

A GENETIC-ALGORITHM BASED DECODER FOR LOW DENSITY PARITY CHECK CODES

A.G. SCANDURRA[†], A.L. DAI PRA[‡], L. ARNONE[†], L. PASSONI[†], J. CASTIÑEIRA MOREIRA[†]

*School of Engineering, Universidad Nacional de Mar del Plata- UNMDP,
Juan B. Justo 4302 (7600) Mar del Plata, Argentina*

[†] *Department of Electronics
scandu@fi.mdp.edu.ar*

[‡] *Department of Mathematics
daipra@fi.mdp.edu.ar*

Abstract. This paper presents a Genetic-Algorithm based decoder for a medium-sized Low Density Parity Check code (GAMD decoder). The main advantage of the proposed GAMD decoder is that no information on the noise level transmission channel is required, an essential condition for the well-known sum product algorithm. The proposed methodology combines a Genetic Algorithm stage with a meta-decision process. Genetic Algorithms were selected due to their capacity to solve this type of multiple minimum. Encouraging results were reached when comparing the Bit Error Rate (BER) performance of the proposed algorithm with that of the traditional sum-product decoding algorithm. The performance of the proposed decoder is very close to that of the optimal sum-product decoder, with the additional benefit of not requiring channel information (signal-to-noise ratio). In order to improve Bit Error Rate performance and/or reduce the complexity of the proposed decoder, the fitness function and parameters of the GA can be optimized.

Keywords— LDPC codes, Genetic Algorithm, Sum-product Algorithm.

I. INTRODUCTION

One of the main issues in communication theory is the design of a coding technique rendering reliable transmissions over noisy channels. Since Shannon's prediction (Shannon, 1948), there have been different approaches to accomplishing limit performance for a reliable transmission over a non-reliable channel. Low Density Parity Check (LDPC) codes appeared as a very suitable coding technique that, under some conditions, can yield a Bit Error Rate (BER) performance close to the Shannon's limit by fractions of a dB. LDPC codes were invented by Gallager (1963) and later rediscovered by MacKay and Neal (1997), becoming one of the most powerful error correction techniques known in these days.

A LDPC code is a linear block code defined by a very sparse parity check matrix H . The decoding algorithm is easily understood by means of a graphic representation called bipartite graph. In this graphic representation, the decoding procedure is seen as the interchange of probabilistic information between symbol (or bit) nodes and parity check nodes. The

relationship between the bits of a code vector is determined by the parity check matrix H . A given code vector satisfies the whole set of parity check conditions described in the parity check matrix H .

When these codes are decoded using Gallager's iterative probabilistic decoding method, also known as the sum-product algorithm or belief propagation algorithm, their empirical BER performances are found to be excellent (MacKay, 1999; Richardson and Urbanke, 2001). This is true when the length of the code vector is large enough. The decoding algorithm is an iterative decoding procedure that depends on the knowledge of the noise level in the channel. MacKay and Hesketh (2003) investigated the dependence of the performance of a LDPC code on both the assumed and actual noise levels of a binary symmetric channel and a Gaussian channel, respectively.

Genetic Algorithms (Goldberg, 1989) are search algorithms that apply operations from natural genetics to guide the trek through a search space. GAs have theoretically and empirically proven to provide robust search capability in complex spaces, offering a valid approach to a problem requiring efficient and effective search.

GAs have been previously applied to communication network design (Daijin and Sunha, 1999), VLSI layout (Valenzuela and Wang, 2002), maximal distance codes (Dontas and De Jong, 1990), to name a few examples.

In this work, we propose a new approach to a LDPC decoding algorithm based on a hybrid system that combines a Genetic Algorithm with a Meta-Decision Process (Bonissone, 2003). As the main advantage of the proposed decoding scheme, we emphasize that no knowledge of the signal to noise ratio present in the channel is necessary.

