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

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 and long $LDGM$ 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. 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 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 [6], [7] 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 [8],[9],[2]. However, we also consider scenarios with a large number of sources (up to 100). 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 [10].

The rest of the paper is organized as follows. Section II gives a brief description of the system model. Section III show the bit-flipping algorithm. The proposed algorithm for joint decoding is explained in Section IV where we also describe a modified version of Sipser-Spielman bit-flipping decoding [11]. In Section V we compare the obtained performance results. Finally, Section VI makes some final remarks and conclusions.

II. SYSTEM MODEL

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

$$u^m = u^0 \oplus q^m, \tag{1}$$

where $\oplus$ denotes the modulus two summation, the source u^0 is binary and identically distributed, the sources q^m are binaries and mutually statistically independent with $P(q^m = 1) = p_m$ and statistically independent from u^0. The correlation between u^0 and u^m is equal to $corr(u^0, u^m) = (1 - 2p_m)$ and the correlation between u^m and u^a, $\forall a \in \{1, 2, \ldots, M\} \wedge m \neq a$, is equal to $corr(u^m, u^a) = (1 - 2p_{ma})$, where $p_{ma} = p_m + p_a - 2p_m p_a$.

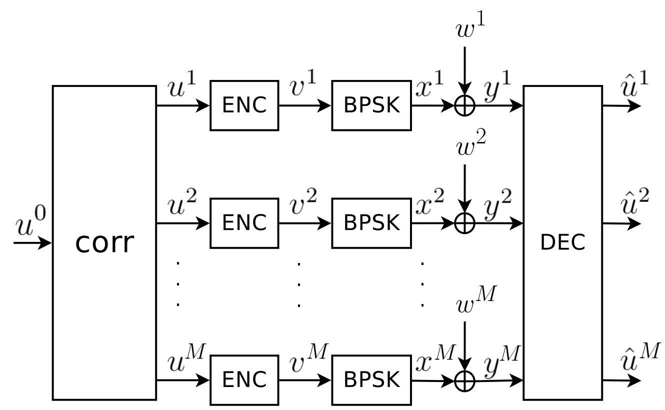

Fig. 1. System Model.

Each source vector $\mathbf{u}^m$ is encoded into $\mathbf{v}^m$, using a systematic Low Density Generator Matrix ($LDGM$) code [12], $G = [I\ P]$, where P is a $K \times (N - K)$ sparse matrix (and, therefore, the code rate is $r = K/N$). A codeword at the output of each encoder is mapped with a binary phase shift keying (BPSK) into a signaling vector $\mathbf{x}^m = (x_1^m, \cdots, x_n^m, \cdots, x_N^m)$, where $x_n^m \in \{\pm 1\}$, $\forall n \in \{1, 2, \ldots, N\}$, and is transmitted on an additive white Gaussian noise (AWGN) channel. At the receiver, the input to the decoder is given by $\mathbf{y}^m = (y_1^m, \cdots, y_n^m, \cdots, y_N^m)$, where $y_n^m = x_n^m + w_n^m$ being w_n^m an additive white Gaussian noise with zero mean and variance $\sigma_m^2 = (2E_s/N_{0m})^{-1}$, E_s is the energy per transmit

Manuscript received XXXX XX, 2014; revised XXXXX XX, 2014.

symbol and N_{0m} the one-sided noise spectral density for a m-th source. After applying hard-decision on $\mathbf{y}^m$, a vector $\mathbf{z}^m = (z_1^m, \cdots, z_n^m, \cdots, z_N^m)$ is obtained. The decoding of this vector is made using the parity check matrix $H = [P^T\ I]$, where the super index T indicates the transpose of the matrix P. After error correction decoding over $\mathbf{z}^m$, the K first bits of $\mathbf{z}^m$ represent an approximation $\hat{\mathbf{u}}^m$ of $\mathbf{u}^m$.

III. Bit-Flipping Decoding

In this section we describe bit-flipping algorithm [6], [7], for obtaining estimations $\hat{\mathbf{u}}^m = (\hat{u}_1^m, \cdots, \hat{u}_k^m, \cdots, \hat{u}_K^m)$, $\forall m \in \{1, 2, \ldots, M\}$, of the information transmitted by all sources u^m in Fig. 1. In the algorithm all decoders work in parallel and none of the decoders receive side information from others. This algorithm will be referred as *independent decoding.*

A. Independent Decoding

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

In Bit-Flipping (BF) decoding [6], [7], the decoder computes flipping function values

$$i_n = \sum_{l \in \mathcal{M}(n)} s_l, \tag{2}$$

and flips all bits z_n for which $i_n \geq \delta$. New syndromes are recomputed and the process is repeated until all syndromes equal zero, as in [6]. 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 [13]. They also show that allowing a limited amount of negative progress is improved the sequential algorithm.

