Generalized Low Density (Tanner) Codes

J. Boutros†, O. Pothier‡, G. Zémor†
† Ecole Nationale Supérieure des Télécommunications,
46 Rue Barrault, 75634 Paris cedex 13, France
‡ Laboratoire d'électronique Philips,
22, av. Descartes, BP15, 94453 Limeil-Brévannes cedex, France
Email : {boutros,pothier,zemor}@enst.fr

Abstract

We build a class of pseudo-random error correcting codes, called *Generalized Low Density* codes (GLD), from the intersection of two interleaved block codes. GLD codes performance approaches the channel capacity limit and the GLD decoder is based on simple and fast SISO (Soft Input - Soft Output) decoders of smaller block codes. GLD codes are a special case of Tanner codes and a generalization of Gallager's LDPC codes. It is also proved by an ensemble performance argument that these codes are asymptotically good in the sense of the minimum distance criterion. The flexibility in selecting the parameters of GLD codes makes them suitable for small and large block length forward error correcting schemes.

1 Introduction

We build a class of pseudo-random error correcting codes (called GLD codes) by generalizing Gallager's construction of low density parity check codes (LDPC) [1]. Each parity check equation of an LDPC code (N, K) is replaced by the parity check matrix of a small linear code (n, k) called the *constituent code*.

LDPC codes are usually defined by their parity check matrix, but they can also be described with a bipartite graph. The left part of the graph contains the code symbols and the right one contains the parity check nodes. A parity check node is associated to the trivial parity check code $(n, n-1)$. This representation of LDPC codes has been used by Sipser and Spielman [2] to study the influence of the graph expansion on the code parameters.

The graphical representation of block codes has been first exploited and generalized by Tanner [3]. Tanner codes based on a bipartite deterministic graph are obtained by replacing the $(n, n-1)$ code associated to one parity check node with a less trivial constituent code (n, k). Thus, building a Tanner code on a random graph (instead of a deterministic one) is a second method to construct GLD codes.

In the sequel, we restrict our description of GLD codes to their matrix representation. As explained in the next section, the GLD class is obtained from the intersection of two or more interleaved subcodes. The subcodes of length N are a direct sum of N/n constituent codes.

GLD codes exhibit an excellent performance on both AWGN and Rayleigh channels and present a high BER slope at high SNR due to their large minimum distance. The decoding algorithm is based on a SISO (Soft Input - Soft Output) decoding of the small constituent code and has a very low complexity. The decoding time can be dramatically reduced when the SISO decoders of the N/n constituent codes are executed in parallel. It is also proved that GLD codes are asymptotically good in the sense of the minimum distance criterion.

2 Structure of the GLD code

Figure 1 shows the parity check matrix H of an LDPC code (N, K) with length $N = 12$ and rate $R \geq 1/4$. The matrix H is the concatenation of $J = 3$ submatrices. The first submatrix H_1 of size 3×12 defines a subcode by the direct sum of three parity check codes $(n, n-1)$ with $n = 4$. The whole matrix H is obtained by concatenating H_1, $H_2 = \pi_1(H_1)$ and $H_3 = \pi_2(H_1)$, where π_1 and π_2 are two pseudo-random column permutations. This example can be generalized to build any LDPC code (N, K) using $J - 1$ permutations. Gallager showed that LDPC codes are asymptotically good when $J \geq 3$. He also described an iterative decoding algorithm which is the ancestor of turbo decoding [4] and he exploited the low density of the parity check matrix to reduce the decoder complexity when computing the a posteriori probabilities.

Each line of the LDPC matrix H is a parity check equation defined by the $(n, n-1)$ parity code. We replace this line by $n - k$ lines including one copy of the parity check matrix H_0 of a constituent code $C_0(n, k)$. This operation is depicted on Figure 2. The first submatrix produces the

0-7803-5284-X/99/$10.00 © 1999 IEEE.

