Volume 51, Number 4, 2010 **ACTA TECHNICA NAPOCENSIS**

Electronics and Telecommunications

NEW PSEUDO-RANDOM INTERLEAVER WITH INCREASED PARAMETERS

Georgian Alexandru LAZAR, Nicoleta Roxana BUZATU, Elena COJOCARIU, Lucian TRIFINA, Razvan VIERU

Technical University of Iasi, Romania

Bd. Carol I, no. 11, phone: +40232213737, fax: +402132217720,

Email: lazara@etti.tuiasi.ro

Abstract: This paper presents a new algorithm of obtaining a pseudo-random code-matched interleaver leading both to very good interleaver parameters and performance. The difference between the proposed permutation and other code matching techniques is that not only the distance spectrum is improved but also the parameters of the interleaver. The design procedure is described in depth, and the benchmarking is done against the High Spread-Random and the deterministic Long Term Evolution (LTE) standard interleavers.

Keywords: *Interleaved coding, modulation coding, code matched, turbo codes, error correcting codes.*

I. INTRODUCTION

The advent of turbo codes represents one of the most important breakthroughs in coding theory over the recent years, closing the gap on the Shannon limit in terms of error correcting performance. The low error rate of the turbo coding scheme is achieved by combining two convolutional encoders using an interleaver, which is basically equivalent to multiplying the input sequence with a permutation matrix.

Turbo codes have a sparse distance spectrum, which provides leads to very good performances at a low signal to noise ratio (SNR). This is mainly caused by the spreading randomness effects introduced by the interleaver, which translates into a low multiplicity of low-weight codewords. The main drawback of turbo codes resides in their low minimum distance, which produces an error-floor limitation [1].

The solution to this problem is to either increase the size of the interleaver, or to use a code –matched interleaver design. There are several deterministic code–matching techniques, which can provide a high minimum distance [2-4]. Because these designs are highly structured, the randomness of the generated permutations is very poor. A different pseudo-random code-matching technique is presented in [5]. However, in this case the obtained spreading factors are average.

II. TYPICAL INTERLEAVER PARAMETERS

Supposing that i and j are two indexes in the permutation $\pi(x)$ of length K, then the following parameters are relevant when designing a good interleaver:

a. The spreading factor S

The spreading factor S is the maximum number for which:

$$|i-j|< S \Rightarrow |\pi(i)-\pi(j)| \geq S \qquad (1)$$

The larger the S parameter is, the better the burst errors can be corrected [6].

b. The dispersion Γ and normalized dispersion γ

The dispersion Γ is the number of distinct pairs:

$$\{\Delta_x, \Delta_y\} = \{j-i, \pi(j)-\pi(i)\}; i < j \qquad (2)$$

The larger the Γ parameter is the better the statistical independence between the two inputs of the encoders is [6]. This parameter is a reflection of the degree of randomness of the turbo encoder.

The normalized dispersion γ can be obtained by dividing the dispersion to a factor of K*(K-1).

c. The S_{new} and the D spreading factors

The new spreading factor S_{new} is defined as [7]:

$$S_{new}(i,j) = \min_{i,j}(|\pi(i)-\pi(j)|+|i-j|); i \neq j \qquad (3)$$

The D-spreading factor is defined in [8]. It is similar to the S_{new} spread, but instead of the Manhattan metric, the Lee metric is used.

$$D = \min_{i,j}(|\pi(i)-\pi(j)|_K + |i-j|_K); i \neq j \qquad (4)$$

Where

$$|x-y|_K = \min_{x \neq y}(mod(x-y,K), mod(y-x,K)); \qquad (5)$$

Manuscript received June 26, 2010; revised December 12, 2010

d. The free distance d_{free}

The free distance d_{free} is the minimum Hammning distance between different code words. For an Additive White Gaussian Noise (AWGN) channel at high and moderate SNRs, the bit error rate (BER) and the frame error rate (FER) can be expressed through the following bounds:

$$FER = 0.5 * N_{free} * erfc\left(\sqrt{d_{free} * R_C * \frac{E_b}{N_0}}\right) \quad (6)$$

$$BER = 0.5 * \frac{\omega_{free}}{K} * erfc\left(\sqrt{d_{free} * R_C * \frac{E_b}{N_0}}\right) \quad (7)$$

Where K is the length of the interleaver; R_C is the coding rate; erfc is the error function complement; N_0 is the one sided noise power spectral density; E_b is the bit energy; d_{free} is the free distance; N_{free} is the multiplicity of the code words with the Hamming weight equal to d_{free} and ω_{free} is the sum of the N_{free} information words that produce code words with the Hamming weight equal to d_{free}. The interleaver has to increase d_{free} and decrease N_{free} and ω_{free} in order to lower the error floor at moderate to high SNRs.

