Approaching Capacity with Asymptotically Regular LDPC Codes

Lentmaier, Michael; Fettweis, Gerhard; Zigangirov, Kamil; Costello Jr., Daniel J.

Published in:
2009 Information Theory and Applications Workshop

DOI:
10.1109/ITA.2009.5044941

2009

Approaching Capacity with Asymptotically Regular LDPC Codes

Michael Lentmaier and Gerhard P. Fettweis
Vodafone Chair Mobile Communications Systems
Dresden University of Technology
01062 Dresden, Germany
Email: {michael.lentmaier, fettweis}@ifn.et.tu-dresden.de

Kamil Sh. Zigangirov and Daniel J. Costello, Jr.
Department of Electrical Engineering
University of Notre Dame
Notre Dame, IN 46556, USA
Email: {Zigangirov.1, Costello.2}@nd.edu

***Abstract*—We present a family of protograph based LDPC codes that can be derived from permutation matrix based regular (J, K) LDPC convolutional codes by termination. In the terminated protograph, all variable nodes still have degree J but some check nodes at the start and end of the protograph have degrees smaller than K. Since the fraction of these stronger nodes vanishes as the termination length L increases, we call the codes *asymptotically regular*. The density evolution thresholds of these protographs are better than those of regular (J, K) block codes. Interestingly, this threshold improvement gets stronger with increasing node degrees (at a fixed rate) and it does not decay as L increases. Terminated convolutional protographs can also be derived from standard irregular protographs and may exhibit a significant threshold improvement.**

I. Introduction

The performance of a belief propagation (BP) decoder for LDPC codes [1] is strongly influenced by the degrees of the different variable nodes and check nodes in the considered Tanner graph representation [2]. The original ensembles introduced by Gallager in [1] consisted of regular (J, K) LDPC codes with fixed variable node degree J and check degree K. One shortcoming of such regular ensembles is that a small K, which leads to short parity-check equations and improves the decoder performance, also implies a small J for a given rate of the code. For this reason, irregular code ensembles [3] [4] with a variety of different node degrees are usually used in practice. In these ensembles, the degrees of variable nodes and check nodes are considered as random variables that are characterized by their degree distributions $\lambda(x)$ and $\rho(x)$, respectively. Each coefficient in the polynomials $\lambda(x)$ and $\rho(x)$ corresponds to the fraction of edges connected to nodes of a certain degree. Gallager's regular (J, K) LDPC code ensembles correspond to the special case $\lambda(x) = x^{J-1}$ and $\rho(x) = x^{K-1}$.

For the binary erasure channel (BEC), a density evolution analysis of the BP decoder can be performed explicitly by means of the equation

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

where ε denotes the erasure probability of the channel and $p^{(i)}$ the probability that a variable to check node message in decoding iteration i corresponds to an erasure, averaged over all codes of the ensemble. Due to this averaging, the message probabilities are equal for all edges in the graph. The density evolution threshold of an ensemble, defined as the maximal value of the channel parameter ε for which $p^{(i)}$ converges to zero as i tends to infinity, directly follows from (1). Equation (1) is also the key to the design of degree distribution pairs (λ, ρ) for *capacity achieving sequences* of codes with a vanishing gap between the threshold and the Shannon limit $\varepsilon_{\mathrm{sh}} = 1 - R$ [5]. Check-concentrated or even check-regular ensembles are known to provide a good trade-off between complexity (measured by the average node degrees) and gap to capacity.

A double exponential decrease of the decoding erasure probability with iterations implies that the probability of *erased frames* also converges to zero [6]. The lower bounds in [7] on the decoding complexity of general message passing decoders, obtained using sphere-packing arguments, also predict a double exponential reduction of the error probability with the number of iterations. A Taylor expansion of (1) reveals that the message probability $p^{(i)}$ converges to zero at least doubly exponentially with i if all nodes have a variable node degree of at least three. An analysis by means of the messages' Bhattacharyya parameter shows that this is also true for general binary-input output-symmetric memory-less channels [6]. For generalized LDPC codes, where the parity-check equations are replaced by stronger subcodes with minimum distance greater than two, it can be shown that a minimal variable node degree of two is sufficient. Unfortunately, unstructured irregular LDPC ensembles with thresholds close to capacity exhibit a non-vanishing fraction of degree two variable nodes.

