



HAL
open science

Blind frame synchronization of product codes based on the adaptation of the parity check matrix

Rodrigue Imad, Sébastien Houcke, Christophe Jego

► To cite this version:

Rodrigue Imad, Sébastien Houcke, Christophe Jego. Blind frame synchronization of product codes based on the adaptation of the parity check matrix. IEEE International Conference on Communications (ICC 2009), Jun 2009, Dresde, Germany. 10.1109/ICC.2009.5198792 . hal-02118509

HAL Id: hal-02118509

<https://hal.science/hal-02118509>

Submitted on 3 May 2019

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Blind frame synchronization of product codes based on the adaptation of the parity check matrix

Rodrigue Imad, Sebastien Houcke and Christophe Jego
Institut TELECOM; TELECOM Bretagne; UMR CNRS 3192 Lab-STICC
Technopôle Brest Iroise, CS 83818, 29238 Brest Cedex 3
Université Européenne de Bretagne, France
Email: {rodrigue.imad, sebastien.houcke, christophe.jego}@telecom-bretagne.eu

Abstract—We present in this paper a blind frame synchronization method based on the adaptation of the parity check matrix of the code. The blind synchronizer is initially based on the calculation of the Log-Likelihood Ratios (LLR) of the syndrome elements, obtained using the parity check matrix of the code. Before applying our synchronization procedure, we propose in this paper to rearrange the parity check matrix of the code according to the reliability of the received symbols as previously introduced for decoding linear block codes with high density parity check matrix. Simulation results show that the Frame Error Rate (FER) curves obtained after applying the proposed synchronization method to product codes are very close to the ones with perfect synchronization. In addition to its powerful synchronization properties, the main advantage of the proposed synchronization algorithm is its capability of being introduced as a part of the decoder so that no additional material is required for the synchronization step.

I. INTRODUCTION

In most challenging digital communication systems, error correcting coding is becoming increasingly important. Several studies [1][2][3] have demonstrated that actual codes such as LDPC codes, turbocodes and product codes, are very powerful and are able to decode heavily corrupted sequences. However, these works were based on the assumption of coherent detection, which may be less than realistic for many digital communication systems. Conventional frame synchronization is achieved by adding to the transmitted symbols a synchronization sequence known by the receiver. This sequence is detected at the reception by running the received symbols through a correlator [4][5]. In order to achieve good performance on low signal to noise ratios, traditional synchronization methods should increase the length of the inserted training sequence. This reduces the spectral efficiency of the transmission especially when codes of small sizes are used. Another solution to improve the system performance is to take advantage of the code structure by considering frame synchronization jointly with the decoding as in [6][7]. Rather than placing the synchronization bits in a separate header, the authors of [8] propose to place them in a midamble. These bits are then encoded as part of the data sequence.

We are interested in developing blind methods of frame synchronization wherein no additional sequence is added to the coded one. We have previously proposed in [9] a blind frame synchronization method that is based on a Maximum A Posteriori probability (MAP) approach in the sense of

maximizing the probability that a position t corresponds to the correct synchronization moment, knowing the received samples. This method has been first introduced for a Binary Symmetric Channel (BSC) [10] and then generalized for a Gaussian Channel [11]. It has been also applied in an Interleaved Division Multiple Access (IDMA) context [12] and it gave promising results. Applied to codes having sparse parity check matrices such as LDPC codes and convolutional codes, the previous method presents good performance. Moreover, it has been shown in [9] that this synchronization method outperforms another technique of blind frame synchronization introduced in [13]. However, these methods are not suitable for codes having a high number of nonzero elements in their parity check matrix, case of linear block codes as Bose-Chaudhuri-Hocquenghem (BCH) codes, Reed-Solomon (RS) codes and product codes.

A decoder using the Belief Propagation (BP) algorithm can be used for decoding linear block codes. This algorithm is considered to be the reference decoding algorithm of LDPC codes having a sparse parity check matrix. However, it is not applied to BCH codes or RS codes, for which the parity check matrix is not sparse. A modified version of this algorithm has been introduced in [14] (Adaptive Belief Propagation, ABP) for RS codes and modified in [15] (m-ABP) for BCH and product codes, where the authors propose an adaptation step of the parity check matrix of the code according to the reliability of the received symbols. In this paper, we use the general idea of the above modified decoding algorithm and propose a new blind frame synchronization method especially adapted for product codes.

