Joint Source-Channel Turbo Coding for Binary Markov Sources

Guang-Chong Zhu, *Member, IEEE* and Fady Alajaji, *Senior Member, IEEE*

***Abstract*— We investigate the construction of joint source-channel (JSC) Turbo codes for the reliable communication of binary Markov sources over additive white Gaussian noise and Rayleigh fading channels. To exploit the source Markovian redundancy, the first constituent Turbo decoder is designed according to a modified version of Berrou's original decoding algorithm that employs the Gaussian assumption for the extrinsic information. Due to interleaving, the second constituent decoder is unable to adopt the same decoding method; so its extrinsic information is appropriately adjusted via a weighted correction term. The Turbo encoder is also optimized according to the Markovian source statistics and by allowing different or asymmetric constituent encoders. Simulation results demonstrate substantial gains over the original (unoptimized) Turbo codes, hence significantly reducing the performance gap to the Shannon limit. Finally, we show that our JSC coding system considerably outperforms tandem coding schemes for bit error rates smaller than 10^{-4}, while enjoying a lower system complexity.**

***Index Terms*— Joint source-channel coding, turbo codes, AWGN and Rayleigh fading channels, Shannon limit, Markov sources, iterative decoding, bit error rate.**

I. Introduction

THE fundamental goal of a communication system is to efficiently and reliably transmit information data from a source to a destination over a physical channel where noise distortion may occur. For the sake of efficiency, the source output, which often contains a substantial amount of redundancy, is usually compressed prior to transmission. This *data compression* procedure (also known as *source coding*), renders the sources more vulnerable to noise corruption. In order to make the transmission reliable, *channel coding* is usually employed to combat the noise by adding controlled redundancy to the data.

Traditionally, source and channel coding are implemented independently, resulting in the so-called *tandem* coding scheme.[1] Furthermore, in almost all the theory and practice of error-control coding, the data at the input of the channel encoder is assumed to be uniform and memoryless. Obviously, this is true only when the source coding part is optimal; that is, the source-coded bit stream contains zero redundancy. However, in reality, most existing source coding schemes are only sub-optimal (particularly fixed-length source codes), resulting in a certain amount of *residual* redundancy in the compressed bit stream. For example, the 2.4 kbits/s US Federal Standard MELP speech vocoder produces pitch and gain parameters that contain 43 to 52.7% of residual redundancy due to non-uniformity and memory [14]. For uncompressed sources such as image and speech signals, the natural redundancy can be much higher. Therefore, transmission of sources with a considerable amount of natural or residual redundancy is an important issue. Several studies (e.g., [2], [3], [14], [22], [25], [28]–[31], [34], etc.) have shown that when the source redundancy is exploited in the channel coding design, the system performance can be significantly improved.

The introduction of Turbo codes in 1993 [7] is regarded as one of the most significant achievements in channel coding. Since then, there has been a large number of publications regarding various aspects of Turbo codes. The original work by Berrou *et al.* demonstrated excellent performance of Turbo codes for the transmission of uniform memoryless sources over additive white Gaussian noise (AWGN) channels [7], [8]. Turbo codes were later extended to Rayleigh fading channels showing comparable performance [21]. However, most Turbo coding studies assume that the input stream to the encoder is a uniform memoryless source. To the best of our knowledge, only limited attention has been paid to the problem of exploiting the source redundancy in the Turbo coding context. This is in essence a joint source-channel (JSC) coding issue. The design of (systematic and non-systematic) Turbo codes for the transmission of non-uniform memoryless sources has been recently studied in [35], [36], where close to Shannon limit performance was achieved. For the scenario of sources with memory, Garcia-Frias *et al.* [15], [16] investigated the design of Turbo codes for Hidden Markov sources over AWGN channels. Also in [14], Fazel *et al.* considered employing Turbo codes for the MELP speech coder output whose parameters are modeled by Markov chains. In [15], [16] and [14], the modifications to the BCJR algorithm [4] are only proposed for the first constituent decoder. Due to interleaving, the second constituent decoder remains the same as for memoryless sources. Another related work is by Cai *et al.* [11], who proposed a simple modification to the extrinsic information terms of both constituent decoders; but their decoding algorithm remains the same as for uniform memoryless sources. Other related source-channel coding works based on the Turbo coding principle include [1], [17], [20].

Manuscript received December 9, 2003; revised September 29, 2004 and March 8, 2005; accepted April 9, 2005. The associate editor coordinating the review of this paper and approving it for publication was R. Negi. This work was supported in part by the Natural Sciences and Engineering Research Council (NSERC) of Canada and the Premier's Research Excellence Award (PREA) of Ontario. Part of this work was presented at the *Canadian Workshop on Information Theory*, Waterloo, Ontario, Canada, May 2003.

Guang-Chong Zhu is with the Dept. of Mathematics and Computer Science, Lawrence Technological University, 21000 West Ten Mile Road, Southfield, MI, USA (e-mail: gzhu@ltu.edu).

Fady Alajaji is with the Dept. of Mathematics and Statistics and Dept. of Electrical and Computer Engineering, Queen's University, Kingston, Ontario, Canada (e-mail: fady@mast.queensu.ca).

Digital Object Identifier 10.1109/TWC.2006.05014

[1] This is justified by Shannon's well-known separation principle (or information-transmission theorem) [23] for single-user communication systems which states that there is no loss of performance optimality if separately designed optimal source and channel codes are implemented in tandem as opposed to designing the codes jointly or using a single optimal joint source-channel code. This result inherently assumes that unlimited encoding/decoding delay and complexity are available at the transmitter and the receiver.

1536-1276/06$20.00 © 2006 IEEE

In this work, we consider stationary ergodic binary first-order Markov sources. We investigate the design of (systematic) Turbo codes for transmitting such Markov sources over binary phase-shift keying (BPSK) modulated AWGN channels and Rayleigh fading channels with known channel state information. One main goal is to obtain the best performance that is as close to the Shannon limit as possible. The proposed framework can be extended to high-order Markov sources by using non-binary Turbo codes. Our contributions regarding decoding and encoding design are as follows.

- According to the general principle of iterative decoding [7], [24], after decomposing the log-likelihood ratio (LLR) generated by the decoder of each constituent code, the channel transition term and the a priori term should not be passed on to the other constituent decoder; otherwise, performance degradation would occur. When the BCJR algorithm is modified for the first decoder to incorporate the Markovian property of the source, the LLR can only be decomposed into two separate terms (instead of three as in the case of memoryless sources), where the new extrinsic information contains the a priori term. Furthermore, the extrinsic information generated from the second constituent decoder cannot be used in the same fashion as the a priori term for the first constituent decoder to "update" the new extrinsic information.
  In the first Turbo coding paper by Berrou *et al.* [7], the original BCJR decoding algorithm was relatively complex to implement and, as a result, the extrinsic information was approximated by a Gaussian distribution and modeled as an additional input sequence to the other constituent decoder. Later Robertson [24] improved and simplified Berrou's BCJR algorithm, and his technique has been widely adopted ever since. We however herein observe that, for the Turbo coding of Markov sources, Berrou's original method with the Gaussian assumption is indeed a valid solution (to the above issues) for the first constituent decoder.
- Due to interleaving, the second constituent decoder is unable to adopt the above method; it thus employs Robertson's decoding technique. Nevertheless, we attempt to exploit the source memory by modifying its extrinsic information term using a weighted correction factor involving the Markov source statistics. Our proposed technique, is an improvement of an earlier method of Cai *et al.* [11].
- Since the two constituent decoders employ different decoding algorithms and thus have different error-correcting capabilities, a symmetric Turbo code (i.e., a code with identical constituent encoders) may not be the best structure. We herein allow the use of asymmetric encoders. Furthermore, as revealed in [35], [36], optimization with respect to the choice of the tap coefficients in each constituent encoder plays an important role vis-a-vis system performance. Therefore, for a given source transition probability matrix, we optimize the encoder structure to further improve the performance.

Our resulting JSC Turbo coding system demonstrates substantial gains (from 0.45 up to 3.57 dB) over the original Turbo codes that do not exploit the source statistics at both encoder and decoder. Furthermore, it performs within 0.73 to 1.45 dB from the Shannon limit for symmetric Markov sources, and approximately within 0.64 to 1.67 dB for asymmetric Markov sources.

In relation to previous works in [15], [16], our approach regarding encoding/decoding design is different in the following aspects: (a) we solve the design problem of the first constituent decoder in order to exploit the source memory using Berrou's [7] original Gaussian approximation method for the extrinsic information, while [15], [16] employ a trellis based generalized version of Robertson's method [24]; (b) we attempt to exploit (although to a limited extent) the source redundancy in the second constituent decoder in spite of the use of interleaving; (c) [15], [16] do not optimize the encoding structure since a fixed symmetric encoder is employed. However, it is important to point out that [15], [16] consider a wider class of sources with memory, hidden Markov sources. Furthermore [16] introduces a new "universal" Turbo decoder that does not require advance knowledge of either the hidden Markov source statistics or the channel noise variance. The decoder instead iteratively estimates the source parameters and the channel noise power, and it is shown to produce no noticeable performance degradation.

