# Bi-Directional Timing Recovery for Perpendicular Magnatic Recording Channels

Chanon Warisarn and Pornchai Supnithi
Faculty of Engineering and Industry/University Cooperative
Research Center in Data Storage Technology
and Applications, KMITL, Thailand
Email: s9060053@kmitl.ac.th

Piya Kovintavewat
Data Storage Technology Research Unit
Nakhon Pathom Rajabhat University
Nakhon Pathom, Thailand
Email: piya@npru.ac.th

***Abstract*—Magnetic recording systems employ conventional timing recovery to synchronize the sampler with the readback signal. However, conventional timing recovery does not perform well when the timing error is large. This paper proposes the bi-directional timing recovery, which utilizes conventional timing recovery to sample the readback signal both in forward direction and in backward direction. The outputs of these two operations are averaged and sent them to the Viterbi detector to determine the most likely input sequence. Results indicate that the bi-directional timing recovery performs better than conventional timing recovery, especially when the timing error is large.**

***Index Terms*—Bi-directional timing recovery, conventional timing recovery, perpendicular recording, timing error.**

## I. Introduction

Timing recovery is the process of synchronizing the sampler with the received analog signal. Sampling at the wrong times can have a devastating impact on overall system performance. Therefore, the quality of synchronization is very important for all applications. Practically, magnetic recording systems employ the conventional timing recovery with a 2nd-order phase-locked loop (PLL), which consists of a timing error detector (TED), a loop filter, and a voltage controlled oscillator (VCO), as illustrated in Fig. 1.

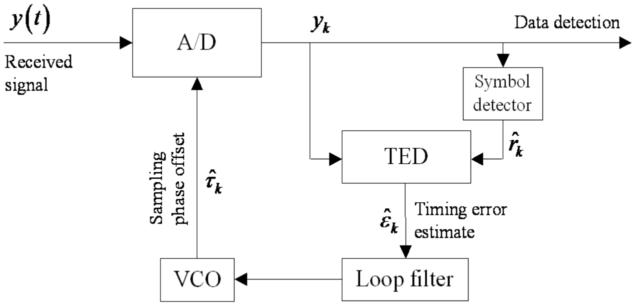

Fig. 1. A conventional timing recovery system.

Many timing recovery systems have been proposed in the literature [1], [2], [3]. Most of them can be catagorized into two types, namely *deductive* timing recovery and *inductive* timing recovery, depending on where the timing information embedded in the received analog signal is extracted [1]. Specifically, the deductive (or *feed-forward*) timing recovery extracts the timing information before the sampler, whereas the inductive (or *feedback*) one extracts the timing information after the sampler. However, both timing recovery architectures utilize a PLL to find the location to sample the received signal. Because the inductive timing recovery is widely used in many applications [1], it will then be referred to as *conventional timing recovery*, whose architecture is shown in Fig. 1.

This paper proposes a simple timing recovery architecture, which consists of two timing recovery blocks running in parallel as depicted in Fig. 2. The first block (i.e., branch $\mathbf{A}$) employs a conventional timing recovery to sample the readback signal, while the second block (i.e., branch $\mathbf{B}$) *reverses* the whole readback signal before passing the reversed signal to conventional timing recovery. The outputs of the two timing recovery blocks are averaged and sent the resulting sequence to the Viterbi detector (VD) to determine the most likely input sequence. We refer to the proposed timing recovery architecture as "bi-directional timing recovery." It can be seen in simulations that the bi-directional timing recovery can help improve the system performance if compared to conventional timing recovery.

This paper is organized as follows. Section II describes our channel model and explains how conventional timing recovery works. The bi-directional timing recovery scheme is described in Section III, and its performance is compared with conventional timing recovery in Section IV. Finally, Section V concludes this paper.

