Iterative Equalization and Source Decoding for Vector Quantized Sources

Lie-Liang Yang, Jin Wang, Rob Maunder and Lajos Hanzo
School of ECS, University of Southampton, SO17 1BJ, UK
Tel: +44-23-8059 3364; Fax: +44-23-8059 4508
E-mail: lly@ecs.soton.ac.uk; http://www-mobile.ecs.soton.ac.uk

Abstract— **In this contribution an iterative (turbo) channel equalization and source decoding scheme is considered. In our investigations the source is modelled as a Gaussian-Markov source, which is compressed with the aid of vector quantization. The communications channel is modelled as a time-invariant channel contaminated by intersymbol interference (ISI). Since the ISI channel can be viewed as a rate-1 encoder and since the redundancy of the source cannot be perfectly removed by source encoding, a joint channel equalization and source decoding scheme may be employed for enhancing the achievable performance. In our study the channel equalization and the source decoding are operated iteratively on a bit-by-bit basis under the *maximum a-posteriori* (MAP) criterion. The channel equalizer accepts the *a priori* information provided by the source decoding and also extracts *extrinsic* information, which in turn acts as *a priori* information for improving the source decoding performance. Simulation results are presented for characterizing the achievable performance of the iterative channel equalization and source decoding scheme. Our results show that iterative channel equalization and source decoding is capable of achieving an improved performance by efficiently exploiting the residual redundancy of the vector quantization assisted source coding.**

I. Introduction

In practice many communications systems may encounter the intersymbol interference (ISI), when communicating over channels having dispersion resulting from delay-spread. Conventionally, the ISI can be effectively mitigated with the aid of various channel equalization techniques [1], which are usually implemented without invoking any knowledge about the other components, such as error-control coding and source coding. However, when error-control coding is employed, a joint (turbo) equalization and channel decoding scheme may significantly outperform the scheme that implements equalization and channel decoding separately [2], [3].

In this contribution we investigate a joint channel equalization and source decoding scheme, where equalization and source decoding are carried out iteratively, so as to attain an improved BER performance in comparison to the scheme employing channel equalization and source decoding separately. Specifically, when the receiver obtains a set of observation samples, the channel equalizer is first operated based on the MAP criterion, so as to extract *extrinsic* information and convey *a priori* information to the source decoder through an interleaver. With the aid of the *a priori* information extracted from the channel equalizer, source decoding is also carried out based on the MAP principle and, correspondingly, *extrinsic* information is generated. This *extrinsic* information is further fed back to the channel equalizer, so as to provide it with *a priori* information for enhancing the next round of channel equalization. The above-described iterative channel equalization and source decoding process can be continued until no further iteration gain is available or until the maximum number of iterations is reached.

In our study the source is assumed to be a Gaussian-Markov source [4], which is encoded with the aid of vector quantization. The channel is modelled as a time-invariant ISI channel. Our study and simulation results show that the iterative equalization and source decoding scheme is capable of efficiently exploiting the residual redundancy of vector quantization for improving the BER performance. Specifically, for both the 3-path and the 5-path ISI channels considered, an approximately 2dB iteration gain may be achievable, when employing iterative channel equalization and source decoding, instead of carrying out channel equalization and source decoding separately.