Finally, we compare, for identical transmission rates, the performance of our JSC coding system with that of two tandem coding schemes, which employ an 8^{th}-order Huffman code followed by a standard Turbo code. The 8^{th}-order Huffman code performs near optimal data compression while the standard Turbo code is chosen such that it gives either an excellent water-fall bit error rate (BER) performance, or a lower error-floor performance with a slightly worse water-fall performance. It is shown that the tandem schemes inevitably suffer from a high error-floor effect for medium to high signal-to-noise ratios (SNRs), while our less complex JSC coding system offers a robust and substantially superior performance (for BERs $\leq 10^{-4}$).

The rest of the paper is organized as follows. In Section II, we give a brief description of the system. In Section III, Turbo decoder design for Markov sources is discussed. Turbo encoder optimization is then described in Section IV. In Section V, we address the determination of the Shannon limit for the transmission of binary Markov sources over AWGN and Rayleigh fading channels. Simulation results are presented in Section VI. In Section VII, we compare the performance of our system with two tandem coding schemes. Finally, concluding remarks are given in Section VIII.

II. System Description

We first describe the JSC Turbo coding system for the transmission of binary Markov sources over noisy channels. We consider a stationary ergodic binary first-order Markov source $\{U_k\}$, whose transition probabilities are described by the transition matrix:

$$\Pi = [\pi_{ij}] = \begin{bmatrix} q_0 & 1-q_0 \\ 1-q_1 & q_1 \end{bmatrix},$$

where the transition probability is defined by

$$\pi_{ij} \overset{\triangle}{=} Pr\{U_k = j | U_{k-1} = i\}, \quad i,\ j \in \{0, 1\}.$$

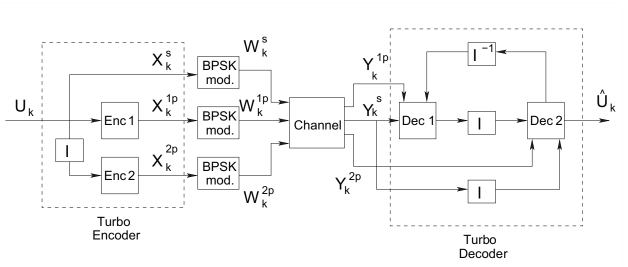

Fig. 1. Block diagram of the system ($\mathbf{I}$ and $\mathbf{I}^{-1}$ stand for the interleaving and de-interleaving operations, respectively).

We denote the marginal distribution of the source as (p_0, p_1) where $p_0 \stackrel{\triangle}{=} Pr\{U_k = 0\} \stackrel{\triangle}{=} 1 - p_1$. By stationarity, it can be easily shown that p_0 and p_1 are given by the source stationary distribution:

$$p_0 = 1 - p_1 = \frac{1 - q_1}{2 - q_0 - q_1}. \tag{1}$$

In general, the source described above is an *asymmetric* Markov source (i.e., with $q_0 \neq q_1$); its redundancy is in the form of both memory and non-uniformity. When $q_0 = 1 - q_1$, the source reduces to a memoryless source with marginal distribution given by $p_0 = q_0$ and $p_1 = q_1$; in this case, the source redundancy is purely in the form of non-uniformity. When $q_0 = q_1 \neq 1/2$, the source becomes a *symmetric* Markov chain with a *uniform* marginal distribution, i.e., p_0=p_1=1/2; in this case, the source redundancy is strictly in the form of memory. When the source is symmetric, we write $q_0 = q_1 \stackrel{\triangle}{=} q$.

A Turbo code encoder consists of two systematic recursive convolutional encoders in parallel concatenation and linked by an interleaver, which re-arranges the information sequence into a different order. In general, there can be more than two constituent encoders and they do not have to be identical. Each constituent encoder has memory size m; therefore, the number of possible encoder states is 2^m. If at time k, the first constituent encoder is in state S_k, then an input bit U_k brings the encoder state into S_{k+1}, generating a parity bit X_k^{1p} (for the second constituent decoder, the parity bit is X_k^{2p}). To keep the notation consistent, we denote the systematic bit as X_k^s, which is identical to U_k. The pair (X_k^s, X_k^{1p}) is denoted by X_k, and after transmission over the channel, it becomes $Y_k = (Y_k^s, Y_k^{1p})$, the noise corrupted version. We denote L-tuples by $Y_1^L \stackrel{\triangle}{=} (Y_1, Y_2, \cdots, Y_L)$, and we use capital letters for random variables and lower case letters for their realizations.

Corresponding to the two parallel concatenated constituent encoders, a Turbo code decoder has two constituent decoders in serial concatenation, separated by an interleaver which is identical to the one used in the encoder. In the feedback loop from the output of the second constituent decoder to the input of the first constituent decoder, a de-interleaver is used to permute the sequence back into the original order.

Our JSC coding system is very simple. As depicted in Fig. 1, a binary Markov source $\{U_k\}$ is directly fed into the Turbo encoder without source encoding. The Turbo coded sequence $\{X_k\}$ is then BPSK-modulated as $\{W_k\}$ (under an average symbol energy of $E_s = 1$) before transmission over the channel. The channel model considered here can be either an AWGN channel or a Rayleigh fading channel described by

$$Y_k = A_k W_k + N_k, \quad k = 1, 2, \cdots, L,$$

where $\{N_k\}$ is a memoryless or independent and identically distributed (i.i.d.) Gaussian noise source with zero mean and variance $N_0/2$. For the AWGN channel, $A_k = 1, k = 1, 2, \cdots$, while for the Rayleigh fading channel, the amplitude fading process $\{A_k\}$ (also known as the channel state information) is assumed to be i.i.d. and Rayleigh distributed. We assume that $\{A_k\}$ is known at the decoder, and that A_k, W_k, and N_k are independent of each other. At the receiver end, Turbo decoding is applied on the received sequence $\{Y_k\}$ to yield the decoded sequence $\{\hat{U}_k\}$.

III. Turbo Decoder Design for Markov Sources

A. BCJR Algorithm for I.I.D. Sources

For the sake of clarity and completeness, we first briefly describe the BCJR algorithm for i.i.d. sources. There are several slightly different versions of the BCJR algorithm in the literature; the one we present here is due to Robertson [24].

To decode $\{U_k\}_{k=1}^L$, based on the observation of the whole received sequence Y_1^L, each constituent decoder evaluates the conditional *log-likelihood ratio (LLR)* for every bit U_k defined by

$$\Lambda^{iid}(U_k) = \log \frac{Pr\{U_k = 1 | Y_1^L = y_1^L\}}{Pr\{U_k = 0 | Y_1^L = y_1^L\}}.$$

For memoryless sources and memoryless channels, the LLR can be decomposed into three separate terms:

$$\Lambda^{iid}(U_k) = L_{ch}(U_k) + L_{ex}^{iid}(U_k) + L_{ap}^{iid}(U_k), \tag{2}$$

which are called the channel transition term, the extrinsic information term, and the a priori term, respectively. Their expressions are given as follows.

$$\begin{aligned}
L_{ch}(U_k) &= \log \frac{p\{Y_k^s = y_k^s | U_k = 1\}}{p\{Y_k^s = y_k^s | U_k = 0\}}, \\
L_{ex}^{iid}(U_k) &= \log \frac{\sum_s \sum_{s'} \gamma(y_k^p, 1, s|s') \alpha_{k-1}(s') \beta_k(s)}{\sum_s \sum_{s'} \gamma(y_k^p, 0, s|s') \alpha_{k-1}(s') \beta_k(s)}, \\
L_{ap}^{iid}(U_k) &= \log \frac{Pr\{U_k = 1\}}{Pr\{U_k = 0\}},
\end{aligned}$$

where $p(\cdot)$ denotes a probability density function, and $\alpha_k(s)$, $\beta_k(s)$ and $\gamma(y_k, i, s|s')$ are defined by

$$\begin{aligned}
\alpha_k(s) &\stackrel{\triangle}{=} Pr\{S_k = s | Y_1^k = y_1^k\}, \\
\beta_k(s) &\stackrel{\triangle}{=} \frac{p\{Y_{k+1}^L = y_{k+1}^L | S_k = s\}}{p\{Y_{k+1}^L = y_{k+1}^L | Y_1^k = y_1^k\}}, \\
\gamma(y_k, i, s|s') &\stackrel{\triangle}{=} p\{Y_k = y_k, U_k = i, S_k = s | S_{k-1} = s'\}.
\end{aligned}$$

