

Constructions for Irregular Repeat-Accumulate Codes

Sarah J. Johnson and Steven R. Weller

School of Electrical Engineering and Computer Science

The University of Newcastle, Callaghan, NSW 2308, Australia

email: {sarah.johnson, steven.weller}@newcastle.edu.au

Abstract—Recent promising theoretical results for irregular repeat-accumulate (IRA) codes, together with their extremely simple encoding, motivates this investigation into the design and implementation of finite-length IRA codes. In this paper interleavers for RA codes are designed using combinatorial techniques to produce RA codes with Tanner graphs suitable for sum-product decoding. Further, a modified RA code accumulator is used to construct new IRA codes with columns of weight 3 in the accumulator. These new codes, called w3IRA codes, can be designed with flexible degree distributions and retain the simple encoding of traditional IRA codes.

I. INTRODUCTION

Turbo error correction codes provide near capacity error correction performances with practical encoding and decoding algorithms [1]. This is achieved through the parallel, or serial, concatenation of two simple constituent convolutional codes via an interleaver, and by decoding with an iterative algorithm which passes likelihood information between the two constituent decoders. Like turbo codes, low-density parity-check (LDPC) codes use simple constituent codes, single parity checks, and decode by iteratively passing likelihood values between the individual single parity-check decoders [2]. LDPC codes offer lower complexity decoding than turbo codes and, for long irregular LDPC codes, performance within a fraction of a decibel (dB) of the Shannon limit. However, a disadvantage of LDPC codes is an encoding complexity which can be quadratic in the code length.

A recent addition to the family of turbo-like codes, repeat-accumulate (RA) codes, provides a solution to this problem. Repeat-accumulate codes were first presented as a class of simple turbo codes for which coding theorems could be developed [3]. However it was soon realized that, although simple, RA codes are powerful turbo codes in their own right. When viewed as a serially concatenated turbo code, the two constituent codes of an RA code are a rate- $\frac{1}{q}$ repetition code and a rate- $1 - \frac{1}{1+D}$ convolutional code, called an accumulator, with a standard interleaver between them. RA codes can also be designed with an irregular degree distribution by using an irregular repetition code [4], [5]. For the binary erasure channel, Jin et al. have established that IRA code ensembles exist which can be decoded reliably in linear time at rates arbitrarily close to channel capacity [4].

Alternatively, IRA codes can be represented as a class of LDPC codes. When viewed as an LDPC code, the accumulator corresponds to weight two columns in the parity-check matrix of the code while the interleaver determines the structure of the remaining columns in the parity-check matrix, whose

weight is determined by the repetition code. The power of this interpretation of IRA codes is that they can be encoded using serial concatenation of the two constituent codes, as for turbo codes, but decoded using sum-product decoding on the code's Tanner graph, as for LDPC codes, thus gaining both the low encoding complexity of turbo codes and the decoding power and parallelization of LDPC codes. It is these codes we consider in this paper.

In this work we design practical, finite length, IRA codes and verify that their decoding performance is indeed comparable to that of LDPC codes. In designing IRA codes we consider two main issues. Firstly, we consider the selection of IRA degree distributions, both by constraining the optimization of density evolution thresholds to obtain degree distributions suitable for IRA codes, and by modifying the accumulator to achieve new IRA codes, called w3IRA codes, which can obtain the degree distributions traditionally used for LDPC codes. Secondly, we consider the design of IRA interleavers using combinatorial methods so that they translate to code Tanner graphs suitable for sum-product decoding.

II. REPEAT-ACCUMULATE CODES

A length N , rate R , IRA code begins with a repetition code, with distribution $[q_1, \dots, q_K]$, which repeats the i -th message bit of a length $K = RN$ message q_i times. The rate of the repetition code is then $1/q$, where

$$q = \frac{1}{K} \sum_{i=1}^K q_i.$$

A length $n = qK$ interleaver, $\Pi = [\pi_1, \pi_2, \dots, \pi_n]$, then permutes the bits $\mathbf{b} = [b_1, b_2, \dots, b_n]$ from the repetition code to give

$$\mathbf{d} = [d_1, d_2, \dots, d_n] = [b_{\pi_1}, b_{\pi_2}, \dots, b_{\pi_n}].$$

The bits at the output of the interleaver are then combined, modulo 2, in M sets with a_i bits in the i -th set, producing a length M output, $\mathbf{r} = [r_1, \dots, r_M]$ where

$$a = \frac{1}{M} \sum_{i=1}^M a_i.$$

Finally, the $M = Kq/a$ parity bits, $\mathbf{p} = [p_1, \dots, p_M]$ at the output of the accumulator are described by

$$p_i = p_{i-1} \oplus r_i, \quad i = 2, \dots, M \quad (1)$$

where $p_1 = r_1$, and \oplus represents modulo 2 addition.