This paper is organized as follows. Section II briefly presents product codes and the m-ABP decoding algorithm proposed in [15]. In Section III we introduce the proposed method of blind frame synchronization of BCH codes and BCH product codes. Simulation results are presented in Section IV. Finally, Section V concludes the work.

II. PRODUCT CODES AND THE M-ABP ALGORITHM

Being first introduced by Elias in 1954 [16], a product code is a multidimensional code constructed from elementary codes. Let us consider two systematic linear block codes $C_1(n_1, k_1)$ and $C_2(n_2, k_2)$. Fig 1 shows the implementation of the product code $C(n_1n_2, k_1k_2)$. k_2 data blocks of length k_1 bits each, are

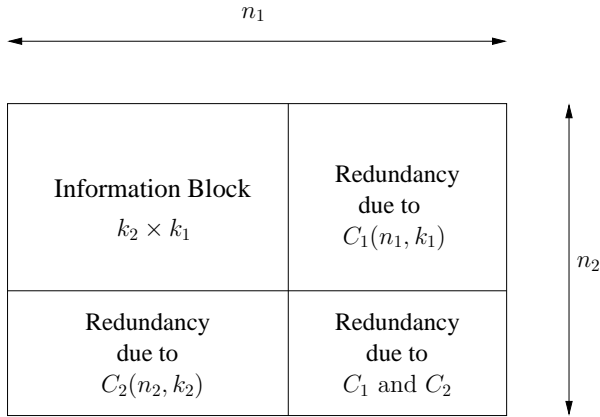


Fig. 1. Product Code constructed from two systematic linear block codes

encoded by encoder C_1 . As a result, k_2 codewords of length n_1 bits each are obtained and placed in a two-dimensional vector. Each column of this vector is then encoded by encoder C_2 to produce a codeword of length n_2 . The final result is a product code of length $n_1 \times n_2$ and dimension $k_1 \times k_2$. Note that the dimension of a code represents the length of the data block contained in this code.

Product codes are usually decoded by the Chase-Pyndiah algorithm [3]. It is a turbo-like decoding algorithm where the rows and columns decoders of the product code exchange extrinsic information on the bits. In [14], the authors propose the ABP algorithm for decoding RS codes. This method adapts the parity check matrix at each iteration of the BP algorithm according to the reliabilities of the received symbols, in order to sparsify the columns of the parity check matrix associated with the least reliable bits.

The ABP algorithm being very complex to implement, a modified version (m-ABP) was proposed by Jegou and Gross in [15]. The m-ABP algorithm applied to BCH product codes is presented in the following paragraph.

First of all, the symbols of each row (column) of the received product code are rearranged according to their reliability. In other words, we order these symbols by their absolute values. Then, the n_r columns of H corresponding to the n_r least reliable symbols are reduced to obtain an identity squared matrix. This is done by applying the Gaussian elimination method. The aim of this procedure is to reduce the number of nonzero elements in the part of the parity check matrix that is associated to the least reliable symbols. The next step of the m-ABP decoding algorithm is the traditional iterative BP algorithm: each row and column of the product code is decoded by a BP algorithm where iterations are called local iterations. Rows and columns decoders exchange extrinsic information via an iterative process. Each iteration of the turbo process is called a global iteration. Finally hard decisions are computed during the last global iteration.

Once applied to BCH product codes, the m-ABP algorithm outperforms the standard ABP algorithm and provides similar performance in terms of bit error rate to the classical Chase-

Pyndiah algorithm [15]. In the next section of this paper we describe how to use a part of the above decoding algorithm in our proposed blind frame synchronization method.

III. PROPOSED METHOD OF BLIND FRAME SYNCHRONIZATION OF BCH AND PRODUCT CODES

Inspired by the m-ABP decoding algorithm, we introduce in this section a new method of blind frame synchronization to be applied to linear block codes as BCH and product codes. In this study, we consider that the transmitter is sending a binary sequence of codewords and is using a Binary Phase Shift Keying (BPSK) mapping. The propagation channel is corrupted by a quasi-static additive white Gaussian noise. For a given code of rate

$$R = \frac{n_c - n_r}{n_c} \quad (1)$$

corresponds a parity check matrix H of size $n_r \times n_c$, where n_c represents the length of a codeword and n_r the number of parity relations.