It is easy to show that $\alpha_k(s)$ and $\beta_k(s)$ can be computed via the following recursive relations:

$$\begin{aligned}
\alpha_k(s) &= \frac{\sum_{s'} \sum_{i=0}^1 \gamma(y_k, i, s|s') \alpha_{k-1}(s')}{\sum_s \sum_{s'} \sum_{i=0}^1 \gamma(y_k, i, s|s') \alpha_{k-1}(s')}, \\
\beta_k(s) &= \frac{\sum_{s'} \sum_{i=0}^1 \gamma(y_{k+1}, i, s'|s) \beta_{k+1}(s')}{\sum_s \sum_{s'} \sum_{i=0}^1 \gamma(y_{k+1}, i, s'|s) \alpha_k(s)}.
\end{aligned}$$

To solve the above forward and backward recursions, properly defined boundary conditions are needed; they are determined by the initial and terminal states of the encoder. Conventionally, the encoder starts and terminates at the all-zero state. However, as we are using very long sequences in this work, we leave the encoder unterminated since there is no noticeable performance loss in comparison with the terminated case. Suppose the encoder has memory size m, then the final state can be any one of the total 2^m possible states. Therefore, the boundary conditions are

$$\begin{aligned}\alpha_0(0) &= 1, \quad \alpha_0(s) \;=\; 0, \;\; \forall\, s \neq 0,\\ \beta_L(s) &= \frac{1}{2^m}, \;\forall\, s = 0, 1, \cdots, 2^m - 1.\end{aligned}$$

In iterative decoding, only $L_{ex}^{iid}(U_k)$ is passed on to the next constituent decoder. After being re-arranged into the appropriate order by the interleaver (or the de-interleaver), the extrinsic information generated by one constituent decoder becomes the a priori information for the other constituent decoder. The reliability of the LLR soft decision improves over the iterations, and the iterative decoding procedure may terminate when the improvement becomes negligible, or the number of iterations exceeds a pre-determined threshold.

B. First Constituent Decoder

To exploit the source memory, the BCJR algorithm employed in the Turbo code decoder has to be modified. The modified BCJR algorithm in this section is derived for systematic Turbo codes; however, the modifications can be extended to non-systematic Turbo codes [36].

Given that the L-tuple $Y_1^L = y_1^L$ is received at the channel output, we have the equation at the top of the next page, where the summation for s is over all 2^m possible states. Noting that conditioned on $U_k = i$ and $S_k = s$, the observation y_{k+1}^L does not depend on y_1^k, we have that

$$p(y_{k+1}^L | i, s, y_1^k) = p(y_{k+1}^L | i, s).$$

Now if we define

$$\begin{aligned}\alpha_k(i,s) &\triangleq Pr\{U_k = i, S_k = s | Y_1^k = y_1^k\}, &(3)\\ \beta_k(i,s) &\triangleq \frac{p(y_{k+1}^L | i, s)}{p(y_{k+1}^L | y_1^k)}, &(4)\end{aligned}$$

then the conditional probability becomes

$$Pr\{U_k = i | Y_1^L = y_1^L\} = \sum_s \alpha_k(i,s)\beta_k(i,s). \tag{5}$$

Note that now $\alpha_k(\cdot,\cdot)$ and $\beta_k(\cdot,\cdot)$ are functions of both the encoder state s and the input bit i. The exploitation of bitwise dependency is evident in the following recursions:

$$\alpha_k(i,s) = \frac{\sum_{i',s'} \gamma(i, s, y_k | i', s')\alpha_{k-1}(i', s')}{\sum_{i,s}\sum_{i',s'} \gamma(i, s, y_k | i', s')\alpha_{k-1}(i', s')}, \tag{6}$$

$$\beta_k(i,s) = \frac{\sum_{i',s'} \gamma(i', s', y_{k+1} | i, s)\beta_{k+1}(i', s')}{\sum_{i,s}\sum_{i',s'} \gamma(i', s', y_{k+1} | i, s)\alpha_k(i, s)}, \tag{7}$$

where (8), located at the top of the next page.

Combining the marginal distribution of the Markov source as given in (1), the boundary conditions for computing $\alpha_k(\cdot,\cdot)$ and $\beta_k(\cdot,\cdot)$ are now

$$\begin{aligned}\alpha_0(0,0) &= p_0, &\alpha_0(1,0) &= p_1, &(9)\\ \alpha_0(i,s) &= 0, &i &= 0,1, \;\; \forall\, s \neq 0, &(10)\\ \beta_L(0,s) &= \frac{p_0}{2^m}, &\beta_L(1,s) &= \frac{p_1}{2^m}, \;\; \forall\, s. &(11)\end{aligned}$$

Note that in the above boundary conditions, the source redundancy in the form of non-uniformity can be exploited when $p_0 \neq 1/2$.

For iterative decoding, the LLR needs to be decomposed. Using (5)-(8), we can decompose the decoder's LLR of U_k into two separate terms:

$$\Lambda_k^{(1)}(U_k) \triangleq \log\frac{Pr\{U_k = 1 | Y_1^L = y_1^L\}}{Pr\{U_k = 0 | Y_1^L = y_1^L\}} = L_{ch}(U_k) + L_{ex}^{(1)}(U_k), \tag{12}$$

where

$$L_{ch}(U_k) = \log\frac{p(y_k^s | 1)}{p(y_k^s | 0)} \tag{13}$$

and (14), located at the top of the next page.
If we compare with the decomposition of the LLR in the case of Turbo coding of uniform i.i.d. sources in (2) where

$$\Lambda_k^{iid}(U_k) = L_{ch}(U_k) + L_{ex}^{iid}(U_k) + L_{ap}^{iid}(U_k),$$

we observe that in (12), the extrinsic information term $L_{ex}^{(1)}(U_k)$ is in essence the combination of $L_{ex}^{iid}(U_k)$ and $L_{ap}^{iid}(U_k)$, and $L_{ap}^{iid}(U_k)$ is actually the extrinsic information generated from the second constituent decoder. However, an important principle of iterative decoding is that the estimation generated by a constituent decoder should not be fed back to itself, otherwise the noise corruption will be highly correlated [7]. Thus, the decomposition in (12) renders the first constituent decoder unable to "update" its a priori term by using the extrinsic information generated from the second constituent decoder in the same way as in the memoryless source case. Therefore, the extrinsic information term for the first constituent decoder must be further modified.

Soon after the debut of Turbo codes in 1993, Robertson [24] improved Berrou's version of the BCJR algorithm in several aspects. His algorithm (which is described in Section III-A) is less complex and uses a different method to exchange the extrinsic information between the two constituent decoders. However, we observe that our decomposition in (5) is more similar to Berrou's version than Robertson's, and that Berrou's method with the Gaussian assumption actually provides a suitable solution for the above decomposition problem.

By using Berrou's method, the input to the first decoder has now three components:

$$\mathbf{y}_k = (y_k^s, y_k^{1p}, y_k^{ex(2)}),$$

where $y_k^{ex(2)}$ is the extrinsic information term from the second constituent decoder, $L_{ex}^{(2)}(U_k)$, after de-interleaving. Using the Gaussian assumption on $y_k^{ex(2)}$ as in [7], [12], [13], we obtain that in (8), $\gamma(i, s, y_k | i', s')$ has an additional factor in its expression; the factor is described by the following density

$$p(y_k^{ex(2)} | i) = \frac{1}{\sqrt{2\pi\bar{\sigma}^2}} e^{-\frac{\left[y_k^{ex(2)} - (2i-1)\bar{M}\right]^2}{2\bar{\sigma}^2}}, \quad i = 0, 1,$$

$$\begin{aligned} Pr\{U_k = i|Y_1^L = y_1^L\} &= \sum_s Pr\{U_k = i, S_k = s|Y_1^L = y_1^L\} \\ &= \sum_s \frac{p\{U_k = i, S_k = s, Y_1^k = y_1^k, Y_{k+1}^L = y_{k+1}^L\}}{p(y_1^k, y_{k+1}^L)} \\ &= \sum_s \frac{p\{U_k = i, S_k = s, Y_1^k = y_1^k\} \cdot p(y_{k+1}^L|i, s, y_1^k)}{p(y_1^k) \cdot p(y_{k+1}^L|y_1^k)} \end{aligned}$$

$$\begin{aligned} \gamma(i, s, y_k|i', s') &\triangleq p\{U_k = i, S_k = s, Y_k = y_k|U_{k-1} = i', S_{k-1} = s'\} \\ &= p(y_k^s|i)p(y_k^p|i, s)Pr\{S_k = s|U_k = i, S_{k-1} = s'\}Pr\{U_k = i|U_{k-1} = i'\} \\ &\triangleq p(y_k^s|i) \cdot p(y_k^p|i, s) \cdot q(s|i, s') \cdot \pi_{i'i} \end{aligned} \tag{8}$$

$$L_{ex}^{(1)}(U_k) = \log \frac{\sum_s \sum_{i',s'} p(y_k^p|1, s)q(s|1, s')\alpha_{k-1}(i', s')\beta_k(1, s)\pi_{i'1}}{\sum_s \sum_{i',s'} p(y_k^p|0, s)q(s|0, s')\alpha_{k-1}(i', s')\beta_k(0, s)\pi_{i'0}} \tag{14}$$

where $\bar{M}$ and $\bar{\sigma}^2$ are the estimated mean and variance obtained via an on-line estimation as follows:

$$\begin{aligned} \bar{M} &= \frac{1}{L}\sum_{i=1}^{L}|y_i|, \\ \bar{\sigma}^2 &= \frac{1}{L-1}\sum_{i=1}^{L}(|y_i| - \bar{M})^2. \end{aligned}$$

Thus, in the forward-backward recursion, both $\alpha_k(i, s)$ and $\beta_k(i, s)$ are computed according to this modification.

