



Blind frame synchronization of Reed-salomon codes

Rodrigue Imad, Charly Pouillat, Sébastien Houcke, Benjamin Gadat

► To cite this version:

Rodrigue Imad, Charly Pouillat, Sébastien Houcke, Benjamin Gadat. Blind frame synchronization of Reed-salomon codes. SPAWC: 11th IEEE International Workshop on Signal Processing Advances in Wireless Communications, Jun 2010, Marrakech, Morocco. hal-00565835

HAL Id: hal-00565835

<https://hal.science/hal-00565835>

Submitted on 14 Feb 2011

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 REED-SOLOMON CODES: NON-BINARY VS. BINARY APPROACH

Rodrigue Imad¹, Charly Poulliat¹, Sebastien Houcke² and Benjamin Gadat³

¹ETIS, CNRS, ENSEA, Université de Cergy-Pontoise, 95000 Cergy Pontoise, France.

²Institut Telecom, Telecom Bretagne, UMR CNRS 3192 Lab-STICC, Brest, France.

³Thales Communications, EDS/SPM, Colombes, France.

ABSTRACT

We consider in this paper the problem of blind frame synchronization of systems using Reed-Solomon (RS) codes and other related families. We present first of all three techniques of blind frame synchronization based on the non-binary parity check matrix of RS codes. While the first two techniques involve the calculation of hard and soft values of the syndrome elements respectively, the third one perform an adaptation step of the parity check matrix before applying the soft criterion. Although RS codes are constructed from non-binary symbols, we show in this paper that it is also possible to synchronize them using the binary image expansion of their parity check matrix. Simulation results show that the synchronization algorithm based on the adaptation of the binary parity check matrix of RS codes has the best synchronization performance among all other techniques. Furthermore, the Frame Error Rate (FER) curves obtained after synchronization and decoding are very close to the perfect synchronization curves.

1. INTRODUCTION

Reed-Solomon (RS) codes became one of the most used codes because of their great correcting power and their utility in several applications. Many algorithms can be applied to decode RS codes. Although the Hard Decision Directed (HDD) decoder is used in most applications, its performance presents a significant degradation compared to soft decision decoding algorithms. The Adaptive Belief Propagation (ABP) algorithm was proposed in [1] by Jiang and Narayanan. This algorithm involves, at each iteration of the iterative procedure, the adaptation of the parity check matrix of the code to the reliability of the information bits. In [2], the authors proposed to achieve the adaptation procedure of the parity check matrix before applying the Belief Propagation (BP) algorithm, leading to the so-called modified-Adaptive Belief Propagation (m-ABP) algorithm.

Although decoding algorithms of RS codes have shown the ability of such codes to correct heavily corrupted sequences, a

great degradation in decoding performance is expected when the receiver does not know the beginning of a codeword. For this reason, frame synchronization should be applied before the decoder. Conventional frame synchronization techniques involve the transmission of a training sequence known by the receiver [3, 4, 5]. The major drawback of such techniques is in reducing the spectral efficiency of the transmission especially when codes of small sizes are used in the system. Other works propose to take advantage of the code structure and consider frame synchronization jointly with the decoding as in [6, 7]. In [8], the authors propose to encode synchronization bits as part of the data sequence.

In our work, we are interested in developing blind methods of frame synchronization wherein no additional sequence is added to the coded one. In [9], the authors proposed a new method of blind frame synchronization of block codes, based on the calculation of hard decisions of the syndrome elements. We refer to this technique by the Hard Syndrome (HS) based synchronization. The Soft Syndrome (SS) based synchronization algorithm was proposed in [10]. This algorithm outperforms the HS and another technique of blind frame synchronization presented in [11]. Although these synchronization methods are well adapted to codes having a sparse parity check matrix such as Low-Density Parity-Check (LDPC) codes, their performance degrades when the number of non zero elements in the parity check matrix of the code increases. Inspired by the m-ABP decoder, a modified version of the SS synchronization algorithm was proposed in [12]. Applying this algorithm to Bose-Chaudhuri-Hocquenghem (BCH) product codes yields very good results. In this paper, we propose and compare several frame synchronization techniques adapted to RS codes. These blind techniques are based on the non-binary and binary parity check matrices.