II. GENERAL CONSIDERATIONS

A. LDPC codes

LDPC codes (McKay, 1999) are a powerful class of linear block codes characterized equivalently by a generator matrix G ($k \times n$ dimension) used to encode a message vector m ($k \times 1$ dimension) into a code vector x ($n \times 1$ dimension), or by its corresponding parity check matrix H ($(n-k) \times n$ dimension), which is such that any code vector satisfies the syndrome condition $H \circ x = 0$.

The design of these codes is based on the construction of this parity check matrix \mathbf{H} , which is a sparse matrix that meets some conditions to provide the system with optimum Bit Error Rate (BER) performance.

A given message vector \mathbf{m} is converted into a code vector \mathbf{x} by performing the matrix operation $\mathbf{x} = \mathbf{G}^T \circ \mathbf{m}$. Both the message vector \mathbf{m} and the code vector \mathbf{x} are defined over the binary field, *i.e.*, vectors with components taken from a discrete alphabet $\{0,1\}$. Digital data transmission is more conveniently carried out using polar format. In said format, bits are transmitted sending signals taken from a discrete alphabet $\{-1,+1\}$, so that, commonly, bit 0 is assigned a -1 signal, and bit 1 is assigned a +1 signal. Then, the code vector \mathbf{x} converts into the signal vector \mathbf{s} . After being transmitted, and in the presence of additive Gaussian noise \mathbf{n} , the signal \mathbf{s} converts into a received vector $\mathbf{y} = \mathbf{s} + \mathbf{n}$. Thus, vectors \mathbf{m} and \mathbf{x} components are taken from a discrete alphabet, whereas the received vector \mathbf{y} components are taken from the set of real numbers y_i in the interval $[-\infty, \infty]$.

As well-known, the aim of the decoding algorithm for a given block code is to find the vector \mathbf{d} , considered as an estimation of the transmitted vector \mathbf{x} , able to satisfy the following condition:

$$\mathbf{H} \circ \mathbf{d} = \mathbf{0} \quad (1)$$

The LDPC sum-product decoding algorithm (Gallager, 1963; MacKay and Neal, 1997) makes an estimation of the A Posteriori Probability (APP) of each symbol as a function of the received symbol and the properties of the channel. In this sense, the decoding algorithm does require to know the signal-to-noise ratio in the channel.

B. Genetic Algorithm

GAs are general purpose search algorithms whose principles lie on natural genetics. GAs can be applied to solve problems in which the objective function is discontinuous, non-differentiable, stochastic, or highly nonlinear.

GA maintains a population of individuals that evolve according to rules of selection and genetic operators, such as reproduction, crossover and mutation. GA begins with a population that consists in randomly created individuals (possible solutions) and repeatedly modifies this population "evolving" towards an optimal solution.

Each individual in the population is assigned a measure of its fitness in the environment. Reproduction focuses its attention on high fitness individuals, thus exploiting the available fitness information. Crossover and mutation perturb those individuals, providing general heuristics for exploration. Although simplistic from a biologist's viewpoint, these algorithms are complex enough to provide robust (good performance across a variety of problem types) and powerful adaptive search mechanisms. The adaptive behaviour of the GA depends on this feedback to drive the population

towards better overall performance (Koza, 1992, Michalewicz, 1992).

Therefore, considering a particular problem, an ad-hoc evaluation or fitness function must be devised.

As already known, GAs' performance is a function of parameter settings (Dontas and De Jong, 1990; Schaffer *et al.*, 1989). The number of possible parameter assignments rules out a factorial design to fix the best parameter setting.

III. GENETIC ALGORITHM META-DECISION DECODER (GAMD)

The proposed GAMD decoder uses the parity check matrix \mathbf{H} to recover the decoded vector \mathbf{d} embedded in a vector \mathbf{y} , which is the transmitted code vector \mathbf{x} corrupted by the Additive White Gaussian Noise (AWGN).

This algorithm can be implemented in three steps. These are the syndrome calculation step, the GA application step, and lastly the meta-decision step.