LDPC code ensembles with a certain predefined structure can be constructed by means of protographs [8]. It has been observed that protograph ensembles often have better thresholds than unstructured irregular random ensembles with the corresponding degree distributions. Even codes with minimal variable node degree three may provide a good trade-off between distance and threshold [9]. Some codes that contain degree two variable nodes can also have a linear asymptotic minimum distance growth [10].

In this paper, we derive asymptotically regular protographs from convolutional protographs by termination. These protographs, which are described in Section III, have a constant variable node degree of at least three and at the same time

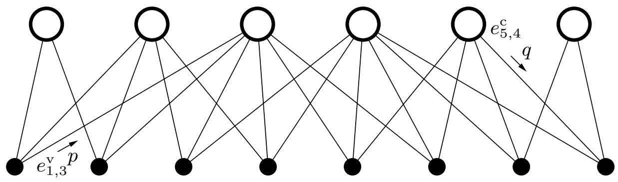

Fig. 1. Example of a protograph with $M_P = 6$ check nodes and $N_P = 8$ variable nodes .

a threshold close to the Shannon limit. In Section IV we generalize this approach to arbitrary irregular protographs. Using an AR4JA code [10] as an example, we demonstrate that a significant threshold improvement is also possible in the irregular case. We begin in Section II with a description of the density evolution equations for protograph based ensembles on the BEC.

II. Density Evolution for Protograph Ensembles

A protograph [8] is a bipartite graph consisting of a set of variable nodes V_n with degree J_n, $n = 1, \ldots, N_P$, a set of check nodes C_m with degree K_m, $m = 1, \ldots, M_P$, and a set $\mathcal{E}$ of edges that connect them. An example of a protograph is shown in Fig. 1. The edges connected to a variable node V_n or a check node C_m are labeled by $e^{\mathrm{v}}_{n,j}$ or $e^{\mathrm{c}}_{m,k}$, respectively, where $j = 1, \ldots, J_n$ and $k = 1, \ldots, K_m$. It follows that the j-th edge of V_n is connected to the k-th edge of C_m if $e^{\mathrm{v}}_{n,j} = e^{\mathrm{c}}_{m,k}$. A protograph can be represented by means of an $M_P \times N_P$ bi-adjacency matrix $\mathbf{B}$, which is called the *base matrix* of the protograph. The entry in row m and column n of $\mathbf{B}$ is equal to the number of edges that connect nodes C_m and V_n. Note that the base matrix representation allows multiple edges between a pair of nodes.

While a protograph is formally equivalent to a Tanner graph [2], it actually represents a family of codes of different lengths whose individual Tanner graphs are obtained from the protograph by a copy-and-permute operation [8]. Then a size M permutation matrix is associated with each edge in the protograph and each node is replicated M times, resulting in a *derived graph* that defines a code of length MN_P. By this procedure, the edges are permuted among these replica in such a way that the structure of the original graph is preserved. As a consequence, a density evolution analysis for the resulting codes can be performed within the protograph.

We assume that belief propagation is used for decoding, after transmission over a BEC with erasure probability ε. In every iteration, first all check nodes and then all variable nodes are updated. The messages that are passed between the nodes represent either an erasure or the correct symbol values 0 or 1. Let $q^{(i)}(e^{\mathrm{c}}_{m,k})$ denote the probability that the check to variable node message which is sent along edge $e^{\mathrm{c}}_{m,k}$ in decoding iteration i is an erasure. This is the case if at least one of the incoming messages from the other neighboring nodes is erased, i.e.,

$$q^{(i)}(e^{\mathrm{c}}_{m,k}) = 1 - \prod_{k' \neq k} \left(1 - p^{(i-1)}(e^{\mathrm{c}}_{m,k'})\right) \ , \tag{2}$$

where $p^{(i-1)}(e^{\mathrm{c}}_{m,k'})$, $k, k' \in \{1, \ldots, K_m\}$, denote the probabilities that the incoming messages computed in the previous iteration are erasures.

The variable to check node message sent along edge $e^{\mathrm{v}}_{n,j}$ is an erasure if all incoming messages from the channel and from the other neighboring check nodes are erasures. Thus we have

$$p^{(i)}(e^{\mathrm{v}}_{n,j}) = \varepsilon \prod_{j' \neq j} q^{(i)}(e^{\mathrm{v}}_{n,j'}) \ , \tag{3}$$

where $j, j' \in \{1, \ldots, J_n\}$.

