PAPER

Iterative Source-Channel Decoding Using Symbol-Level Extrinsic Information

Chun-Feng WU[†], *Nonmember and* **Wen-Whei CHANG**[†a)], *Member*

SUMMARY Transmission of convolutionally encoded source-codec parameters over noisy channels can benefit from the turbo principle through iterative source-channel decoding. We first formulate a recursive implementation based on sectionalized code trellises for MAP symbol decoding of binary convolutional codes. Performance is further enhanced by the use of an interpolative softbit source decoder that takes into account the channel outputs within an interleaving block. Simulation results indicate that our proposed scheme allows to exchange between its constituent decoders the symbol-level extrinsic information and achieves high robustness against channel noises.

key words: *iterative source-channel decoding, sectionalized code trellis, extrinsic information*

1. Introduction

With the rapid development of wireless and multimedia communications, reliable transmission of speech and video signals over band-limited noisy channels are becoming more and more widespread. The basic strategy consists in using a source encoder to extract characteristic parameters of the source signals, which are then error protected by channel codes. According to Shannon's separation theorem [1], conventional source coding schemes are usually designed in a way that any knowledge about channel characteristics is disregarded. Furthermore, in the development of channel codes any possibly given source statistics is neglected. The source-channel separation theorem might hold under impractical requirements only demanding unlimited computational complexity and infinite coding delay. These requirements cannot be strictly met in a practical communication system with limited block lengths. As a consequence, on the one hand, residual redundancy remains in the set of source codec parameters after source coding and, on the other hand, residual errors may exist in the bit sequence after channel decoding. In order to provide a system of higher robustness against channel noise, several paths have been taken toward the joint design of source and channel coders. These methods include pseudo Gray coding [2], channel-optimized quantization [3], source-optimized channel codes [4], and more recently, exploiting the residual source redundancies [5]–[8].

In general, source encoders represent the source signal by a small set of characteristic parameters taken from a finite quantizer codebook, but due to block length constraints, the quantizer indexes will still exhibit considerable redundancies. Such residual redundancy appears on parameter-level, either in terms of a non-uniform distribution or in terms of time-correlation between consecutive indexes. The residual source redundancy can be used for enhancing channel decoding [5], [6] or for effective source decoding [7], [8]. In [7], the quantizer-channel tandem is described as a discrete hidden Markov model and the source decoding is formulated in the form of a sequence-based approximate minimum mean-squared error (SAMMSE) estimation problem. The softbit source decoding (SBSD) algorithm in [8] processes the soft channel outputs and combines them with source *a priori* information to estimate the decoded output. The estimation is carried out for the quantizer indexes rather than for single index-bits since the dependencies of indexes are stronger than the correlations of the index-bits. The error concealing capabilities of SBSD can be further improved if channel coding algorithms add artificial redundancy at the transmitter side. The entire system can be viewed as having the residual source correlations as implicit outer channel code that are serially concatenated with the inner explicit channel codes. Using this interpretation, the concept of extrinsic information from turbo decoding [9] can be adopted for iterative source-channel decoding (ISCD) [10]–[13]. In an ISCD scheme, the decoder is decomposed into two parts, which can be identified as the constituent decoders for the channel-code and source-code redundancies. According to the turbo-principle extracting the extrinsic information from one decoder and using it as additional *a priori* knowledge in the other one is expected to improve the reliability of the decoded signal step-by-step.

With respect to an implementation of ISCD it has to be emphasized that the major part of the iterative process runs on bit-level, but SBSD itself is realized on the parameter-level. This is justified by the fact that in many practical systems binary convolutional codes are utilized, so the soft-output channel decoding can be implemented efficiently by the BCJR algorithm [14]. It causes the problem that only bitwise source *a priori* knowledge can be exploited by the channel decoder, since the BCJR algorithm is derived based on a bit-level code trellis. The technique how to combine the source *a priori* knowledge on parameter-level with the extrinsic information of the channel decoder on bit-level is not widely common so far. In classical ISCD algorithms [10]–[13], it requires the conversion of the index-probabilities to bit-probabilities in each passing of the extrinsic information between the two constituent decoders. This processing step destroys the bit-correlations within an index, thus reducing

Manuscript received July 1, 2010.

[†]The authors are with National Chiao Tung University, Hsinchu, Taiwan, ROC.

a) E-mail: wwchang@cc.nctu.edu.tw

DOI: 10.1587/transcom.E93.B.3555

the effectiveness of iterative decoding. The drawback can be avoided if the extrinsic information due to the source-code and channel-code redundancies are not treated separately on different levels, but jointly on parameter-level. Therefore, we focus on symbol-based trellis decoding algorithms throughout this paper since in contrast to the classical BCJR algorithm [14] a symbol-by-symbol decoding approach is especially well suited to generate the reliability information on the quantizer indexes. The first step toward realization is to use quantizer indexes rather than single index-bits as the bases for the soft-output decoding of a convolutional channel code. This is used in conjunction with sectionalized code trellises to take advantage of the statistical *a priori* information on index-basis. Furthermore, the source decoder is different from the SBSD [8] in that besides the present and past channel outputs, some future channel outputs belonging to the same interleaving block are also evaluated to provide additional reliability gains. For the purpose of applicability, we derive a new formula that shows how the past and future channel outputs can be transformed into the extrinsic information utilizable for iterative decoding.

2. System Model

The transmission of continuous-valued, autocorrelated source samples is considered. Figure 1 shows our model of a transmission system. Suppose at time t, the input source sample v_t is quantized by the index u_t that, after bit mapping, is represented by a bit combination consisting of M bits. For notational convenience, the index u_t is regarded as an integer representing the decimal equivalent of a bitvector $(u_t(0), u_t(1), \ldots, u_t(M-1))$. The quantizer's reproduction level corresponding to the index $u_t = i$ is denoted by c_i, where $i \in \mathcal{I} = \{0, 1, \ldots, 2^M - 1\}$. We can generally assume that there is a certain amount of residual redundancy remaining in the index sequence due to delay and complexity constraints for the quantization stage. In the following, the time-correlations of the quantizer indexes are modelled by a first-order stationary Markov process with index-transition probabilities $P(u_t|u_{t-1})$. After source encoding a block of T bitvectors, written as $U_1^T = \{u_1, u_2, \ldots, u_T\}$ for brevity, are interleaved by an interleaver Φ on a symbol-by-symbol basis. Such an interleaver permutes the input data block in M-bit indexes, but does not change the order of the bits within each index. The interleaved index sequence is denoted by $X_1^T = \{x_1, x_2, \ldots, x_T\}$, where each index $x_t = \Phi(u_t)$ is associated with a bitvector $(x_t(0), x_t(1), \ldots, x_t(M-1))$. Afterwards the interleaved bitstream is encoded by a binary convolutional channel encoder prior to transmission. We considered a rate-1/2 systematic channel encoder whose outputs corresponding to each bit $x_t(m)$ are the systematic bit $x_t(m)$ and the parity bit $z_t(m)$. These outputs are modulated with a BPSK modulator and then transmitted over an AWGN channel. The code-bits are assumed to be bipolar, i.e., $x_t(m), z_t(m) \in \{-1, +1\}$. For notational convenience, the code-symbol sequence $Y_1^T = \{y_1, y_2, ..., y_T\}$ with $y_t = (x_t, z_t)$, will refer to the sequence of pairs of systematic and parity symbols. Let the possibly noisy received code sequence corresponding to $Y_1^T = (X_1^T, Z_1^T)$ be denoted by $\tilde{Y}_1^T = (\tilde{X}_1^T, \tilde{Z}_1^T)$.