In case of a Rayleigh Multiplicative Fading (RMF) channel, the bounds expressed in equations (6) and (7) become:

$$FER = 0.5 * N_{free} * \left(\frac{1}{1 + R_C * \frac{E_b}{N_0}}\right)^{d_{free}} \quad (8)$$

$$BER = 0.5 * \frac{\omega_{free}}{K} * \left(\frac{1}{1 + R_C * \frac{E_b}{N_0}}\right)^{d_{free}} \quad (9)$$

The coding rate depends on the type of trellis termination of the turbo encoder. In the considered simulation scenarios, the LTE- standard termination was adopted, with both component trellises terminated using a post- interleaver termination method. Additionally, the tail bits of the second encoder are not transmitted. In this situation, for an interleaver of size K, the coding rate can be expressed as:

$$R_C = \frac{K}{3 * K + 4 * m} \quad (10)$$

Where m represents the memory of the constituent encoders. The LTE configuration has constituent encoders of memory 3, so the coding rate has the following value:

$$R_C = \frac{K}{3 * K + 12} \quad (11)$$

III. THE HSR AND LTE INTERLEAVERS

The High-Spread Random (HSR) interleaver is randomly generated and its elements respect an user imposed S_{new} spreading factor. The generation time is reasonable and the spreading factor obtained is close to the maximum value of $\sqrt{2K}$ .The overall performance of this interleaver is greater than the one of a simple S-Random interleaver [7].

The LTE (Long Term Evolution) standard is a 4G communication standard that is due to replace the current 3G UMTS (Universal Mobile Telecommunication Standard). The LTE interleaver is derived from the QPP (Quadratic Permutation Polynomial) interleaver and is among the best one known for turbo codes [9].

IV. THE PROPOSED INTERLEAVER DESIGN

The proposed interleaver design contains three distinct stages. In the first stage, a start-up interleaver structure is obtained. In order to reduce the generation time and enhance the performance, the start-up permutation has to have a good S_{new} spreading factor. The second stage, the start-up interleaver is randomly enhanced, so that its parameters (S, S_{new}, D and γ) are increased. This is done through a swap technique. The final stage of the interleaver design has the role of code-matching the interleaver to the particular turbo code used. During this stage, the last spectral line of the code is improved, while the parameters obtained in stage two (except the dispersion γ) are preserved.

a. The first stage

The first stage generates a starting interleaver, that will be enhanced further on. Because of the flexibility of the considered design method, any kind of interleaver can be chosen as the starting interleaver. However, provided a proper type of interleaver is chosen during this stage, not only the overall generation time, but also the performance of the final interleaver can be increased. The most suitable interleavers that can be involved in this stage, have to provide a balance between both the spreading parameters (S, S_{new} and D) and the dispersion (γ). One can use either a S-Random, HS-Random or swap interleaver [10].

b. The second stage

The second stage refers to the improvement of all the start-up interleaver spreading parameters. The algorithm related to this stage is a pseudo-random one, and the parameters that can be increased depend on the type of the interleaver chosen in the first stage. In case of a S-Random interleaver, both the S_{new} and D spreading factors can be increased, whereas in case of a HS-Random interleaver, the S and D spreading factors can be increased. The modified swap interleaver can be enhanced in terms of the D spreading factor. The drawback of this kind of starting interleaver is given by the high values that are obtained for all the spreading factors (D, S and S_{new}). This means that step three (which is the most important one, because it ensures the code-matching feature of the interleaver) is offered less swaping opportunities. In order to solve this problem and increase the final performance of the interleaver by improving the distance spectrum, one of the three spreading parameters has to be sacrificed. The best option is to have a lower S-spreading factor, by selecting the High-Spread Random interleaver as the starting interleaver. This kind of interleaver has a moderate S-spreading factor, a very high S_{new} spreading factor, but a rather low D-spreading factor. Thus, for this type of interleaver, the only spreading factor that has to be increased is the D-spreading factor.

