COMPARISON OF TWO DIFFERENT APPROACHES FOR JOINT SOURCE-CHANNEL DECODING OF VARIABLE LENGTH CODES

Marion Jeanne and Pierre Siohan

IRISA-INRIA
Campus de Beaulieu,
35042 Rennes Cedex, France
email: marion.jeanne@irisa.fr

France Telecom, RESA/BWA
4, rue du Clos Courtel, B.P. 59
35512 Cesson Sévigné Cedex, France
email: pierre.siohan@francetelecom.com

ABSTRACT

Two different approaches providing a joint optimization of source and channel decoding are compared for the decoding of variable length codes. The first one is based on a source-controlled channel decoding principle while the second one applies the turbo principle between the channel and source decoders. Different variants of these two approaches are included in this comparison using computational complexity and transmission performance criteria.

1. INTRODUCTION

It is now well recognized that under complexity or delay constraints, joint source-channel coding (JSCC) and decoding (JSCD) methods can provide efficient alternatives to the tandem transmission scheme that can be derived from Shannon's theorems. However if various JSCC-JSCD decoding methods are now available there is a lack of comparison between some of them.

In this paper, our aim is to compare two different important classes of JSCD methods for the decoding of Huffman-type source codes. The first approach [1, 2] is a source-controlled channel decoding (SCCD) method where an *a priori* source information is inserted into the channel decoder. The source may be assumed to be memoryless or first-order Makov and the method is valid for any channel code admitting a trellis representation, e.g. a turbo code. In the second approach the JSCD method is an application of the turbo principle that leads to an iterative soft decoding sequentially using the channel decoder and the source decoder. In [3, 4] it was assumed that the source was memoryless. Afterwards, in the context of bayesian networks, this initial approach has been extended [5] to cover the case of a first-order Markov source. Note also that, in these two variants of turbo-based JSCD using convolutional codes, the VLC have different trellis representations. In [4] it corresponds to a bit clock while in [3] and [5], it is based on a symbol clock.

In Section 2 we present the main features of these two approaches with their different variants. The complexity issue is presented in Section 3. In Section 4 both approaches are compared in the case of a transmission over an additive white Gaussian noise (AWGN) channel. Finally, in conclusion, we summarize the pros and cons of each approach.

2. THE TWO JSCD APPROACHES

Our comparison is carried out for first-order Markov sources and it is assumed that the exact *a priori* source information is known at the receiver. Our purpose is also to compare JSCD systems that only have one single loop of iteration. For the SCCD approach we choose to protect the data with a turbo code, i.e. the iteration loop is located at the channel decoder level. When applying the turbo principle in an iterative source-channel decoding (ISCD) approach, it is necessary that the iteration loop includes both decoders. Therefore, in this latter case, we choose a convolutional channel code with the same code rate as for the turbo code. Note that in the case of standard communication applications, to be applicable, the ISCD approach requires specific procedures allowing exchanges between physical and application layers [6], which is not strictly necessary for the SCCD approach.

For the source decoding of Huffman codewords, three main possibilities are offered. The ISCD approach obviously involves a soft-input soft-output decoder that, as mentioned above, may operate at a bit or symbol level. The third and most simple possibility is related to the SCCD approach that can be combined with a hard Huffman decoder.

2.1 Source-controlled turbo decoding of VLC

In [1, 2] the key idea is to improve the channel decoding operation using an *a priori* source information related to source symbol probabilities. This information is computed at a bit level in order to modify the channel decoding procedure being used as little as possible. For a first-order Markov source, knowing the symbol transition probabilities, we compute, at first, the corresponding edge probabilities in the VLC tree. Then the JSCD technique involves maintaining a one-to-one correspondence between edges of the VLC tree and edges of the channel decoding trellis, at the receiver level. Thus, the *a priori* probability bit information can be inserted afterwards at the right state of the right stage of the channel decoding trellis. As indicated in [1], due to the internal turbo code interleaver, this insertion is only carried out in the first constituent convolutional decoder.

The corresponding transmission scheme is depicted in Fig. 1. In our comparison the channel code used is a turbo code (TC). To get estimates of the sequence of transmitted symbols, various source decoding methods can be implemented. As in [1], we can proceed to a standard hard decoding of the VLC. Then the SCCD scheme corresponds to a turbo decoding followed by a hard decoding of the VLC bitstream, we denote it (TC-HD). Instead of this, we would favour a soft decoding of the VLC using either the bit-based trellis introduced in [7], or the symbol-based trellis presented in [3]. These JSCD techniques will be called turbo codes with soft bit decoding (TC-SBD) and turbo codes with soft symbol decoding (TC-SSD), respectively. It can be noted

This work is supported in part by the european network of excellence NewCom.

that in the last two variants the *a priori* source information is used twice: at first in the turbo decoding and, secondly, at a bit or a symbol level, in the source decoding trellis.