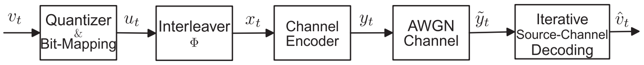

Fig. 1 Model of the transmision system.

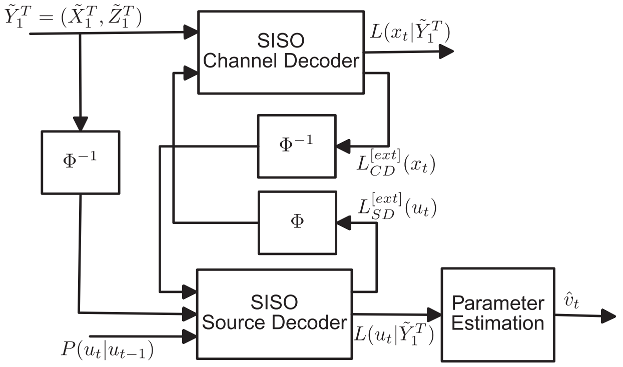

Fig. 2 Symbol-based iterative source-channel decoding.

At the receiver side, the decoder is designed to minimize the mean squared error in signal reconstruction when channel errors occur. For the concatenation of quantization and channel coding, the turbo-like evaluation of residual source redundancy and of artificial channel-code redundancy makes step-wise quality gains possible by iterative decoding. As shown in Fig. 2, the receiver consists of two constituent decoders with soft-inputs and soft-outputs (SISO). For each iteration, the channel decoder processes the received code sequence $\tilde{Y}_1^T$ and combines them with the source *a priori* information to compute the extrinsic information $L_{CD}^{[ext]}(x_t)$ on individual systematic symbol x_t. Goal of the source decoder is to jointly exploit the channel information and the source *a priori* information to compute the *a posteriori* probability (APP) for each of possibly transmitted quantizer index $u_t = i$, which is denoted by $P(u_t = i|\tilde{Y}_1^T)$. It also generates an extrinsic information $L_{SD}^{[ext]}(u_t)$ which becomes additional *a priori* knowledge of the channel decoder for the next iteration. Exchanging extrinsic information between two constituent decoders is iteratively repeated until the reliability gain becomes insignificant. After the last iteration, the APPs are combined with quantizer's reproduction levels to provide the signal estimates as follows:

$$\hat{v}_t = \sum_i P(u_t = i|\tilde{Y}_1^T)c_i \tag{1}$$

3. SISO Source Decoding

The determination rules of extrinsic information of SBSD has been derived in [12], but a slight modification is proposed which allows a delay of T samples in the decoding process. We have chosen the length T in compliance with the defined size of an interleaving block. If on the basis of a first-order Markov model time-correlation between consecutive indexes shall be utilized, then the entire history of received codewords $\tilde{Y}_1^t$ and possibly additionally given future codewords $\tilde{Y}_{t+1}^T$ have to be considered as well. To advance with this, we derive a forward-backward recursive algorithm that shows how the past and future received codewords can be transformed into extrinsic information utilizable in the iterative decoding process. The basic strategy is to jointly exploit the channel information, the source *a priori* information as well as the extrinsic information from the channel decoder. The APP for each of possibly transmitted index $u_t = i$, given the received code sequence $\tilde{Y}_1^T = (\tilde{X}_1^T, \tilde{Z}_1^T)$, is given by

$$P(u_t = i|\tilde{Y}_1^T) = P(u_t = i, \tilde{Y}_1^T)/P(\tilde{Y}_1^T). \tag{2}$$

Since the received sequence is de-interleaved and then processed by the source decoder, we have $P(u_t = i, \tilde{Y}_1^T) = P(u_t = i, \tilde{U}_1^T, \tilde{Z}_1^T)$, where $\tilde{U}_1^T = \Phi^{-1}(\tilde{X}_1^T)$. These probabilities can be further decomposed analogous to the SAMMSE algorithm [7] by using the Bayes theorem as

$$\begin{aligned} &P(u_t = i,, \tilde{U}_1^T, \tilde{Z}_1^T) \\ &\quad = P(u_t = i, \tilde{U}_1^T) P(\tilde{Z}_1^T | u_t = i, \tilde{U}_1^T) \\ &\quad = \alpha_t^u(i) \beta_t^u(i) P(\tilde{Z}_1^T | u_t = i, \tilde{U}_1^T) \end{aligned} \tag{3}$$

where $\alpha_t^u(i) = P(u_t = i, \tilde{U}_1^t)$ and $\beta_t^u(i) = P(\tilde{U}_{t+1}^T | u_t = i, \tilde{U}_1^t)$. Using the Markov property of the indexes and the memoryless assumption of the channel, the forward and backward recursions of the algorithm can be expressed as

$$\begin{aligned} \alpha_t^u(i) &= \sum_j P(u_t = i, u_{t-1} = j, \tilde{u}_t, \tilde{U}_1^{t-1}) \\ &= \sum_j P(\tilde{u}_t | u_t = i, u_{t-1} = j, \tilde{U}_1^{t-1}) \\ &\quad \cdot P(u_t = i | u_{t-1} = j, \tilde{U}_1^{t-1}) \cdot P(u_{t-1} = j, \tilde{U}_1^{t-1}) \\ &= P(\tilde{u}_t | u_t = i) \sum_j P(u_t = i | u_{t-1} = j) \alpha_{t-1}^u(j) \end{aligned} \tag{4}$$

and

$$\begin{aligned} \beta_t^u(i) &= \sum_j P(u_t = i, u_{t+1} = j, \tilde{U}_1^T)/P(u_t = i, \tilde{U}_1^t) \\ &= \sum_j P(\tilde{U}_{t+2}^T | u_{t+1} = j, \tilde{U}_1^{t+1}) \cdot P(\tilde{u}_{t+1} | u_{t+1} = j) \\ &\quad \cdot P(u_t = i, u_{t+1} = j, \tilde{U}_1^t)/P(u_t = i, \tilde{U}_1^t) \\ &= \sum_j P(\tilde{u}_{t+1} | u_{t+1} = j) P(u_{t+1} = j | u_t = i) \beta_{t+1}^u(j) \end{aligned} \tag{5}$$