A. The syndrome calculation step

In this first step the proposed algorithm constructs a modified received vector \mathbf{y}_{hard} , which is basically a hard decision vector of the received vector \mathbf{y} . In this step, the components y_i of the received vector, essentially real numbers, are converted into binary values (taken from the discrete alphabet $\{0,1\}$) using a fixed threshold. Then, the decoding algorithm verifies if this modified vector satisfies the syndrome condition (Eq.1). If the modified received vector \mathbf{y}_{hard} meets this condition, then a valid code vector \mathbf{d} is obtained by $\mathbf{d} = \mathbf{y}_{hard}$. Otherwise, the decoder makes the following two steps.

B. Genetic Algorithm step

The algorithm begins creating an initial population of V candidates: a set of individual vector \mathbf{v} with real components $v_i \in [0,1]$. To avoid an *a priori* reduction of the searching space, an initial random population is generated. A new 500 individual generation (children) is created through the following steps:

- Selects individuals (called parents) based on their fitness value (Eq. 2) through the selection function.
- The two individuals with the best fitness values survive for the next generation (*elite children*=2)
- The crossover fraction ($P_c=0.95$) specifies the fraction of the population, other than elite children, that are made up of crossover children.
- To complete the new generation, mutation children are created by introducing random changes with a given probability rate ($P_m=0.01$) to a single parent.

The algorithm stops when the limit of 25 generations is reached.

The GA parameters were heuristically selected to optimize its performance.

The solution provided by the GA algorithm is a \mathbf{z} vector. The process involves the following fitness function:

$$Fitness = \sum_{j=1}^m b_j + \sum_{i=1}^n |z_i - \tilde{y}_i| \quad (2)$$

where b_j are the components of the vector \mathbf{b} defined as:

$$\mathbf{H} \circ \mathbf{z} = \mathbf{b}$$

In Eq.2, m is the number of rows of the parity check matrix \mathbf{H} , and n the code vector length. Vector \mathbf{z} is obtained as follows:

$$z_i = \begin{cases} 1 & \text{if } \tilde{y}_i > v_i \\ 0 & \text{if } \tilde{y}_i \leq v_i \end{cases} \quad (3)$$

where:

$$\tilde{y}_i = \frac{1}{1 + e^{-2y_i}} \quad (4)$$

The aim of the sigmoid function described in Eq.4, applied component wise, is to map the components of the received vector \mathbf{y} into $[0,1]$. Hence the received vector \mathbf{y} format agrees with the candidate vectors \mathbf{v} format.

The fitness function measures both a component wise distance between the candidate vector and the received one, and also how close the candidate vector satisfies the syndrome condition (Eq.1).

A set of q decoded vectors \mathbf{z} is obtained applying GA algorithm q times, where q is an arbitrary integer value heuristically optimized (partial solutions).

These q vectors are candidates for the following step of the decoding process, which consists in applying the meta-decision process.

C. The Meta-decision process

The Meta-decision process reduces the results scattering of the GA, which comes from the randomness of the initial population.

The z vectors are a set of possible solutions obtained at q GA runs, next a meta-decision stage generates the final solution, *i.e.*, a decoded vector \mathbf{d} .

This process applies the majority logic, a well-known procedure utilized in the error correction decoding theory. This procedure performs a component wise decision over the z candidate vectors, setting each final component d_i as the bit state of higher frequency. In order to perform this meta-decision process, parameter q was heuristically selected equal to $q = 15$. Simulations were performed with Matlab®.

IV. DECODING COMPLEXITY

The comparative analysis of decoding complexity is a rather difficult task, mainly because the proposed GAMD decoding algorithm and the traditional sum-product decoding operate quite differently.

The complexity of the sum-product algorithm is a function of the code parameters. This algorithm is essentially sequential. If n is the code vector length and also the column size of \mathbf{H} matrix, and t is the average value of ones per column for that matrix, the sum-product decoding algorithm involves the calculation of $6nt$ products and $5nt$ sums (average) per iteration.

The GA inherent random feature does not allow specifying an analytical expression due to its complexity.

