LOW-POWER TURBO EQUALIZER ARCHITECTURE

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

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

ABSTRACT

In this paper, we propose a low complexity architecture for turbo equalizers. Turbo equalizers jointly equalize and decode the received signal by exchanging soft information iteratively. The proposed architecture employs early termination of the iterative process when it does not impact the bit-error rate (BER). Early termination enables the powering down parts of the soft-input soft-output (SISO) equalizer and decoder thereby saving power. Simulation results show that the complexity is reduced by 20 % $\sim$ 59 % and 8 % $\sim$ 58 % in equalization and decoding, respectively. In addition, the number of iterations is reduced by 30 % $\sim$ 47 % with negligible degradation in BER.

1. INTRODUCTION

Most high bit-rate communication systems suffer from intersymbol interference (ISI) in addition to noise during transmission over frequency selective channels. Conventional solutions (see Fig. 1) separate the equalization and decoding functions. Recently, the concept of *turbo equalization* has been proposed [1] where equalization and decoding functions are jointly carried out. In turbo equalization, a soft-input soft-output (SISO) linear equalizer and a SISO decoder exchange soft information iteratively. Fig. 2 shows that for very high-speed digital subscriber line (VDSL) systems, turbo equalization increases the transmission data rate by up to 6 % $\sim$ 60 % and transmits data over longer lines ($\geq$ 3.5kft) where non-iterative schemes fail to achieve the recommended BER = 10^{-7} [2]. Furthermore, for a severely attenuated channel with BER = 10^{-4} [3], a turbo equalizer provides a coding gain of 10 dB over conventional systems in which equalization and decoding are carried out separately.

The system level benefits of turbo equalization motivates our research into low-complexity and low-power VLSI architectures for turbo equalizers. Related work includes low-power design techniques which have been proposed for turbo code decoders [4]–[8], where techniques such as early termination, precision optimization, and the optimization of memory accesses have been proposed. So far, however, most research on turbo equalization has focussed on algorithmic issues [1],[9]–[11].

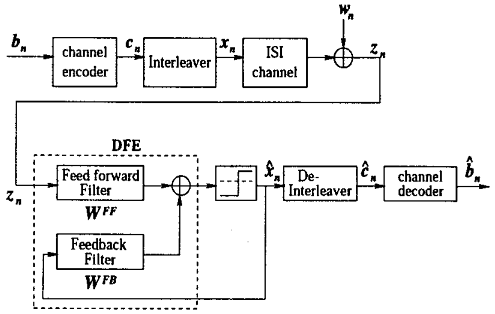


Fig. 1. Conventional communication system : non-iterative receiver.

This paper highlights several key techniques to significantly reduce the complexity required for the linear turbo equalizer implementation. These techniques not only attack the computational inefficiency of the iterative processing by pruning unnecessary operations, but also stop the iteration by employing a novel stopping criterion. Simulation results on typical channels show that the proposed techniques achieve significant savings in computation and hence energy.

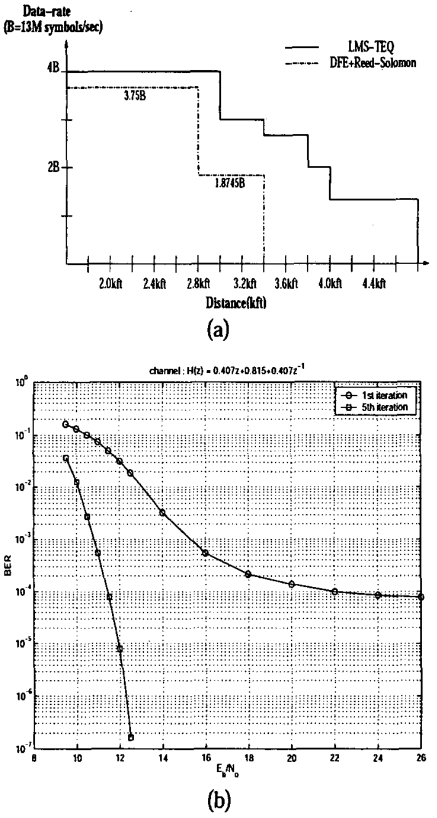
2. TURBO EQUALIZER BASICS

For simplicity, binary phase shift keying (BPSK) is assumed. Extension to higher-order modulation is straightforward and is described in greater detail in [1],[10].