The MAP algorithm is likely to be considered too complex for real-time implementation in a practical system. To avoid the number of complicated operations and also numerical representation problems, realizations of the MAP algorithm in the logarithmic domain have been proposed in [15], [16]. Goal of the classical log-MAP algorithm is to provide us with the logarithm of the ratio of the APP of each information bit being 1 to the APP of it being 0. In this work, however, the use of a symbol-based ISCD scheme implies that the reliability information should be defined for the quantizer indexes in form of index APPs. Recognizing this, we define the reliability of each nonzero index $u_t = i$, $i = 1, 2, \ldots, 2^M - 1$, with respect to $u_t = 0$, by considering log-likelihood ratio (LLR) of the following type

$$L(u_t = i|\tilde{Y}_1^T) = \log \frac{P(u_t = i|\tilde{Y}_1^T)}{P(u_t = 0|\tilde{Y}_1^T)}. \tag{6}$$

This definition for the LLR values allows for easy conversion between the *a posteriori* LLR values and index APPs. The other LLRs are related to the corresponding probabilities in a similar fashion. For instance, the value i gives *a priori* information about the random variable u_t with the LLR in form of $L_a(u_t = i) = \log[P(u_t = i)/P(u_t = 0)]$. The conditional LLR of the received value $\tilde{u}_t$ at the channel output, given that the index $u_t = i$ has been transmitted, is given by $L_c(u_t = i) = \log[P(\tilde{u}_t|u_t = i)/P(\tilde{u}_t|u_t = 0)]$. Assuming an AWGN channel with zero mean and variance $\sigma_n^2 = N_0/2E_s$, the conditional probability density function (pdf) of $\tilde{u}_t$ can be formulated as

$$\begin{aligned} p(\tilde{u}_t|u_t) &= \textstyle\prod_{m=0}^{M-1} p(\tilde{u}_t(m)|u_t(m)) \\ &= \left(\frac{1}{\sqrt{2\pi}\sigma_n}\right)^M \exp\left[-\frac{E_s}{N_0} \textstyle\sum_{m=0}^{M-1} (\tilde{u}_t(m) - u_t(m))^2\right]. \end{aligned} \tag{7}$$

The next step is to reduce the large computational burden complexity which is required for computing the logarithmic values of the $\alpha_t^u(i)$ and $\beta_t^u(i)$ terms in (3). This problem can be solved by using the Jacobian logarithm function [15] defined by the property

$$\log(e^{\delta_1} + e^{\delta_2}) = \max\{\delta_1, \delta_2\} + \log(1 + e^{-|\delta_2 - \delta_1|}). \tag{8}$$

Then, for a finite set of real numbers $\{\delta_1, \delta_2, \ldots, \delta_n\}$, the logarithm of their sum can be computed recursively. Suppose that the logarithm of $\Delta_{l-1} = \sum_{j=1}^{l-1} e^{\delta_j}$ with $1 < l \leq n$ is known, then

$$\begin{aligned} \log(\Delta_l) &= \log(\Delta_{l-1} + e^{\delta_l}) \\ &= \max\{\log \Delta_{l-1}, \delta_l\} + \log(1 + e^{-|\log \Delta_{l-1} - \delta_l|}). \end{aligned} \tag{9}$$

For brevity, we use the following shorthand notation $\max_j^*\{\delta_j\} = \log(\sum_{j=1}^{l} e^{\delta_j})$. By taking the logarithm of $\alpha_t^u(i)$ derived in (4), we obtain

$$\begin{aligned} \hat{\alpha}_t^u(i) &= \log \alpha_t^u(i) \\ &= \log P(\tilde{u}_t|u_t = i) + \max_j{}^*\{\log P(u_t = i|u_{t-1} = j) \\ &\quad + \hat{\alpha}_{t-1}^u(j)\} \end{aligned} \tag{10}$$

and similarly,

$$\begin{aligned} \hat{\beta}_t^u(i) &= \log \beta_t^u(i) \\ &= \max_j{}^*\{\log P(\tilde{u}_{t+1}|u_{t+1} = j) \\ &\quad + \log P(u_{t+1} = j|u_t = i) + \hat{\beta}_{t+1}^u(j)\}. \end{aligned} \tag{11}$$

An iterative process using the SISO source decoder as a constituent decoder is realizable, if the *a posteriori* LLR $L(u_t = i|\tilde{Y}_1^T)$ can be separated into four additive terms: the *a* *priori* LLR $L_a(u_t = i)$, the channel-related LLR $L_c(u_t = i)$, and two extrinsic terms resulting from source and channel decoding. In order to determine each of the four terms, we rewrite (6) in log-likelihood algebra as

$$\begin{aligned} &L(u_t = i|\tilde{Y}_1^T) \qquad (12)\\ &\quad = \hat{\alpha}_t^u(i) + \hat{\beta}_t^u(i) - \hat{\alpha}_t^u(0) - \hat{\beta}_t^u(0)\\ &\qquad + \log \frac{P(\tilde{Z}_1^T|u_t=i,\tilde{U}_1^T)}{P(\tilde{Z}_1^T|u_t=0,\tilde{U}_1^T)}\\ &\quad = L_a(u_t = i) + L_c(u_t = i) + L_{SD}^{[ext]}(u_t = i)\\ &\qquad + L_{CD}^{[ext]}(u_t = i) \end{aligned}$$

where

$$\begin{aligned} &L_{SD}^{[ext]}(u_t = i) \qquad (13)\\ &\quad = \hat{\beta}_t^u(i) + \max_j{}^*\{\log P(u_t = i|u_{t-1} = j)\\ &\qquad + \hat{\alpha}_{t-1}^u(j)\} - \hat{\beta}_t^u(0)\\ &\qquad - \max_j{}^*\{\log P(u_t = 0|u_{t-1} = j) + \hat{\alpha}_{t-1}^u(j)\}. \end{aligned}$$

and

$$L_{CD}^{[ext]}(u_t = i) = \log \frac{P(\tilde{Z}_1^T|u_t=i,\tilde{U}_1^T)}{P(\tilde{Z}_1^T|u_t=0,\tilde{U}_1^T)} \qquad (14)$$

A detailed derivation of (14) is presented in the Appendix. The extrinsic LLR $L_{SD}^{[ext]}(u_t = i)$ contains the new part of information resulting from the source decoder by exploiting the residual source redundancy. With respect to (12), the extrinsic LLR can be separated according to

$$\begin{aligned} L_{SD}^{[ext]}(u_t = i) \;=\; & L(u_t = i|\tilde{Y}_1^T) - L_a(u_t = i) \qquad (15)\\ & - L_{CD}^{[ext]}(u_t = i) - L_c(u_t = i) \end{aligned}$$

which is used after interleaving as *a priori* information in the next channel decoding round.

