Bit-Flipping Algorithm for Joint Decoding of Correlated Sources over Noisy Channels

Fernando Pujaico Rivera
Department of Communications
Faculty of Electrical and Computer Engineering
State University of Campinas, Brazil
Email: fpujaico@decom.fee.unicamp.br

Jaime Portugheis
Division of Telecommunication Technology
Faculty of Technology
State University of Campinas, Brazil
Email: jaime@ft.unicamp.br

***Abstract*—This paper proposes a bit-flipping decoding algorithm suitable for decoding multiple correlated sources in a noisy environment. The correlated sources are separately encoded without knowledge of the correlation and are transmitted over orthogonal noisy channels. Short LDGM codes are used for encoding. The correlation of the sources is exploited at the joint decoder to achieve better performance. Correlation models commonly used in the literature are applied to obtain the performance of the proposed algorithm. However, differently from the literature, scenarios with a large number of sources are considered. Additionally, the correlation values of the models are generated randomly. Although the proposed algorithm do not make any estimation of log-likelihood ratios, it can be used in conjunction with a fusion rule for solving an instance of the CEO problem.**

***Index Terms*—Multiple correlated sources, large scale sensor networks, bit-flipping decoding algorithms.**

I. Introduction

The efficient transmission of information from separate correlated sources to a remote sink node is a challenging problem [1], [2]. This problem becomes more challenging when the number of sources is very large. This is the case of a large scale wireless sensor network [3]. In this case it is reasonable to assume that the sensor nodes transmit their observations over orthogonal noisy channels which implies that theoretical limits can be achieved with separation of source and channel coding [4]. One approach to achieve these limits would be to use complex Slepian-Wolf source codes [5] which might be impractical for large scale networks [3]. Another approach is to use only channel coding and exploit the correlation of the sources at the decoder.

In this paper, we adopt this later approach of using only channel coding and also consider a scenario with a decoder of manageable complexity [3]. In order to cope with this complexity we propose a bit-flipping decoding algorithm suitable for decoding multiple correlated sources. A simple correlation model was adopted to obtain the performance of the proposed algorithm. The adopted models are similar to those described in [6],[7],[2]. However, we consider scenarios with a large number of sources (up to 100) and also with correlation values of the model generated randomly. Moreover, if one of the correlation values is intentionally selected to be equal to one, the scenario can be considered as an instance of decoding with multiple side information [8].

The rest of the paper is organized as follows. Section II gives a brief description of the system model. The proposed algorithm for joint decoding is explained in Section III where we also describe a modified version of Sipser-Spielman bit-flipping decoding[9]. In Section IV we compare the obtained performance results. Finally, Section V makes some final remarks and conclusions.

II. System Model

Fig. 1 shows the model of the distributed encoding and joint decoding system. Each source generates a binary sequence $\mathbf{u}^i = (u_1^i, \cdots, u_k^i, \cdots, u_K^i)$, $i = 1, 2, \ldots, I$, of length K. We assume that

$$u_k^i = u_k^0 \oplus e_k^i \tag{1}$$

where $\oplus$ denotes the modulus two summation, the variables u_k^0 are identically distributed, the e_k^i are mutually independent binary variables with $P(e_k^i = 1) = p_i$ and they are independent from u_k^0. The correlation between u_k^0 and u_k^i, $corr(u_k^0, u_k^i)$, is equal to $(1-2p_i)$ and the correlation between u_k^i and u_k^j, $\forall i \neq j$,$corr(u_k^i, u_k^j)$, is equal to $(1 - 2p_{ij})$, where $p_{ij} = p_i + p_j - 2p_ip_j$.

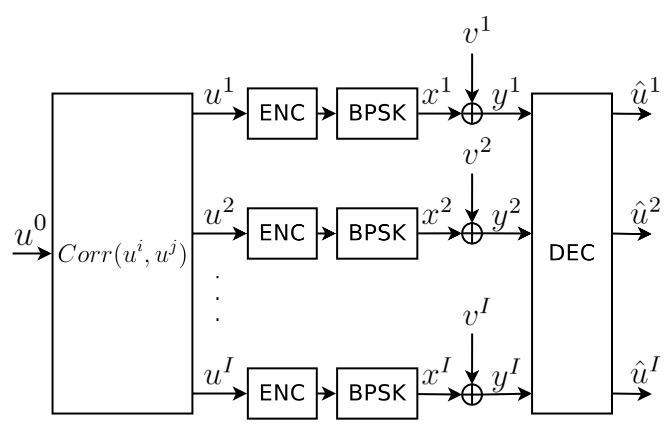

Fig. 1. System Model.