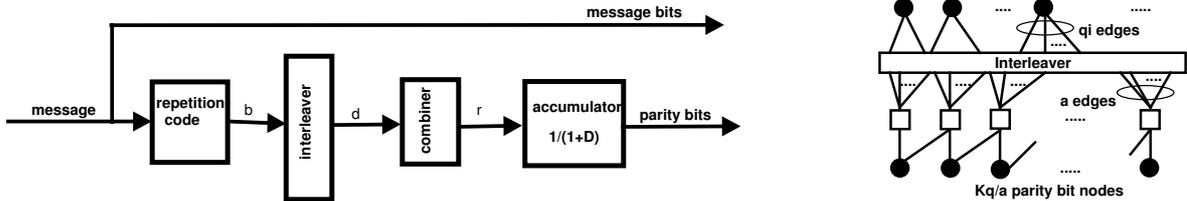


Fig. 1. A systematic IRA code viewed as a turbo code, on the left, and as an LDPC code with Tanner graph shown on the right. Filled circles in the Tanner graph represent bit nodes, while open squares are check nodes.

The original RA codes of [3] are nonsystematic codes with constant q_i , and $a_i = 1$ for all i . In this paper the focus is on systematic codes with variable q_i and $a_i \geq 1$. Thus both the original message bits and the parity bits are explicitly sent to the receiver, and the final codeword is

$$\mathbf{c} = [m_1, m_2, \dots, m_K, p_1, p_2, \dots, p_M],$$

to give a length $N = K(1 + q/a)$, and rate $R = \frac{a}{a+q}$, code.

A systematic IRA code can also be viewed as an LDPC code where the code parity-check matrix H has two parts;

$$H = [H_1, H_2]. \quad (2)$$

Here H_1 is an $M \times K$ binary matrix, with column weights q_1, \dots, q_K , row weights a_1, \dots, a_M , and where the location of the non-zero entries is specified by the interleaver. H_2 is an $M \times M$ matrix which describes (1) and so has the form:

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 \\ \vdots & & \ddots & & \vdots \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 \end{bmatrix}. \quad (3)$$

The Tanner graph of an IRA code is directly described by H and consists of M parity-check nodes and $K + M$ bit nodes. Fig. 1 shows the Tanner graph for a systematic IRA code.

Interestingly, designers of LDPC codes, while considering how to arrange the large number of weight two columns required in LDPC codes by density evolution, have designed LDPC codes which are essentially IRA codes, using Cayley graphs in [6] and a pseudo-random construction in [7]. The pattern of weight two columns in the parity-check matrix of an IRA code is the best way to add weight two columns without adding cycles amongst the degree-2 nodes [6], [7].

An IRA code is completely specified by N , a_1, \dots, a_M , q_1, \dots, q_K and the interleaver pattern Π . Thus designing an IRA code involves choosing the degree distributions and designing the interleaver. We consider the degree distributions in the following section and the interleaver design in Section V.

III. DEGREE DISTRIBUTIONS

The variable degree distribution for an LDPC code is denoted by

$$\lambda(x) = \lambda_2 x + \dots + \lambda_i x^{i-1} + \dots + \lambda_{q_{\max}} x^{q_{\max}-1},$$

where λ_i is the fraction of Tanner graph edges emanating from degree- i bit nodes (including both the message bit and parity bit nodes). The row degrees are similarly represented by $\rho(x)$. The length N , rate R , IRA code with degree distribution $\lambda(x)$ will thus have

$$N \frac{\lambda_i/i}{\sum_j \lambda_j/j}$$

message nodes of degree- i , for $i > 2$, and

$$N \frac{\lambda_2/2}{\sum_j \lambda_j/j} - (1 - R)N$$

message nodes of degree-2.