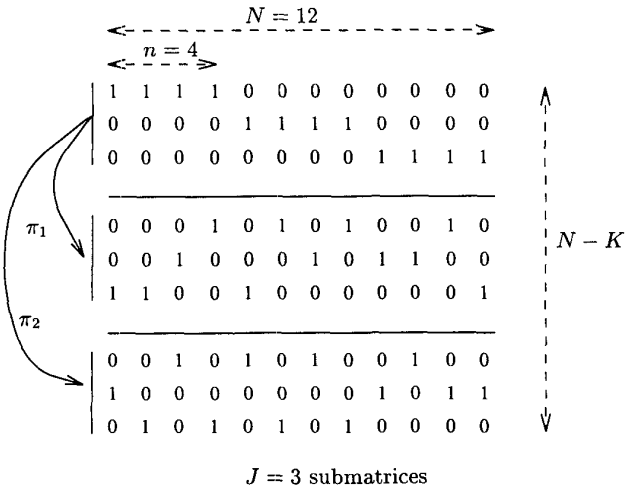

Figure 1: Example of an LDPC matrix with $J = 3$ levels.

direct sum of N/n identical codes $C_0(n,k)$. The matrix H has J submatrices derived by interleaving the columns of the first submatrix. This type of parity check matrices H defines the class of GLD codes. Thus, a GLD code C is the intersection of J subcodes C_j, i.e. $C = \bigcap_{j=1}^{J} C_j$ where $C_{j+1} = \pi_j(C_1)$ for $j = 1 \ldots J-1$, and $C_1 = C_0 \oplus \ldots \oplus C_0$. If C_0 has a rate $r = k/n$, the total rate of the GLD code is $R = 1 - J(1-r)$ when the permutations π_j are random. Note that it is not possible to define the GLD code as a serial (neither parallel, nor hybrid) concatenation of two or multiple constituent codes.

In our study, we considered binary GLD codes with only $J = 2$ levels (the total rate is $R = 2r - 1$) based on binary Hamming codes. As shown in the next section, we need only $J = 2$ (i.e. one interleaver) to make the GLD code asymptotically good. For practical applications, efficient GLD codes can be built from primitive, shortened or extended binary BCH codes.

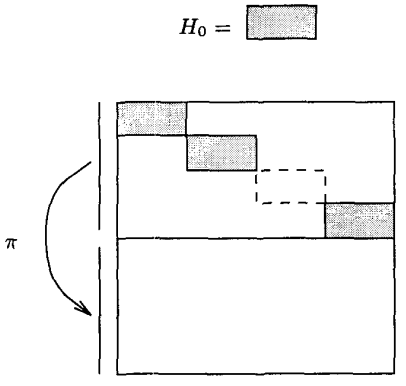

Figure 2: Structure of a GLD parity check matrix ($J = 2$).

3 Ensemble Performance

Without loss of generality, we restrict the theoretical study in this section to the case of a two levels GLD code based on the binary BCH code $C_0(7,4)$. It is shown how to compute the average weight distribution of the generalized low density code, and the latter is proved to be asymptotically good. We also consider a BSC channel with a transition probability $0 < p < 1/2$ and find the maximal value of p for which $P_{ew} \to 0$ when N is large.

3.1 Weight distribution and asymptotical performance

Let us start by computing the average weight distribution of the ensemble of GLD codes built with $C_0(7,4)$ and a random column permutation π. In other words, the weight coefficients are obtained by averaging over all the possible interleavers π. The moment-generating function $g(s)$ of C_0 is given by :

$$g(s) = \frac{1 + 7e^{3s} + 7e^{4s} + e^{7s}}{16}$$

The first subcode C_1 of length N is the direct sum of N/n independent codes C_0. Hence, its moment-generating function $G(s)$ is simply a power of $g(s)$:

