/2L$ obtained using (9) is larger for a given L, *i.e.*, there is less rate loss. Asymptotically in L, both ensembles approach $R_\infty = 1/2$ and a $(3,6)$-regular degree distribution.

The BEC thresholds and distance growth rates calculated for the $\mathcal{C}(3,6,L)$ and $\mathcal{C}_A(3,6,L)$ ensembles are displayed in Table III. We observe that, even though the $\mathcal{C}_A(3,6,L)$ ensembles have a very small structured irregularity (only one reduced degree check node at either end of the chain), like the $\mathcal{C}(3,6,L)$ ensembles their BP thresholds still saturate to the optimal MAP decoding threshold of the underlying

[8] Here we observe numerically that the BP thresholds of the $\mathcal{C}(J,2J,L)$ ensembles saturate to the MAP threshold of the underlying (J,K)-regular LDPC-BC ensemble. Hence these ensembles are not capacity achieving in the strict sense. However, the randomized $\mathcal{C}(J,2J,L)$ ensembles in [21] and [22] are provably capacity achieving in this regard.

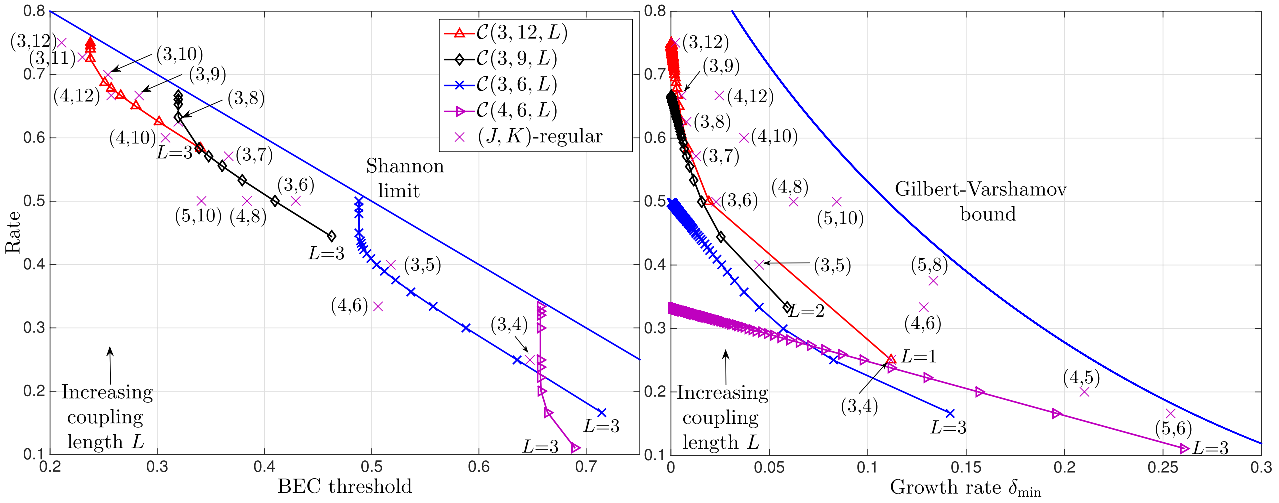


Fig. 12. BEC iterative BP decoding thresholds and minimum distance growth rates of four $\mathcal{C}(J, K, L)$ SC-LDPC-BC ensembles and several uncoupled (J, K)-regular LDPC-BC ensembles.

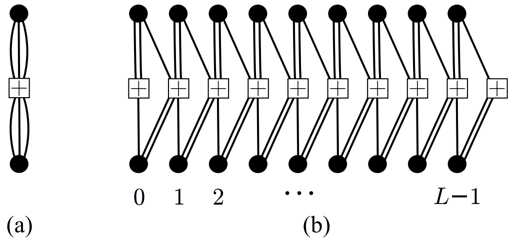


Fig. 13. Protographs of (a) a $(3, 6)$-regular LDPC-BC ensemble, and (b) a resulting SC-LDPC-BC ensemble with coupling length L obtained by applying the Edge Spreading Rule with coupling width $w = 1$.

TABLE III

BEC THRESHOLDS AND DISTANCE GROWTH RATES FOR SC-LDPC-BC ENSEMBLES OBTAINED BY TWO DIFFERENT EDGE SPREADINGS OF A $(3, 6)$-REGULAR LDPC-BC PROTOGRAPH

Design Rate	$\mathcal{C}(3,6,L)$		$\mathcal{C}_A(3,6,L)$	
R_L	ε^*	$\delta_{\min}^{(L)}$	ε^*	$\delta_{\min}^{(L)}$
1/4	0.6353	0.0814	0.6448	0.0950
1/3	0.5574	0.0449	0.5671	0.0524
3/8	0.5223	0.0324	0.5301	0.0375
2/5	0.5046	0.0258	0.5103	0.0298
5/12	0.4955	0.0215	0.4993	0.0248
3/7	0.4911	0.0184	0.4933	0.0213
7/16	0.4892	0.0161	0.4903	0.0186
19/40	0.4881	0.0065	0.4881	0.0074
1/2	0.4881	0	0.4881	0

$(3, 6)$-regular LDPC-BC ensemble. Moreover, the $\mathcal{C}_A(3, 6, L)$ ensembles are also asymptotically good and display *both* larger growth rates and better thresholds than the $\mathcal{C}(3, 6, L)$ ensembles.

The larger distance growth rates obtained for the $\mathcal{C}_A(3, 6, L)$ ensembles can be attributed to having no degree 2 check nodes and a larger proportion of non-zero elements in $\mathbf{B}_{[0,L-1]}$, *i.e.*, a denser base matrix. We note that by retaining some repeated edges in the $\mathcal{C}_A(3, 6, L)$ ensemble protograph, the memory requirements for decoder implementation are reduced, *i.e.*, the SC-LDPC-CC constraint length is $\nu = M(w + 1)b_v = 4M$, compared to $\nu = M(w + 1)b_v = 6M$ for the $\mathcal{C}(3, 6, L)$ ensembles. Moreover, constructing SC-LDPC-CC ensembles from protographs with repeated edges in order to reduce memory requirements has been shown to improve the performance of a windowed decoder [52]. □

There are many ways of spreading the edges among the component submatrices $\mathbf{B}_i$ of a base matrix $\mathbf{B}$, and different constructions can result in varying thresholds and ensemble growth rates. (For some other examples of different $(3, 6)$-regular edge spreadings see [75], [76].) Choices containing all-zero rows and/or columns in the submatrices should be avoided, since they can lead to disconnected subgraphs. Simple row and column permutations (applied to all component submatrices) do not affect the graph structure, and so, in turn, they do not affect the threshold and distance growth rate of the ensemble. A good threshold is expected when the check nodes at the ends of the graph have low degree (but at least degree 2). This gives an initial convergence boost to the iterative decoder, and the spatially coupled structure allows this reliable information generated at the ends of the graph to propagate through the chain to the center.

We conclude this section by investigating the minimum distance and threshold trade-off for the irregular $\mathcal{C}_{\text{ARJA}}(L)$ and $\mathcal{C}_{\text{AR4JA}}(L)$ SC-LDPC-BC ensembles.

Example 4 (cont.): The minimum distance growth rates and BEC iterative decoding thresholds for the $\mathcal{C}_{\text{ARJA}}(L)$ ensembles are given in Table IV. (For reference, the underlying ARJA LDPC-BC ensemble has minimum distance growth rate $\delta_{\min} = 0.0145$ and BEC threshold $\varepsilon^* = 0.4387$.) Similar to the $\mathcal{C}(J, K, L)$ ensembles, we observe that the ensembles are each asymptotically good; but as the coupling length $L \to \infty$, the minimum distance growth rate $\delta_{\min}^{(L)} \to 0$. We also observe from Table IV that the scaled growth rates $n_t\delta_{\min}^{(L)}$ converge

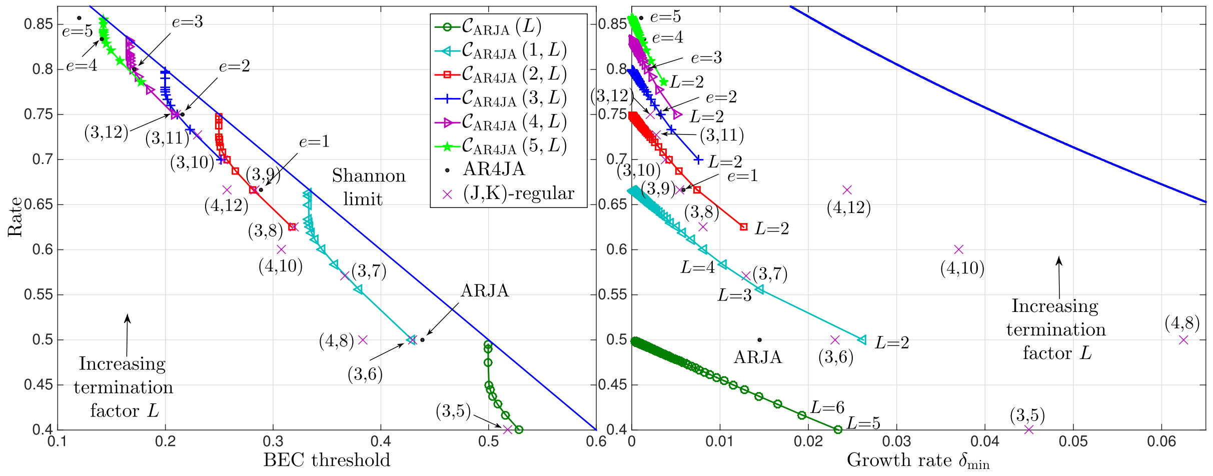


Fig. 14. BEC iterative BP decoding thresholds and minimum distance growth rates for the $\mathcal{C}_{\text{AR4JA}}(e, L)$ SC-LDPC-BC ensembles, the underlying AR4JA LDPC-BC ensembles, and several (J, K)-regular LDPC-BC ensembles.