The algorithm used for increasing one of the spreading factors (for example the D-spreading factor), while preserving the other two spreading factors (for example S and S_{new}) is described below:

1) Compute the S,S_{new} and D spreading factors
2) Initialize the number of total tries per spreading value to 25*the interleaver's length and the maximum number of accepted elements that violate the spreading nmax=10
3) For d_current=D to the desired spreading value do:
4) Compute the number of terms from the interleaver that violate the current spreading value and store these terms in a vector v
6) Compute and the normalized dispersion γ
7) For i=1 to the desired number of tries do:
8) Generate two distinct random positions, one from the whole interleaver and one from the vector v of the interleaver's elements that violate the spread and swap them
9) The swap is kept provided that the following conditions are met:
a) The initial S and S_{new} spreading parameters are preserved or increased
b) The number of terms from the interleaver that violate the current spreading value d_current has diminished or remained unchanged, but the normalized dispersion γ has increased
10) Exit the loop if the number of elements that violate d_current is zero
11) If after performing the number of tries, the number of elements that violate d_current is not zero, but is smaller then nmax, than the following thorough swap is performed: 12) Take each element from the interleaver and perform a swap with each element from the vector v that violates the d_current spreading value
13) The swap is kept using the same conditions as the one from step 9
14) If after performing a complete scan of the interleaver the interleaver, there are still elements that violate d_current, than the spreading value cannot be larger than d_current-1, and the algorithm ends, and all the swaps made in search of a d_current spread are discarded.

c. The third stage

In the third stage is a code-matching stage, which tries to improve the last three spectral lines of the interleaver and increase the dispersion γ. In order to generate a high performance code-matched permutation, a method of computing the distance spectrum of the specific code is necessary. Furthermore, because in the simulations, the post-interleaver trellis termination is considered, the distance spectrum calculation algorithm has to take this aspect into account as well. The most reliable distance spectrum computation method is the true distance measurement method, which is able to reliably compute the first three terms of the distance spectrum. The disadvantage of this approach is that the complexity increases severely with the free distance (which in its turn is dependent on the interleaver's length) [11].

The design algorithm for the code matched interleaver can be synthesized as follows:

1) Initialize the number of tries with 150*the interleaver's length
2) Calculate the S, D and S_{new}-spread of the interleaver and the first term of the distance spectrum d_{free}, n_{free} and w_{free}, taking into consideration the post-interleaver flushing termination. Furthermore, the normalized dispersion γ is computed. A cost function I is defined, where $I=(S+S_{new}+D)^*\gamma$
3) For the desired number of iterations perform the following operations:
4) Generate two random interleaver positions and perform a swap
5) The swap is kept only if a series of conditions in the following order are met:
a). If the new interleaver doesn't have a spreading factor at least equal to the initial S, D and S_{new} spread values, the swap is discarded and the algorithm returns to step 4, otherwise jump to 5.b
b) The first term of the distance spectrum is computed. If there is an improvement in the sequence d-n-w (if FER optimization is desired) or in the sequence d-w-n (if BER optimization is desired) the swap is kept and the algorithm returns to step 4. In case there is no change in the distance spectrum, the algorithm jumps to step 5.c
c) The cost function $I=(S+S_{new}+D)^*\ \gamma$ are computed. The swap is kept if the cost function is improved, otherwise the algorithm jumps to step 4.

The algorithm has several advantages over other code matched interleavers. First of all, the design offers flexibility, not only in terms of the number of iterations that are user definable in steps IV.b and IV.c, but also from the point of view of FER or BER optimization. The values chosen for these variables are chosen empirically in order to provide a reasonable compromise between performance and time consumption for the studied frame lengths (under 300). In case of larger interleaver sizes, these values should be lowered.

Secondly, a real distance spectrum calculation algorithm is used, instead of a distance spectrum estimation using various error patterns. Thirdly, unlike other deterministic code-matched schemes [12], not only the degree of randomness is improved, but also the spreading factors are increased, during the second design stage. Furthermore, the third design step is

not deterministic, but rather pseudo-random, which drastically reduces the overall generation time.
The selected cost function I from step IV.c, was selected so that there is a balance between all the spreading parameters and the normalized dispersion factor.