The remaining of this paper is organized as follows. In Section 2 we give the non-binary representation of RS codes and present three non-binary techniques of blind frame synchronization. We show in Section 3 that it is also possible to synchronize RS codes using the binary image expansion of their parity check matrix. Simulation results are presented in Section 4 and a conclusion to this work is drawn in Section 5.

This work has been supported by the French research agency under grant ANR-08-VERS-002 (CAIMAN project).

2. BLIND FRAME SYNCHRONIZATION OF RS CODES BASED ON A NON-BINARY APPROACH

We consider in this study that the system is using a Binary Phase Shift Keying (BPSK) mapping and that the propagation channel is corrupted by an additive white Gaussian noise of variance $\sigma_w^2 = N_0/2$. At the reception, the k^{th} received sample is written as:

$$r(k) = c(k - t_0) + w(k), \quad (1)$$

where $c(k)$ is the k^{th} symbol to be transmitted, $w(k)$ is the white Gaussian noise and t_0 is an integer representing the shift of the transmitted symbols. The goal of frame synchronization is to estimate t_0 .

Synchronization techniques that we present in this paper to synchronize RS codes are inspired by the ones proposed in [10, 12]. These techniques are based on a Maximum A Posteriori (MAP) approach in the sense of maximizing the probability that a position t corresponds to the correct synchronization moment, given the received samples. In other words, it maximizes the *a posteriori* probability $Pr[t|\mathbf{r}]$, where t is a possible value of the correct synchronization position and \mathbf{r} is the received sequence of samples. Maximizing this probability is done using a sliding synchronization window on the received sequence. The size of the synchronization window can be a multiple of the size of a codeword.

In order to confirm that a position t is the correct synchronization position, we should verify that the block transmitted at this position corresponds to a valid codeword and that all other blocks in the remaining tested positions are not. The easiest way to check whether a block corresponds to a valid codeword or not is by calculating its syndrome which is obtained according to the parity check matrix of the code.

2.1. Non-binary representation of Reed-Solomon codes

A Reed-Solomon $RS(N, K)$ code is a non-binary cyclic code of length N symbols constructed from K data symbols. Each symbol is defined in the Galois Field $GF(2^m)$ and can be represented by binary m -tuples, where m is an integer greater than 2. The generator polynomial of RS codes is written as:

$$g(x) = (x + \alpha)(x + \alpha^2) \dots (x + \alpha^{2t}), \quad (2)$$

where α is a primitive element in $GF(2^m)$ and $t = (N - K)/2$ is the symbol-error correcting capacity of the code.

The non-binary parity check matrix of the $RS(N, K)$ code has a size $((N - K) \times N)$ symbols and can be written as [1]:

$$H = \begin{pmatrix} 1 & \alpha & \alpha^2 & \dots & \alpha^{(N-1)} \\ 1 & \alpha^2 & \alpha^4 & \dots & \alpha^{2(N-1)} \\ \vdots & & \ddots & & \vdots \\ 1 & \alpha^{N-K} & \alpha^{(N-K)2} & \vdots & \alpha^{(N-K)(N-1)} \end{pmatrix} \quad (3)$$

We propose next three techniques of blind frame synchronization designed for non-binary codes, in particular for RS codes.

2.2. Non Binary Hard Syndrome based synchronization

We denote by h_{ij} the element at row i and column j of the non-binary parity check matrix of the RS code. From hard decisions on received samples, we define the k^{th} syndrome element as:

$$S(k) = \sum_{l=1}^N h_{kl} \bar{y}(l), \quad (4)$$

where $\bar{y}(l)$ are hard decisions on symbols in $GF(2^m)$. In a noise free channel, if a received block is a valid codeword then the $(N - K)$ syndrome elements are all equal to zero. When the synchronization position t_0 is unknown, a block could contain arbitrary samples that do not form a codeword and therefore, some syndrome elements are not equal to zero. Inspired by this idea, the Non Binary Hard Syndrome (NBHS) based synchronization algorithm that we propose in this section estimates the synchronization position by:

$$\hat{t}_{NBHS} = \underset{t=0, \dots, mN-1}{\operatorname{argmin}} \left\{ \sum_{k=1}^{M(N-K)} s_t(k) \right\}, \quad (5)$$

where

$$s_t(k) = \begin{cases} 0 & \text{if } S_t(k) = 0 \\ 1 & \text{if } S_t(k) \neq 0 \end{cases} \quad (6)$$

and $S_t(k)$ is the non-binary syndrome calculated at position t of the sliding synchronization window. The variable M represents the number of blocks (of size mN samples each) contained in the sliding window. Synchronization performance is improved when M increases.

2.3. Non Binary Soft Syndrome based synchronization

Calculating hard values of syndrome elements involves a loss of information. Inspired by the algorithm of [10], we present in this section the Non Binary Soft Syndrome (NBSS) based synchronization algorithm. Indeed, we can show that the synchronization criterion of [10] in its original form (before applying approximations proposed by the authors) is valid for any field order. Hence, the optimal estimate of the frame synchronization position can be derived as:

$$\hat{t}_{NBSS} = \underset{t=0, \dots, mN-1}{\operatorname{argmax}} \left\{ \sum_{k=1}^{M(N-K)} \log (Pr[S_t(k) = 0]) \right\}. \quad (7)$$

In the non-binary case, the probability that a syndrome element is equal to zero can be calculated after half iteration of the non-binary BP decoder, which can be efficiently implemented in the Fourier domain. Details on the non-binary BP algorithm are omitted due to lack of space but further information on Fast Fourier Transform (FFT) based BP and related low complexity algorithms can be found in [13].

2.4. Adaptive Non Binary Soft Syndrome based synchronization

Synchronization algorithms proposed in Sections 2.2 and 2.3 are well adapted for codes having a sparse non-binary parity check matrix. However, (3) shows that the parity check matrix of RS codes does not contain any element that is equal to zero. Inspired by [1], we propose in this section to perform, at each position of the sliding window, an adaptation of the non-binary parity check matrix of the code, to the reliability of the non-binary symbols.

Many methods can be applied to compute the reliabilities of symbols defined over $GF(2^m)$ [1]. In this paper, we define the reliability of a non-binary symbol $y(k)$ as:

$$Rel(y(k)) = \min_{i=1,\dots,m} Rel(r_i(k)) = \min_{i=1,\dots,m} |r_i(k)|, \quad (8)$$

where $r_i(k)$ is the i^{th} sample constituting the non-binary symbol $y(k)$.

Once reliabilities are computed and sorted by ascending order, the first $(N - K)$ columns of the non-binary parity check matrix, corresponding to the $(N - K)$ least reliable symbols, are reduced to obtain a diagonal squared matrix. This can be done by applying a Gaussian elimination method on $GF(2^m)$ or a more computationally efficient method as described in [1]. 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 samples.

Once adaptation operations are done, the Adaptive Non Binary Soft Syndrome (ANBSS) based synchronization performs the same steps as the ones done by the NBSS algorithm and estimates the frame synchronization position using (7).

3. BLIND FRAME SYNCHRONIZATION OF RS CODES BASED ON A BINARY APPROACH

Although RS codes are non-binary codes, we propose in this section to synchronize them using the binary image expansion of their parity check matrix. For the non-binary matrix H of (3), we can find an equivalent binary parity check matrix H_b of size $(m(N - K) \times mN)$. This matrix is obtained by replacing α in matrix H by the transpose of the binary companion matrix A of size $m \times m$. Indeed, if

$$p(x) = a_0 + a_1x + \dots + x^m \quad (9)$$

is a primitive polynomial having its coefficients in $GF(2)$, then the companion matrix of $p(x)$ is the $m \times m$ matrix defined by:

$$A = \begin{pmatrix} 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & \dots & 0 \\ 0 & 0 & 0 & \dots & 1 \\ a_0 & a_1 & a_2 & \dots & a_{m-1} \end{pmatrix} \quad (10)$$

Although the binary parity check matrix of RS codes contains some zeros, the number of ones in it is still large. Therefore, the synchronization technique of [10] is not adapted to RS codes. We propose in this section to apply the synchronization technique proposed in [12] for synchronizing BCH codes, to synchronize RS codes using the binary image expansion of their parity check matrix.