Figure 1 describes the transmitter and the conventional receiver model where a decision feedback equalizer (DFE) is used. The binary data b_n is encoded yielding the coded

This work is supported by NSF grant CCR 99-79381 and ITR 00-85929.

0-7803-7587-4/02/$17.00 ©2002 IEEE



(a)

(b)

Fig. 2. Turbo equalizer vs. conventional equalizer: (a) data-rate advantage for VDSL channels, and (b) BER advantage for a severely attenuated channel [3].

sequence c_n, the interleaver permutes the coded sequence c, and then BPSK modulated symbols $x_n \in \{-1, +1\}$ are transmitted over the 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 channel response h_k and noise sequence w_n. The DFE computes $\hat{x}_n$ from z_n and the hard decision symbols from the slicer, $\tilde{x}_n$, as follows

$$\hat{x}_n = \sum_{k=-L}^{L} W_k^{FF} \cdot z_{n-k} + \sum_{k=-M}^{M} W_k^{FB} \cdot \tilde{x}_{n-k} \tag{2}$$

where $2L+1$ and $2M+1$ are the number of taps in the feed-forward (FF) and feedback (FB) filters, respectively. The slicer outputs are then processed by a channel decoder.

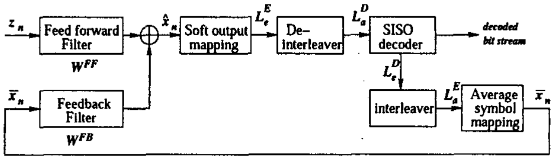


Fig. 3. Linear turbo-equalizer block diagram.

The turbo equalizer shown in Fig. 3 improves receiver performance by exchanging soft information between each SISO block [1]. In the SISO equalizer, soft symbols $\bar{x}_n$ from the previous iteration are used in a feedback filter instead of hard symbols $\tilde{x}_n$, and a soft output mapping converts each estimated symbol to a log-likelihood ratio (LLR), $L_e^E(\cdot)$, which is approximated to be

$$\begin{aligned} L_e^E(x_n) &\simeq \ln \frac{Pr(x_n = +1|\hat{x}_n)}{Pr(x_n = -1|\hat{x}_n)} \\ &= \frac{2\hat{x}_n \mu_{\hat{x}}}{\sigma_{\hat{x}}^2} + L_a^E(x_n), \end{aligned} \tag{3}$$

where $\hat{x}_n \sim N(\mu_{\hat{x}}, \sigma_{\hat{x}}^2)$ if $x_n = +1$ was sent and $\hat{x}_n \sim N(-\mu_{\hat{x}}, \sigma_{\hat{x}}^2)$ if $x_n = -1$ was transmitted. The SISO equalizer outputs, $L_e^E(\cdot)$, are passed through a de-interleaver and fed to the SISO decoder, which updates the extrinsic information on coded bits, c_n, and produces the LLR of each coded bit,$L_e^D(\cdot)$, [1],[9]. $L_e^D(\cdot)$ is passed through interleaver and used as inputs to the average symbol mapping block, which converts LLR values to the average symbol value, $\bar{x}_n$ as follows

$$\begin{aligned} \bar{x}_n &= Pr\{x_n = 1\} - Pr\{x_n = -1\} \\ &= \frac{\exp(L_e^D(x_n)) - 1}{1 + \exp(L_e^D(x_n))}. \end{aligned} \tag{4}$$

The averaged symbols, $\bar{x}_n$, are then fed back into the SISO equalizer for the next iteration.

3. COMPLEXITY REDUCTION TECHNIQUES

3.1. SISO equalization

A snapshot of averaged and estimated symbols over all iterations reveals that different symbols converge at different rates. This fact can be exploited in reducing the complexity because one can stop the iterations after convergence.

