Available online at www.sciencedirect.com

ScienceDirect

Journal of the Franklin Institute 345 (2008) 60–74

Journal
of The
Franklin Institute

www.elsevier.com/locate/jfranklin

# A new method for performance evaluation of bit decoding algorithms using statistics of the log likelihood ratio ☆

Ali Abedi[a,*], Amir K. Khandani[b]

[a] *Electrical and Computer Engineering Department, University of Maine, Orono, ME, USA*
[b] *Electrical and Computer Engineering Department, University of Waterloo, Waterloo, Ont., Canada*

Received 5 March 2007; accepted 19 June 2007

## Abstract

This paper presents a new method for the performance evaluation of bit decoding algorithms. The method is based on estimating the probability density function (pdf) of the bit log likelihood ratio (LLR) by using an exponential model. It is widely known that the pdf of the bit LLR is close to the normal density. The proposed approach takes advantage of this property to present an efficient algorithm for the pdf estimation. The moment matching method is combined with the maximum entropy principle to estimate the underlying parameters. We present a simple method for computing the probabilities of the point estimates for the estimated parameters, as well as for the bit error rate. The corresponding results are used to compute the number of samples that are required for a given precision of the estimated values. It is demonstrated that this method requires significantly fewer samples as compared to the conventional Monte-Carlo simulation.

*Keywords:* Bit decoding; Log likelihood ratio; Maximum entropy; Probability density function; Turbo-like code

☆ This work has been presented in part at the 22nd Queen's Biennial Symposium on Communications [1] and at the Fourth International Symposium on Turbo-Codes [2]. Financial support was provided by Communications and Information Technology Ontario (CITO) and by Natural Sciences and Engineering Research Council of Canada (NSERC).

*Corresponding author. Tel.: +1 207 5812231; fax: +1 207 5814531.
*E-mail addresses:* abedi@eece.maine.edu (A. Abedi), khandani@cst.uwaterloo.ca (A.K. Khandani).

0016-0032/$32.00 © 2007 The Franklin Institute. Published by Elsevier Ltd. All rights reserved.
doi:10.1016/j.jfranklin.2007.06.002

## 1. Introduction

In the application of channel codes, one of the most important issues is to develop an efficient method for performance evaluation, since the Monte-Carlo (MC) simulation is extremely time consuming for low bit error rate (BER) values. In 1993, a new class of channel codes, called turbo-codes, were announced [3]. They have an astonishing performance, and at the same time, allow for a simple iterative decoding method by using the reliability information produced by a bit decoding algorithm. Since then, there have been numerous efforts devoted to the performance evaluation of turbo-codes, or more generally, codes relying on iterative bit decoding algorithms (turbo-like codes). Most of these approaches derive bounds on the average performance assuming maximum likelihood (ML) decoding [4–6].

Performance evaluation of trellis codes is considered in [7] where the Bahl–Cocke–Jelinek–Raviv (BCJR) [8] algorithm is adopted to find the APP values for convolutional codes. Some researchers have considered simplified cases of analytical BER calculations. An analytical method for computing the bit error probability of a two-state convolutional code with maximum a posteriori probability (MAP) decoding is presented in [9]. An analytic expression for the exact bit error probability of the $(7,5)$ convolutional code is derived in [10]. The Pearson system of distributions is adopted in [11] to compute the error probability expectations, where moment matching is used to estimate the parameters of the model. Estimating the parameters of the generalized Gaussian probability density function (pdf) by using entropy matching is considered in [12].

Other researchers have employed the importance sampling (IS) method to improve the performance of the MC simulation by increasing the weight of the rare error events. In this method, the samples are selected from a modified distribution which concentrates the points around the rare error events. This modified distribution is obtained from the original distribution by the application of a biasing function. This ensures a variance reduction if the biasing function is appropriately selected. The Gaussian tail (GT) and Rayleigh tail (RT) biasing functions are investigated in [13]. The IS method is applied to evaluate the performance of a digital communications system with inter-symbol interference (ISI) in [14], and is extended to evaluate the performance of multi-hop satellite links in [15]. A general formulation of the IS method in probability space notation is introduced in [16]. The IS method is used in [17] to simulate the Viterbi decoder by examining the trellis structure in relation to the rare error events. A comparison between the mean translation (MT) technique and the variance scaling (VS) technique is performed in [18], where it is shown that the structure of the error regions determines the better method. Recently, Ref. [19] has revisited the IS method with the strategy to increase the rate by which the variance approaches zero, instead of reducing the variance itself.

