SWITCHING LMS LINEAR TURBO EQUALIZATION

Seok-Jun Lee, Andrew C. Singer, and Naresh R. Shanbhag

Coordinated Science Laboratory, Electrical and Computer Eng. Dept.
University of Illinois at Urbana-Champaign
1308 West Main Street, Urbana, IL 61801
Email: [slee6,acsinger,shanbhag]@uiuc.edu

ABSTRACT

Turbo equalization using linear filters for data detection has been shown to perform nearly as well as those based on the original maximum *a posteriori* probability (MAP) detection approach. Such linear equalization methods have taken on many forms in the literature, from simple least-mean-square (LMS)-based adaptive filtering approaches, to minimum mean square error (MMSE)-based methods that are recursively computed for each output symbol for each iteration. In this paper, we consider a class of turbo equalization algorithms in which complexity requirements dictate that a fixed set of filter coefficients must be used for all symbols and for all iterations. By computing one such set of coefficients via the LMS algorithm assuming unreliable soft information, and another set assuming highly reliable soft information, we show that a switching strategy can be employed, nearly achieving the performance of recomputing the coefficients at each iteration.

1. INTRODUCTION

Since the introduction of turbo codes [1], a wide variety of iterative algorithms have been adapted into communication systems. Turbo equalization was proposed to protect data transmission over an intersymbol interference (ISI) channel. The original turbo equalization approach [2] employed separate maximum *a posteriori* probability (MAP) decoders for the channel (the equalizer) and the code (the decoder), but this often results in impractically high complexity [3]. A class of low-complexity soft-input soft-output (SISO) equalizers have been proposed [3]–[5] which replace the SISO MAP equalizer, resulting in the so-called "linear turbo equalizer".

Recently, a hybrid technique (called "switching turbo equalization" in this paper) was introduced in [4, 6], where the set of filter coefficients and the method of computing them are switched from one that works well as the iterative process begins (with poor soft-information), to another that works well near convergence (with reliable soft-information). This switching of equalizer coefficients with $O(M)$ complexity results in performance nearly as good as the minimum mean square error (MMSE) optimal approaches (which have $O(M^3)$ or $O(M^2)$ complexity), where the "big O" notation specifies the computational complexity of the required multiplications and M is the number of filter taps. In this paper, we first describe the differences between the approach in [4, 6] and this new least-mean-square (LMS) based switching approach. Then, we describe the algorithm and show the iterative behavior of the switching turbo equalizer using extrinsic information transfer (EXIT) charts [4, 7]. We also evaluate the performance in terms of bit and frame error rates. We observe a 1.5 dB gain at 10^{-4} bit error rate (BER) in comparison with an equalizer where one set of coefficients is employed over all iterations.

This material is based upon work supported by the National Science Foundation under Grant CCR 99-79381, CCR-0092598, and ITR 00-85929.

2. SWITCHING LINEAR TURBO EQUALIZER

In this section, we briefly describe the switching linear turbo equalizer architecture and the computation of equalizer coefficients via the LMS algorithm.

2.1. Architecture

The system model we assume has a transmitter as depicted in Fig. 1 with block-based transmission. Binary data b_n is encoded yielding the coded sequence c_n, which is permuted by the interleaver. The permuted sequence is mapped onto symbols $x_n \in \{-1, +1\}$ (for BPSK modulation) that are transmitted over an ISI channel with additive white Gaussian noise (AWGN). The channel output z_n is given by

$$z_n = \sum_{k=-M_1}^{M_2} h_k x_{n-k} + w_n, \tag{1}$$

for a length $M_1 + M_2 + 1$ channel response h_k and noise sequence w_n.

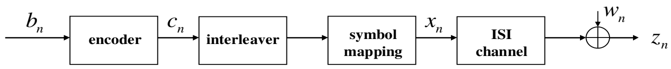

Fig. 1. Channel model including the transmitter.

