

# Protograph LDPC Codes with Node Degrees at Least 3

Dariush Divsalar  
Jet Propulsion Laboratory  
California Institute of Technology  
4800 Oak Grove Drive  
Pasadena, CA 91109-8099  
Email: Dariush.Divsalar@jpl.nasa.gov

Christopher Jones  
Jet Propulsion Laboratory  
California Institute of Technology  
4800 Oak Grove Drive  
Pasadena, CA 91109-8099  
Email: christop@jpl.nasa.gov

**Abstract**—In this paper we present protograph codes with a small number of degree-3 nodes and one high degree node. The iterative decoding threshold for proposed rate 1/2 codes are lower, by about 0.2 dB, than the best known irregular LDPC codes with degree at least 3. The main motivation is to gain linear minimum distance to achieve low error floor. Also to construct rate-compatible protograph-based LDPC codes for fixed block length that simultaneously achieves low iterative decoding threshold and linear minimum distance. We start with a rate 1/2 protograph LDPC code with degree-3 nodes and one high degree node. Higher rate codes are obtained by connecting check nodes with degree-2 non-transmitted nodes. This is equivalent to constraint combining in the protograph. The condition where all constraints are combined corresponds to the highest rate code. This constraint must be connected to nodes of degree at least three for the graph to have linear minimum distance. Thus having node degree at least 3 for rate 1/2 guarantees linear minimum distance property to be preserved for higher rates. Through examples we show that the iterative decoding threshold as low as 0.544 dB can be achieved for small protographs with node degrees at least three. A family of low- to high-rate codes with minimum distance linearly increasing in block size and with capacity-approaching performance thresholds is presented. FPGA simulation results for a few example codes show that the proposed codes perform as predicted.

## I. INTRODUCTION

Low-density parity-check (LDPC) codes were proposed by Gallager [1] in 1962. After introduction of turbo codes by Berrou et al [2] in 1993, researchers revisited LDPC codes, and extended the work of Gallager using the code graphs introduced by Tanner [3] in 1981. After 1993 there have been many contributions to the design and analysis of LDPC codes; see for example [10], [12], [4], [13], [14], [15], and references there. Recently a flurry of work has been conducted on the design of LDPC codes with imposed sub-structures, starting with the introduction of multi-edge-type codes in [9] and [11].

Fixed block length codes are desirable in scenarios where a framing constraint is imposed on the physical layer. This occurs, for instance, when orthogonal frequency division modulation is used. Prior work on fixed block length LDPC codes has been described in [23] where the authors provide a technique for combining rows of a rate 1/2 parity check matrix to form higher rate codes. In this paper we use degree-2 punctured nodes to implement row combining (thereby

achieving rate compatibility), but use no transmitted degree-2 nodes. By avoiding transmitted degree-2 nodes we obtain a code structure whose minimum distance grows linearly with blocksize  $n$ . Note that limiting code design to the use of degree-3 and higher variable nodes is a sufficient, but not necessary condition for minimum distance to grow linearly with  $n$  [18]. In this paper we demonstrate that small protograph based codes can in fact achieve competitively low thresholds without the use of degree-2 variable nodes and without degree-1 pre-coding in conjunction with the puncturing of a high degree node [25].

## II. PROTOGRAPH LDPC CODES

To aid in implementation of high-speed decoding, it is advantageous for an LDPC code to be constructed from a protograph [7] or projected graph [8]. A protograph is a Tanner graph with a relatively small number of nodes. A “copy-and-permute” operation [7] can be applied to the protograph to obtain larger derived graphs of various sizes. This operation consists of first making  $N$  copies of the protograph, and then permuting the endpoints of each edge among the  $N$  variable and  $N$  check nodes connected to the set of  $N$  edges copied from the same edge in the protograph. The derived graph is the graph of a code  $N$  times as large as the code corresponding to the protograph, with the same rate and the same distribution of variable and check node degrees. LDPC codes with protograph structure are a subclass of multi-edge-type LDPC codes.