Turbo-product codes of a small block length are simulated in [20] by partitioning the error regions and by using MT for each sub-region independently. This method becomes inefficient, as the complexity of the code increases. In the case of turbo-like codes with a large block length, the search for the appropriate biasing function will be lengthy, which renders this method even more complicated than the conventional MC simulation.

It is observed in [3,6,21] that the pdf of the bit log likelihood ratio (LLR) is nearly Gaussian. There have been some efforts on estimating this pdf using Gaussian mixture model [22,23]. Two soft decision MC simulation techniques based on probabilities or LLR values are introduced in [24], and later compared in [25]. In this article, we propose an exponential model for the pdf of the bit LLR with a polynomial in the exponent. The aforementioned model has the ability to efficiently capture the deviation of the pdf from Gaussian. We use the moments of the bit LLR to estimate the parameters of the proposed model.

This paper is organized as follows. We model the pdf of the bit LLR in Section 2. In Section 3, the maximum entropy method is applied to find the parameters of the proposed model. A method to compute the probabilities of the point estimates (PPEs) for the estimated parameters, as well as the estimated BER, is detailed in Section 4. The numerical results and conclusions are presented in Sections 5 and 6, respectively.

## 2. Modeling the pdf of the bit LLR

We consider binary input additive white Gaussian noise channels. The focus will be on turbo-like codes,[1] while the proposed method is applicable to a wide variety of codes and channels as long as the symmetry properties defined in [26] are satisfied.

A common tool to express the bit probabilities in bit decoding algorithms is based on the so-called LLR. The LLR of the $k$th bit position is defined by the following equation:

$$LLR(k) = \log \frac{P(c_k = 1|\mathbf{x})}{P(c_k = 0|\mathbf{x})}, \tag{1}$$

where $c_k$ is the value of the $k$th bit in the transmitted code-word, $\mathbf{x}$ is the received vector, and log represents the natural logarithm. Let us define the random variable $Y = LLR(k)$ with its pdf denoted as $f(y)$. It is proved in [26] that the pdf of the bit LLR is independent of the transmitted code-word, as long as the value of the bit position under consideration remains unchanged. Relying on this result, we consider the pdf of the bit LLR conditioned on sending the all-zero code-word. For the simplicity of notation, we use the notation $f(y)$ (instead of the conditional probability) to refer to this pdf. We propose the following model for the pdf of the bit LLR:

$$f(y) \simeq \exp\left(-\sum_{i=0}^{N} a_i y^i\right), \tag{2}$$

where $a_i \in \mathscr{R}$, the set of real numbers. This model can easily capture the deviation of the pdf of the bit LLR from Gaussian density. Using the consistency condition [27], it can be shown that all the odd ordered terms (except for the first term) in Eq. (2) are equal to zero [28], i.e.,

$$f(y) \simeq \exp\left(-\frac{y}{2} - \sum_{i=0}^{M} b_i y^{2i}\right). \tag{3}$$

However, we rely on the full expansion as given in Eq. (2) because the iterative turbo decoding only generates an approximation of the actual LLR values.

[1] This refers to any code that can be decoded using iterative bit decoding, or in general, belief propagation algorithm. This includes both turbo-codes and LDPC codes.

The received bit is decoded to 0 (or 1), if the corresponding LLR is negative (or positive). Therefore, the following integral simplifies the remaining BER calculation:

$$P_e \simeq \int_0^{\infty} f(y)\,\mathrm{d}y. \tag{4}$$

In the next section, we use the maximum entropy principle to find the parameters of the proposed model by using the moments of the bit LLR.

## 3. Moment matching using the maximum entropy principle

There are various methods for parameter estimation. Typically, the unknown parameters of a pdf can be found by adopting moment matching, entropy matching, or ML. In this paper, we use the moment matching method with the maximum entropy principle because it is mathematically tractable, and has been successfully implemented in a variety of applications [29]. An attractive feature of the class of distributions with the maximum entropy is that a simple iterative maximization technique can be employed to compute their parameters. The maximum entropy principle was first introduced by Jaynes [29] in 1982. Since then it has been widely used in various applications. In this method, the search, while satisfying the constraints on the moments, is limited to the pdf with the maximum entropy. For more recent discussions on this method, refer to [30,31]. We follow an approach that is similar to the one introduced in [32]. The maximum entropy density can be found by maximizing the following expression with respect to $\hat{f}(y)$:

$$\text{Maximize} \quad -\int_{-\infty}^{+\infty} \hat{f}(y)\log[\hat{f}(y)]\,\mathrm{d}y \tag{5}$$

$$\text{Subject to} \quad \hat{\mu}_i = \mu_i, \quad i = 1, 2, \ldots, N, \tag{6}$$

with

$$\mu_i = \int_{-\infty}^{+\infty} y^i f(y)\,\mathrm{d}y \tag{7}$$

and

$$\hat{\mu}_i = \int_{-\infty}^{+\infty} y^i \hat{f}(y)\,\mathrm{d}y, \tag{8}$$

$$\int_{-\infty}^{+\infty} \hat{f}(y)\,\mathrm{d}y = 1, \tag{9}$$

where $N$ is the number of moments used in the parameter estimation. This maximization problem can be solved with the Lagrange multipliers $a_k$, $k = 0, 1, \ldots, N$, by following the methods of the calculus of variations [33].