Finally, in the decomposition of $\Lambda^{(1)}(U_k)$, due to interleaving, $y_k^{ex(2)}$ can be regarded as weakly correlated with y_k^s and y_k^{1p}; therefore, the LLR soft-output generated by the first decoder is:

$$\Lambda_k^{(1)} = L_{ch}(U_k) + L_{ex}^{(1)}(U_k) + L_{ap}^{(1)}(U_k),$$

where the a priori term becomes

$$L_{ap}^{(1)}(U_k) \triangleq \log \frac{p(y_k^{ex(2)}|1)}{p(y_k^{ex(2)}|0)} = \frac{2\bar{M}}{\bar{\sigma}^2} y_k^{ex(2)}.$$

At this point, we can use the extrinsic information $L_{ex}^{(1)}(U_k)$ as the a priori term for the second constituent decoder.

In [14], Fazel *et al.* consider designing a Turbo code for the 2400 bps MELP speech coder's output modeled by non-binary Markov chains. However, in [14], both $\alpha_k(\cdot)$ and $\beta_k(\cdot)$ have the same forms as for memoryless sources (see Section III-A), and only γ is changed to take account of the source Markovian property. Therefore, the source memory is not fully exploited by such modifications. Our approach is in the same spirit as in [15], [16], where Garcia-Frias *et al.* consider the issue of using Turbo codes for Hidden Markov sources (although in [15], [16], the modification of the extrinsic information in (14) is not explicitly addressed, to the best of our knowledge).

C. Second Constituent Decoder

As mentioned in [14], [15], due to interleaving, the Markovian property in the input sequence of the second constituent decoder is destroyed. This renders the second constituent decoder unable to adopt the same modifications as in Section III-B. Therefore for this decoder, we employ Robertson's BCJR algorithm for i.i.d. sources presented in Section III-A, with the exception that we alter the extrinsic information, as described below.

In [11], Cai *et al.* proposed a different modification in an attempt to exploit the source memory. In their scheme, the original BCJR algorithm is employed in both decoders; however, the extrinsic information probability term P_{ex} is updated by adopting a modified probability according to:

$$P'_{ex}(u_k) = \frac{1}{2}\left[P_{ex}(u_k) + \sum_{u_{k-1}} \pi_{u_{k-1}u_k} P_{ex}(u_{k-1})\right], \quad u_k = 0, 1.$$

Although this method is simple to implement, it suffers from the problem that the above modified probabilities may not sum up to 1 due to round-up errors. This may cause an error propagation in the decoder. Furthermore, the reliability of the correction term is unknown; therefore, putting equal weights on the original probability and the correction term is questionable. Thus, we propose the following modification to the extrinsic information generated from the second constituent decoder:

$$L_{ex}^{(2)}(U_k) = c_1 L_{ex}^{iid}(U_k) + c_2 \log\left[\frac{\sum_i \pi_{i1} Pr\{\hat{U}_{k-1} = i\}}{\sum_i \pi_{i0} Pr\{\hat{U}_{k-1} = i\}}\right], \tag{15}$$

where $\{U_k\}$ refers to the original uninterleaved source sequence, $L_{ex}^{iid}(\cdot)$ is the extrinsic information defined in Section III-A and $c_1, c_2 \in [0, 1]$ with $c_1 + c_2 = 1$. The values of c_1 and c_2 are empirically chosen to yield the best possible improvement. $Pr\{\hat{U}_{k-1} = i\}$ (with i=0, 1) is directly obtained from the extrinsic information described by

$$Pr\{\hat{U}_{k-1} = 0\} = 1 - Pr\{\hat{U}_{k-1} = 1\} = \frac{1}{1 + e^{L_{ex}(U_{k-1})}},$$

for $k \geq 2$ with

$$Pr\{\hat{U}_1 = 0\} = p_0 = 1 - Pr\{\hat{U}_1 = 1\}.$$

IV. Encoder Structure Optimization

When the Markov source distribution is biased, the input sequence to the encoder would contain long segments of 1s or 0s. To see how the encoder structure may affect the performance, we first consider sources whose transitional probability distribution is extremely biased. For example, a "1" followed by a very long segment of 0s. Suppose we are using Berrou's $(37, 21)$ encoder [8], which starts at the all-zero state 0000, where each digit represents the content of each shift register. The encoder would remain in this state until the first "1" arrives, which would cause a transition to state 1000. The following state transitions would be $1100 \rightarrow 0110 \rightarrow 0011 \rightarrow 0001 \rightarrow 1000$, and from then on the state transition would be confined in a short cycle consisting of the above 5 states while all the other states remain dormant. Similarly, a long segment of 1s will also drive the encoder circulating among five states. This clearly indicates that the encoder memory is not fully exploited. In fact, it is equivalent to an encoder with smaller memory size. For such extremely biased sources, it is easy to see that an encoder with feedback 31 can excite all 16 states, and thus is intuitively a better choice than one with feedback 37. As the source gets less biased, the advantage of adopting an encoder with longer cycles becomes less obvious. For non-uniform i.i.d. sources, we showed in [36] that certain feed-forward polynomials are more desirable than others for a chosen feedback polynomial; however, the proof for the Markov source case is not obvious. Nevertheless, it is evident that encoder optimization plays an important role in determining the system performance. Since we employ different decoding methods in the constituent decoders, they exploit the source redundancy with different degrees. It is thus judicious to allow different encoding structures in our search for the best constituent encoders.

Since the original Turbo codes employ 16-state recursive systematic convolutional (RSC) constituent encoders, we focus on the design of 16-state encoders ($m = 4$). We denote the coefficients of the feedback and feed-forward polynomials of a 16-state RSC encoder in binary form as $\{f_0, f_1, f_2, f_3, f_4\}$ and $\{g_0, g_1, g_2, g_3, g_4\}$, respectively, where $f_i, g_j = 0$ or 1, $i, j = 0, 1, \cdots, 4$. An exhaustive search for the best encoding coefficients is however computationally expensive. The search is indeed more taxing than in the case of symmetric (i.e., identical) encoders [35], since we allow different (or asymmetric) constituent encoders. We then search for sub-optimal encoder structures. Our search, with the restriction that both constituent encoders have f_0=f_4=$g_0 = 1$, is performed as follows.

1) Fix the second constituent encoder as, for example, (31, 23), find (by simulation) the best feed-forward and feedback polynomials of the first constituent encoder via the iterative steps described in [35].

2) Fix the best structure for the first constituent encoder as found in step 1), find the best feed-forward and feedback polynomials of the second constituent encoder.

The initial structure of the second constituent encoder in step 1) is selected according to our results obtained in [35]. For example, for a general asymmetric Markov source, if the marginal distribution $p_0 \geq 0.9$, then the second constituent encoder is initially chosen as (31, 23); if $0.6 \leq p_0 \leq 0.8$, we may choose it as (35, 23). If the source is symmetric, since there is no redundancy in the form of non-uniformity that can be exploited by the second constituent decoder, we may initially choose the second constituent encoder in step 1) as Berrou's (37, 21) code, which offers very good performance for uniform i.i.d. sources. In our search, we used a sequence length $L = 65536$ bits.[2]

V. Shannon Limit

Shannon's *Lossy Information Transmission Theorem* [23], [27] states that for a given memoryless source and a given memoryless channel with capacity C, the source can be transmitted, for sufficiently large source blocklengths, via a source-channel code over the channel at a transmission rate of r source symbols/channel symbol and reproduced at the receiver end within an end-to-end distortion given by D if the following holds

$$r \cdot R(D) < C, \tag{16}$$

where $R(D)$ is the source rate-distortion function. As shown in [27], the above result can be achieved by concatenating (in tandem) separately designed source and channel codes at an overall rate of $r = R_c/R_s$, where R_c and R_s are the source and channel coding rates, respectively. The above theorem is also applicable when the source and the channel additive noise are stationary ergodic [6]. Under the Hamming distortion measure, the distortion $D = P_e$, the BER. Note that the capacity C for the considered BPSK-modulated channels with average symbol energy E_s is a function of E_b/N_0, where $E_b = E_s/r$ is the average energy per source bit. Thus, solving (16) by assuming equality yields the optimum value of E_b/N_0 to guarantee a BER of P_e; this value is called the *Shannon limit*.