TABLE IV
DISTANCE GROWTH RATES AND BEC THRESHOLDS FOR THE ARJA SC-LDPC-BC ENSEMBLES $\mathcal{C}_{\text{ARJA}}(L)$

L	Rate R_L	Growth rate $\delta_{\min}^{(L)}$	Scaled $n_t\delta_{\min}^{(L)}$	BEC threshold	Capacity ε_{sh}	Gap to Capacity
2	1/4	0.0946	0.757	0.6608	0.7500	0.0892
3	1/3	0.0461	0.553	0.5864	0.6667	0.0803
4	3/8	0.0306	0.490	0.5496	0.6250	0.0750
5	2/5	0.0234	0.469	0.5284	0.6000	0.0716
6	5/12	0.0192	0.462	0.5159	0.5833	0.0674
7	3/7	0.0164	0.461	0.5083	0.5714	0.0631
8	7/16	0.0144	0.461	0.5039	0.5625	0.0586
9	4/9	0.0128	0.461	0.5016	0.5556	0.0540
10	9/20	0.0115	0.461	0.5004	0.5500	0.0496
∞	1/2	0		0.4996	0.5000	0.0004

as L increases. We will see in Section IV that the scaled growth rates converge to a bound on the *free distance* growth rate of the unterminated $\mathcal{C}_{\text{ARJA}}$ ensemble. This allows us to estimate the minimum distance growth rate $\delta_{\min}^{(L)}$ of the $\mathcal{C}_{\text{ARJA}}(L)$ ensembles for large L by dividing this value by the number of transmitted nodes in the protograph $n_t = 4L$.

In addition, we see the same type of threshold behavior exhibited by the $\mathcal{C}(J, K, L)$ ensembles. The BEC iterative decoding threshold saturates to $\varepsilon^* = 0.4996$ as L becomes sufficiently large and does not further decay as $L \to \infty$. This is very close to the Shannon limit $\varepsilon_{sh} = 0.5$ for rate $R_\infty = 1/2$ and is significantly larger than the BP threshold of the ARJA LDPC-BC ensemble. As the coupling length L increases, we also observe that the gap to capacity decreases, resulting in the usual trade-off between distance growth rate and threshold.

Fig. 14 shows the results obtained for the $\mathcal{C}_{\text{AR4JA}}(e, L)$ SC-LDPC-BC ensembles, the underlying AR4JA LDPC-BC ensembles, and several (J, K)-regular LDPC-BC ensembles. For the $\mathcal{C}_{\text{AR4JA}}(e, L)$ ensembles with $e = 1, \ldots, 5$, we observe that, as in the $e = 0$ case, increasing the coupling length L results in asymptotically good code ensembles with capacity approaching iterative decoding thresholds and declining minimum distance growth rates. For each family, the iterative decoding threshold converges to a value close to the Shannon limit for R_∞ (and significantly larger than the BP threshold of the underlying AR4JA LDPC-BC ensemble) as L gets large. The design rates R_L of the $\mathcal{C}_{\text{AR4JA}}(e, L)$ ensembles overlap for increasing extension parameter e, allowing a large selection of asymptotically good codes to be obtained in the rate range $1/4 \leq R \leq 6/7$, and the achievable code rate can be increased further by considering larger extension parameters e.

We also observe that the minimum distance growth rates of the $\mathcal{C}_{\text{AR4JA}}(e, L)$ ensembles for small coupling lengths L typically exceed those of $(3, K)$-regular codes for $K \geq 6$. Further, for the same extension parameter e and large L, the $\mathcal{C}_{\text{AR4JA}}(e, L)$ ensembles have significantly better thresholds and less complexity than the underlying AR4JA LDPC-BC ensembles,[9] but smaller distance growth rates and slightly lower code rates. Further, by increasing the extension parameter e, and for small L, the minimum distance growth rates of the $\mathcal{C}_{\text{AR4JA}}(e, L)$ ensembles are larger than those of the AR4JA ensemble with only a slightly worse threshold and some increase in complexity.

Fig. 15 shows the minimum distance growth rates against the fractional gap to capacity $(\varepsilon_{sh} - \varepsilon^*)/\varepsilon_{sh}$ for the $\mathcal{C}_{\text{AR4JA}}(e, L)$ SC-LDPC-BC ensembles with coupling lengths $L = 2, \ldots, 10, 20, 50, 100$, the underlying AR4JA LDPC-BC ensembles, and several (J, K)-regular LDPC-BC ensembles. The trade-off we observe effectively allows a code designer to 'tune' between distance growth rate and threshold by choosing the parameters e and L. We observe that, in particular, intermediate values of L provide thresholds with a small gap to capacity while maintaining a reasonable distance growth rate with only a small loss in code rate. The complexity of the $\mathcal{C}_{\text{AR4JA}}(e, L)$ ensembles (measured by average variable and

[9] Complexity, as earlier, is measured by average variable and check node degrees. When comparing the $\mathcal{C}_{\text{AR4JA}}(e, L)$ ensembles to the underlying AR4JA LDPC-BC ensembles with equal extension parameters, the average variable node degree is the same for all L, but the average check node degree is less for the $\mathcal{C}_{\text{AR4JA}}(e, L)$ ensembles because of the termination.

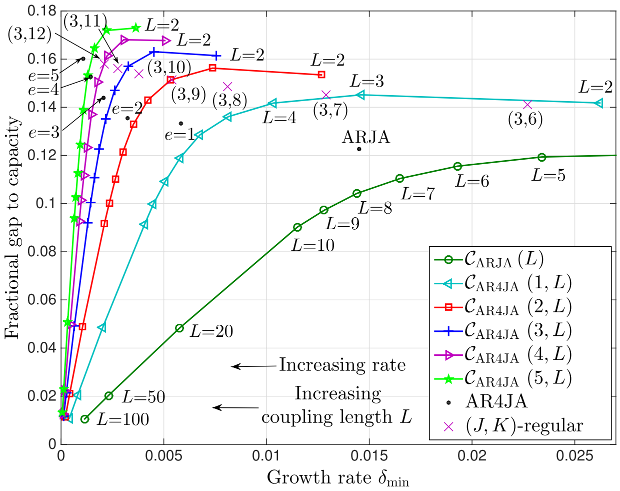


Fig. 15. Minimum distance growth rate vs. the fractional gap to capacity for the $\mathcal{C}_{\text{AR4JA}}(e, L)$ SC-LDPC-BC ensembles, the underlying AR4JA LDPC-BC ensembles, and several (J, K)-regular LDPC-BC ensembles.

check node degrees) increases slowly with L and approaches that of the underlying AR4JA LDPC-BC ensemble for a given extension parameter e. Further, as L becomes sufficiently large for the scaled growth rates to converge, we observe that the gaps to capacity are approximately proportional to L for all of the $\mathcal{C}_{\text{AR4JA}}(e, L)$ ensembles. For example, we obtain about a 10% gap to capacity by terminating after $L = 9$ time instants; a 5% gap after $L = 20$ time instants; a 2% gap after $L = 50$ time instants; and a 1% gap after $L = 100$ time instants. Finally, choosing the extension parameter e allows additional flexibility, where a larger e gives a higher code rate but a lower distance growth rate and greater complexity. □

C. Thresholds for the AWGNC

In this section, we perform an AWGNC threshold analysis of protograph-based SC-LDPC-BC ensembles and show that the dramatic threshold improvement obtained by terminating SC-LDPC-CC also extends to the AWGNC. Exact DE is far more complex for the AWGNC than for the BEC since the densities are vectors not scalars and one must also consider a density for each individual edge in the graph rather than simply an erasure probability. Consequently, it is only feasible for simple protographs, so here we make use of the reciprocal channel approximation (RCA) technique introduced in [77], which has been successfully applied to the analysis of protograph-based ensembles in [48]. With this approach, the calculation of approximate AWGNC thresholds for a variety of regular and irregular protographs becomes feasible with reasonable accuracy.