In an IRA code the accumulator requires that there be at least M nodes of degree-2 (for the purposes of finding a degree distribution we count the single weight-1 column in H_2 as a degree-2 node); thus we constrain the optimization process to require

$$\lambda_2 \geq \frac{2(1-R)}{R} \sum_{j=3}^{\max(q)} \frac{\lambda_j}{j} = 2 \sum_{i=1}^{\max(a)} \frac{\rho_i}{i}, \quad (4)$$

so that we only consider degree distributions for IRA codes with sufficiently many weight two columns to form the accumulator of the code.

The optimal column weight distribution for an IRA code of a given rate is determined using density evolution [8] as for LDPC codes, but with the addition of the constraint in (4). We use differential evolution, using the method described in [9], to find the degree distribution which optimizes the ensemble capacity thresholds. The average check degree is not specified in advance, rather the valid check degrees are treated as independent variables in the same way as the bit degrees.

Table I gives the degree distributions and corresponding thresholds, σ^* , for rate- $\frac{1}{4}$, rate- $\frac{1}{2}$ and rate- $\frac{3}{4}$ codes on the additive white Gaussian noise (AWGN) channel. Shown are the degree distributions for LDPC codes, labeled 'Ldpc', and, where they differ, the degree distributions satisfying (4) for IRA codes, labeled 'IRA'. The rate- $\frac{1}{4}$ 'Ldpc' distribution is from LdpcOpt [10], and the rate- $\frac{1}{2}$ 'Ldpc' distribution is from [7]. Further, since density evolution is only optimal for asymptotically long codes, and we consider practical length codes in this paper, we also consider codes with constrained degree distributions for improved error floors, found using the method in [11] by setting $E = 0.01$.

TABLE I
DEGREE DISTRIBUTIONS FOR SELECTED CODE RATES (AWGN)

	Rate-1/4			Rate-1/2			Rate-3/4	
	Ldpc	IRA	$E_{0.01}$	Ldpc	IRA	$E_{0.01}$	Ldpc	$E_{0.01}$
λ_2	0.4162	0.5001	0.0086	0.3078	0.3330	0.0075	0.2470	0.0043
λ_3	0.2362	0.0700	0.9711	0.2729	0.3851	0.9759	0.4321	0.8931
λ_4	0.0001	0.0253	0.0006		0.0002	0.0033	0.0878	0.0126
λ_5	0.0657	0.0163						
λ_6	0.0775	0.1728	0.0059					
λ_7			0.0110	0.4193	0.1392	0.0038		
λ_8					0.1425	0.0095		
λ_{10}	0.2043	0.2155	0.0028				0.2331	0.0900
ρ_3			0.0118					
ρ_4	1	1	0.9332					
ρ_5			0.0550			0.0016		
ρ_6				0.4	0.9849	0.9600		
ρ_7				0.6	0.0151	0.0384		
ρ_{11}							0.0086	0.0676
ρ_{12}							0.1800	0.0200
ρ_{13}							0.8114	0.9124
σ^*	1.50	1.11	1.266	0.9405	0.9241	0.8817	0.633	0.622

IV. COLUMN WEIGHT THREE ACCUMULATORS

In Table I we see that the optimal degree distribution for a rate- $\frac{1}{4}$ length-1000 LDPC code requires that 624 of the columns are weight-2, which is fewer than the 750 weight-2 columns required in a traditional IRA code. In general, for low rate codes density evolution typically returns degree distributions requiring fewer than M weight-2 columns and thus, for these rates, traditional IRA codes are not able to achieve these degree distributions. Similarly the degree distributions for codes with improved error floor also require fewer weight two columns than those in the accumulator and thus will not be achievable with a traditional IRA code.

Further, the stability condition for the binary input AWGN channel with variance σ^2 [8]:

$$\lambda'(0)\rho'(1) < e^{\frac{1}{2\sigma^2}} \quad (5)$$

places an upper bound on the allowed values of λ_2 . For the rate half codes the only way to satisfy both (4) and (5), is to reduce the code density, and thus the thresholds, σ^* , for these degree distributions will be reduced.

Motivated by these limitations of traditional IRA codes we consider a new design for IRA codes, called w3IRA codes, which have accumulators which permit the construction of codes with many fewer than M weight-2 columns.

