Joint Timing Recovery and Turbo Equalization for Coded Partial Response Channels

Aravind R. Nayak, John R. Barry, and Steven W. McLaughlin

***Abstract*—We propose a method for jointly performing timing recovery and turbo equalization on partial response channels with error-correction codes. The proposed detector uses soft decisions from previous turbo iterations to improve timing estimates before the next, and each iteration is only marginally more complex than that of a conventional turbo equalizer. As compared to a conventional receiver with separate timing recovery and turbo equalization at $\mathrm{BER} = 2 \times 10^{-5}$, the proposed receiver is superior by 4.7 dB in SNR with a rate-1/4 outer code, and by 2 dB with a rate-8/9 outer code.**

***Index Terms*—Iterative methods, PLL, synchronization.**

I. Introduction

THE PUSH for higher recording densities has motivated the development of iterative error-control codes of unprecedented power, whose large coding gains enable low error rates at very low SNR. Consequently, timing recovery—which typically derives no benefit from coding—must be performed at an SNR lower than ever before.

At a high SNR, the timing-recovery process can be separated from the decoding process with little penalty; timing recovery can use an instantaneous decision device to provide tentative decisions that are adequately reliable, which can then be used to estimate the timing error. In essence, the timing-recovery process is able to ignore the presence of the code, and assume instead that neighboring symbols are independent. At low SNR, however, timing recovery and decoding are intertwined. The timing-recovery process must exploit the presence of the code to get reliable decisions, and the decoder must be fed well-timed samples to function properly.

In principle one could formulate the problem of jointly determining the maximum-likelihood estimates of the timing offsets and message bits, but the complexity would be prohibitive. A solution based on the expectation–maximization algorithm would also be complex [1]. We propose a method for jointly performing the tasks of timing recovery and turbo equalization, with complexity comparable to a conventional turbo equalizer.

In Section II, we describe our channel model. In Section III, we describe conventional timing recovery and conventional turbo equalization. In Section IV, we describe the proposed detector that jointly performs timing recovery and turbo equalization. Finally, in Section V, we present performance results for the proposed detector.

Manuscript received February 13, 2002; revised May 19, 2002. This work was supported by the National Storage Industry Consortium.

The authors are with the School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA 30332 USA (e-mail: nayak@ece.gatech.edu; barry@ece.gatech.edu; swm@ece.gatech.edu).

Digital Object Identifier 10.1109/TMAG.2002.801870.

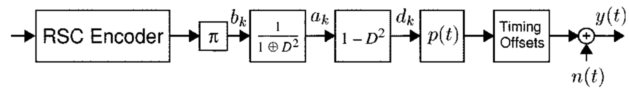

Fig. 1. Data encoding and partial response channel.

II. System Description

We consider the partial-response system shown in Fig. 1, where perfect equalization to a PR-IV target leads to an equalized readback waveform $y(t)$ of

$$y(t) = \sum_k a_k h(t - kT - \tau_k) + n(t) \tag{1}$$

where

T	bit period;
a_k	$\in \{\pm 1\}$precoded symbols;
$h(t)$	$= p(t) - p(t - 2T)$—perfect PR-IV pulse;
$p(t)$	$= \sin(\pi t/T)/(\pi t/T)$—0% excess bandwidth pulse;
$n(t)$	additive white Gaussian noise;
τ_k	unknown timing offset for the kth symbol.

We model the timing offset as a random walk, according to

$$\tau_{k+1} = \tau_k + \mathcal{N}\left(0, \sigma_w^2\right) \tag{2}$$

where σ_w^2 determines the severity of the timing jitter. The random walk model was chosen because of its simplicity and because of its ability to model a wide range of channels by varying a single parameter. We assume perfect acquisition by setting $\tau_0 = 0$.

As shown in Fig. 1, message bits are encoded by a serial concatenation of a recursive systematic convolutional encoder, an s-random interleaver [2], and a $1/(1 \oplus D^2)$ precoder.

III. Conventional Timing Recovery

At the detector, a front-end low-pass filter (with impulse response $p(t)/T$) is used to eliminate out-of-band noise from the readback waveform $y(t)$, producing the bandlimited waveform $r(t)$. Based on the estimates $\hat{\tau}_k$ of τ_k produced by a timing-recovery system, the waveform $r(t)$ is then sampled at the time instants $kT + \hat{\tau}_k$, producing $r_k = \sum_l a_l h(kT + \hat{\tau}_k - lT - \tau_l) + n_k$, where $\{n_k\}$ are independent zero-mean Gaussian samples of variance σ^2.