As an example for protograph based LDPC codes we consider the rate-1/3 Repeat-Accumulate (RA) code depicted in Fig. 1(a). For this code the minimum  $E_b/N_0$  threshold with iterative decoding is 0.502 dB. This code has a protograph representation shown in Fig. 1(b), as long as the interleaver  $\pi$  is chosen to be decomposable into permutations along each edge of the protograph. The iterative decoding  $E_b/N_0$  threshold is unchanged despite the additional constraint imposed by the protograph. The protograph consists of 4 variable nodes and 3 check nodes, connected by 9 edges. Three variable nodes are connected to the channel (transmitted nodes) and are shown as dark filled circles. One variable node is not connected to the channel (i.e., it is punctured) and is depicted by a blank circle. The three check nodes are depicted by circles with a plus sign inside.

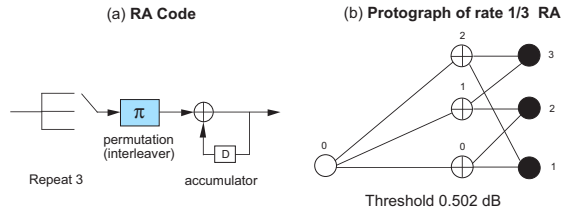


Fig. 1. (a) A rate-1/3 RA code with repetition 3, and (b) its corresponding protograph.

Repeat-Accumulate (RA) [5], Irregular Repeat-Accumulate (IRA) [6], and recently proposed Accumulate-Repeat-Accumulate (ARA) [16] codes, with suitable definitions of their interleavers, all have simple protograph representations. These codes provide fairly low iterative decoding thresholds but have sublinear minimum distance. However for certain applications linear minimum distance is required for low error floor performance.

### III. RECIPROCAL CHANNEL APPROXIMATION IN PROTOGRAPHS

Computation of iterative decoding thresholds for the protographs in this paper follows a fast and accurate approximation to density evolution originally proposed in [22]. Less than 0.005 dB deviations from true density evolution thresholds have been observed by the application of this approximation to protographs in BI-AWGN channels.

The reciprocal channel approximation (RCA) makes use of a single real-valued parameter, in this case signal-to-noise ratio (SNR)  $s$ , as a stand-in for full density evolution. For every value of  $s$ , a reciprocal of SNR,  $r$ , is defined such that  $C(s) + C(r) = 1$ , where  $C(x)$  denotes the capacity of the binary-input AWGN channel with SNR  $x$ . In the reciprocal channel approximation, the parameter  $s$  is additive at variable nodes, and the reciprocal parameter  $r$  is additive at check nodes. Chung's self-inverting reciprocal energy function,  $R(x) = C^{-1}(1 - C(x))$ , transforms between the parameters  $s$  and  $r$ , namely  $r = R(s)$  and  $s = R(r)$ .

To apply the RCA technique to a protograph we first identify all transmitted variable nodes and select a target channel SNR  $s_{chan}$ . As shown in Fig. 2 messages  $\vec{s}_e$  are passed along edges leaving variable nodes ( $\vec{s}_e = s_{chan}$  from transmitted nodes and  $\vec{s}_e = 0$  from punctured nodes). The transformation  $R(\vec{s}_e)$  is applied and an extrinsic return message,  $\vec{r}_e$ , is determined by computing the sum of all incoming messages save the one along edge  $e$ . Transformation  $R(\vec{r}_e)$  is then reapplied to produce  $\vec{s}_e$ . The process continues and a threshold is determined by the smallest value of  $s_{chan}$  for which unbounded growth of all messages  $\vec{s}_e$  can be achieved.