In this paper we use a modified version of Sipser and Spielman algorithm allowing any amount of negative progress. The flipping function is then given by

$$i_n = \sum_{l \in \mathcal{M}(n)} (2s_l - 1) \tag{3}$$

for to get the reliability vector $\mathbf{i} = (i_1, ..., i_n, ..., i_N)$. We describe now the modified version: the Parallel Hard Bit-Flipping (PHBF) Decoding Algorithm [11].

PHBF Algorithm.

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

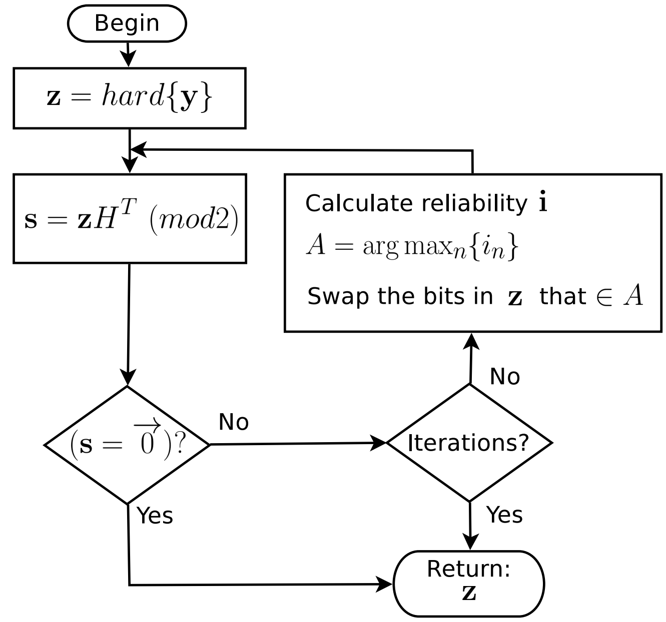

Fig. 2. Flow diagram of Parallel Hard Bit-Flipping algorithm.

The Fig. 2 show a simplify form the last algorithm.

IV. Distributed Bit-Flipping Decoding

Of similar way of Section III, in this section will be presented an algorithm for obtaining estimations $\hat{\mathbf{u}}^m$, of the information transmitted by all sources u^m in Fig. 1. with the difference that in this algorithm all decoders receive side information from others. This algorithm will be called *joint decoding*.

A. Understanding the Joint Decoding

For understanding the joint decoding algorithm, we first should remember that in the Section III, the validation of each receive vector $\mathbf{z}^m = (z_1^m, \ldots, z_n^m, \ldots, z_N^m)$ is analyzed using the flipping function $\mathbf{i}^m = (i_1^m, ..., i_n^m, ..., i_N^m)$. Thus, in the m-th source, each couple $\{z_n^m, i_n^m\}$ represent a reliability i_n^m of that n-th bit in $\mathbf{z}^m$ be equal to z_n^m. In independent decoding for to decide over $\mathbf{z}^m$ a horizontal analysis is employed for to calculate each element i_n^m in $\mathbf{i}^m$ $\forall n \in \{1, 2, \ldots, N\}$. In the joint decoding a second vertical analysis is defined for to obtain the joint reliability vector $\mathbf{j}^m = (j_1^m, ..., j_n^m, ..., j_N^m)$, this second analysis use vertically the couples $\{z_n^m, i_n^m\}$ $\forall$ $m \in \{1, 2, \ldots, M\}$ for to get each j_n^m in $\mathbf{j}^m$, this is side information. Thus, in terms of human behavior the independent reliability $\mathbf{i}^m$ represent the reliance that one person have in your own solution $\mathbf{z}^m$, and the joint reliability $\mathbf{j}^m$ represent the reliance that the others have over my solution. Here, we can say that each people use a weighting factor β different for to estimate the total reliability $\mathbf{t}^m = \mathbf{i}^m + \beta\mathbf{j}^m$, this reliability will be the new flipping function. A person that only listening to itself will use only $\mathbf{i}^m$ for to estimate $\mathbf{t}^m$, a person that don't have reliance in itself will use only $\mathbf{j}^m$ for to estimate $\mathbf{t}^m$.