V. SIMULATIONS AND RESULTS

The simulations were run for two code matched interleavers, deriving from the high-spread random interleavers of lengths equal to 152 and 256, in case of both Additive White Gaussian Noise (AWGN) and Rayleigh Multiplicative Fading (RMF). The number of decoder iterations was set to 12, the modulation used was Binary Phase Shift Keying (BPSK) with Bit Interleaved Coded Modulation (BICM). The constituent convolutional encoders chosen were identical and equal to (15/13) in octal with post interleaver trellis termination and the decoding algorithm was log-MAP. The the BER and FER curves are shown in figures 1, 2, 3 and 4 for L1=152 and in figures 5, 6, 7 and 8 for L2=256. Additionally, tables 1, 2, 3, 4, 5 and 6 illustrate the first terms of the distance spectrum computed for post-interleaver trellis termination and the most important parameters as computed for the high-spread random, matched high-spread random and LTE interleavers for the simulated lengths. The results of the simulations yield to a clear improvement of the performance, due to the better distance spectrum and the proper choice of the cost function I, which maximizes both the S- spread and the dispersion γ.

VI. CONCLUSIONS AND FUTURE WORK

This paper presents a new three-stage code matched interleaver design technique. It's performances are evaluated against both the HS random and the deterministic LTE standard interleaver. The results of the simulations show a clear improvement in each of the considered scenario. Furthermore, through the design that involves not only code – matching but also a careful increase of some of the interleaver's key parameters, this structure can be used in some practical settings where the turbo code is fixed and the frame length is short. Future work should address to the study of this code matched interleaver for longer frame sizes. In this case, a faster distance spectrum computation method should be used, such as that described in [13].

