

The Design of Rate-Compatible Protograph LDPC Codes

Thuy Van Nguyen, *Student Member, IEEE*, Aria Nosratinia, *Fellow, IEEE*, and Dariush Divsalar, *Fellow, IEEE*

Abstract—This paper presents a simple yet effective method for designing nested families of LDPC codes. Rate compatible codes are essential for many communication applications, e.g. hybrid automatic repeat request (HARQ) systems, and their design is nontrivial due to the difficulty of simultaneously guaranteeing the quality of several related codes. Puncturing can be used to generate rate-compatible LDPC codes, but it produces a gap to capacity that, in practice, often significantly exceeds the gap of the mother code. We propose an alternative method based on successively extending a high-rate protograph. The resulting codes not only inherit the advantages of protograph codes, namely low encoding complexity and efficient decoding algorithms, but also cover a wide range of rates and have very good performance with iterative decoding thresholds that are within 0.2 dB of their capacity limits.

Index Terms—LDPC codes, protograph codes, rate-compatible, iterative decoding threshold, EXIT chart.

I. INTRODUCTION

A Rate-compatible family of codes allows coding across a range of rates using a common encoder/decoder infrastructure [1]–[8]. This allows convenient matching of the code rate to channel conditions and also provides an efficient realization for incremental hybrid automatic repeat request (HARQ). In HARQ, upon an unsuccessful reception, the transmitter transmits incremental parity bits to allow the decoder to decode the data with a lower-rate (more powerful) code. This requires a rate-compatible family of codes.

The most common way of generating a rate-compatible family of codes is by puncturing, i.e., starting with a low-rate mother code and then selectively discarding some of the coded bits (parity bits) to arrive at higher rate codes [2]–[4]. This approach is simple but is not free of problems [9]. Specifically, (1) the mother code is optimally designed for low rates, so higher-rate punctured codes have iterative decoding thresholds with a wider gap to capacity than that of the mother code; (2) the optimal low rate code structure and puncturing patterns are designed separately, which is suboptimal. Even though it has been shown that puncturing can theoretically achieve the same gap to capacity as the mother code, in

practice puncturing has increased the gap significantly [2], [10].

In this paper, we propose a simple, systematic procedure to search for good nested protograph LDPC codes that have a low decoding threshold (close to the capacity limit) and also a minimum distance that grows linearly with block-length [11], [12], a necessary condition for avoiding an error floor. A protograph code is an LDPC code that can be constructed from a small protomatrix with a few elements [13]. We start by producing nested protograph codes where the parity check matrix of the higher rate code is a lengthened version of the lower rate one. This lengthening structure is motivated by [11] but the resulting codes are better than those reported in [11]. This family is suitable for applications where adaptive coding and/or unequal error protection is required as well as low complexity. However, code lengthening changes the information block size, thus the resulting codes are not useful for some applications, e.g., HARQ.

To achieve full rate-compatibility, we use another approach involving code extension. Code extension starts with a high-rate code (a daughter code), then lower rate codes are obtained by extending the parity check matrix of the daughter code [5], [9], [14], [15]. Most existing extension-based LDPC codes [5], [9], [14] are designed as rate-compatible irregular LDPC codes with highly optimized framework and unstructured design that does not promote low-complexity encoding. In contrast, the proposed rate-compatible protograph-based codes can achieve very good thresholds with low encoding complexity allowed by circulant permutations [11], [16].

Several protograph-based rate-compatible codes have appeared in the literature [11], [17]–[20]. While some of these codes have reasonably good performance, the designs are by trial-and-error and thus time consuming. Some designs manually manipulate within a given structure [11], [21] or use node splitting and edge growth techniques [11], [18]. But without a comprehensive search one may easily miss good codes in this class, in fact this paper shows the protograph codes mentioned above can be improved upon. Furthermore, manual design makes it impractical to use larger protographs that are needed for implementing a wide range of rates, and also for better performance. Also noteworthy is [15] which produced rate-compatible protograph codes based on node-splitting and attaching additional accumulators, but [15] only produces rates higher than 1/2 and furthermore the resulting iterative decoding threshold gaps to capacity are higher than that of the codes reported in this paper.

To summarize, the main contribution of this paper is a simple method to design rate-compatible codes within a wide

Paper approved by O. Milenkovic, the Editor for Coding Theory and Applications of the IEEE Communications Society. Manuscript received January 4, 2011; revised October 26, 2011 and April 8, 2012.

T. V. Nguyen and A. Nosratinia are with the Department of Electrical Engineering, The University of Texas at Dallas, Richardson, TX 75083-0688, USA (e-mail: {nvanthuy, aria}@utdallas.edu).

D. Divsalar is with the Jet Propulsion Laboratory, California Institute of Technology, Pasadena, CA 91109-8099, USA (e-mail: dariush.divsalar@jpl.nasa.gov).

The work in this paper was presented in part in the Allerton Conference on Communications, Control and Computing, 2010.

Digital Object Identifier 10.1109/TCOMM.2012.081012.110010

range of rates, for adaptive coding and HARQ applications. Rate compatible codes are produced with rates from 0.32 to 0.88 with iterative decoding thresholds within 0.2 dB of capacity and a linear minimum distance growth property [11]. As a by-product of our main results, we also produce nested protograph codes that do *not* have uniform information size. The rates of these codes range from 1/2 to 9/10 and have thresholds within 0.2 dB of capacity. Our protograph codes perform close to capacity uniformly and have the lowest iterative decoding thresholds among structured LDPC codes with linear minimum distance growth property reported so far in the literature.

II. DESIGN OF PROTOGRAPH CODES

A protograph [13] is a Tanner graph with a relatively small number of nodes, connected by a small number of edges, which also allows parallel edges. A protograph is represented by a protomatrix whose entries indicate the number of edges connecting the respective variable and check nodes. A protograph code (an equivalent LDPC code) is a larger derived graph constructed by applying a “copy-and-permutation” operation on a protograph. Protograph LDPC codes are a subclass of multi-edge type LDPC codes [22] where each edge of a protograph corresponds to one edge type.

The decoding threshold of a protograph refers to the minimum channel SNR that supports reliable iterative decoding of asymptotically large LDPC codes built from the protograph. The decoding threshold of an LDPC code can be computed by using density evolution (DE) [23]. The density evolution algorithm tracks the distribution of messages exchanged in the Tanner graph, but it is computationally cumbersome, especially for our purposes because we intend to use it inside an optimization loop as explained in the sequel. There are approximate techniques that reduce the complexity of the calculation of the decoding threshold: the extrinsic information transfer (EXIT) chart [24] and reciprocal channel approximation (RCA) [11]. In this paper we use a variation of the EXIT chart.

It is important to note that threshold calculation for a protograph code does not exactly follow the ordinary LDPC threshold calculation, because the copy-and-permute operation does not allow full randomization of the edges of the code graph; the edge(s) arriving in a node in one copy of the protograph can only be permuted with the edges of the similar node in other copies of the protograph. Thus, a protograph code has more structure than a general LDPC code with similar degree distribution, and therefore may not have the same threshold. In fact, two LDPC protograph codes arising from two different protographs, even though having the same degree distribution, may not have the same threshold. The decoding threshold of a protograph code is a function of both the degree distribution as well as its protograph protomatrix.