This reasoning can be applied directly to the linear SISO equalizer. In each iteration the linear equalizer computes

$$\hat{x}_n^i = \sum_{k=-L}^{L} W_k^{FF} \cdot z_{n-k} + \sum_{k=-M}^{M} W_k^{FB} \cdot \bar{x}_{n-k}^{i-1}, \tag{5}$$

where i denotes the iteration index. If the difference, $\bar{x}^i$-$\bar{x}^{i-1}$', is very small, then $\hat{x}_n^{i+1}$ is close to $\hat{x}_n^i$. It implies that

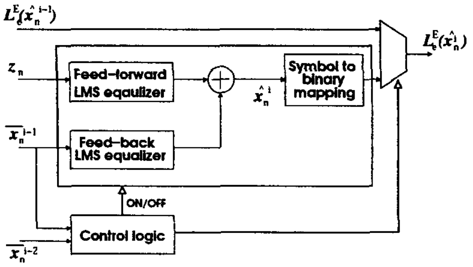


Fig. 4. Modified architecture to control equalization block.

in a certain iteration all the $\hat{x}^{i+1}$ variables in a frame do not need to be updated. However, the very first iteration needs computation of all $\hat{x}^{i+1}$ values. From the second iteration, the equalizer operations, (2) and (3), are selectively carried out only if the update on $L_e^E(\hat{x}_n^i)$ is needed. Figure 4 shows the modified equalizer data-path of Fig. 3, where the extrinsic information of SISO equalizer is computed selectively by using

$$L_e^E(\hat{x}_n^i) = \begin{cases} L_e^E(\hat{x}_n^{i-1}) & \text{if } |\bar{x}_{n-k}^{i-1} - \bar{x}_{n-k}^{i-2}| < \delta_E \\ \frac{2\hat{x}_n^i \mu_{\hat{x}}}{\sigma_{\hat{x}}^2} + L_a^E(x_n) & \text{else} \end{cases} \tag{6}$$

where, δ_E is the equalizer power down threshold and $-L \leq k \leq +L$.

3.2. SISO decoding

In SISO decoding, it is observed that the absolute values of $L_e^D(\cdot)$ can be very small in the first iteration and can be larger than 20.0 in the last iteration [6]. Though $L_e^D(\cdot)$ has a wide dynamic range, a value of $L_e^D(c_n) \geq 8.0$ makes little difference in (4) because $L_e^D(c_n) = 8.0$ is equivalent to an *a priori* probability of $\{c_n = +1\}$ being 0.999 [6]. Therefore, $L_a^D(c_n)$, the input priori value into a SISO decoder, does not need to be updated in case of $L_a^D(c_n) \geq 8.0$. This is because the updated LLR value, $L_e^D(c_n)$, must be larger than 8.0 and $L_a^D(c_n)$ produces the same soft symbol value in (4) as $L_e^D(c_n)$.

In order to remove such unnecessary operations, we employ a sliding window SISO decoding algorithm such as Viterbi's MAX-Log-MAP [12]. The advantage obtained by Viterbi's approach is that processing of a frame can be parallelized without significant BER performance loss [12]. If the size of one sub-frame is set to $L = 6K$, where K denotes the constraint length of a convolutional encoder, and the condition for the jth sub-frame,

$$|L_a^D(c_k)| \geq \gamma, \text{for } j \cdot L \leq k \leq (j+1) \cdot L - 1, \tag{7}$$

is satisfied, then the jth sub-frame does not need to update the LLR's. The control logic in Fig. 5 checks (7) and decides whether the SISO decoding of each sub-frame is to be performed or not. If not, the input LLR's are passed on to the soft symbol mapping block.

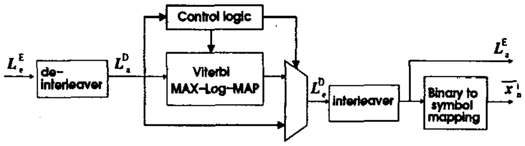


Fig. 5. Modified architecture to control Viterbi MAX-Log-MAP block.