III. Terminated Regular LDPC Convolutional Codes

A. LDPC convolutional codes

A rate $R = b/c$ time-varying binary LDPC convolutional (LDPCC) code [11] can be defined as the set of infinite sequences $\mathbf{v} = [\ldots, \mathbf{v}_0, \mathbf{v}_1, \ldots, \mathbf{v}_t, \ldots]$ satisfying the equation $\mathbf{v}\mathbf{H}^{\mathrm{T}} = \mathbf{0}$, where $\mathbf{v}_t = [v_t^{(1)}, \ldots, v_t^{(c)}]$, $v_t^{(\cdot)} \in \mathrm{GF}(2)$, and

$$\mathbf{H}^{\mathrm{T}} = \begin{bmatrix} \ddots & & \ddots & & \\ \mathbf{H}_0^{\mathrm{T}}(0) & \ldots & \mathbf{H}_{m_{\mathrm{s}}}^{\mathrm{T}}(m_{\mathrm{s}}) & & \\ & \ddots & & \ddots & \\ & & \mathbf{H}_0^{\mathrm{T}}(t) & \ldots & \mathbf{H}_{m_{\mathrm{s}}}^{\mathrm{T}}(t+m_{\mathrm{s}}) \\ & & \ddots & & \ddots \end{bmatrix} \tag{4}$$

is an infinite transposed parity-check matrix, also called a *syndrome former*. The elements $\mathbf{H}_i^{\mathrm{T}}(t)$, $i = 0, 1, \ldots, m_{\mathrm{s}}$, in (4) are binary $c \times (c-b)$ submatrices

$$\mathbf{H}_i^{\mathrm{T}}(t) = \begin{bmatrix} h_i^{(1,1)}(t) & \cdots & h_i^{(1,c-b)}(t) \\ \vdots & & \vdots \\ h_i^{(c,1)}(t) & \cdots & h_i^{(c,c-b)}(t) \end{bmatrix} \ , \tag{5}$$

where $\mathbf{H}_{m_{\mathrm{s}}}^{\mathrm{T}}(t) \neq 0$ for at least one $t \in \mathbb{Z}$ and $\mathbf{H}_0^{\mathrm{T}}(t)$ has full rank for all t. We call m_{s} the syndrome former memory and $\nu_{\mathrm{s}} = (m_{\mathrm{s}} + 1) \cdot c$ the associated decoding constraint length. These parameters determine the span of the nonzero diagonal region of $\mathbf{H}^{\mathrm{T}}$. Sparsity of the syndrome former is ensured by demanding that the Hamming weights of its columns are much smaller than ν_{s}. The code is said to be regular if its syndrome former $\mathbf{H}^{\mathrm{T}}$ has exactly J ones in every row and K ones in every column. The other entries are zeros. We will refer to a code with these properties as a (J, K) LDPCC code.

B. LDPCC Codes from Fully Connected Protographs

The convolutional counterparts of the (J, K) LDPC block code ensembles in [12], with syndrome formers $\mathbf{H}^{\mathrm{T}}$ composed of blocks of size M permutation matrices, have been considered in [13]. Analogously to block codes, these codes can be represented by protographs. 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 $K = aK'$ and $\gcd(J', K') = 1$. The ensemble of rate $R = Mb/(Mc) = 1 - J'/K'$ convolutional codes considered in [13] can be

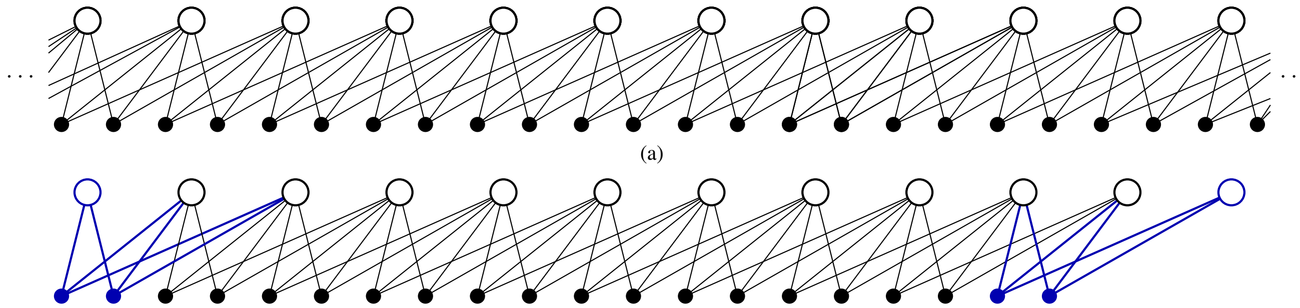

Fig. 2. (a) The convolutional protograph of a regular (3,6) LDPCC code and (b) the protograph for termination after $L = 10$ time instants. If the symbols at $t = 1$ and $t = 10$ are perfectly known, the edges connected to these nodes can be removed and the protograph becomes equivalent to that for $L = 8$.