In the following we drop the upper indexes “i” for simplicity. Each source is encoded according to a Low Density Generator Matrix (LDGM) code of rate $R = K/N$, where N is its block length [10]. A codeword at the output of each encoder is mapped into a signaling vector $\mathbf{x} = (x_1, \cdots, x_n, \cdots, x_N)$, $x_n \in \{\pm 1\}$, and is transmitted on an additive white Gaussian noise channel. At the receiver, the input to the decoder is given by $\mathbf{y} = (y_1, \cdots, y_n, \cdots, y_N)$, where $y_n = x_n + v_n$, $n = 1, \cdots, N$ with v_n the additive noise with zero mean and variance $\sigma^2 = (2RE_b/N_0)^{-1}$, E_b the energy per information bit and N_0 the one-sided noise spectral density. After applying hard-decision on $\mathbf{y}$, a vector $\mathbf{z} = (z_1, \cdots, z_n, \cdots, z_N)$ is obtained.

III. Bit-Flipping Decoding

In this Section we describe two different bit-flipping algorithms for obtaining estimations $\hat{\mathbf{u}}^i = (\hat{u}_1^i, \cdots, \hat{u}_k^i, \cdots, \hat{u}_K^i)$, $i = 1, 2, \ldots, I$, of the information transmitted by all sources in Fig. 1. In both algorithms all decoders work in parallel. In the first algorithm none of the decoders receive side information from others. This algorithm will be referred as *independent decoding*. On the other hand, in the second algorithm all decoders receive side information from others. This algorithm will be called *joint decoding*.

A. Independent Decoding

In the following we also drop the upper indexes "i" for simplicity. Let $H = [h_{m,n}]$ be the parity-check matrix of a LDGM code. The m-th syndrome component is given by the check $s_m = \sum_n h_{m,n} z_n$ (mod 2). Denote the set of bits that participates in check m and the set of checks in which bit n participates by $\mathcal{N}(m) = \{n | h_{m,n} = 1\}$ and $\mathcal{M}(n) = \{m | h_{m,n} = 1\}$, respectively.

In BF decoding, the decoder computes flipping function values

$$E_n = \sum_{m \in \mathcal{M}(n)} s_m, \tag{2}$$

and flips all bits for which $E_n \geq \delta$. New syndromes are recomputed and the process is repeated until all syndromes equal zero. As in [11], we set $\delta = \max_n E_n$.

Sipser and Spielman proposed hard-decision algorithms based on a simple criterion: a bit is flipped if it has more non-null syndrome values than null ones [12]. The first algorithm flips the bits sequentially and the second algorithm flips them in parallel. A possible implementation of the first algorithm was given in [12].

They also show that allowing a limited amount of negative progress improves the sequential algorithm. In this paper we use a modified version of their parallel algorithm allowing any amount of negative progress. The flipping function is then given by

$$E_n = \sum_{m \in \mathcal{M}(n)} (2s_m - 1). \tag{3}$$

We describe now the modified version: the Parallel Hard Bit-Flipping (PHBF) Decoding Algorithm [9].

PHBF Algorithm.

1) Initialize $k = 0$ and $\mathbf{z}^{(k)} = (z_1^{(k)}, \cdots, z_n^{(k)}, \cdots, z_N^{(k)}) = \mathbf{z}$.
2) Compute the syndrome values $s_m = \sum_n h_{m,n} z_n^{(k)}$ (mod 2) for $m = 1, \cdots, (N-K)$. If all values are zero then stop decoding since $\mathbf{z}^{(k)}$ is the output codeword.
3) For $n = 1, \cdots, N$ compute the flipping function E_n in (3).
4) Identify the set of bits $\{n^*\}$ for which $n^* = \arg\max_n E_n$.
5) Flip all bits in $\{n^*\}$ in parallel. The flipped hard-decision vector is stored in $\mathbf{z}^{(k+1)}$.
6) If a maximum number of iterations is not reached, set $k = k + 1$ and go to Step 2). Otherwise stop and $\mathbf{z}$ is the output codeword.