## II. System Descriptions

We consider the perfectly equalized PR2 channel model shown in Fig. 2, where the readback signal can be written as

$$s\left(t\right) = \sum_{k=0}^{L-1} a_k h(t - kT - \tau_k) + n(t), \tag{1}$$

where $a_k \in \pm 1$ is an input data sequence of length $L$ with bit period $T$, $h(t) = p(t) + 2p(t - T) + p(t - 2T)$ is a PR2 pulse, $p(t) = \sin(\pi t/T)/(\pi t/T)$ is an ideal zero-excess-bandwidth Nyquist pulse, and $n(t)$ is additive white Gaussian noise (AWGN) with two-sided power spectral density $N_0$/2. The timing offset, $\tau_k$, is modeled as a random walk model [4] according to

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

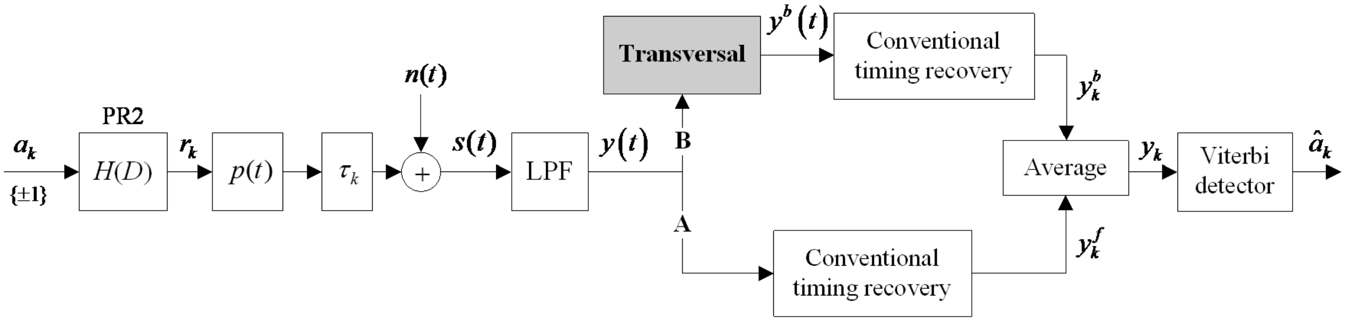

Fig. 2. The perfectly equalized PR2 channel model with bi-directional timing recovery.

where $\sigma_w$ determines the severity of the timing offset. The random walk model is chosen because of its simplicity to represent a variety of channels by changing only one parameter. We also assume perfect acquisition by setting $\tau_0 = 0$.

At the receiver, the readback signal $s(t)$ is filtered by an ideal low-pass filter (LPF), whose impulse response is $p(t)/T$, to eliminate the out-of-baud noise, and is sampled at time $kT + \hat{\tau}_k$, creating

$$y_k^f = y(kT + \hat{\tau}_k^f) = \sum_i a_i h(kT + \hat{\tau}_k^f - iT - \tau_i) + n_k, \quad (3)$$

where $\hat{\tau}_k$ is the receiver's estimate of $\tau_k$, and $n_k$ is *i.i.d.* zero-mean Gaussian random variable with variance $\sigma_n^2 = N_0/(2T)$.

Conventional timing recovery is based on a PLL as shown in Fig. 1. A decision-directed TED [1] computes the receiver's estimate of the timing error $\epsilon_k = \tau_k - \hat{\tau}_k$ using the well-known Mueller and M$\ddot{u}$ller (M&M) TED algorithm [5] according to

$$\hat{\epsilon}_k^f = \frac{6T}{40}\{y_k^f \hat{r}_{k-1} - y_{k-1}^f \hat{r}_k\}, \quad (4)$$

where $\hat{r}_k$ is the $k$-th estimate of the noiseless *channel output* obtained from the symbol detector. The constant $6T/40$ assures that there is no bias at high signal-to-noise ratio (SNR) so that $E[\hat{\epsilon}_k|\epsilon] = \epsilon$ (see a proof in Appendix). Note that the symbol detector used in the timing loop is the VD with a decision delay of $4T$. Because perfect acquisition is assumed and our model has no frequency offset component, the sampling phase offset is then updated by a 1st-order PLL according to

$$\hat{\tau}_{k+1} = \hat{\tau}_k + \alpha\hat{\epsilon}_k, \quad (5)$$

where $\alpha$ is a PLL gain parameter [1]. Eventually, the VD performs maximum-likelihood equalization to determine the most likely input data sequence, $\hat{a}_k$.