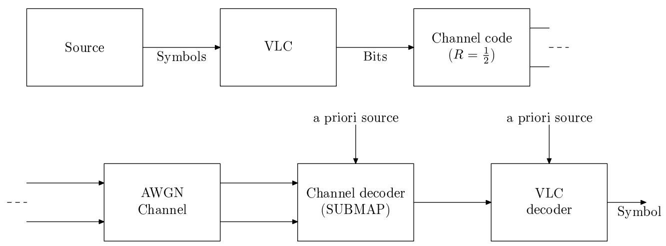

Figure 1: Source-controlled turbo decoding of VLC using *a priori* source information.

2.2 Iterative source channel decoding of VLC

The second JSCD method corresponds to an application of the turbo principle with a serial turbo code composed of a first constituent VLC encoder and a second constituent being a convolutional code [4, 3, 5]. Each decoding block may be fed with soft inputs and can deliver soft outputs. This soft information is exchanged, in an iterative process, between the channel decoder and the source decoder. By analogy with serial turbo codes, in order to avoid correlated errors at the decoder side, an interleaver, I, is inserted in the overall encoder between the source and channel coders. The corresponding transmission scheme, with the deinterleaver denoted by I^*, is depicted in Fig. 2. Two different trellis descriptions are proposed for the VLC, one operates at a bit level and the other one at a symbol level. This leads us to two different variants of the ISCD approach, one is named "convolutional code soft bit decoding" (CC-SBD) and the other one is named "convolutional code soft symbol decoding" (CC-SSD). As, in both algorithms, the interleaver breaks the one-to-one correspondence between edges of the VLC tree and the channel decoder trellis, the *a priori* source probabilities are only used for soft source decoding.

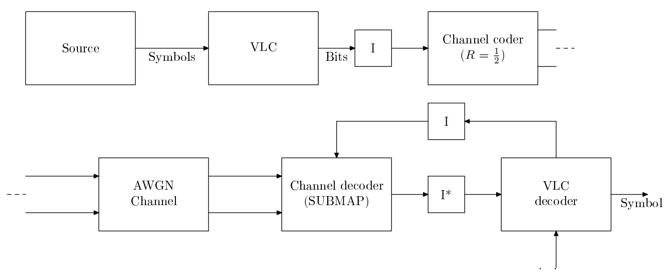

Figure 2: Iterative decoding scheme between source and channel for the JSCD of VLC.

As it is usual for JSCD of VLC, in all our schemes the source data are packetized in order to limit the error propagation. For methods using a hard decoding of the VLC, or a soft decoding with a bit-based trellis, we only use the number of bits per packet. Otherwise, for a symbol-based VLC trellis decoder, the number of bits and symbols per packet are required.

Table 1: Complexity order expressed in number of states.

Method	Complexity
Tandem	$N_i \times (2\ \ \times M + 2\ \ \times M)$
TC-HD	$N_i \times (2\ \ \times M + 2\ \ \times M)$
TC-SBD	$N_i \times (2\ \ \times M + 2\ \ \times M) + (N-1) \times M$
CC-SBD	$N_i \times (2\ \ \times M + (N-1) \times M)$

3. COMPLEXITY COMPARISON

The comparison is carried out for the basic JSCD schemes and their different variants, i.e. five decoding schemes. In addition, to get a reference with respect to a conventional scheme, we also present the computational cost for a tandem scheme which corresponds to a turbo channel decoder not using the *a priori* source information. For all systems the channel decoding is based on a Max-Log-MAP algorithm [8]. For the soft source decoding of the VLC, a BCJR algorithm [9] has to be used for the symbol-based trellises [3] while a Max-Log-MAP algorithm can be applied for the bit-based ones.

A first fundamental evaluation of the complexity may be given by the number of states (or nodes) involved by the different schemes. We denote by K the number of transmitted symbols and by M the number of transmitted bits. The number of nodes in a symbol-based trellis is in $O(K^2)$ or, equivalently, in $O(M^2)$ [10]. For a bit-based trellis, such as the one derived from [7], and for a CC trellis, the number of nodes is in $O(M)$. Moreover, as a BCJR algorithm involves much more basic operations than a Max-Log-MAP algorithm [8], it is obvious that the two methods using a symbol-based trellis for source decoding (TC-SSD and CC-SSD) are much more complex than the other ones.