$$G(s) = g(s)^{N/n} = \sum_{\ell} Q(\ell) e^{\ell s}$$

where $Q(\ell)$ is the probability that a vector of weight ℓ belongs to C_1. Since the total number of codewords in C_1 is $(2^k)^{N/n}$, then the average number in C_1 of codewords of weight ℓ is $N_1(\ell) = 2^{(kN/n)} Q(\ell)$. Exploiting the fact that C_1 and $C_2 = \pi(C_1)$ are totally independent, the probability that a vector of weight ℓ belongs to $C = C_1 \bigcap C_2$ can be written as :

$$P(\ell) = \left(\frac{N_1(l)}{\binom{N}{\ell}} \right)^2$$

Finally, the average number of codewords in C having weight ℓ is :

$$\overline{N(\ell)} = \binom{N}{\ell} \times P(\ell) = \frac{2^{(2kN/n)} Q(\ell)^2}{\binom{N}{\ell}} \qquad (1)$$

By using exactly the same bounding technique as in [1], *i.e.* upper bounding each of the coefficients $Q(\ell)$ with $G(s)e^{-\ell s}$, and after applying the extended Stirling approximation (valid for large N), we get an upper bound

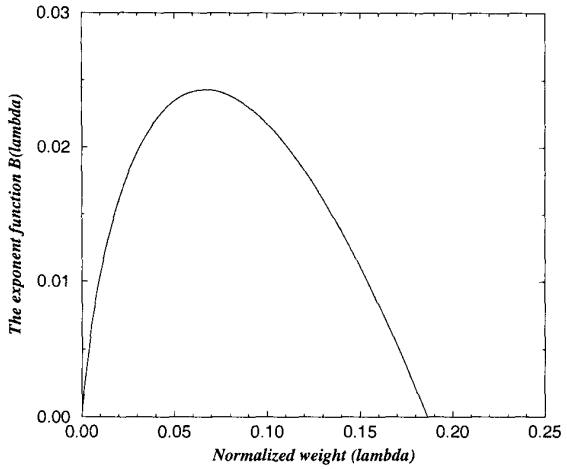

Figure 3: The exponent function $B(\lambda)$ versus the normalized weight λ. The low density code is built from the $(7,4)$ Hamming code.

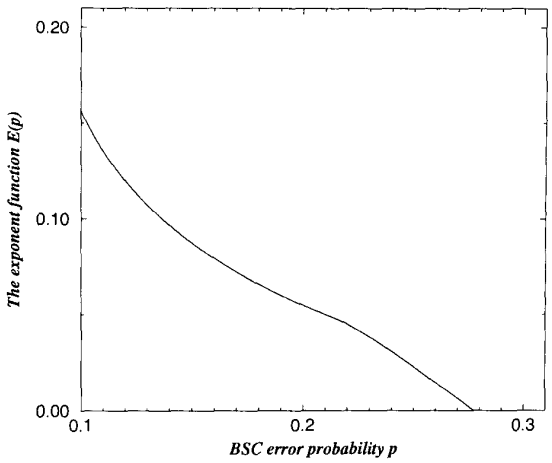

Figure 4: The exponent function $E(p)$ versus the BSC channel transition probability p. The low density code is built from the $(7,4)$ Hamming code.

on the average number of codewords of weight ℓ in the GLD code (we omit the details) :

$$\overline{N(\ell)} \leq C(\lambda, N) \times e^{-NB(\lambda)}$$

where $\lambda = \ell/N$ is the normalized weight.

The two functions $C(\lambda, N)$ and $B(\lambda)$ are expressed as follows :

$$C(\lambda, N) = \sqrt{2\pi N\lambda(1-\lambda)} \times e^{1/(12N\lambda(1-\lambda))}$$

$$B(\lambda) = H(\lambda) - \frac{2}{n}\left[\mu(s) + k\ log2\right] + 2s\lambda$$

where $H(\lambda)$ is the natural entropy function and $\mu(s) = \log(g(s))$. The upper bound has been optimized and the optimal value of s is related to the weight by $\lambda = \mu'(s)/n$, where $\mu'(s)$ is the derivative of $\mu(s)$ relative to s.

