POLITECNICO DI TORINO
Repository ISTITUZIONALE

Reducing the memory for iteration-exchanged information and border future metrics in the HomePlug AV turbo decoder implementation

Original
Reducing the memory for iteration-exchanged information and border future metrics in the HomePlug AV turbo decoder implementation / Lorenzo, G., Paola, B., Martina, M., Masera, G.. - STAMPA. - (2012), pp. 180-184. (International Symposium on Turbo Codes and Iterative Information Processing (ISTC) Gothemburg, Sweden 27-31 agosto 2012) [10.1109/ISTC.2012.6325223].

Availability:
This version is available at: 11583/2503018 since:

Publisher:
IEEE / Institute of Electrical and Electronics Engineers Incorporated:445 Hoes Lane:Piscataway, NJ 08854:

Published
DOI:10.1109/ISTC.2012.6325223

Terms of use:

This article is made available under terms and conditions as specified in the corresponding bibliographic description in the repository

Publisher copyright

(Article begins on next page)

12 October 2026

Reducing the memory for iteration-exchanged information and border future metrics in the HomePlug AV turbo decoder implementation

Lorenzo Guerrieri and Paola Bisaglia
DORA S.p.A., STMicroelectronics Group
Via Lavoratori Vittime del Col du Mont 28
11100 Aosta, Italy
Email: lorenzo-dora-spa.guerrieri@st.com,
paola-dora-spa.bisaglia@st.com

Maurizio Martina and Guido Masera
Dipartimento di Elettronica e Telecomunicazioni
Politecnico di Torino
Corso Duca degli Abruzzi, 24
I-10129, Torino, Italy
Email: maurizio.martina@polito.it,
guido.masera@polito.it

***Abstract*—HomePlug AV is the most successful standard for in home power line communications. To combat non-ideality of the power line channel it includes a double binary turbo forward error correcting scheme. Unfortunately, it is known that the memory required by double binary turbo decoders for iteration-exchanged information is roughly three times the memory required for binary turbo codes. Moreover, high throughput implementations based on border state metric inheritance, require additional memories to store border state metrics from an iteration to the next one. This work faces these two aspects by analyzing compression techniques to reduce the amount of memory required to store both iteration-exchanged information and border future metrics. Experimental simulations show that non-uniform quantization and least significant bits dropping allow for a significant memory reduction (up to 30%) with a bit error rate performance loss of about 0.1 dB and a negligible logic gates overhead.**

I. Introduction

Since their discovery, turbo codes [1] attracted a lot of interest due to their excellent error correction capabilities. Indeed, they have been included as a possible Forward Error Correcting (FEC) strategy in several standards both for wireless and wired communication systems, e.g. [2]–[4]. Since a turbo encoder is based on the parallel concatenation of two constituent convolutional encoders through an interleaver, the corresponding decoder relies on the concatenation of two constituent decoders by means of the interleaver. Each constituent decoder is a Soft-Input-Soft-Output (SISO) module that performs a Maximum-A-Posteriori (MAP) estimation [5]. According to [1], [5] this estimation is obtained by processing the data received from the noisy channel, referred to as *intrinsic information*, with the *a-priori* information. Removing the a-priori information and a part of the intrinsic information from the MAP estimation produces the so called *extrinsic* information. Let SISO1 and SISO2 be the two constituent decoders working on in-order and interleaved data, respectively, then the extrinsic information produced by SISO1 becomes the a-priori information of SISO2 and vice-versa. Unfortunately, this decoding algorithm is iterative (one half iteration from SISO1 to SISO2 and one half iteration from SISO2 to SISO1) and the time required to decode a physical block increases with the block length N.