Let us define the Lagrangian as

$$\int_{-\infty}^{+\infty} \hat{f}(y)\log[\hat{f}(y)]\,\mathrm{d}y + c\int_{-\infty}^{+\infty} \hat{f}(y)\,\mathrm{d}y + \sum_{k=1}^{N} a_k \int_{-\infty}^{+\infty} y^k \hat{f}(y)\,\mathrm{d}y. \tag{10}$$

Setting the variations of Eq. (10) with respect to $\hat{f}(y)$ to zero, we have

$$\log[\hat{f}(y)] + a_0 + \sum_{k=1}^{N} a_k y^k = 0, \tag{11}$$

where $a_0 = c + 1$. Solving for $\hat{f}(y)$ results in

$$\hat{f}(y) = \exp\left(-\sum_{k=0}^{N} a_k y^k\right). \tag{12}$$

Using Eq. (9), we obtain

$$\mathrm{e}^{a_0} = \int_{-\infty}^{+\infty} \exp\left(-\sum_{k=1}^{N} a_k y^k\right) \mathrm{d}y. \tag{13}$$

If Eq. (13) is substituted for $\mathrm{e}^{a_0}$ in (12), then

$$\hat{f}(y) = \exp\left\{-\sum_{k=1}^{N} a_k y^k - \log\left[\int_{-\infty}^{+\infty} \exp\left(-\sum_{k=1}^{N} a_k z^k\right) \mathrm{d}z\right]\right\}. \tag{14}$$

The objective is to estimate the parameters $a_k$, $k = 1, \ldots, N$. As we will see later, one can estimate the parameters $a_k$, $k = 1, \ldots, N$, using the first $N$ moments of the bit LLR. In practice, the statistical estimates of the moments are used instead of the true moments. Using Eq. (8), we have

$$\hat{\mu}_i(\mathbf{a}) = \int_{-\infty}^{+\infty} y^i \exp\left\{-\sum_{k=1}^{N} a_k y^k - \log\left[\int_{-\infty}^{+\infty} \exp\left(-\sum_{k=1}^{N} a_k z^k\right) \mathrm{d}z\right]\right\} \mathrm{d}y, \quad i = 1, 2, \ldots, N, \tag{15}$$

where $\mathbf{a} = \{a_1, a_2, \ldots, a_N\}$.

Setting $\hat{\mu}_i(\mathbf{a})$ equal to the statistical estimates of the moments, we can find the unknown parameters. Since there is no closed form solution for this problem, we continue with numerical methods. Newton's iterative method [32] is employed to iteratively solve the following problem (starting from the initial point of zero):

$$G_i(\mathbf{a}) = \hat{\mu}_i(\mathbf{a}) - \mu_i \tag{16}$$

$$= \int_{-\infty}^{+\infty} (y^i - \mu_i) \exp\left\{-\sum_{k=1}^{N} a_k y^k - \log\left[\int_{-\infty}^{+\infty} \exp\left(-\sum_{k=1}^{N} a_k z^k\right) \mathrm{d}z\right]\right\} \mathrm{d}y = 0, \quad i = 1, 2, \ldots, N. \tag{17}$$

Notation $\mathbf{a}^{(r)} = \{a_1^{(r)}, a_2^{(r)}, \ldots, a_N^{(r)}\}$ is used to denote the result after $r$ iterations of the Newton–Raphson method. For the small changes $\Delta\mathbf{a}^{(r)}$ in the $\mathbf{a}^{(r)}$, we write

$$\mathbf{a}^{(r+1)} = \mathbf{a}^{(r)} + \Delta\mathbf{a}^{(r)}. \tag{18}$$

This signifies that

$$G_i(\mathbf{a}^{(r)} + \Delta\mathbf{a}^{(r)}) \simeq G_i(\mathbf{a}^{(r)}) + \sum_{k=1}^{N} \frac{\partial G_i(\mathbf{a})}{\partial a_k}\bigg|_{\mathbf{a}=\mathbf{a}^{(r)}} \Delta a_k^{(r)}, \quad i = 1, 2, \ldots, N. \tag{19}$$