B. Joint Decoding

The joint decoding algorithm is obtained by modifying the flipping function of the independent algorithm. The new flipping function, T_n^i, will be the function E_n^i given in (3) combined with a joint flipping function, C_n^i. The joint function depends on the bits received by other decoders and also on correlations between sources. The new function for the i-th decoder is then

$$T_n^i = E_n^i + \lfloor \beta C_n^i \rfloor. \tag{4}$$

where $\lfloor b \rfloor$ is the greatest integer less than or equal to b and C_n^i is given by

$$C_n^i = \sum_{\substack{a=1 \\ a \neq i \\ z_n^i = z_n^a}}^{I} \frac{E_n^a \; corr(i,a)}{I-1} - \sum_{\substack{a=1 \\ a \neq i \\ z_n^i \neq z_n^a}}^{I} \frac{E_n^a \; corr(i,a)}{I-1} \tag{5}$$

The parameter β in (4) is to be optimized. If $\beta = 0$ joint decoding becomes independent decoding. Note that introducing the rounding operation $\lfloor b \rfloor$ leads to a flipping function that operates only with integers.

IV. Numerical Results

Numerical results were obtained for scenarios with the number of sources $I = 3$, $I = 10$ and $I = 100$. They were obtained for the uncoded case and for the case where the same short LDGM code with parameters $K = 204$ and $N = 306$ was used for encoding all sources. The maximum number of iterations in all algorithms was selected to be equal to 15. For scenarios with $I = 3$, $I = 10$ and $I = 100$, the optimal values obtained for parameter β were 1.0, 2.0 and 3.0, respectively.

Figures 2, 3 and 4 compares the performance of independent and joint decoding for scenarios with $I = 3$,$I = 10$ and $I = 100$ correlated sources, respectively. The average performance of all decoders is also shown in the Figures. As can be seen in the Figures, for all scenarios, joint decoding performs better than independent decoding for all sources.

For all three scenarios, the set of values for the probabilities p_i, $i = 1, 2, \ldots, I$, were generated randomly according to an

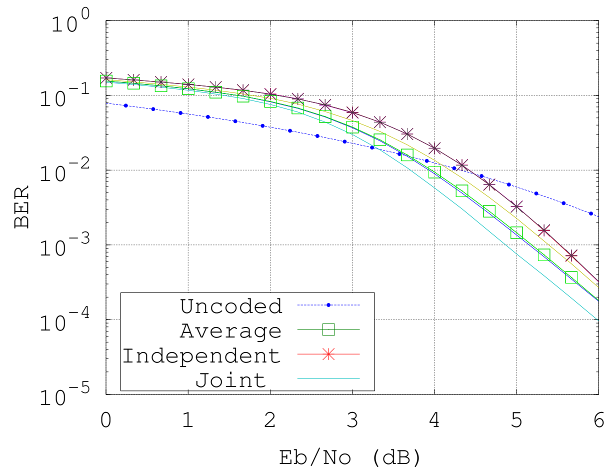

Fig. 2. Bit error rate (BER) for a system with 3 sources using Independent Decoding ("Independent") and Joint Decoding ("Joint"). The average bit error rate for Joint Decoding ("Average") is also shown.