Motivation for applying RCA to the BI-AWGN channel most likely derived from the fact that a similar reciprocal channel definition yields exact density evolution results [22] when applied to the binary erasure channel (BEC). In the case of a BEC with erasure probability  $\epsilon$  and capacity  $C = 1 - \epsilon$ ,

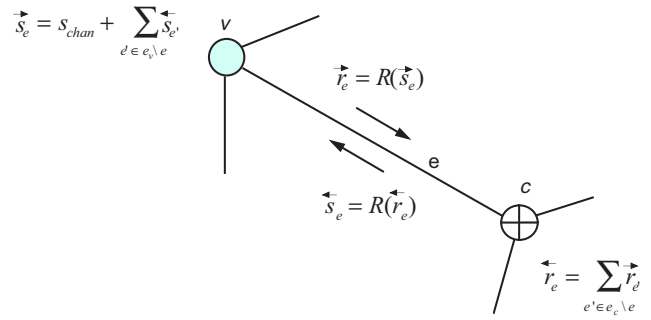


Fig. 2. The reciprocal channel approximation in use on a protograph.

a parameter  $s = \text{snr}$  is additive at variable nodes, a reciprocal parameter  $r = \text{snr}^{-1}$  is additive at check nodes, and  $s$  and  $r$  are related by  $C(s) + C(r) = 1$ .

### IV. PROTOGRAPH OF REGULAR LDPC CODES

Classic regular LDPC codes, in addition to simplicity, have low error floors. However, their iterative decoding thresholds are high. For example the (3,6) regular LDPC codes have an iterative decoding threshold of 1.102 dB while their ensemble asymptotic minimum distance grows like  $0.023n$  as  $n$  goes to infinity. For comparison the asymptotic minimum distance of random codes grows as  $0.11n$ . We express the normalized logarithmic asymptotic weight distribution of a code as  $r(\gamma) = \frac{\ln(A_d)}{n}$  where  $d$  is Hamming weight,  $\gamma = \frac{d}{n}$ , and  $A_d$  is the ensemble weight distribution. If  $r(\gamma)$  starts out negative near  $\gamma = 0$  and has a first zero crossing at  $\gamma = \gamma_{min} > 0$ , then the average minimum distance of the code ensemble is  $d_{min} = n \gamma_{min}$ , which grows linearly with  $n$  at the rate  $\gamma_{min}$ . This growth rate  $\gamma_{min}$  is a characteristic of the specific protograph from which the LDPC code ensemble is constructed. Methods to compute the asymptotic weight enumerators for LDPC codes with protograph structure are presented in [19] and [20].

Fig. 3 compares the asymptotic weight distribution of (3,6) regular LDPC codes to that of rate 1/2 random codes.

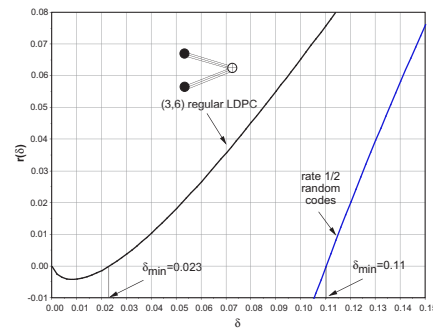


Fig. 3. Asymptotic weight distributions and zero crossings for (3,6) regular LDPC and rate 1/2 random codes.

### V. PROTOGRAPHS LDPC CODES WITH NODE DEGREES AT LEAST 3

We wish to construct rate-compatible protograph-based LDPC codes for fixed block lengths that simultaneously