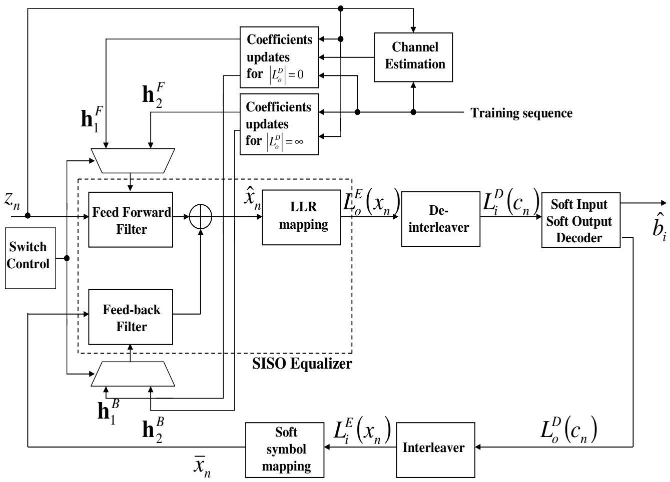

Fig. 2. A switching linear turbo-equalizer block diagram.

Figure 2 depicts the switching linear turbo equalizer architecture, where more than two sets of coefficients can be employed. In this paper two sets of equalizer coefficients are selected, according to the status of the iterative procedure. The linear SISO equalizer consists of two operations: symbol estimation and soft-information mapping of estimated symbols. In comparison with the well-known decision-feedback equalizer (DFE), instead of hard decisions (quantized), soft symbols $\bar{x}_n$ are passed to a feedback filter from the previous iteration of the SISO decoder. The log-likelihood ratio (LLR) mapping converts each estimated symbol $\hat{x}_n$ to a LLR $L_o^E(\cdot)$, which is calculated as

$$L_o^E(x_n) = \ln \frac{Pr(\hat{x}_n | x_n = +1)}{Pr(\hat{x}_n | x_n = -1)}, \tag{2}$$

for BPSK signals. The superscript E and D denote equalization and decoding, respectively, and the subscript o and i refer to the output and input, respectively, in Fig. 2.

The updated $L_o^E(\cdot)$ are fed to the SISO decoder after de-interleaving with a mapping $x_n = 2\Pi(c_n) - 1$, where $\Pi(\cdot)$ is an interleaver operator. The decoder attempts to improve the soft information on the coded bits c_n and produces $L_o^D(\cdot)$, the LLR of each coded bit. In turn, $L_o^D(\cdot)$ is passed to an interleaver with a mapping $x_n = 2\Pi(c_n) - 1$ and then converted to soft symbols $\bar{x}_n$ which are computed as

$$\bar{x}_n = \frac{\exp(L_i^E(x_n)) - 1}{1 + \exp(L_i^E(x_n))}, \tag{3}$$

where $L_i^E(x_n)$ is the interleaved version of $L_o^D(c_n)$ [3]. This soft symbol is then fed back to the equalizer block for the next iteration. The details of such a SISO decoder algorithm are described in [8].

2.2. Coefficient Computation via LMS Algorithm

As shown in Fig. 2, the feedback filter provides the connection between iterations. Thus, this feedback filter coefficients should be computed properly for the algorithm to converge. Note that, in linear SISO equalizers, the inputs to the feedback filter may initially be all zero when the initial values of the LLR satisfy $L_i^E = 0$. After several iterations, as $|L_i^E|$ becomes large, we have that $\bar{x}_n \approx x_n$. However, in a conventional LMS equalizer, perfect knowledge of transmitted symbols is typically assumed to update the coefficients. Thus, the conventional DFE suffers from error propagation when the feedback symbols are not as reliable as assumed. On the other hand, if the feedback coefficients are adapted assuming $L_i^E = 0$, the feedback filter coefficients converge to zero because $\bar{x}_n$ is nearly zero.

In [4], under the $L_i^E = 0$ condition, the estimated symbols are computed in an MMSE sense as

$$\hat{x}_n = (\mathbf{h}_1^F)^T (\mathbf{z}_n - \mathbf{H}\bar{\mathbf{x}}_n), \tag{4}$$

where $\mathbf{h}_1^F$ is expressed as

$$\mathbf{h}_1^F = (\sigma_{w_n}^2 \mathbf{I}_N + \mathbf{H}\mathbf{H}^T)^{-1} \mathbf{s}, \tag{5}$$

$$\mathbf{H} \quad \begin{bmatrix} h_{-M_1} & \cdots & h_{M_2} & 0 & \cdots & \cdots & 0 \\ 0 & h_{-M_1} & \cdots & h_{M_2} & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \cdots & \cdots & \cdots & \cdots & h_{M_2} \end{bmatrix},$$