Double binary codes [6] are a viable solution to half the decoding time for a given N. They rely on the use of non-binary codes, where each symbol is made of two bits. This advantage comes at the expense of increasing the amount of memory required to store the extrinsic information along iterations. In particular, since the information is routinely represented in the form of Logarithmic-Likelihood-Ratios (LLRs), this amount of memory is three times the one required for binary codes for a given N. Some works have been proposed in the literature to combat this effect by applying alternative representations that reduce the number of bits [7]–[10] required to represent the extrinsic information.

In this scenario the HomePlug AV standard [4] is an interesting case of study. HomePlug AV uses an orthogonal frequency division multiplexing scheme on the $[1.8-30]$ MHz frequency band to achieve data rates up to 150 Mbit/s with per carrier adaptive modulations ranging from BPSK to 1024-QAM. It is based on a double binary turbo code, whose performance and efficient implementation is addressed by few works in the literature. The aim of this work is to investigate the impact of reducing the number of bits to represent iteration-exchanged information on the performance and complexity of a HomePlug AV turbo decoder implementation. In particular, since the HomePlug AV constituent convolutional encoder is systematic, the intrinsic information for each SISO is made of systematic and parity terms. The systematic term sequence is in natural order so an interleaver is required to obtain the sequence for feeding SISO2. However, this interleaver can be embedded into the one used for extrinsic information exchange if SISO1 sends to SISO2 the result of the addition between the extrinsic information and the intrinsic systematic information (one-interleaver architecture). Unfortunately, since all previous works [7], [8], [10] assumed a decoder architecture based on extrinsic information exchange, the impact of such techniques on a one-interleaver architecture has not been investigated yet. This work aims also to fill this gap analyzing the per-

formance loss introduced by reducing the number of bits to represent iteration-exchanged information in a HomePlug AV one-interleaver architecture context. In particular, in this work the techniques originally proposed in [7] for compressing the extrinsic information in binary turbo codes and in [9] for reducing the bit-width of the extrinsic information are adapted to compress iteration-exchanged information. Finally, further complexity is saved by applying a *least significant bit dropping* technique, inspired by the one proposed in [9] for the extrinsic information, to future border metric encoding with a negligible performance loss.

The paper is organized as follows. In Section II the decoding algorithm is briefly summarized whereas in Section III the compression techniques proposed and compared for iteration-exchanged and future border metrics are described. Section IV presents the experimental results and conclusions are drawn in Section V.

II. Decoding algorithm

Let k be a step in the trellis representation of the constituent convolutional encoder, and u an uncoded symbol. The extrinsic information computed by each constituent decoder is $\lambda_k^{ext}[u] = \sigma \cdot (\lambda_k^{apo}[u] - \lambda_k^{apr}[u] - \lambda_k[\mathbf{c}^u])$ with $\sigma \leq 1$ that is scaling factor [11] and $\lambda_k^{apo}[u]$ is the a-posteriori information:

$$\lambda_k^{apo}[u] = \overset{*}{\max_{e:u(e)=u}}\{b(e)\} - \overset{*}{\max_{e:u(e)=\tilde{u}}}\{b(e)\} \tag{1}$$

where $\lambda_k[\mathbf{c}^u]$ and $\lambda_k^{apr}[u]$ are the systematic component of the intrinsic information and the a-priori information, respectively. $\tilde{u} \in \mathcal{U}$ is an uncoded symbol taken as a reference (usually $\tilde{u} = \mathbf{0}$) and $u \in \mathcal{U} \setminus \{\tilde{u}\}$ with $\mathcal{U}$ the set of uncoded symbols; e is a trellis transition and $u(e)$ is the corresponding uncoded symbol. The $\overset{*}{\max}\{x_i\}$ function is implemented as $\max\{x_i\}$ followed by a correction term often stored in a small Look-Up-Table (LUT) [12]. The correction term, usually adopted when decoding binary codes (Log-MAP), can be omitted for double binary turbo codes with minor Bit Error Rate (BER) performance degradation (Max-Log-MAP). The term $b(e)$ in (1) is defined as:

$$b(e) = \alpha_{k-1}[s^S(e)] + \gamma_k[e] + \beta_k[s^E(e)] \tag{2}$$

$$\alpha_k[s] = \max_{e:s^E(e)=s}\left\{\alpha_{k-1}[s^S(e)] + \gamma_k[e]\right\} \tag{3}$$

$$\beta_k[s] = \max_{e:s^S(e)=s}\left\{\beta_{k+1}[s^E(e)] + \gamma_k[e]\right\} \tag{4}$$

$$\gamma_k[e] = \lambda_k^{apr}[u(e)] + \lambda_k[\mathbf{c}(e)] \tag{5}$$

where $s^S(e)$ and $s^E(e)$ are the starting and the ending states of e, $\alpha_k[s^S(e)]$ and $\beta_k[s^E(e)]$ are the forward and backward metrics associated to $s^S(e)$ and $s^E(e)$, respectively. The term $\lambda_k[\mathbf{c}(e)]$ represents the intrinsic information received from the channel. For further details on the decoding algorithm the reader can refer to [13]. Since in the following analysis we address a one-interleaver architecture, we will refer to iteration-exchanged information as $\lambda_k[I]$ (the input to the SISO) and $\lambda_k[O]$ (the output from the SISO).

III. Compression of iteration-exchanged information and border future metrics

In a double binary turbo code $\mathcal{U} = \{00, 01, 10, 11\}$, as a consequence $\lambda_k[I]$ ($\lambda_k[O]$) is an array containing three elements. Thus, if each element is represented on n_λ bits, each $\lambda_k[I]$ ($\lambda_k[O]$) requires $3 \times n_\lambda$ bits. In the following analysis two techniques to reduce the bit-width of $\lambda_k[I]$ ($\lambda_k[O]$) are discussed and compared in the framework of the HomePlug AV turbo decoder. The former is inspired by the non-uniform quantization proposed in [7] for binary turbo codes. The second one stems from the *bit dropping* technique proposed in [9] for a binary turbo code based on serially concatenated convolutional codes and the WiMax double binary turbo code. Other techniques proposed in other contexts, such the ones in [10] and [14] were also considered. Even if the complexity reduction achieved with these techniques is interesting the BER performance loss introduced by the approximations is not acceptable for the HomePlug AV scenario.

Several quantization schemes have been tested to implement a finite precision model. Among the others the best performance/complexity trade-off was obtained representing the extrinsic information and the state metrics on 7 and 10 bits respectively. As a consequence, in the following paragraphs the performance of this quantization scheme is considered as a reference.

A. Non Uniform Quantization (NUQ)

As argued in [7], non linear quantization of the extrinsic information achieves very good results. In particular, in [7] it is shown that a heuristically-determined nonlinear quantizer allows to represent the extrinsic information with a small number of bits. In this work we extend this concept to the double binary case.

The steps for the quantizer have been determined according to the following idea. As long as the decoding proceeds from an iteration to the next one, the distance among the three extrinsic information corresponding to a couple of bits tends to increase. Thus, hard decision depends mainly on the difference among the components of $\lambda_k[u]$ rather than on their absolute values. As a consequence, we employ a coarser quantization for large metrics and a finer quantization for metrics with low magnitude. The corresponding quantization scheme is shown in Fig. 1. The compression operation performed at the output of the SISO corresponds to generating the array $\hat{\lambda}_k[O]$ using the following equations:

$$\hat{\lambda}_k[O] = \begin{cases} \left\lfloor \frac{|\lambda_k[O]|}{2} \right\rfloor \cdot \mathrm{sgn}(\lambda_k[O]) & |\lambda_k[O]| < 16 \\ \left(\left\lfloor \frac{|\lambda_k[O]|}{4} \right\rfloor + 4 \right) \cdot \mathrm{sgn}(\lambda_k[O]) & 16 \leq |\lambda_k[O]| < 24 \\ \left(\left\lfloor \frac{|\lambda_k[O]|}{8} \right\rfloor + 7 \right) \cdot \mathrm{sgn}(\lambda_k[O]) & 24 \leq |\lambda_k[O]| \leq 64 \end{cases} \tag{6}$$

where $\lfloor \cdot \rfloor$ represents the integer part operation and $\mathrm{sgn}(\cdot)$ is equal to $+1$ and to -1 for positive and negative values of the argument, respectively.