Because of this dependency the general EXIT chart, which is designed for generic LDPC ensembles, cannot be accurately applied to protograph codes: it can neither account for degree-1 variables nor the dependency on the protomatrix as mentioned above [25]. To solve these problems, Liva and Chiani [25] proposed the PEXIT method. The PEXIT algorithm

computes the iterative threshold of a protograph based on its edge degree distribution (via its protomatrix) rather than node degree distribution as in general EXIT chart [24]. We use the PEXIT algorithm in this paper.

A. Properties of a Good Protograph

1) *Low Iterative Decoding Threshold:* It is well known that optimized degree distribution of unstructured LDPC code ensemble should contain at least one very high degree variable nodes and a substantial fraction of degree-2 variable nodes [23]. This also applies to protograph structures [11], [13]. Protograph-based iterative decoding threshold can be further improved by including a few degree-1 variable nodes (usually called a pre-coder) and a very high degree variable node that is optionally punctured. This construction appeared in the accumulate-repeat-accumulate code [21]. The role of degree-1 variable nodes was also observed by Richardson in multi-edge type codes [22]. To summarize, the combination of one or more degree-1 variable node, one very high degree variable node and several degree-2 variable nodes are likely to lead to a good protograph.

2) *Linear Minimum Distance Growth:* Linear minimum distance growth is a desired property of an LDPC code, which can be verified by computing the asymptotic ensemble weight enumerator [11]. It is known that the LDPC codes with variable node degrees greater than or equal to 3 automatically have such a property [23]. However, as seen in the previous subsection, a good protograph usually includes degree-2 and even degree-1 variable nodes. In order to include degree-2 variable nodes within the graph structure, Divsalar, et al. [11] proposed a check node splitting method that allows to build a graph with degree-2 from a graph having variable nodes that have higher degree (≥ 3) while still preserving its linear minimum distance growth property. This technique requires the maximum number of degree-2 variable nodes in the protograph to be limited by total number of checks minus 1, and also there should be neither parallel edges nor cycles among these degree-2 nodes [26].

3) *Protograph Structure with Degree-1 Variable Nodes:* Degree-1 variable nodes (accumulators) are pre-coders that improve the iterative decoding threshold. We divide the protograph into two sub-graphs: One sub-graph contains the degree-1 variable nodes, the corresponding check nodes, and any variables connected to these check nodes. The other subgraph contains all variables with degree greater than 1. The latter subgraph is LDPC-like, since good random LDPC codes do not have degree-1 variable nodes. The two subgraphs have in common certain variables of degree > 1 that are connected to the check node(s) associated with degree-1 node(s) (see for example Fig. 1).

Since the degree-1 variables do not affect the growth of minimum distance with codeword length [11], any discussion regarding linear growth of minimum distance applies only to the LDPC subgraph.

As an example, the structure of the rate-1/2 AR4JA code from [11] is shown in Fig. 1 in which dark circles are transmitted variable nodes, the white circle is un-transmitted (punctured) variable node and plus circles are check nodes.

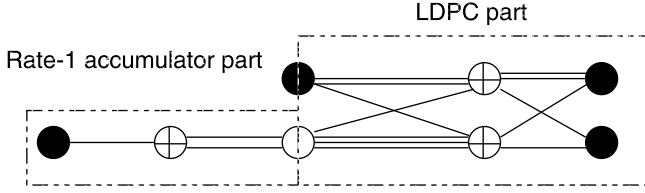


Fig. 1. The rate 1/2 AR4JA protograph as a concatenation of a pre-coder and a protograph-based LDPC code.

It is a concatenation of a pre-coder (degree-1 part or a rate-1 accumulator) with an LDPC subgraph. The protomatrix of the rate-1/2 AR4JA protograph is

$$H_{1/2} = \left(\begin{array}{c|cccc} 1 & 2 & 0 & 0 & 0 \\ 0 & 3 & 1 & 1 & 1 \\ 0 & 1 & 2 & 2 & 1 \end{array} \right) \quad (1)$$

Using the PEXIT technique, we find this rate-1/2 code has a threshold of 0.6337 dB.

The LDPC protograph subgraph, i.e., the two rows in the right-bottom part of (1), consists of only two check nodes. Following the linear minimum distance criteria, there is only one degree-2 variable node allowed in the protograph structure.

Although it is a good code, the AR4JA code has not been designed by a systematic search of the design space and therefore it can be improved upon. In the following, we outline a method for a systematic and efficient search of the design space that finds better codes.

B. Design Method

Based on the discussion above, our design procedure is:

- 1) Start with a desired code rate, determine number of check and variable nodes.
- 2) Impose empirical constraints for good threshold (see Section II-A1.)
- 3) Impose constraints for linear minimum distance growth (see Section II-A2.)
- 4) Among the graphs satisfying the constraints, select the graph with the lowest iterative decoding threshold.

The rate of the code does not uniquely determine the number of check and variable nodes, rather, if the number of check and variable nodes is respectively n_c and n_v , assuming n_p punctured variable nodes, $R = \frac{n_v - n_c}{n_v - n_p}$. So for the same rate, we may have larger or smaller protographs. Larger protographs of the same rate may yield better thresholds because of the larger search space and the design flexibility. However, as protographs grow bigger there is a point of diminishing returns for the optimization of threshold and the search algorithm becomes more complex. Furthermore, if the overall codeword length is constant, a larger protograph means that the repetition factor of the protograph will be smaller, which restricts the design of the interleaver and makes it more difficult to avoid an error floor due to trapping sets.

With all that in mind, consider the following example for the design of a rate-1/2 protograph code. We begin with a simple protograph structure which has 7 variable nodes (of which the third node is punctured) and 4 check nodes. We

shall see that this choice of the number of check and variable nodes leads to a threshold that is within a small fraction of a dB to capacity, therefore a larger protograph is not needed.¹ The rate of the proposed protograph is $R = \frac{7-4}{7-1} = \frac{3}{6} = \frac{1}{2}$. We include a punctured variable node since it has been observed that punctured (un-transmitted) variable nodes can improve the performance [11], [18], [22]. We represent the protograph by its 4×7 protomatrix. This protomatrix contains 28 elements each indicating how many parallel edges connect the respective check node (row) and variable node (column). Optimization over these 28 variables is computationally complex and the finding of the optimum in a high-dimensional space is difficult, therefore adjustments to this optimization problem are necessary to make a practical solution viable.