3.3. Stopping criterion

In a turbo equalizer, not all transmitted data frames need the same number of iterations for convergence [11]. It can be reasonably assumed that the magnitudes of $L_e^D(\cdot)$ values are very large once the decoding process has converged [6]. Thus, the average symbols between two consecutive iterations are close to each other and the difference becomes less than a small value (δ_S) as shown below

$$|\bar{x}_n^i - \bar{x}_n^{i-1}| = |\frac{e^{L_e^D(\hat{x}_n^i)} - 1}{e^{L_e^D(\hat{x}_n^i)} + 1} - \frac{e^{L_e^D(\hat{x}_n^{i-1})} - 1}{e^{L_e^D(\hat{x}_n^{i-1})} + 1}| < \delta_S, \forall n, \tag{8}$$

where, $L_e^D(\hat{x}_n^i)$ denotes the decoder extrinsic value on $\hat{x}_n^i$, the interleaved sequence of $\hat{c}_n^i$. This criterion corresponds to the condition when all equalizer operations in a frame are turned off if δ_S is equal to δ_E, the equalizer power down threshold in (6). Therefore, the $\hat{x}_n^{i+1}$ values are not updated leading to the same $L_e^E(\hat{x}_n^{i+1})$ as in the previous iteration. This results in the same decoder extrinsic values, $L_e^D(\hat{c}_n^{i+1})$, as the previous iteration, implying that the decoder has converged. Hence, (8) can be used as a stopping criterion. This stopping criterion exploits the fact that the average symbols do not fluctuate much when the decoder extrinsic values become large and does such a function as the SISO decoder thresholding method. The effect of this stopping criterion on reducing complexity of the SISO decoder is discussed in the next section.

Though various stopping criteria for turbo code decoders have been proposed [4]–[7], they are too complex when applied to a turbo equalizer. However, the proposed scheme is well suited for linear turbo equalizer, and has minimal hardware overhead. Note that the control logic in Fig. 4 can be shared to check the condition in (8). Figure 6 shows the proposed architecture which removes the operational redundancy in joint equalization and decoding processing.

3.4. Determination of threshold values, δ_E, δ_S, and γ

The threshold values, δ_E, δ_S, and γ play an important role in determining the BER and power-down efficiency. The reason is that BER may be degraded too much if δ_E and δ_S

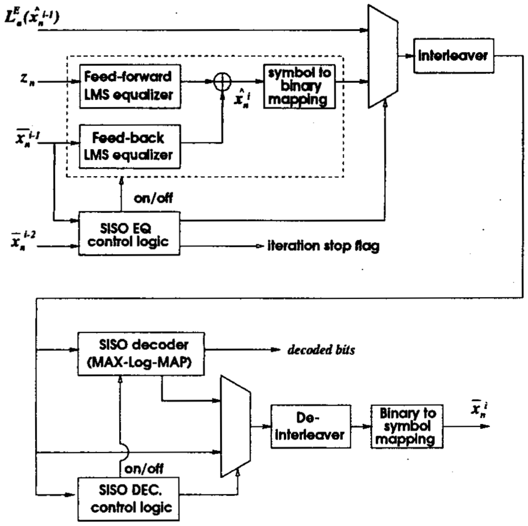


Fig. 6. Proposed reduced-complexity turbo equalizer architecture.

are large while the power-saving efficiency becomes low if δ_E and δ_S are set to a small value. To find out the optimal threshold values, the mutual information (I) between LLR's of SISO decoder and transmitted symbols is measured as

$$I = \frac{1}{2} \sum_{x \in \{\pm 1\}} \sum_{l} \{ f_{L^D_e}(l|x) \cdot \log_2 \frac{2 f_{L^D_e}(l|x)}{f_{L^D_e}(l| + 1) + f_{L^D_e}(l| - 1)} \}, \tag{9}$$

$$f_{L^D_e}(l|x) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(l - \frac{x\sigma^2}{2})^2}{2\sigma^2}\right), \tag{10}$$

where σ denotes the variance of the extrinsic value $L^D_e(\cdot)$ and I can be computed using numerical (Monte Carlo) integration [13]. In (9), I is a measure of the amount of information that L^D_e contains about the transmitted symbols, and hence can be exploited to test the convergence of the SISO decoder performance [14]. Using (9), we measure I_{REF}, of the reference turbo equalizer where all iterations are performed and I_{RED} of a reduced-complexity turbo equalizer where power down techniques are coordinated with the early termination scheme. The threshold values are determined such that

$$I_{REF} - I_{RED} < 10^{-4}, \tag{11}$$

which is equivalent to the condition that the loss in BER is about 10^{-5}. The next section discuss the benefit of this criterion be discussed.