Conventional timing recovery is based on a phase-locked loop (PLL). For simplicity and because our model has no frequency offset component, we restrict attention to a first-order PLL, which updates its estimate of τ_k according to

$$\hat{\tau}_{k+1} = \hat{\tau}_k + \alpha \hat{\epsilon}_k \tag{3}$$

where α is the PLL gain and $\hat{\epsilon}_k$ is the detector's estimate of the estimation error $\epsilon_k = \tau_k - \hat{\tau}_k$. The widely used Mueller and Müller timing-error detector (TED) generates this estimate according to [3]

$$\hat{\epsilon}_k = \frac{3T}{16}\left(r_k \hat{d}_{k-1} - r_{k-1}\hat{d}_k\right) \tag{4}$$

where $\hat{d}_k$ is an estimate of $d_k = a_k - a_{k-2} \in \{0, \pm 2\}$, typically obtained by a memoryless three-level quantization of r_k. The constant $3T/16$ ensures that there is no bias at high SNR, so that $E[\hat{\epsilon}_k] = \epsilon_k$.

Performance of the Mueller and Müller TED can be improved by using soft estimates $\tilde{d}_k$ in place of hard estimates $\hat{d}_k$[4], [5]. Choosing $\tilde{d}_k = E[d_k | r_k]$ leads to a memoryless soft slicer of the form

$$\tilde{d}_k = \frac{2\sinh\left(\frac{2r_k}{\sigma^2}\right)}{\cosh\left(\frac{2r_k}{\sigma^2}\right) + e^{2/\sigma^2}}. \tag{5}$$

In a conventional setting, the timing-recovery process described above is followed by a turbo equalizer described in [6]. The turbo equalizer iterates between a soft-in soft-out (SISO) equalizer for the precoded PR-IV channel, and a SISO decoder for the outer code, both based on BCJR, which is a maximum *a posteriori* symbol estimation algorithm.

IV. Iterative Timing Recovery

To motivate the proposed method, consider first a conventional detector with a PLL-based timing-recovery followed by a turbo equalizer. After the first iteration, the turbo equalizer could produce soft symbol estimates $\{\tilde{d}_k\}$ that would be more reliable than the tentative decisions of (5). If we were to run the front-end PLL again using the original readback waveform but using the soft decisions from the turbo equalizer, we would get an improved set of timing estimates $\{\hat{\tau}_k^{\text{new}}\}$. Rather than store the continuous-time readback waveform, we would only need to store the original set of samples, since the bandlimited nature of $r(t)$ makes them sufficient statistics. Thus, the second pass of the PLL could arrive at new samples $\{r_k^{\text{new}}\}$ through an interpolation of $\{r_k\}$ according to

$$r_k^{\text{new}} = \sum_l r_l p(kT - lT + \hat{\tau}_k^{\text{new}} - \hat{\tau}_l). \tag{6}$$

We now describe the proposed detector, which is shown in Fig. 2. It begins as described above for the first iteration, with a real-time PLL feeding samples $\{r_k\}$ to a turbo equalizer, which feeds soft estimates $\{\tilde{d}_k\}$ to a second PLL which produces improved timing estimates $\{\hat{\tau}_k^{\text{new}}\}$. The readback waveform is then effectively resampled at the improved sampling instants using interpolation of the original samples. These new samples are then used in the second iteration of the turbo equalizer. The process then repeats: after each iteration of the turbo equalizer, soft estimates from the turbo equalizer are used to improve the timing estimates, which are then used to interpolate the original samples before going on to the next turbo iteration.

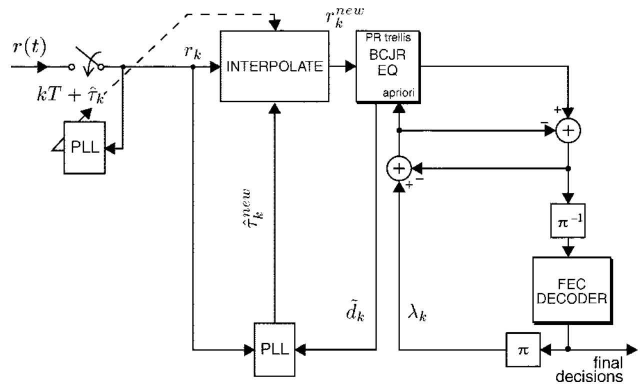

Fig. 2. Joint timing recovery and turbo equalization.