Similar equations can be easily derived for the decompression step. As discussed in Section IV, simulations show that

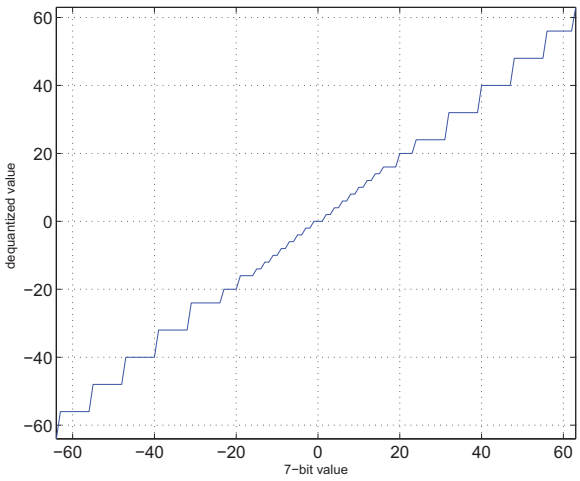

Fig. 1. Heuristically-determined nonlinear quantizer employed for 5-bit representation of the iteration-exchanged information.

this heuristically-determined nonlinear quantizer introduces a BER performance loss of less than 0.1 dB (see Fig. 2).

B. Least Significant Bit Dropping (LSBD)

In [9] it is shown that the amount of memory required to store the extrinsic information in a turbo decoder architecture can be reduced with a limited BER performance loss by dropping the Least Significant Bits (LSBs). As a consequence, each SISO appends a proper number of zeros to the extrinsic information read from the memory. In this work the same idea is applied to iteration-exchanged information instead, namely Q bits are dropped at the output of the SISO. When the SISO reads from the memory the data produced during the previous half iteration, it appends Q zeros to each element of $\lambda_k[I]$. The operation performed at the output of the SISO corresponds to generating the array

$$\hat{\lambda}_k[O] = \left\lfloor \frac{\lambda_k[O]}{2^Q} \right\rfloor . \tag{7}$$

Similarly, at the input of the SISO $\lambda_k[I]$ is obtained as

$$\lambda_k[I] = \hat{\lambda}_k[O] \times 2^Q. \tag{8}$$

In presenting the numerical results in Section IV, we will denote by Q-LSBD the technique just described in order to specify the chosen Q.

C. Compression of border future metrics

As it can be inferred from (3) and (4) state metrics tend to increase as long as the processing proceeds along the trellis. It is known that a possible solution to this problem relies on normalizing the state metrics at each trellis step. However, it has been shown in [15] that the wrapping metrics technique proposed in [16] for the Viterbi decoder can be used as well in turbo decoder architectures not only to reduce the complexity, but also to increase the achievable clock frequency without impairing the BER performance. Moreover, in this work we assume that the decoder works in a sliding-window fashion [17] where the forward recursion is performed first and forward metrics (α) are stored in a buffer. Thus, iteration-exchanged information follows the backward recursion order (β) and so future border metrics initialization is required. According to [18] an effective technique to initialize border metrics without impairing the decoder throughput relies on the inheritance of border metric through consecutive iterations. Moreover, since the turbo code used in the HomePlug AV standard uses tail bitten termination, the first and the last sections of the trellis ought to be initialized as well. As argued in [19] the inheritance technique proposed in [18] can be exploited to initialize the first and last sections of the trellis too. However, these advantages come at the expense of some memory to store the metrics that will be inherited during the next iteration. In [14] a simple yet effective technique to compress border metrics is proposed. Unfortunately, this technique relies on normalized metrics; as a consequence, it does not provide good results when applied to architectures where state metrics are wrapped. To reduce the bit-width of border state metrics we propose to use the LSBD technique. Since the BER performance of the decoder are less sensitive to border state metrics than to the iteration-exchanged information, we give a coarser representation to border state metrics than to the iteration-exchanged information.