The tested GAMD decoder is of higher computing complexity if compared to the Sum Product Algorithm

in medium-sized LDPC codes. However this methodology allows a parallel design process (letting simultaneous GA runs and just concentrating on a very fast meta-decision process) which would strongly decrease processing time.

V. RESULTS

In this preliminary study the BER performance of an irregular (60, 30) LDPC code is evaluated (Richardson *et al.*, 2001). The simulations are carried out for the Additive White Gaussian Noise Channel. The comparison between the proposed GAMD decoder and the traditional sum-product decoding algorithm (SPDA) (Gallager, 1963; MacKay and Neal, 1997) is presented. The fact that the latter, unlike the GAMD decoder, needs information of the channel signal-to-noise ratio is noted. The simulations were performed with 300 words of 60 bits each, considering different noise power levels, expressed as ratio E_b/N_o [dB] (E_b : average bit energy, N_o : noise power spectral density) in Fig.1 and Fig.2.

Figure 1 shows the performance of the sum-product decoding algorithm of an irregular (60, 30) LDPC to be taken as a comparison reference for the BER performance of the proposed GAMD decoder. Results are evaluated for 2, 6, 10 and 14 iterations, showing that the BER performance does not improve significantly after 14 iterations (SPD-14). As widely known, this algorithm is a Maximum A Posteriori (MAP) Algorithm, hence its performance is considered optimal.

Table 1 lists the number of bit errors yielded by a 300-codeword transmission for different values of noise standard deviation σ . The number of errors is calculated over the message bits.

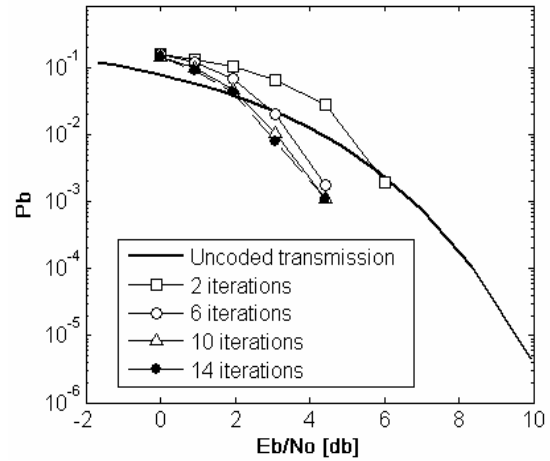


Fig. 1. BER performance of an irregular (60, 30) LDPC code decoded using the sum-product algorithm for different number of iterations

Table 1. Errors over 300 words

σ	0.6	0.7	0.8	0.9	1.0
GAMD	10	113	382	812	1181
SPD 14	10	72	376	828	1287
GAMD: GA meta-decision decoder SPD 14 : Sum-product decoding algorithm with 14 iterations					

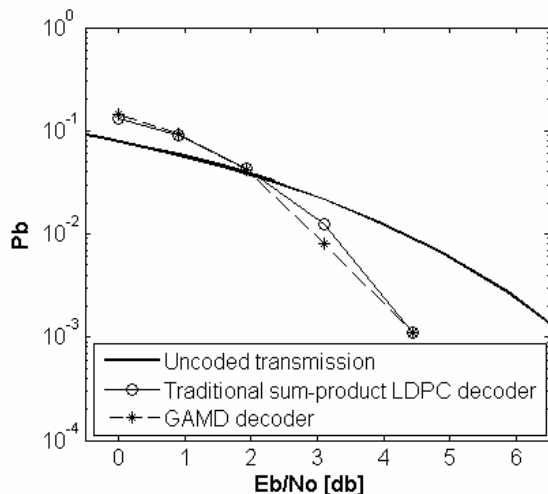


Fig. 2. BER performance of two decoders for an irregular (60, 30) LDPC code.

Figure 2 shows the BER performance of these two decoding algorithms.

VI. CONCLUSIONS

The GAMD decoder is tested on a medium-sized LDPC code. As shown in Table 1, for high noise levels ($\sigma = 0.9$, $\sigma = 1$) the GAMD decoder performance is better than the SPD-14 is. Besides, regarding $E_b/N_0 \approx 3$ dB, the traditional sum-product decoding algorithm performs slightly better than the GAMD decoder.