II. Preliminaries

A. Vector Quantization

Assume that the source $\{\mathbf{X}_n\}$ is a zero-mean, stationary and Gaussian-Markov vector process. The source encoder is a vector quantizer [4], which is described as follows. The source encoder, $\mathcal{E}$: $\mathcal{R}^k \rightarrow \mathcal{I}_N$, where $\mathcal{I}_N = \{0, 1, \ldots, N-1\}$ and where it is assumed that $N = 2^L$, maps the k-dimensional real-valued source vector $\mathbf{X}_n \in \mathcal{R}^k$ to a finite representation $I_n \in \mathcal{I}_N$, where $I_n = \mathcal{E}(\mathbf{X}_n)$. The encoder mapping $\mathcal{E}$ is defined by a partition $\{\mathcal{R}_i\}_{i=0}^{N-1}$ of the k-dimensional hyperspace $\mathcal{R}^k$ such that we have $\mathbf{X}_n \in \mathcal{R}_i \Rightarrow I_n = i$. Hence, the set $\mathcal{R}_i$ is referred to as the ith encoder region or cell. Let us define the encoder centroids, $\{\mathbf{c}(i)\}_{i=0}^{N-1}$ as $\mathbf{c}_i \triangleq E[\mathbf{X}_n | I_n = i] = E[\mathbf{X}_n | \mathbf{X}_n \in \mathcal{R}_i]$. The ordered set $\{\mathbf{c}(i)\}_{i=0}^{N-1}$ is usually referred to as the codebook of size N, while $i \in \mathcal{I}_N$ is the index of the ith codeword. Let $b_l(i) \in \{-1, +1\}$, $l = 0, 1, \ldots, L-1$ be the bits in the binary representation of the (arbitrary) integer $i \in \mathcal{I}_N$. As shown in Fig. 1, let $\{\mathbf{X}_n = \mathbf{x}_n\}_{n=0}^{N_s-1}$ be N_s number of source vectors and $\{I_n = i_n\}_{n=0}^{N_s-1}$ be their corresponding

0-7803-9392-9/06/$20.00 (c) 2006 IEEE

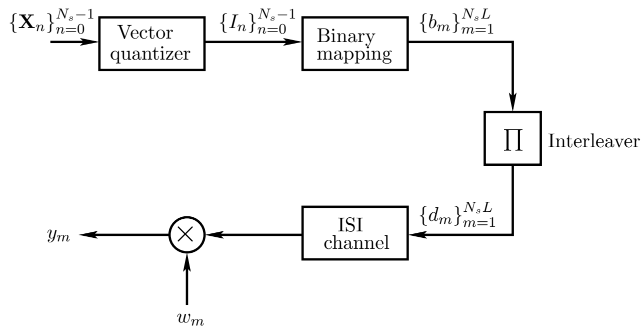

Fig. 1. Representation of a data transmission system including source coding and an interleaver signaling over an intersymbol interference (ISI) channel contaminated by additive Gaussian noise.

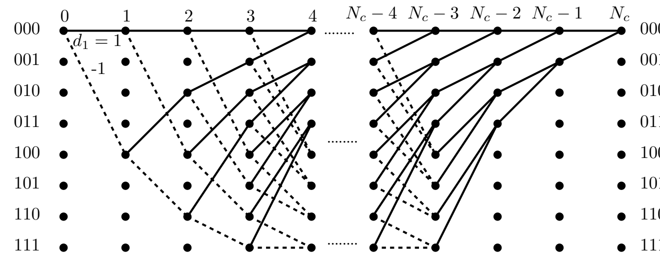

Fig. 2. Trellis diagram of an ISI channel model associated with $M = 3$.

indices at the encoder output. Then, as shown in Fig. 1, these indeces are mapped to a binary sequence $\{b_m\}_{m=1}^{N_sL}$, where $b(nL + l + 1) = b_l(i_n)$. Then, the binary sequence is input to an interleaver, where the binary data is interleaved according to the interleaving function Π. The output of the interleaver is expressed as $\{d_m\}_{m=1}^{N_sL}$. Finally, the interleaved data $\{d_m\}_{m=1}^{N_sL}$ is transmitted over an ISI channel with additive white Gaussian noise (AWGN), as shown in Fig. 1.

B. Channel Model

We assume a coherent symbol-spaced receiver front-end and perfect knowledge of the signal phase and symbol timing, such that the channel can be approximated by an equivalent, discrete-time baseband model. Consequently, the transmitter filter, the channel and the receiver filter can be jointly represented by a discrete-time linear filter with its finite-length impulse response expressed as

$$h_m = \sum_{k=0}^{M} h_l \delta(l-k) \tag{1}$$

where the (real-valued) channel coefficients $\{h_k\}$ are assumed to be time-invariant and known to the receiver. Given the channel impulse response (CIR) of (1) and binary phase shift keying (BPSK) data modulation, the channel output y_m, which is also the equalizer's input, is given by