## III. Bi-directional Timing Recovery

The key idea of bi-directional timing recovery is to sample the readback signal both in forward direction and in backward direction. Specifically, for forward direction (i.e., branch **A**), the readback signal is sampled by the same conventional timing recovery as explained in the Section II to obtain a sequence $y_k^f$. Nonetheless, for backward direction (i.e., branch **B**), the whole readback signal is reversed to obtain the reversed signal $y^b(t)$ before passing it to conventional timing recovery to obtain a sequence $y_k^b$.

The reversed signal $y^b(t)$ is sampled at time $kT + \hat{\tau}_k^b$ to obtain

$$y_k^b = y^b(kT + \hat{\tau}_k^b), \quad (6)$$

where $\hat{\tau}_k^b$ is the $k$-th sampling phase offset in backward direction. Note that to sample the reversed signal $y^b(t)$, we set $\hat{\tau}_0^b = -\hat{\tau}_L^f$, where $\hat{\tau}_0^b$ is the first sampling phase offset in backward direction and $\hat{\tau}_L^f$ is the last sampling phase offset in forward direction.

We still use the M&M TED algorithm to compute the estimate of the backward timing error, $\epsilon_k^b$, which can be obtained by

$$\hat{\epsilon}_k^b = \frac{6T}{40}\left\{y_k^b \hat{r}_{k-1} - y_{k-1}^b \hat{r}_k\right\}. \quad (7)$$

Then, the next sampling phase offset in backward direction is updated by a 1st-order PLL according to

$$\hat{\tau}_{k+1}^b = \hat{\tau}_k^b + \alpha\hat{\epsilon}_k^b, \quad (8)$$

where the same PLL gain parameter, $\alpha$, is employed.

Because conventional timing recovery in forward direction produces a set of $\{y_k^f, \hat{\tau}_k^f\}$ and that in backward direction also produces a set of $\{y_k^b, \hat{\tau}_k^b\}$, there are two options to exploit this information to improve the performance of synchronization. The first option is to find the averaged sampling phase offset according to

$$\hat{\tau}_k = \frac{\hat{\tau}_k^f + \hat{\tau}_k^b}{2}. \quad (9)$$

Then, we resample the readback signal $y(t)$ using a set of $\{\hat{\tau}_k\}$ to obtain $y_k = y(kT + \hat{\tau}_k)$. However, to reduce the complexity, we can directly average the sampler outputs $\{y_k^f, y_k^b\}$ according to

$$y_k = \frac{y_k^f + y_k^b}{2} \quad (10)$$

Consequently, a sequence $\{y_k\}$ is sent to the VD to perform sequence detection.

Based on extensive simulations, we found that bi-directional timing recovery based on both options yields a similar result. Therefore, in this paper, we will consider only bi-directional

```
(A-1)   Initialize $\hat{\tau}_0^f = 0$
(A-2)   For $k = 0, 1, \ldots, L-1$
(A-3)       $y_k^f = y^f(kT + \hat{\tau}_k^f)$
(A-4)       $\hat{\epsilon}_k^f = \frac{6T}{40}\{y_k^f \hat{r}_{k-1} - y_{k-1}^f \hat{r}_k\}$
(A-5)       $\hat{\tau}_{k+1}^f = \hat{\tau}_k^f + \alpha \hat{\epsilon}_k^f$
(A-6)   End
(A-7)   Reverse the readback signal $y(t)$ to obtain $y^b(t)$
(A-8)   Initialize $\hat{\tau}_0^b = -\hat{\tau}_L^f$
(A-9)   For $k = 0, 1, \ldots, L-1$
(A-10)      $y_k^b = y^b(kT + \hat{\tau}_k^b)$
(A-11)      $\hat{\epsilon}_k^b = \frac{6T}{40}\{y_k^b \hat{r}_{k-1} - y_{k-1}^b \hat{r}_k\}$
(A-12)      $\hat{\tau}_{k+1}^b = \hat{\tau}_k^b + \alpha \hat{\epsilon}_k^b$
(A-13)  End
(A-14)  Average the sampler outputs by $y_k = (y_k^f + y_k^b)/2$
(A-15)  Send $y_k$ to the Viterbi detector to obtain $\hat{a}_k$
```

Fig. 3. Algorithm of bi-directional timing recovery.

timing recovery based on the second option because it has less complexity if compared to the first option. Fig. 3 shows the algorithm of bi-directional timing recovery, which will be used to compared the performance with the conventional timing recovery in Section IV.