As discussed earlier, we institute one degree-1 variable (one column of weight-1), and one degree-2 variable node (column of weight-2). Therefore the protomatrix is:

$$H_{1/2}^{search} = \left(\begin{array}{c|cccccc} 1 & 0 & y_1 & y_2 & y_3 & y_4 & y_5 \\ 0 & 1 & x_1 & x_4 & x_7 & x_{10} & x_{13} \\ 0 & 1 & x_2 & x_5 & x_8 & x_{11} & x_{14} \\ 0 & 0 & x_3 & x_6 & x_9 & x_{12} & x_{15} \end{array} \right)_{4 \times 7} \quad (2)$$

The variables x_i and y_j designate the remainder of the protograph to be designed. The overall code is effectively the concatenation of an LDGM code (first row and column) with an LDPC code (columns 2 through 7, rows 2 through 4). The linear minimum distance growth only involves the LDPC part of (2), therefore all the comments in the remainder of this paragraph are focused on this submatrix. As shown in [26], to have linear minimum distance growth, the number of degree-2 nodes in an LDPC protograph must be limited to the number of check nodes in the LDPC part minus 1, that is $3 - 1$. It is also known that degree-2 nodes cannot form a cycle among themselves. However, in our design, we only search for one degree-2 variable node since extensive experiments show that two degree-2 variable nodes will produce an inferior threshold.

Other variable nodes within the LDPC part of the matrix must have degree at least 3, therefore the sum of columns containing x_i , except for up to two degree-2 nodes, must be 3 or higher.

To limit the search space, we constrain the maximum number of parallel edges in the protograph. Via experiments, we observed that increasing the number of parallel edges beyond a certain point is not useful. The reason is that subject to a given code length, increasing the number of parallel edges will increase the likelihood of short cycles. For the remaining nodes, we set the maximum number of parallel edges to 3.

These constraints reduce both the dimensionality of the search space as well as the breadth of the discrete search in each dimension, making a systematic search viable. Thus we find:

$$H'_{1/2} = \left(\begin{array}{c|cccccc} 1 & 0 & 2 & 0 & 0 & 1 & 0 \\ 0 & 1 & 3 & 1 & 1 & 1 & 0 \\ 0 & 1 & 1 & 2 & 2 & 2 & 1 \\ 0 & 0 & 2 & 0 & 0 & 0 & 2 \end{array} \right)_{4 \times 7} \quad (3)$$

¹Experiments showed that smaller protographs were not satisfactory.

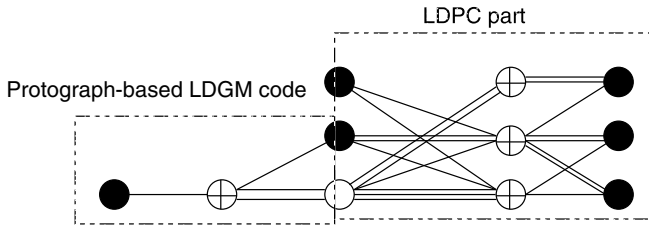


Fig. 2. The new rate 1/2 protograph with a threshold of 0.395 dB.

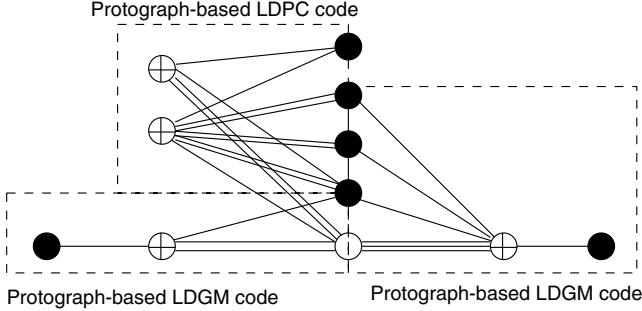


Fig. 3. A rate 1/2 protograph with a threshold of 0.250 dB.

The threshold of this code is 0.395 dB which shows a gap of 0.208 dB of capacity. The new protograph is shown in Fig. 2. This code shows improvements over the AR4JA rate-1/2 code reported in [11]. However, it should also be mentioned that the AR4JA protograph has fewer nodes, and within the class of protographs with 5 variable nodes the AR4JA code remains very attractive.

It is instructive to note that if we do not impose the linear minimum distance growth criteria, the optimization procedure yields a protograph with a lower threshold that has a protomatrix as follows

$$H_{1/2}^{opt} = \left(\begin{array}{c|cccc|c} 1 & 0 & 2 & 0 & 0 & 1 & 0 \\ 0 & 1 & 2 & 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 2 & 2 & 2 & 0 \\ 0 & 0 & 3 & 1 & 1 & 1 & 1 \end{array} \right) \quad (4)$$

This code has a threshold of 0.250 with a gap of only 0.063 dB to capacity. This is the best iterative decoding threshold reported for a structured (non-random) code in the literature, but it does not mean it is a good code. In fact, this is the concatenation of two LDGM codes (the first column-row; and last column-row in (4)) with one LDPC code (corresponding to the middle sub-matrix of (4)) as plotted in Fig. 3. The LDPC subgraph has too many degree-2 variable nodes (3 degree-2 variables and 2 checks) which violates the condition for linear minimum distance [11], [26]. Therefore the minimum distance of this code does not grow linearly with codeword length and it suffers from an error floor.

III. DESIGN OF HIGH RATE PROTOGRAPH CODES BY LENGTHENING

So far we have concentrated on rate-1/2 protograph codes. Designing higher rate protographs is more difficult because the protomatrix contains many more elements. In this section, we propose a simple approach to overcome this difficulty. We produce codes whose iterative decoding thresholds are close

to capacity, form a nested family of codes and offer a range of rates.

We construct a family of high-rate protograph codes from a low-rate protograph by code lengthening, i.e., by lengthening the parity check matrix of a lower-rate code (a base code) in the form of

$$H_{high\ rate} = [H_1 H_e] \quad (5)$$

where H_1 is the parity check matrix of the low-rate code and H_e is an extension matrix.

Several families of nested protograph codes based on code lengthening have been reported in the literature [11], [21] using node splitting and permutation of the edges in an ad-hoc manner that cannot be generalized to arbitrary mother codes and rates. Therefore, there is motivation to find a systematic approach for the design of nested protograph codes.

We will describe in the following an example for the design of high-rate codes with rate $R = \frac{n+1}{n+2}$, $n = 1, 2, \dots$ as an extension of our optimized rate-1/2 protograph of Eq. (3). These codes have a minimum distance that grows linearly with code length. Similar to the previous section, we describe the search space for our code in the following way

$$H_{\frac{n+1}{n+2}}^{search} = \left(\begin{array}{c|ccc} & y_1 & y_2 & y_3 \\ \hline H'_{\frac{n}{n+1}} & x_1 & x_4 & x_7 \\ & x_2 & x_5 & x_8 \\ & x_3 & x_6 & x_9 \end{array} \right) \quad (6)$$

where variables y_j in the first row correspond to the check node that connects to the degree-1 variable node. Variables x_i in rows 2-4 constitute the extension to the LDPC subgraph. In order to preserve the linear growth of minimum distance for the new high rate codes, the column sums in the LDPC subgraph, namely sums of columns designated with variable x_i , should be 3 or higher. This constraint ensures that all variables in LDPC part of the extension structures have degree ≥ 3 .