Example 2 (cont.): In Fig. 16, we plot the AWGNC BP thresholds (in terms of the noise standard derivation σ) obtained using the RCA technique for several $\mathcal{C}(3, 6, L)$ SC-LDPC-BC ensembles along with the Shannon limit for the given design rate $R_L = (L-2)/2L$. We observe the same behavior as demonstrated for the BEC in Fig. 10: we find that the threshold decreases monotonically with increasing rate, but the gap to capacity for the given rate also decreases. For example, the threshold values are equal to $\sigma^* = 1.446$ for $L = 3$ and $\sigma^* = 0.9638$ for $L = 10$. As L is further increased, the thresholds saturate to a constant value $\sigma^* = 0.948$ and do not further decay as $L \to \infty$ and $R_L \to 1/2$. We observe that $\sigma^* = 0.948$, which is equal to the MAP decoding threshold of the underlying $(3, 6)$-regular LDPC-BC ensemble, is much closer to the Shannon limit $\sigma_{\text{Sh}} = 0.979$ than the BP threshold $\sigma^* = 0.881$ of the $(3, 6)$-regular LDPC-BC ensemble. □

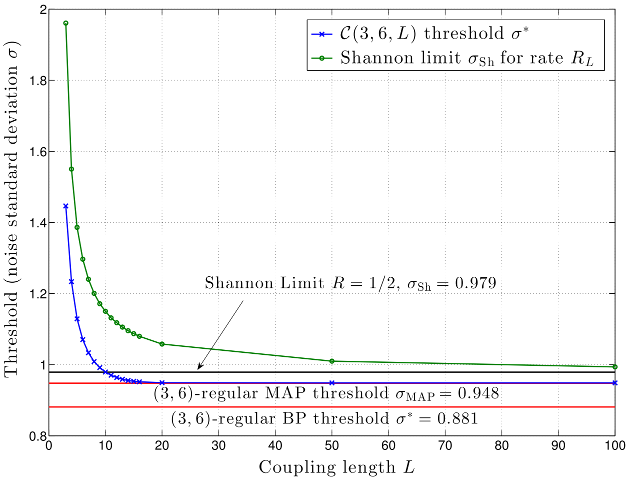
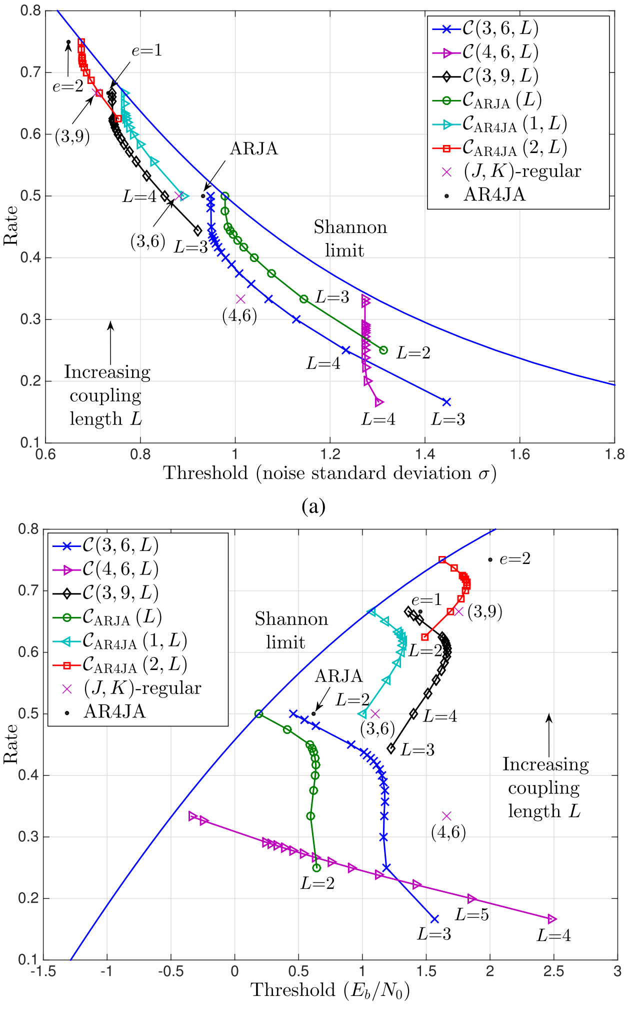


Fig. 16. AWGNC BP thresholds in terms of the noise standard deviation σ for $\mathcal{C}(3, 6, L)$ SC-LDPC-BC ensembles with design rate $R_L = (L-2)/2L$ and the corresponding Shannon limit for rate R_L. Also shown for comparison are the BP and MAP decoding thresholds for the underlying $(3, 6)$-regular LDPC-BC ensemble, $\sigma^* = 0.881$ and $\sigma_{\text{MAP}} = 0.948$, respectively, and the Shannon limit for $R = 1/2$ codes, $\sigma_{\text{Sh}} = 0.979$.

This behavior, which can be observed for all of the regular and irregular ensembles considered, is similar to the corresponding results for the BEC, presented in Section III-B. In Fig. 17, we display calculated thresholds for the AWGNC for a variety of $\mathcal{C}(J, K, L)$, $\mathcal{C}_{\text{ARJA}}(L)$, and $\mathcal{C}_{\text{AR4JA}}(e, L)$ ensembles along with the thresholds of the underlying LDPC-BC ensembles. Fig. 17(a) plots the thresholds in terms of the standard deviation σ against the ensemble design rate. We observe that as L increases, the design rate increases and the threshold decreases monotonically; however, as L becomes sufficiently large, the thresholds saturate to the MAP thresholds of the underlying LDPC-BC ensembles. Further, they are close to the Shannon limit and, importantly, they do not decrease further as $L \to \infty$ and the design rate and degree distribution approach those of the underlying LDPC-BC ensembles.

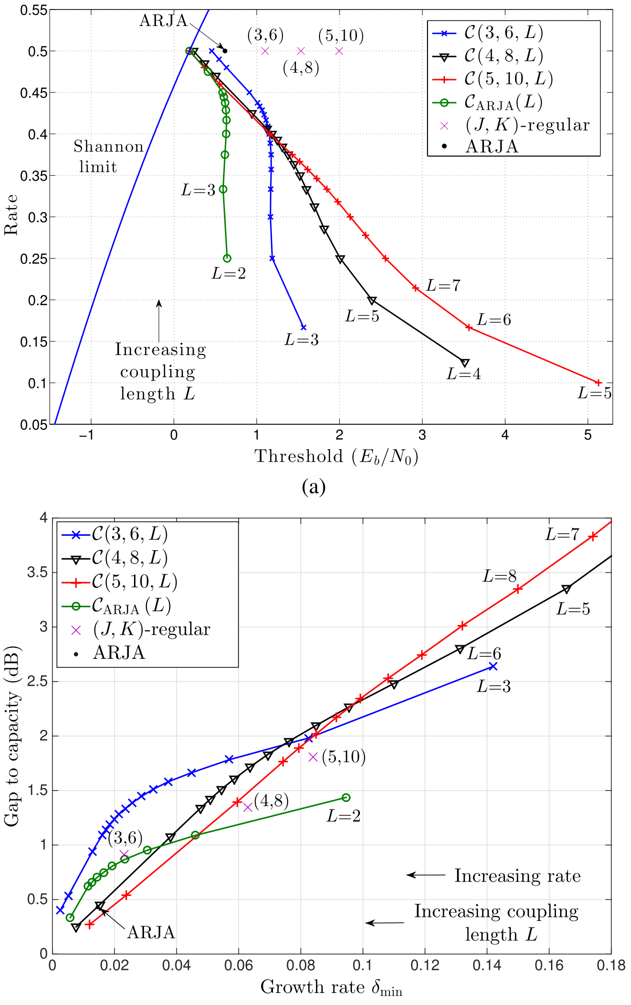
The same thresholds are depicted in Fig. 17(b) in terms of the SNR E_b/N_0. Since E_b/N_0 takes into account the code rate overhead, the ensembles with lower rate have a larger noise variance, and the monotonic behavior of the thresholds noted in Fig. 17(a) is no longer visible. In both plots, however, we see that the gap to capacity decreases with increasing L. Consequently, in a similar fashion to the BEC, varying the coupling length L results in SC-LDPC-BC ensembles with different design rates and a trade-off between iterative decoding threshold and minimum distance growth rate.



(a)

(b)

Fig. 17. AWGNC BP thresholds in terms of (a) standard deviation σ and (b) signal-to-noise ratio E_b/N_0 (dB) for several families of SC-LDPC-BC ensembles with different coupling lengths L.



(a)

(b)

Fig. 18. Comparison of the $\mathcal{C}(J,2J,L)$ SC-LDPC-BC ensembles, the $\mathcal{C}_{\mathrm{ARJA}}(L)$ SC-LDPC-BC ensembles, and the underlying LDPC-BC ensembles: (a) AWGNC thresholds in terms of SNR E_b/N_0 (dB), and (b) minimum distance growth rate vs. threshold gap to capacity.

To conclude this section, we examine the AWGNC BP thresholds for various SC-LDPC-BC ensembles with design rates approaching $R_\infty = 1/2$ and varying graph densities. In addition, we use simulation to examine the finite length performance of SC-LDPC-BC ensembles on the AWGNC and demonstrate that the excellent performance promised by the asymptotic results also translates into improved decoding performance for finite code lengths.