The calculus of couple $\{z_n^m, j_n^m\}$ of n-th bit in $\mathbf{z}^m$, is similar to case of one student, of a classroom of M students, that go out of test with one question of true or false type, and find to get the joint reliability j_n^m, for your answer z_n^m, knowing the answers and independent reliabilities of the others students. Thus, this student decide that the joint reliability j_n^m is a weighted average of independents reliabilities, of the others students, with the reliability that he have in the answer of other students. In this sense the reliability in the answer of other students will be proportional to correlation between these students. Here is necessary to note that for calculating the joint reliability j_n^m, is necessary first homologate the independents reliability the others students. This homologation is motivated by the case in that, in the a-th student, the couple $\{z_n^a, i_n^a\} = \{1, +A\}$ and in the m-th student the couple $\{z_n^m, i_n^m\} = \{0, +B\}$. Thus, the couple of a-th student should be homologate to $\{z_n^a, i_n^a\} = \{0, -A\}$, for that both independent reliabilities talk about your decision of that your bit in study be equal to zero. In general the homologation rule will be that a swap in bit value z_n^a will be link to sing change in i_n^a.

B. Joint Decoding

The joint decoding algorithm is obtained by modifying the flipping function of the independent algorithm. The new flipping function, $\mathbf{t}^m = (t_1^m, ..., t_n^m, ..., t_N^m)$, will be the function $\mathbf{i}^m = (i_1^m, ..., i_n^m, ..., i_N^m)$ given in (3) combined with a joint flipping function, $\mathbf{j}^m = (j_1^m, ..., j_n^m, ..., j_N^m)$. The joint function depends on the bits received by other decoders and also on correlations between sources. The new flipping function $\forall n \in \{1, 2, ..., N\}$, in the m-th decoder is

$$t_n^m = i_n^m + \lfloor \beta j_n^m \rfloor. \tag{4}$$

where $\lfloor \alpha \rfloor$ is the greatest integer less than or equal to α and j_n^m is given by

$$\lambda_{m,a,n} = \begin{cases} -1 & , se \quad z_n^m \neq z_n^a \\ 0 & , se \quad m = a \\ +1 & , se \quad z_n^m = z_n^a \end{cases}, \tag{5}$$

$$j_n^m = \frac{1}{M-1} \sum_{a=1}^{M} \lambda_{m,a,n} \; i_n^a \; corr(u^m, u^a), \tag{6}$$

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

Joint PHBF Algorithm.

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

The Fig. 3 show a simplify form the last algorithm.

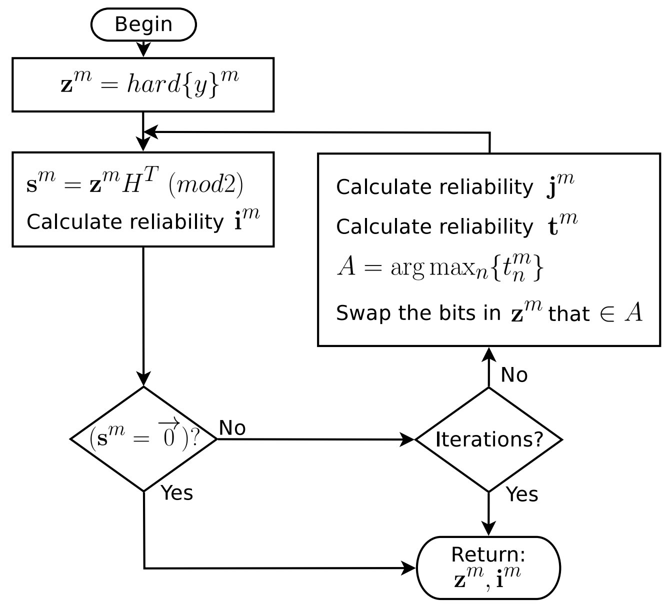

Fig. 3. Flow diagram of distributed Parallel Hard Bit-Flipping algorithm.