Therefore, $\Delta\mathbf{a}^{(r)}$ is a solution of the following linear equation:

$$G_i(\mathbf{a}^{(r)}) = \hat{\mu}_i(\mathbf{a}^{(r)}) - \mu_i = -\sum_{k=1}^{N} \frac{\partial G_i(\mathbf{a})}{\partial a_k}\bigg|_{\mathbf{a}=\mathbf{a}^{(r)}} \Delta a_k^{(r)}, \quad i = 1, 2, \ldots, N, \tag{20}$$

where notation $\hat{\mu}_i(\mathbf{a}^{(r)})$ is employed to point out that the estimated moments are updated by replacing $\mathbf{a}^{(r)}$ in Eq. (15) after the $r$th iteration. Differentiating Eq. (17) with respect to $a_k$ yields

$$\frac{\partial G_i(\mathbf{a})}{\partial a_k}\bigg|_{\mathbf{a}=\mathbf{a}^{(r)}} = \hat{\mu}_k(\mathbf{a}^{(r)})\hat{\mu}_i(\mathbf{a}^{(r)}) - \hat{\mu}_{k+i}(\mathbf{a}^{(r)}). \tag{21}$$

The algorithm (adopted from [32]) is summarized in the following steps:

*Step* 1: Start with an initial value of zero[2] for $\mathbf{a}^{(0)} = \{a_1^{(0)}, a_2^{(0)}, \ldots, a_N^{(0)}\}$.

*Step* 2: Compute the estimated moments by replacing $\mathbf{a}^{(r)}$ into Eq. (15).

*Step* 3: Plug the estimated moments into Eqs. (20), (21) to find $\Delta\mathbf{a}^{(r)}$.

*Step* 4: Compute the new parameters $\mathbf{a}^{(r+1)} = \mathbf{a}^{(r)} + \Delta\mathbf{a}^{(r)}$.

*Step* 5: Go to Step 2, if $\|\Delta\mathbf{a}^{(r)}\| > \varepsilon$, where $\varepsilon$ is the desired precision and $\|.\|$ denotes the norm of a vector.

Using properties of the Hessian matrix, it can be shown that this is a convex maximization problem [34]. The convexity of the problem guarantees that if a stationary point is found for some finite values of $a_1, \ldots, a_N$, it must be a unique absolute minimum. However, the convexity alone does not imply that such a minimum should exist. Imposing a set of conditions on the moments can guarantee the existence of a stationary point. These conditions are derived in [34] for a special case of Hausdorff moment problem. More discussions on the convexity of the problem and existence of the solution can be found in [35].

## 4. Probabilities of the point estimates

In the following, we first present a method to compute the PPEs for the estimated parameters in terms of the covariance matrix of the estimated moments. Subsequently, we derive a relationship between the PPE on the BER and the PPE on the parameters.

### 4.1. PPE for the estimated parameters

If the moment estimator satisfies a set of mild conditions, it follows that the estimated parameters are asymptotically normal with a derivable covariance matrix [36]. This allows

[2] Note that for ranges of interest (low BER), the LLR has large values, and, consequently, the moments have large values. As the error probability reduces, the values of these moments increase; however, the rate of increase in the first moment (mean value) is larger than the rest. This causes the parameters $\mathbf{a} = \{a_1, a_2, \ldots, a_N\}$ to have small values.

for the PPE statements to be made concerning $\hat{f}(y)$. In the following, we present a method to compute the covariance matrix of the estimated parameters in terms of the covariance matrix of the moments. The covariance matrix of the moments can be computed by using the method described in Appendix A.1.

Let us assume that the iterative algorithm has converged after $r_{\max}$ iterations, implying that in Eq. (20), $\Delta a_k^{(r_{\max})} \simeq 0$, and $\hat{\mu}_i(\mathbf{a}^{(r_{\max})}) = \tilde{\mu}_i$, where $\tilde{\mu}_i$ is the estimated moment. Let us define $u_{ij}$ as the covariance of the estimated moments. Referring to Appendix A.1, we have

$$u_{ij} = \text{cov}(\tilde{\mu}_i, \tilde{\mu}_j) = \frac{1}{n}(\mu_{i+j} - \mu_i\mu_j) \simeq \frac{1}{n}(\tilde{\mu}_{i+j} - \tilde{\mu}_i\tilde{\mu}_j). \tag{22}$$

It is assumed that

$$\mu_i = E[\tilde{\mu}_i] = E[\hat{\mu}_i^{(r_{\max})}], \quad \forall i. \tag{23}$$