achieve low iterative decoding thresholds and linear minimum distance ( $\hat{c}_{min} > 0$ ) such that error floors may be effectively suppressed. We start with a rate 1/2 protograph LDPC code with degree-3 nodes and one high degree node (which serves to lower threshold). Higher rate codes are obtained by connecting check nodes with degree-2 non-transmitted nodes. This is equivalent to constraint combining in the protograph. The condition where all constraints are combined corresponds to the highest rate code. This constraint must be connected to nodes of degree at least three for the graph to have linear minimum distance growth. Thus having node degrees at least 3 for rate 1/2 guarantees linear minimum distance property to be preserved for higher rates. In particular the highest code rate protograph after combining all checks will have only one check. Thus if for rate 1/2 protograph we allow degree 2 nodes, then the highest code rate protograph corresponds to an Irregular Repeat Accumulate code (IRA). We know that IRA codes do not have the linear minimum distance property. This is the main reason that the rate 1/2 protograph should have node degrees of at least 3 for rate-compatible structure. Otherwise we know that if the number of degree 2 nodes is less than the number of checks, linear minimum distance is preserved. This can be easily proved using the results in [20].

It has been shown that optimized rate 1/2 degree distributions for irregular unstructured LDPC codes with node degrees at least 3, and maximum degree 25, can achieve iterative decoding thresholds of 0.73 dB. We have computed this result using [17]. The output of the computer search is shown below:

```

Rate= 0.500
Gap to Shannon limit= 0.53238dB
Min left degree= 3
Max left degree= 17
Avgrd= 8.5000000
Variable node edge distribution (node perspective):
3 0.6419580000 (0.9062939465435)
16 0.2893450000 (0.07659134883901)
17 0.0686965000 (0.017114704617489)
Check node edge distribution (node perspective):
8 0.5 (0.52941176470588)
9 0.5 (0.47058823529412)

```

We next show that it is in fact possible to design protograph based rate 1/2 LDPC codes with degrees at least 3 and maximum degree not more than 20, with iterative decoding threshold less than 0.73 dB. We start with a rate 1/2, eight node protograph with variable node degrees 3 as shown in Fig. 4(a). As expected the iterative decoding threshold for this code is 1.102 dB. We next change one of the nodes to degree 16 (in similar fashion to the degree 16 node used in the optimum irregular unstructured degree distribution) as shown in Fig. 4(b). The iterative decoding threshold for this code is 0.972 dB. Note that very little improvement is obtained by using one high degree node.

Now change the connections of variable node to check nodes asymmetrically. After few hand selected searches we obtain

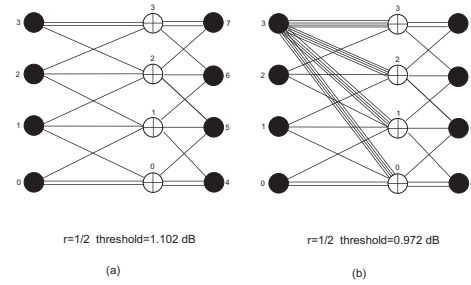


Fig. 4. (a) an eight node rate 1/2 protograph with node degrees 3, (b) the same protograph with one node of degree 16.

the protograph shown in Fig. 5, which has threshold 0.618 dB.

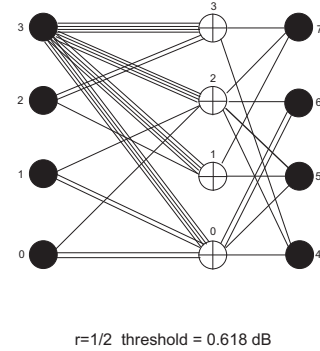


Fig. 5. A protograph with 7 nodes of degree 3, and one node of degree 16.

Note that it is possible to obtain an even lower threshold (0.544 dB) if the number of nodes in the protograph is increased to 12 and highest node degree is set to 18 as shown in Fig. 6

## VI. CONSTRUCTION OF HIGHER CODE RATES

Higher rate codes are obtained by connecting check nodes with degree-2 non-transmitted nodes. This is equivalent to constraint combining in the protograph. The condition where all constraints are combined corresponds to the highest rate code. Fig. 7 shows such construction for the 8 node protograph in Fig. 5. Thresholds computed via RCA for the rate 1/2, 5/8, 3/4, and, 7/8 are 0.618 (gap to capacity 0.43 db), 1.296 (gap to capacity 0.48 db), 1.928 (gap to capacity 0.30 db), 3.052 dB (gap to capacity 0.21 db) respectively. The protograph in Fig. 7 can be used for rates 1/2, 5/8, 3/4, and 7/8 if the nodes 8, 9, and 10 are properly set to "0" bit (equivalent to not having degree-2 node connections), or "X", where "X" represent no bit assignment to the node (not transmitted node). See table below for node assignments. At decoder the corresponding nodes to "0" bits are assigned highly reliable values, and to nodes "X" zero reliability values.