The exponent function $B(\lambda)$ is sketched in Figure 3. Asymptotically, when $N \to \infty$, the average number $\overline{N(\ell)}$ of codewords of weight ℓ goes to zero if $B(\lambda) > 0$. The first value of $\lambda \in]0 \ldots 1/2[$ corresponding to a sign transition gives us a *lower bound* on the minimum distance $\delta(C) = d_{min}(C)/N$ of the GLD code. As seen in Figure 3, $\delta \geq 0.186$. Thus, C is **asymptotically good** with $d_{min} \geq 0.186N$ and $R = 1/7$. Note that the Gilbert-Varshamov bound gives $\delta_0 = H_2^{-1}(1-R) = 0.281$.

3.2 BSC channel threshold

Now, let us compute the maximal value of p for which the word error probability P_{ew} of an ML decoder goes to zero when N is arbitrarily large. An upper bound on P_{ew} is obtained by assuming that a decoding error occurs when at least half of the codeword non zero symbols are covered. If j denotes the channel error weight, ℓ the weight of a codeword and i the number of covered non zero bits, we have the following upper bound :

$$P_{ew} \leq \sum_{j=1}^{N} p^j (1-p)^{N-j} \sum_{\ell=d_{min}}^{N} \overline{N(\ell)} \sum_{i=\ell/2}^{\ell} \binom{\ell}{i} \binom{N-\ell}{j-i}$$

When N is large enough, an expression similar to the upper bound on $\overline{N(\ell)}$ can be found for P_{ew}. After some algebraic manipulations, we get (details omitted) :

$$P_{ew} \leq D(N,p) \times e^{-NE(p)}$$

where the exponent function $E(p)$ is given by :

$$E(p) = Min_\lambda \left[B(\lambda) + H(p) - \lambda log2 - (1-\lambda)H(\frac{p-\lambda/2}{1-\lambda})\right]$$

Figure 4 shows $E(p)$ versus p. From this curve, we conclude that an ML decoder for this GLD code achieves $P_{ew} \to 0$ if $p < 0.277$ (not far from 0.281 given by the BSC channel capacity).

4 Bounds on minimum distance and error probability

In this section, we check that GLD codes are asymptotically good by giving an *upper bound* for the minimum

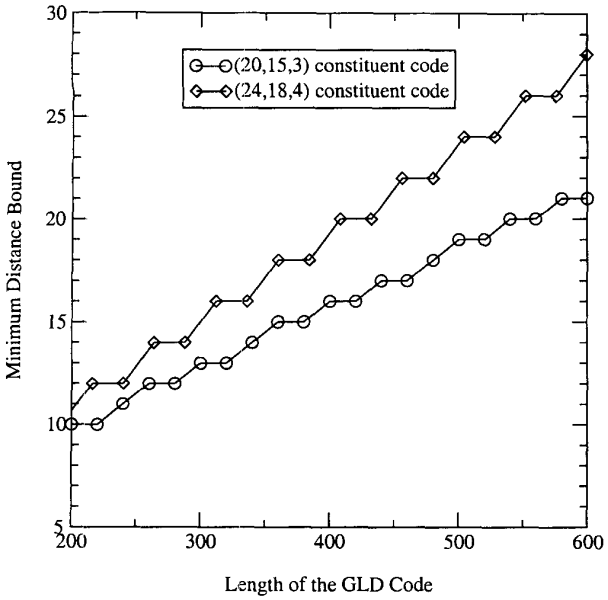

Figure 5: Upper bounds based on (3) for GLD Codes of two different constituent codes

distance derived from the weight distribution in expression (1). Secondly, we present an average bound for the bit error probability when maximum likelihood decoding is used.