In general, for asymmetric Markov sources, there are no closed-form expressions for $R(D)$. In this case, Blahut's algorithm [9] can be employed to compute the normalized nth order source rate-distortion function $R_n(D)$, which is an upper bound to $R(D)$. It can also be used to compute the Wyner and Ziv lower bound [33] to $R(D)$ which was later tightened by Berger [6]. In other words, the following two bounds for $R(D)$ can be calculated:

$$R_n(D) \geq R(D) \geq R_n(D) - \left[\frac{1}{n}H(U^n) - H(U_\infty)\right], \tag{17}$$

where $H(U^n) = H(U_1, U_2, \cdots, U_n)$ is the source block entropy and $H(U_\infty)$ is the source entropy rate. Remark that the above bounds hold for general stationary sources with memory. For a stationary Markov source, (17) reduces to

$$R_n(D) \geq R(D) \geq R_n(D) - \frac{1}{n}\left[H(U_1) - H(U_2|U_1)\right], \tag{18}$$

where $H(U_1)$ is the source entropy and $H(U_2|U_1)$ is the source conditional entropy of U_2 given U_1. Note that the above two bounds are asymptotically tight as $n \rightarrow \infty$; however, the

[2] We observed that when the performance of an encoder is better than another encoder for a given sequence length, then this behaviour also holds for longer sequence lengths. We used a shorter sequence length than in the simulations section to speed up the optimization procedure.

complexity for computing $R_n(D)$ grows exponentially in n. Therefore, for a desired BER level and a given overall rate r in source symbol/channel symbol, we may substitute $R(D)$ with its upper or lower bound in (16), and hence obtain an upper or lower bound on the corresponding Shannon limit.

For the special case of binary *symmetric* Markov sources with transition probability

$$Pr\{U_{k+1} = j | U_k = i\} = q, \quad \text{for } i \neq j, \tag{19}$$

where it is assumed that $q > 1/2$ without loss of generality, Gray proved that [18]

$$R(D) = h_b(q) - h_b(D), \quad \text{if } 0 \leq D \leq D_c, \tag{20}$$

where $h_b(\cdot)$ is the binary entropy function and D_c is the *critical distortion* defined by

$$D_c = \frac{1}{2}\left(1 - \sqrt{1 - \frac{(1-q)^2}{q^2}}\right). \tag{21}$$

Therefore, when the desired distortion D satisfies $D \leq D_c$, $R(D)$ admits a closed-form expression identical to that of a non-uniform i.i.d. source with $Pr\{U_k = 0\} = q$. Under the Hamming distortion measure, we obtain that, for the values of q that we considered, Gray's critical distortion in (21) is indeed greater than our target BER (which is 10^{-5}); hence, the Shannon limit can be determined exactly using (16) and (20).

VI. Numerical Results and Discussion

We present simulation results of our JSC systematic Turbo codes for the transmission of binary Markov sources over BPSK-modulated AWGN and Rayleigh fading channels. All simulated Turbo codes have 16-state constituent encoders and use the same pseudo-random interleaver introduced in [7]. To keep a tolerable computational complexity, the sequence length is limited to $L = 512 \times 512 = 262144$ and at least 200 blocks are simulated; this would guarantee a reliable BER estimation at the 10^{-5} level with 524 errors.

In our modification of the extrinsic information generated from the second constituent decoder, the weights c_1 and c_2 in (15) between the original extrinsic information term and the correction term are selected by simulations. Given that $c_1 + c_2 = 1$, we tried the following values for c_1: 0, 0.1, 0.2, $\cdots$, 0.9, 1.0. The resulting best choice is to set c_1=0.8 and c_2=0.2, and this choice is robust for the (q_0,q_1) pairs that we have tested.

A. Symmetric Markov Sources

We first present the results for symmetric Markov sources. The number of iterations in the Turbo code decoder is 20. Simulations are performed for rate $R_c = 1/3$ with q=0.7, 0.8 and 0.9. The best encoder we found for q=0.9 has the first constituent encoder as (31, 23), and the second as (35, 23). For q=0.8 and 0.7, assigning (35, 23) for both constituent encoders turns out to be the best choice.

Fig. 2 shows the performance of our systematic Turbo codes designed for transmitting binary symmetric Markov sources over AWGN channels. The Shannon limit curves, computed via (20) and (16), are also plotted for different target BERs. To illustrate the gains achieved due to the exploitation of the source redundancy (in both encoder and decoder), we include the performance curve of the standard Berrou (37,21) Turbo code which ignores the source statistics. The decoder of the Berrou code hence assumes that the source is uniform i.i.d. and employs Robertson's decoding algorithm of Section III-A. In Fig. 2, we refer to such system using the label "w/o JSC Dec." which stands for "without joint source-channel decoding." We clearly observe from the plots that at a BER level of 10^{-5}, when $q = 0.7$, our system offers a gain of 0.45 dB over the "w/o JSC Dec." scheme, yielding a gap of 0.73 dB from the Shannon limit; when q=0.8 and 0.9, the gains become 1.29 dB and 3.03 dB, respectively, which are 0.94 dB and 1.36 dB away from the Shannon limit.

Fig. 3 illustrates similar results for BPSK-modulated Rayleigh fading channels with known channel state information. When $q = 0.7$, the gain over the "w/o JSC Dec." scheme due to exploiting the source memory in our design is 0.52 dB, which is 0.87 dB away from the Shannon limit. When q=0.8, the gain increases to 1.55 dB, bringing the performance at a distance of 1.08 dB away from the Shannon limit. For $q = 0.9$, the gain reaches 3.57 dB; resulting an gap of 1.45 dB to the Shannon limit. The exact Shannon limit values and the performance gaps vis-a-vis the Shannon limit for a BER target

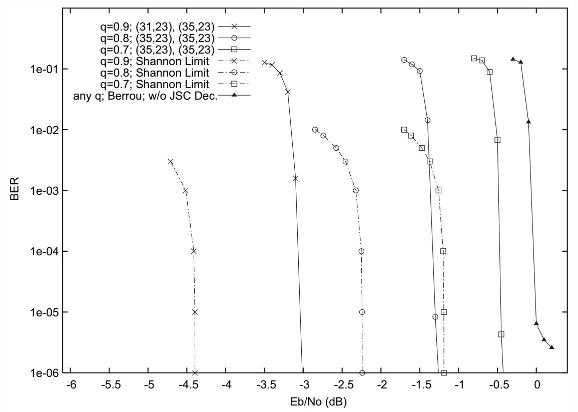

Fig. 2. Turbo code performance and Shannon limit for binary symmetric Markov sources, R_c=1/3, L=262144, AWGN channel.

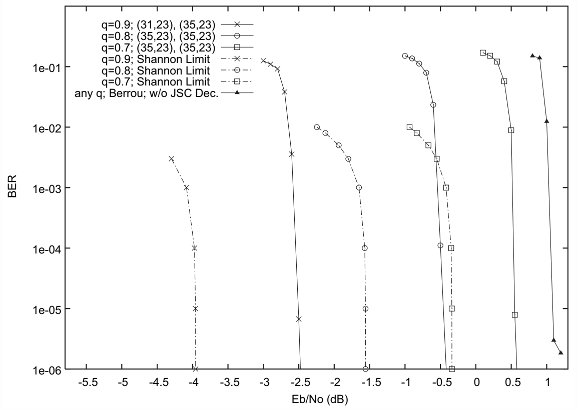

Fig. 3. Turbo code performance and Shannon limit for binary symmetric Markov sources, R_c=1/3, L=262144, Rayleigh fading channel with known channel state information.

TABLE I

SHANNON LIMIT (SL) VALUES AND GAPS IN E_b/N_0 AT BER=10^{-5} LEVEL (IN dB), TURBO CODES FOR BINARY SYMMETRIC MARKOV SOURCE, R_c=1/3, L=262144, AWGN AND RAYLEIGH FADING CHANNEL. THE EXACT SHANNON LIMIT VALUE IS COMPUTED USING (20) AND (16).