| Code rate | Node 8 | Node 9 | Node 10 |
|-----------|--------|--------|---------|
| 1/2       | 0      | 0      | 0       |
| 5/8       | X      | 0      | 0       |
| 3/4       | X      | X      | 0       |
| 7/8       | X      | X      | X       |

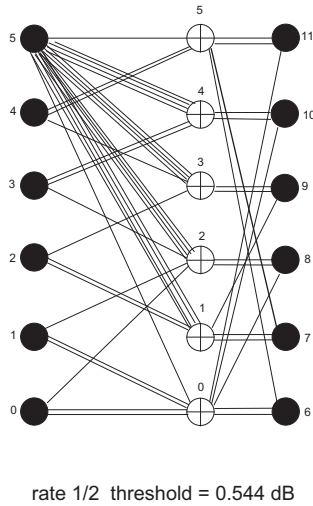


Fig. 6. A protograph with 11 nodes of degree 3, and one node of degree 18.

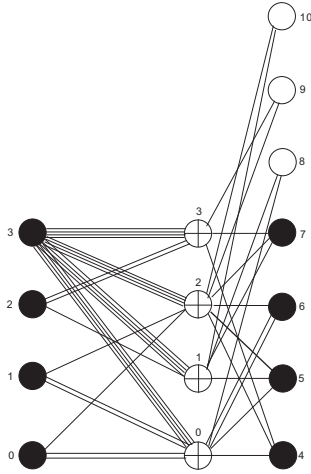


Fig. 7. Construction of rates 5/8, 3/4, and 7/8 using an 8-node rate 1/2 protograph.

## VII. CODE CONSTRUCTION AND SIMULATION RESULTS

The protograph associated with the rate 7/8 code (all nodes and edges in Fig. 7) was lifted by a factor of 4 using progressive edge growth [24] to remove all multiple parallel edges. The resulting graph was then lifted using the ACE algorithm [21] to find phases associated with circulants of size 181. These circulants are described in the transposed  $H$  matrix given in Table I. Note that in the table the first twelve rows correspond to nodes 8, 9, 10 (lifted by 4) from Fig. 7. Intermediate rows correspond to nodes 0, 1, 2, 4, 5, 6, 7 and the last four rows in Table I represent node 3. Each entry denoted by  $x^i$  represent a circulant permutation where  $i$  represents the amount of right circular shift of non-zero elements in the identity matrix.

Fig. 8 shows bit (solid curves) and frame (dashed curves) error rate FPGA simulation results computed by JPL's Universal Decoder for Sparse Codes. The lowest three rates exhibit no error flooring at frame error rates of  $10^{-6}$  and higher. While

the rate 7/8 code does display error events near the  $10^{-6}$  level due to error trapping sets. Note that the circulants for rates 1/2, 2/3, and 3/4 are the same as those for the rate 7/8 code with the exception of removal of circulants (and edges) associated with node 10 in Fig. 7. In the context of a decoder implementation, instead of being removed, nodes 8, 9, 10 could be assigned highly reliable values to "0" bits to obviate contributions to their respective check nodes as it was discussed in section VI.