Example 5 (cont.): We consider once more several $\mathcal{C}(J,2J,L)$ ensembles and the $\mathcal{C}_{\mathrm{ARJA}}(L)$ ensembles in Fig. 18(a). We observe that, as L increases, the ensemble design rate of the SC-LDPC-BC ensembles increases (approaching $R = 1/2$ asymptotically) and the thresholds improve, nearing the Shannon limit for large L. Further, we note that the $(J,2J)$-regular LDPC-BC ensemble thresholds worsen as we increase J, and we see that this is also the case for the $\mathcal{C}(J,2J,L)$ ensembles on the AWGNC for small coupling lengths L, as was previously noted for the BEC. Thus, for $(J,2J)$-regular LDPC-BC ensembles and $\mathcal{C}(J,2J,L)$ ensembles with small L, thresholds worsen and distance growth rates improve by increasing J (and hence iterative decoding complexity). However, as L increases, the behavior of the $\mathcal{C}(J,2J,L)$ ensembles changes and their thresholds saturate to a value numerically indistinguishable from the MAP decoding threshold of the underlying $(J,2J)$-regular LDPC-BC ensemble and this value approaches the Shannon limit as we increase J. This indicates that, for large L, both the distance growth rates and the thresholds improve with increasing complexity, and we expect this trend to continue as we further increase the variable node degree J, although the improvement will diminish with increasing J. As a final observation, we note that for all achievable rates the $\mathcal{C}_{\mathrm{ARJA}}(L)$ ensembles have better thresholds than the $\mathcal{C}(3,6,L)$ ensembles. This is expected, since the ARJA LDPC-BC ensemble has been designed to have a good

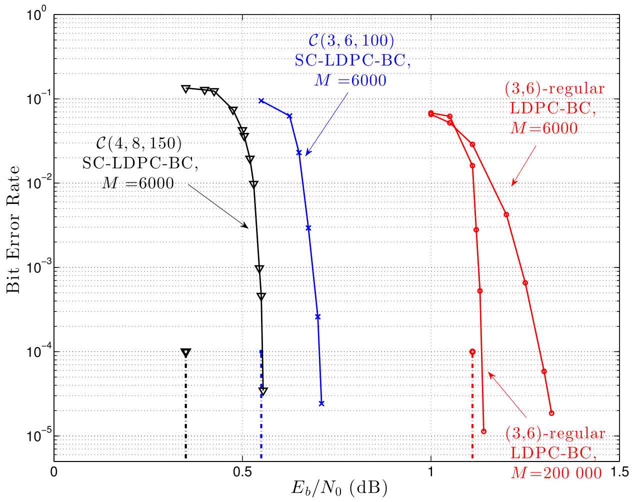


Fig. 19. AWGNC decoding performance (solid lines) and BP decoding thresholds (dashed lines) of $\mathcal{C}(3,6,100)$ and $\mathcal{C}(4,8,150)$ SC-LDPC-BCs with lifting factor $M = 6000$ and rate $R = 0.49$. For comparison, the performance of $(3,6)$-regular LDPC-BCs with $M = 6000$ and $M = 200000$ are also shown along with the associated BP decoding threshold. (The BP thresholds of $(4,8)$- and $(5,10)$-regular LDPC-BCs are 1.61dB and 2.04dB, respectively.)

iterative decoding threshold. However, we also observe that, for large L, the $\mathcal{C}(4,8,L)$ and $\mathcal{C}(5,10,L)$ ensembles have comparable thresholds to the $\mathcal{C}_{\text{ARJA}}(L)$ ensembles, demonstrating the benefit that derives from the spatially coupled convolutional structure, *i.e.*, we obtain near capacity threshold performance with an almost regular code graph.

Fig. 18(b) plots the minimum distance growth rates against the threshold gap to capacity (the difference between the AWGNC threshold (in terms of E_b/N_0) of an ensemble and capacity for the ensemble design rate) for the $\mathcal{C}(J,2J,L)$ ensembles with coupling lengths $L = w+1,\ldots,16,20,50,100$, the $\mathcal{C}_{\text{ARJA}}(L)$ ensembles with $L = 2,\ldots,10$, and the underlying $(J,2J)$-regular and ARJA LDPC-BC ensembles. We observe that, in particular, intermediate values of L provide thresholds with a small gap to capacity while maintaining linear minimum distance growth with only a slight loss in code rate. We also note that, for a fixed gap to capacity close to zero, the largest minimum distance growth rate is obtained by choosing the $\mathcal{C}(J,2J,L)$ ensemble with the largest J, and that the $\mathcal{C}_{\text{ARJA}}(L)$ ensembles falls in between the $\mathcal{C}(3,6,L)$ and $\mathcal{C}(4,8,L)$ ensembles. (In this region, with the gap to capacity close to zero, the rates of all the ensembles are approximately equal and close to $1/2$.) For larger fixed gaps to capacity, we see that the order changes and that the reverse order holds for large gaps to capacity.

Now consider choosing L such that the ensemble design rate is $R = 0.49$ and the BP threshold values of the $\mathcal{C}(J,2J,L)$ ensembles improve with J. The thresholds of the $\mathcal{C}(3,6,100)$ and $\mathcal{C}(4,8,150)$ ensembles are shown in Fig. 19, along with the simulated performance of randomly chosen codes from the underlying ensembles with permutation matrix size $M = 6000$. Also shown, for comparison, is the BP threshold of the $(3,6)$-regular LDPC-BC with design rate $R = 0.5$ along with the simulated decoding performance of two randomly chosen codes from the ensemble with permutation matrix sizes $M = 6000$ and $M = 200,000$. A standard LDPC-BC decoder employing the BP decoding algorithm was used in each case. For the SC-LDPC-BCs with $M = 6000$, we observe that the waterfall performance is within 0.2dB of the threshold, and we expect the gap to decrease for larger permutation matrix sizes M. By choosing a larger L, the rate increases (approaching $1/2$), and the thresholds and corresponding waterfall performance of codes chosen from these ensembles will improve slightly. Note, in particular, that the SC-LDPC-BCs are operating far beyond the threshold of the $(3,6)$-regular LDPC-BC ensemble, and as the graph density J increases this improvement will become more pronounced, since the thresholds and corresponding waterfall performance of the SC-LDPC-BCs gets better whereas the thresholds and performance of the LDPC-BCs will become worse. □

Remark 8 (Importance of Coupling Length L): From the BP threshold results presented in Sections III-B and III-C, it follows that one should select a large coupling length L where possible, since the gap to capacity decreases with increasing L (see, *e.g.*, Figs. 12 and 17). As will be discussed further in the next section, the strength of SC-LDPC-BCs is in fact *independent of* L, and consequently one should avoid the viewpoint of comparing a SC-LDPC-BC to a large LDPC-BC with an overall block length of $n = LMb_v$ that is decoded using standard LDPC-BC decoding techniques. In practice, the implementation of a windowed decoder is *crucial* to reduce the memory, latency, and complexity requirements of SC-LDPC-BCs; moreover, the window size W, lifting factor M, and coupling width w are the design parameters that must be optimized for any given system constraints, whereas L is less important but should not be chosen to be small. Such design aspects have been investigated in the literature and simulated decoding performance of SC-LDPC-BCs for various W, M, w, and L, can be found in [12], [44], [51], and [65].

IV. Free Distance Growth Rates of SC-LDPC-CC Ensembles

In the previous section, we saw that the BP decoding thresholds of SC-LDPC-BC ensembles tend toward the MAP thresholds of the underlying LDPC-BC ensembles on both the BEC and AWGNC with increasing coupling length L, while the minimum distance growth rates $\delta_{\min}^{(L)}$ tend to zero as $L \to \infty$. Considering the SC-LDPC-BC as an LDPC-BC with given finite block length $n = MLb_v$, a careful choice of the parameters M and L becomes necessary to achieve the best performance. The minimum distance growth rate $\delta_{\min}^{(L)}$ provides a useful measure for comparison of the distance properties of SC-LDPC-BC ensembles; however, based on their spatially coupled (convolutional) structure, it is clear that the potential strength of SC-LDPC-BC ensembles for large L scales with the constraint length $\nu = M(w+1)b_v$, which increases with M but is independent of L. Consequently, the free distance growth rate δ_{free} of the closely related SC-LDPC-CC ensemble, which is independent of L, is a more appropriate measure of the performance of SC-LDPC-BC ensembles than their minimum distance growth rates $\delta_{\min}^{(L)}$.

This fact is supported by the excellent decoding performance of a continuous sliding window decoder [15], [51], [52], which only passes messages across a window of fixed size W, typically a small multiple of the constraint length ν (independent of L), as opposed to passing messages directly across the entire length of the graph (which grows with L), like a standard LDPC-BC BP decoder. Provided that W is not chosen to be too small, the strength of the SC-LDPC-BC is contained within the window and there is no perceivable loss in performance compared to a standard decoder [51], [52]. Moreover, this strategy is of particular practical importance since it allows one to fully exploit the localized structure of SC-LDPC-BC ensembles in terms of minimizing decoding latency and memory requirements.

In the remainder of this section, we investigate the connection between the minimum distance of SC-LDPC-BC ensembles and the free distance of the closely related SC-LDPC-CC ensembles.

Definition 9: The minimum *free distance* of a convolutional code, denoted by d_{free}, is defined as the minimum Hamming distance between any two distinct code sequences in the code $\mathbf{x}_{[0,\infty]}$ and $\mathbf{y}_{[0,\infty]}$. Since convolutional codes are linear, this condition simplifies to

$$d_{\text{free}} = \min_{\mathbf{x}_{[0,\infty]} \neq \mathbf{0}} w(\mathbf{x}_{[0,\infty]}),$$

where $w(\cdot)$ denotes the Hamming weight of the argument, i.e., d_{free} is the weight of the minimum Hamming weight nonzero code sequence.

In order to obtain our result, we introduce a sub-ensemble of the SC-LDPC-CC ensemble given in Definition 2 where each member is *periodically time-varying*.

Definition 10: An ensemble of *periodically time-varying* protograph-based SC-LDPC-CCs with coupling width w, design rate $R = 1 - b_c/b_v$, *constraint length* $\nu = M(w+1)b_v$, and period T is obtained as the collection of all M-fold graph covers of a convolutional protograph where the permutation applied to edge l of variable node v_y, $l \in \{1, \ldots, \partial(v_y)\}$, $y \in \{0, 1, \ldots, b_v - 1\}$, at time t is also applied to edge l at times $\{t + kT | k \in \mathbb{Z}\backslash 0\}$ and T is the smallest natural number for which this condition holds.