Table 1. Parameters of the LTE interleaver for L1=152

Parameter	S	S_{new}	D	γ	d_{free}	n_{free}	ω_{free}
Value	6	12	12	0.0381	15	1	1

Table 2. Parameters of the HSR interleaver for L1=152

Parameter	S	S_{new}	D	γ	d_{free}	n_{free}	ω_{free}
Value	7	14	7	0.7565	18	4	8

Table 3. Parameters of the matched-HSR interleaver for L1=152

Parameter	S	S_{new}	D	γ	d_{free}	n_{free}	ω_{free}
Value	7	14	13	0.8247	18	1	2

Table 4. Parameters of the LTE interleaver for L2=256

Parameter	S	S_{new}	D	γ	d_{free}	n_{free}	ω_{free}
Value	8	16	16	0.0413	16	1	2

Table 5. Parameters of the HSR interleaver for L2=256

Parameter	S	S_{new}	D	γ	d_{free}	n_{free}	ω_{free}
Value	9	18	9	0.7891	17	1	1

Table 6. Parameters of the matched- HSR interleaver for L2=256

Parameter	S	S_{new}	D	γ	d_{free}	n_{free}	ω_{free}
Value	9	18	18	0.8126	20	1	2

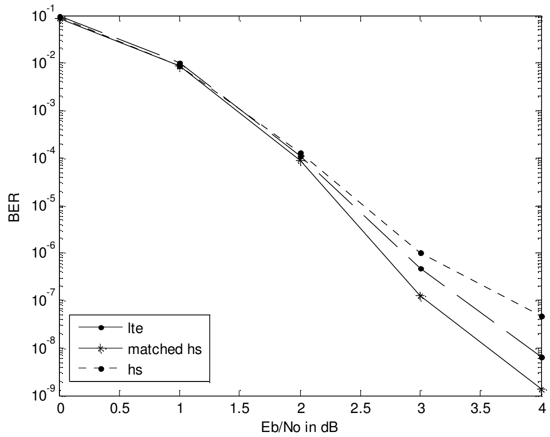

Figure 1. BER for AWGN channel and length L1=152

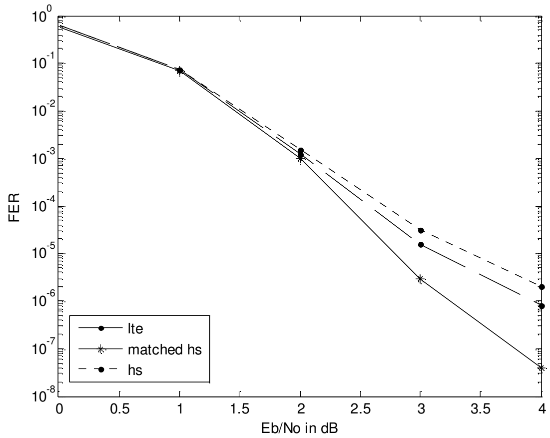

Figure 2. FER for AWGN channel and length L1=152

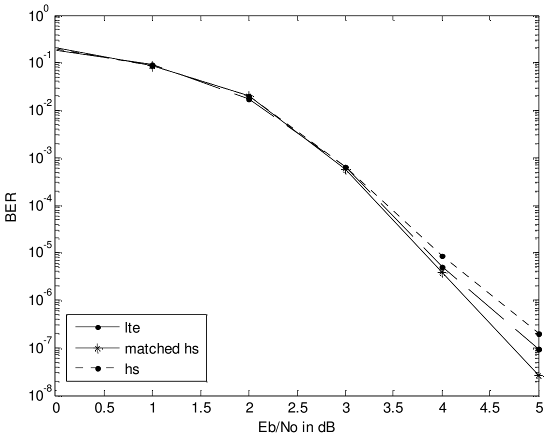

Figure 3. BER for RMF channel and length L1=152

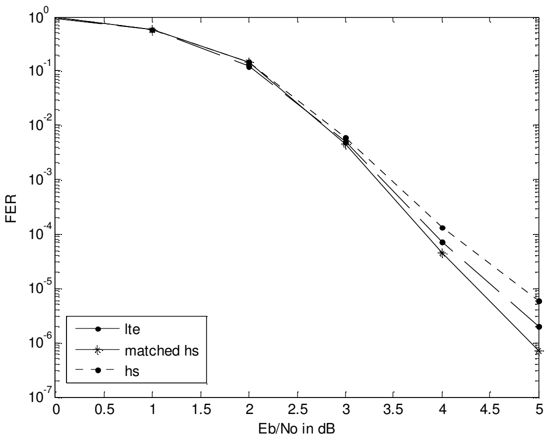

Figure 4. FER for RMF channel and length L1=152

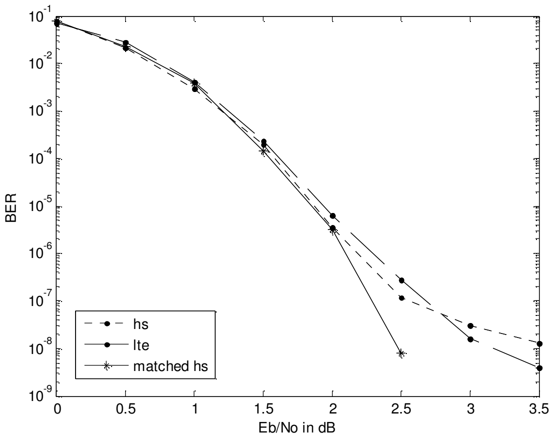

Figure 5. BER for AWGN channel and length L2=256

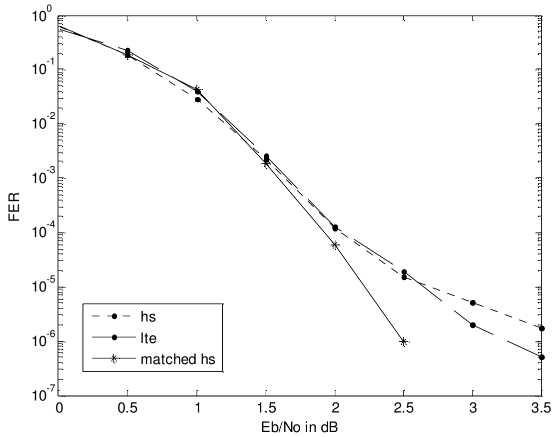

Figure 6. FER for AWGN channel and length L2=256

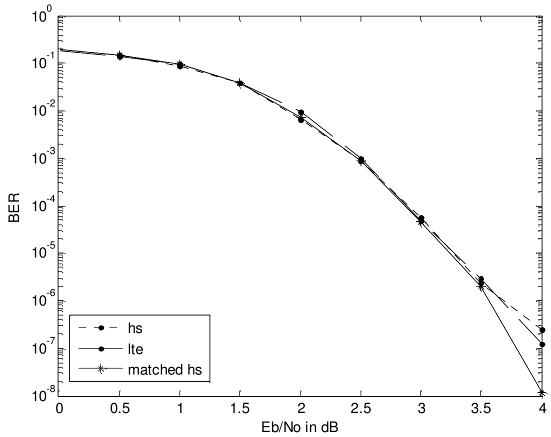

Figure 7. BER for RMF channel and length L2=256

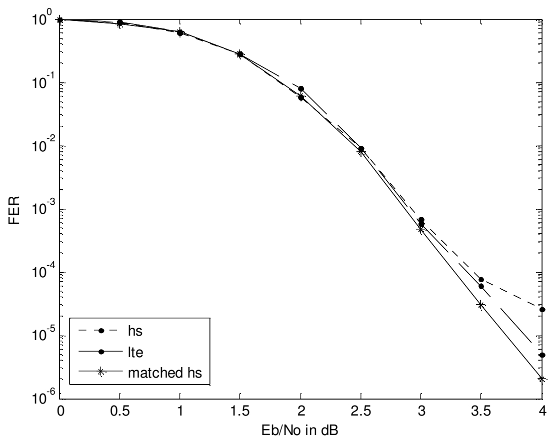

Figure 8. FER for RMF channel and length L2=256

REFERENCES

[1] Perez, L., Seghers J., Costello D., "A Distance Spectrum Interpretation of Turbo Codes" *IEEE Transactions on Information Theory, special issue on coding and complexity* vol.42, no.6, pp 1698-1709, 1996.

[2] Crozier, S. and Guinand, P., "High-Performance Low-Memory Interleaver Banks for Turbo-Codes", *IEEE Vehicular Technology Conference*, vol. 4, 2934-2938, October 7-11, 2001.