Let $b(i) = \pm 1$ be the i^{th} coded and modulated bit to be transmitted. At the reception, the i^{th} received sample is given by:

$$r(i) = b(i - t_0) + w(i), \quad (2)$$

where t_0 is an integer representing the shift of the transmitted symbols due to the delay introduced by the propagation channel and $w(i)$ a white Gaussian noise. The received sequence of N samples can now be written as:

$$r = [r(1), \dots, r(N)]. \quad (3)$$

The main target of frame synchronization is to find the position of a codeword in the received sequence. In other words, we have to estimate the delay t_0 that we assume (without loss of generality) to be lower than n_c .

Let us consider a BCH code of length n_c bits. In order to apply the blind frame synchronization procedure, we consider a sliding synchronization window of length Kn_c . Fig. 2 shows three different positions for the sliding window corresponding to $t = 0, 1$ and t_0 .

At a position t and for each block of size n_c contained in the sliding window, we order its symbols according to their reliabilities and adapt the corresponding parity check matrix. Once this operation is done, we calculate the vector of Log-Likelihood Ratios (LLR) of the syndrome elements. This vector is written as:

$$L(S_t) = [L(S_t(1)), \dots, L(S_t(Kn_r))], \quad (4)$$

where $L(S_t(k))$ is the LLR of the k^{th} syndrome element. According to [17], $L(S_t(k))$ is proportional to:

$$\hat{L}(S_t(k)) = (-1)^{u_k + 1} \prod_{j=1}^{u_k} \text{sign}(r(t+k_j)) \min_{j=1, \dots, u_k} |r(t+k_j)|, \quad (5)$$

where u_k and k_j represent the number of ones in the k^{th} row of the parity check matrix of the code and the position of the

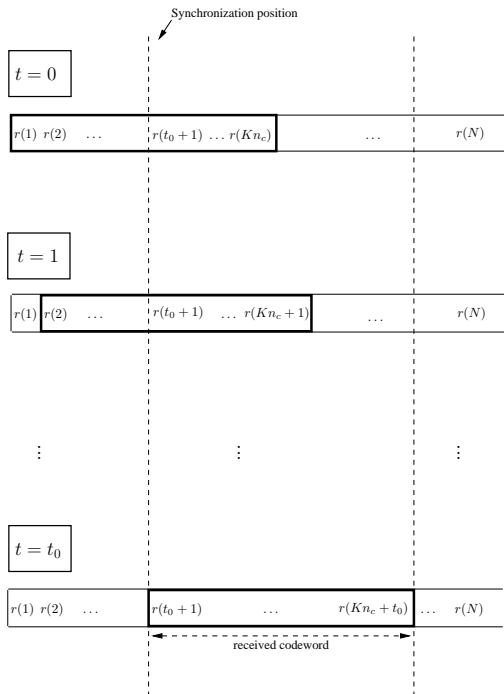


Fig. 2. Blind frame synchronization principle.

j^{th} non zero element in this k^{th} row, respectively. Having this, we compute

$$\hat{\phi}(t) = \sum_{k=1}^{K n_r} \hat{L}(S_t(k)) \quad (6)$$

and the frame synchronization position is estimated by:

$$\hat{t}_0 = \underset{t=0, \dots, n_c-1}{\operatorname{argmin}} \{ \hat{\phi}(t) \}. \quad (7)$$

In the case of product codes, we take advantage of their multidimensional structure and propose a synchronization method especially adapted for these codes. Therefore, consider a product code $C(n_1 n_2, k_1 k_2)$ constructed from two elementary systematic linear block codes $C_1(n_1, k_1)$ and $C_2(n_2, k_2)$. The product code being of length $n_c = n_1 n_2$, we consider a sliding synchronization window of length $K n_1 n_2$. For each block of size $n_1 n_2$ contained in the sliding window, we divide the proposed synchronization procedure into two parts.

First of all, we arrange the symbols in each row of the product code according to their reliabilities. Each permuted row causes the adaptation of the parity check matrix of code C_1 , as explained in the previous section. Then, for each row of the received product code we calculate the LLR of the syndrome elements as in (5) by using the adapted parity check matrix of code C_1 . Let $\hat{\phi}_1(t)$ be the sum of these syndrome elements. Each column of the product code being a codeword of C_2 , the second part of the proposed synchronization procedure is the same as the previous one but applied this time to the columns of the received code. The result is $\hat{\phi}_2(t)$, sum of the syndrome elements of the code columns, obtained using the adapted parity check matrix of code C_2 .