Using (1), it is easy to numerically compute the average distance distribution $\overline{N(\ell)}$ of any GLD code C from the moment generating function $g(s)$ of the constituent code C_0. Then from the following inequality

$$\text{Prob}(d_{Hmin} \leq D) \leq \sum_{\ell=1}^{D} \overline{N(\ell)}, \tag{2}$$

we compute an upper bound for the minimum Hamming distance of C by taking the right hand side of (2) equal to 1. So, we have :

$$d_{Hmin} \leq \Delta$$

where Δ is the smallest integer such that

$$\left\lfloor \sum_{\ell=1}^{\Delta} \overline{N(\ell)} \right\rfloor = 1 \tag{3}$$

Figure 5 shows this bound Δ calculated for two sets of GLD codes. The first one is based on the (20,15,3) constituent code, a (31,26,3) shortened BCH code. The second one is based on the (24,18,4) constituent code,a shortened (32,26,4) BCH code. Both of them are two-level, rate 1/2 GLD codes. The granularity of the curves is due to the floor function in (3) and to the fact that the length of a GLD code is a multiple of the constituent code length. The average minimum distance of these codes is clearly a linear function of their length.

The average distance distribution $\overline{N(\ell)}$ can also be used for computing the bit error probability of ML decoding. Actually, the interleaver acts on all coded bits, so that they are equally protected. Thus, we can write the following Union-Bound (UB) for a transmission over an AWGN channel :

$$P_{eb} \leq \sum_{\ell=1}^{N} \frac{\ell}{N} \times \overline{N(\ell)} \times \frac{1}{2}\text{erfc}\left(\sqrt{R\ell\frac{E_b}{N_0}}\right) \tag{4}$$

where E_b/N_0 is the signal-to-noise ratio per information bit and R the GLD code rate. It is also possible to derive an improved bound as presented in [5], based on the Gallager's bound, which is tighter for low SNR. Both bounds are presented and compared to simulation results on Figure 6.

5 Decoding scheme

Gallager presented in [1] an iterative decoding scheme for LDPC codes. This algorithm computes iteratively the probability of each coded bit given a set of received channel observations that becomes larger with the iteration steps. The goal is to estimate the *a posteriori* probability, namely the probability of the coded bits given all received samples. This algorithm looks very similar to the one proposed in [4] for the well known turbo codes[1] and may be considered as the ancestor of all turbo decoding techniques.

GLD codes are decoded using the same idea. For each bit, we compute its probability given the received samples and considering that it belongs only to the first subcode C_1. Exploiting the fact that a subcode is composed of N/n independent constituent codes C_0, this can be done using N/n simple Soft-Input Soft-Output (SISO) decoders working in parallel on every constituent code. This first step generates for each coded bit an "*a posteriori*" probability and an *extrinsic* probability. The latter is fed through the interleaver to the second step as an *a priori* information for the SISO decoders working on the constituent codes of C_2. This process is iterated on each subcodes : $C_1 \rightarrow C_2 \rightarrow \cdots \rightarrow C_J \rightarrow C_1 \rightarrow \cdots$

6 Simulation results

Two GLD codes have been tested for an additive white Gaussian noise channel (AWGN). The modulation is a

[1] It can be shown that turbo codes can be described as a particular case of GLD codes, where the interleaver acts only on information bits.