described by infinite convolutional protograph base matrices

$$\mathbf{B}_{[-\infty,\infty]} = \begin{bmatrix} \ddots & & \ddots & & \\ \mathbf{B}_{m_s} & \dots & \mathbf{B}_0 & & \\ & \ddots & & \ddots & \\ & & \mathbf{B}_{m_s} & \dots & \mathbf{B}_0 \\ & & & \ddots & & \ddots \end{bmatrix} , \quad (6)$$

where $m_s = a-1$ and $\mathbf{B}_i$, $i = 0, \dots, m_s$, are $K' \times J'$ identical component base matrices with all entries equal to one. It follows immediately that for $m_s = 0$ we obtain a sequence of disconnected block code protographs with base matrix $\mathbf{B}_0$. From this point of view, block codes can be regarded as a special case of convolutional codes. On the other hand, if we start a convolutional code at time $t = 1$ and terminate it after L time instants, we obtain an L times larger block code. We denote the base matrix of such a block code by $\mathbf{B}_{[1,L]}$.

Example 1: A (3,6) LDPC code can be represented by the base matrix

$$\mathbf{B} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix} . \quad (7)$$

The corresponding ensemble in [13] is defined by component base matrices $\mathbf{B}_0 = \mathbf{B}_1 = \mathbf{B}_2 = [1\ 1]$. For $L = 4$ the protograph base matrix of the terminated codes is equal to

$$\mathbf{B}_{[1,4]} = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \end{bmatrix} , \quad (8)$$

which defines the protograph depicted in Fig. 1. The convolutional protograph with base matrix $\mathbf{B}_{[-\infty,\infty]}$ and the terminated graph with base matrix $\mathbf{B}_{[1,10]}$ are illustrated in Fig. 2. □

The protograph of the terminated code has $N_P = LK'$ variable nodes and $M_P = (L+m_s)J'$ check nodes. The design rate R_L of the block code defined by $\mathbf{B}_{[1,L]}$ is therefore equal to

$$R_L = 1 - \left(\frac{L+m_s}{L}\right)\frac{J'}{K'} = 1 - \left(1+\frac{m_s}{L}\right)(1-R) \ , \quad (9)$$

where $R = Mb/(Mc) = 1 - J'/K'$ is the rate of the convolutional code. The density evolution thresholds for the protographs of terminated codes, defined by the base matrices $\mathbf{B}_{[1,L]}$, can be estimated by recursive application of (2) and (3) for different channel parameters ε. For the codes considered in Example 1, the estimated threshold values are equal to $\varepsilon^* = 0.635$ for $L = 4$ (where $R_4 = 1/4$ and $\varepsilon_{sh} = 0.75$) and $\varepsilon^* = 0.505$ for $L = 10$ (where $R_{10} = 2/5$ and $\varepsilon_{sh} = 0.6$), respectively. When L is further increased and the rate approaches $R_\infty = b/c = 1/2$, the threshold eventually converges to a constant value $\varepsilon^* = 0.488$. The Shannon limit is equal to $\varepsilon_{sh} = 0.5$ for rate $R_\infty = 1/2$.

The interesting phenomenon that the threshold does not decay as L increases was first observed empirically in [14]. In [15] it was shown that this holds for arbitrarily large L. 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 level $t = 1$. Once the variable-to-check node message probabilities $p(e^{\mathrm{v}}_{n,j})$, $j = 1, \dots, J_n$, are below some value ε_0 for all nodes V_n at time t, the window is shifted one time unit further. This leads to the following theorem.