Table 1. Number of taps in equalization.

channel	FF	FB(1st iter.)	FB(other iter.'s)
A	15	7	14
B	7	4	6
C	11	5	10

4. EXPERIMENTAL RESULTS

4.1. Simulation setup

For a channel encoder, we employ a recursive systematic convolutional (RSC) encoder with a generator polynomial $(23, 35)_8$. Before mapping, the coded bit stream is passed through a random interleaver [15] followed by a 4 level pulse amplitude modulation (PAM). For purposes of comparison, we considered three different static channel models (channel **A**, **B** and **C**),

$$\begin{aligned} \mathbf{H}_A(z) &= 0.04z^5 - 0.05z^4 + 0.07z^3 - 0.21z^2 - 0.5z \\ &\quad +0.72 + 0.36z^{-1} + 0.21z^{-3} + 0.03z^{-4} \\ &\quad +0.07z^{-5} \\ \mathbf{H}_B(z) &= 0.407z + 0.815 + 0.407z^{-1} \\ \mathbf{H}_C(z) &= 0.227z^2 + 0.46z + 0.688 + 0.46z^{-1} \\ &\quad +0.227z^{-2}, \end{aligned}$$

where $\mathbf{H}_A(z)$ is a good channel, $\mathbf{H}_B(z)$ has the severe ISI, and $\mathbf{H}_C(z)$ has the worst spectral characteristics [3]. We employ a 1024-bit interleaver for channel **A** and **B** while a 4096-bit interleaver is employed for channel **C** because the larger interleaver overcomes the worst ISI channel, $\mathbf{H}_C(z)$, more effectively . To determine W^{FF}_n and W^{FB}_n, we employ a least-mean-square (LMS) adaptive algorithm. The number of taps used in SISO equalizers are summarized in Table 1. Viterbi's MAX-Log-MAP is employed and 5 iterations for channel **A** and **B** and 8 iterations for channel **C** are carried out. For performance comparison, the decoder BER is measured by transmitting as many frames as is needed so that at least ten erroneous frames are received.

4.2. Results and discussion

Threshold values satisfying (11) are determined via simulations and found to be $\delta_E = \delta_S = 1.0$ and $\gamma = 8.0$ for 4-level PAM. Figure 7 shows that the BER curves of reference and reduced-complexity turbo equalizer are close and these thresholding methods explained in the previous section make little impact on the BER performance.

To measure the efficiency of the proposed scheme, we

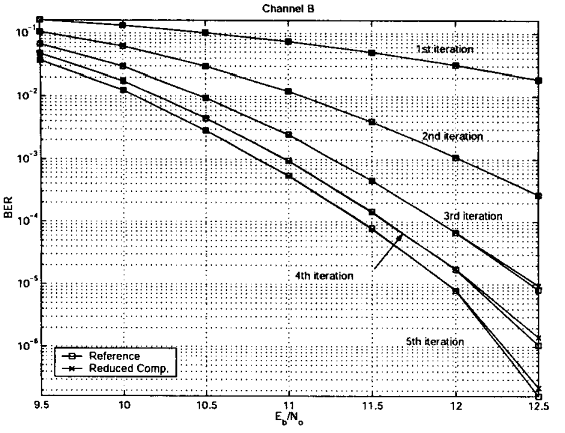


Fig. 7. BER performance comparison versus channel SNR.