$$y_m = \sum_{k=0}^{M} h_k d_{m-k} + w_m,\ m = 1, \ldots, N_c \tag{2}$$

where w_m is the zero-mean AWGN having a variance of σ^2, $N_c = N_sL + N_h$ and N_h represents the tail-bits concatenated by the transmitter, in order to take into account the ISI delay. Specifically, $N_h = M$ number of 0s can be transmitted at the tail of a message, so that the discrete-time linear filter converges to the state of zero. Explicitly, the channel output sample of y_m obeys a conditional Gaussian distribution having the probability density function (PDF) expressed as

$$\begin{aligned} & p_{y_m}(y|d_m, d_{m-1}, \ldots, d_{m-M}) \\ & \quad = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(\left[y - \sum_{k=0}^{M} h_k d_{m-k} \right]^2 \Big/ 2\sigma^2 \right) \end{aligned} \tag{3}$$

Let us denote the state of the equivalent discrete-time channel at time nT_b, where T_b represents the bit duration by $\mathbf{q}_n = (d_n, d_{n-1}, \ldots, d_{n-M+1})$. Then, the channel output sample y_n at nT_b depends on the channel state $\mathbf{q}_{n-1}$ and on the input binary symbol d_n. Hence, the equivalent discrete-time channel can be modelled as a Markov chain and its behavior can be represented by a trellis diagram. As an example, Fig.2 shows the trellis diagram of an ISI channel model associated with $M = 3$. In this figure and also in our forthcoming discourse we assume that the channel state starts and terminates at state zero.

III. Iterative Channel Equalization and Source Decoding

In this section we consider the principles of iterative channel equalization and source decoding, and consider the *extrinsic* information that is exchanged between the equalizer and the source decoder. We assume that the source is Gaussian-Markov and that the indeces output by the source encoder can be modelled by a Vth-order Gaussian-Markov process [5]. At the receiver, both the equalizer and the source decoder output soft information, which is derived based on the *maximum a-posteriori* (MAP) criterion. Finally, soft source decoding [6], [7] is employed for the reconstruction of the transmitted source vectors, in the sense that the source decoder carries out a mapping of the continuous-valued information to a source vector estimate.

The receiver structure performing iterative equalization and source decoding is shown in Fig.3. The ultimate objective of the receiver is to provide estimates to the source $\{\mathbf{X}_n\}$, given the channel output samples $\mathbf{y} \triangleq y_1^{N_c}$. More specifically, in the receiver structure of Fig.3, the channel equalizer's information is forwarded for each received data bit to the source decoder through a deinterleaver expressed by the deinterleaving function Π^{-1}. The information conveyed on the feed-forward path is identified by $\Lambda_f(\ )$. By contrast, the source decoder feeds back the *a priori* information associated with each data bit to the channel equalizer through an interleaver having the interleaving function expressed by Π. The information conveyed on the feedback path is identified by $\Lambda_b(\,)$. This process is repeated for a number of times, until finally the source decoder outputs the estimates $\left\{\hat{\mathbf{X}}_n\right\}$ for the source vectors $\{\mathbf{X}_n\}$, with the aid of the information provided by $\{\Lambda_f(b_m)\}$. Let us first investigate the channel equalization process a little further.

A. MAP-Assisted Equalization

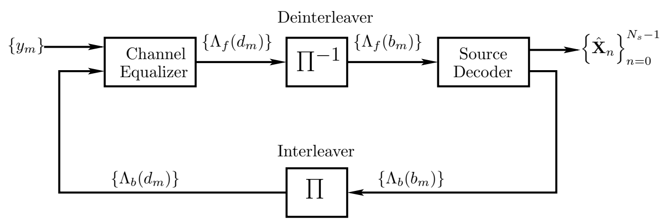

Fig. 3. Receiver schematic performing iterative channel equalization and source decoding.

