An Adaptive Minimum-Frame-Error Rate Detector for Magnetic Recording

Shanwei Shi and John R. Barry, Senior Member, IEEE
School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA 30332 USA

A magnetic recording read channel has numerous parameters that must be carefully tuned for best performance; these include not only the equalizer coefficients but also any parameters inside the soft-output detector, some of which may be pattern dependent, including signal levels, predictor coefficients, and residual noise variances. Conventional tuning strategies based on a minimum-mean-squared error criterion are not optimal in terms of frame-error rate and ultimately areal density. Here we propose a strategy for optimizing the parameters with the aim of minimizing the frame-error rate after error-control decoding. The proposed strategy exploits the close connection between the frame-error rate and the gap between the two curves in an extrinsic information transfer chart. A stochastic gradient algorithm applied to a cost function that quantifies this gap leads to our proposed adaptive minimum-frame-error rate (AMFER) algorithm for adapting the equalizer and detector parameters. Numerical results based on a quasi-micromagnetic simulated channel show that the AMFER parameters can reduce the frame-error rate by more than two orders of magnitude, leading to a 7% gain in areal density over conventional MMSE parameters.

Index Terms—Soft-output detection, turbo processing, extrinsic information transfer (EXIT) chart.

I. INTRODUCTION

The exploding demand for cloud storage [1] is motivating a push for higher areal densities, with narrower track pitches and shorter bit lengths. The resulting increase in interference and media noise requires improvements in read channel signal processing to keep pace. A widely used technique for mitigating media noise is pattern-dependent noise prediction (PDNP) [2, 3]; in this approach, each bit pattern has its own set of parameters for computing a branch metric. PDNP was extended to multitrack detection in [4].

The traditional strategy for choosing the read channel parameters (including the signal levels, noise predictor coefficients, and residual variances) is based on a minimum mean-square-error (MMSE) criterion, in part because it guarantees a closed-form solution and a single local minimum. However, MSE is not as relevant to the end user as other metrics such as bit-error rate (BER), frame-error rate (FER), and areal density. The adaptive minimum-bit-error rate (AMBER) algorithm for tuning the parameters to minimize BER instead of MSE was recently proposed [5, 6]; results showed that AMBER can provide a 20% decrease in BER and an 8% increase in areal density. In [7, 8], the parameters to reduce BER and computation time are estimated via deep neural networks. The BER after detection, while more relevant than MSE, is still not as relevant to the end user as the FER after error-control decoding, because the FER is what ultimately determines areal density. In a state-of-the-art read channel, a soft-output detector and a soft-input error-control decoder work iteratively in a turbo fashion with soft information transferring between them. The soft-output detector is typically either the BCJR algorithm [9] or the soft-output Viterbi algorithm (SOVA) [10].

Techniques for analyzing the convergence of iterative decoding schemes have been widely studied. Density evolution for low-density parity-check (LDPC) codes was developed by tracking the probability distributions of extrinsic log-likelihood ratios (LLR’s) [11]. This tracking is simplified considerably when the probability density function (PDF) depends on only a single parameter. Other single-parameter metrics include the fidelity of [12] and the mutual information of [13]. In [13], Ten Brink developed the extrinsic information transfer (EXIT) chart to visualize the evolution of LLR’s and analyze the convergence behavior. Prior work regarding EXIT charts has been limited to the design of error-control codes [14]–[16], not detectors.

In this paper, we propose a cost function for optimizing the detector parameters based on an EXIT chart. By choosing detector parameters to minimize this cost function, the FER can be dramatically reduced. We develop an adaptive algorithm for tuning the parameters of the equalizer and detector aiming to minimize FER. We apply the algorithm to a SOVA detector, and test it on both a simple intersymbol interference (ISI) channel and a more realistic set of simulated magnetic recording channel waveforms.