## IV. Numerical Results

We consider the system in moderate and severe timing offsets (i.e., $\sigma_w/T = 0.7\%$ and $\sigma_w/T = 1.2\%$). We employ the PLL gain parameter, $\alpha$, designed to recover phase change within $C$ = 100 symbols based on a linearized model of PLL [1], assuming that the S-curve slope [1] is one at the origin, and there is no noise in the system. The $\alpha$ designed for the delay of $4T$ is 0.027. We also assume that one data packet consists of 4096 data bits. The performance of different timing recovery schemes will be compared in terms of the root mean square (RMS) timing error, $\sigma_\epsilon = \sqrt{E[(\tau_k - \hat{\tau}_k)^2]}$, where $E[\cdot]$ denotes the expectation operator, and the bit-error rate (BER).

We first compare the performance of different schemes at moderate timing offset, i.e., $\sigma_w/T = 0.7\%$, by plotting $\sigma_\epsilon/T$ performance as a function of per-bit SNRs ($E_b/N_0$'s) in decibel (dB), as depicted in Fig. 4, where the curve labeled "Trained PLL" is conventional timing recovery whose PLL has access to all correct decisions, thus serving as a lower bound for all *symbol-rate* timing recovery schemes that are based on PLL. Clearly, the bi-directional timing recovery yields lower RMS timing error than other (symbol-rate) timing recovery schemes. This might be because the bi-directional timing recovery can be viewed as *oversampled timing recovery* [7] that oversamples the readback signal by twice the symbol rate to get more timing information to perform synchronization.

Fig. 5 compares the RMS performance of different timing recovery schemes at severe timing offset, i.e., $\sigma_w/T = 1.2\%$. Again, the bi-directional timing recovery still performs better than other timing recovery schemes in terms of $\sigma_\epsilon/T$. In addition, we can see that the performance gain obtained from the bi-directional timing recovery increases as the severity of the timing offset, $\sigma_w/T$, increases (see Fig. 4 and Fig. 5). We also compare the BER performance of different timing recovery schemes at $\sigma_w/T = 1.2\%$ as depicted in Fig. 6, where the curve labeled "Perfect timing" represents the conventional timing recovery system that uses $\hat{\tau}_k = \tau_k$ to sample $y(t)$. It is evident that the bi-directional timing recovery has lower BER than conventional timing recovery. Specifically, at BER = $10^{-4}$, the bi-directional timing recovery provides a performance gain of 0.2 dB and 0.3 dB over the Trained PLL and the conventional timing recovery, respectively.

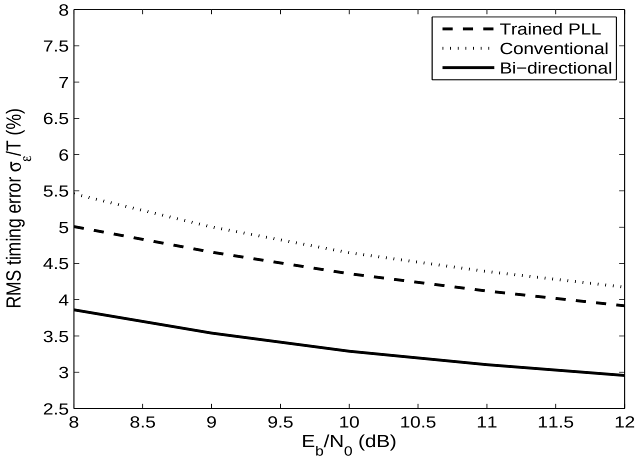

Fig. 4. RMS performance of different timing recovery schemes at $\sigma_w/T = 0.7\%$.