The channel equalizer considered in this contribution is based on the MAP algorithm, which has been widely employed in turbo decoding [8]–[10]. The MAP equalizer outputs the channel information, which is expressed as $\{\Lambda_f(d_m)\}$, associated with all data bits, given N_c received samples expressed by $\mathbf{y} = y_1^{N_c}$ as well as the *a priori* information $\{\Lambda_b(d_m)\}$ corresponding to all data bits. The feed-forward information $\Lambda_f(d_m)$ corresponding to bit d_m is given by

$$\begin{aligned}\Lambda_f(d_m) &\triangleq \ln\frac{P(d_m=+1|\mathbf{y})}{P(d_m=-1|\mathbf{y})}-\ln\frac{P(d_m=+1)}{P(d_m=-1)}\\ &= \Lambda(d_m|\mathbf{y})-\Lambda_b(d_m) \qquad (4)\end{aligned}$$

where $\Lambda(d_m|\mathbf{y}) = \ln\left[P(d_m=+1|\mathbf{y})/P(d_m=-1|\mathbf{y})\right]$ represents the *a posteriori* log likelihood ratio (LLR) of the data d_m, which can be expressed as by following the approaches in [8], [9], [11], [12],

$$\Lambda\left(d_m|\mathbf{y}\right) = \ln\frac{\sum_{\mathbf{q}_m}\sum_{\mathbf{q}_{m-1}}\alpha_{m-1}(\mathbf{i})\gamma_{+1}(y_m,\mathbf{i},\mathbf{j})\beta_m(\mathbf{j})}{\sum_{\mathbf{q}_m}\sum_{\mathbf{q}_{m-1}}\alpha_{m-1}(\mathbf{i})\gamma_{-1}(y_m,\mathbf{i},\mathbf{j})\beta_m(\mathbf{j})} \qquad (5)$$

where we have

$$\begin{aligned}\alpha_m(\mathbf{i}) &= \sum_{\mathbf{q}_{m-1}}\sum_{u}\alpha_{m-1}(\mathbf{i}')\gamma_u(y_m,\mathbf{i}',\mathbf{i}) \qquad (6)\\ \beta_m(\mathbf{j}) &= \sum_{\mathbf{q}_{m+1}}\sum_{u}\gamma_u(y_{m+1},\mathbf{j},\mathbf{j}')\beta_{m+1}(\mathbf{j}') \qquad (7)\\ \gamma_u(y_m,\mathbf{i},\mathbf{j}) &= P\left(d_m=u;\mathbf{q}_m=\mathbf{j};y_m|\mathbf{q}_{m-1}=\mathbf{i}\right) \qquad (8)\end{aligned}$$

which can be determined using the BJCR algorithm [11] or the SOVA algorithm [8].

B. MAP-Assisted Source Decoding

The information output by the equalizer is conveyed to the source decoder through a deinterleaver associated with the deinterleaving function of Π^{-1}. The corresponding *a priori* information input to the source decoder is expressed as $\Lambda_f(b_m)$ for the mth bit, where $m = 1, 2, \ldots, N_c$. For the sake of convenience, we use $\mathbf{\Lambda}_f$ to express the collection of $\Lambda_f(b_m)$ values for $m = 1, 2, \ldots, N_c$, and use $\Lambda_f(b_{m_1}^{m_2})$ to express the collection of $\Lambda_f(b_m)$ for $m = m_1, m_1+1, \ldots, m_2$. The MAP source decoder computes the *a posteriori* probabilities (APPs) $P(b_m = u|\mathbf{\Lambda}_f) = P(b_m = u|\Lambda_f(b_1), \Lambda_f(b_2), \ldots, \Lambda_f(b_{N_c}))$, $u = +1, -1$ from the N_c bit LLRs $\mathbf{\Lambda}_f$, and outputs the difference

$$\begin{aligned}\Lambda_b(b_m) &= \ln\frac{P\left(b_m=+1|\mathbf{\Lambda}_f\right)}{P\left(b_m=-1|\mathbf{\Lambda}_f\right)}-\ln\frac{P(b_m=+1)}{P(b_m=-1)} \qquad (9)\\ &= \Lambda\left(b_m|\mathbf{\Lambda}_f\right)-\Lambda_f(b_m),\ m=1,\ldots,N_c \qquad (10)\end{aligned}$$

where $\Lambda_f(b_m)$ represents the *a-priori* LLR of the data bit b_m, which is provided by the equalizer seen in Fig.3. In (10) $\Lambda\left(b_m|\mathbf{\Lambda}_f\right) = \ln\left[P\left(b_m=+1|\mathbf{\Lambda}_f\right)/P\left(b_m=-1|\mathbf{\Lambda}_f\right)\right]$ represents the *a-posteriori* LLR of b_m, which is the information about the data bit b_m gleaned from the a priori information about the other data bits based on the characteristics of the source code. Below a recursive algorithm is derived for computing $\Lambda\left(b_m|\mathbf{\Lambda}_f\right),\ m = 1, 2, \ldots, N_c$, which is essentially the modified version of the BCJR algorithm [11].