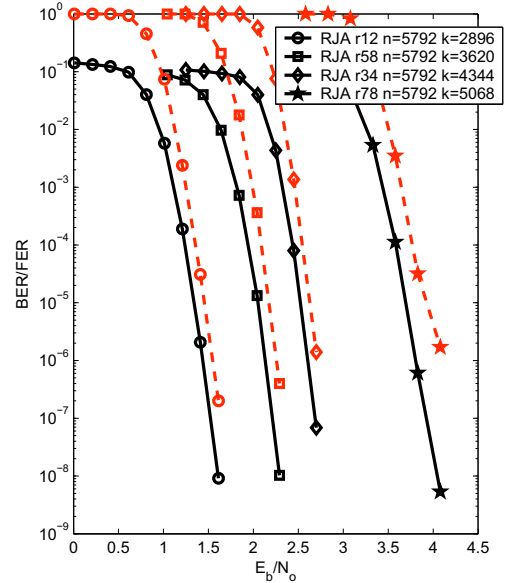


Fig. 8. Performance of  $n=5792$  rate 1/2, 5/8, 3/4, 7/8 family of codes.

## VIII. CONCLUSION

In this paper we introduced a new construction technique for designing ensembles of structured codes with constant block length. These codes exhibit both good threshold performance and a minimum distance that for an average instance from the ensemble increases linearly with blocklength.

## ACKNOWLEDGMENT

This research was carried out at the Jet Propulsion Laboratory, California Institute of Technology, under contract with the National Aeronautics and Space Administration.

## REFERENCES

- [1] R. G. Gallager, *Low Density Parity Check Codes*. Cambridge, MA: MIT Press, 1963.
- [2] C. Berrou and A. Glavieux, "Near optimum error correcting coding and decoding: Turbo-codes," *IEEE Trans. Commun.*, Vol. 44, pp. 1261-1271, October 1996.
- [3] M. R. Tanner, "A recursive approach to low complexity codes," *IEEE Trans. Inform. Theory*, vol. 27, pp. 533-547, 1981.
- [4] T. Richardson, A. Shokrollahi, and R. Urbanke, "Design of capacity-approaching irregular low-density parity-check codes," *IEEE Trans. Inform. Theory*, vol. 47, pp. 619-637, 2001.
- [5] D. Divsalar, H. Jin, and R. McEliece, "Coding theorems for Turbo-like codes," in *Proceedings of the 1998 Allerton Conference*, 1998.
- [6] H. Jin, A. Khandekar, and R. McEliece, "Irregular repeat-accumulate codes," in *Proc. 2nd International Symposium on Turbo Codes*, 2000.
- [7] Jeremy Thorpe, "Low Density Parity Check (LDPC) Codes Constructed from Protographs," JPL INP Progress Report 42-154, August 15, 2003.

TABLE I

PARITY MATRIX :  $n = 5792$  RATE =  $\{1/2, 5/8, 3/4, 7/8\}$ . CIRCULANT SIZE = 181.