C. Joint Decoding Applied to the CEO Problem

As is seen in [14] a good idea for to get a estimation $\hat{u}^0$ in the model of the Fig. 1, is to use a criterion of maximum a posteriori (MAP), making the quotient

$$\phi = \frac{P(u^0 = 1 | u^1 u^2 \ldots u^M)}{P(u^0 = 0 | u^1 u^2 \ldots u^M)}, \tag{7}$$

$$\phi = \frac{P(u^1|u^0=1)P(u^2|u^0=1)\ldots P(u^M|u^0=1)}{P(u^1|u^0=0)P(u^2|u^0=0)\ldots P(u^M|u^0=0)}, \tag{8}$$

for to get the $\hat{u}^0$ approximation with

$$\hat{u}^0 = \begin{cases} 1 & \text{if } \phi > 1 \\ 0 & \text{if } \phi \leq 1 \end{cases}, \tag{9}$$

This procedure will be used to obtain the numerical results.

V. Numerical Results

Numerical results were obtained for scenarios with $M = 3$, $M = 5$, $M = 10$ and $M = 100$ sources. The limit for the bit error rate [12] in independent decoding with systematic $LDGM$ matrix is also shown. In the codewords all the systematic bit nodes have degree $\hat{d}$ and each one of the coded bit nodes has degree 1.

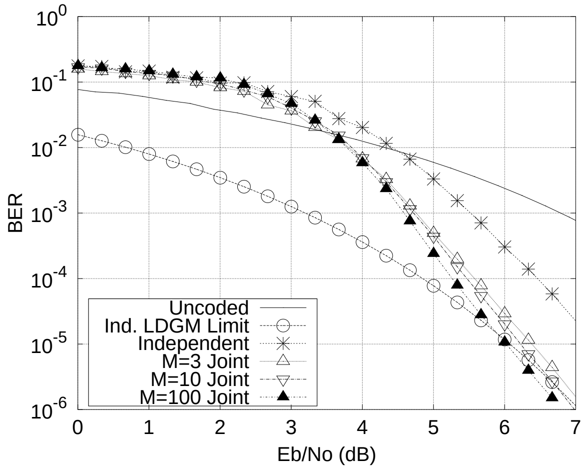

Fig. 4. Bit error rate (BER) for a system with 3, 10 and 100 sources using independent decoding ("Independent") and joint decoding ("Joint"). The limit for the bit error rate in independent decoding with systematic $LDGM$ matrix ("Ind. LDGM limit") is also shown.

Fig. 4 compare the average performance of independent and joint decoding for scenarios with $M = 3$, $M = 10$ and $M = 100$ correlated sources. A short length systematic $LDGM$ code with parameters $K = 204$, $N = 306$ and $\hat{d} = 5$ were used for encoding the sources. In the three scenarios, the set of values for the probabilities $p_m = 0.1$, $\forall m \in \{1, 2, \ldots, M\}$. The optimal value obtained for the β parameter was 0.6. The maximum number of iterations was selected to be equal to 15. As can be seen in the figure, for all scenarios, joint decoding performs better than independent decoding. The scenario with most number of sources have the better performance, but the improved grows slowly with M.

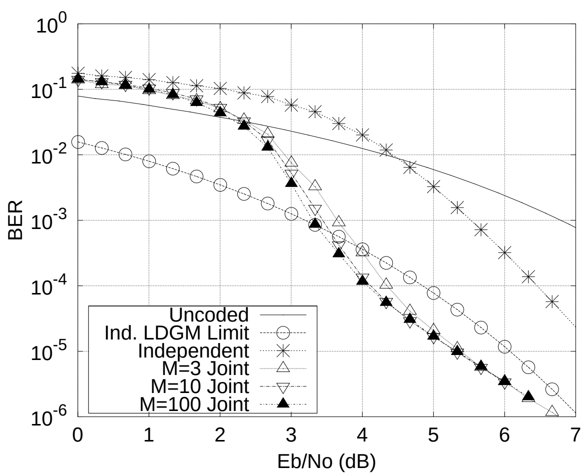

Fig. 5. Bit error rate (BER) for a system with 3, 10 and 100 sources using independent decoding ("Independent") and joint decoding ("Joint"). The limit for the bit error rate in independent decoding with systematic $LDGM$ matrix ("Ind. LDGM limit") is also shown.

Fig. 5 compare a case similar to Fig. 4 with the same value of K,N, β parameter and with the difference that we have a set of values for the probabilities $p_m = 0.005$, $\forall m \in \{1, 2, \ldots, M\}$.