We further simplify the problem by setting the maximum number of parallel edges to 2 (i.e., $x_i, y_j \in \{0, 1, 2\}$). We obtain:

$$\begin{aligned} H'_{2/3} &= \left(\begin{array}{c|ccc} & 0 & 1 & 1 \\ \hline H'_{1/2} & 1 & 0 & 1 \\ & 2 & 2 & 2 \\ & 0 & 1 & 0 \end{array} \right) & H'_{3/4} &= \left(\begin{array}{c|ccc} & 0 & 0 & 1 \\ \hline H'_{2/3} & 1 & 2 & 0 \\ & 2 & 1 & 2 \\ & 0 & 0 & 2 \end{array} \right) \\ H'_{4/5} &= \left(\begin{array}{c|ccc} & 0 & 1 & 2 \\ \hline H'_{3/4} & 1 & 1 & 2 \\ & 2 & 2 & 1 \\ & 0 & 0 & 0 \end{array} \right) & H'_{5/6} &= \left(\begin{array}{c|ccc} & 0 & 0 & 0 \\ \hline H'_{4/5} & 0 & 2 & 2 \\ & 2 & 1 & 1 \\ & 2 & 0 & 0 \end{array} \right) \\ H'_{6/7} &= \left(\begin{array}{c|ccc} & 0 & 1 & 2 \\ \hline H'_{5/6} & 1 & 1 & 2 \\ & 2 & 2 & 2 \\ & 0 & 0 & 0 \end{array} \right) & H'_{7/8} &= \left(\begin{array}{c|ccc} & 0 & 0 & 1 \\ \hline H'_{6/7} & 2 & 2 & 0 \\ & 1 & 2 & 1 \\ & 0 & 0 & 2 \end{array} \right) \\ H'_{8/9} &= \left(\begin{array}{c|ccc} & 0 & 0 & 1 \\ \hline H'_{7/8} & 1 & 2 & 1 \\ & 2 & 1 & 2 \\ & 0 & 1 & 0 \end{array} \right) & H'_{9/10} &= \left(\begin{array}{c|ccc} & 0 & 0 & 2 \\ \hline H'_{8/9} & 0 & 2 & 0 \\ & 1 & 1 & 2 \\ & 0 & 0 & 2 \end{array} \right) \end{aligned}$$

The iterative decoding thresholds of these codes are given in Table I. For rates $> 2/3$, the produced codes have decoding thresholds within 0.09 dB of capacity, and the rate-2/3 code has a threshold within 0.122 dB of capacity. This shows an improvement of 0.2 dB compared with the AR4JA family [11].

TABLE I
 THRESHOLD OF THE NEW PROTOGRAPH CODE FAMILY

Code Rate	Protograph threshold (dB)	Capacity threshold (dB)	Gap to capacity
1/2	0.395	0.187	0.208
2/3	1.181	1.059	0.122
3/4	1.701	1.626	0.075
4/5	2.112	2.040	0.062
5/6	2.445	2.362	0.083
6/7	2.702	2.625	0.077
7/8	2.930	2.845	0.085
8/9	3.123	3.042	0.081
9/10	3.288	3.199	0.089

IV. DESIGN OF RATE-COMPATIBLE PROTOGRAPH CODES

Although we have successfully designed a family of nested codes in Section III, the information block-lengths of the nested codes are not identical. The same is true of AR4JA codes [11]. Thus, these codes are not truly rate-compatible and are unsuitable e.g. for HARQ applications. Several works in the literature design irregular LDPC rate-compatible codes using puncturing and code extension [2], [3], [5], [9], [14] in a manner that requires painstaking optimization and furthermore their encoders are unstructured therefore computationally complex. Our design avoids the weaknesses of puncturing by using a code extension approach.

El-Khamy *et al.* [18] design rate-compatible protograph LDPC codes by first extending and randomly pruning from an existing protograph to produce a low-rate code, and then obtaining other high-rate codes by puncturing this low-rate code. As mentioned in Section I, puncturing has several weaknesses and usually results in codes that have iterative decoding thresholds with a wider gap to capacity than that of the low-rate mother code. The weaknesses of puncturing can be avoided by using a code extension approach, as discussed below for our designs.

The new rate-compatible family of codes is constructed by extending the parity check matrix H of a high-rate protograph code by an equal number of columns (variable nodes) and rows (check nodes), ensuring that the new code will have the same information block size (Fig. 4). In this figure, the parity check matrix $H_{1/2}$ of the rate 1/2 code is extended to create $H_{1/3}$, which in turn is extended to obtain $H_{1/4}$. The zero submatrices ensure that the incremental variable nodes are indeed new parity bits determined only by the new parity check equations, thus guaranteeing that the high-rate code is nested inside the low-rate code.

In the interest of brevity, we only present one construction of a nested rate-compatible family of codes. However, the proposed method is completely general; we can start from any high-rate code to build a successive set of lower-rate codes. The starting point for the following construction is the rate-5/6 protograph code designed in the last section. Since an equal number of columns and rows are added to the protomatrix, the lower-rate codes obtained from the rate-5/6 code above have rates $R = \frac{19-4}{19-1+n} = \frac{15}{18+n}$, where n is the number of checks and variables added. For each value of n , the new code will

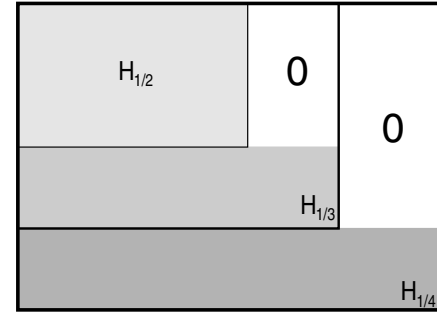


Fig. 4. Rate-compatible parity check matrices by extension.

require a search. As a representative sample, we concentrate on the search for $n = 2$, which yields a rate $R = \frac{15}{20} = 0.75$ code. The search space for this new code is in the form

$$H_{0.75}^1 = \left(\begin{array}{ccccccccc|cc} & & & & & & & & & 0 & 0 \\ & & & & & & & & & 0 & 0 \\ & & & & & & & & & 0 & 0 \\ & & & & & & & & & 0 & 0 \\ & & & & & & & & & 1 & 0 \\ 0 & x_1 & y_1 & x_3 & . & . & . & x_{17} & & z & 1 \\ 0 & x_2 & y_2 & x_4 & . & . & . & x_{34} & & & \end{array} \right) \quad (7)$$

The problem is to find edges that connect the old graph (the rate-5/6 code) to 2 new check nodes and variable nodes. The unknown edges are represented with variables x_i , y_j and z , which are divided into two types. The first edge type connects the variables of the old graph to new checks, denoted by x_i and y_j , where variables y_j is the number of edges connecting to the highest degree variable node in the old graph, and variables x_i are the remaining edges connecting to the rest of old variables except degree-1 node. The other type consists of edges that connect the new variable nodes to the new check nodes, denoted by variable z in the right-bottom corner sub-matrix. In order to maintain linear growth of minimum distance, more care is needed in this case since degree-2 variables might arise that form short cycles. Therefore we also further simplify the problem by setting this submatrix to be lower triangular.