$$\mathbf{s} \quad \mathbf{H}[\mathbf{0}_{1\times(N_2+M_2-1)}, 1, \mathbf{0}_{1\times(N_1+M_1-1)}]^T,$$

$$\bar{\mathbf{x}}_n = [\bar{x}_{n+N_1+M_1}, \cdots, \bar{x}_{n+1}, 0, \bar{x}_{n-1}, \cdots, \bar{x}_{n-N_2-M_2}]^T,$$

$$\mathbf{z}_n = [z_{n-N_1}, \cdots, z_n, \cdots, z_{n+N_2}]^T,$$

$\mathbf{I}_k$ is $k \times k$ identity matrix, and $\mathbf{0}_{k_1\times k_2}$ is $k_1 \times k_2$ zero matrix. In this paper, the channel knowledge is estimated via the LMS algorithm and $\mathbf{h}_1^F$ is computed as

$$\epsilon_n = x_n - (\mathbf{h}_{1,n}^F)^T \mathbf{z}_n, \tag{6}$$

$$\mathbf{h}_{1,n+1}^F = \mathbf{h}_{1,n}^F + \mu_F \cdot \epsilon_n \cdot \mathbf{z}_n, \tag{7}$$

where μ_F is the step-size for the feedforward filter and n is a time-index during training mode. Then, $\mathbf{h}_1^B$ is computed as

$$\mathbf{h}_1^B = (\mathbf{h}_1^F)^T \hat{\mathbf{H}}, \tag{8}$$

where $\hat{\mathbf{H}}$ is the estimated channel matrix. In order to compute coefficient sets for the highly reliable soft symbol region ($|L_o^D| \to \infty$), $\bar{x}_n$ is assumed to be x_n and then, $\mathbf{h}_2^F$ and $\mathbf{h}_2^B$ can be computed as

$$\epsilon_n = x_n - \{(\mathbf{h}_{2,n}^F)^T \mathbf{z}_n - (\mathbf{h}_{2,n}^B)^T \bar{\mathbf{x}}_n\}, \tag{9}$$

$$\mathbf{h}_{2,n+1}^F = \mathbf{h}_{2,n}^F + \mu_F \cdot \epsilon_n \cdot \mathbf{z}_n, \tag{10}$$

$$\mathbf{h}_{2,n+1}^B = \mathbf{h}_{2,n}^B + \mu_B \cdot \epsilon_n \cdot \bar{\mathbf{x}}_n, \tag{11}$$

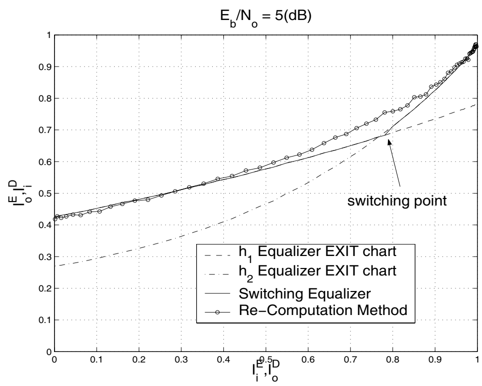

Fig. 3. Comparison on different SISO equalizations using EXIT charts.

where μ_B is the step-size for the feedback filter.

3. EXIT CHART ANALYSIS

A convenient tool for visualizing the soft information evolution of linear SISO equalizers is the so-called EXIT chart [4, 7], which traces the mutual information I_i^E or I_o^E ranging from 0 (no knowledge of the transmitted symbols) to 1 (transmitted symbols are perfectly known) as the algorithm proceeds through iterations. Here, I_i^E and I_o^E are the mutual information between soft information (L_i^E or L_o^E) and x_n (transmitted symbols), respectively. To obtain the linear SISO equalizer EXIT chart, the input soft information L_i^E is reasonably well approximated [4, 7] as independent and identically distributed with

$$f_L(l|x_n) \quad f_L(l|X = x_n) = N(\frac{x_n \sigma_L^2}{2}, \sigma_L^2), \qquad (12)$$