Source Distribution	Turbo Codes	AWGN		Rayleigh	
		SL value	SL gap	SL value	SL gap
$q = 0.7$	(35,23), (35,23)	-1.19	0.73	-0.34	0.87
$q = 0.8$	(35,23), (35,23)	-2.24	0.94	-1.56	1.08
$q = 0.9$	(31,23), (35,23)	-4.40	1.36	-3.96	1.45

TABLE II

SHANNON LIMIT LOWER BOUND (SL-LB), UPPER BOUND (SL-UB), AVERAGE VALUE (SL-AVG) AND GAP IN E_b/N_0 AT BER=10^{-5} LEVEL (IN dB) OF TURBO CODES FOR BINARY ASYMMETRIC MARKOV SOURCES OVER AWGN CHANNELS WITH R_c=1/3, L=262144 AND 30 DECODING ITERATIONS. THE SHANNON LIMIT LB AND UB VALUES ARE OBTAINED FROM (18) AND (16) BY CALCULATING THE SOURCE n'TH RATE-DISTORTION FUNCTION $R_n(D)$ VIA BLAHUT'S ALGORITHM USING BLOCKLENGTH $n = 12$ FOR ALL SOURCES (EXCEPT THE FIRST AND FOURTH SOURCE FOR WHICH $n = 10$ IS USED).

Source Distribution	Turbo Codes	SL-LB	SL-UB	SL-avg	SL gap
$q_0 = 0.7$, $q_1 = 0.65$, $p_0 = 0.538$	(37,21), (35,23)	-1.04	-0.99	-1.015	0.64
	(35,23), (35,23)	-1.04	-0.99	-1.015	0.69
$q_0 = 0.8$, $q_1 = 0.7$, $p_0 = 0.6$	(35,23), (35,23)	-1.80	-1.70	-1.75	0.83
$q_0 = 0.9$, $q_1 = 0.8$, $p_0 = 0.667$	(35,23), (35,23)	-3.59	-3.32	-3.455	1.14
$q_0 = 0.7$, $q_1 = 0.25$, $p_0 = 0.714$	(37,21), (35,23)	-1.31	-1.31	-1.31	0.89
	(35,23), (35,23)	-1.31	-1.31	-1.31	0.92
$q_0 = 0.9$, $q_1 = 0.7$, $p_0 = 0.75$	(35,23), (35,23)	-3.42	-3.25	-3.335	1.31
	(31,23), (35,23)	-3.42	-3.25	-3.335	1.36
$q_0 = 0.95$, $q_1 = 0.8$, $p_0 = 0.8$	(31,23), (35,23)	-5.49	-5.13	-5.31	1.67

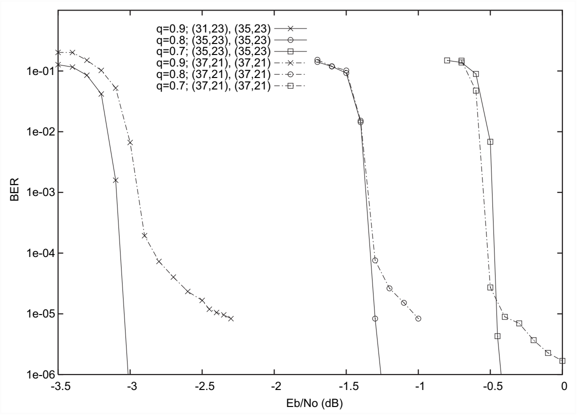

Fig. 4. Comparison with Berrou's (37, 21) code for binary symmetric Markov sources, R_c=1/3, L=262144, AWGN channel.

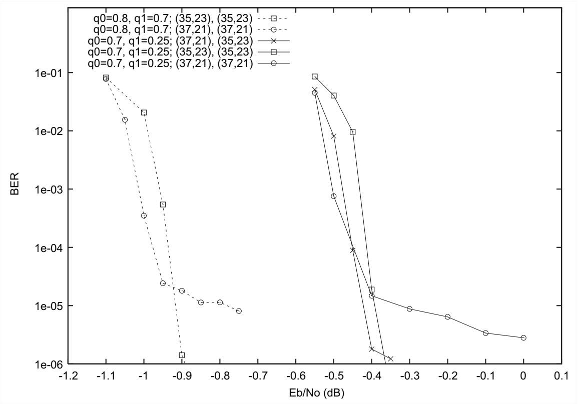

Fig. 5. Turbo code performance for binary asymmetric Markov sources, R_c=1/3, L=262144, AWGN channel.

of 10^{-5} are summarized in Table I.

To demonstrate the gains achieved by encoder optimization, we compare our system with Berrou's (37, 21) code, which also exploits the source statistics at the decoder in exactly the same way. The results are shown in Fig. 4 for AWGN channels (a similar behavior is also observed for Rayleigh fading channels). We remark that as the source transitional probability increases, the gain due to encoder optimization becomes more significant yielding substantially lower error floors. For example, at the 10^{-5} BER level, the gain is around 0.65 dB when $q = 0.9$. We have also separately obtained the performance of our systems studied thus far for shorter sequence lengths ($L = 32 \times 32 = 1024$ and $L = 128 \times 128 = 16384$), since in some practical situations, delay may be limited. Our results indicate a similar performance advantage of our JSC coding system even for relatively small values of L (e.g., $L = 1024$).

B. Asymmetric Markov Sources

We next present simulation results for asymmetric Markov sources. The sequence length is again $L = 262144$ and the rate is $R_c = 1/3$. The number of decoding iterations is 30. By varying the values of (q_0, q_1), we generate a series of Markov sources which have different amounts of redundancy in the form of memory and non-uniformity; in total six asymmetric sources are considered. The results are summarized in Table II.

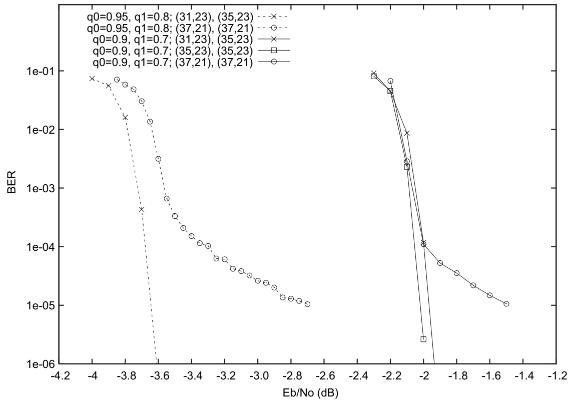

Fig. 6. Turbo code performance for binary asymmetric Markov sources, R_c=1/3, L=262144, AWGN channel.

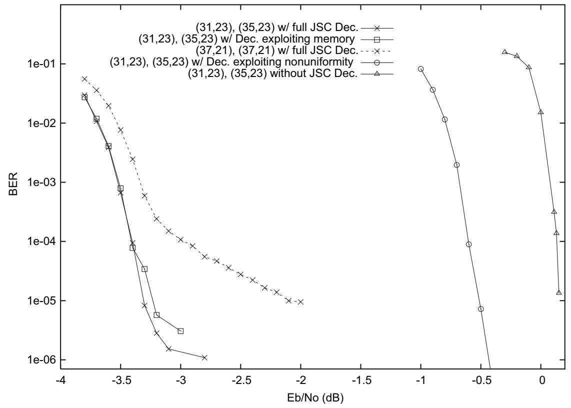

Fig. 7. Turbo code performance due to optimizing the encoder and exploiting the source memory/non-uniformity to various degrees for binary asymmetric Markov source with transition distribution given by $q_0 = 0.95$ and $q_1 = 0.8$ ($p_0 = 0.8$), R_c=1/3, $L = 16384$, AWGN channel.

In Figs. 5 and 6, performance curves of our optimized Turbo codes are provided for four sources. We also illustrate the benefits of our encoder optimization by plotting the performance of the Berrou code under identical decoding. We clearly note that our codes offer considerably lower error floors than the Berrou code.

The performance of our codes vis-a-vis the Shannon limit is also assessed in Table II for all six asymmetric sources. Since the Shannon limit cannot be exactly computed for such sources, we provide the lower and upper bounds to the Shannon limit using (18) and (16). We then use the average of the lower and upper bounds as a benchmark to compute the performance gap to the Shannon limit. As shown in Table II, the gap varies from 0.64 dB to 1.67 dB. More conservatively, if we instead use the Shannon limit lower bound as a benchmark, then the performance gap to the Shannon limit varies from 0.665 dB to 1.85 dB.

In Fig. 7, we present the effects of optimizing the encoder and exploiting the source memory and non-uniformity to various degrees at the decoder on the Turbo code performance for the transmission of the binary asymmetric Markov source with $q_0 = 0.95$ and $q_1 = 0.8$ over AWGN channels. A sequence length of $L = 16384$ bits is used with at least 3200 simulated blocks; 30 iterations are used at the decoder. Note that the Markov source has substantial asymmetry as its marginal distribution is $p_0 = 0.8$. As shown in Table II, the best encoding structure for this source is (31,23) for the first constituent encoder and (35,23) for the second constituent encoder. The performance of five systems is plotted in Fig. 7:

(i) Our JSC system using the best (31,23), (35,23) encoder with full JSC decoding as described in Sections III-B and III-C.
(ii) Our JSC system using the (31,23), (35,23) encoder and JSC decoding as in Sections III-B and III-C with the exception that it assumes that the source marginal distribution is uniform in the boundary conditions (9) and (11) for $\alpha(\cdot,\cdot)$ and $\beta(\cdot,\cdot)$ of the first constituent decoder. This decoder hence fully exploits the source memory, and it exploits the source asymmetry partially as it depends on the received systematic stream $\{Y_k^s\}$.
(iii) An unoptimized encoder system using the Berrou (37,21) code with full JSC decoding.
(iv) A system using the (31,23), (35,23) encoder and a decoder that only exploits the source asymmetry. The decoder assumes that the source is non-uniform i.i.d. and is designed as in [35].
(v) A system using the (31,23), (35,23) encoder without JSC decoding. In other words, the decoder assumes that the source is uniform and i.i.d. and employs Robertson's method [24].