IV. Experimental results

In this section we show the experimental results obtained using the techniques described in Section III to reduce the number of bits required to represent iteration-exchanged information and border metrics.

In the reference decoder, the extrinsic information is represented on 7 bits and the future metrics require 10 bits to be represented. The performance of this reference decoder is shown in terms of BER and Block Error Rate (BLER) in Figs. 2 and 3, respectively, as a small-diamond-marked cyan curve for an additive white Gaussian noise (AWGN) channel with BPSK modulation. The performance achieved applying the non-uniform quantization described in Section III-A to iteration-exchanged information is shown as a circle-marked blue curve (labeled "Ex 5 bits, NUQ"). Compared to the reference curve, less than 0.1 dB and 0.15 dB losses are observed in terms of BER and BLER, respectively. On the other hand, the LSBD technique detailed in Section III-B applied to iteration-exchanged information is represented by the cross-marked red (labeled "Ex 6 bits, 1-LSBD") and square-marked green (labeled "Ex 5 bits, 2-LSBD") curves, respectively. As it can be observed, the Q-LSBD technique performs better than the non-uniform quantization only when $Q = 1$, where the losses are very negligible both in terms of BER and BLER. Thus, in the HomePlug AV scenario the non-uniform quantization of iteration-exchanged information is the preferred solution to save 2 bits. This leads to a memory reduction of more than 28% on the memory needed to store iteration-exchanged information and requires a negligible

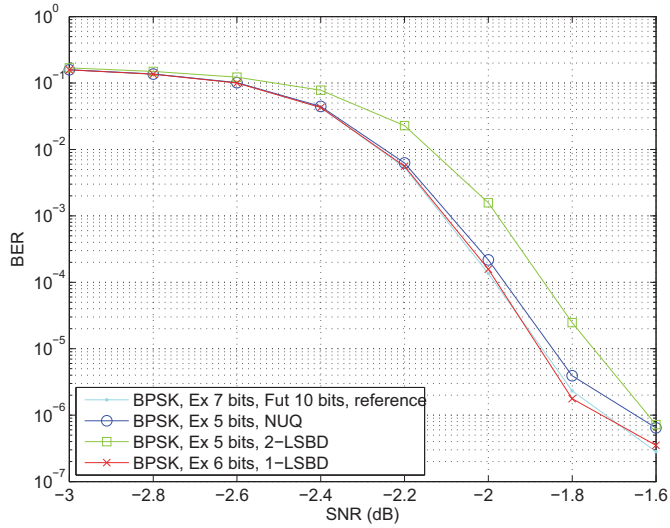

Fig. 2. BER with different compressions of the iteration-exchanged information.

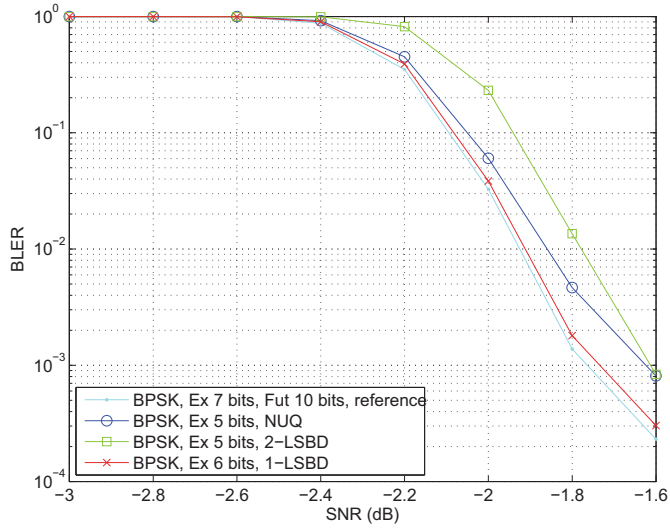

Fig. 3. BLER with different compressions of the iteration-exchanged information.