$$H^t = \begin{pmatrix} x^{118} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & x^{178} & 0 & 0 & 0 \\ 0 & x^{65} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & x^{17} & 0 & 0 \\ 0 & 0 & x^{59} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & x^{64} & 0 \\ 0 & 0 & 0 & x^{141} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & x^{98} \\ 0 & 0 & 0 & 0 & 0 & x^{118} & 0 & 0 & x^{120} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & x^{68} & 0 & 0 & x^{165} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & x^{39} & 0 & 0 & x^{65} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & x^{164} & 0 & 0 & 0 & 0 & 0 & 0 & x^{144} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & x^{165} & 0 & 0 & 0 & 0 & 0 & 0 & x^{40} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & x^{164} & 0 & 0 & 0 & 0 & 0 & 0 & x^{123} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & x^{137} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & x^{127} \\ 0 & 0 & 0 & 0 & x^{78} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & x^{154} \\ x^{96} & 0 & x^4 & 0 & 0 & 0 & 0 & 0 & 0 & x^{87} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & x^{67} & 0 & x^{37} & 0 & 0 & 0 & 0 & 0 & x^{39} & 0 & 0 & 0 & 0 & 0 & 0 \\ x^{82} & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & x^{39} & 0 & 0 & 0 & 0 & 0 \\ 0 & x^{114} & 0 & x^{31} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & x^{76} & 0 & 0 & 0 & 0 \\ x^{79} & x^{179} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & x^{29} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & x^{160} & x^{172} & 0 & 0 & 0 & 0 & 0 & 0 & x & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & x^{70} & x^{108} & 0 & 0 & 0 & 0 & 0 & 0 & x^{120} & 0 & 0 & 0 & 0 & 0 \\ x^{115} & 0 & 0 & x^{128} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & x^{83} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & x^{33} & 0 & 0 & 0 & 0 & 0 & x^{176} & 0 & 0 & x^{43} \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & x^{20} & 0 & 0 & 0 & 0 & x^{87} & x^3 & 0 & 0 \\ 0 & 0 & 0 & 0 & x^{149} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & x^{52} & x^{116} & 0 \\ 0 & 0 & 0 & 0 & 0 & x^{109} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & x^{140} & x^{162} \\ 0 & 0 & x^{79} & 0 & 0 & 0 & 0 & 0 & x^6 & 0 & 0 & 0 & x^{144} & 0 & 0 & 0 \\ 0 & 0 & 0 & x^{123} & 0 & 0 & 0 & 0 & 0 & x^{131} & 0 & 0 & 0 & x^{45} & 0 & 0 \\ x^{109} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & x^{72} & 0 & 0 & 0 & 0 & x^{71} \\ 0 & x^{97} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & x^{58} & 0 & 0 & 0 & x^{101} \\ x^{109} & 0 & 0 & 0 & x^{83} & 0 & 0 & 0 & x^{158} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & x^{65} & 0 & 0 & 0 & x^{165} & 0 & 0 & 0 & x^{125} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & x^{137} & 0 & 0 & 0 & x^{147} & 0 & 0 & 0 & x^{93} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & x^{103} & 0 & 0 & 0 & x^{76} & 0 & 0 & 0 & x^{112} & 0 & 0 & 0 & 0 \\ x^{113} & x^{53} & 0 & 0 & 0 & 0 & 0 & 0 & x^{20} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & x^{65} & x^{128} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & x^{70} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & x^{56} & x^{56} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & x^{46} & 0 & 0 & 0 & 0 \\ x^{29} & 0 & 0 & x^{64} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & x^{88} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & x^{54} & 0 & 0 & 0 & x^{83} & 0 & 0 & 0 & x^{123} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & x^{60} & 0 & 0 & 0 & 0 & x^{159} & 0 & 0 & 0 & x^{170} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & x^{167} & 0 & 0 & 0 & x^{67} & 0 & 0 & 0 & 0 & x^{105} \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & x^{21} & 0 & 0 & 0 & x^{125} & 0 & 0 & 0 & x^{106} \\ 1 & x^{112} & x^{32} & x^{122} & x^{55} & x^{174} & x^{51} & x^{26} & x^{117} & x^{143} & x^{33} & x^{13} & x^{97} & x^{84} & x^{111} & x^{159} \\ x^{150} & x^{44} & x^7 & x^{90} & x^{57} & x^{148} & x^{57} & x^{22} & x^{73} & x^{60} & x^{46} & x^{31} & x^{84} & x^{122} & x^{75} & x^{99} \\ x^{49} & x^{154} & x^{180} & x^{68} & x^{60} & x^{41} & x^9 & x^{22} & x^{178} & x^{78} & x^{16} & x^{94} & x^{144} & x^{151} & x^{115} & x^{160} \\ x^{80} & x^{153} & x^{37} & x^8 & x^{27} & x^{42} & x^6 & x^{59} & x^{112} & x^{135} & x^{172} & x^{66} & x^{129} & x^{63} & x^{150} & x^{110} \end{pmatrix}$$