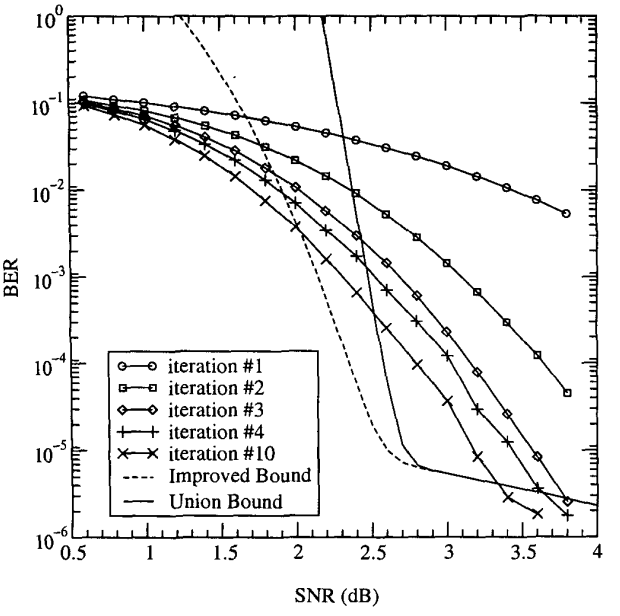

Figure 6: Simulated performance and analytical bounds of the GLD code built from the $(15, 11)$ Hamming code, length $N = 420$, total rate $R = 0.466$, AWGN channel.

BPSK with symbols equal to $\pm\sqrt{2RE_b}$, where E_b is the average energy per information bit. We used the forward-backward algorithm [6] on the syndrome trellis of the constituent codes as a SISO decoder for our iterative scheme.

The first code, suitable for mobile radio transmissions or small frame systems, has length $N = 420$, and $K = 196$. Its constituent code is the (15,11,3) BCH code, and it is a two-level GLD code. Its performance is shown in Figure 6 for different iteration steps, and compared to the Union Bound and the improved bound described in section 4.

The second code, suitable for deep space communications or image transmissions, has length $N = 65534$. Figure 7 shows its BER versus the decoding iteration number. This code achieves zero error probability at $1.8dB$ with a rate $R = 0.67$. Its performance is 0.72dB away from the capacity limit (1.08dB for $R = 0.67$ and a BPSK input).

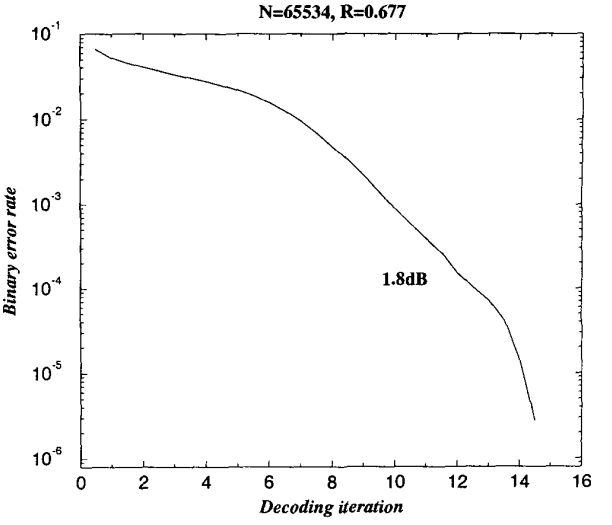

Figure 7: Performance of the GLD code built from the $(31, 26)$ Hamming code, length $N = 65534$, total rate $R = 0.677$, AWGN channel.

References

[1] R.G. Gallager: Low-density parity-check codes, MIT Press, 1963.

[2] M. Sipser and D.A. Spielman: "Expander codes", 1994.

[3] R.M. Tanner: "A recursive approach to low complexity codes", *IEEE Trans. on Information Theory*, Vol. IT-27, Sept 1981.

[4] C. Berrou, A. Glavieux, P. Thitimajshima : "Near Shannon limit error-correcting coding and decoding : turbo-codes," *Proceedings of ICC'93*, Genève, pp. 1064-1070, Mai 1993.

[5] T. M. Duman, M. Salehi, "New Performance Bounds for Turbo Codes", *IEEE Trans. on Communications*, Vol. 46, No 6, June 1998, pp. 717-723.

[6] L.R. Bahl, J. Cocke, F. Jelinek, J. Raviv, "Optimal decoding of linear codes for minimizing symbol error rate" *IEEE Trans. on Inf. Theory*, Vol. 20, pp. 284–287, March 1974.