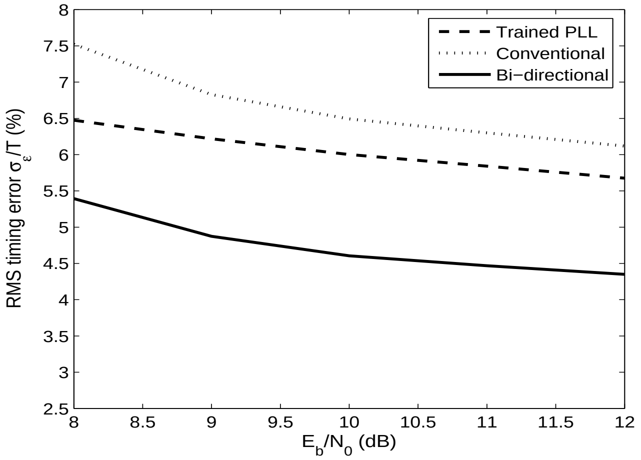

Fig. 5. RMS performance of different timing recovery schemes at $\sigma_w/T = 1.2\%$.

## V. Conclusion

In this paper, we propose the bi-directional timing recovery for perpendicular recording channels, which utilizes the conventional timing recovery to sample the readback signal both in forward direction and in backward direction. Simulation results show that the bi-directional timing recovery performs

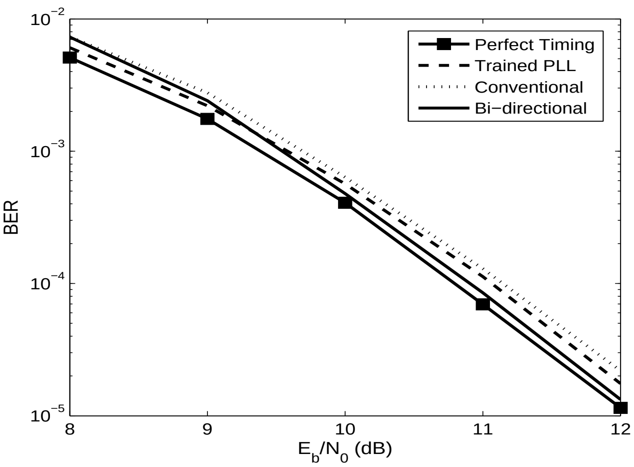

Fig. 6. BER performance of different timing recovery schemes at $\sigma_w/T = 1.2\%$.

better than the Trained PLL and the conventional timing recovery, especially when timing error is large. This might be because the bi-directional timing recovery acts as the oversampled timing recovery, which oversamples the readback signal by twice the symbol rate to get more timing information to perform synchronization. The more the timing information, the better the quality of synchronization.

## APPENDIX

In this section, we will show that the S-curve slope [1] of a PR2 channel is $40/(6T)$. The S-curve of a PR2 channel can be computed from

$$\begin{aligned} S_{\text{TED}}(\epsilon) &= E[\hat{\epsilon}_k | \epsilon, \hat{r}_{k-1} = r_{k-1}, \hat{r}_k = r_k] \\ &= E[y_k r_{k-1} - y_{k-1} r_k] \\ &= E[y_k r_{k-1}] - E[y_{k-1} r_k], \end{aligned} \tag{11}$$

where $E[\cdot]$ is the expectation operator. For a PR2 channel, the noiseless channel output is given by

$$r_k = a_k + 2a_{k-1} + a_{k-2}, \tag{12}$$

and the sampler output can be expressed as

$$\begin{aligned} y_k &= \sum_i a_i [\text{sinc}(kT - iT + \epsilon) + 2\text{sinc}(kT - T - iT + \epsilon) \\ &\quad + \text{sinc}(kT - 2T - iT + \epsilon)] \end{aligned} \tag{13}$$

where $\text{sinc}(t)$ is a sinc function. Given (12) and (13), the first term in (11) can be expressed as

$$\begin{aligned} E[y_k r_{k-1}] &= \text{sinc}(-T + \epsilon) + 4\text{sinc}(\epsilon) + 6\text{sinc}(T + \epsilon) \\ &\quad + 4\text{sinc}(2T + \epsilon) + \text{sinc}(3T + \epsilon), \end{aligned} \tag{14}$$

and the second term in (11) can be written as

$$\begin{aligned} E[y_{k-1} r_k] &= \text{sinc}(T + \epsilon) + 4\text{sinc}(\epsilon) + 6\text{sinc}(-T + \epsilon) \\ &\quad + 4\text{sinc}(-2T + \epsilon) + \text{sinc}(-3T + \epsilon). \end{aligned} \tag{15}$$