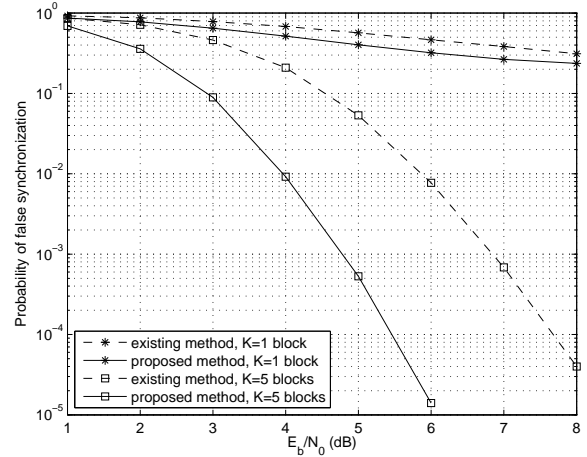


Fig. 3. Comparison between the proposed and existing methods applied to the extended BCH (32, 26) code.

Finally, we calculate a new criterion for the proposed method:

$$\hat{\phi}_p(t) = \hat{\phi}_1(t) + \hat{\phi}_2(t) \quad (8)$$

and the frame synchronization position is estimated by:

$$\hat{t}_0 = \underset{t=0, \dots, n_c-1}{\operatorname{argmin}} \{ \hat{\phi}_p(t) \}. \quad (9)$$

In addition to its powerful synchronization properties, the advantage of the above synchronization algorithm is its capability of being presented as a part of the m-ABP decoder. Indeed, once the parity check matrix of the component codes is rearranged for the synchronization procedure, it can be directly used by the m-ABP decoder. Furthermore, the calculation of (5) used for synchronization is equivalent to the first half iteration of the m-ABP algorithm. Therefore, using the above synchronization method for product codes does not require any additional materials then the ones initially used for decoding. The performance of the proposed synchronization method is evaluated in the next section of this paper.

IV. SIMULATION RESULTS

In order to study the performance of our synchronization method, we estimated the probability of false synchronization. The evaluation of this probability is realized by Monte Carlo simulation where for each configuration, the noise, information bits and the delay of the channel were randomly chosen.

In the case of cyclic codes and in order to keep the good performance of our synchronization technique, a pseudo-random interleaver should be used to permute the transmitted symbols. This interleaver should be known by the receiver so that the cyclic property of the code is retrieved after the synchronization procedure. Furthermore, the size of the synchronization window should be multiple of the interleaver length.

Let us designate by “existing method” the blind synchronization method proposed in [9]. Fig. 3 shows a comparison between the proposed and existing synchronization methods

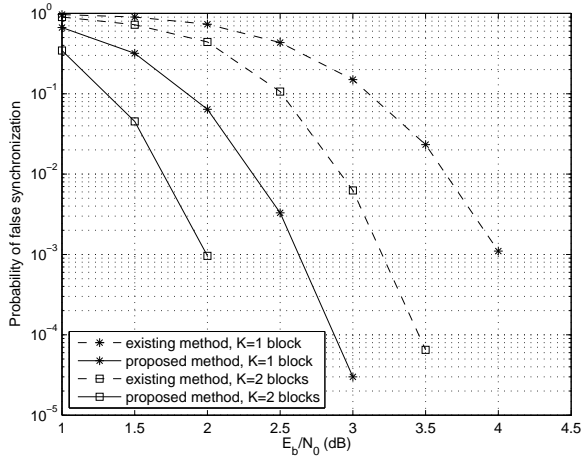


Fig. 4. Comparison between the proposed and existing methods applied to the $(32, 26)^2$ BCH product code.

once applied to the extended BCH $(32, 26)$ code. For a synchronization window of size $K = 1$ block, it is clear that both synchronization methods do not present good performance. This is due to the small size of the code (32 bits) and the low number of parity equations (6 equations). However and as shown in [10], the probability of false synchronization decreases with the length of the synchronization window K . For $K = 5$ blocks and at a probability of false synchronization equal to 10^{-3} , a gap of 2 dB is observed between the existing and proposed synchronization methods.

In the remaining of this paper, the only difference between the “existing” and “proposed” methods is the adaptation step of the parity check matrix to the symbols reliability.

Consider now the $(32, 26)^2$ BCH product code. This notation means that the product code C is constructed from two identical $(32, 26)$ extended BCH codes. Fig. 4 shows a comparison between the existing synchronization method and the proposed one. For the same length of the synchronization window, it is clear that the proposed method outperforms the existing one. A gain of around 1.3 dB is achieved at a probability of false synchronization equal to 10^{-3} .