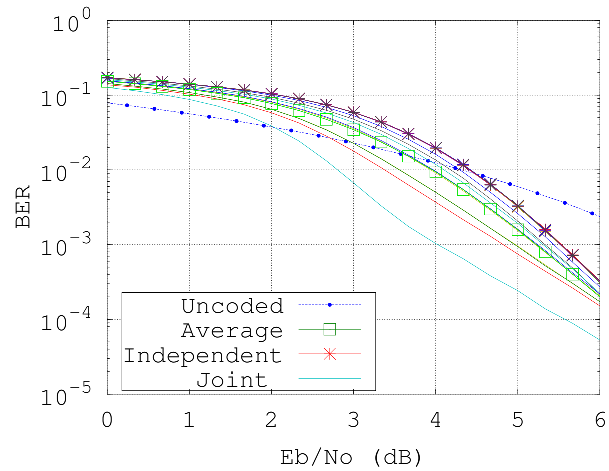

Fig. 3. BER for a system with 10 sources using Independent Decoding ("Independent") and Joint Decoding ("Joint"). The average bit error rate for Joint Decoding ("Average") is also shown.

uniform distribution ranging from $p_i = 0.001$ and $p_i = 0.999$. For the scenarios with $I = 3$ and $I = 10$, the lowest value ($p_1 = 0.001$) was intentionally selected for one of the sources and the others were generated randomly. For $I = 3$, $p_2 = 0.258$ and $p_3 = 0.142$. Table I shows the set of values for the probabilities p_i for the scenario with $I = 10$ explicitly. For the scenario with $I = 100$ the smallest generated value was $p_i = 0.036181$ and the greatest value was $p_i = 0.95469$. It is worth noting that, for all scenarios, the decoder for the source with highest absolute correlation value $corr(u^0, u^i)$ has the best performance.

TABLE I
PROBABILITIES p_i FOR FOR A MODEL WITH 10 SOURCES

p_1	p_2	p_3	p_4	p_5
0.001	0.097	0.358	0.070	0.592
p_6	p_7	p_8	p_9	p_{10}
0.295	0.904	0.245	0.822	0.188

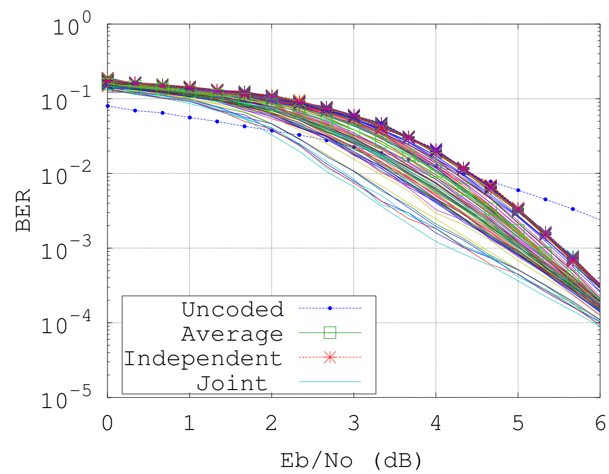

Fig. 4. BER for a system with 100 sources using Independent Decoding ("Independent") and Joint Decoding ("Joint"). The average bit error rate for Joint Decoding ("Average") is also shown.

V. Final Remarks and Conclusions

It was observed in the simulation results that the source with highest absolute value of the correlation $corr(u^0, u^i)$ is recovered with the lowest value of BER. By comparing the results in Figures 3 and 4 for the decoders with best performances, it can be seen that for E_b/N_0 values higher than 4 dB the scenario with 10 sources leads to slightly better results. This can be explained by the fact that the highest absolute value of correlation $corr(u^0, u^i)$ in this scenario is equal to 0.998 whereas for the scenario with 100 sources is equal to 0.928.

It is not difficult to show that the higher the absolute value of the correlation $corr(u^0, u^i)$, the greatest the mutual information $I(u^0, u^i)$. However, in the system model of Section II, source u_0 plays the role of a common hidden source [2]. This implies that in a real situation one cannot estimate the mutual information $I(u^0, u^i)$, $i = 1, 2, \ldots, I$. Let $\mathbf{u}_{j\neq i} = (u^1, \ldots, u^{i-1}, u^{i+1}, \ldots, u^I)$ be a vector representing all source variables excluding the i-th source. It is reasonable to argue that the source with the highest value of the joint mutual information $I(u^i, \mathbf{u}_{j\neq i})$ should be recovered with the lowest value of BER. Instead of calculating $I(u^i, \mathbf{u}_{j\neq i})$ we obtained the sum

$$MI(i) = \sum_{\substack{j=1 \\ j\neq i}}^{I} I(u^i, u^j) \tag{6}$$

for many sets of randomly generated correlation values. For all sets, the highest value of $MI(i)$ indicated the source recovered with the lowest value of BER.