4. SISO Channel Decoding

For the transmission scheme with channel coding, a soft-output channel decoder can be used to provide both estimated bits and their reliability information for further processing to improve the system performance. The commonly used BCJR algorithm is a trellis-based MAP decoding algorithm for both linear block and convolutional codes. The derivation presented in [14] led to a forward-backward recursive computation on the basis of a bit-level trellis diagram, which has two branches leaving each state and every branch represents a single index-bit. Proper sectionalization of a bit-level code trellis may result in useful trellis structural properties [17], [18] and allows us to devise SISO channel decoding algorithms which incorporate parameter-oriented extrinsic information from the source decoder. To proceed with this, we propose a modified BCJR algorithm which parses the received code-bit sequence into M-bit blocks and computes the APP for each quantizer index on a symbol-by-symbol basis. Unlike classical BCJR algorithm that decodes one bit at a time, our scheme proceeds with decoding the quantizer indexes as nonbinary symbols that are matched to the number of bits in an index. By parsing the code-bit sequence into M-bit symbols, we are in essence merging M stages of the original bit-level code trellis into one. As an example, we illustrate in Fig. 3 two stages of the bit-level trellis diagram of a rate 1/2 convolutional encoder with generator polynomial $(7, 5)_8$. The solid lines and dashed lines correspond to the input bits of 0 and 1, respectively. Figure 3 also shows the sectionalized trellis diagram when two stages of the original bit-level trellis are merged together. In general, there are 2^M branches leaving and entering each state in a M-stage merged trellis diagram. Having defined the trellis structure as such, there will be one symbol APP corresponding to each branch which represents a particular quantizer index $x_t = i$. For convenience, we say that the sectionalized trellis diagram forms a finite-state machine defined by its state transition function $F_s(x_t, s_t)$ and output function $F_p(x_t, s_t)$. Viewing from this perspective, the code-symbol associated with the branch from state s_t to state $s_{t+1} = F_s(x_t, s_t)$ can be written as $y_t = (x_t, z_t)$, where $z_t = F_p(x_t, s_t)$ is the parity symbol given state s_t and input x_t.

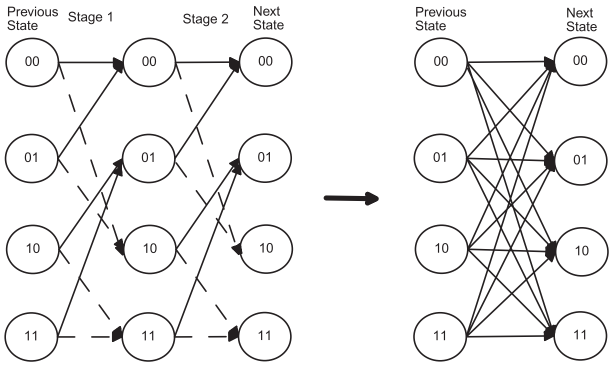

Fig. 3 Bit-level and merged trellis diagrams.

We next applied the sectionalized code trellis to compute the APP of a systematic symbol $x_t = i$ given the received code sequence $\tilde{Y}_1^T = \{\tilde{y}_1, \tilde{y}_2, ..., \tilde{y}_T\}$ in which $\tilde{y}_t = (\tilde{x}_t, \tilde{z}_t)$. Taking the trellis state s_t into consideration, we rewrite the APP as follows:

$$\begin{aligned} P(x_t = i|\tilde{Y}_1^T) \;&=\; C \sum_{s_t} P(x_t = i, s_t, \tilde{Y}_1^T) \qquad (16)\\ &=\; C \sum_{s_t} \alpha_t^x(i, s_t)\beta_t^x(i, s_t), \end{aligned}$$

where $\alpha_t^x(i, s_t) = P(x_t = i, s_t, \tilde{Y}_1^t)$, $\beta_t^x(i, s_t) = P(\tilde{Y}_{t+1}^T|x_t = i, s_t, \tilde{Y}_1^t)$, and $C = 1/P(\tilde{Y}_1^T)$ is a normalizing factor. For the recursive implementation, the forward and backward recursions are to compute the following metrics:

$$\begin{aligned} &\alpha_t^x(i, s_t) \qquad (17)\\ &\quad = \sum_{s_{t-1}} \sum_j P(x_t = i, s_t, x_{t-1} = j, s_{t-1}, \tilde{y}_t, \tilde{Y}_1^{t-1})\\ &\quad = \sum_{s_{t-1}} \sum_j \alpha_{t-1}^x(j, s_{t-1})\gamma_{i,j}(\tilde{y}_t, s_t, s_{t-1}) \end{aligned}$$

$$\begin{aligned}&\beta_t^x(i,s_t)\\&\quad=\sum_{s_{t+1}}\sum_j P(x_{t+1}=j,s_{t+1},\tilde{Y}_{t+1}^T|x_t=i,s_t,\tilde{Y}_1^t)\\&\quad=\sum_{s_{t+1}}\sum_j \beta_{t+1}^x(j,s_{t+1})\gamma_{j,i}(\tilde{y}_{t+1},s_{t+1},s_t)\end{aligned}\tag{18}$$

and in (17)

$$\begin{aligned}&\gamma_{i,j}(\tilde{y}_t,s_t,s_{t-1})\\&\quad=P(x_t=i,s_t,\tilde{y}_t|x_{t-1}=j,s_{t-1},\tilde{Y}_1^{t-1})\\&\quad=P(s_t|x_{t-1}=j,s_{t-1})P(x_t=i|x_{t-1}=j)\\&\qquad\cdot P(\tilde{y}_t|x_t=i,s_t).\end{aligned}\tag{19}$$

Having a proper representation of the branch metric $\gamma_{i,j}(\tilde{y}_t,s_t,s_{t-1})$ is the critical step in applying symbol decoding to error mitigation and one that conditions all subsequent steps of the implementation. As a practical manner, several additional factors must be considered to take advantage of the sectionalized trellis structure and AWGN channel assumption. First, making use of the merged code trellis, the value of $P(s_t|x_{t-1}=j,s_{t-1})$ is either one or zero depending on whether index j is associated with transition from state s_{t-1} to state $s_t=F_s(x_{t-1}=j,s_{t-1})$. The second term in (19) is reduced to $P(x_t=i)$ under the assumption that x_t is uncorrelated with x_{t-1}, which is indeed the case as x_t is the interleaved version of quantizer indexes. For AWGN channels, the third term in (19) is reduced to

$$\begin{aligned}&P(\tilde{y}_t|x_t=i,s_t)\\&\quad=P(\tilde{x}_t|x_t=i)P(\tilde{z}_t|z_t=F_p(x_t=i,s_t))\end{aligned}\tag{20}$$