Fig. 5 shows the performance of the proposed method applied to the $(16, 11)^2$ and the $(32, 26)^2$ BCH product codes. Although the $(16, 11)^2$ has 8 non zero elements in each line of H and the $(32, 26)^2$ has 16, the latter presents better performance for E_b/N_0 greater than 1.5 dB. This is due to the fact that code $(32, 26)^2$ is longer than code $(16, 11)^2$ (1024 bits instead of 256) and has a greater number of parity check equations (6 instead of 5). In this same figure, we also compared our results to the ones obtained with Massey’s synchronization approach, which locates a synchronization word periodically embedded in the binary data [5]. Even for a synchronization word of length $L = 20$ bits, our proposed method converges quicker than the one introduced by Massey. For the $(32, 26)^2$ product code and at a probability of false synchronization equal to

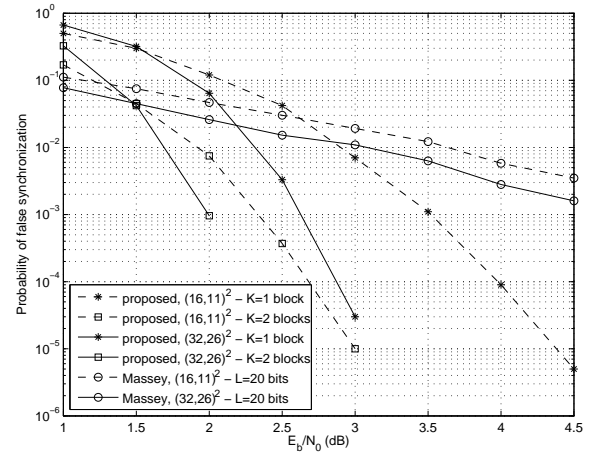


Fig. 5. The proposed synchronization method applied to the $(16, 11)^2$ and $(32, 26)^2$ BCH product codes and compared to Massey’s method.

10^{-3} , there is a gap of around 2 dB between the proposed method ($K = 1$ block) and the existing synchronization method of Massey ($L = 20$ bits).

In order to evaluate the robustness of the proposed synchronization method, we plotted the Frame Error Rate (FER) curves obtained by decoding product codes using the m-ABP algorithm, which is applied after achieving the blind frame synchronization step. For the m-ABP decoder, three local iterations and eight global iterations were done. Fig. 6 shows the FER curves obtained after decoding the $(16, 11)^2$ BCH product code. It is clear that the existing synchronization method is not suitable for product codes. However, a gap of only 0.6 dB is observed between the FER curves obtained after applying the proposed synchronization method ($K = 1$ block) and the ones of the perfect synchronization case. Furthermore, when $K = 2$ blocks and for E_b/N_0 greater than 1.5 dB, the curves of FER for the proposed method are almost the same as the ones obtained in the case of perfect synchronization.

In the case of the $(32, 26)^2$ BCH product code, Fig. 7 shows that even for $K = 1$ block, the FER curves obtained after applying the proposed synchronization method are the same as the ones of the perfect synchronization case.

V. CONCLUSION

In this paper, we have proposed a blind frame synchronization method based on the adaptation of the parity check matrix of the code. The proposed synchronization method clearly outperforms an existing method of blind frame synchronization previously introduced. Once the proposed synchronization method is applied to BCH product codes, simulation results have shown that the frame error rate curves obtained after decoding were almost the same as the ones of the perfect synchronization case. Another advantage of the proposed method is its capability of introducing the blind synchronizer as a part of the m-ABP decoder so that no additional material is needed for the synchronization process.

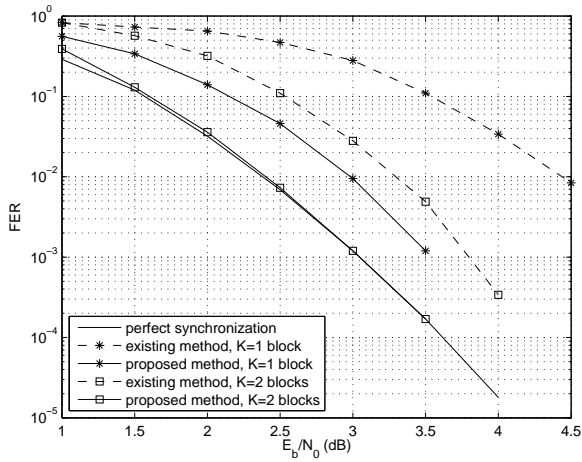


Fig. 6. Frame Error Rate of the m-ABP decoder applied to the $(16, 11)^2$ BCH product code.