For the w3IRA codes we use a rate-1

$$\frac{1}{1 + D + D^{g+1}}$$

convolutional code as the second constituent code, where g becomes a design parameter of the new codes. Translating to the code parity-check matrix, the i -th column of H_2 will be non-zero in the i -th, $(i+1)$ -th and $(i+1+g)$ -th rows, where these rows exist. Thus g specifies the number of rows between the second and third non-zero entries in the columns of H_2 as well as the number of weight two columns remaining. While reducing g increases the number of weight-3 columns in H_2 , cycles of size $2(g+1)$ introduced by the new accumulator become a problem if g is very small.

With this modification we gain the flexibility to design w3IRA codes with the degree distributions optimized for LDPC codes without losing the low complexity encoding advantage of IRA codes. For example, a length-1000 rate- $\frac{1}{4}$ IRA code with the LDPC degree distribution can now be constructed by using a w3IRA code and setting $g = 623$.

V. COMBINATORIAL INTERLEAVERS

The decoding performance of an IRA code is determined by the properties of the code's Tanner graph, as for an LDPC code. Thus we wish to design the IRA code interleaver with a focus on the resulting Tanner graph, and in particular, on avoiding small cycles in the graph. A cycle in the Tanner graph is a sequence of connected nodes which start and end at the same node and contain no other node more than once. The length of the cycle is the number of edges it contains and the girth of a graph is the length of its shortest cycle.

For LDPC codes combinatorial constructions have successfully been used to construct a wide range of codes with excellent decoding performances, in part by guaranteeing Tanner graphs with a certain minimum girth. The incidence matrices of Steiner 2-designs translate directly to the parity-check matrices of regular LDPC codes which, by the properties of the design, are free of cycles with length less than 6. For RA codes the mapping between combinatorial designs and codes is less straightforward due to the constraints of the accumulator. However, we showed in [12] that regular RA codes with $q = 3$ can be constructed from the incidence matrix of any Steiner 2-design with order $\gamma = 3$. In Theorem 2, we extend this result to show that for every possible Steiner 2-design, of any order $\gamma \geq 3$, there exists a row and column permutation that will map the incidence matrix of the design into the form required for an RA code interleaver and accumulator, thus producing an RA code with Tanner graph of girth 6 for any q .

A Steiner 2-design is an arrangement of a set of v points into $b = v(v-1)/\gamma(\gamma-1)$ subsets, called *blocks* such that [13]:

- D1 there are exactly γ points in each block,
- D2 there are exactly $\rho = (v-1)/(\gamma-1)$ blocks which contain each point, and
- D3 every pair of points of the design appear in exactly one block together.

A Steiner 2-design can be described by a binary $v \times b$ incidence matrix \mathcal{N} where each column in \mathcal{N} represents a block B_j of the design and each row a point P_i :

$$\mathcal{N}_{i,j} = \begin{cases} 1 & \text{if } P_i \in B_j, \\ 0 & \text{otherwise.} \end{cases}$$

Lemma 1: Any row or column permutation of \mathcal{N} is also the incidence matrix of a Steiner 2-design.

Proof: Immediate from the independence of the properties of a Steiner 2-design (D1-D3) from the ordering of points and blocks. \blacksquare

By Property D3 we also have

$$\forall i \in \{1, \dots, v-1\} \exists B_j \in \mathcal{N} : P_i, P_{i+1} \in B_j. \quad (6)$$

We denote by $B(i)$ the block containing both P_i and P_{i+1} . Then if

$$B(i) \neq B(j) \quad \forall i, j \leq v-1, i \neq j, \quad (7)$$

the columns $B(1), \dots, B(v-1)$ are distinct. Otherwise we need to apply a permutation to the rows of \mathcal{N} so that (7) holds.

Lemma 2: For any \mathcal{N} , the incidence matrix of an Steiner 2-design, with $\gamma \geq 3$ we can find some row permutation π so that (7) holds.

Proof: Take any column c of \mathcal{N} . There are

$$B_c = (v-2) \binom{\gamma}{3} 3!(v-3)! \quad (8)$$

permutations of the rows of \mathcal{N} that will place three of the non-zero entries of column c into three consecutive rows. We call these the ‘‘bad’’ permutations for column c . There are $v(v-1)/\gamma(\gamma-1)$ columns in \mathcal{N} , so for any of the $v!$ row permutations to cause no bad columns each permutation which causes a bad column must cause at least $\gamma-2$ bad columns.