where the conditional pdfs for the received systematic and parity symbols can be computed analogous to (7). Next, the Jacobian logarithm is used for computing the logarithmic APPs and the corresponding logarithmic values of the $\alpha_t^x(i,s_t)$ and $\beta_t^x(i,s_t)$. By formulating the algorithm in the log-likelihood algebra, we obtain

$$\begin{aligned}\hat{\alpha}_t^x(i,s_t)&=\log\alpha_t^x(i,s_t)\\&=\max_{s_{t-1}}{}^*\{\max_j{}^*\{\hat{\alpha}_{t-1}^x(j,s_{t-1})+\hat{\gamma}_{i,j}(\tilde{y}_t,s_t,s_{t-1})\}\}\end{aligned}\tag{21}$$

and

$$\begin{aligned}\hat{\beta}_t^x(i,s_t)&=\log\beta_t^x(i,s_t)\\&=\max_j{}^*\{\max_{s_{t+1}}{}^*\{\hat{\beta}_{t+1}^x(j,s_{t+1})+\hat{\gamma}_{j,i}(\tilde{y}_{t+1},s_{t+1},s_t)\}\}\end{aligned}\tag{22}$$

and in (21)

$$\begin{aligned}&\hat{\gamma}_{i,j}(\tilde{y}_t,s_t,s_{t-1})=\log\gamma_{i,j}(\tilde{y}_t,s_t,s_{t-1})\\&\quad=\log P(x_t=i)+\log P(s_t|x_{t-1}=j,s_{t-1})\\&\qquad+\log P(\tilde{x}_t|x_t=i)+\log P(\tilde{z}_t|z_t=F_p(x_t=i,s_t)).\end{aligned}\tag{23}$$

Goal of the SISO channel decoder is to compute the *a posteriori* LLR for each systematic symbol $x_t=i$, which can be written as

$$\begin{aligned}L(x_t=i|\tilde{Y}_1^T)&=\log\frac{P(x_t=i|\tilde{Y}_1^T)}{P(x_t=0|\tilde{Y}_1^T)}\\&=\max_{s_t}{}^*\{\hat{\alpha}_t^x(i,s_t)+\hat{\beta}_t^x(i,s_t)\}\\&\quad-\max_{s_t}{}^*\{\hat{\alpha}_t^x(0,s_t)+\hat{\beta}_t^x(0,s_t)\}\end{aligned}\tag{24}$$

Assuming a memoryless channel and channel encoding of systematic form, this *a posteriori* LLR can be separated according to Bayes' theorem into three additive terms: *a priori* term $L_a(x_t=i)=\log[P(x_t=i)/P(x_t=0)]$, the channel-related term $L_c(x_t=i)=\log[P(\tilde{x}_t|x_t=i)/P(\tilde{x}_t|x_t=0)]$, and an extrinsic term $L_{CD}^{[ext]}(x_t=i)$. Substituting (21) and (22) into (24) leads to

$$\begin{aligned}&L(x_t=i|\tilde{Y}_1^T)\\&\quad=L_a(x_t=i)+L_c(x_t=i)+L_{CD}^{[ext]}(x_t=i)\end{aligned}\tag{25}$$

with the extrinsic LLR

$$\begin{aligned}&L_{CD}^{[ext]}(x_t=i)\\&\quad=\max_{s_t}{}^*\{\hat{\beta}_t^x(i,s_t)+\log P(\tilde{z}_t|z_t=F_p(x_t=i,s_t))\\&\qquad+\max_{s_{t-1}}{}^*\{\max_j{}^*\{\log P(s_t|x_{t-1}=j,s_{t-1})\\&\qquad+\hat{\alpha}_{t-1}^x(j,s_{t-1})\}\}\}-\max_{s_t}{}^*\{\hat{\beta}_t^x(0,s_t)\\&\qquad+\log P(\tilde{z}_t|z_t=F_p(x_t=0,s_t)\\&\qquad+\max_{s_{t-1}}{}^*\{\log P(s_t|x_{t-1}=0,s_{t-1})\\&\qquad+\max_j{}^*\{\hat{\alpha}_{t-1}^x(j,s_{t-1})\}\}\}\end{aligned}\tag{26}$$

With respect to an implementation of ISCD the *a priori* LLR in (25) is initialized to be $L_a(x_t=i)$ in terms of the source distribution $P(x_t=i)$. Within iterations the precision of the APP estimation can be enhanced by replacing $L_a(x_t=i)$ with the interleaved extrinsic LLR $L_{SD}^{[ext]}(x_t=i)$ provided by the SISO source decoder. Therefore, the extrinsic LLR resulting from the channel decoding can be calculated by

$$L_{CD}^{[ext]}(x_t=i)=L(x_t=i|\tilde{Y}_1^T)-L_{SD}^{[ext]}(x_t=i)-L_c(x_t=i)\tag{27}$$

and is passed to the source decoder as new *a priori* information for the next iteration. Finally, we summarize the proposed symbol-level ISCD scheme as follows:

1. Initialization: Set the extrinsic information of source decoding to $L_{SD}^{[ext]}(x_t)=0$. Set the iteration counter to $n=0$ and define an exit condition n_{max}.
2. Read series of received sequences $\tilde{Y}_1^T$ and map all received systematic symbols $\tilde{x}_t$ to channel-related LLR $L_c(x_t)$.
3. Perform log-MAP channel decoding by an efficient realization of (24) and then compute the extrinsic LLR $L_{CD}^{[ext]}(x_t)$ using (27).
4. Perform source decoding by inserting the deinterleaved extrinsic LLR $L_{CD}^{[ext]}(u_t)$ into (12) to compute the *a posteriori* LLR $L(u_t|\tilde{Y}_1^T)$. Then, the extrinsic LLR $L_{SD}^{[ext]}(u_t)$ is computed by (15) and is forwarded to the channel decoder as *a priori* information.
5. Increase the iteration counter $n\leftarrow n+1$. If the exit condition $n=n_{max}$ is fulfilled, then continue with step

6, otherwise proceed with step 3.