The four other techniques are of comparable complexity, a more precise evaluation is therefore required. As the same decoding algorithm, here a Max-Log-MAP, is used, the number of nodes in the segmentation trellises provides the first and more important indication. Let the CC memory be denoted by . The number of states for each stage of the CC decoder is 2 , so, the total number of states is $2\ \ \times M$. For a source with N symbols, the number of nodes in the corresponding bit-based trellis is $(N-1) \times M$. Then, denoting the number of iterations by N_i, we get in Table 1 the order of complexity of each decoding method. Table 1 clearly shows that, in a first approximation, the four decodings are of similar complexity for usual values of the different parameters, e.g. a few units for   and N_i, around tens for N and thousands for M. If, as in [11], the CC is replaced by a TC in an ISCD scheme, there is a double loop of iteration that adds a significant extra cost. However, this complexity remains in $O(M)$ when using a bit-based trellis.

There are also some differences of less importance between the decoding schemes that do not appear in Table 1. For the SCCD schemes, the computation of the *a priori* source in the VLC tree is pre-computed and does not enter in the decoder complexity. Furthermore, the insertion of this *a priori* in the first component of the turbo decoder [2] does not increase the number of states. However, the size of the states has to be increased with the addition of a pointer in order to keep the one to one relation with the VLC tree. Moreover, a multiplication by an *a priori* source probability has to be realized in the Max-Log-Map algorithm.

Table 2: Transition probabilities of the first-order 3-symbol Markov source.

$Y \downarrow \| X \rightarrow$	a	b	c
a	0.94	0.03	0.03
b	0.18	0.712	0.108
c	0.18	0.108	0.712

4. SIMULATION RESULTS

The JSCD systems have been designed and parameterised in order to allow fair comparisons between systems of similar complexities, see Table 1. We also measure the impact on the transmission performance of the extra complexity introduced when using symbol-based VLC decoders. Each packet at the source output contains M bits, if needed stuffing bits are added to complete the packet. M is also the size of the line-column, or block, interleaver used in each system. The CC and TC have the same code rate ($R = \frac{1}{2}$). For the SCCD scheme, the turbo code used corresponds to a parallel concatenation of two recursive systematic coders with the same generator $(1+D+D^2+D^4)/(1+D^3+D^4)$, i.e, $= 4$. This polynomial is also used in the ISCD method for the CC. Comparisons are carried out over an AWGN channel, with the symbol error rate (SER) criterion for a given ratio of the useful bits energy over noise ($\frac{E_b}{N_0}$).

4.1 Comparison for a small size source ($N = 3$)

The six decoding schemes have been tested at first with a first-order 3 symbol Markov source {a, b, c} encoded as {0, 10, 11}, respectively. The first-order symbol correlation is introduced with probability transitions ($Pr(X|Y)$) given in Table 2. Denoting the relative residual redundancy of a source, i.e. the difference between the Huffman code rate and the source entropy divided by the Huffman code rate, by R_r, it can be checked that $R_r = 0.54$. In Fig. 3 we report a first set of results for $M = 16 \times 16$ i.e. packets of length 256 bits corresponding on average to 205 symbols. In Fig. 4 two different sizes of interleaver are considered. For the bit-based VLC decoders (SBD) we can go up to $M = 64 \times 64$, i.e. packets of size 4096 bits, while for the symbol-based decoders (SSD) the complexity of the corresponding symbol trellises becomes too high and, therefore, we do not go beyond $M = 32 \times 32$.

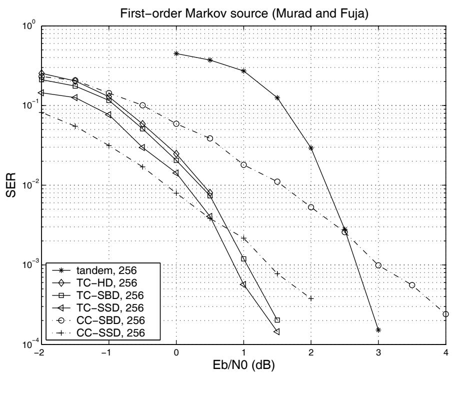

Figure 3: Comparison, after three iterations, of symbol error rates for the first-order 3-symbol Markov source with VLC encoding and a 16×16 block interleaver.