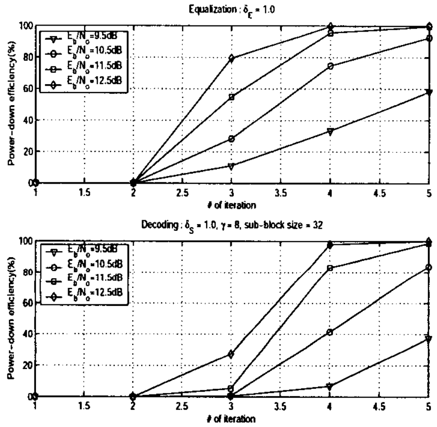


Fig. 8. Power-down efficiency($P_{down}^{(i)}$) vs. iteration index for channel **B**.

define power-down efficiency, $P_{down}^{(i)}$, of ith iteration as

$$P_{down}^{(i)}(\%) = 100 \times \frac{\text{\# of symbols processed by a SISO block}}{\text{\# of transmitted symbols}}, \quad (12)$$

and the average power-down efficiency over all iterations, P_{down}, is computed as

$$P_{down}(\%) = \frac{\sum_i P_{down}^{(i)}}{\text{\# of iterations}}. \quad (13)$$

Figure 8 shows the power down efficiency $P_{down}^{(i)}$ for channel **B**. We see the complexity is reduced much more in the high signal-to-noise ratio (SNR) region than the low SNR region. In particular, the 4th and 5th iterations need little computation, which results in high power-down efficiency. The average efficiency over all iterations is plotted in Fig. 9 for each channel. We see that savings of 20 % to 59 % and 8 % to 58 % have been achieved in the equalizer and decoder, respectively. Table 2 shows the average number of iterations for different channel SNRs. We find that the reduced complexity equalizer achieves a reduction of 30 % to 47 % in the number of iterations.

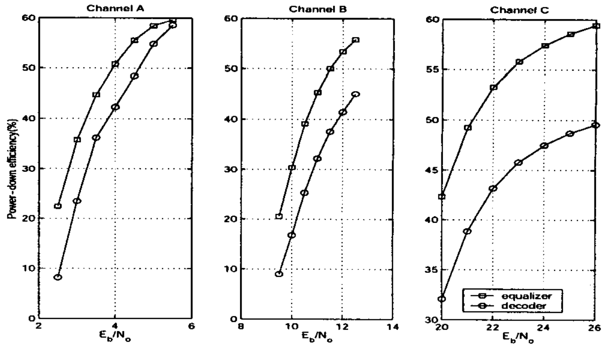


Fig. 9. Power down efficiency(P_{down}).

Table 2. Average number of iterations in reduced complex scheme.

E_b/N_o	$\mathbf{H}_A(z)$	E_b/N_o	$\mathbf{H}_B(z)$	E_b/N_o	$\mathbf{H}_C(z)$
2.5 dB	4.61	9.5 dB	4.57	20 dB	5.5
3.0 dB	3.84	10.0 dB	4.17	21 dB	4.94
3.5 dB	3.2	10.5 dB	3.74	22 dB	4.59
4.0 dB	2.89	11.0 dB	3.39	23 dB	4.37
4.5 dB	2.58	11.5 dB	3.13	24 dB	4.23
5.0 dB	2.26	12.0 dB	2.93	25 dB	4.13
5.5 dB	2.07	12.5 dB	2.75	26 dB	4.06

Figure 10 shows the power down efficiency of two different schemes. One employs three complexity reduction techniques explained in the previous section and the other applies only the stopping criterion to the linear turbo equalizer. It is easily observed that the SISO decoding power down efficiency and the number of required iterations of compared two schemes are very close to each other, implying that the stopping criterion, (8), not only reduces the complexity of SISO decoding, but also terminates the whole iteration, equalization and decoding, when the turbo equalizer performance is converged. However, there is a noticeable difference in the SISO equalizer power down efficiency. Hence, these experimental results verifies that the low complexity SISO equalization technique and the stopping criterion must be jointly employed to achieve the maximum power efficiency.

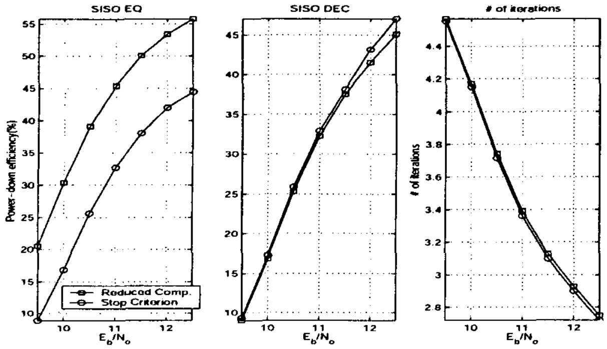


Fig. 10. Power down efficiency(P_{down}) comparison when only stopping criterion is applied.