For each bad permutation there are $3\binom{v-1}{\gamma-1}$ columns in \mathcal{N} which intersect column c at one of the three consecutive points and the B_c permutations are bad for each of these columns

$$B_I = 3! \binom{\gamma}{3} \left[3! \binom{\gamma-1}{3} (v-4)! + \frac{2}{3} \binom{\gamma-1}{2} 2!(v-4)! \right]$$

times. Further, the $b - 3\binom{v-1}{\gamma-1} - 1$ remaining columns in \mathcal{N} which do not intersect c at the three consecutive points are each bad for

$$B_N = 3! \binom{\gamma}{3} \left[2 \binom{v-4}{2} 3! \binom{\gamma}{3} (v-6)! \right] \quad (9)$$

of the B_c permutations. Thus the B_c permutations cause

$$B_T = B_c + 3\binom{v-1}{\gamma-1} B_I + (b - 3\frac{v-1}{\gamma-1} + 2) B_N$$

bad columns in total. Therefore the theorem is satisfied when

$$\frac{B_T}{B_c} = 1 + \frac{(\gamma-2)(v(v-8) - \gamma^2 + 8\gamma)}{(v-2)(v-3)} > \gamma - 2$$

which is when

$$v > \frac{1}{2}(3\gamma - 1 + \sqrt{4\gamma^3 - 31\gamma^2 + 82\gamma - 71}). \quad (10)$$

Existence of a Steiner 2-design requires $v \geq \gamma^2 - \gamma + 1$, and for $\gamma \geq 3$ (10) is satisfied for all such v . ■

By Lemma 1, (6) holds for any permutation of the rows of \mathcal{N} , and by Lemma 2 we can always find a permutation of the rows of \mathcal{N} such that (7) holds. The columns $B(1), \dots, B(v)$, where $B(v)$ is a chosen column with $P_v = 1$, can be used to form H_2 by replacing the ‘1’ entries of $B(i)$ which are not in the i -th and $(i+1)$ -th rows with ‘0’. The remaining columns of \mathcal{N} then give H_1 by removing $\gamma - q_i$ of the non zero entries in each of the columns, $i = 1, \dots, K$, choosing the entries to be removed to achieve the required row degrees. Finally, the interleaver can be determined directly from H_1 .

Since every pair of points occurs in exactly one block together, no two columns of \mathcal{N} have a non-zero entry in more

than one row in common. Removing entries from \mathcal{N} cannot change this, and thus the IRA code constructed as above has a Tanner graph free of 4-cycles. For IRA codes with larger girth other combinatorial structures such as partial geometries may be employed.

For w3IRA codes with columns of weight three in the accumulator, there are now three times as many row pairs present in H_2 and so it becomes more difficult to construct interleavers without adding 4-cycles into the code. However, by using the properties of Steiner 2-designs it is still relatively straightforward to construct 4-cycle free codes.

Construction 1: Using the incidence matrix, \mathcal{N} , of a Steiner 2-design on v points construct a length N , rate K/N , 4-cycle free IRA code, with column weights $[q_1, \dots, q_K]$ and with all but $g+1$ of the columns of the accumulator weight 3, as follows:

S1 For $i = 1, \dots, M$,

- a) Find the column in \mathcal{N} containing the pair of points $\{i, i+1\}$ and swap it with the i -th column of \mathcal{N} .
- b) (when $i < M - g - 1$) If the new i -th column has a ‘1’ entry in row $j \geq i + g + 1$, switch the j -th and $(i + g + 1)$ -th rows. Otherwise place a ‘1’ in entry $\mathcal{N}(i + g + 1, i)$.
- c) If any row r , $r > i + 1$, with non-zero entry in column i is adjacent to any other row with a non-zero entry in column i , randomly swap row r with some other row t , with $t > i + g + 1$. In the case that $r < i + 1 + g$ place a ‘1’ in entry $\mathcal{N}(r, r - 1 - g)$ and place a ‘0’ in $\mathcal{N}(t, r - 1 - g)$.
- d) Replace the non-zero entries in column i , other than those in rows $i, i + 1$, and $i + g + 1$, with ‘0’.