Theorem 1: Consider density evolution for the BEC with erasure probability ε and the window updating schedule for an arbitrary termination length L. Let the message probabilities at times $t < 1$ be initialized by some value $\varepsilon_0 > 0$. If, under these conditions, the value ε_0 is reached at time $t = 1$ after some number of iterations, so that the window can be shifted one step further, then, for the actual initial probabilities $p(e^{\mathrm{v}}_{n,j}) = 0$, $j = 1, \dots, J_n$, of nodes V_n at times $t < 1$, the value ε_0 can be reached at all times t, $t = 1, \dots, L$. □

Intuitively, one can explain the result as follows: during the iterations, due to the lower check node degrees at the start and end of the graph, the messages along edges at times $t = 1$ and $t = L$ will be the most reliable ones. Their erasure probabilities have the potential to converge to zero even for channel parameters ε beyond the threshold of the corresponding regular code. But when the symbols at $t = 1$ and $t = L$ are perfectly known, the connected edges can be removed from the protograph, which results in the shortened protograph $\mathbf{B}_{[2,L-1]}$, as illustrated in Fig. 2(b). It follows now by induction that the messages eventually converge to zero at all times $t = 1, \ldots, L$ for arbitrary values L. A proof of Theorem 1 can be found in [16].

C. Protograph LDPCC codes for arbitrary J and K

If we want to construct convolutional protographs for arbitrary rates, we have to face the problem that, in the above described construction, $m_\mathrm{s} = a - 1$ becomes zero if J and K are relatively prime. This results in a sequence of disconnected protographs $\mathbf{B}_0$, each defining a standard regular block code. An essential property of a convolutional protograph is that edges from variable nodes at time t are spread among check nodes at *different* times $t, t+1, \ldots, t+m_\mathrm{s}$, as illustrated for the case $m_\mathrm{s} = 1$ in Fig. 3. Starting from an arbitrary (J, K) block protograph with base matrix $\mathbf{B}$ we can achieve such an *edge spreading* by dividing the entries of $\mathbf{B}$ among various matrices $\mathbf{B}_0, \mathbf{B}_1, \ldots, \mathbf{B}_{m_\mathrm{s}}$. This procedure ensures that the degrees of variable nodes and check nodes of the resulting convolutional protograph are the same as those of the original block protograph.

Example 2: Consider the construction of a convolutional protograph from a $(5, 6)$ LDPC block code, defined by a 5×6 all-one base matrix $\mathbf{B}$. A convolutional protograph, defining a rate $R = Mb/(Mc) = 1/6$ code with $m_\mathrm{s} = 1$, follows from (6) with the component base matrices

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

which can be obtained from $\mathbf{B}$ by edge spreading. The threshold of the terminated protographs $\mathbf{B}_{[1,L]}$ approaches $\varepsilon^* = 0.829$ as L increases. This value is remarkably close to the Shannon limit $\varepsilon_\mathrm{sh} = 0.833$ for rate $R_\infty = 1/6$. □

The convolutional protograph in Fig. 2 can also be constructed using this procedure with

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

The convolutional code has now rate $R = 3M/(6M)$ and memory $m_\mathrm{s} = 1$ instead of $R = M/(2M)$ and $m_\mathrm{s} = 2$. As a consequence, the nodes of the two protographs have different time instants associated with them, but otherwise the structure of the two graphs is the same.

There are many ways of spreading the edges among component base matrices, and different assignments can lead to different thresholds. Even for $m_\mathrm{s} > 0$ there exist assignments that result in a sequence of disconnected subgraphs, e.g., if all-zero columns or rows exist in the component base matrices. A good threshold value is expected when the checks at time $t = 1$

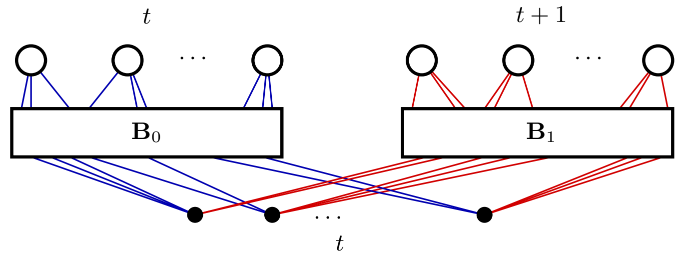

Fig. 3. The protograph connections from variable nodes at time t, represented by the component base matrices $\mathbf{B}_0$ and $\mathbf{B}_1$ for the case $m_\mathrm{s} = 1$.

have low degree (but at least degree two). The convolutional protograph in Example 2 is designed in such a way that all rows in $\mathbf{B}_0$ have weight two and the entries are spread among all rows and columns. Note that, by symmetry, if we reverse the order of the component base matrices (e.g., exchanging $\mathbf{B}_0$ and $\mathbf{B}_1$ in Example 2), the convolutional protograph is simply mirrored horizontally and the threshold is consequently the same. Simple row or column permutations, applied simultaneously to all component base matrices, also do not affect the graph structure.