6. Compute the APP for each estimated index $u_t = i$ as follows:

$$P(u_t = i|\tilde{Y}_1^T) = e^{L(u_t=i|\tilde{Y}_1^T)} / \sum_{j=0}^{2^M-1} e^{L(u_t=j|\tilde{Y}_1^T)}. \qquad (28)$$

7. Estimate the decoder output signals $\hat{v}_t$ by (1) using the index APPs from step 6.

5. Experimental Results

Computer simulations were conducted to compare the performance of various ISCD schemes for transmission over AWGN channels. First a bit-level iterative decoding scheme ISCD1 [11] is considered for error mitigation using the BCJR algorithm for soft-output channel decoding and assisted with the bit reliability information provided by the SBSD. For the ISCD1 scheme with bit interleaving, the source correlation of quantizer indexes is not exploited so that each received channel code-bit is independently decoded. Two approaches to symbol-level iterative decoding, denoted by ISCD2 and ISCD3, are presented and investigated. They both applied a symbol interleaver and performed log-MAP symbol decoding of binary convolutional codes based on sectionalized code trellises. The source decoder in the ISCD3 is different from the SBSD of ISCD2 in that a total of T channel outputs within an interleaving block are evaluated to provide additional reliability gains using an interpolation technique. Specifically, the index APP to be computed for the interpolative SBSD is $P(u_t = i|\tilde{Y}_1^T)$ in (2), and $P(u_t = i|\tilde{Y}_1^t)$ for the SBSD. Following the work of [11], the input signals were the first-order Gauss-Markov sources described by $v_t = \rho v_{t-1} + w_t$, where w_t is a zero-mean, unit-variance white Gaussian noise, with correlation coefficients of $\rho = 0.8$ and $\rho = 0.95$. As indicated in [11], a value of $\rho = 0.95$ can be found for scale factors determined in the MPEG audio codec for digital audio broadcasting. On the other hand, $\rho = 0.8$ provides a good fit to the long-time-averaged autocorrelation function of 8 kHz-sampled telephone speech that is bandpass-filtered to the frequency range of 0.3–3.4 kHz [19]. A total of 3000000 input samples is processed by a scalar M-bit Lloyd-Max quantizer. After natural binary encoding of the quantizer indexes with M bits per index, the resulting bitstreams are divided into blocks of 300 bits. Each of these blocks, consisting of $T = 300/M$ indexes, was spread by an interleaver and afterwards they were encoded by a rate-1/2 recursive systematic convolutional channel code with memory order $\nu = 2$ and generator polynomial $\mathbf{G}(D) = (1, (1 + D^2)/(1 + D + D^2))$.

A preliminary experiment was first performed to examine the step-wise quality gains due to the turbo-like evaluation of channel-code and source-code redundancies. The variation of parameter signal-to-noise ratio (SNR) as a function of the channel SNR E_s/N_0 for ISCD3 simulation of Gauss-Markov sources with $\rho = 0.95$ and $M = 3$ is shown in Fig. 4. One iteration consists of log-MAP channel decoding followed by interpolative SBSD. Within the iterations the curves obtained after the channel decoding process are marked by an upper "+." The lowest curve shows the experimental results of a separate decoding scheme which applied classical channel decoding and source decoding by hard decision and table lookup. For the 0^+-th iteration the log-MAP algorithm for channel decoding with zero extrinsic information $L_{SD}^{[ext]}(x_t) = 0$ is carried out, resulting in a SNR gain of 1.43 dB for a channel condition of $E_b/N_0 = -4$ dB. Next, the 1st iteration is completed, when in addition the deinterleaved extrinsic LLR from channel decoding $L_{CD}^{[ext]}(u_t)$ is exploited in the source decoding process. Due to the high correlation coefficient of $\rho = 0.95$, the parameter SNR can further be improved by up to 9.04 dB. If the updated $L_{SD}^{[ext]}(x_t)$ is fed back to the channel decoder as additional *a priori* information, one additional iteration improves the performance by about 1.37 dB for $E_b/N_0 = -4$ dB. We see that a turbo-like refinement of the extrinsic information from both constituent decoders makes substantial quality improvements possible. The full gain in parameter SNR is reached after three iterations. Figure 5 shows the SNR performance for ISCD3 simulation of Gauss-Markov sources with $\rho = 0.95$ and $M = 4$. The investigation further showed that the improved performance achievable using ISCD3 is more noticeable for higher quantizer rates. Compared to a separate decoding approach, the maximum gain in parameter SNR amounts respectively to 13.86 dB and 16.0 dB for $M = 3$ and $M = 4$ at $E_b/N_0 = -4$ dB.

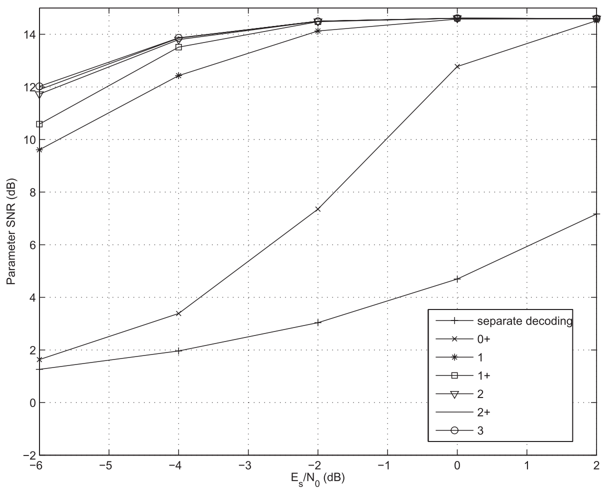

Fig. 4 ISCD3 performance for Gauss-Markov sources with $\rho = 0.95$ and quantizer rate $M = 3$.

To elaborate further, SNR performances of various ISCD schemes were examined for Gauss-Markov sources with $\rho = 0.8$ and $\rho = 0.95$. We provide results for experiments on rate $M = 3$ and $M = 4$ Lloyd-Max quantizers in Figs. 6 and 7, respectively. Three iterations of the algorithm were performed by each decoder as further iterations did not result in a significant improvement. The performances of ISCD schemes were also compared with a non-iterative joint

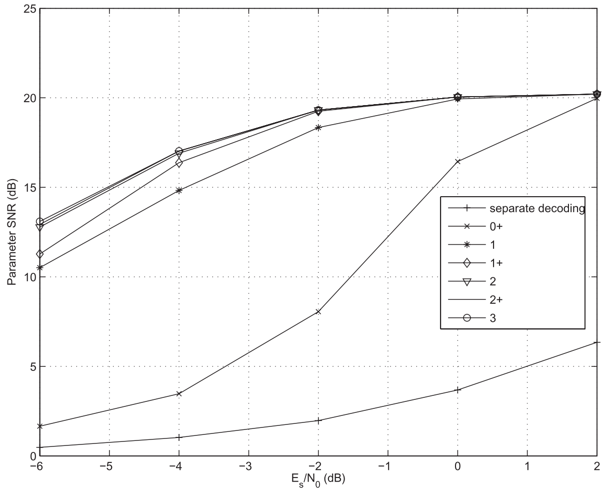

Fig. 5 ISCD3 performance for Gauss-Markov sources with $\rho = 0.95$ and quantizer rate $M = 4$.