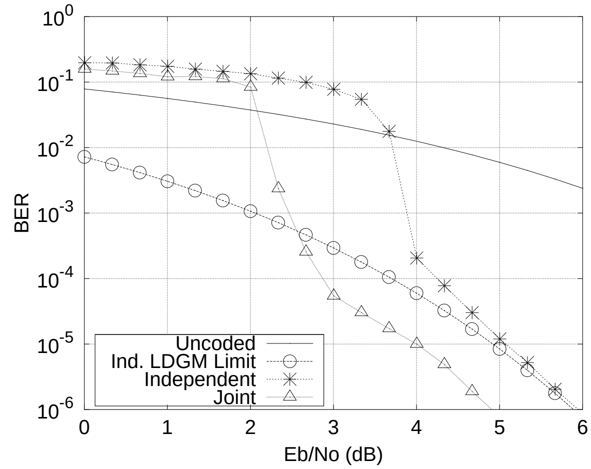

Fig. 6. BER for a system with 3 sources using independent decoding ("Independent") and joint decoding ("Joint"). The limit for the bit error rate in independent decoding with $LDGM$ matrix ("Ind. LDGM limit") is also shown.

Fig. 6 compare the performance of independent and joint decoding for $M = 3$ correlated sources. A long length systematic $LDGM$ code with parameters $K = 15000$, $N = 30000$ and $\hat{d} = 9$ were used for encoding the sources. The set of values for the probabilities are $p_1 = p_2 = p_3 = 0.005$. The maximum number of iterations was selected to be equal to 300. The β parameter has been optimized for each value of $E_{bm}/N_{0m} = E_b/N_0$. The Table I shows these values. In the Figure can be seen that joint decoding performs is ever better than independent decoding.

TABLE I
OPTIMAL β PARAMETER AND CORRESPONDING E_b/N_0 IN THE FIG. 6

E_b/N_0	β	E_b/N_0	β	E_b/N_0	β
2.0	0.7	2.333	0.7	2.666	0.6
3.0	0.4	3.333	0.4	3.666	0.375
4.0	0.3	4.333	0.2	4.666	0.2

Fig. 7 compare the performance of independent and joint decoding for $M = 5$ correlated sources. In this figure was used the same systematic $LDGM$ matrix and the maximum number of iterations that in the Fig. 4. The set of values for the probabilities $p_m, \forall m \in \{1, 2, \ldots, 5\}$, are distributed according the Table II, being the minimum value of p_m equal to 0.0001 and the maximum value 0.2. The value used in the β parameter was 0.6. As can be seen in this figure, there are 5 different curves named "joint", this happen because each source u^m have a different error probability p_m, thus the performance of each curve is different. In all sources the joint decoding performs are ever better than independent decoding. The figure also show a average performance of all joint decoding, this curve is plotted with $\triangle$.

Fig. 8 compare the performance in the estimation $\hat{u}^0$ of u^0 for the $M = 5$ correlated sources seen in Fig. 7. The algorithm used in the estimation is described in the Section IV-C. In the figure, can be seen the estimation CEO for many cases. Exist the case in that the data used for the

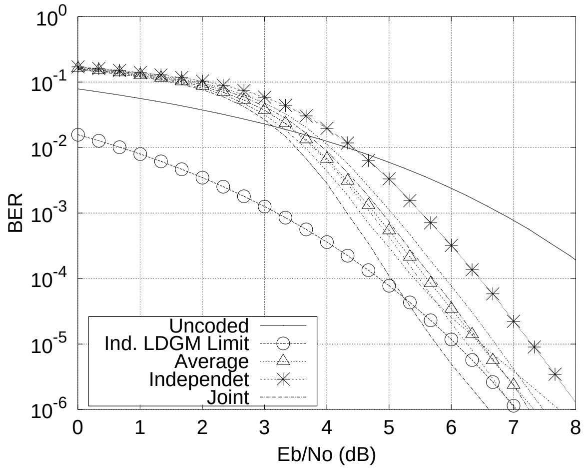

Fig. 7. BER for a system with 5 sources using independent decoding ("Independent"), joint decoding ("Joint") and the average bit error rate for joint decoding ("Average"). The limit for the bit error rate in independent decoding with $LDGM$ matrix ("Ind. LDGM limit") is also shown.