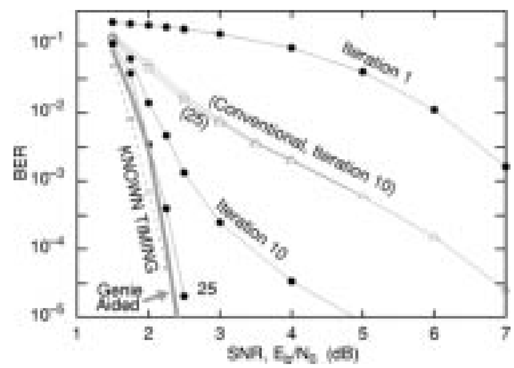

Fig. 3. Results for the rate-1/4 system with moderate timing jitter.

One minor complication is that the second PLL requires soft estimates $\{\tilde{d}_k\}$, but a conventional turbo equalizer for the precoded PR-IV channel produces log-likelihood ratios $\{\lambda_k\}$ for the precoder input $\{b_k\}$. We augment the SISO equalizer to produce both $\{\tilde{d}_k\}$ and $\{\lambda_k\}$. Specifically, the BCJR can track LLRs $\{\lambda'_k\}$ for $\{a_k\}$, and use

$$\tilde{d}_k = E\left[d_k | \{\lambda_k\}, \{\lambda'_k\}\right] = \frac{-2\tanh\left(\frac{\lambda'_{k-2}}{2}\right)}{(1 + e^{-\lambda_k})}. \tag{7}$$

The proposed decoder of Fig. 2 is essentially a modified turbo equalizer, with an interpolation step inserted between consecutive iterations. The complexity increase is marginal, because the complexity of interpolation is usually negligible relative to each turbo iteration. It is worth noting that although we perform timing recovery and turbo equalization jointly, the front end has remained unchanged, and we still sample the continuous-time waveform only once. The modified turbo equalizer is able to correct for poor timing at the front-end PLL.

V. Results

We first consider a rate-1/4 encoder with generator polynomial $[1, 1, 1, (1 \oplus D/1 \oplus D \oplus D^2)]$, which maps blocks of 1278 message bits onto blocks of 5120 coded bits. The rate of the encoder was purposefully chosen to be low, so as to lower the operating SNR and make the timing-recovery problem a difficult one. Fig. 3 plots bit-error rate (BER) versus SNR for this system with $\sigma_w/T = 0.3\%$, which represents a moderate

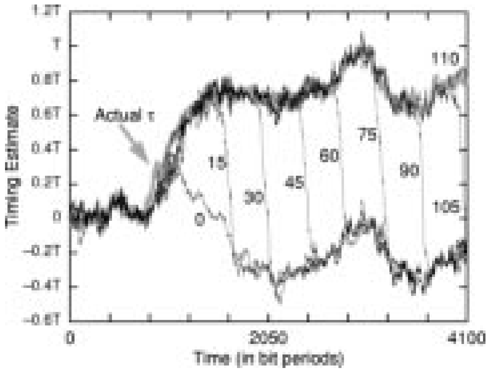

Fig. 4. Cycle-slip correction.

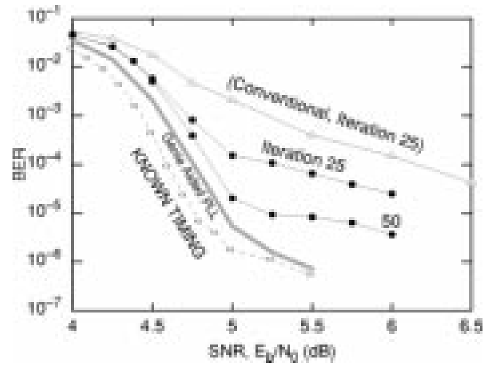

Fig. 5. Results for the rate-8/9 system with severe timing jitter.

random walk with a low probability of cycle slips; $\alpha = 0.025$, chosen to minimize the mean squared error between τ and $\hat{\tau}$ at $\mathrm{SNR} = 5$ dB for the conventional system with memoryless soft decisions; an interleaver parameter of $s = 16$; 21 interpolation coefficients for (6); and at most 50 000 packets for each SNR. The number of packets was chosen to ensure statistical sufficiency. At $\mathrm{BER} = 2 \times 10^{-5}$ and after 25 iterations, we observe a performance gain of 4.7 dB over a conventional system with separate timing recovery and turbo equalization. The performance of the proposed system is 0.2 dB from a turbo equalizer with perfect timing $(\hat{\tau}_k = \tau_k)$.

To further explore the gap between the proposed detector and a detector with perfect timing, Fig. 3 also shows the performance of a genie-aided detector whose PLL has access to training for all bits, which is followed by a conventional turbo equalizer. This ficticious detector provides a lower bound for an iterative timing-recovery scheme that is based on a PLL. As shown in Fig. 3, the genie-aided detector essentially achieves the known timing performance for low target BERs.