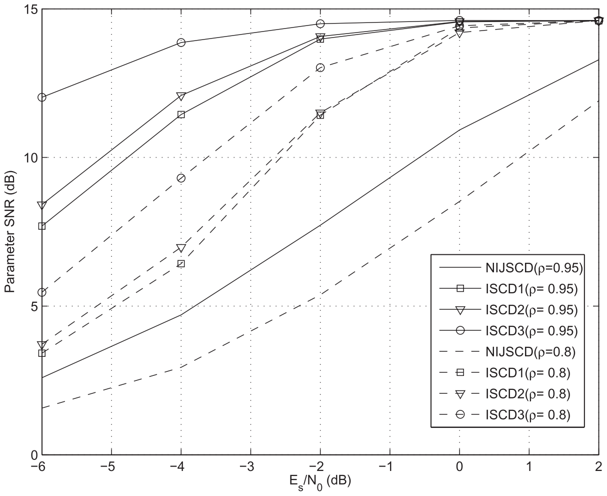

Fig. 6 SNR performance of different decoders for quantizer rate $M = 3$ and Gauss-Markov sources ($\rho = 0.8, 0.95$).

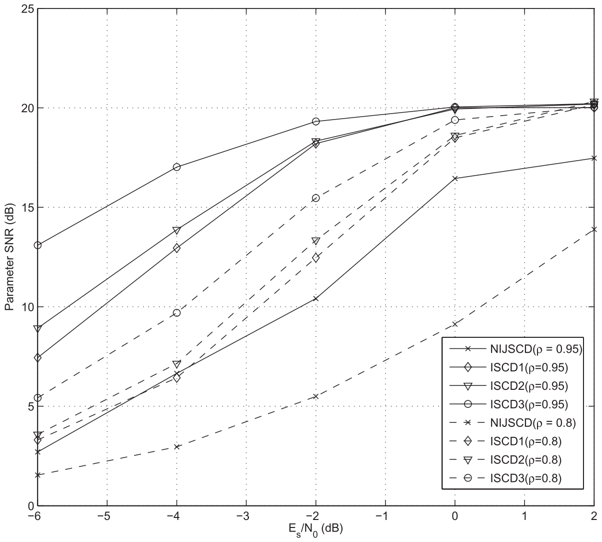

Fig. 7 SNR performance of different decoders for quantizer rate $M = 4$ and Gauss-Markov sources ($\rho = 0.8, 0.95$).

Table 1 Complexity analyses of bit-level and symbol-level channel decoders.

Scheme	Variable	Number of additions
Symbol-level Decoder	$\hat{\alpha}$	$3N_1 \cdot (2^M + 3N_2) \cdot 2^\nu \cdot T$
	$\hat{\beta}$	$3N_1 \cdot (2^\nu + 3N_2) \cdot 2^\nu \cdot T$
Bit-level Decoder	$\hat{\alpha}$	$3N_1 \cdot 2^\nu \cdot T \cdot M$
	$\hat{\beta}$	$3N_1 \cdot 2^\nu \cdot T \cdot M$
	Conversion between bit and index APP	$T \cdot M \cdot [6(2^{M-1} - 1) + 1]$ $+2^{M+1} \cdot T$

source-channel decoder (NIJSCD), which consists of a hard-decision channel decoder followed by a SAMMSE source decoder [7]. Compared with the NIJSCD, the better performances of ISCD can be attributed to its ability to exchange iteratively the reliability gains resulting from SBSD and from channel decoding. The results also show the improved performance achievable using symbol decoders ISCD2 and ISCD3 in comparison to that of bit-based ISCD1. Furthermore, the improvement has a tendency to increase for worse channel conditions and for more heavily correlated Gaussian sources. This indicates that the extrinsic information between two constituent decoders is better to be exploited at the symbol level. The investigation further showed that there is a considerable gap between the ISCD2 and ISCD3 schemes. The difference between them is due to the fact that the ISCD2 only accounts for the past and current channel outputs through the knowledge of APP $P(u_t = i|\tilde{Y}_1^t)$. On the other hand, the ISCD3 decoder takes into account not only the past channel outputs but also a look-ahead of some future channel observations belonging to the same interleaving block.

Finally, we investigate decoding complexity of the ISCD schemes working at bit and symbol levels. The difference between our channel decoding algorithm and the known BCJR algorithm is the substitution of bit-level code trellis by the sectionalized code trellis. The merging of M stages of bit-level trellis results in fewer stages for the forward and backward recursions, and consequently, less $\hat{\alpha}_t^x$'s and $\hat{\beta}_t^x$'s values need to be computed. On the other hand, the merged code trellis has 2^M branches leaving and entering each state, which requires more LLR values to be calculated. Consider the computational complexity of the log-MAP algorithm for symbol decoding of a rate-1/2 convolutional code with memory order ν. Since three additions are required to compute the Jacobian logarithm in (8), a total of $3N_1 \cdot (2^M + 3N_2)$ and $3N_1 \cdot (2^\nu + 3N_2)$ additions are required to compute $\hat{\alpha}_t^x$'s and $\hat{\beta}_t^x$'s in (21) and (22), respectively, where $N_1 = 2^\nu - 1$ and $N_2 = 2^M - 1$. Table 1 gives the number of additions required for symbol-based log-MAP decoding of convolutional codes. The complexity analysis of the ISCD based on bit-level trellis is also included, taking into account the additional complexity for converting the index-level and bit-level extrinsic information from the constituent decoders. For the case of $(T, M, v) = (100, 3, 2)$, the symbol-based channel decoder is about five times as complex as the bit-based channel decoder.

6. Conclusions

In this paper we presented a new ISCD scheme which permits to exchange between its two constituent decoders the symbol-level extrinsic information. First a log-MAP symbol decoding scheme is proposed to decode convolutionally encoded quantizer indexes over AWGN channels and is shown to be superior to the bit-level trellis decoding algorithms. Performance is further enhanced by the use of an interpolative SBSD scheme that exploits a look-ahead of some future channel outputs for better decoding in addition to the past observations. Experimental results indicate that the proposed symbol-level iterative decoding algorithm achieves significant improvements in error robustness for quantization of Gauss-Markov sources over AWGN channels.

Acknowledgement

This study was supported by the National Science Council, Republic of China, under contract NSC 98-2221-E-009-090-MY3.

References

[1] C.E. Shannon, "A mathematical theory of communication," Bell Syst. Tech. J., vol.27, pp.379–429, July 1948.

[2] K. Zeger and A. Gersho, "Pseudo-Gray Coding," IEEE Trans. Commun., vol.38, no.12, pp.2147–2158, Dec. 1990.

[3] N. Farvardin and V. Vaishampayan, "On the performance and complexity of channel-optimized vector quantizers," IEEE Trans. Inf. Theory, vol.37, no.1, pp.155–160, Jan. 1991.

[4] S. Heinen and P. Vary, "Source optimized channel codes for parameter potection," Proc. Int. Symp. on Information Thoery, Sorrento, Italy, June 2000.

[5] J. Hagenauer, "Source-controlled channel decoding," IEEE Trans. Commun., vol.43, no.9, pp.2449–2457, Sept. 1995.