The main advantage of the proposed GAMD decoder is that noise level transmission channel information needs not be known, an essential condition for sum product algorithm.

As further work, the BER performance of the proposed GA based decoder for LDPC codes will be studied for larger length codewords, *i.e.*, for larger parity check matrices. Another interesting feature of the GAMD decoder is the feasibility of implementing parallel computing, taking advantage of the independence of the algorithm q rounds.

The tested GAMD decoder exhibits higher computing complexity (approximately six fold) than the Sum Product Algorithm in medium-sized LDPC codes. Notwithstanding this, this methodology allows a parallel design process (letting simultaneous GA runs and just concentrating on a very fast meta-decision process) which would strongly decrease processing time.

The independence of the proposed decoding algorithm with respect to the channel characteristics makes it perfectly suitable to be applied not only to other types of channels, such as the fading channel, but also to other codes. This is proposed as a further research field.

VII. REFERENCES

Bonissone P., "Soft Computing and Meta-heuristics: using knowledge and reasoning to control search

and vice-versa," *Proc. SPIE Applications and Science of Neural Networks, Fuzzy Systems and Evolutionary Computation V*, S. Diego, CA, 133--149, (2003).

Daijin K., and A. Sunha, "A MS-GS VQ Codebook Design for Wireless Image Communication Using Genetic Algorithms," *IEEE Trans. Evol. Comput.*, **3**, 35-52 (1999).

Dontas K. and K. De Jong, "Discovery of Maximal Distance Codes Using Genetic Algorithms," *Proc. of the 2nd International IEEE Conference on Tools for Artificial Intelligence*, Herndon, VA, 805-811, (1990).

Gallager R.G., *Low-Density Parity-Check Codes*, MIT Press, Cambridge, MA, (1963).

Goldberg, D.E., *Genetic Algorithms in Search, Optimization, and Machine Learning*, Addison-Wesley, New York, (1989).

Koza, J.R., *Genetic Programming: On the Programming of Computers by Means of Natural Selection*, MIT Press, Cambridge, MA, (1992).

MacKay D.J.C. and R.M. Neal, "Near Shannon Limit Performance of Low-Density Parity-check Codes," *IEE Elect. Lett.*, **33**, 457-458, (1997).

MacKay D.J.C. and C.P. Hesketh, "Performance of low density parity check codes as a function of actual and assumed noise levels," *Electronic Notes in Theoretical Computer Science*, **74**, 1-8, (2003).

MacKay D.J.C., "Good error-correcting codes based on very sparse matrices" *IEEE Trans. Inform. Theory*, **45**, 399-431, (1999).

Michalewicz, Z., *Genetic algorithms+data structures = evolution programs*, Springer-Verlag, Berlin, (1992).

Richardson T., A. Shokrollahi and R. Urbanke, "Design of capacity approaching irregular Low-Density Parity-Check codes," *IEEE Trans. Inform. Theory*, **47**, 619-637, (2001).

Richardson T. and R. Urbanke, "The capacity of Low-Density Parity-Check codes under message-passing decoding," *IEEE Trans. Inform. Theory*, **47**, 599-618, (2001).

Schaffer J., J. Caruana, L. Eshelman and R. Das, "A study of control parameters affecting online performance of genetic algorithms for function optimization," *Proc. of the Third international Conference on Genetic Algorithms*, San Mateo, CA, 51-60, (1989).

Shannon C.E., "A mathematical theory of communication," *Bell Syst. Tech. J.*, **27**, 379-423, (1948).

Valenzuela, C.L. and P.Y. Wang, "VLSI placement and area optimization using a genetic algorithm to breed normalized postfix expressions," *IEEE Trans. Evol. Comput.*, **6**, 390-401, (2002).

Received: September 21, 2005.

Accepted: March 22, 2006.

Recommended by Guest Editors C. De Angelo, J. Figueroa, G. García and J. Solsona