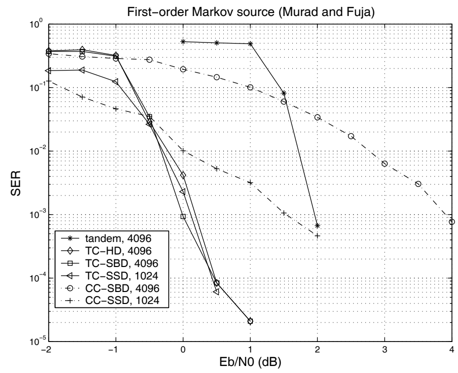

Figure 4: Comparison, after three iterations, of symbol error rates for the first-order 3-symbol Markov source with VLC encoding and with 32×32 and 64×64 block interleavers.

4.1.1 *Comparison between SCCD variants*

For the source-controlled channel turbo decoding, it can be firstly noted that a soft decoding of the VLC (TC-SBD and TC-SSD) does not bring a significant improvement compared to a simple hard decoding (TC-HD). On the other hand it can be seen in Fig. 3 and 4 that this JSCD approach takes advantage of an increase of the interleaver size. For instance, at $SER = 10^{-3}$, there is a 0.3 dB gain when the interleaver size is increased from 16×16 to 64×64 for the TC-SBD solution. These two features clearly illustrate the TC impact on the SCCD results.

4.1.2 *Comparison between ISCD variants*

In the case of the ISCD approach, it appears that a symbol-based (CC-SSD) trellis for the soft decoding of the VLC provides a significant improvement compared to the bit-based decoding (CC-SBD). Thus, with the ISCD approach we can really take advantage of the extra complexity, i.e, a higher number of states, a BCJR decoding and a perfect knowledge of M and K. Indeed, at $SER = 10^{-2}$, a SSD yields 1.75 dB gain compared to a SBD when $M = 16 \times 16$ and the gain goes up to 2.8 dB when $M = 32 \times 32$. It can also be noted that an increase of the interleaver size may result in a loss of performances. Indeed, there is a 0.16 dB loss for the CC-SSD variant when M goes from 16×16 to 32×32 and a 1.25 dB loss for the CC-SBD variant when M is increased from 16×16 to 64×64. But naturally, as for any decoding method based on the turbo principle, this interleaver is essential to have the JSCD scheme working properly.

4.1.3 *Comparison between SCCD and ISCD variants*

It can be noted at first that for all curves we recover the typical behavior of the channel codes, i.e. a steeper slopes for TC than for CC. At high SER, the CC-SSD variant gives better results than all decoding methods with turbo codes, while at low SER we are in the inverse situation. This tends to prove that, if the relative residual redundancy is sufficiently important, ISCD techniques can provide better results than SCCD techniques with powerful turbo codes. But that is only true at high SER. At low SER, i.e. high SNR ratios over the AWGN channel, the source impact on the source-controlled decoder becomes less significant, and the error correcting code becomes the predominant element.

4.2 Comparison for a medium size source ($N = 16$)

Let us now check with VLC sources of higher dimensions that the conclusions drawn for the 3-symbol source are still valid. Naturally, from the complexity aspect, an increase in size of the source alphabet constitutes a strong penalty for all JSCD techniques using a soft source decoding algorithm, particularly for the symbol-based ones. Fig. 5 and Fig. 6 present the simulation results for a first-order Gauss-Markov source with a correlation coefficient of 0.9, a 16-level uniform quantizer, and a Huffman encoder. Its relative residual redundancy is $R_r = 0.33$. Two interleaver sizes, i.e. two data packet sizes are considered. In Fig. 5 the interleaver and the packet size are 576 bits and in Fig. 6 they are fixed to 4096 bits.

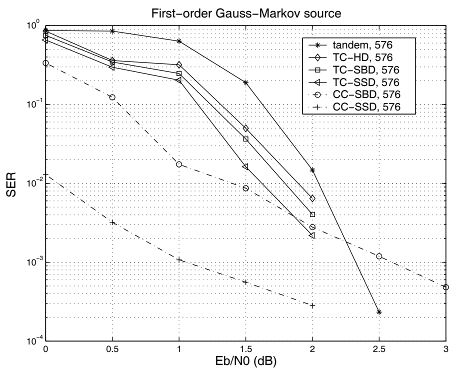

Figure 5: Comparison, after three iterations, of symbol error rates for the Gauss-Markov source with a 16-level quantizer, VLC encoding and a 24×24 block interleaver.