The optimization process is simplified by further limiting the maximum number of parallel edges. For example, we can set y_j to the values of 1, 2, x_i and z_k to the values of 0, 1. The resulting rate-0.75 protograph has a protomatrix given by:

$$H_{0.75} = \left(\begin{array}{cccccccccccccccc|cc} 1 & 0 & 2 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 1 & 2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 3 & 1 & 1 & 1 & 0 & 1 & 0 & 1 & 1 & 2 & 0 & 1 & 1 & 2 & 0 & 2 & 2 & 0 & 0 \\ 0 & 1 & 1 & 2 & 2 & 2 & 1 & 2 & 2 & 2 & 2 & 1 & 2 & 2 & 2 & 1 & 2 & 1 & 1 & 0 & 0 \\ 0 & 0 & 2 & 0 & 0 & 0 & 2 & 0 & 1 & 0 & 0 & 0 & 2 & 0 & 0 & 0 & 2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 1 \end{array} \right) \quad (8)$$

This protograph has an iterative decoding threshold of 1.706 dB with a gap of only 0.078 dB to capacity. Applying this procedure, we are able to generate a family of rate-compatible codes with rates ranging from 0.32 to 0.88, based on the rate-5/6 protograph. To get as many lower rate codes as possible, we only add one variable node and one check node to the mother code. These codes have rates $R = \frac{15}{19-1+n}$, $n = 1, \dots, 28$, i.e., rates from $\frac{15}{46} = 0.32$ to $\frac{5}{6} = 0.83$. The

TABLE II
THRESHOLD OF RATE-COMPATIBLE PROTOGRAPH CODE FAMILY

Code Rate	Protophraph threshold (dB)	Capacity threshold (dB)	Gap to capacity	Code Rate	Protophraph threshold (dB)	Capacity threshold (dB)	Gap to capacity
15/17	3.137	2.945	0.192	15/32	0.132	0.050	0.082
5/6	2.445	2.362	0.083	15/33	0.080	-0.014	0.094
15/19	2.046	1.948	0.098	15/34	0.032	-0.070	0.102
3/4	1.706	1.628	0.078	3/7	0.002	-0.118	0.116
15/21	1.434	1.364	0.070	15/36	-0.067	-0.170	0.103
15/22	1.208	1.143	0.065	15/37	-0.106	-0.217	0.111
15/23	1.027	0.968	0.059	15/38	-0.144	-0.256	0.112
15/24	0.878	0.816	0.062	15/39	-0.171	-0.295	0.124
3/5	0.785	0.682	0.103	15/40	-0.231	-0.334	0.103
15/26	0.706	0.559	0.147	15/41	-0.259	-0.375	0.116
5/9	0.548	0.471	0.077	15/42	-0.316	-0.410	0.094
15/28	0.432	0.357	0.075	15/43	-0.357	-0.432	0.075
15/29	0.383	0.265	0.118	15/44	-0.382	-0.469	0.087
1/2	0.278	0.188	0.090	1/3	-0.402	-0.501	0.099
15/31	0.198	0.109	0.089	15/46	-0.421	-0.532	0.111

rate-0.88 (15/17) protograph is obtained by puncturing the degree-1 variable node in the rate-5/6 protograph structure. The protomatrix of the smallest rate ($R = 15/46$) code in this rate-compatible structure is given in Eq. (9), from which all other protomatrices can be deduced. This protograph has 47 variable and 32 check nodes with the highest degree variable node is punctured. The thresholds of the new rate-compatible codes are given in Table II. As seen in the table, these codes have thresholds uniformly within 0.2 dB of their capacity limits. All the codes have linear minimum distance growth property [11].

V. ENSEMBLE TRAPPING SET ANALYSIS

In the design of LDPC codes with belief propagation decoding one must avoid graph imperfections due to non-codewords such as trapping sets, which may result in error floors. Obtaining the exact trapping sets has difficulty that grows with block length,² therefore we provide an ensemble trapping set analysis that applies to protograph codes with linear minimum distance growth property, a property that our codes enjoy. Our simulations show that from the viewpoint of the end results, this approach has worked well.

Instead of computing trapping set enumerators for specific protograph codes [27], we prove that under certain conditions the minimum size (a, b) trapping sets grows linearly with the block length if linear minimum distance property for protograph code is satisfied. Our codes satisfy this property via the number of degree-2 nodes with respect to the checks and lack of loops among degree-2 nodes. This is in fact a condition that is required for linear minimum size growth of (a, b) trapping sets with the block length if $b < a$. Here we present a sketch of the proof.

An (a, b) trapping set $\tau_{a,b}$ is a set of variable nodes of size a which induces a subgraph with exactly b odd-degree check nodes (unsatisfied check nodes) and an arbitrary number of even-degree check nodes.

Define $\Delta = b/a$, and trapping sets $\tau_\Delta = \{\tau_{a,b} | b = \Delta \cdot a\}$. Let $d(\Delta)$ represent the size of the $(d, \Delta \cdot d)$ trapping set in τ_Δ .

²Not coincidentally, works involving exact trapping set calculation often target short block lengths.

Let the code block length to be $n = N \times n_v$ where N is the protograph expansion factor. The number of variable nodes is denoted by n_v , the number of checks by n_c . Denote the degree of variable node v_i by q_{v_i} and degree of check node c_i by q_{c_i} . Consider a class of protograph LDPC codes where degree-2 subgraph of the protograph does not have any cycles. This requires the number of degree-2 nodes in the subgraph be strictly less than the number of checks in the subgraph. Denote the total number of degree-2 variable nodes by $t_2 < n_c$. The enumerator in [27] to count the trapping sets (a, b) is based on modifying the base protograph by adding an additional degree-1 variable node to each check for counting the unsatisfied check nodes. Thus the total number of variable nodes in the modified protograph is $n'_v = n_v + n_c$, and the modified check nodes are denoted by c'_j . With this modification the weight enumerators analysis can be used to count the trapping sets (a, b) .

Theorem 1: For the ensemble of protograph-based LDPC codes with linear minimum distance property, a typical minimum trapping set size $\delta_{min}(\Delta) > 0$ exists for some $0 < \Delta < \Delta^*$, i.e. $Pr\{d(\Delta) < N\delta_{min}(\Delta)\} \rightarrow 0$ as $N \rightarrow \infty$.

Proof: Define $X_{d(\Delta)}$ as the number of $(d, \Delta \cdot d)$ trapping sets with size $d(\Delta)$. The probability that a trapping set in τ_Δ with this size exists is $P(d) = Pr\{X_d \geq 1\} \leq A_{d,\Delta \cdot d}$. Thus

$$Pr\{d(\Delta) < N\delta_{min}(\Delta)\} < \sum_{d=1}^{\lfloor N\delta_{min}(\Delta) \rfloor} A_{d,\Delta \cdot d}$$

In [27], the trapping set enumerator $A_{a,b}$ is given by

$$A_{a,b} = \sum_{\{d_i: v_i \in S\}} \sum_{\{d_k: v_k \in S_f\}} A(\mathbf{d}), \quad (10)$$

under the constraint $\sum_{\{d_i: v_i \in S\}} d_i = a$ and $\sum_{\{d_i: v_i \in S_f\}} d_i = b$, where S represents the variable nodes in the trapping set, S_f represents the set of degree-1 variable nodes that count the unsatisfied checks, and

$$A(\mathbf{d}) = \frac{\prod_{j=1}^{n_c} A_{c'_j}^{d_j}}{\prod_{i=1}^{n'_v} \binom{N}{d_i}^{q_{v_i}-1}}. \quad (11)$$

where $b = \Delta \cdot a$. For notations and details please refer to [27].