This paper is organized as follows. In Section II, we review EXIT charts. In Section III, we propose the minimum-FER tuning strategy and the AMFER algorithm. In Section IV, we evaluate the performance of the proposed algorithm, and compare it to traditional MMSE parameters. In Section V, we summarize this paper.

II. THE EXIT CHART

We consider the magnetic recording read channel illustrated in Fig. 1 [17]. A vector r of readback waveform samples is fed to an equalizer, producing a vector y of equalizer outputs. An a posteriori probability (APP) detector has two inputs: the equalizer output y, and a vector \( \lambda_2 = [\lambda_{2,0}, ..., \lambda_{2,N-1}]^T \) of a priori LLR’s about the \( N \) written coded bits \( a = [a_0, ..., a_{N-1}]^T \), where \( \lambda_{2,k} = \ln(\frac{P(a_k=1)}{P(a_k=0)}) \). Based on these inputs, the APP detector estimates the vector \( L = [L_0, ..., L_{N-1}]^T \) of APP LLR’s about the written coded bits, where \( L_k = \ln(\frac{P(a_k=1|y)}{P(a_k=0|y)}) \). The difference \( \lambda_1 = L - \lambda_2 \) is a vector of extrinsic LLR’s, 0018-9464 (c) 2021 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.
which are fed as a priori information to the LDPC decoder. In a symmetric fashion, the LDPC decoder estimates the APP LLR’s about the coded bits, which after subtracting the decoder a priori information leads to the vector $\lambda_2$ of extrinsic information from the decoder. As shown in the figure, $\lambda_2$ is then interpreted as a priori information by the detector in the next iteration. The detector and decoder thus work in an iterative (turbo) fashion until a coded frame is successfully decoded or a maximum iteration number is reached.

As indicated in Fig. 1, we use $\Theta$ to denote all of the parameters to be optimized, including the equalizer coefficients and any parameters within the detector. For simplicity, Fig. 1 omits any interleavers that might be used to prevent burst errors.

A Gaussian distribution is characterized by two parameters, mean and variance. A Gaussian distribution is said to be consistent when the variance is twice the mean, and is thus characterized by a single parameter. Likewise, the a priori information $\lambda_2$ is said to be consistent when the random variables $\{z_{2,k} = \lambda_{2,k}a_k\}$ are i.i.d. $N(\mu_2, 2\mu_2)$ for some parameter $\mu_2$. These random variables will be neither consistent nor independent in general; nevertheless (as recognized in [13]) they can be roughly approximated as such, and we will see that optimizing the detector is dramatically simplified when the a priori information coming from the decoder is assumed to be consistent.

The quality of the a priori LLR’s is captured by the mutual information $I_2 \triangleq I(\{\lambda_{2,k}\}; \{a_k\})$ between the written bits and the a priori LLR’s [13], which (under the consistent Gaussian approximation) reduces to (see the Appendix):

$$I_2 = 1 - E(\log_2(1 + e^{-Z_2})),$$

where $Z_2 \sim N(\mu_2, 2\mu_2)$.

The mutual information $I_1 \triangleq I(\{\lambda_{1,k}\}; \{a_k\})$ after the detector will depend in a predictable way on the mutual information $I_2$ after decoding. Let

$$I_1 = T_1(I_2)$$

denote the function that relates $I_2$ to $I_1$; it is an increasing function that approaches $T_1(I_2) \to 1$ as $I_2 \to 1$. Similarly, let $T_2$ denote the function that relates $I_1$ to $I_2$:

$$I_2 = T_2(I_1).$$

The $T_1(\cdot)$ and $T_2(\cdot)$ functions are the so-called extrinsic information transfer (EXIT) functions. Given readback waveforms, the shape of $T_1$ is determined by the parameters of the equalizer and detector. Closed-form expressions for these transfer functions are unknown, they must be measured empirically. We simplify the calculation of $T_2$ by implementing a time-averaged version of (1), which strictly speaking only applies when $\{a_k:\lambda_{2,k}\}$ are i.i.d. consistent Gaussian, even when they are not:

$$I_2 = 1 - \frac{1}{N} \sum_{k=0}^{N-1} \log_2(1 + e^{-a_k\lambda_{2,k}}).$$

Likewise, the mutual information between the extrinsic LLR’s after the detector and the written bits can be estimated using:

$$I_1 = 1 - \frac{1}{N} \sum_{k=0}^{N-1} \log_2(1 + e^{-a_k\lambda_{1,k}}).$$

A chart showing both $T_1$ and $T_2$ is known as an EXIT chart, and is a powerful tool for understanding convergence of iterative detectors. We illustrate an EXIT chart with an example.

**Example 1.** Consider an ISI channel $H(z) = 0.5 + z^{-1} + 0.5z^{-2}$ with AWGN, and suppose the written bits are coded by a rate-8/9 LDPC code of length 16200 from DVB-S2 [18]. A five-tap equalizer and a two-tap partial-response monic target are jointly chosen according to the MMSE criterion, followed by a two-state SOVA detector. In Fig. 2 we plot the transfer function $T_1(I_2)$ versus $I_2$ for SNR = $\sum_k h_k^2/(2\sigma_n^2)$ values of 6.7 dB, 6.3 dB, and 5.9 dB. Each curve is found by sweeping through all possible $I_2$ values, or equivalently through all possible $\mu_2$ values, and for each one generate an artificial consistent Gaussian a priori vector $\lambda_2 = \mu_2 a + w_2$, where the components of $w_2$ are i.i.d. $N(0, 2\mu_2)$. After SOVA uses $\lambda_2$ as a priori information to generate $L$, the extrinsic information $\lambda_1 = L - \lambda_2$ is used to estimate $I_1$ via (5). Improved estimates are found by averaging these $I_1$ values over repeated trials. Also shown in the figure is the transfer curve $T_2^{-1}(I_2)$ for the LDPC decoder, which is independent of SNR and is found in a symmetric way. For each possible $\mu_1$ generate a consistent Gaussian a priori vector $\lambda_1 = \mu_1 a + w_1$, where the components of $w_1$ are i.i.d. $N(0, 2\mu_1)$. The extrinsic information $\lambda_2$ is achieved by subtracting a priori information $\lambda_1$ from the decoder outputs and is used to estimate $I_2$ via (4). As shown in the figure, the $T_1$ curve drops lower as the SNR decreases, and the two curves intersect when SNR = 5.9 dB. The green dashed staircase curve shows a sample decoding trajectory for the case when SNR = 6.7 dB. The trajectory starts at $(0, 0)$, bounces between the two transfer functions, and stops near $(1, 1)$ after the decoding is successful.

**III. ADAPTIVE MINIMUM-FRAME-ERROR RATE ALGORITHM**

A direct optimization of the parameters $\Theta$ to minimize FER would require an analytical expression for FER in terms of the parameters to be optimized; such an expression is unknown and likely to be unwieldy even if it were known. Instead, we propose to indirectly optimize FER by optimizing the EXIT chart. It has been shown in [13] that when the curves in an EXIT chart intersect, there is a high probability of decoding failure. In contrast, a clear tunnel between the two curves means the decoding is very likely to be successful. The wider the tunnel is, the faster the convergence will be.
Decoding success is not so much dependent on the entire shape of the two curves in an EXIT chart, but is instead determined largely by what the two curves look like when they are closest to each other. Motivated by this observation, and with the understanding that a larger gap between the two curves leads to a higher probability of decoding successfully and faster decoding, we propose to choose the parameters \( \Theta \) so as to maximize the gap between the two curves at a particular value \( I_2^* \) for \( I_2 \), where \( I_2^* \) is chosen in the “bottleneck” region of the chart where the two curves are closest. In the example of Fig. 2, a value of \( I_2^* \) around 0.2 would be an appropriate choice when SNR = 5.9 dB.