IV. Improving Thresholds of Irregular Protographs

The edge spreading procedure described above is not restricted to regular protographs, but can be applied to any conventional protograph, including those with multiple edges between a pair of nodes. Starting from an arbitrary block protograph, defined by an $M_P \times N_P$ base matrix $\mathbf{B}$, we divide the edges among times $t, t+1, \ldots, t+m_\mathrm{s}$. For a given target memory m_s, any set of component base matrices $\mathbf{B}_0, \mathbf{B}_1, \ldots, \mathbf{B}_{m_\mathrm{s}}$ which satisfies the condition

$$\sum_{i=0}^{m_\mathrm{s}} \mathbf{B}_i = \mathbf{B} \tag{10}$$

corresponds to a possible assignment of edges, resulting in a convolutional protograph with the same variable and check node degrees as the original block protograph. The corresponding convolutional base matrix $\mathbf{B}_{[-\infty,\infty]}$ follows from (6). Termination of such a convolutional protograph, after an arbitrary number of time instants L, results in a block protograph $\mathbf{B}_{[1,L]}$ with LN_P variable nodes and $(L+m_\mathrm{s})M_P$ check nodes, corresponding to a design rate

$$R_L = 1 - \left(\frac{L+m_\mathrm{s}}{L}\right)\frac{M_P}{N_P} = 1 - \left(1 + \frac{m_\mathrm{s}}{L}\right)(1 - R_\mathrm{B}) \ , \tag{11}$$

where $R_\mathrm{B} = 1 - M_P/N_P$ is the design rate of codes obtained from the original block protograph.

Example 3: Consider the protograph of an accumulate-repeat-by-4-jagged-accumulate (AR4JA) code [10], as depicted in Fig. 4. The base matrix of this code is equal to

$$\mathbf{B} = \begin{bmatrix} 1 & 2 & 0 & 0 & 0 \\ 0 & 3 & 1 & 1 & 1 \\ 0 & 1 & 2 & 1 & 2 \end{bmatrix} . \tag{12}$$

The variable nodes corresponding to the second column in $\mathbf{B}$

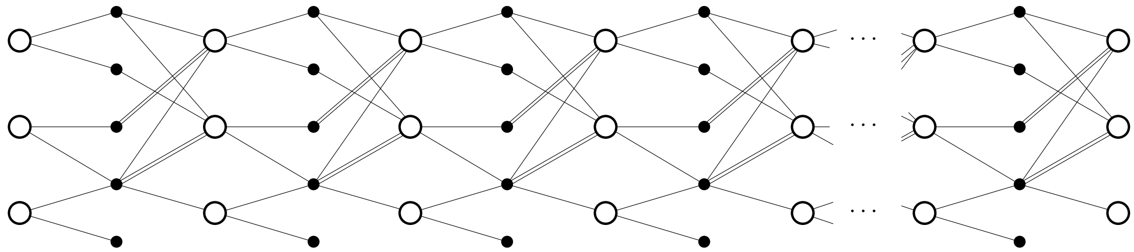

Fig. 5. A terminated convolutional protograph obtained from the AR4JA code in Fig. 4.

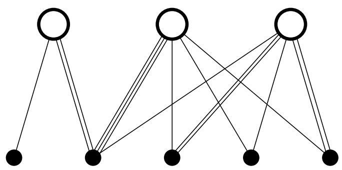

Fig. 4. The protograph of an AR4JA code.

are punctured, resulting in a design rate equal to $R_{\mathrm{B}} = 1/2$ and a threshold of $\varepsilon^* = 0.4387$. Using the edge spreading procedure, we derive the base matrices

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

The resulting terminated convolutional protograph $\mathbf{B}_{[1,L]}$ is illustrated in Fig. 5. Its threshold approaches $\varepsilon^* = 0.4996$ as L increases, which is very close to the Shannon limit $\varepsilon_{\mathrm{sh}} = 0.5$ for rate $R_\infty = 1/2$. □