Manipulation of this inequality results the following upper bound on L .

$$L \leq \gamma_1 u + \gamma_2 \Delta d \quad (19)$$

The worst case upper bound on γ_1 and γ_2 are $\gamma_1 \leq \frac{t_2}{2} q_v^{max}$, and $\gamma_2 \leq \frac{t_2}{2}$ where $q_v^{max} = \max_i q_{v_i}$. If there are no degree-2 nodes then $\gamma_1 = 0$, and $\gamma_2 = 0$ and $L = 0$.

Since $u = d - L$ then $L \leq \frac{\gamma_1 + \gamma_2 \Delta}{1 + \gamma_1} d$. This implies $\Delta \leq \frac{1}{\gamma_2}$. Note that the second derivative of the function $E(d, L)$ with respect to L is strictly positive. Thus the function is convex, and boundary values at $L \leq \frac{\gamma_1 + \gamma_2 \Delta}{1 + \gamma_1} d$ are greater than the boundary value at $L = 0$ in the region of interest as mentioned before. Therefore $E(d, L) \leq E(d, \frac{\gamma_1 + \gamma_2 \Delta}{1 + \gamma_1} d)$. The weight vector enumerator can be upper bounded as

$$A(\mathbf{d}) \leq \nu e^{E(d, \frac{\gamma_1 + \gamma_2 \Delta}{1 + \gamma_1} d)} \quad (20)$$

where $E(d, \frac{\gamma_1 + \gamma_2 \Delta}{1 + \gamma_1} d) = \phi_1 d \ln \frac{d}{N} + (\phi_2 - (1 + \Delta))d$. We obtain

$$\phi_1 = \frac{1 - \Delta/\Delta^*}{2(1 + \gamma_1)}, \quad (21)$$

$$2\phi_2 = (2 + q_c^{max} \ln 2)(2 + \Delta - \frac{1 - \gamma_2 \Delta}{1 + \gamma_1}) + \frac{1 - \gamma_2 \Delta}{1 + \gamma_1} \ln \frac{1 - \gamma_2 \Delta}{1 + \gamma_1} - \Delta \ln \frac{\Delta}{n_c} + 2(1 + \Delta) \quad (22)$$

where $\Delta^* = \frac{1}{1 + \gamma_1 + \gamma_2}$.

$$A_{d,\Delta,d} = \sum_{\{d_i: v_i \in S\}} \sum_{\{d_k: v_k \in S_f\}} A(\mathbf{d}) \leq \nu |\{d_i: v_i \in S\}| |\{d_k: v_k \in S_f\}| \times e^{\phi_1 d \ln \frac{d}{N} + (\phi_2 - (1 + \Delta))d} \quad (23)$$

However $|\{d_i: v_i \in S\}| = \binom{d+n_v-1}{d} \leq e^{d+n_v-1}$, and $|\{d_k: v_k \in S_f\}| = \binom{\Delta d+n_c-1}{\Delta d} \leq e^{\Delta d+n_c-1}$.

Following a similar steps as in Appendix C of [27], for weight enumerators we can show that

$$A_{d,\Delta,d} \leq \nu e^{n_v+n_c-2} e^{NF(\delta)} \quad (24)$$

where $\delta = d/N$, and $F(\delta) = \phi_1 \delta \ln \delta + \phi_2 \delta$. Both $\phi_1 > 0$, and $\phi_2 > 0$ depend on Δ . To have $\phi_1 > 0$ we require that $0 < \Delta < \Delta^*$. This upper bound requires the number of degree-2 nodes to be less than checks and no loop to exist between degree-2 nodes and checks connected to them. This is the same condition for protographs to have linear minimum distance property. Define $\delta_o = e^{-\phi_2/\phi_1}$, then $F(\delta)$ is convex and negative for $0 \leq \delta \leq \delta_o$. Then if we choose $\delta_{min}(\Delta) = \delta_o - \frac{4}{N} \ln(N\delta_o/2)$ we can show that

$$\sum_{d=1}^{\lfloor N\delta_{min}(\Delta) \rfloor} A_{d,\Delta,d} < cN^{-2\phi_1}$$

where c is a constant independent of N . It is worth noting that when there is no degree-2 variable nodes in the protograph, $\Delta^* = 1$. This number decreases as we add more degree-2 nodes to a protograph. When $\Delta = 0$ then the trapping sets $(a, 0)$ represent the true codewords and our analysis coincides with ensemble codeword weight analysis.

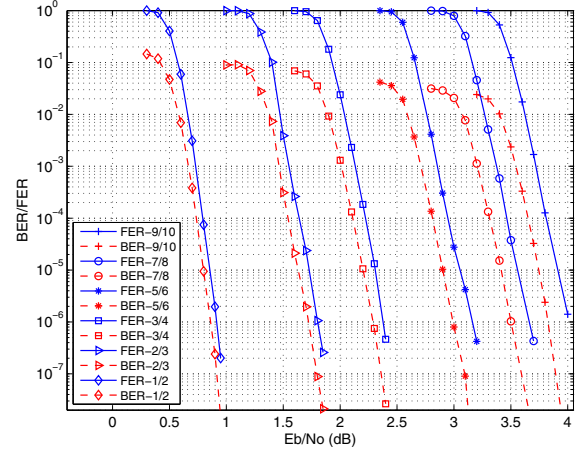


Fig. 5. Performance of nested codes of Section III.

VI. SIMULATION RESULTS

LDPC protograph codes are derived from our proposed protographs in two lifting steps. First, the protograph was lifted by a factor of 4 using the progressive edge growth (PEG) algorithm [28] in order to remove all parallel edges. Then based on desired information block-length, the resulting graph was lifted again using PEG to determine a circulant permutation of each edge class in order to maximize the girth of an overall bipartite graph.

In our nested codes of Section III, the parity check matrix for the lower-rate code can be obtained by removing certain columns from the parity check matrix of the higher rate code, and this produces economies in the design of the decoders. In fact, it is enough to design a decoder for the largest rate code (9/10). To decode the lower-rate codes, the missing coded bits are replaced by erasure at the decoder. In the same manner, the rate-compatible codes generated in Section IV only need the decoder of the lowest rate. Other higher-rate codes are decoded by replacing missing parity bits by erasure at the decoder.