Since only the \( T_1 \) curve depends on \( \Theta \), maximizing the gap is equivalent to maximizing \( T_1 \). Exploiting (5), we propose to choose \( \Theta \) to minimize the following cost function:

\[
J(\Theta) = \frac{1}{N} \sum_{k=0}^{N-1} \log_2(1 + e^{-a_k \lambda_{1,k}^*}),
\]

where \( \{\lambda_{1,k}^*\} \) are the extrinsic LLR’s produced by the detector when it is fed with a consistent Gaussian a priori vector \( \lambda_2 = \mu_2^T a + w_2 \), where the components of \( w_2 \) are i.i.d. \( N(0,2\mu_2^2) \), where \( \mu_2 \) is the value of \( \mu_2 \) that leads to \( I_2^* \) in (1).

As a result, the cost function does not depend on any features of the decoder or of the code itself. The proposed cost function requires knowledge of the written bits, which can be viewed as a form of training.

By either enlarging the bottleneck or opening a tunnel in the EXIT chart, we enhance the probability that the decoding trajectory passes through the bottleneck region. In the end, the FER will decrease. We should point out that maximizing \( T_1 \) at one point \( I_2 \) may cause the mutual information to degrade at other values of \( I_2 \); if this effect is undesirable one could swap different parameter sets for different \( I_2 \) values, so that the detector parameters would change from one iteration to the next.

Applying the stochastic gradient algorithm to \( J(\Theta) \) leads to the adaptive minimum-frame-error rate (AMFER) algorithm for adapting the parameters \( \Theta \):

\[
\Theta_{k+1} = \Theta_k + \frac{\alpha a_k}{1 + e^{a_k \lambda_{1,k}^*}} \nabla \Theta \lambda_{1,k}^*,
\]

where \( \alpha \) is the step size. Note the factor \( \alpha a_k/(1 + e^{a_k \lambda_{1,k}^*}) \) in (7) can be interpreted as a time-varying step size that is negligible when \( a_k \lambda_{1,k}^* \) is large; only when \( a_k \lambda_{1,k}^* \) is negative or small will the parameters change significantly.

The AMFER algorithm of (7) is general and can be applied to a wide range of detector and equalizer architectures. Here we present a concrete example based on a 2\( ^n \)-state SOVA detector, where the parameters \( \Theta \) to be optimized are the equalizer coefficients \( c \), the noise standard deviation \( \sigma \), and the signal levels \( s(a_k) \) associated with each bit pattern \( a_k = [a_k, \ldots, a_{k-\mu}]^T \). SOVA estimates the extrinsic LLR for \( a_k \) using:

\[
\lambda_{1,k}^* = \hat{a}_k \Delta_k - \lambda_{2,k}^*,
\]

where \( \hat{a}_k \) is the bit chosen by the detector, \( \Delta_k \) is chosen from a path metric margin \( M_k \) after time \( k \) (see Algorithm 1), and where the path metric margin \( M_k \) is the difference between the survivor path metric and the competing path metric:

\[
M_k = \sum_{i=0}^{\ell_k-1} -\log P(\hat{a}_{k-i} = 1) + (\Theta_k + a_k) - \sum_{i=0}^{\ell_k-1} -\log P(\hat{a}_{k-i} = 0) + (\Theta_k + a_k)
\]

\[
= \sum_{i=0}^{\ell_k-1} \lambda_{1,k}^* - 2(\Theta_k + a_k) + \frac{\sigma^2}{2} (2c^T \Theta_k - s(\hat{a}_{k-i}) - s(\hat{a}_{k-i}))
\]

where \( \lambda_{1,k}^* \) is the separation length of the two paths, and \( \hat{a}_k \) is the competing bit. Plugging (9) and (8) into the AMFER algorithm (7), along with the relationship between \( M_k \) and \( \Delta_k \) described in Algorithm 1, leads to the following update equations for the parameters \( \Theta \):