- [8] Richardson, et al., "Methods and apparatus for decoding LDPC codes," United States Patent 6,633,856, October 14, 2003.
- [9] T. Richardson, "Multi-Edge Type LDPC Codes," presented at the Workshop honoring Prof. Bob McEliece on his 60th birthday, California Institute of Technology, Pasadena, California, May 24-25, 2002.
- [10] D.J.C. MacKay, R.M. Neal, "Near Shannon limit performance of low density parity check codes," Electronics Letters, Vol. 32, Issue 18, 29 Aug. 1996, Page(s) 1645.
- [11] T. Richardson and R. Urbanke, "The Renaissance of Gallager's Low-Density Parity-Check Codes," IEEE Communications Magazine, pages 126-131, August 2003.
- [12] M. Luby, M. Mitzenmacher, A. Shokrollahi, and D. Spielman, "Analysis of low density codes and improved designs using irregular graphs," IEEE Trans. Inform. Theory, vol. 47, pp. 585-598, 2001.
- [13] T. Richardson and R. Urbanke, "The capacity of low-density parity check codes under message-passing decoding," IEEE Trans. Inform. Theory, vol. 47, pp. 599-618, 2001.
- [14] Y. Kou, S. Lin, and M.P.C. Fossorier, "Low-density parity-check codes based on finite geometries: a rediscovery and new results," IEEE Transactions on Information Theory, vol. 47, Nov. 2001, pp. 2711-2736.
- [15] F.R. Kschischang, "Codes defined on graphs," IEEE Communications Magazine, Vol. 41, Issue 8, Aug. 2003, Pages 118-125.
- [16] A. Abbasfar, D. Divsalar, and K. Yao, "Accumulate Repeat Accumulate Codes," (abstract) IEEE ISIT 2004, Chicago, IL, June 27-July 2, and IEEE Globecom 2004, Dallas, Texas, 29 November - 3 December, 2004.
- [17] EPFL "Degree distribution optimizer for irregular unstructured LDPC codes (LdpcOpt)," (<http://lthcwww.epfl.ch/research/lldpcopt/>)
- [18] D. Divsalar, C. Jones, S. Dolinar, J. Thorpe, "Protograph Based LDPC Codes with Minimum Distance Linearly Growing with Block Size," IEEE Globecom 2005.
- [19] S.L. Fogle, Robert McEliece, Jeremy Thorpe "Enumerators for Protograph Ensembles of LDPC Codes," IEEE ISIT 2005, Adelaide, Australia 4-9 September, 2005.
- [20] D. Divsalar, "Finite Length Weight Enumerators for Protograph Based LDPC Code Ensembles," IEEE Communication Theory Workshop, Park City, Utah, June 12-15, 2005.
- [21] Tian, T.; Jones, C.; Villasenor, J.; and Wesel, R. D.; "Characterization and selective avoidance of cycles in irregular LDPC code construction," IEEE Transactions on Communications, Aug. 2004, pp. 1242-1247.
- [22] Chung S. Y., "On the Construction of Some Capacity-Approaching Coding Schemes," Ph.D. dissertation, Massachusetts Institute of Technology, Cambridge, Massachusetts, Sept., 2000.
- [23] Vila Casado, A.; Weng, W.; Wesel, R.; "Multiple rate low-density parity-check codes with constant blocklength," Proceedings, 38th Asilomar Conference, 2004. 7-10 Nov. 2004 Page(s):2010-2014
- [24] Xiao-Yu Hu; Eleftheriou, E.; Arnold, D.-M.; "Progressive edge-growth Tanner graphs," Global Telecommunications Conference, 2001. GLOBECOM '01. IEEE Volume 2, 25-29 Nov. 2001 Page(s):995 - 1001 vol.2
- [25] D. Divsalar, S. Dolinar, J. Thorpe, C. Jones, "Constructing LDPC Codes from Simple Loop-Free Encoding Modules," International Conference on Communications (ICC) 2005. Seoul, Korea.