For the sake of generality, we assume a Vth order Markov source, which can be modelled by a stationary stochastic process $\{I_n\}_{n=1}^{\tilde{N}_s}$, where $\tilde{N}_s \geq N_s + \lceil N_h/L\rceil$ and $\lceil x\rceil$ represents the lowest integer no less than x. Let

$$\mathbf{s}_t = (I_t, I_{t-1}, \ldots, I_{t-V+1}) \qquad (11)$$

be the state of the Markov process at the reference time instant of t, which corresponds to the tth received L-bit source symbol. It can be shown that the state of $\mathbf{s}_t$ is fully determined by the state $\mathbf{s}_{t-1}$ as well as the input source symbol I_t at time t. Hence, the Markov source and its behavior can be represented by a trellis diagram. This trellis diagram has a total of 2^{VL} states, and there are 2^L branches emanating from and entering each of the states. A path segment in the trellis from $t = a$ to $t = b > a$ is determined by the states that are located on this path within $a \leq t \leq b$, which can be expressed as

$$\mathcal{L}_a^b \triangleq (\mathbf{s}_a, \mathbf{s}_{a+1}, \ldots, \mathbf{s}_b) \qquad (12)$$

Denote the input source symbol that engenders the state transition from $\mathbf{s}_{t-1} = s'$ to $\mathbf{s}_t = s$ by $I_t = i_t(s', s)$. Let us assume that the L number of bits in $i_t(s', s)$ are expressed by $\left\{b_{i_t}^l(s', s)\right\}_{l=0}^{L-1}$, and that they are independent, since there is an interleaver between the source and the ISI channel, as seen in Fig.1. Furthermore, we assume that the trellis diagram commences at state $\boldsymbol{s}_0 = \mathbf{0}$, i.e., the state is initialized by binary 0's, and that the trellis diagram also terminates at state $\boldsymbol{s}_{\tilde{N}_s} = \mathbf{0}$.

Consequently, the transition probability from $\mathbf{s}_{t-1} = s'$ to $\mathbf{s}_t = s$, given $\mathbf{\Lambda}_f$, can be expressed as

$$P(\mathbf{s}_t = s|\mathbf{s}_{t-1} = s'; \mathbf{\Lambda}_f) = P[I_t = i_t(s', s)|\mathbf{\Lambda}_f] = \prod_{l=0}^{L-1} P\left[b_{i_t}^l(s', s)|\mathbf{\Lambda}_f\right] \tag{13}$$

where $P\left[b_{i_t}^l(s', s)|\mathbf{\Lambda}_f\right]$ can be expressed as [13]

$$P\left[b_{i_t}^l(s', s)|\mathbf{\Lambda}_f\right] = \frac{1}{2}\left\{1 + b_{i_t}^l(s', s)\tanh\left[\frac{1}{2}\Lambda_f(b_{L(t-1)+l})\right]\right\} \tag{14}$$

Let $\boldsymbol{S}_l^+$ be the set of state pairs (s', s) such that the lth bit, where $l = 0, 1, \ldots, L-1$, of the source symbol $I_t = i_t(s', s)$ is $+1$, i.e., such that $b_{i_t}^l(s', s) = +1$. Similarly, we define $\boldsymbol{S}_l^-$ as the set of state pairs (s', s) such that $b_{i_t}^l(s', s) = -1$. Then, the LLR of $\Lambda(b_m|\mathbf{\Lambda}_f)$ corresponding to $m = (t-1)L + l$ in (10) can be expressed as