\[
c_{k+1} = c_k + \frac{a_k \hat{a}_k}{1 + e^{a_k \lambda_{1,k}^*}} \sum_{i=0}^{\ell_k-1} \sigma^2 (s(\hat{a}_{k-i}) - s(\hat{a}_{k-i}))
\]

\[
\sigma_{k+1} = \sigma_k + \frac{a_k \hat{a}_k}{1 + e^{a_k \lambda_{1,k}^*}} \sum_{i=0}^{\ell_k-1} \sigma^2 (2c^T \Theta_k - s(\hat{a}_{k-i}) - s(\hat{a}_{k-i}))
\]

for each \( i \in \{0, 1, \ldots, \ell_k - 1\} \):

\[
s(\hat{a}_{k-i})_{k+1} = s(\hat{a}_{k-i})_{k} + \frac{a_k \hat{a}_k}{1 + e^{a_k \lambda_{1,k}^*}} \frac{c^T \Theta_k - s(\hat{a}_{k-i})}{\sigma^2}
\]

\[
s(\hat{a}_{k-i})_{k+1} = s(\hat{a}_{k-i})_{k} - \frac{a_k \hat{a}_k}{1 + e^{a_k \lambda_{1,k}^*}} \frac{c^T \Theta_k - s(\hat{a}_{k-i})}{\sigma^2}
\]

A common modification in iterative detectors (see e.g. [7]) is to introduce an extra parameter that scales the LLR’s after the detector and before the decoder. Instead of choosing this parameter based on an ad hoc search based on empirical results, as is commonly done, it can be folded into \( \Theta \) and optimized by AMFER.
The pseudocode of a SOVA detector whose parameters are adapted according to the proposed AMFER algorithm is shown in Algorithm 1. Code lines 5 to 18 are the conventional SOVA algorithm. Lines 19 to 20 implement the AMFER shown in Algorithm 1. Code lines 5 to 18 are the conventional information and the step size is $\alpha = 10^{-5}$. The bottleneck point we try to break through is $I_2 = 0.2$.

In Fig. 3 we compare MMSE and AMFER EXIT charts, when SNR $= 5.9$ dB. The red curve shows $T_1$ with AMFER parameters, while the orange curve shows $T_1$ with MMSE parameters. The blue curve shows $T_2^{-1}$ for the LDPC decoder, which is independent of the parameters. From the figure we can see that in the case of MMSE parameters, the two curves intersect. On the other hand, in the case of AMFER parameters, there is a distinct gap near $I_2 = 0.2$ between the $T_1$ curve and the $T_2^{-1}$ curve. We also observe that although the AMFER curve is not superior to the MMSE curve when $I_2 > 0.6$, the ultimate FER is dramatically better; the FER with AMFER is 0.068, as opposed to an FER of 0.605 with MMSE.

In Fig. 4, we plot the FER versus SNR for both the MMSE and AMFER parameters. The error bars indicate the 95% confident interval. At around FER of $10^{-5}$, the AMFER parameters outperform MMSE parameters by 0.4 dB.
B. Ehime waveforms

We tested our algorithm on simulated waveforms provided by Ehime University, which were produced by realistic head fields and a Voronoi medium with Stoner-Wohlfarth switching [19]. Five consecutive tracks were written in a shingled fashion. In each track, there are 40950 independent pseudorandom bits sequence and 128 bits preamble and postamble, respectively. A total of 900 readback waveforms were produced with a fixed bit length of 7.3 nm, track pitches from 16.1 nm to 26.1 nm in 2 nm increments, reader widths from 70% to 145% in 15% increments (relative to a nominal reader width, whose full-width at half-maximum (FWHM) of the reader sensitivity function is 20.8 nm), and 25 reader positions in one-eight of a track pitch increments, spanning from the second to the fourth tracks. The readback waveforms from different tracks were perfectly synchronized. Despite the fact that the waveforms were not created using an error-control encoder, we can still test the turbo detector of Fig. 1 through the use of coset leaders; an uncoded block was not created using an error-control encoder, we can still test the turbo detector of Fig. 1 through the use of coset leaders; an uncoded block can be interpreted as a codeword in the coset code \( c \oplus C \) for any linear code \( C \). We also consider the written bits are interleaved and thus an interleaver and a de-interleaver are applied in the system.