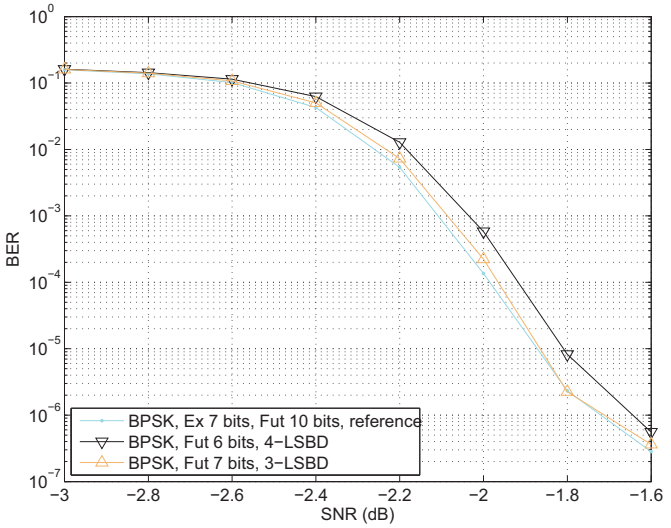

Fig. 4. BER with different compressions of the future border metrics.

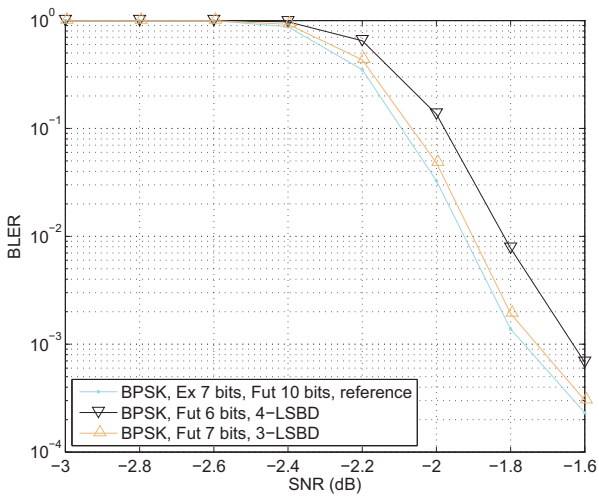

Fig. 5. BLER with different compressions of the future border metrics.

increase in terms of logic gates, about 0.35% of the area of one SISO.

In Figs. 4 and 5, the performance related to the compression of the future border metrics is depicted. As it can be seen, simulations show that reducing the bit-width of each border future metric from 10 to 7 bits (see the triangle-marked brown curve labeled "Fut 7 bits, 3-LSBD") leads to an absolutely negligible performance loss compared to the reference. A further reduction of the bit-width to 6 bits determines a loss which is lower than 0.1 dB and 0.15 dB in terms of BER and BLER, respectively (see the reversed triangle-marked black curve labeled "Fut 6 bits, 4-LSBD" in Figs. 4 and 5). It is worth noting that reducing the border future metrics from 10 to 7 bits leads to a 30% reduction of the memory necessary to store them.