The main idea is to slide the synchronization window on mN consecutive positions knowing that, one of these positions corresponds to the correct frame synchronization. At each position of this window and for each block of size mN contained in the sliding window, we rearrange its samples according to their reliabilities (i.e their absolute values). Then, using a Gaussian elimination on $GF(2)$, we rearrange the binary parity check matrix of the RS code. The first $m(N - K)$ columns of the binary parity check matrix, corresponding to the $m(N - K)$ least reliable samples, are thus reduced to obtain an identity squared matrix.

Once the rearrangement procedure is achieved, the Adaptive Binary Soft Syndrome (ABSS) based synchronization algorithm estimates the frame synchronization position by:

$$\hat{t}_{ABSS} = \underset{t=0,\dots,mN-1}{\operatorname{argmin}} \{ \psi(t) \}, \quad (11)$$

where

$$\psi(t) = \sum_{k=1}^{Mm(N-K)} \log \left(1 + \exp \left(L(S_t(k)) \right) \right). \quad (12)$$

$L(S_t(k)) = \log \left(\frac{\Pr[S_t(k)=1]}{\Pr[S_t(k)=0]} \right)$ is the Log-Likelihood Ratio (LLR) of the k^{th} syndrome element and it can be easily computed after half iteration of the binary BP decoder [14, 10]. Note that (11) is strictly equivalent to (7) if the latter was applied in the binary domain.

4. SIMULATION RESULTS

We present in this section simulation results obtained after applying the proposed synchronization techniques to RS codes. Monte Carlo simulations were done where, for each configuration, the noise, information bits and the delay of the channel were randomly chosen. RS codes being cyclic codes and in order to keep good synchronization performance, a pseudo-random interleaver should be used to permute the transmitted symbols [12]. Fig. 1 shows curves of the probability of false synchronization versus the signal-to-noise ratio (E_b/N_0) once synchronization techniques were applied to $RS(7, 5)$ code. A sliding window of size $M = 5$ blocks was considered. From Fig. 1, it is shown that performance of the NBHS (NBSS respectively) is better than the HS (SS respectively), especially for high E_b/N_0 . On the other hand, it is clearly seen that the ABSS algorithm is the most effective among all other synchronization techniques.

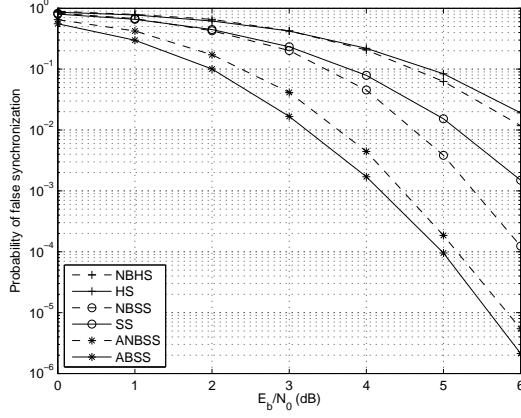


Fig. 1. Probability of false synchronization of different synchronization techniques when applied to $RS(7, 5)$ code.

The ANBSS and ABSS were also applied to the $RS(31, 25)$ code. Simulation results are shown in Fig. 2. We also plotted in this figure curves obtained by the algorithm proposed by Massey in [4], which locates a synchronization word periodically embedded in the data. Even for a synchronization word having a size that corresponds to 7.1% of the total size of the $RS(31, 25)$ code, the ABSS and ANBSS algorithms converge quicker than [4]. Due to the small size of RS codes ($RS(31, 25)$ has a size of 155 bits), it is clear that a synchronization window of size $M = 1$ block does not afford good synchronization performance. Increasing the size of the synchronization window yields better results. Once again, the ABSS outperforms the ANBSS. A gap of around 1 dB exists between the two curves. However, notice that the non-binary adaptation performed by the ANBSS algorithm can be parallelized and thus, it is less complex to implement than the binary adaptation procedure [1].