[3] Berrou C., Saouter Y., Douillard C., Kerouedan S., Jezequel M., "Designing good Permutations for Turbo Codes : Towards a single model" *Proceedings of IEEE International Conference on Communications*, vol. 1, pp. 341-345, 2004.

[4] Sun J., Takeshita O., "Interleavers for Turbo Codes Using Permutation Polynomial Over Integers Rings" *IEEE Transactions on Information Theory*, vol.51, no.1, pp. 101-119, 2005.

[5] Feng W., Yuan J., Vucetic B. "A code-matched Interleaver Design for Turbo Codes" *IEEE Transaction on Communications*, vol. 50, no. 6, pp. 926-937, 2002.

[6] Heegard C., Wicker S., "*Turbo Coding*" . Kluwer Academic Publisher, 1999.

[7] Crozier S., "New high-spread high-distance interleavers for turbo codes" *Proceedings of Symposium of Communications* Kingston, pp. 3-7, 2000.

[8] Takeshita O., "Permutation polynomial interleavers: an algebraic-geometric perspective" *IEEE Transactions on Information Theory*, vol. 32, no. 6, 2116-2132, 2007.

[9] Sesia S., Toufik I., Baker M., "*LTE-The UMTS long term evolution- from theory to practice*". John Wiley & Sons, 2009.

[10] Lee B. G., Bae S. J., Kang S. G., Joo E. K., "Design of Swap Interleaver for Turbo Codes" *Electronics Letters* vol. 35, no. 22, pp. 1939-1940, 1999.

[11] Garello R., Pierleoni P., Benedetto S.- "Computing the Free Distance of Turbo Codes and Serially Concatenated Codes with Interleavers: Algorithms and Applications", *IEEE Journal on Selected Areas in Comminications* vol. 53, no. 1, pp 800-812, 2001.

[12] Lazar G. A., Buzatu N. R., Cojocariu E., Trifina L. "New deterministic code matched interleaver for turbo codes" *Journal of Theoretical and Applied Information Technology (JATIT)*, vol. 10, no. 2, pp. 89-93, 2009.

[13] Crozier, S., Gracie, K. and Kerr, R., „Estimating the Minimum Distance of Large-Block Turbo Codes using the Event Impulse Method", *Proc. 6th International Symposium on Turbo Codes & Iterative Information Processing*, September 6-10, 2010