To avoid confusion with the notation, we will refer to the periodically time-varying SC-LDPC-CC sub-ensembles of the $\mathcal{C}(J, K)$, $\mathcal{C}_{\text{ARJA}}$, and $\mathcal{C}_{\text{AR4JA}}(e)$ SC-LDPC-CC ensembles with period T as $\mathcal{T}(J, K, T)$, $\mathcal{T}_{\text{ARJA}}(T)$, and $\mathcal{T}_{\text{AR4JA}}(e, T)$, respectively. It is known that the average free distance of an ensemble of periodically time-varying protograph-based SC-LDPC-CCs with period T constructed as described in Definition 10 can be bounded below by the average minimum distance of the associated ensemble of TB-SC-LDPC-BCs derived from the base matrix $\mathbf{B}_{[0,L-1]}^{\text{tb}}$ with coupling length $L = T$ [18], [19]. Here, we show that the average free distance of this ensemble can also be bounded above by the average minimum distance of the terminated SC-LDPC-BC ensemble derived from the base matrix $\mathbf{B}_{[0,L-1]}$ with $L = T$.

Theorem 11: Consider a rate $R = 1 - b_c/b_v$ periodically time-varying SC-LDPC-CC ensemble with coupling width w, constraint length $\nu = M(w+1)b_v$, and period T derived from a convolutional protograph with base matrix $\mathbf{B}_{[-\infty,\infty]}$. Let $\overline{d}_{\min}^{(L)}$ be the average minimum distance of the associated SC-LDPC-BC ensemble with coupling length L and block length $n = MLb_v$ derived from the terminated convolutional protograph with base matrix $\mathbf{B}_{[0,L-1]}$. Then the ensemble average free distance $\overline{d}_{\text{free}}^{(T)}$ of the SC-LDPC-CC ensemble is bounded above by $\overline{d}_{\min}^{(L)}$ for coupling length $L = T$, i.e.,

$$\overline{d}_{\text{free}}^{(T)} \leq \overline{d}_{\min}^{(T)}. \tag{17}$$

Proof: There is a one-to-one relationship between members of the periodically time-varying SC-LDPC-CC ensemble and members of the associated SC-LDPC-BC ensemble with coupling length $L = T$. For any such pair of codes, every codeword $\mathbf{x}_{[0,MLb_v-1]} = [x_0\ x_1\ \cdots\ x_{MLb_v-1}]$ in the terminated code be viewed as a codeword $\mathbf{x}_{[0,\infty]} = [x_0\ x_1\ \cdots\ x_{LNb_v-1}\ 0\ 0\ \cdots]$ in the unterminated code. It follows that the free distance $d_{\text{free}}^{(T)}$ of the unterminated code can not be larger than the minimum distance $d_{\min}^{(T)}$ of the terminated code. The ensemble average result $\overline{d}_{\text{free}}^{(T)} \leq \overline{d}_{\min}^{(T)}$ then follows directly. □

Since there is no danger of ambiguity, we will henceforth drop the overline notation when discussing ensemble average distances.

For SC-LDPC-CCs, conventionally defined as the null space of a sparse parity-check matrix $\mathbf{H}_{[0,\infty]}$, it is natural to define the free distance growth rate with respect to the constraint length ν, i.e., as the ratio of the free distance d_{free} to the constraint length ν. By bounding $d_{\text{free}}^{(T)}$ using (17), we obtain an upper bound on the free distance growth rate as

$$\delta_{\text{free}}^{(T)} = \frac{d_{\text{free}}^{(T)}}{\nu} \leq \frac{\delta_{\min}^{(T)} T}{(w+1)}, \tag{18}$$

where $\delta_{\min}^{(T)} = d_{\min}^{(T)}/n = d_{\min}^{(T)}/(MTb_v)$ is the minimum distance growth rate of the SC-LDPC-BC ensemble with coupling length $L = T$ and base matrix $\mathbf{B}_{[0,T-1]}$.[10] Similarly, it was shown in [19] that

$$\delta_{\text{free}}^{(T)} \geq \frac{\check{\delta}_{\min}^{(T)} T}{(w+1)}, \tag{19}$$

where $\check{\delta}_{\min}^{(T)}$ is the minimum distance growth rate of the TB-SC-LDPC-BC ensemble with tail-biting coupling length $L = T$ and base matrix $\mathbf{B}_{[0,L-1]}^{\text{tb}}$.

Example 2 (cont.): As an example, consider the $\mathcal{C}(3, 6, L)$ SC-LDPC-BC ensembles. Using (18), we calculate the upper bound on the free distance growth rate of the periodically time-varying SC-LDPC-CC ensemble $\mathcal{T}(3, 6, T)$, with design rate $R = 1/2$, as $\delta_{\text{free}}^{(T)} \leq \delta_{\min}^{(T)} T/3$ for coupling lengths $L = T \geq 3$. Fig. 20 displays the minimum distance growth rates $\delta_{\min}^{(L)}$ of the $\mathcal{C}(3, 6, L)$ ensembles defined by $\mathbf{B}_{[0,L-1]}$

[10] The free distance growth rate $\delta_{\text{free}}^{(T)}$ that we bound from above using (18) is, by definition, an existence-type lower bound on the free distance typical of most members of the ensemble, i.e., with high probability a randomly chosen code from the ensemble has free distance at least as large as $\delta_{\text{free}}^{(T)} \nu$ as $\nu \to \infty$.

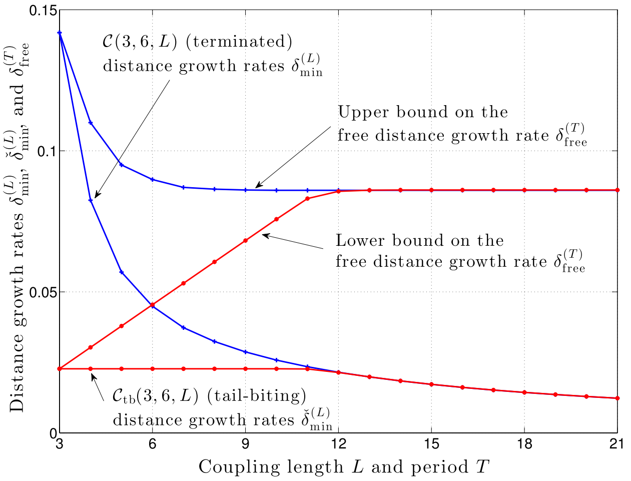


Fig. 20. Minimum distance growth rates of the $\mathcal{C}(3,6,L)$ (terminated) and $\mathcal{C}_{\text{tb}}(3,6,L)$ (tail-biting) SC-LDPC-BC ensembles with upper and lower bounds on the free distance growth rate of the associated periodically time-varying SC-LDPC-CC ensembles $\mathcal{T}(3,6,L)$.

for $L = 3,4,\ldots,21$ and the associated upper bounds on the free distance growth rate $\delta_{\text{free}}^{(T)} \le \delta_{\min}^{(T)} T/3$ for $L = T$. Also shown are the minimum distance growth rates $\check{\delta}_{\min}^{(L)}$ of the $\mathcal{C}_{\text{tb}}(3,6,L)$ TB-SC-LDPC-BC ensembles, defined by base matrix $\mathbf{B}_{[0,L-1]}^{\text{tb}}$ for $L = 3,4,\ldots,21$, and the associated lower bounds, calculated using (19), on the free distance growth rate $\delta_{\text{free}}^{(T)} \ge \check{\delta}_{\min}^{(T)} T/3$ for $L = T$.

We observe that the $\mathcal{C}_{\text{tb}}(3,6,L)$ minimum distance growth rates $\check{\delta}_{\min}^{(L)}$ remain constant for $L = 3,\ldots,11$ and then start to decrease as the coupling length L grows, tending to zero as L tends to infinity. Correspondingly, as L exceeds 11, the lower bound on $\delta_{\text{free}}^{(T)}$ levels off at $\delta_{\text{free}}^{(T)} \ge 0.086$. As discussed in Section III-A, the $\mathcal{C}(3,6,L)$ minimum distance growth rates $\delta_{\min}^{(L)}$ are large for small values of L (where the rate loss is larger) and decrease monotonically to zero as $L \to \infty$. Using (18) to obtain an upper bound on the free distance growth rate we observe that, for $T \ge 12$, the upper and lower bounds on $\delta_{\text{free}}^{(T)}$ coincide, indicating that, for these values of the period T, $\delta_{\text{free}}^{(T)} = 0.086$, significantly larger than the underlying (3, 6)-regular LDPC-BC ensemble minimum distance growth rate $\delta_{\min} = 0.023$. This leveling-off phenomenon occurs as a result of the fact that the minimum weight codeword in a typical member of the SC-LDPC-CC ensemble also appears as a codeword in a typical member of the SC-LDPC-BC ensemble once L exceeds 11. In addition, we note that, at the point where the bounds coincide, the minimum distance growth rates for both the terminated and tail-biting ensembles coincide. (Recall that the bounds diverge for smaller values of L since the $\mathcal{C}_{\text{tb}}(3,6,L)$ ensembles have rate 1/2 for all L, whereas the rate of the $\mathcal{C}(3,6,L)$ ensembles is a function L given by (9).) □

Numerically, it becomes problematic to evaluate $\delta_{min}^{(L)}$ and $\check{\delta}_{min}^{(L)}$ for large values of L, but the leveling-off effect noted in Fig. 20, which also occurs in all the other cases we have examined, strongly suggests that the free distance growth rate $\delta_{free}^{(T)}$ remains constant once T increases beyond a certain value. This is due to the fact that, for a fixed constraint length, further increases in the period cannot result in convolutional ensembles with larger free distances. We set $\delta_{free} \triangleq \max_T \delta_{free}^{(T)}$, and the leveling-off numerical results obtained for the $\mathcal{T}(3,6,T)$ ensembles suggests that the free distance growth rate δ_{free} of the associated (non-periodic) $\mathcal{C}(3,6)$ SC-LDPC-CC ensemble converges to 0.086. Lower bounds on free distance growth rates were calculated for a wide variety of (J,K)-regular and irregular protograph-based SC-LDPC-CC ensembles in [19] and, using the technique described here, we can form upper bounds on the free distance growth rates that coincide numerically with the lower bounds for sufficiently large T, resulting in exact free distance growth rates. Growth rates for a variety of $\mathcal{C}(J,K)$ and $\mathcal{C}_{\text{AR4JA}}(e)$ SC-LDPC-CC ensembles are plotted in Fig. 21, along with the minimum distance growth rates of the underlying LDPC-BC ensembles. Also shown are the Gilbert-Varshamov and Costello [78] lower bounds on the growth rates of general ensembles of random block and convolutional codes, respectively.[11] We observe that the convolutional free distance growth rates are significantly larger than the corresponding block minimum distance growth rates for each ensemble. This general technique can be used to find the free distance growth rate of any regular or irregular periodically time-varying protograph-based SC-LDPC-CC ensemble.