[6] S. Lin and D.J. Costello, Error Control Coding, Prentice-Hall, Englewood Cliffs, N.J., 1983.

[7] D.J. Miller and M. Park, "A sequence-based approximate MMSE decoder for source coding over noisy channels using discrete hidden Markov models," IEEE Trans. Commun., vol.46, no.2, pp.222–231, Feb. 1998.

[8] T. Fingscheidt and P. Vary, "Softbit speech decoding: A new approach to error concealment," IEEE Trans. Speech Audio Process., vol.9, no.3, pp.240–251, March 2001.

[9] C. Berrou and A. Glavieux, "Near optimum error correcting coding and decoding: Turbo codes," IEEE Trans. Commun., vol.44, no.10, pp.1261–1271, 1996.

[10] N. Gortz, "A generalized framework for iterative source-channel decoding," Annals of Telecommunications, Special Issue on Turbo Codes, pp.435–446, July/Aug. 2001.

[11] N. Gortz, "On the iterative approximation of optimal joint source-channel decoding," IEEE J. Sel. Areas Commun., vol.19, no.9, pp.1662–1670, 2001.

[12] M. Adrat, P. Vary, and J. Spittka, "Iterative source-channel decoder using extrinsic information from softbit source decoding," Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Processing, vol.4, pp.2653–2656, Salt Lake City, Utah, USA, May 2001.

[13] M. Adrat and P. Vary, "Iterative source-channel decoding: Improved system design using EXIT charts," EURASIP J. Applied Signal Processing, 2005, vol.6, pp.928–941, May 2005.

[14] L.R. Bahl, J. Cocke, F. Jelinek, and J. Raviv, "Optimal decoding of linear codes for minimizing symbol error rate," IEEE Trans. Inf. Theory, vol.IT-20, no.2, pp.284–287, March 1974.

[15] J.A. Erfanian, S. Pasupathy, and G. Gulak, "Reduced complexity symbol detectors with parallel structures for ISI channels," IEEE Trans. Commun., vol.42, no.2/3/4, pp.1661–1671, 1994.

[16] P. Robertson, E. Villebrun, and P. Hoeher, "A comparison of optimal and sub-optimal MAP decoding algorithms operating in the log domain," Proc. IEEE International Conference on Communication, vol.2, pp.1009–1013, June 1995.

[17] Y. Liu, S. Lin, and M.P.C. Fossorier, "MAP algorithms for decoding linear block codes based on sectionalied trellis diagrams," IEEE Trans. Commun., vol.48, no.4, pp.577–587, April 2000.

[18] M. Bingeman and A.K. Khandani, "Symbol-based turbo codes," IEEE Commun. Lett., vol.3, no.10, pp.285–287, Oct. 1999.

[19] N.S. Jayant and P. Noll, Digital Coding of Waveforms, Prentice-Hall, Englewood Cliffs, N.J., 1984.

Appendix

In this Appendix we shall present a detailed derivation of (14). The APP of a systematic symbol $x_t = i$, given the received code sequence $\tilde{Y}_1^T = (\tilde{X}_1^T, \tilde{Z}_1^T)$ can be decomposed by using the Bayes theorem as

$$\begin{aligned}
&P(x_t = i|\tilde{Y}_1^T) \qquad\qquad (\text{A}\cdot 1)\\
&\quad = P(x_t = i, \tilde{X}_1^T, \tilde{Z}_1^T)/P(\tilde{Y}_1^T)\\
&\quad = P(x_t = i, \tilde{X}_1^T) \cdot P(\tilde{Z}_1^T|x_t = i, \tilde{X}_1^T)/P(\tilde{Y}_1^T)\\
&\quad = P(\tilde{x}_t|x_t = i, \tilde{X}_1^{t-1}, \tilde{X}_{t+1}^T) \cdot P(x_t = i, \tilde{X}_1^{t-1}, \tilde{X}_{t+1}^T)\\
&\qquad \cdot P(\tilde{Z}_1^T|x_t = i, \tilde{X}_1^T)/P(\tilde{Y}_1^T)\\
&\quad = P(\tilde{x}_t|x_t = i, \tilde{X}_1^{t-1}, \tilde{X}_{t+1}^T) \cdot P(x_t = i)\\
&\qquad \cdot P(\tilde{X}_1^{t-1}, \tilde{X}_{t+1}^T) \cdot P(\tilde{Z}_1^T|x_t = i, \tilde{X}_1^T)/P(\tilde{Y}_1^T)\\
&\quad = C \cdot P(\tilde{x}_t|x_t = i) \cdot P(x_t = i) \cdot P(\tilde{Z}_1^T|x_t = i, \tilde{X}_1^T)
\end{aligned}$$

where $C = P(\tilde{X}_1^{t-1}, \tilde{X}_{t+1}^T)/P(\tilde{Y}_1^T)$. We rewrite (A· 1) in log-likelihood algebra as

$$\begin{aligned}
&L(x_t = i|\tilde{Y}_1^T) \qquad\qquad (\text{A}\cdot 2)\\
&\quad = L_a(x_t = i) + L_c(x_t = i) + L_{CD}^{[ext]}(x_t = i)
\end{aligned}$$

with

$$L_{CD}^{[ext]}(x_t = i) = \log \frac{P(\tilde{Z}_1^T|x_t=i,\tilde{X}_1^T)}{P(\tilde{Z}_1^T|x_t=0,\tilde{X}_1^T)}. \qquad (\text{A}\cdot 3)$$

Since the de-interleaved sequence of $L_{CD}^{[ext]}(x_t = i)$ is used by the source decoder, we have

$$L_{CD}^{[ext]}(u_t = i) = \log \frac{P(\tilde{Z}_1^T|u_t=i,\tilde{U}_1^T)}{P(\tilde{Z}_1^T|u_t=0,\tilde{U}_1^T)} \qquad (\text{A}\cdot 4)$$

where $u_t = \Phi^{-1}(x_t)$ and $\tilde{U}_1^T = \Phi^{-1}(\tilde{X}_1^T)$.

Chun-Feng Wu received the B.S. degree in Mathematics from National Taiwan University and the M.S. degree in communication engineering from National Chiao Tung University in 2003 and 2006, respectively. Currently, he is working toward the Ph.D. degree in communication engineering at National Chiao Tung University. His research instrests include multimedia communication, joint source-channel coding and wireless communication.

Wen-Whei Chang received the B.S. degree in communication engineering form National Chiao Tung University, Hsinchu, Taiwan, ROC, in 1980 and the M.Eng. and Ph.D. degrees in electrical engineering from Texas A&M University, College Station, TX, in 1985 and 1989, respectively. Since August 2010, he has been a professor with the Department of Communication Engineering, National Chiao Tung University, Hsinchu, Taiwan, ROC. His current research interests include speech processubg, joun source-channel coding and wireless communication.