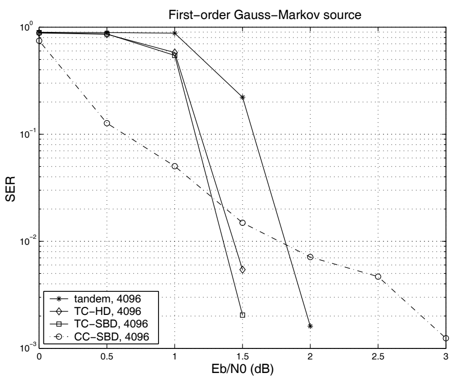

Figure 6: Comparison, after three iterations, of symbol error rates for the Gauss-Markov source with a 16-level quantizer, VLC encoding and a 64×64 block interleaver.

In these displays we recover some of the features already noted for $N = 3$. However, as R_r is smaller, the TC variants that directly use the *a priori* source do not give as good performances as with the 3-symbol source. At the contrary the CC variants seem less affected by the decrease of R_r. Indeed, they give results very close to the ones obtained with the 3-symbol source. Hence, at $SER = 10^{-2}$, the CC-SSD variant with a small interleaver size is 1.2 dB better than the TC-SBD variant with a big interleaver size.

5. CONCLUSION

We have presented a comparison between two JSCD approaches in the case of first-order Markov sources being Huffman encoded. The first JSCD method corresponds to a source-controlled channel decoding approach where the channel decoder takes advantage of the VLC source statistics. The second one is an application of the turbo principle between source and channel decoders. Compared to a classical tandem decoding scheme, both approaches allow performance gains that increase with the residual redundancy of the source. Each method can use two different soft source decoding strategies; the first one uses a bit trellis and the second one a symbol trellis. This later one requires a much higher complexity. In the case of the ISCD scheme, this extra complexity leads to a significant improvement compared to the results obtained with a bit-based trellis. It is not the case for the source-controlled JSCD methods. Finally, for high SER, a comparison of the best variants for each approach shows that the variant called CC-SSD outperforms the one named TC-SSB. At low SER we are in the inverse situation.

REFERENCES

[1] L. Guivarch, J-C. Carlach, and P. Siohan, "Joint source-channel soft decoding of Huffman sources with Turbocodes," in *Proc. of DCC*, Snowbird, USA, Mar. 2000, pp. 83–92.

[2] M. Jeanne, J-C. Carlach, and P. Siohan, "Joint source channel decoding of variable length codes for convolutional codes and turbo codes," *IEEE Trans. on Communications*, vol. 53, no. 1, pp. 10–15, Jan. 2005.

[3] R. Bauer and J. Hagenauer, "Iterative source/channel-decoding using reversible variable length codes," in *Proc. of DCC*, Snowbird, USA, Mar. 2000, pp. 93–102.

[4] R. Bauer and J. Hagenauer, "On variable length codes for iterative source/channel decoding," in *Proc. of DCC*, Snowbird, USA, Mar. 2001, pp. 273–282.

[5] A. Guyader, E. Fabre, C. Guillemot, and M. Robert, "Joint source-channel turbo decoding of entropy-coded sources," *IEEE Jour. on Selec. Areas in Communications*, vol. 19, no. 9, pp. 1680–1696, Sept. 2001.

[6] S. Merigeault and C. Lamy, "Concepts for exchanging extra information between protocol layers transparently for the standard protocol stack," in *Proceedings of 10th International Conference on Telecommunications ICT'2003*, Tahiti, Feb. 2003.

[7] V. B. Balakirsky, "Joint source-channel coding for variable length codes," in *Proc. of ISIT*, Ulm, Germany, June/July 1997, p. 419.

[8] P. Robertson, P. Hoeher, and E. Villebrun, "Optimal and sub-optimal maximum a posteriori algorithms suitable for turbo decoding," *European Transaction on Telecommunication*, vol. 8, no. 2, pp. 119–126, 1997.

[9] L. R. Bahl, J. Cocke, F. Jelinek, and J. Raviv, "Optimal decoding of linear codes for minimizing symbol error rate," *IEEE Trans. on Information Theory*, pp. 284–287, Mar. 1974.

[10] C. Weidmann, "Reduced complexity soft-in soft-out decoding of variable-length codes," in *Proc. of ISIT*, Yokohama, Japan, July 2003.

[11] X. Jaspar and L. Vandendorpe, "New iterative decoding of variable length codes with turbo-codes," in *Proc. of ICC*, Paris, France, June 2004, vol. 5, pp. 2606–2610.