5. CONCLUDING REMARKS

In this paper, we proposed a high power-down efficient and low complexity scheme by pruning unnecessary operations and employing the new early termination scheme suited for the linear turbo equalizer. These techniques need negligible control logic overhead compared with the main computation logic and are easy to be implemented in VLSI circuits. Computer simulations over various channel conditions reveal that the SISO equalizer complexity reduction technique and the stopping criterion should be jointly employed for the maximum power down efficiency purpose, 20% $\sim$ 59% and 8% $\sim$ 58% computational complexity are reduced in equalization and decoding, respectively, and 30% $\sim$ 47% of required number of iterations are decreased with the negligible BER performance degradation.

6. REFERENCES

[1] C. Laot, A. Glavieux, and J. Labat, "Turbo Equalization: adaptive equalization and channel decoding jointly optimized," *IEEE Journal of Selected Areas in Comm.*, vol. 19, pp. 1744-1752, September 2001.

[2] S. J. Lee, A. C. Singer, and N. R. Shanbhag, "Adaptive Turbo equalizer and its application to VDSL channels," *submitted to Globecom'02.*

[3] J. G. Proakis, *Digital Communications*, 3rd ed. McGRAW-HILL, 1995.

[4] O. Leung, C. Yue, C. Tsui, and R. Cheng, "Reducing power consumption of Turbo Code decoder using adaptive iteration with variable supply voltage," in *Proc. of IEEE Int. Symp. Low Power Electronics Design (ISLPED'99)*, San Diego, CA, 1999, pp. 36-41.

[5] S. Hong, J. Yi, and W. E. Stark, "VLSI design and implementation of low-complexity adaptive TURBO-Code encoder and decoder for wireless mobile communication applications," in *Proc. of IEEE Signal Processing Systems(SiPS): Design and Implementation*, 1998, pp. 233-242.

[6] Z. Wang, H. Suzuki, and K. K. Parhi, "VLSI implementation issues of Turbo decoder design for wireless applications," in *Proc. of IEEE Signal Processing Systems(SiPS): Design and Implementation*, October 1999, pp. 503-512.

[7] D. Garrett, B. Xu, and C. Nicol, "Energy efficient Turbo decoding for 3G mobile," in *Proc. of IEEE Int. Symp. Low Power Electronics Design (ISLPED'01)*, Huntington Beach, CA, 2001, pp. 328-333.

[8] C. Schurgers, F. Catthoor, and M. Engels, "Energy efficient data transfer and storage organization for a MAP Turbo decoder module," in *Proc. of IEEE Int. Symp. Low Power Electronics Design (ISLPED'99)*, San Diego, CA, 1999, pp. 76-81.

[9] G. Bauch and V. Franz, "A comparison of soft-in/soft-out algorithms for turbo detection," in *Proc. of IEEE Int. Conf. on Telecommunications(ICT'98)*, June 1998, pp. 259-263.

[10] M. Tüchler, A. Singer, and R. Koetter, "Minimum mean squared error equalization using a-priori information," *IEEE Trans. on Signal Processing*, vol. 50, no. 3, pp. 673-683, March 2002.

[11] G. Bauch, H. Khorram, and J. Hagenauer, "Iterative equalization and decoding in mobile communications systems," in *Proc. of European Personal Mobile Comm. Conf.(EPMCC)*, October 1997.

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

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

[14] S. ten Brink, "Convergence behavior of iteratively decoded parallel concatenated codes," *IEEE Trans. on Comm.*, vol. 49, pp. 1727-1737, October 2001.

[15] J. Hokfelt, O. Edfors, and T. Maseng, "Interleaver design for turbo codes based on the performance of iterative decoding," in *Proc. of International Conf. on Comm.*, 1999, pp. 93-97.