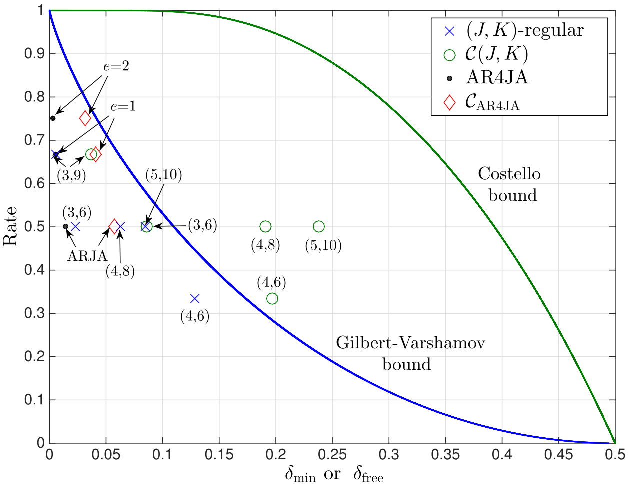


Fig. 21. Asymptotic free distance growth rates for some $\mathcal{C}(J,K)$ and $\mathcal{C}_{\text{AR4JA}}(e)$ SC-LDPC-CC ensembles.

The usefulness of the above result is twofold: on the one hand, the fact that the minimum distance growth rates of SC-LDPC-BC ensembles scale to a constant allows us to approximate the growth rates for large L (as noted earlier in Sections III-A and III-B), where otherwise it would not

[11] The constraint length ν that we define in this paper (see Definition 2) is often referred to as the *decoding constraint length*. In order to facilitate comparison to the Costello bound, rather than using (18), the free distance growth rates shown in Fig. 21 are normalized by the *encoding constraint length*. See [19] for further details.

be computationally feasible to do so; on the other hand, for large L, the free distance is arguably a more appropriate indicator of the strength of convolutional-like SC-LDPC-BC ensembles. In particular, when using convolutional decoding strategies, such as the sliding window decoder discussed above, it is intuitively clear that increasing L will not have a negative effect on performance. In this regard, it is natural that an appropriate distance measure for SC-LDPC-BC ensembles should be independent of L, like the free distance growth rate δ_{free} of the associated SC-LDPC-CC ensemble, rather than decaying with L, like the minimum distance growth rate $\delta_{\min}^{(L)}$, which tends to zero as $L \to \infty$. Numerous empirical studies and simulation results (see [10], [12], [15]) indeed have shown that the performance of SC-LDPC-BCs does not suffer as L is increased, indicating that a distance measure independent of L, such as δ_{free}, is a more appropriate measure of decoding performance than $\delta_{\min}$.

V. Concluding Remarks

In this paper, we have considered protograph-based spatially coupled LDPC codes. By coupling together a series of L disjoint, or uncoupled, block protographs into a single coupled chain by means of an edge spreading operation, we introduce memory into the code design and obtain the graph of a SC-LDPC-BC ensemble. By varying L, we obtain a flexible family of code ensembles with varying rates and code properties that can share the same encoding and decoding architecture for arbitrary L. For the $\mathcal{C}(J, K, L)$ ensembles, despite being almost regular, we demonstrated that the resulting codes combine the best features of optimized irregular and regular codes in one design: capacity approaching iterative BP decoding thresholds *and* linear growth of minimum distance with block length. In particular, we saw that, for sufficiently large L, the BP thresholds on both the BEC and AWGNC *saturate* to a value significantly larger than the BP decoding threshold and numerically indistinguishable from the MAP decoding threshold of the underlying LDPC-BC ensemble. Since all variable nodes have degree greater than two, asymptotically the error probability converges at least doubly exponentially with decoding iterations, and we obtain sequences of asymptotically good LDPC codes with fast convergence rates and BP thresholds close to the Shannon limit. The gap to the Shannon limit decreases as the density of the graph increases, opening up a new way to construct capacity achieving codes on MBS channels with low-complexity BP decoding.

The key to the excellent threshold performance of SC-LDPC-BC ensembles is a slight structured irregularity introduced to the graph at the boundaries. As L increases, we obtain a family of codes with increasing design rates and a trade-off between capacity approaching iterative decoding thresholds and declining minimum distance growth rates. However, we saw that the growth rates, while declining with L, converge to a bound on the free distance growth rate of the closely related SC-LDPC-CC ensemble, which is independent of L and significantly larger than the minimum distance growth rate of the underlying LDPC-BC ensemble, indicating that, particularly in conjunction with convolutional decoding strategies such as a sliding window decoder, an appropriate distance measure for SC-LDPC-BC ensembles should also be independent of L. Finally, we showed that the threshold saturation effect obtained by spatial coupling a sequence of disjoint graphs is a general phenomenon and can be applied to both regular and irregular LDPC-BC ensembles. Moreover, carefully designing the edge spreading, increasing the density of the component graphs, and coupling optimized irregular graphs can further improve performance in terms of both asymptotic minimum distance growth rate and iterative BP decoding threshold.

References

[1] R. M. Tanner, “A recursive approach to low complexity codes,” *IEEE Trans. Inf. Theory*, vol. 27, no. 5, pp. 533–547, Sep. 1981.

[2] R. G. Gallager, “Low-density parity-check codes,” *IRE Trans. Inf. Theory*, vol. 8, no. 1, pp. 21–28, Jan. 1962.

[3] M. G. Luby, M. Mitzenmacher, M. A. Shokrollahi, and D. A. Spielman, “Improved low-density parity-check codes using irregular graphs,” *IEEE Trans. Inf. Theory*, vol. 47, no. 2, pp. 585–598, Feb. 2001.

[4] T. J. Richardson and R. L. Urbanke, “The capacity of low-density parity-check codes under message-passing decoding,” *IEEE Trans. Inf. Theory*, vol. 47, no. 2, pp. 599–618, Feb. 2001.

[5] S.-Y. Chung, G. D. Forney, Jr., T. J. Richardson, and R. Urbanke, “On the design of low-density parity-check codes within 0.0045 dB of the Shannon limit,” *IEEE Commun. Lett.*, vol. 5, no. 2, pp. 58–60, Feb. 2001.

[6] P. Oswald and M. A. Shokrollahi, “Capacity-achieving sequences for the erasure channel,” *IEEE Trans. Inf. Theory*, vol. 48, no. 12, pp. 3017–3028, Dec. 2002.

[7] A. Jiménez Felström and K. S. Zigangirov, “Time-varying periodic convolutional codes with low-density parity-check matrix,” *IEEE Trans. Inf. Theory*, vol. 45, no. 6, pp. 2181–2191, Sep. 1999.

[8] D. J. Costello, Jr., A. E. Pusane, S. Bates, and K. S. Zigangirov, “A comparison between LDPC block and convolutional codes,” in *Proc. Inf. Theory Appl. Workshop*, San Diego, CA, USA, Feb. 2006, pp. 1–10.

[9] D. J. Costello, Jr., A. E. Pusane, C. R. Jones, and D. Divsalar, “A comparison of ARA- and protograph-based LDPC block and convolutional codes,” in *Proc. Inf. Theory Appl. Workshop*, San Diego, CA, USA, Feb. 2007, pp. 111–119.

[10] R. M. Tanner, D. Sridhara, A. Sridharan, T. E. Fuja, and D. J. Costello, Jr., “LDPC block and convolutional codes based on circulant matrices,” *IEEE Trans. Inf. Theory*, vol. 50, no. 12, pp. 2966–2984, Dec. 2004.

[11] A. E. Pusane, A. J. Feltström, A. Sridharan, M. Lentmaier, K. S. Zigangirov, and D. J. Costello, Jr., “Implementation aspects of LDPC convolutional codes,” *IEEE Trans. Commun.*, vol. 56, no. 7, pp. 1060–1069, Jul. 2008.

[12] A. E. Pusane, R. Smarandache, P. O. Vontobel, and D. J. Costello, Jr., “Deriving good LDPC convolutional codes from LDPC block codes,” *IEEE Trans. Inf. Theory*, vol. 57, no. 2, pp. 835–857, Feb. 2011.

[13] A. Sridharan, M. Lentmaier, D. J. Costello, Jr., and K. S. Zigangirov, “Convergence analysis for a class of LDPC convolutional codes on the erasure channel,” in *Proc. 42nd Allerton Conf. Commun., Control, Comput.*, Monticello, IL, USA, Sep. 2004, pp. 953–962.