In the remaining of this paper and for complexity reasons, the criterion of (12) in the SS and the ABSS algorithms is replaced by the approximation given in [10]. In order to verify the adequacy of the ABSS to synchronize RS codes, we plotted in Fig. 3 the Frame Error Rate (FER) curves obtained after synchronization and decoding of the $RS(31, 25)$. The decoder used is the ABP proposed in [1]. For a synchronization window of size $M = 5$ blocks, it is clearly seen that the FER curve obtained after synchronization and decoding is very close to the perfect synchronization case. For small E_b/N_0 , a gap of less than 0.4 dB exists between the two curves. These curves become almost one for an E_b/N_0 greater than 4 dB.

Let us now apply the ABSS synchronization technique to RS product codes. A product code $C(N_1 N_2, K_1 K_2)$ is a bidimensional code constructed from two elementary sys-

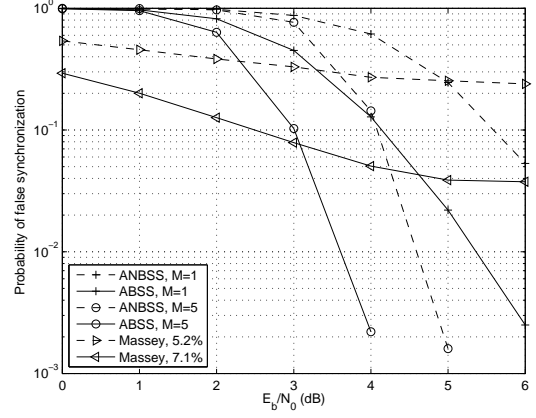


Fig. 2. Probability of false synchronization of different synchronization techniques when applied to $RS(31, 25)$ code.

tematic block codes $C_1(N_1, K_1)$ and $C_2(N_2, K_2)$. Hence, rows and columns of code C are codewords of C_1 and C_2 , respectively. We propose now to take advantage of the bidimensional structure of RS product codes and apply the ABSS on their elementary binary parity check matrices. Therefore, the synchronization criterion calculated in (12) is computed now twice: once on the rows of the RS product code, we get $\psi_r(t)$, and the other time on its columns, we get $\psi_c(t)$. The frame synchronization position is now estimated by:

$$\hat{t} = \underset{t=0, \dots, mN-1}{\operatorname{argmin}} \{ \psi_r(t) + \psi_c(t) \}. \quad (13)$$

We consider now the $RS(7, 5)^2$ product code. This notation means that this product code is constructed by two elementary $RS(7, 5)$ codes. In order to verify the effectiveness of the proposed technique on synchronizing RS product codes, we plotted in Fig. 4 the FER curves obtained after synchronization and decoding. The decoder used to decode these codes is the m-ABP decoder presented in [2], which has a performance that is equivalent (and sometimes better) than the Chase-Pyndiah decoder. In Fig. 4 we can see that applying the SS algorithm to synchronize the $RS(7, 5)^2$ product code does not yield good results. For a synchronization window of size $M = 1$ block and at a FER equal to 10^{-2} , a gap of around 1 dB is observed between the SS and the perfect synchronization case. This gap in FER disappears when we apply the ABSS algorithm. Hence, the ABSS is also well adapted to RS product codes.

5. CONCLUSION

We have presented in this paper original techniques for blind frame synchronization of RS codes based on their binary and non-binary parity check matrices. Simulation results have

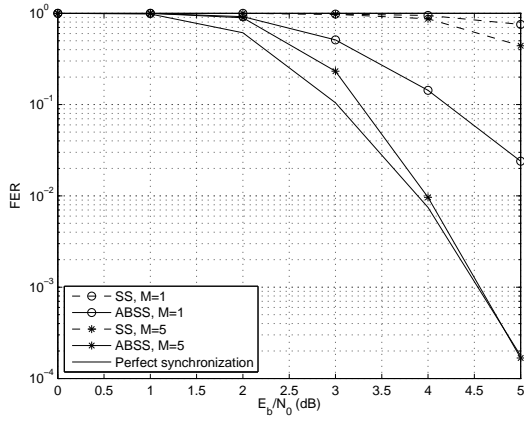


Fig. 3. Frame Error Rate (FER) curves after synchronization and decoding of the $RS(31, 25)$.