V. Conclusion

We presented a technique for the construction of asymptotically regular protographs with thresholds close to the Shannon limit. These protographs can be derived by termination from convolutional protographs, which were obtained from regular (J, K) protographs by means of an edge spreading technique, where we assume that $J > 2$. Since all variable nodes have degree greater than two, asymptotically the error probability converges at least doubly exponentially with decoding iterations and the minimum distance grows linearly with the length of the codes. As a result we obtain sequences of asymptotically good LDPC codes with fast convergence rates and thresholds close to capacity. The construction can also be generalized to arbitrary irregular protographs. Although we restricted the discussion to the BEC, the results for $(J, 2J)$ codes in [15] indicate that a similar behavior can be expected for the additive white Gaussian noise channel.

References

[1] R. Gallager, *Low-Density Parity-Check Codes*, MIT Press, Cambridge, MA, 1963.

[2] R. M. Tanner, "A recursive approach to low complexity codes," *IEEE Transactions on Information Theory*, vol. IT-27, no. 9, pp. 533–547, Sept. 1981.

[3] M. Luby, M. Mitzenmacher, M. A. Shokrollahi, D. A. Spielman, and V. Stemann, "Practically loss-resilient codes," in *Proc. 29th Annual ACM Symposium on Theory of Computing*, 1997, pp. 150–149.

[4] M. Luby, M. Mitzenmacher, M. A. Shokrollahi, and D. A. Spielman, "Efficient erasure correcting codes," *IEEE Transactions on Information Theory*, vol. IT-47, no. 2, pp. 569–584, Feb. 2001.

[5] P. Oswald and A. Shokrollahi, "Capacity-achieving sequences for the erasure channel," *IEEE Trans. Inform. Theory*, vol. 48, no. 12, pp. 3017–3028, Dec. 2002.

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

[7] A. Sahai and P. Grover, "The price of certainty: iterative decoding from a total power perspective," *Information Theory and Applications Workshop, 2008*, pp. 293–302, Feb. 2008.

[8] J. Thorpe, "Low-density parity-check (LDPC) codes constructed from protographs," in *IPN Progress Report 42-154, JPL*, Aug. 2003.

[9] D. Divsalar and C. Jones, "Protograph LDPC codes with node degrees at least 3," *Global Telecommunications Conference, 2006. GLOBECOM '06. IEEE*, pp. 1–5, Dec. 2006.

[10] D. Divsalar, S. Dolinar, and C. Jones, "Construction of protograph LDPC codes with linear minimum distance," *Information Theory, 2006 IEEE International Symposium on*, pp. 664–668, July 2006.

[11] A. Jiménez Feltström and K.Sh. Zigangirov, "Periodic time-varying convolutional codes with low-density parity-check matrices," *IEEE Trans. Inform. Theory*, vol. 45, no. 5, pp. 2181–2190, Sept. 1999.

[12] A. Sridharan, M. Lentmaier, D. V. Truhachev, D. J. Costello, Jr., and K. Sh. Zigangirov, "On the minimum distance of low-density parity-check codes with parity-check matrices constructed from permutation matrices," *Probl. Inf. Transm. (Probl. Pered. Inform.)*, vol. 41, no. 1, pp. 33–44, 2005.

[13] A. Sridharan, D.V. Truhachev, M. Lentmaier, D.J. Costello, Jr., and K.Sh. Zigangirov, "Distance bounds for an ensemble of LDPC convolutional codes," *IEEE Trans. Inform. Theory*, vol. 53, no. 12, Dec. 2007.

[14] A. Sridharan, M. Lentmaier, D. J. Costello, Jr., and K. Sh. Zigangirov, "Convergence analysis of a class of LDPC convolutional codes for the erasure channel," in *Proceedings of the 42nd Allerton Conference on Communication, Control, and Computing*, Monticello, IL, USA, 2004.

[15] M. Lentmaier, A. Sridharan, K. Sh. Zigangirov, and D. J. Costello, Jr., "Terminated LDPC convolutional codes with thresholds close to capacity," in *Proc. IEEE International Symposium on Information Theory*, Adelaide, Australia, Sept. 2005, pp. 1372–1376.

[16] M. Lentmaier, A. Sridharan, D.J. Costello, Jr., and K.Sh. Zigangirov, "Iterative decoding threshold analysis for LDPC convolutional codes," submitted to *IEEE Trans. Inform. Theory*.