Stemming from these results, we investigated the impact on the decoder performance of the combination of iteration-exchanged information compression and future metric compression. Results are reported in Figs. 6 and 7. In particular, the hexagram-marked yellow curve (labeled "[Ex 6 bits, 1-LSBD], [Fut 7 bits, 3-LSBD]") refers to the LSBD technique applied to both iteration-exchanged information (1-LSBD) and future border metrics (3-LSBD). The losses compared to the reference are very contained and lower than 0.1 dB both in terms of BER and BLER. In order to save one more bit, the pentagram-marked violet curve (labeled "[Ex 5 bits, NUQ], [Fut 7 bits, 3-LSBD]") illustrates the performance of the non-uniform quantization of the iteration-exchanged information combined with the LSBD technique (3-LSBD) applied to border future metrics. The maximum observed losses are within 0.1 dB and 0.15 dB in terms of BER and BLER,

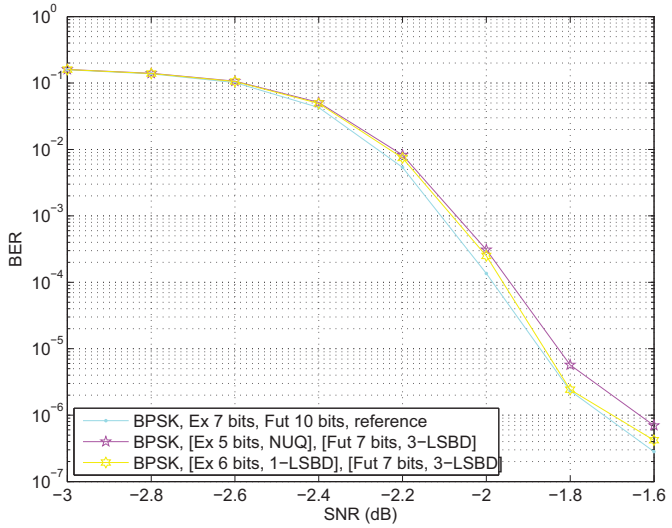

Fig. 6. BER with different compressions of the iteration-exchanged information and future border metrics.

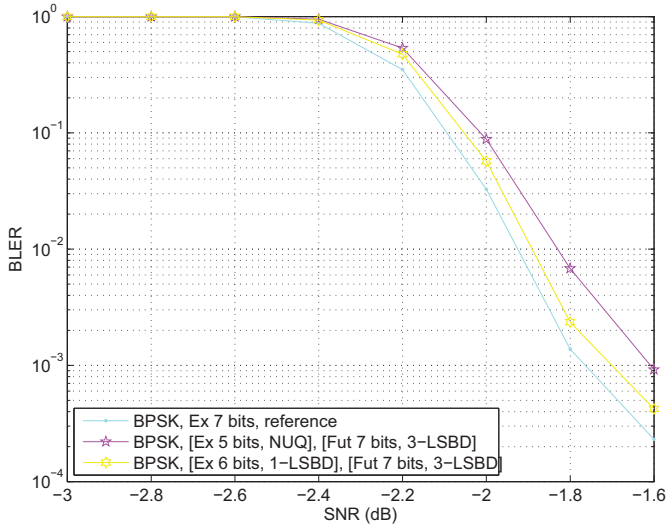

Fig. 7. BLER with different compressions of the iteration-exchanged information and future border metrics.

respectively. Hence, it is possible to obtain both the mentioned 30 % memory saving advantages, by combining conveniently the analyzed compression techniques while still keeping the performance reasonable.

V. Conclusions

In this work memory reduction techniques are applied to the HomePlug AV turbo decoder scenario. Non-uniform quantization and least significant bit dropping have been successfully applied to iteration-exchanged information reducing the bit-width of the corresponding memory from 7 bits to 5 bits. Experimental results highlighted that the non-uniform quantization performs better than least significant bit dropping and it introduces a BER performance loss of less than 0.1 dB on AWGN. Moreover, least significant bit dropping has been applied to each border state metrics reducing the bit-width from 10 to 7 bits with a negligible impact on the error correcting performance of the decoder. The presented techniques allow obtaining nearly 30% memory saving on the storage of the considered metrics with a negligible overhead in terms of logic gates, about 0.35% of the area of one SISO.