S2 Form the submatrix H_2 consisting of the first M columns and the first M rows of \mathcal{N} , and the sub matrix H_1 consisting of the first M rows of any choice of K of the remaining columns of \mathcal{N} . If a ‘1’ was entered in $\mathcal{N}(i + g + 1, i)$, or $\mathcal{N}(r, r - 1 - g)$, in step b) the two columns with pairs $\{i + g + 1, i\}$ and $\{i + g + 1, i + 1\}$, or $\{r, r - g - 1\}$ and $\{r, r - g\}$, are not chosen when selecting H_1 .

S3 Remove $\gamma_i - q_i$ of the ones in the i -th column of H_1 , (where γ_i is the weight of the i -th column), choosing the entries to remove to achieve the required row degrees. $H = [H_1, H_2]$ is then the IRA code, and the interleaver can be read directly from H_1 . □

Again the codes constructed as above have Tanner graphs free of cycles with size less than 6 due to the properties of the Steiner 2-design.

We demonstrate in Figs. 2-4 the performance, on an AWGN channel, of IRA and w3IRA codes constructed from a $v = 1024$, $\gamma = 32$ design, compared to IRA and w3IRA codes with randomly constructed interleavers. Also shown are LDPC codes with the same degree distribution, which are constructed pseudo-randomly to avoid 4-cycles.

In this paper we have considered three main issues in designing IRA codes: choosing the degree distributions, designing the accumulator, and constructing the interleaver.

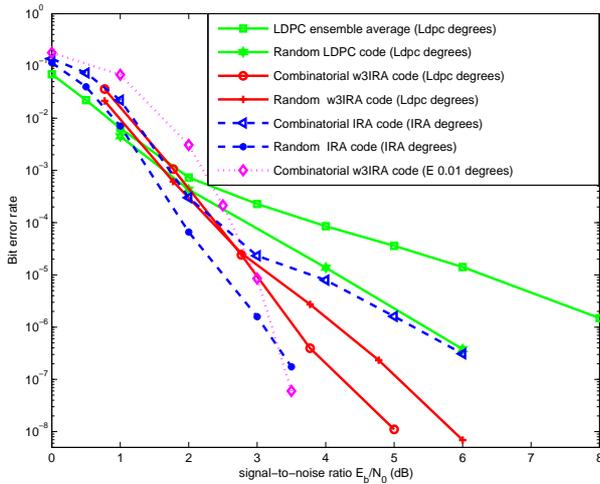


Fig. 2. The error correction performance on an AWGN channel of a length 1000 rate-1/4 IRA codes decoded with sum-product decoding to a maximum of 100 iterations.

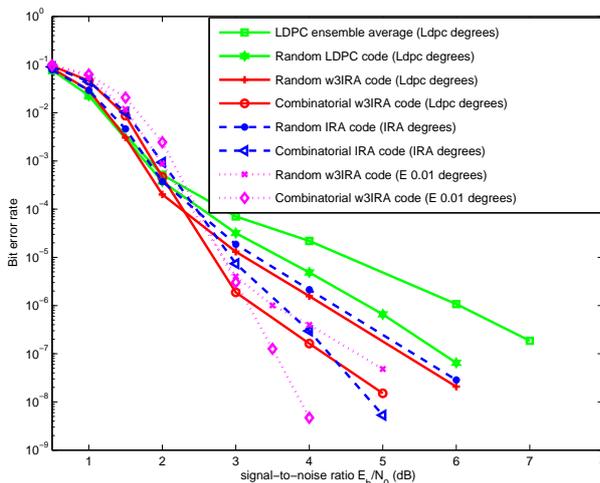


Fig. 3. The error correction performance on an AWGN channel of length 1023 rate-1/2 IRA codes decoded with sum-product decoding to a maximum of 100 iterations.

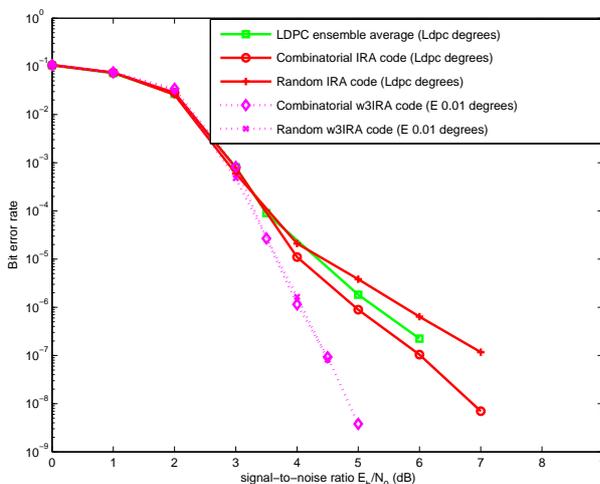


Fig. 4. The error correction performance on an AWGN channel of length 1000 rate-3/4 IRA codes decoded with sum-product decoding to a maximum of 100 iterations.