$$\Lambda(b_m|\mathbf{\Lambda}_f) = \ln \frac{P(b_m = +1|\mathbf{\Lambda}_f)}{P(b_m = -1|\mathbf{\Lambda}_f)} = \ln \frac{\sum_{(s',s)\in \boldsymbol{S}_l^+} \alpha_{t-1}^{(s)}(s')\beta_t^{(s)}(s)P(\boldsymbol{s}_t = s|\boldsymbol{s}_{t-1} = s'; \mathbf{\Lambda}_f)}{\sum_{(s',s)\in \boldsymbol{S}_l^-} \alpha_{t-1}^{(s)}(s')\beta_t^{(s)}(s)P(\boldsymbol{s}_t = s|\boldsymbol{s}_{t-1} = s'; \mathbf{\Lambda}_f)} \tag{15}$$

where, by definition, we have

$$\alpha_t^{(s)}(s) = \sum_{\mathcal{L}_0^t:\ \boldsymbol{s}_t = s} P\left(\mathcal{L}_0^t|\mathbf{\Lambda}_f\right) \tag{16}$$

$$\beta_t^{(s)}(s) = \sum_{\mathcal{L}_t^{\tilde{N}_s}:\ \boldsymbol{s}_t = s} P\left(\mathcal{L}_t^{\tilde{N}_s}|\mathbf{\Lambda}_f\right) \tag{17}$$

Explicitly, according to the BCJR algorithm [11], the quantities $\alpha_t^{(s)}(s)$ in (16) and $\beta_t^{(s)}(s)$ in (17) can be computed using the following forward and backward recursion equations:

$$\alpha_t^{(s)}(s) = \sum_{\boldsymbol{s}_{t-1} = s'} \alpha_{t-1}^{(s)}(s')P(\boldsymbol{s}_t = s|\boldsymbol{s}_{t-1} = s'; \mathbf{\Lambda}_f), \quad t = 1, 2, \ldots, \tilde{N}_s \tag{18}$$

$$\beta_t^{(s)}(s) = \sum_{\boldsymbol{s}_{t+1} = s'} \beta_{t+1}^{(s)}(s')P(\boldsymbol{s}_{t+1} = s'|\boldsymbol{s}_t = s; \mathbf{\Lambda}_f), \quad t = \tilde{N}_s - 1, \tilde{N}_s - 2, \ldots, 0 \tag{19}$$

The boundary conditions for (18) are $\alpha_0^{(s)}(\mathbf{0}) = 1$ and $\alpha_0^{(s)}(s) = 0$ for $s \neq \mathbf{0}$, and for (19) are $\beta_{\tilde{N}_s}^{(s)}(\mathbf{0}) = 1$ and $\beta_{\tilde{N}_s}^{(s)}(s) = 0$ for $s \neq \mathbf{0}$. Finally, in (15), (18) and (19) the probability expressed in the form of $P(\boldsymbol{s}_t = s|\boldsymbol{s}_{t-1} = s'; \mathbf{\Lambda}_f)$ can be computed by using (13).

C. Source Symbol Reconstruction

After a few iterations between the channel equalizer described by (4) and the source decoder shown in (10), the source decoder of Fig.3 finally attempts to reconstruct the transmitted message using soft source decoding. The soft source decoding is a minimum mean square error (MMSE) estimator [6]. Its objective is to compute the estimate, $\hat{\mathbf{x}}_n$ of the source symbol, $\mathbf{x}_n$ based on the LLRs of $\Lambda(b_m|\mathbf{\Lambda}_f)$ $m = 1, 2, \ldots, N_s L$. The MMSE estimator can be expressed as

$$\begin{aligned}\hat{\mathbf{X}}_n(\mathbf{\Lambda}_f) &= \sum_{i=0}^{N-1} E[\mathbf{X}_n|I_n = i]\, P(I_n = i|\mathbf{\Lambda}_f) \\ &= \sum_{i=0}^{N-1} \mathbf{c}(i)P(I_n = i|\mathbf{\Lambda}_f)\end{aligned} \tag{20}$$

$$\hat{\mathbf{X}}_n(\mathbf{\Lambda}_f(n)) \approx \sum_{i=0}^{N-1} \mathbf{c}(i)P(I_n = i|\mathbf{\Lambda}_f(n)), \quad n = 0, 1, \ldots, N_s - 1 \tag{21}$$