Future work could analyze the impact of the proposed techniques on the performance when a realistic power line scenario is considered.

References

[1] C. Berrou, A. Glavieux, and P. Thitimajshima, "Near Shannon limit error correcting coding and decoding: Turbo codes," in *IEEE International Conference on Communications*, 1993, pp. 1064–1070.

[2] "http://www.3gpp.org."

[3] "IEEE Std 802.16, part 16: air interface for fixed broadband wireless access systems," Oct. 2004.

[4] H. P. Alliance, "HomePlug AV baseline specification, version 1.0.00," Dec. 2005.

[5] L. R. Bahl, J. Cocke, F. Jelinek, and J. Raviv, "Optimal decoding of linear codes for minimizing symbol error rate," *IEEE Transactions on Information Theory*, vol. 20, no. 3, pp. 284–287, Mar 1974.

[6] C. Berrou, M. Jezequel, C. Douillard, and S. Kerouedan, "The advantages of non-binary turbo codes," in *IEEE Information Theory Workshop*, 2001, pp. 61–63.

[7] J. Vogt, J. Ertel, and A. Finger, "Reducing bit width of extrinsic memory in turbo decoder realisations," *IEE Electronics Letters*, vol. 36, no. 20, pp. 1714–1716, Sep 2000.

[8] S. M. Park, J. Kwak, and K. Lee, "Extrinsic information memory reduced architecture for non-binary turbo decoder implementation," in *IEEE Vehicular Technology Conference*, 2008, pp. 539–543.

[9] A. Singh, E. Boutillon, and G. Masera, "Bit-width optimization of extrinsic information in turbo decoder," in *International Symposium on Turbo Codes & Related Topics*, 2008, pp. 134–138.

[10] J. H. Kim and I. C. Park, "Bit-level extrinsic information exchange method for double-binary turbo codes," *IEEE Transactions on Circuits and Systems II*, vol. 56, no. 1, pp. 81–85, Jan 2009.

[11] J. Vogt and A. Finger, "Improving the max-log-MAP turbo decoder," *IEE Electronics Letters*, vol. 36, no. 23, pp. 1937–1939, Nov 2000.

[12] S. Papaharalabos, P. T. Mathiopoulos, G. Masera, and M. Martina, "On optimal and near-optimal turbo decoding using generalized max* operator," *IEEE Communications Letters*, vol. 13, no. 7, pp. 522–524, Jul 2009.

[13] E. Boutillon, C. Douillard, and G. Montorsi, "Iterative decoding of concatenated convolutional codes: Implementation issues," *Proceedings of the IEEE*, vol. 95, no. 6, pp. 1201–1227, Jun 2007.

[14] J. H. Kim and I. C. Park, "Double-binary circular turbo decoding based on border metric encoding," *IEEE Transactions on Circuits and Systems II*, vol. 55, no. 1, pp. 79–83, Jan 2008.

[15] E. Boutillon, W. J. Gross, and G. Gulak, "VLSI architectures for the MAP algorithm," *IEEE Transactions on Communications*, vol. 51, no. 2, pp. 175–185, Feb 2003.

[16] A. P. Hekstra, "An alternative to metric rescaling in Viterbi decoders," *IEEE Transactions on Communications*, vol. 37, no. 11, pp. 1220–1222, Nov 1989.

[17] S. Benedetto, D. Divsalar, G. Montorsi, and F. Pollara, "Algorithm for continuous decoding of turbo codes," *IET Electronics Letters*, vol. 32, no. 4, pp. 314–315, Feb 1996.

[18] A. Abbasfar and K. Yao, "An efficient and practical architecture for high speed turbo decoders," in *IEEE Vehicular Technology Conference*, 2003, pp. 337–341.

[19] C. Zhan, T. Arslan, A. T. Erdogan, and S. MacDougall, "An efficient decoder scheme for double binary circular turbo codes," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2006, pp. 229–232.