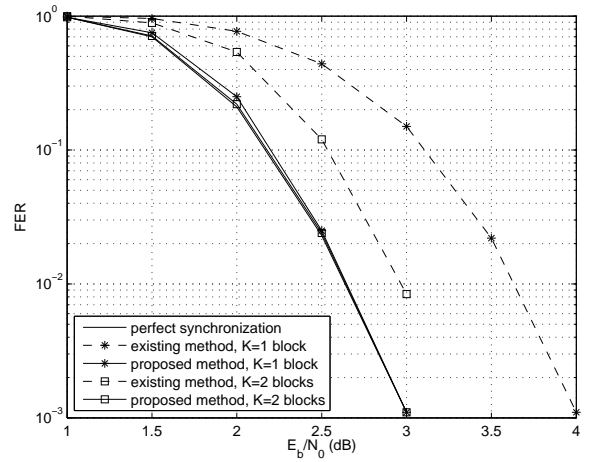


Fig. 7. Frame Error Rate of the m-ABP decoder applied to the $(32, 26)^2$ BCH product code.

REFERENCES

- [1] C. Berrou, A. Glavieux, and P. Thitimajshima, "Near shannon limit error-correcting coding and decoding: turbo-codes," *ICC*, pp. 1067–1070, May 1993.
- [2] D. J. C. MacKay and R. M. Neal, "Near shannon-limit performance of low-density parity-check codes," *Electronics Letters*, vol. 32, pp. 1645–1646, August 1996.
- [3] R. Pyndiah, A. Glavieux, A. Picart, and S. Jacq, "Near optimum decoding of product codes," in *Proc. of GLOBECOM*, November 1994, pp. 339–343.
- [4] R. H. Barker, "Group synchronization of binary digital systems," *Communication Theory*, pp. 273–287, 1953.
- [5] J. L. Massey, "Optimum frame synchronization," *IEEE Transactions on Communications*, vol. COM-20, pp. 115–119, April 1972.
- [6] P. Robertson, "A generalized frame synchronizer," in *Proc. of GLOBECOM*, December 1992, pp. 365–369.
- [7] T. M. Cassaro and C. N. Georghiadis, "Frame synchronization for coded systems over AWGN channels," *IEEE Transactions on Communications*, vol. 52, pp. 484–489, March 2004.
- [8] M. M. K. Howlader and B. D. Woerner, "Decoder-assisted frame synchronization for packet transmission," *IEEE J. Sel. Areas Comm.*, vol. 19, pp. 2231–2345, December 2001.
- [9] R. Imad, G. Sicot, and S. Houcke, "Blind frame synchronization for error correcting codes having a sparse parity check matrix," *accepted for publication in IEEE Transactions on Communications*, 2009.
- [10] S. Houcke and G. Sicot, "Blind frame synchronization for block code," in *Proc. of EUSIPCO, European Sig. Proc.*, September 2006.
- [11] R. Imad, S. Houcke, and C. Douillard, "Blind frame synchronization on gaussian channel," in *Proc. of EUSIPCO, European Sig. Proc.*, September 2007.
- [12] S. Houcke, G. Sicot, and M. Debbah, "Blind detection for block interleaved division multiple access," in *Proc. of GLOBECOM*, 2006.
- [13] J. Sun and M. C. Valenti, "Optimum frame synchronization for preamble-less packet transmission of turbo codes," *Asilomar Conference on Signals, Systems, and Computers*, vol. 1, pp. 1126–1130, November 2004.
- [14] J. Jiang and K. R. Narayanan, "Iterative soft-input-soft-output decoding of reed-solomon codes by adapting the parity check matrix," *IRE Trans. Inf. Theory*, pp. 3746–3756, August 2006.
- [15] C. Jego and W. J. Gross, "Turbo decoding of product codes based on the modified adaptive belief propagation algorithm," *IEEE International Symposium on Information Theory*, June 2007.
- [16] P. Elias, "Error-free coding," *IRE Trans. Inf. Theory*, pp. 29–37, January 1954.
- [17] J. Hagenauer, E. Offer, and L. Papke, "Iterative decoding of binary block and convolutional codes," *IEEE trans. on Information Theory*, vol. 42, no. 2, pp. 429–445, March 1996.