TABLE II
PROBABILITY p_m AND CORRELATION IN THE DISTRIBUTED SOURCE MODEL IN THE FIG. 7.

	p_1	p_2	p_3	p_4	p_5
p_m	0.0001	0.05	0.1	0.15	0.2
$corr(u^0, u^m)$	0.9998	0.9	0.8	0.7	0.6

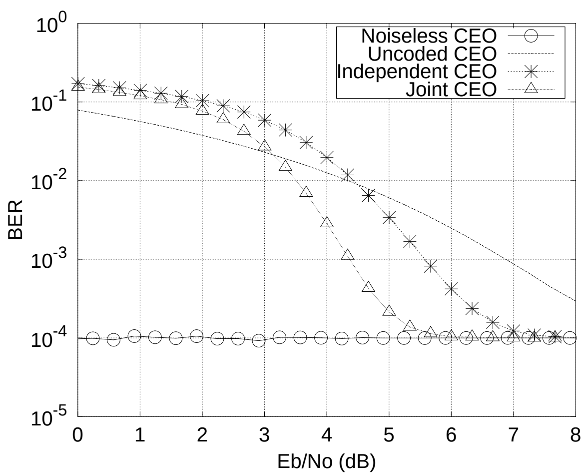

Fig. 8. BER in the estimation $\hat{u}^0$ of u^0 for a system with 5 sources using the data after independent decoding ("Independent CEO"), joint decoding ("Joint CEO"), uncoded data ("Uncoded CEO") and noiseless channel data ("Noiseless CEO").

CEO estimation is obtained after the joint decoding (Joint CEO), also the case when the information is obtained after the independent decoding (Independent CEO) or the case in that the information used was obtained after to receive uncoded data (Uncoded CEO). Also we have the ideal case in that the data used passes through noiseless channels (Noiseless CEO). The better performance is reached for the "Joint CEO" estimation, but this happen only after $E_{bm}/N_{0m} = 3.1dB$. The reason for this is that between all sources, the source u^1 have better performance and this source also is better than uncoded performance after $E_{bm}/N_{0m} = 3.1dB$. Known also that the correlation between u^1 and u^0 is very close to one, the source u^1 is dominant in the CEO estimation.

VI. Final Remarks and Conclusions

It was observed in the simulation results that the source u^m, with highest absolute value of the correlation $corr(u^0, u^m)$ is recovered with the lowest value of BER in the joint algorithm.

By comparing the results in Fig. 5 and Fig. 6 is interesting show that in both graphics the joint decoding curves accompanying the curve of independent LDGM limit described in [12]. This is interesting because highlights the possibility of existence of a similar limit for the case of distributed sources.

By comparing the results in Fig. 7 for the decoder with best performance, it can be seen that for E_b/N_0 values higher than 4 dB leads to significantly better results. It is not difficult to show that the source u^m of higher absolute value of correlation $corr(u^0, u^m)$, is the same that of the greatest mutual information $I(u^0, u^m)$. 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 $I(u^0, u^m)$.

We have also obtained performance results using different sets of values for the probabilities p_m. It was observed that the performance is highly dependent on the choice of these sets for the source model. However, for the scenario of Fig. 7, the lowest value of BER was always attained by the sources whose correlation values were intentionally selected to be approximately equal to one. These scenarios can be considered as an instance of decoding with multiple side information [10].

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 [8]. Although the proposed algorithm do not make any estimation of log-likelihood ratios, The data in out of the decoders is used together with the rule of the Section IV-C 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 ——— ——— ——— ——— ——— ——— ——— ———

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. 155-170, 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. 1037-1076, 1973.

[6] 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.

[7] Gallager, R.G., *Low-density parity-check codes*, Information Theory, IRE Transactions on , vol.8, no.1, pp.21,28, January 1962

[8] 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.

[9] 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.

[10] S. Shamai and S. Verdu,*Capacity of channels with uncoded side information*, European Trans. Telecommun., vol. 6, no. 5, pp. 587-600, September-October 1995.

[11] ——— ——— ——— ——— ——— ——— ——— ——— ——— ——— ———

[12] 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.

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

[14] Chair, Z.; Varshney, P.K., "Optimal Data Fusion in Multiple Sensor Detection Systems," Aerospace and Electronic Systems, IEEE Transactions on , vol.AES-22, no.1, pp.98,101, Jan. 1986