We used the first 16200 bits for training the AMFER parameters, and the following 24999 bits to measure FER. Because the codeword length 16200 is much shorter than 24999, we are able to test multiple codewords from a single waveform by looking at different segments of the waveform. In particular, we consider 8800 consecutive bits as different starting locations for 8800 different codewords, and these frames are used for estimating FER. To limit any correlation between consecutive frames, we apply independent electronic noise (with an electronic SNR value of \( \text{SNR}_e = 24.6 \text{ dB} \)) to each frame.

We test the AMFER algorithm in two scenarios. For the first scenario, the detector is SOVA without any pattern-dependent noise predictor. We consider the case of ten equalizer coefficients and four signal levels. For the second scenario, the detector is SOVA with PDNP. We set ten equalizer coefficients and eight patterns. Each pattern has three parameters (one signal level, one residual noise variance and one noise predictor coefficient). The update equations can be obtained by plugging the PDNP path metric [2, 3] into (8) and applying AMFER algorithm. The AMFER parameters are initialized with MMSE parameters and \( I_2 = 0.3 \).

We compare the EXIT charts for different schemes in Fig. 5, assuming the track pitch is 24.1 nm and a centered reader with 70% width. The dashed curves are for the detectors without PDNP, while the solid curves are for the PDNP detectors. The MMSE curves are orange, while the AMFER curves are red. Without PDNP, the MMSE detector achieves FER = 0.937, while the AMFER detector achieves FER = 0.212, an improvement by more than a factor of 4. With PDNP, the MMSE detector achieves FER = 0.805, while the AMFER detector achieves FER = 0.009, an improvement by more than a factor of 89.

The areal density advantage of the AMFER algorithm is illustrated in Fig. 6, where we plot FER versus track pitch for the case of a centered reader with 70% width. The legend in this figure is as same as that in Fig. 5. The horizontal distance between two curves translates to areal density gain at a given value of FER. From the figure we see that the areal density advantage of AMFER over MMSE is roughly 5% without PDNP, and it is roughly 7% with PDNP.

V. Conclusion

In this paper we propose a strategy for adapting the parameters of an equalizer and a soft-output detector so as to minimize the FER. The AMFER algorithm is general and applies to a wide range of detectors and read channels, including those using multiple readers and multitrack detection. For the case of single-track detection using PDNP SOVA and a single reader, numerical results using realistic readback waveforms show that the parameters found by the AMFER algorithm can lead to dramatic improvements in FER and areal density compared to MMSE parameters.
APPENDIX

DERIVATION OF (1)

The mutual information between a discrete random variable \( a \in \mathcal{A} \) and continuous random variable \( \lambda \) is

\[
I = \sum_{a \in \mathcal{A}} \int_{-\infty}^{+\infty} P(a, \lambda) \log_2 P(\lambda|a) P(\lambda) \, d\lambda.
\]

When \( P(a = -1) = P(a = 1) = 1/2 \) and \( P(\lambda|a) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(\lambda - a)^2}{2\sigma^2}} \), \( I \) reduces to:

\[
I = \sum_{a \in \{-1, 1\}} \int_{-\infty}^{+\infty} -P(a, \lambda) \log_2 \left( 1 + e^{-a\lambda} \right) \frac{1}{2} d\lambda = 1 - E(\log_2(1 + e^{-a\lambda})).
\]

ACKNOWLEDGMENT

This work was supported by IDEMA ASRC. The authors would like to thank Yasuaki Nakamura and Yoshihiro Okamoto from Ehime University, Japan, for providing the simulated waveforms, and also W. Radich, and F. Erden for their guidance and support.

REFERENCES