Next, we consider a rate-8/9 system in which blocks of 3636 bits are encoded by the rate-1/2 generator $[1, (1 \oplus D \oplus D^3 \oplus D^4/1 \oplus D \oplus D^4)]$, and then punctured to a block length of 4095 bits by retaining only every eighth parity bit. To test the performance in the face of cycle slips, we increase the severity of the random walk jitter to $\sigma_w/T = 0.7\%$, which was found to increase the occurence of cycle slips.

A benefit of the proposed detector is that it can correct cycle slips. Fig. 4 shows the timing waveforms for a sample packet for the rate-8/9 system at $\mathrm{SNR} = 5.0$ dB and $\sigma_w/T = 0.7\%$. The gray curve represents the actual τ sequence, the curve labeled 0 shows $\hat{\tau}$ after the first pass of the PLL, and the other curves show $\hat{\tau}$ after the number of iterations indicated by the corresponding label. We see that the first-pass PLL is not able to track the rapid increase in the actual τ sequence that occurs after about 1000 symbols; instead, $\hat{\tau}$ wanders downward for a few hundred symbol periods until it eventually converges to approximately $\tau - T$, which represents a cycle slip. However, by the time we reach the 15th iteration, the region where the PLL had wandered has been corrected, so that the PLL transitions from perfect lock to a cycle slip in a very short period of time. The resulting steep slope forms a boundary between perfect lock and cycle slip, and this boundary moves from left to right as iterations progress, until eventually the cycle slip is eliminated.

In practice, we can reduce the number of required iterations by detecting the cycle slip and correcting for it [7]. Although cycle-slip detection is difficult in general, it is made easy in our iterative detector, because a cycle slip eventually leads to an abrupt change in $\hat{\tau}$ by $\pm T$, as shown in Fig. 4. Hence, a simple and effective detection method is to declare a slip whenever the magnitude of $\delta_k = \hat{\tau}_k - \hat{\tau}_{k-d}$ exceeds a given threshold Δ, for some delay d. To correct the slip, the detector need only add $\pm T$ to all $\hat{\tau}$ after the slip occurs, with the sign determined by the sign of δ_k.

Fig. 5 shows BER versus SNR for the proposed system with cycle-slip detection and correction, with $\Delta = 0.75$ T, $d = 100$, $s = 24$, $\sigma_w/T = 0.7\%$, $\alpha = 0.04$, chosen as for the rate-1/4 system, and a maximum of 150 000 packets per SNR. At $\mathrm{SNR} = 7$ dB, the BER was 2×10^{-5} after 50 iterations (not shown). Therefore, at $\mathrm{BER} = 2 \times 10^{-5}$ after 50 iterations, the proposed detector is 2 dB better than a conventional detector and 0.3 dB from a turbo equalizer with perfect timing. The genie-aided detector suffers a 0.2 dB penalty relative to a detector with perfect timing. This gap can be closed only by discarding the PLL as the basis for translating symbol estimates to timing estimates. The proposed detector is 0.1 dB worse than the genie-aided detector, a loss that can be attributed to a nonzero BER after turbo equalization.

References

[1] C. N. Georghiades and D. L. Snyder, "The expectation-maximization algorithm for symbol unsynchronized sequence detection," *IEEE Trans. Commun.*, vol. 39, pp. 54–61, Jan. 1991.

[2] C. Heegard and S. Wicker, *Turbo Coding*. Norwell, MA: Kluwer, 1999, pp. 58–61.

[3] K. H. Mueller and M. Müller, "Timing recovery in digital synchronous data receivers," *IEEE Trans. Commun.*, vol. COM-24, pp. 516–531, May 1976.

[4] A. Nayak, J. Barry, and S. McLaughlin, "Timing recovery at low SNR," *NSIC Quarterly Rep.*, Mar. 2000.

[5] H. Song, J. Liu, B. Kumar, and Y. Yuan, "Application of soft output viterbi algorithm (SOVA) in timing recovery for iterative soft decoded magnetic recording channels," presented at the IEEE Int. Magnetics Conf., 2001.

[6] T. Souvignier, A. Friedmann, M. Oberg, P. Seigel, R. Swanson, and J. Wolf, "Turbo decoding for PR4: Parallel vs. Serial concatenation," in *Proc. Int. Conf. Communication*, vol. 3, 1999, pp. 1638–1642.

[7] X. Jin and A. Kavčić, "Cycle-slip detection using soft-output information," in *Proc. Int. Conf. Communication*, vol. 9, 2001, pp. 2706–2710.