We note from the figure that our JSC system (i) offers the best performance. Its gain over system (ii) though is minor as the only difference between the two is that system (ii) does not exploit the source non-uniformity in (9) and (11). The gain of our system over system (iii) due to encoder optimization is significant: for a BER of 10^{-5} the performance gain is more than 1 dB. Furthermore, our system outperforms system (iv) by more than 3.5 dB for the same BER level. This large gain indicates that exploiting the source memory is considerably more important than exploiting the source asymmetry for this source. Finally, an additional 0.7 dB is gained (for a total gain of more than 4.2 dB) over system (iv) which exploits no source redundancy at the decoder.

In spite of its good performance, our JSC system is suboptimal for both symmetric and asymmetric Markov sources, particularly for q_0, $q_1 \geq 0.9$. As remarked earlier, the interleaving prevents the second decoder from systematically exploiting the source memory. By modifying the extrinsic information as shown in (15), an improvement is achieved; but clearly this method does not fully exploit the source dependencies. Furthermore, for asymmetric Markov sources, exploiting the source non-uniformity via non-systematic Turbo codes [36] may yield further improvements, as it will reduce the mismatch (due to the systematic structure of the Turbo encoder) between the biased distribution of the systematic bitstream at the encoder output and the uniform input distribution needed to achieve the capacity of the binary modulated AWGN/Rayleigh channel (cf. [26], [36]). Another promising method for exploiting the source non-uniformity, particularly if the source is strongly asymmetric, is to use non-binary modulation with unequal energy allocation as recently investigated in [10]. Hence, the design of a more sophisticated Turbo coding system, which can perform even closer to the Shannon limit for Markov sources with very biased transition

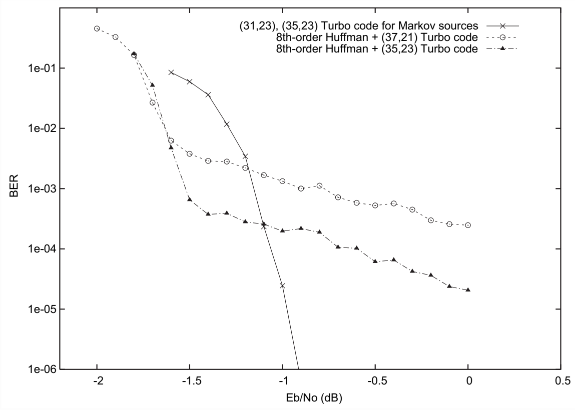

Fig. 8. Performance comparison of the JSC Turbo coding system (with $R_c = 1/2$) with that of tandem coding schemes (using Huffman coding with $R_s = 2/3$ and standard Turbo coding with $R_c = 1/3$) for binary symmetric Markov sources, q=0.848315, L=12000, AWGN channel.

distribution (e.g., q_0, $q_1 \geq 0.9$), remains an interesting and challenging problem.

VII. Comparison with Tandem Coding Schemes

In this section, we compare our JSC Turbo codes with two tandem coding schemes for the transmission of a symmetric Markov source with transition probability q. As in [36], the tandem schemes consist of a Huffman code followed by a standard symmetric Turbo code with constituent encoders (37, 21) or (35, 23), where the (35, 23) Turbo code offers a lower error-floor performance than that of its (37, 21) peer at the price of a slight performance loss in the water-fall region.

The comparison is made at the same overall rate $r = R_c/R_s = 1/2$ source symbol/channel symbol. Our JSC coding system has $R_c = 1/2$ and $R_s = 1$ (no source coding); the tandem scheme, however, has $R_c = 1/3$, therefore it needs to have $R_s = 2/3$. This can be achieved by numerically choosing the value of q such that the Huffman code produces an output sequence with average codeword length close to 2/3. Due to the source memory, a Markov source is harder to compress than a non-uniform i.i.d. source. In fact, using a 4^{th}-order Huffman code, to guarantee $R_s = 2/3$, we need to have q=0.881685. However, at this value of q, the entropy rate of the symmetric Markov source is 0.524. To achieve a better compression efficiency, we adopt an 8^{th}-order Huffman code, which can achieve R_s=0.666667 at q=0.848315, while the source entropy rate is 0.614.

The sequence length is L=12000, and at least 60000 blocks are simulated to produce a reliable average performance in the error-floor region. An S-random interleaver with S=10 is adopted replacing the pseudo-random interleaver. The number of iterations in the Turbo decoder is 20. The detailed implementation procedure is similar to that described in [36].

Fig. 8 shows the performance comparison of our JSC coding system with the two tandem schemes over AWGN channels. We observe that although initially the tandem schemes offer better water-fall performance, they are quickly outmatched by our system due to their high BER performance from medium to high SNRs. By using the (35, 23) code in the constituent encoders, the error-floor performance of the second tandem scheme is improved over the Berrou code based tandem scheme (the floor is lowered from the 10^{-3} to the 10^{-4} BER level), at the cost of a slight water-fall degradation. However, both tandem schemes have a high error floor (with respect to our system) due to the inevitable error-propagation incurred by the Huffman decoder. The use of a fixed-length source code instead of the Huffman code (or other entropy codes) would still not yield a satisfactory overall performance for the tandem schemes, since standard fixed-length source codes may not on their own achieve good compression (although they would remedy the error propagation problem). Unlike the tandem systems, our JSC coding system enjoys lower complexity (since no source encoding and decoding are performed), while offering a superior performance for BERs less than 10^{-4}. Note that the tandem system performance can be improved if we replace its source decoder with a soft-input source decoder (e.g., [32]), although at a price of increased complexity. Furthermore, alternative JSC coding systems that employ variable-length source encoding followed by Turbo encoding and jointly designed soft-input variable-length source decoding and Turbo decoding with possible iterations between both decoders (e.g., [5], [19]), may be compared in a future study to our JSC coding system; however, our system is significantly simpler in terms of complexity. On the other hand, the tandem scheme may have less delay since in comparison with the JSC system, the interleaver size in its Turbo coding part is shorter.

VIII. Summary

In this work, we investigate the design of JSC Turbo codes for binary stationary ergodic Markov sources. In the first constituent decoder, the decoding algorithm is modified to exploit the source redundancy in the form of memory and non-uniformity, where Berrou's original decoding method based on the Gaussian assumption for the extrinsic information provides a suitable solution for the LLR decomposition problem. Due to interleaving, the second constituent decoder is unable to adopt the same modifications. However, its extrinsic information is modified via a weighted correction term that depends on the Markovian source statistics. The Turbo encoder structure is also optimized in accordance with the source statistics. Substantial gains (from 0.45 up to 3.57 dB) are demonstrated by our system over the original Turbo codes, and its performance gap vis-a-vis the Shannon limit ranges from 0.73 to 1.45 dB for symmetric Markov sources, and from approximately 0.64 to 1.67 dB for asymmetric Markov sources. Finally, our system is compared with two tandem coding schemes, which employ 8^{th}-order Huffman coding (to achieve near-optimal data compression) and regular rate-1/3 Turbo coding. Our system is shown to provide a robust and substantially better performance (for BERs less than 10^{-4}) while benefiting from a lower overall complexity.

IX. Acknowledgment

The authors are grateful to the editor and the anonymous reviewers for their comments which were instrumental in significantly improving the paper.

REFERENCES

[1] M. Adrat, J.-M. Picard, and P. Vary, "Softbit-source decoding based on the turbo-principle," in *Proc. Veh. Tech. Conf.*, Oct. 2001, vol. 4, pp. 2252–2256.

[2] F. Alajaji, N. Phamdo, N. Farvardin, and T. Fuja, "Detection of binary Markov sources over channels with additive Markov noise," *IEEE Trans. Inform. Theory,* vol. 42, pp. 230–239, Jan. 1996.

[3] F. Alajaji, N. Phamdo, and T. Fuja, "Channel codes that exploits the residual redundancy in CELP-encoded speech," *IEEE Trans. Speech and Audio Processing*, vol. 4, pp. 325–336, Sep. 1996.