shown that the adaptive binary approach (ABSS) outperforms the non-binary one (ANBSS). Furthermore, the FER curves obtained after synchronization and decoding have proven that the ABSS technique is well adapted to RS codes. This technique was also applied to RS product codes and the results were very satisfactory. Despite the fact that the ABSS algorithm outperforms the ANBSS, the adaptation procedure used by the latter can be parallelized, which is very beneficial for hardware implementation. A perspective of this work may be to use the non-binary matrix for the adaptation (first part of the ANBSS) then compute its binary image expansion followed by half iteration of the binary BP algorithm (second part of the ABSS). Doing this may lead to reduce the complexity and calculation time of the proposed synchronization algorithm.

6. REFERENCES

- [1] J. Jiang and K. R. Narayanan, "Iterative soft-input-soft-output decoding of reed-solomon codes by adapting the parity check matrix," *IRE Transactions on Information Theory*, vol. 52, no. 8, pp. 3746–3756, August 2006.
- [2] C. Jegou and W. J. Gross, "Turbo decoding of product codes based on the modified adaptive belief propagation algorithm," in *Proc. of ISIT*, June 2007, pp. 641–644.
- [3] R. H. Barker, "Group synchronization of binary digital systems," *Communication Theory*, pp. 273–287, 1953.
- [4] J. L. Massey, "Optimum frame synchronization," *IEEE Transactions on Communications*, vol. 20, no. 2, pp. 115–119, April 1972.
- [5] G. L. Lui and H. H. Tan, "Frame synchronization for gaussian channels," *IEEE Transactions on Communications*, vol. 35, no. 8, pp. 818–829, August 1987.

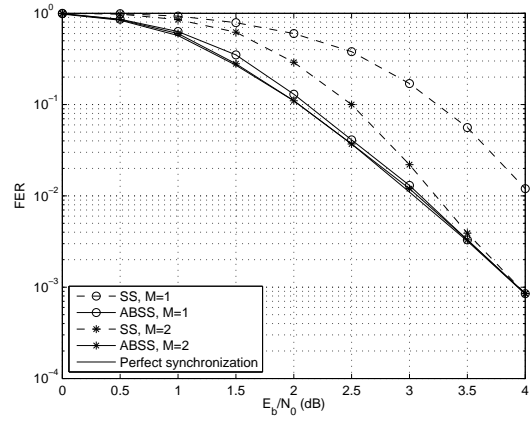


Fig. 4. Frame Error Rate (FER) curves after synchronization and decoding of the $RS(7, 5)^2$ product code.

- [6] P. Robertson, "A generalized frame synchronizer," in *Proc. of GLOBECOM*, December 1992, pp. 365–369.
- [7] T. M. Cassaro and C. N. Georgiades, "Frame synchronization for coded systems over AWGN channels," *IEEE Transactions on Communications*, vol. 52, no. 3, 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, no. 12, pp. 2231–2345, December 2001.
- [9] S. Houcke and G. Sicot, "Blind frame synchronization for block code," in *Proc. of EUSIPCO, European Sig. Proc.*, September 2006.
- [10] R. Imad, G. Sicot, and S. Houcke, "Blind frame synchronization for error correcting codes having a sparse parity check matrix," *IEEE Transactions on Communications*, vol. 57, no. 6, pp. 1574–1577, June 2009.
- [11] 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.
- [12] R. Imad, S. Houcke, and C. Jegou, "Blind frame synchronization of product codes based on the adaptation of the parity check matrix," in *Proc. of ICC*, June 2009.
- [13] D. Declercq and M. Fossorier, "Decoding algorithms for nonbinary LDPC codes over $GF(q)$," *IEEE Transactions on Communications*, vol. 55, no. 4, pp. 633–643, 2007.
- [14] F. Kschischang, B. Frey, and H. A. Loeliger, "Factor graphs and the sum product algorithm," *IEEE Transactions on Information theory*, vol. 47, no. 2, pp. 498–519, February 2001.