We have also obtained performance results for all three scenarios described in Section IV using different sets of values for the probabilities p_i, $i = 1, 2, \ldots, I,$. It was observed that the performance is highly dependent on the choice of these sets for the source model. However, for the scenarios of Figures 3 and 4, the lowest value of BER is always attained by the sources whose correlation values were intentionally selected to be approximately equal to one. Moreover, by using 10 sources instead of 3 the performance becomes better. These scenarios can be considered as an instance of decoding with multiple side information [8].

If one is interested in estimating the single hidden binary source u^0, the problem considered in this paper becomes an instance of the binary central estimating officer (CEO) problem [6]. Although the proposed algorithm do not make any estimation of log-likelihood ratios, by taking into account decoding error probabilities, it can be used together with a fusion for solving this problem.

We have proposed a bit-flipping decoding algorithm suitable for decoding multiple correlated sources over noisy channels. The complexity of the proposed algorithm is adequate to be used with a large number of correlated sources. Performance results were obtained for hard-decision bit-flipping decoding algorithms. The proposed algorithm can be easily extended for decoding with soft-decisions.

ACKNOWLEDGMENT

This work was supported in part by The State of Sao Paulo Research Foundation (FAPESP) under grant 2012/22641-5.

REFERENCES

[1] A. Abrardo, G. Ferrari, and M. Martalò,*On non-cooperative block-faded orthogonal multiple access schemes with correlated sources*, IEEE Trans. Commun., vol. 59, no. 7, pp. 1916 1926, July 2011.

[2] G. Ferrari, M. Martalò, A. Abrardo, and R. Raheli,*Orthogonal Multiple Access and Information Fusion: How Many Observations Are Needed?*, in Proc. Inform. Theory and Applications Workshop (ITA), UCSD, San Diego, CA, USA, February 2012.

[3] J. Barros and M. Tüchler, *Scalable Decoding on Factor Trees: A Practical Solution for Wireless Sensor Networks*, IEEE Trans. Commun., vol. 54, no. 2, pp. 284-294, Feb. 2006.

[4] J. Barros and S. D. Servetto,*Network information flow with correlated sources*, IEEE Trans. Inform. Theory, vol. 52, no. 1, pp. 155170, January 2006.

[5] D. Slepian and J. K.Wolf, *A coding theorem for multiple access channels with correlated sources*, Bell Syst. Tech. J., vol. 52, no. 7, pp. 10371076, 1973.

[6] J. Haghighat, H. Behroozi, and D.V. Plant, *Iterative joint decoding for sensor networks with binary CEO model*, IEEE 9th Workshop on Signal Processing Advances in Wireless Communications, July 2008, Recife, Brazil.

[7] K. Kobayashi, T. Yamazato and M. Katayama,*Decoding of Separately Encoded Multiple Correlated Sources Transmitted over Noisy Channels*, IEICE Trans. Fundamentals, vol. E92-A , no 10,pp 2402-2410, October 2009.

[8] S. Shamai and S. Verdu,*Capacity of channels with uncoded side information*, European Trans. Telecommun., vol. 6, no. 5, pp. 587600, September-October 1995.

[9] F. Pujaico, Hard decision algorithms for LDGM Codes, (In portuguese), Master Thesis, State University of Campinas, Brazil, p. 55, 2011, Available: http://www.bibliotecadigital.unicamp.br.

[10] J. F. Garcia-Frias e W. Zhong, *Approaching Shannon performance by iterative decoding of linear codes with low-density generator matrix, IEEE Commun. Lett.*, v. 7, n. 6, pp. 266-268, Junho 2003.

[11] Y. Kou, S. Lin and M. Fossorier, *Low-density parity-check codes based on finite geometries: a rediscovery and new results*, IEEE Trans. Inform. Theory, vol. 47, pp. 2711-2736, Nov. 2001.

[12] M. Sipser and D. Spielman, *Expander codes*, IEEE Trans. Inform. Theory, vol. 42, pp. 1710-1722, Nov. 1996.