where σ_L^2 is the variance of the soft information. Then, given each L_i^E distribution (I_i^E), the output mutual information I_o^E is measured and plotted in Fig. 3, where the switching point (crossing point of two EXIT charts) is clearly observed. Thus, the soft information evolution trajectory of a switching turbo equalizer should at first follow the EXIT chart of the $\mathbf{h}_1$ equalizer in the less reliable region and then switch to the $\mathbf{h}_2$ equalizer after the switching point. The best linear SISO equalizer (in an LMS or MMSE sense) must re-adapt the coefficients given the new distribution of soft symbols at each iteration. Thus, given each L_i^E distribution, the filter coefficients are adapted and the soft information transfer curve is measured as shown with the line with circles in Fig. 3. This EXIT chart will upper-bound that of $\mathbf{h}_1$ and $\mathbf{h}_2$. However, the switching equalizer (the solid line in Fig. 3) well approximates this performance by employing only two equalizer coefficient sets (with less complexity) and properly switching between them.

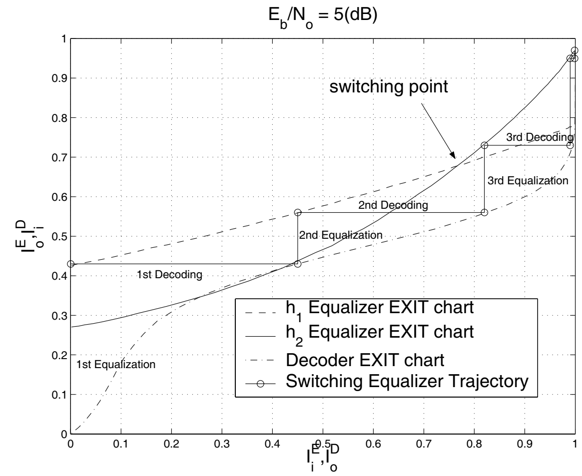

Fig. 4. EXIT chart example for a switching turbo equalizer.

Figure 4 plots both the SISO equalizer and decoder EXIT charts, where the decoder EXIT chart is obtained empirically [7]. Note that, as shown in Fig. 2, L_o^E becomes L_i^D in the current iteration and L_o^D is passed as L_i^E to the SISO equalizer for the next iteration. By mapping $I_o^E \rightarrow I_i^D$ and $I_i^E \rightarrow I_o^D$, a graphical convergence analysis can be undertaken [4, 7]. In Fig. 4, the $\mathbf{h}_1$ generates $I_o^E = 0.43$ in the first iteration, then the decoder produces an improved soft output ($I_o^D = 0.45$). Thus, the iterative procedure computation can be traced by the circled line of Fig. 4 and after 2 iterations the switching occurs and after 5 iterations, the $I_o^D = 1$ condition, and algorithmic convergence, is nearly achieved. Note that, in order to guarantee convergence, a "tunnel" between the two EXIT charts of the equalizer and decoder must appear. Essentially the switching turbo equalizer widens the tunnel between the $\mathbf{h}_1$ and $\mathbf{h}_2$ equalizers and the decoder EXIT charts. Hence, such a switching equalizer could be more robust to error propagation in early stage iterations, where unreliable soft information may lead to poor equalization. Note that if $\mathbf{h}_2$ is employed from the first iteration, the performance improvement stops around $I_o^D = 0.25$. On the other hand, if $\mathbf{h}_1$ is employed over all iterations, $I_e^E = 0.78$ would be achieved near the convergence point, but the switching turbo equalizer can do better with $I_o^E = 0.97$ near the convergence point.

4. EXPERIMENTAL RESULTS

In this section, the performance of the switching turbo equalizer is evaluated over a frequency selective channel. We employ a recursive systematic convolutional (RSC) encoder at the transmitter with a generator polynomial $(23, 35)_8$. The coded bit stream is first passed through a random interleaver followed by a BPSK modulator. We considered the follow-

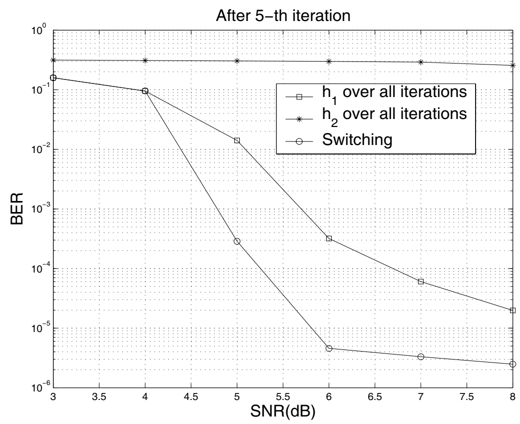

Fig. 5. BER measurement for channel $\mathbf{B}$ with BPSK.