[14] M. Lentmaier, A. Sridharan, K. S. Zigangirov, and D. J. Costello, Jr., “Terminated LDPC convolutional codes with thresholds close to capacity,” in *Proc. IEEE Int. Symp. Inf. Theory*, Adelaide, SA, Australia, Sep. 2005, pp. 1372–1376.

[15] M. Lentmaier, A. Sridharan, D. J. Costello, Jr., and K. S. Zigangirov, “Iterative decoding threshold analysis for LDPC convolutional codes,” *IEEE Trans. Inf. Theory*, vol. 56, no. 10, pp. 5274–5289, Oct. 2010.

[16] M. Lentmaier, D. V. Truhachev, and K. S. Zigangirov, “To the theory of low-density convolutional codes. II,” *Problems Inf. Transmiss.*, vol. 37, no. 4, pp. 288–306, Aug. 2001.

[17] A. Sridharan, D. Truhachev, M. Lentmaier, D. J. Costello, Jr., and K. S. Zigangirov, “Distance bounds for an ensemble of LDPC convolutional codes,” *IEEE Trans. Inf. Theory*, vol. 53, no. 12, pp. 4537–4555, Dec. 2007.

[18] D. Truhachev, K. S. Zigangirov, and D. J. Costello, Jr., “Distance bounds for periodically time-varying and tail-biting LDPC convolutional codes,” *IEEE Trans. Inf. Theory*, vol. 56, no. 9, pp. 4301–4308, Sep. 2010.

[19] D. G. M. Mitchell, A. E. Pusane, and D. J. Costello, Jr., "Minimum distance and trapping set analysis of protograph-based LDPC convolutional codes," *IEEE Trans. Inf. Theory*, vol. 59, no. 1, pp. 254–281, Jan. 2013.

[20] R. Smarandache, A. E. Pusane, P. O. Vontobel, and D. J. Costello, Jr., "Pseudocodeword performance analysis for LDPC convolutional codes," *IEEE Trans. Inf. Theory*, vol. 55, no. 6, pp. 2577–2598, Jun. 2009.

[21] S. Kudekar, T. J. Richardson, and R. L. 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.

[22] 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.

[23] S. Kudekar and H. D. Pfister, "The effect of spatial coupling on compressive sensing," in *Proc. Allerton Conf. Commun., Control, Comput.*, Monticello, IL, USA, Sep./Oct. 2010, pp. 347–353.

[24] D. L. Donoho, A. Javanmard, and A. Montanari, "Information-theoretically optimal compressed sensing via spatial coupling and approximate message passing," *IEEE Trans. Inf. Theory*, vol. 59, no. 11, pp. 7434–7464, Nov. 2013.

[25] Z. Si, R. Thobaben, and M. Skoglund, "Bilayer LDPC convolutional codes for half-duplex relay channels," in *Proc. IEEE Int. Symp. Inf. Theory*, St. Petersburg, Russia, Jul./Aug. 2011, pp. 1464–1468.

[26] H. Uchikawa, K. Kasai, and K. Sakaniwa, "Spatially coupled LDPC codes for decode-and-forward in erasure relay channel," in *Proc. IEEE Int. Symp. Inf. Theory*, St. Petersburg, Russia, Aug. 2011, pp. 1474–1478.

[27] S. Schwandter, A. Graell i Amat, and G. Matz, "Spatially coupled LDPC codes for two-user decode-and-forward relaying," in *Proc. Int. Symp. Turbo Codes Iterative Inf. Process.*, Gothenburg, Sweden, Aug. 2012, pp. 46–50.

[28] L. Wei, D. J. Costello, Jr., and T. E. Fuja, "Coded cooperation using rate-compatible spatially-coupled codes," in *Proc. IEEE Int. Symp. Inf. Theory*, Istanbul, Turkey, Jul. 2013, pp. 1869–1873.

[29] V. Rathi, R. Urbanke, M. Andersson, and M. Skoglund, "Rate-equivocation optimal spatially coupled LDPC codes for the BEC wiretap channel," in *Proc. IEEE Int. Symp. Inf. Theory*, St. Petersburg, Russia, Aug. 2011, pp. 2393–2397.

[30] S. Kudekar and K. Kasai, "Spatially coupled codes over the multiple access channel," in *Proc. IEEE Int. Symp. Inf. Theory*, St. Petersburg, Russia, Jul./Aug. 2011, pp. 2816–2820.

[31] A. Yedla, P. S. Nguyen, H. D. Pfister, and K. R. Narayanan, "Universal codes for the Gaussian MAC via spatial coupling," in *Proc. 49th Annu. Allerton Conf. Commun., Control, Comput.*, Monticello, IL, USA, Sep. 2011, pp. 1801–1808.

[32] D. Truhachev, "Achieving AWGN multiple access channel capacity with spatial graph coupling," *IEEE Commun. Lett.*, vol. 16, no. 5, pp. 585–588, May 2012.

[33] C. Schlegel and D. Truhachev, "Multiple access demodulation in the lifted signal graph with spatial coupling," *IEEE Trans. Inf. Theory*, vol. 59, no. 4, pp. 2459–2470, Apr. 2013.

[34] Z. Si, R. Thobaben, M. Skoglund, and T. J. Oechtering, "Bidirectional broadcasting by using multi-edge type LDPC convolutional codes," in *Proc. Int. Symp. Turbo Codes Iterative Inf. Process.*, Gothenburg, Sweden, Aug. 2012, pp. 91–95.

[35] S. Kudekar and K. Kasai, "Threshold saturation on channels with memory via spatial coupling," in *Proc. IEEE Int. Symp. Inf. Theory*, St. Petersburg, Russia, Jul./Aug. 2011, pp. 2562–2566.

[36] P. S. Nguyen, A. Yedla, H. D. Pfister, and K. R. Narayanan, "Threshold saturation of spatially-coupled codes on intersymbol-interference channels," in *Proc. IEEE Int. Conf. Commun.*, Ottawa, ON, Canada, Jun. 2012, pp. 2181–2186.

[37] K. Takeuchi, T. Tanaka, and T. Kawabata, "Improvement of BP-based CDMA multiuser detection by spatial coupling," in *Proc. IEEE Int. Symp. Inf. Theory*, St. Petersburg, Russia, Jul./Aug. 2011, pp. 1489–1493.

[38] G. Liva, E. Paolini, M. Lentmaier, and M. Chiani, "Spatially-coupled random access on graphs," in *Proc. IEEE Int. Symp. Inf. Theory*, Boston, MA, USA, Jul. 2012, pp. 478–482.

[39] V. Aref, N. Macris, R. Urbanke, and M. Vuffray, "Lossy source coding via spatially coupled LDGM ensembles," in *Proc. IEEE Int. Symp. Inf. Theory*, Boston, MA, USA, Jul. 2012, pp. 373–377.

[40] M. Hagiwara, K. Kasai, H. Imai, and K. Sakaniwa, "Spatially coupled quasi-cyclic quantum LDPC codes," in *Proc. IEEE Int. Symp. Inf. Theory*, St. Petersburg, Russia, Aug. 2011, pp. 638–642.

[41] I. Andriyanova, D. Maurice, and J.-P. Tillich, "Spatially coupled quantum LDPC codes," in *Proc. IEEE Inf. Theory Workshop*, Lausanne, Switzerland, Sep. 2012, pp. 327–331.

[42] S. H. Hassani, N. Macris, and R. Urbanke, "Coupled graphical models and their thresholds," in *Proc. IEEE Inf. Theory Workshop*, Dublin, Ireland, Aug./Sep. 2010, pp. 1–5.

[43] P. M. Olmos and R. Urbanke, "Scaling behavior of convolutional LDPC ensembles over the BEC," in *Proc. IEEE Int. Symp. Inf. Theory*, St. Petersburg, Russia, Aug. 2011, pp. 1816–1820.

[44] P. M. Olmos and R. Urbanke, "A closed-form scaling law for convolutional LDPC codes over the BEC," in *Proc. IEEE Inf. Theory Workshop*, Sevilla, Spain, Sep. 2013, pp. 1–5.

[45] A. Julé and I. Andriyanova, "Performance bounds for spatially-coupled LDPC codes over the block erasure channel," in *Proc. IEEE Int. Symp. Inf. Theory*, Istanbul, Turkey, Jul. 2013, pp. 1879–1883.

[46] J. Thorpe, "Low-density parity-check (LDPC) codes constructed from protographs," Jet Propuls. Lab., Pasadena, CA, USA, Tech. Rep. 42-154, Aug. 2003.

[47] T. Richardson and R. Urbanke, *Modern Coding Theory*. Cambridge, U.K.: Cambridge Univ. Press, 2008.

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

[49] S. Lin and D. J. Costello, Jr., *Error Control Coding: Fundamentals and Applications*, 2nd ed. Englewood Cliffs, NJ, USA: Prentice-Hall, 2004.

[50] *Low Density Parity Check Codes for Use in Near-Earth and Deep Space Applications*, CCSDS document 131.1-O-2, Sep. 2007. [Online]. Available: http://public.ccsds.org/publications/SilverBooks.aspx

[51] M. Lentmaier, M. M. Prenda, and G. P. Fettweis, "Efficient message passing scheduling for terminated LDPC convolutional codes," in *Proc. IEEE Int. Symp. Inf. Theory*, St. Petersburg, Russia, Jul./Aug. 2011, pp. 1826–1830.

[52] A. R. Iyengar, M. Papaleo, P. H. Siegel, J. K. Wolf, A. Vanelli-Coralli, and G. E. Corazza, "Windowed decoding of protograph-based LDPC convolutional codes over erasure channels," *IEEE Trans. Inf. Theory*, vol. 58, no. 4, pp. 2303–2320, Apr. 2012.