where the approximation of (20) by (21) becomes possible by considering only the LLRs related to the nth source symbol, where $\mathbf{c}(i) = E[\mathbf{X}_n|I_n = i]$, $i = 0, 1, \ldots, N-1$, are the codewords or centroids of each vector quantization partition. Let the binary representation of $I_n = i$ be expressed as $[b_0(i), b_1(i), \ldots, b_{L-1}(i)]$ and their corresponding LLRs as $\mathbf{\Lambda}_f(n) = [\Lambda_f(0), \Lambda_f(1), \ldots, \Lambda_f(L-1)]$. Furthermore, we assume that $b_0(i)$, $b_1(i), \ldots, b_{L-1}(i)$ are independent. Then, the second term in (21) can be written as

$$\begin{aligned}P(I_n = i|\mathbf{\Lambda}_f(n)) &= \prod_{l=0}^{L-1} P[b_l(i)|\Lambda_f(l)] \\ &= \prod_{l=0}^{L-1} \frac{1}{2}\left\{1 + b_l(i)\tanh\left[\frac{1}{2}\Lambda_f(l)\right]\right\}\end{aligned} \tag{22}$$

IV. Performance Examples

In this section we use two examples in order to show the advantages of using iterative channel equalization and source decoding of vector quantization sources. In our simulations characterized in both Figs.4 and 5 a first-order Markov source was assumed, and the source space was partitioned into 16 regions. Hence, the corresponding codebook employs 16 code vectors, which were indexed by 16 number of 4-bit indices. In the context of Fig.4 the 3-path ISI channel was assumed and its channel impulse response (CIR) was assumed to satisfy the constraint of CIR=$[0.407, 0.815, 0.407]$, where the quantity represents the root mean square power of a corresponding path. By contrast, in Fig.5 a 5-path ISI channel was assumed and its CIR was CIR=$[0.227, 0.46, 0.688, 0.46, 0.277]$.

From the results of Figs.4 and 5 we can observe that by employing iterative channel equalization and source decoding, 1-2dB SNR gain can be achieved for both cases. As shown in Figs.4 and 5, the attainable iteration gain is mainly achieved by the first iteration. In both examples considered, further iterations contribute little additional iterative gain. When comparing the results of Fig.4 corresponding to a 3-path channel and that of Fig.5 for a 5-path channel, we can see that when the channel becomes more dispersive resulting in long channel memory, using iterative channel equalization and source decoding is

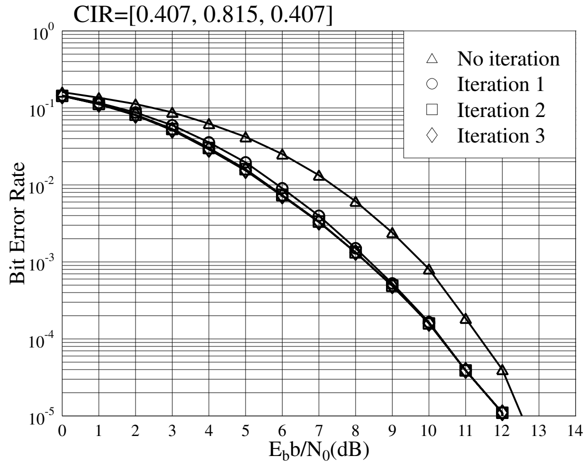

Fig. 4. BER versus SNR per bit performance for iterative equalization and source decoding, when the 3-path channel has the channel impulse response (CIR) obeying CIR=$[0.407, 0.815, 0.407]$ and when a first-order Markov source having the correlation coefficient $\rho = 0.9$ is assumed.

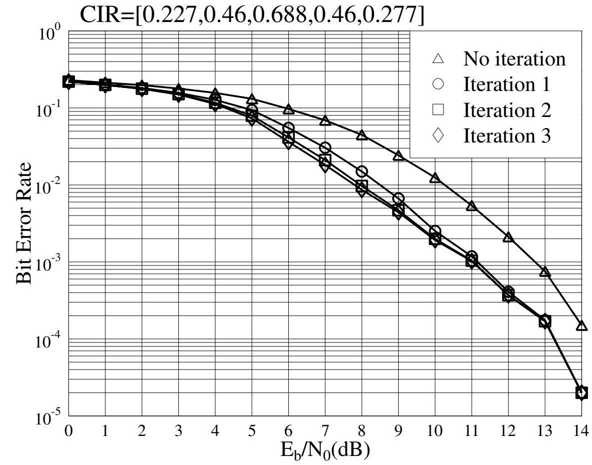

Fig. 5. BER versus SNR per bit performance for iterative equalization and source decoding, when the 5-path channel has the channel impulse response (CIR) obeying CIR=$[0.227, 0.46, 0.688, 0.46, 0.277]$ and when a first-order Markov source having the correlation coefficient $\rho = 0.9$ is assumed.