The performances of our nested high-rate codes with rates 1/2, 2/3, 3/4, 5/6, 7/8 and 9/10 over the binary-input AWGN channel are shown in Fig. 5. All codes are simulated with the information block-length of 16k. The nested high-rate protographs given in Section III are first lifted by a factor of 4. This results in larger protographs that are lifted again with factors of 1365, 683, 455, 273, 195 and 152, respectively. For the rate-compatible protographs given in Section IV, we only need to lift the largest protograph (rate-15/46), by factor of 4 and then 273. Other high rate codes are obtained by removing the redundant parity bits. The performances of rate-compatible codes are shown in Figs. 6 and 7.

The resulting protograph codes have girth ≥ 8 except the protograph codes with rates 7/8, 9/10 and 15/46 where the girth is 6. The equivalent parity check matrices of all protograph codes have full rank except the rate-1/2 protograph plotted in Fig. 2 whose the parity check matrix has a row null space of dimension one. Thus, the rate loss is negligible. The maximum number of iterations is set to 200. No error floors were observed up to FER of 10^{-6} . The practical performance

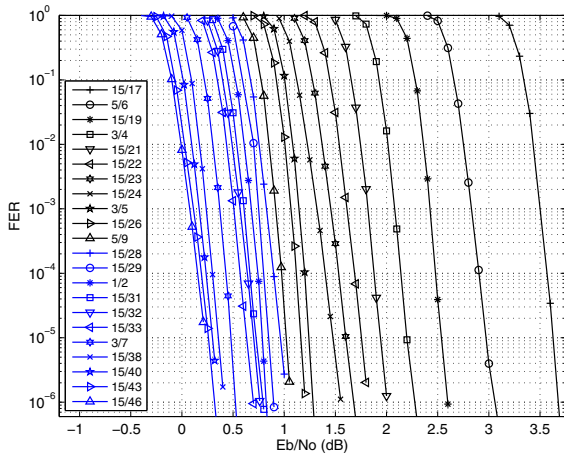


Fig. 6. FER performance of rate-compatible codes of Section IV.

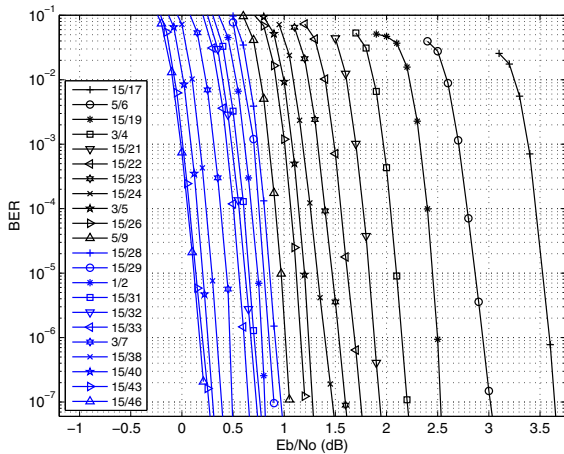


Fig. 7. BER performance of rate-compatible codes of Section IV.

at $FER = 10^{-6}$ shows a gap of less than 0.6 dB to threshold and 0.8 dB to capacity, thus the waterfall regions are steep.

Fig. 8 shows a comparison between our rate-1/2 code (protomatrix deduced from Equation (9)) and the rate-1/2 AR4JA code reported in CCSDS standard [29]. The new code outperforms the AR4JA code by approximately 0.2 dB. A comparison is also shown in the same figure with the rate-1/2 LDPC code reported in the DVB-S2 standard for video broadcasting [30]. The standard provides the code structure as a concatenation of BCH and LDPC codes, and reports the code performance with the information block-length of 32k. Even with a smaller block-length, the new code outperforms the standard code.

VII. CONCLUSION

This paper presents a simple approach for constructing rate-compatible LDPC codes based on protographs which perform within 0.2 dB of their capacity limits. The rate-compatible structure has the advantage of low encoding complexity and efficient decoding algorithm with one hardware implementation for all members of the code family.

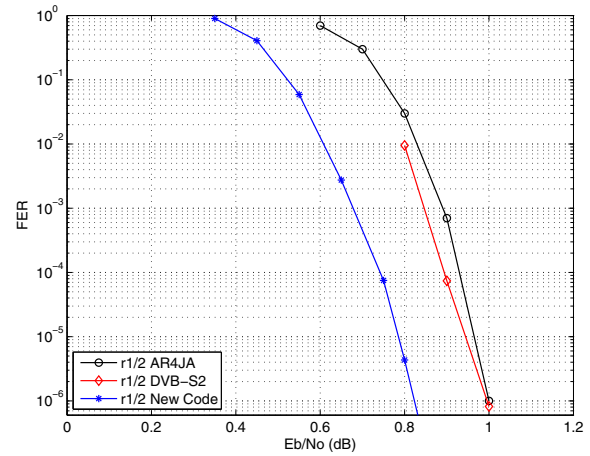


Fig. 8. Comparison of the rate-1/2 code produced in Section IV and AR4JA, both with information block-length of 16k, and DVB-S2 code with information block-length of 32k.

ACKNOWLEDGMENTS

This research was supported in part by the VOSP program from the Ministry of Education and Training, Vietnam, by the grant 009741-0084-2007 from THECB. This research was in part carried out at the Jet Propulsion Laboratory, California Institute of Technology, under a contract with NASA.

REFERENCES

- [1] J. Hagenauer, "Rate-compatible punctured convolutional codes (RCP codes) and their applications," *IEEE Trans. Commun.*, vol. 36, no. 4, pp. 389–400, Apr. 1988.
- [2] J. Ha, J. Kim, and S. McLaughlin, "Rate-compatible puncturing of low-density parity-check codes," *IEEE Inf. Theory*, vol. 50, no. 11, pp. 2824–2836, Nov. 2004.
- [3] H. Pishro-Nik and F. Fekri, "Results on punctured LDPC codes," in *Proc. 2004 IEEE Inf. Theory Workshop*, pp. 215–219.
- [4] T. Tian and C. R. Jones, "Construction of rate-compatible LDPC codes utilizing information shortening and parity puncturing," *EURASIP J. Wirel. Commun. Netw.*, vol. 2005, no. 5, pp. 789–795, 2005.
- [5] M. R. Yazdani and A. H. Banihashemi, "On construction of rate-compatible low-density parity-check codes," *IEEE Commun. Lett.*, vol. 8, no. 3, pp. 159–161, Mar. 2004.
- [6] J. Ha, J. Kim, D. Klinc, and S. McLaughlin, "Rate-compatible punctured low-density parity-check codes with short block lengths," *IEEE Trans. Inf. Theory*, vol. 52, no. 2, pp. 728–738, Feb. 2006.
- [7] H. Y. Park, J. W. Kang, K. S. Kim, and K. C. Whang, "Efficient puncturing method for rate-compatible low-density parity-check codes," *IEEE Trans. Wireless Commun.*, vol. 6, no. 11, pp. 3914–3919, Nov. 2007.
- [8] B. Vellambi and F. Fekri, "Finite-length rate-compatible LDPC codes: a novel puncturing scheme," *IEEE Trans. Commun.*, vol. 57, no. 2, pp. 297–301, Feb. 2009.
- [9] N. Jacobsen and R. Soni, "Design of rate-compatible irregular LDPC codes based on edge growth and parity splitting," in *Proc. 2007 IEEE Veh. Technol. Conf. – Fall*, pp. 1052–1056.
- [10] C.-H. Hsu and A. Anastasopoulos, "Capacity achieving LDPC codes through puncturing," *IEEE Trans. Inf. Theory*, vol. 54, no. 10, pp. 4698–4706, Oct. 2008.
- [11] 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.
- [12] R. G. Gallager, "Low-density parity-check codes," Ph.D. dissertation, MIT, Cambridge, Massachusetts, 1963.
- [13] J. Thorpe, "Low-density parity-check (LDPC) codes constructed from protographs," IPN Progress Report 42-154, Aug. 2003.
- [14] J. Li and K. R. Narayanan, "Rate-compatible low density parity check codes for capacity-approaching ARQ schemes in packet data communications," in *Proc. 2002 CIIT*.