Substituting (14) and (15) into (11) yields

$$\begin{aligned} S_{\text{TED}}(\epsilon) &= \frac{\sin(\epsilon\pi/T)}{3\pi + \epsilon\pi/T} - 4\frac{\sin(\epsilon\pi/T)}{2\pi + \epsilon\pi/T} \\ &\quad + 5\frac{\sin(\epsilon\pi/T)}{\pi + \epsilon\pi/T} - 5\frac{\sin(\epsilon\pi/T)}{-\pi + \epsilon\pi/T} \\ &\quad + 4\frac{\sin(\epsilon\pi/T)}{-2\pi + \epsilon\pi/T} - \frac{\sin(\epsilon\pi/T)}{-3\pi + \epsilon\pi/T} \end{aligned} \tag{16}$$

Assuming that $\epsilon$ is very small, (16) can be rewritten as

$$\begin{aligned} S_{\text{TED}}(\epsilon) &\approx \frac{\sin(\epsilon\pi/T)}{3\pi} - 4\frac{\sin(\epsilon\pi/T)}{2\pi} \\ &\quad + 5\frac{\sin(\epsilon\pi/T)}{\pi} + 5\frac{\sin(\epsilon\pi/T)}{\pi} \\ &\quad + 4\frac{\sin(\epsilon\pi/T)}{-2\pi} + \frac{\sin(\epsilon\pi/T)}{3\pi} \\ &= \frac{40}{6\pi} \sin\left(\frac{\epsilon\pi}{T}\right). \end{aligned} \tag{17}$$

The S-curve slope can be obtained by differentiating (17) with respect to $\epsilon$, i.e.,

$$\frac{\partial S_{\text{TED}}(\epsilon)}{\partial \epsilon} = \frac{\pi}{T} \frac{40}{6\pi} \cos\left(\frac{\epsilon\pi}{T}\right). \tag{18}$$

Then, the S-curve slope of a PR2 channel can be obtained by setting $\epsilon = 0$ in (18), i.e.,

$$\frac{\partial S_{\text{TED}}(\epsilon)}{\partial \epsilon} = \frac{40}{6T}. \tag{19}$$

As a consequence, the estimated timing error $\hat{\epsilon}_k$ must be scaled by $40/(6T)$ so as to make the S-curve slope to be one at the origin. That is the $\hat{\epsilon}_k$ must be computed from

$$\hat{\epsilon}_k = \frac{6T}{40} [y_k \hat{r}_{k-1} - y_{k-1} \hat{r}_k], \tag{20}$$

as given in (4).

## ACKNOWLEDGMENT

This work was supported by National Electronics and Computer Technology Center (NECTEC) and I/U CRC in Data Storage Technology and Applications (D*STAR) under grant HDDA50-001D.

## REFERENCES

[1] J. W. M. Bergmans, *Digital baseband transmission and recording*. Boston/London/Dordrecht: Kluwer Academic Publishers, 1996.

[2] S. Raghavan and H. K. Thapar, "Feed-forward timing recovery for digit magnetic recording," in *Proc. of ICC1991*, pp. 794 – 498, 1991.

[3] F. Dolivo, W. Schott, and G. Ungerboeck, "Fast timing recovery for partial-response signaling system," in *Proc. of ICC1989*, Boston, MA, June 1989.

[4] A. N. Andrea, U. Mengali, and G. M. Vitetta, "Approximate ML decoding of coded PSK with no explicit carrier phase reference," *IEEE Trans. Commun.*, vol. 42, pp. 1033 – 1039, Feb/Mar/Apr 1994.

[5] K. H. Mueller and M. Muller, "Timing recovery in digital synchronous data receivers," *IEEE Trans. Commun.*, vol. 24, no. 5, pp. 516 – 531, May 1976.

[6] G. D. Forney, "Maximum-likelihood sequence estimation of digital sequences in the presence of intersymbol interference," *IEEE Trans. Inform. Theory*, vol. IT-18, no. 3, pp. 363 – 378, May 1972.

[7] P. Kovintavewat, M. F. Erden, E. M. Kurtas, and J. R. Barry,"Oversampled Timing Recovery for Magnetic Recording Channels," in *Proc. of the IEEE International Conference on Magnetics (Intermag) 2003*, pp. DT-06, 2003.