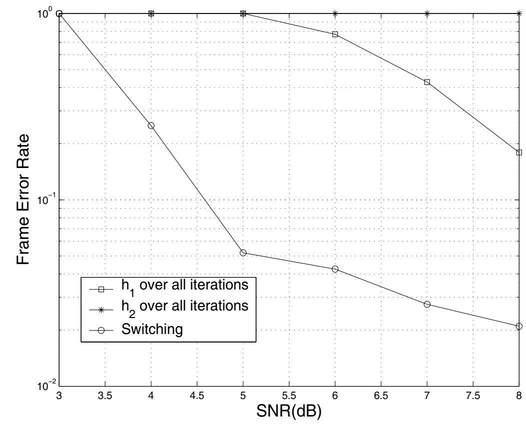

Fig. 6. FER measurement for channel $\mathbf{B}$ with BPSK.

ing static channel model,

$$\mathbf{H}_B(z) = 0.227z^2 + 0.46z + 0.688 + 0.46z^{-1} + 0.227z^{-2},$$

where the channel has severe ISI (strong spectral null near $\omega = 0.6\pi$). We use 65536-bit random interleavers and an LMS algorithm with 2,000 training symbols is used to determine the linear equalizer coefficients ($\mathbf{h}_1$ and $\mathbf{h}_2$). After training, the coefficients held fixed, i.e. they are not updated during the data segment of the transmission. The number of taps used in the feedforward path is 11, and the number of taps in the feedback path is 15. A sliding window log-MAP decoder [8] is employed and 10 iterations are carried out. The switching point is determined based on EXIT chart analysis.

Figures 5 and 6 show bit and frame error rates in comparison with approaches where $\mathbf{h}_1$ and $\mathbf{h}_2$ are employed over all iterations. As expected, the switching turbo equalizer improves the BER and frame error rate (FER) noticeably. At a BER of 10^{-4}, a 1.5 dB gain is observed. Note that if $\mathbf{h}_2$ was employed from the first iteration, the BER would not be improved because the tunnel between the two SISO blocks is closed (see Fig. 4). However, the switching equalizer overcomes this problem by employing $\mathbf{h}_1$ for the first few iterations and then switching to $\mathbf{h}_2$.

5. CONCLUDING REMARKS

We presented a switching LMS turbo equalization, where the MMSE-optimal performance is nearly achieved by selecting different sets of filter coefficients (computed via the LMS algorithm) according to the current status of the iterative processing. Computer simulations demonstrate that a 1.5 dB gain is observed at a BER of 10^{-4} in comparison with approaches where one set of coefficients is used for all iterations.

6. REFERENCES

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

[2] C. Douillard, M. Jezequel, C. Berrou, A. Picart, P. Didier, and A. Glavieux, "Iterative correction of intersymbol interference: Turbo equalization," *European Trans. on Telecommunication,* vol. 6, pp. 507–511, September-October 1995.

[3] A. Glavieux, C. Laot, and J. Labat, "Turbo equalization over a frequency selective channel," in *Proc. of Int. Symp. on Turbo Codes & Related topics,* September 1997, pp. 96–102.

[4] M. Tüchler, R. Koetter, and A. Singer, "Turbo-equalization: principles and new results," *IEEE Trans. on Comm.,* vol. 50, no. 5, pp. 754–767, May 2002.

[5] M. Tüchler, A. Singer, and R. Koetter, "Minimum mean squared error equalization using a-priori information," *IEEE Trans. on Signal Processing*, vol. 50, pp. 673–683, Mar. 2002.

[6] M. Tüchler, R. Koetter, and A. Singer, "Hybrid equalization strategies for iterative equalization and decoding," *Proc. of ISIT,* June 2001, pp. 267.

[7] S. ten Brink, "Convergence behavior of iteratively decoded parallel concatenated codes," *IEEE Trans. on Comm.,* vol. 49, pp. 1727–1737, October 1999.

[8] A. J. Viterbi, "An intuitive justification and a simplified implementation of the MAP decoder for convolutional codes," *IEEE Journal on Selected Areas in Communications*, vol. 16, no. 2, pp. 260–264, Feb. 1998.