- [15] S. Dolinar, "A rate-compatible family of protograph-based LDPC codes built by expurgation and lengthening," in *Proc. 2005 IEEE ISIT*, pp. 1627–1631.
- [16] J. Thorpe, K. Andrews, and S. Dolinar, "Methodologies for designing LDPC codes using protographs and circulants," in *Proc. 2004 IEEE ISIT*, pp. 236–236.
- [17] D. Divsalar, S. Dolinar, and C. R. Jones, "Low-rate LDPC codes with simple protograph structure," in *Proc. 2005 IEEE ISIT*, pp. 1622–1626.
- [18] M. El-Khamy, J. Hou, and N. Bhushan, "Design of rate-compatible structured LDPC codes for hybrid ARQ applications," *IEEE J. Sel. Areas. Commun.*, vol. 27, no. 6, pp. 965–973, Aug. 2009.
- [19] J. Kim, A. Ramamoorthy, and S. McLaughlin, "The design of efficiently-encodable rate-compatible LDPC codes," *IEEE Trans. Commun.*, vol. 57, no. 2, pp. 365–375, Feb. 2009.
- [20] C. Shi and A. Ramamoorthy, "Design and analysis of E2RC codes," *IEEE J. Sel. Areas. Commun.*, vol. 27, no. 6, pp. 889–898, Aug. 2009.
- [21] A. Abbasfar, D. Divsalar, and K. Yao, "Accumulate-repeat-accumulate codes," *IEEE Trans. Commun.*, vol. 55, no. 4, pp. 692–702, Apr. 2007.
- [22] T. Richardson and R. Urbanke, "Multi-edge type LDPC codes," EPFL, Tech. Rep., 2004.
- [23] T. J. Richardson, M. A. Shokrollahi, and R. L. Urbanke, "Design of capacity-approaching irregular low-density parity-check codes," *IEEE Trans. Inf. Theory*, vol. 47, no. 2, pp. 619–637, Feb. 2001.
- [24] S. ten Brink, "Convergence behavior of iteratively decoded parallel concatenated codes," *IEEE Trans. Commun.*, vol. 49, no. 10, pp. 1727–1737, Oct. 2001.
- [25] G. Liva and M. Chiani, "Protograph LDPC codes design based on EXIT analysis," in *Proc. 2007 IEEE GLOBECOM*, pp. 3250–3254.
- [26] S. Abu-Surra, D. Divsalar, and W. Ryan, "On the existence of typical minimum distance for protograph-based LDPC codes," in *Proc. 2010 Inf. Theory Applications Workshop*, pp. 1–7.
- [27] —, "Enumerators for protograph-based ensembles of LDPC and generalized LDPC codes," *IEEE Trans. Inf. Theory*, vol. 57, no. 2, pp. 8585–886, Feb. 2011.
- [28] X.-Y. Hu, E. Eleftheriou, and D.-M. Arnold, "Regular and irregular progressive edge-growth Tanner graphs," *IEEE Trans. Inf. Theory*, vol. 51, pp. 386–398, 2003.
- [29] CCSDS, "Low density parity check codes for use in near-earth and deep space applications," CCSDS 131.1-O-2. Orange Book, Issue 2, Sep. 2007.
- [30] ETSI TR 102 376 v1.1.1, "Digital video broadcasting (DVB) user guidelines for the second generation system for broadcasting, interactive services, news gathering and other broadband satellite applications (DVB-S2)," 2005.



Thuy van Nguyen (S'10) received the B.S. degree in electronics and telecommunications from Hanoi University of Technology, Hanoi, Vietnam, in 1999 and the M.S. degree in electrical engineering from New Mexico State University in 2005. Currently, he is working towards the Ph.D. degree in electrical engineering at The University of Texas at Dallas. His research interests include information theory, signal processing, and channel coding with applications in wireless communications and data storage.



Aria Nosratinia (S'87-M'97-SM'04-F'10) is the Jonsson Distinguished Professor of Engineering at the University of Texas at Dallas. He received his Ph.D. in electrical and computer engineering from the University of Illinois at Urbana-Champaign in 1996. He has held visiting appointments at Princeton University, Rice University, and UCLA. His interests lie in the broad areas of information theory and signal processing, with applications in wireless communications. He was the secretary for the IEEE Information Theory Society in 2010–2011 and was the treasurer for ISIT 2010 in Austin, Texas. He has served as editor for the IEEE TRANSACTIONS ON INFORMATION THEORY, the IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS, IEEE SIGNAL PROCESSING LETTERS, the IEEE TRANSACTIONS ON IMAGE PROCESSING, and *IEEE Wireless Communications Magazine*. He has been the recipient of the National Science Foundation CAREER award, and is a fellow of IEEE.



Dariush Divsalar (S'76-M'78-SM'90-F'97) received the Ph.D. degree in electrical engineering from the University of California, Los Angeles (UCLA), in 1978. Since then, he has been with the Jet Propulsion Laboratory (JPL), California Institute of Technology (Caltech), Pasadena, where he is a Principal Scientist. At JPL, he has been involved with developing state-of-the-art technology for advanced deep-space communications systems and future NASA space exploration. Since 1986, he has taught graduate courses in communications and coding at UCLA and Caltech. He has published more than 200 papers, co-authored a book entitled *An Introduction to Trellis Coded Modulation with Applications*, contributed to two other books, and holds 20 U.S. patents in the above areas.

Dr. Divsalar was a co-recipient of the 1986 paper award of the IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY. He was also a co-recipient of the joint paper award of the IEEE Information Theory and IEEE Communication Theory Societies in 2008. The IEEE Communications Society has selected one of his papers for inclusion in a book entitled *THE BEST OF THE BEST: FIFTY YEARS OF COMMUNICATIONS AND NETWORKING RESEARCH*, containing the best 56 key research papers ever published in the Society's 50-year history. He has received over 40 NASA Tech Brief awards and a NASA Exceptional Engineering Achievement Medal in 1996. He served as an Area Editor for the IEEE TRANSACTIONS ON COMMUNICATIONS from 1989 to 1996.