Firstly, we note from the performance results that there is significant benefit in constraining the degree distributions of IRA codes away from those traditionally used for LDPC codes, both to obtain degree distributions suitable for traditional IRA codes (particularly for low rate codes), and to obtain degree distributions for lower error floors (particularly for high rate codes). Secondly, our results show that by carefully placing the weight-2 columns in H , the accumulator of the IRA and w3IRA codes provide not just an encoding complexity advantage over the traditional LDPC codes but a decoding advantage as well. The w3IRA accumulators in particular provide the flexibility to construct easily encodable codes with optimized thresholds ('ldpc' degrees) or reduced thresholds and improved error floor performance (' $E_{0.01}$ ' degrees).

Finally, as for LDPC codes, we see that there can be an advantage for higher rate codes in considering IRA and w3IRA codes constructed using Steiner 2-designs. In the terminology of LDPC codes we have, equivalently, provided a construction method for 4-cycle free irregular LDPC codes which have an extremely simple encoding circuit.

VI. ACKNOWLEDGEMENT

The authors would like to thank Prof. Bill Ryan and Li Yan from the University of Arizona, for sharing their discretized density evolution software with us.

REFERENCES

- [1] C. Berrou and A. Glavieux, "Near optimum error correcting coding and decoding: Turbo codes," *IEEE Trans. Commun.*, vol. 44, no. 10, pp. 1261–1271, October 1996.
- [2] R. G. Gallager, "Low-density parity-check codes," *IRE Trans. Inform. Theory*, vol. IT-8, no. 1, pp. 21–28, January 1962.
- [3] D. Divsalar, H. Jin, and R. J. McEliece, "Coding theorems for "turbo-like" codes," in *Proc. 36th Allerton Conf. on Communications, Control, and Computing*, Allerton, Illinois, September 1998, pp. 201–210.
- [4] H. Jin, D. Khandekar, and R. J. McEliece, "Irregular repeat-accumulate codes," in *Proc. of the Second International Symposium on Turbo Codes and Related Topics*, Brest, France, September 2000, pp. 1–8.
- [5] A. Roumy, S. Guemghar, G. Caire, and S. Verdú, "Design methods for irregular repeat-accumulate codes," *IEEE Trans. Inform. Theory*, vol. 50, no. 8, pp. 1711–1727, August 2004.
- [6] P. Vontobel, "Algebraic Coding for Iterative Decoding," Ph.D. dissertation, Swiss Federal Institute of Technology, Zurich, 2003.
- [7] M. Yang, W. E. Ryan, and Y. Li, "Design of efficiently encodable moderate-length high-rate irregular LDPC codes," *IEEE Trans. Comm.*, vol. 52, no. 4, pp. 564–571, April 2004.
- [8] T. J. Richardson, M. A. Shokrollahi, and R. L. Urbanke, "Design of capacity-approaching irregular low-density parity-check codes," *IEEE Trans. Inform. Theory*, vol. 47, no. 2, pp. 619–637, February 2001.
- [9] J. Hou, P. H. Siegel, and L. B. Milstein, "Performance analysis and code optimisation of low density parity-check codes on Rayleigh fading channels," *IEEE J. Selected Areas Commun.*, vol. 19, no. 5, pp. 924–934, May 2001.
- [10] A. Amraoui and R. L. Urbanke, *LdpcOpt*: (<http://lthcwww.epfl.ch/research/ldpcopt/>).
- [11] S. J. Johnson and S. R. Weller, "Constraining LDPC degree distributions for improved error floor performance," *submitted to IEEE Communications Letters*, 2005.
- [12] —, "Interleaver and accumulator design for systematic repeat-accumulate codes," in *Proc. 6th Australian Communications Theory Workshop (AusCTW'05)*, Brisbane, Australia, 2–4 February 2005.
- [13] C. J. Colbourn and J. Dinitz (Eds.), *The CRC Handbook of Combinatorial Designs*. Boca Raton: CRC Press, 1996.