capable of achieving a higher iteration gain. More specifically, at the BER of 10^{-3}, the iterative gain for the 3-path channel is about 1.5dB, while that for the 5-path channel is about 1.8dB. The highest achievable iterative gain for the 3-path channel of Fig.4 is about 1.8dB, while that for the 5-path channel of Fig.5 is about 2.3dB.

V. CONCLUSIONS

In this contribution we have investigated iterative (turbo) channel equalization and source decoding, when vector quantized source message is transmitted over time-invariant ISI channels. From our analysis and simulation results we may conclude that the iterative channel equalization and source decoding scheme is capable of efficiently exploiting the redundancy that is not removable by the vector quantization assisted source encoding scheme and hence the overall BER performance can be improved. In comparison to the scheme carrying out channel equalization and source decoding separately, the iterative scheme is capable of achieving a SNR gain of about 2dB, when the source is modelled as a first-order Gaussian-Markov source and when the ISI channel has three or five paths, respectively.

ACKNOWLEDGMENT

The financial support of the EPSRC, UK and that of the EU under the auspices of the Phoenix and Newcom projects is gratefully acknowledged.

REFERENCES

[1] J. G. Proakis, *Digital Communications*. McGraw Hill, 3rd ed., 1995.

[2] C. Douillard, M. Jezequel, and C. Berrou, “Iterative correction of intersymbol interference: turbo-equalization,” *European Transactions on Telecommunications*, vol. 6, pp. 507 – 511, September - October 1995.

[3] M. Tuchler, R. Koetter, and A. C. Singer, “Turbo equalization: Principles and new results,” *IEEE Transactions on Communications*, vol. 50, pp. 754 – 767, May 2002.

[4] A. Gersho and R. M. Gray, *Vector Quantization and Signal Compression*. Boston/Dordrecht/London: Kluwer Academic Publishers, 1991.

[5] H. J. Larson and B. O. Shubert, *Probabilitic Models in Engineering Sciences, Volume I: Random Variables and Stochastic Processes*. New York: John Wiley & Sons, 1979.

[6] M. Skoglund, “Soft decoding for vector quantization over noisy channels with memory,” *IEEE Transactions on Informations*, vol. 45, pp. 1293–1307, May 1999.

[7] M. O. Skoglund, “Bit-estimate based decoding for vector quantization over noisy channels with intersymbol interference,” *IEEE Transactions on Communications*, vol. 48, pp. 1309–1317, August 2000.

[8] C. Berrou and A. Glavieux, “Near optimum error correcting coding and decoding: turbo codes,” *IEEE Transactions on Communications*, vol. 44, pp. 1261–1271, October 1996.

[9] J. Hagenauer, E. Offer, and L. Papke, “Iterative decoding of binary block and convolutional codes,” *IEEE Transactions on Information Theory*, vol. 42, pp. 429–445, March 1996.

[10] L. Hanzo, C. H. Wong, and M. S. Yee, *Adaptive Wireless Transceivers : Turbo-Coded, Turbo-Equalised and Space-Time Coded TDMA, CDMA, MC-CDMA and OFDM Systems*. John Wiley & Sons - IEEE Press, 2002.

[11] L. R. Bahl, J. Cocke, F. Jelinek, and J. Raviv, “Optimal decoding of linear codes for minimizing symbol error rate,” *IEEE Transactions on Information Theory*, vol. 20, pp. 284–287, March 1974.

[12] L. R. Rabiner, “A tutorial on hidden Markov models and selected applications in speech recognition,” *Proceedings of the IEEE*, vol. 77, pp. 257–286, February 1989.

[13] X. Wang and H. V. Poor, *Wireless Communication Systems - Advanced Techniques for Signal Reception*. Prentice Hall, 2003.