[4] L. R. Bahl, J. Cocke, F. Jelinek, and J. Raviv, "Optimal decoding of linear codes for minimizing symbol error rate," *IEEE Trans. Inform. Theory,* vol. 20, pp. 248-287, Mar. 1974.

[5] R. Bauer and J. Hagenauer, "On variable length codes for iterative source/channel decoding," in *Proc. Data Compression Conf.*, Mar. 2001, pp. 273–282.

[6] T. Berger, *Rate Distortion Theory: A Mathematical Basis for Data Compression*. Englewood Cliffs, NJ: Prentice Hall, 1971.

[7] C. Berrou, A. Glavieux, and P. Thitimajshima, "Near Shannon limit error-correcting coding and decoding: Turbo-codes(1)," in *Proc. IEEE Int. Conf. Commun.*, May 1993, pp. 1064–1070.

[8] C. Berrou and A. Glavieux, "Near optimum error correcting coding and decoding: Turbo-codes," *IEEE Trans. Commun.*, vol. 44, pp. 1261–1271, Oct. 1996.

[9] R. E. Blahut, *Principles and Practice of Information Theory*. Reading, MA: Addison Wesley, 1988.

[10] F. Cabarcas, R. D. Souza, and J. Garcia-Frias, "Source-controlled turbo coding of nonuniform memoryless sources based on unequal energy allocation," in *Proc. Int. Symp. Inform. Theory,* June-July 2004, p.166.

[11] Z. Cai, K. R. Subramanian, and L. Zhang, "Source-controlled channel decoding using nonbinary turbo codes," *IEEE Electron. Lett.*, vol. 37, no. 1, pp. 39–40, Jan. 2001.

[12] G. Colavolpe, G. Ferrari, and R. Raheli, "Extrinsic information in Turbo decoding: a unified view," in *Proc. Globecom*, Dec. 1999, pp. 505–509.

[13] H. El Gamal and A. R. Hammons, Jr., "Analyzing the turbo decoder using the Gaussian approximation," *IEEE Trans. Inform. Theory*, vol. 47, pp. 671–686, Feb. 2001.

[14] T. Fazel and T. Fuja, "Robust transmission of MELP-compressed speech: An illustrative example of joint source-channel decoding," *IEEE Trans. Commun.*, vol. 51, pp. 973–982, June 2003.

[15] J. Garcia-Frias and J. D. Villasenor, "Combining hidden Markov source models and parallel concatenated codes," *IEEE Commun. Lett.,* vol. 1, pp. 111–113, July 1997.

[16] J. Garcia-Frias and J. D. Villasenor, "Joint turbo decoding and estimation of hidden Markov sources," *IEEE J. Select. Areas Commun.*, vol. 19, pp. 1671–1679, Sep. 2001.

[17] N. Görtz, "On the iterative approximation of optimal joint source-channel decoding," *IEEE J. Select. Areas Commun.*, vol. 19, no. 9, pp. 1662–1670, Sep. 2001.

[18] R. M. Gray, "Information rates for autoregressive processes," *IEEE Trans. Inform. Theory*, vol. 16, pp. 412-421, July 1970.

[19] L. Guivarch, J.-C. Carlach, and P. Siohan, "Joint source-channel soft decoding of Huffman codes with Turbo-codes," in *Proc. Data Compression Conf.*, Mar. 2000, pp. 83–92.

[20] J. Hagenauer and N. Görtz, "The turbo principle in joint source-channel coding," in *Proc. IEEE Inform. Theory Workshop*, Mar.–Apr. 2003, pp. 275–278.

[21] E. K. Hall and S. G. Wilson, "Design and analysis of Turbo codes on Rayleigh fading channels," *IEEE J. Selec. Areas Commun.*, vol. 16, pp. 160–174, Feb. 1998.

[22] J. Kroll and N. Phamdo, "Source-channel optimized trellis codes for bitonal image transmission over AWGN channels," *IEEE Trans. Image Processing,* vol. 8, pp. 899–912, July 1999.

[23] R. J. McEliece, *The Theory of Information and Coding,* 2nd ed.. Cambridge, UK: Cambridge University Press, 2002.

[24] P. Robertson, "Illuminating the structure of parallel concatenated recursive systematic (turbo) codes," in *Proc. Globecom*, Nov. 1994, vol. 3, pp. 1298-1303.

[25] K. Sayood and J. C. Borkenhagen, "Use of residual redundancy in the design of joint source-channel codes," *IEEE Trans. Commun.*, vol. 39, pp. 838–846, June 1991.

[26] S. Shamai and S. Verdú, "The empirical distribution of good codes," *IEEE Trans. Inform. Theory*, vol. 43, pp. 836–846, May 1997.

[27] C. E. Shannon, "Coding theorems for a discrete source with a fidelity criterion," *IRE Nat. Conv. Rec.*, pt. 4, pp. 142–163, Mar. 1959.

[28] M. Skoglund, "Soft decoding for vector quantization over noisy channels with memory," *IEEE Trans. Inform. Theory,* vol. 45, no. 4, pp. 1293–1307, May 1999.

[29] K. P. Subbalakshmi and J. Vaisey, "On the joint source-channel decoding of variable-length encoded sources: The BSC case," *IEEE Trans. Commun.*, vol. 49, pp. 2052–2055, Dec. 2001.

[30] G. Takahara, F. Alajaji, N. C. Beaulieu, and H. Kuai, "Constellation mappings for two-dimensional signaling of non-uniform sources," *IEEE Trans. Commun.*, vol. 51, pp. 400–408, Mar. 2003.

[31] R. E. Van Dyck and D. Miller, "Transport of wireless video using separate, concatenated, and joint source-channel coding," in *Proc. IEEE*, Oct. 1999, vol. 87, pp. 1734–1750.

[32] J. Wen and J. Villasenor, "Soft-input soft-output decoding of variable length codes," *IEEE Trans. Commun.*, vol. 50, pp. 689–692, May 2002.

[33] A. Wyner and J. Ziv, "Bounds on the rate-distortion function for stationary sources with memory," *IEEE Trans. Inform. Theory*, vol. 17, pp. 508–513, Sep. 1971.

[34] W. Xu, J. Hagenauer, and J. Hollmann, "Joint source-channel decoding using the residual redundancy in compressed images," in *Proc. ICC*, June 1996, pp. 142–148.

[35] G.-C. Zhu and F. Alajaji, "Turbo codes for non-uniform memoryless sources over noisy channels," *IEEE Commun. Lett.*, vol. 6, pp. 64–66, Feb. 2002.

[36] G.-C. Zhu, F. Alajaji, J. Bajcsy, and P. Mitran, "Transmission of non-uniform memoryless sources via non-systematic Turbo codes," *IEEE Trans. Commun.*, vol. 52, pp. 1344–1354, Aug. 2004.

Guang-Chong Zhu (M'04) received the B.Sc. and M.Sc. degrees in computational mathematics from Shandong University, Shandong, China, in 1994 and 1997, respectively. He then received the M.Sc. degree in applied mathematics and the Ph.D. degree in mathematics and engineering from Queen's University, Kingston, ON, Canada, in 1998 and 2003, respectively. From 2003 to 2005, he was a Postdoctoral Fellow in the Department of Mathematics and Statistics at Queen's University, and then in the Department of Electrical and Computer Engineering at University of Toronto. In August 2005, He joined Lawrence Technological University as an assistant professor in the Department of Mathematics and Computer Science.

His current research interests include channel coding, joint source-channel coding, and information theory.

Fady Alajaji (S'90–M'94–SM'00) was born in Beirut, Lebanon, on May 1, 1966. He received the B.E. degree (with Distinction) from the American University of Beirut, Lebanon, and the M.Sc. and Ph.D. degrees from the University of Maryland, College Park, all in electrical engineering, in 1988, 1990 and 1994, respectively. He held a postdoctoral appointment in 1994 at the Institute for Systems Research, University of Maryland.

In 1995, he joined the Department of Mathematics and Statistics at Queen's University, Kingston, Ontario, where he is currently an Associate Professor of Mathematics and Engineering. Since 1997, he has also been cross-appointed in the Department of Electrical and Computer Engineering at the same university. He held research visits to the Department of Electrical and Computer Engineering, McGill University, Montreal, in the Fall of 2001, and to the Department of Mobile Communications, Technical University of Berlin, Germany, in the Winter of 2004. His research interests include information theory, joint source-channel coding, error control coding, data compression and digital communications.

Dr. Alajaji currently serves as Editor for Source and Source-Channel Coding for the IEEE TRANSACTIONS ON COMMUNICATIONS. He served as co-chair of the Technical Program Committee of the 2004 Biennial Symposium on Communications. In 1999, he co-chaired and organized the Sixth Canadian Workshop on Information Theory in Kingston, Ontario. He also served as chair of the Selection Committee for the 2000 Canadian Award in Telecommunications. In 2001, he received the Premier's Research Excellence Award (PREA) from the Province of Ontario.