[53] G. Solomon and H. C. A. Tilborg, "A connection between block and convolutional codes," *SIAM J. Appl. Math.*, vol. 37, no. 2, pp. 358–369, Oct. 1979.

[54] H. Ma and J. K. Wolf, "On tail biting convolutional codes," *IEEE Trans. Commun.*, vol. 34, no. 2, pp. 104–111, Feb. 1986.

[55] R. M. Tanner, "Convolutional codes from quasi-cyclic codes: A link between the theories of block and convolutional codes," Baskin Center Comput. Eng. Inf. Sci., Univ. California, Santa Cruz, Santa Cruz, CA, USA, Tech. Rep. UCSC-CRL-87-21, 1987.

[56] S. Myung, K. Yang, and J. Kim, "Quasi-cyclic LDPC codes for fast encoding," *IEEE Trans. Inf. Theory*, vol. 51, no. 8, pp. 2894–2901, Aug. 2005.

[57] Z. Li, L. Chen, L. Zeng, S. Lin, and W. H. Fong, "Efficient encoding of quasi-cyclic low-density parity-check codes," *IEEE Trans. Commun.*, vol. 54, no. 1, pp. 71–81, Jan. 2006.

[58] Z. Wang and Z. Cui, "Low-complexity high-speed decoder design for quasi-cyclic LDPC codes," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 15, no. 1, pp. 104–114, Jan. 2007.

[59] Y. Dai, Z. Yan, and N. Chen, "Memory-efficient and high-throughput decoding of quasi-cyclic LDPC codes," *IEEE Trans. Commun.*, vol. 57, no. 4, pp. 879–883, Apr. 2009.

[60] Y. Kou, S. Lin, and M. P. C. Fossorier, "Low-density parity-check codes based on finite geometries: A rediscovery and new results," *IEEE Trans. Inf. Theory*, vol. 47, no. 7, pp. 2711–2736, Nov. 2001.

[61] L. Chen, J. Xu, I. Djurdjevic, and S. Lin, "Near-Shannon-limit quasi-cyclic low-density parity-check codes," *IEEE Trans. Commun.*, vol. 52, no. 7, pp. 1038–1042, Jul. 2004.

[62] J. Kang, Q. Huang, L. Zhang, B. Zhou, and S. Lin, "Quasi-cyclic LDPC codes: An algebraic construction," *IEEE Trans. Commun.*, vol. 58, no. 5, pp. 1383–1396, May 2010.

[63] D. J. C. MacKay and M. C. Davey, "Evaluation of Gallager codes for short block length and high rate applications," in *IMA Volumes in Mathematics and Its Applications: Codes, Systems, and Graphical Models*, vol. 123. New York, NY, USA: Springer-Verlag, 2001, pp. 113–130.

[64] M. P. C. Fossorier, "Quasi-cyclic low-density parity-check codes from circulant permutation matrices," *IEEE Trans. Inf. Theory*, vol. 50, no. 8, pp. 1788–1793, Aug. 2004.

[65] K. Huang, D. G. M. Mitchell, L. Wei, X. Ma, and D. J. Costello, Jr., "Performance comparison of LDPC block and spatially coupled codes over GF(q)," *IEEE Trans. Commun.*, vol. 63, no. 3, pp. 592–604, Mar. 2015.

[66] N. ul Hassan, M. Lentmaier, and G. P. Fettweis, "Comparison of LDPC block and LDPC convolutional codes based on their decoding latency," in *Proc. Int. Symp. Turbo Codes Iterative Inf. Process.*, Gothenburg, Sweden, Aug. 2012, pp. 225–229.

[67] N. ul Hassan, A. E. Pusane, M. Lentmaier, G. P. Fettweis, and D. J. Costello, Jr., "Reduced complexity window decoding schedules for coupled LDPC codes," in *Proc. IEEE Inf. Theory Workshop*, Lausanne, Switzerland, Sep. 2012, pp. 20–24.

[68] S. L. Fogal, R. McEliece, and J. Thorpe, "Enumerators for protograph ensembles of LDPC codes," in *Proc. IEEE Int. Symp. Inf. Theory*, Adelaide, SA, Australia, Sep. 2005, pp. 2156–2160.

[69] E. N. Gilbert, "A comparison of signalling alphabets," *Bell Syst. Tech. J.*, vol. 31, no. 3, pp. 504–522, May 1952.

[70] R. R. Varshamov, "Estimate of the number of signals in error correcting codes," *Doklady Akademii Nauk SSSR*, vol. 117, no. 5, pp. 739–741, 1957.

[71] P. Grover, K. Woyach, and A. Sahai, "Towards a communication-theoretic understanding of system-level power consumption," *IEEE J. Sel. Areas Commun.*, vol. 29, no. 8, pp. 1744–1755, Sep. 2011.

[72] M. Lentmaier, D. V. Truhachev, K. S. Zigangirov, and D. J. Costello, Jr., "An analysis of the block error probability performance of iterative decoding," *IEEE Trans. Inf. Theory*, vol. 51, no. 11, pp. 3834–3855, Nov. 2005.

[73] A. K. Pradhan, A. Subramanian, and A. Thangaraj, "Deterministic constructions for large girth protograph LDPC codes," in *Proc. IEEE Int. Symp. Inf. Theory*, Istanbul, Turkey, Jul. 2013, pp. 1680–1684.

[74] G. Miller and D. Burshtein, "Bounds on the maximum-likelihood decoding error probability of low-density parity-check codes," *IEEE Trans. Inf. Theory*, vol. 47, no. 7, pp. 2696–2710, Nov. 2001.

[75] M. Lentmaier, D. G. M. Mitchell, G. P. Fettweis, and D. J. Costello, Jr., "Asymptotically regular LDPC codes with linear distance growth and thresholds close to capacity," in *Proc. Inf. Theory Appl. Workshop*, San Diego, CA, USA, Jan./Feb. 2010, pp. 1–8.

[76] D. G. M. Mitchell, R. Smarandache, M. Lentmaier, and D. J. Costello, Jr., "Quasi-cyclic asymptotically regular LDPC codes," in *Proc. IEEE Inf. Theory Workshop*, Dublin, Ireland, Aug. 2010, pp. 1–5.

[77] S.-Y. Chung, "On the construction of some capacity-approaching coding schemes," Ph.D. dissertation, Dept. Elect. Eng. Comput. Sci., Massachusetts Inst. Technol., Cambridge, MA, USA, Sep. 2000.

[78] D. J. Costello, Jr., "Free distance bounds for convolutional codes," *IEEE Trans. Inf. Theory*, vol. 20, no. 3, pp. 356–365, May 1974.

David G. M. Mitchell (S'08–M'10) received the Ph.D. degree in Electrical Engineering from the University of Edinburgh, United Kingdom, in 2009. He is currently an Assistant Professor in the Klipsch School of Electrical and Computer Engineering at the New Mexico State University, USA. He previously held Visiting Assistant Professor and Post-Doctoral Research Associate positions in the Department of Electrical Engineering at the University of Notre Dame, USA. His research interests are in the area of digital communications, with emphasis on error control coding and information theory.

Michael Lentmaier (S'98–M'03–SM'11) received the Dipl.-Ing. degree in Electrical Engineering from University of Ulm, Germany in 1998, and the Ph.D. degree in Telecommunication Theory from Lund University, Sweden in 2003. He then worked as a Post-Doctoral Research Associate at University of Notre Dame, Indiana and at University of Ulm. From 2005 to 2007 he was with the Institute of Communications and Navigation of the German Aerospace Center (DLR) in Oberpfaffenhofen, where he worked on signal processing techniques in satellite navigation receivers. From 2008 to 2012 he was a senior researcher and lecturer at the Vodafone Chair Mobile Communications Systems at TU Dresden, where he was heading the Algorithms and Coding research group. Since January 2013 he is an Associate Professor at the Department of Electrical and Information Technology at Lund University. His research interests include design and analysis of coding systems, graph based iterative algorithms and Bayesian methods applied to decoding, detection and estimation in communication systems. He is a senior member of the IEEE and served as an editor for IEEE COMMUNICATIONS LETTERS from 2010 to 2013 and IEEE TRANSACTIONS ON COMMUNICATIONS since 2014. He was awarded the Communications Society & Information Theory Society Joint Paper Award (2012) for the paper "Iterative Decoding Threshold Analysis for LDPC Convolutional Codes."

Daniel J. Costello, Jr. (S'62–M'69–SM'78–F'85–LF'08) received his Ph.D. in Electrical Engineering from the University of Notre Dame in 1969. Since 1985, he has been a Professor of Electrical Engineering at Notre Dame and from 1989 to 1998 served as Chair of the Department. In 2000, he was named the Leonard Bettex Professor of Electrical Engineering.

Dr. Costello has been a member of IEEE since 1969 and was elected Fellow in 1985. In 2000, the IEEE Information Theory Society selected him as a recipient of a Third-Millennium Medal, he was a co-recipient of the 2009 IEEE Donald G. Fink Prize Paper Award and the 2012 ComSoc & Information Theory Society Joint Paper Award, and he received the 2013 IEEE Information Theory Society Aaron D. Wyner Distinguished Service Award and the 2015 IEEE Leon J. Kirchner Graduate Teaching Award.

Dr. Costello's research interests are in the area of digital communications, with emphasis on error control coding and coded modulation. He has numerous technical publications in his field, and in 1983 he co-authored a textbook entitled "*Error Control Coding: Fundamentals and Applications*," the 2nd edition of which was published in 2004.