From Eq. (21), we have

$$h_{ik}^{(r_{\max})} = -\frac{\partial G_i(\mathbf{a})}{\partial a_k}\bigg|_{\mathbf{a}=\mathbf{a}^{(r_{\max})}} = \hat{\mu}_{k+i}(\mathbf{a}^{(r_{\max})}) - \hat{\mu}_k(\mathbf{a}^{(r_{\max})})\hat{\mu}_i(\mathbf{a}^{(r_{\max})}) = \tilde{\mu}_{k+i} - \tilde{\mu}_k\tilde{\mu}_i. \tag{24}$$

The random variations in the estimated moments, $\delta\mu_k$, $k = 1, \ldots, N$, result in random variations $\delta a_k$, $k = 1, \ldots, N$, in the computed parameters. In ranges of low BER (large LLR values), the $a_k$'s and $\delta a_k$'s are much smaller as compared to $h_{ik}$'s. This is justified noting that: (i) from Eq. (24), the $h_{ik}$'s are of the same order as the moments of LLR (large values for low BER), and (ii) $a_1$ is rather insensitive to the scaling of LLR (this statement would be exact if the pdf were Gaussian), while $a_i$'s, $i > 1$, reduce as LLR increases. Relying on this observation and applying first order approximation, from Eq. (20) we obtain

$$\delta\mu_i \simeq \sum_{k=1}^{N} h_{ik}^{(r_{\max})}\delta a_k. \tag{25}$$

We have

$$\begin{aligned} u_{ij}^{(r_{\max})} &\simeq E\left[\left(\sum_{k=1}^{N} h_{ik}^{(r_{\max})}\delta a_k\right)\left(\sum_{m=1}^{N} h_{jm}^{(r_{\max})}\delta a_m\right)\right] && (26)\\ &\simeq \sum_{k=1}^{N}\sum_{m=1}^{N} h_{ik}^{(r_{\max})} h_{jm}^{(r_{\max})} E[\delta a_k \delta a_m] && (27)\\ &= \sum_{k=1}^{N} h_{ik}^{(r_{\max})} \sum_{m=1}^{N} h_{jm}^{(r_{\max})} c_{mk}. && (28) \end{aligned}$$

In matrix notation, the following is defined:

$$\mathbf{H} = \{h_{ij}^{(r_{\max})}\}, \tag{29}$$

$$\mathbf{U} = \{u_{ij}^{(r_{\max})}\}, \tag{30}$$

and

$$\mathbf{C} = \{c_{ij}\}. \tag{31}$$

The following equations relate these matrices, where the superscript t denotes the transpose of a matrix:

$$\mathbf{U} = \mathbf{H}(\mathbf{HC})^{\mathrm{t}} = \mathbf{HC}^{\mathrm{t}}\mathbf{H}^{\mathrm{t}}, \tag{32}$$

$$\mathbf{C} = (\mathbf{H}^{-1}\mathbf{U}(\mathbf{H}^{\mathrm{t}})^{-1})^{\mathrm{t}} = \mathbf{H}^{-1}\mathbf{U}^{\mathrm{t}}(\mathbf{H}^{\mathrm{t}})^{-1}. \tag{33}$$

We use the notation $\tilde{\mathbf{U}} + \delta_{\mathbf{U}}$ to specify the small random variation of $\mathbf{U}$ around its mean value. Replacing $\tilde{\mathbf{U}} + \delta_{\mathbf{U}}$, we obtain

$$\mathbf{C} \simeq \mathbf{H}^{-1}\tilde{\mathbf{U}}^{\mathrm{t}}(\mathbf{H}^{\mathrm{t}})^{-1}, \tag{34}$$

where $\tilde{\mathbf{U}}$ is computed using the method described in Appendix A.1. Noting Eqs. (24) and (A.6), we have

$$\tilde{\mathbf{U}} = \frac{1}{n}\mathbf{H}. \tag{35}$$

This results in simplification of Eq. (34) as follows:

$$\mathbf{C} \simeq \frac{1}{n}\mathbf{H}^{-1}, \tag{36}$$

where $n$ is the number of samples used in moment estimation as described in Appendix A.1.

It should be also noted that the matrix $\tilde{\mathbf{H}}$ with elements given in Eq. (24) is symmetric, positive-definite. Consequently, from Eq. (25), a small error in the estimation of moments results in a small error in the computation of $\Delta\mathbf{a}^{(r)}$, and, consequently, in the final estimate of the model parameters.

### 4.2. PPE for the estimated BER

In the following, we use the previous results to compute the PPE on the BER. Given the threshold values $\alpha_1, \ldots, \alpha_N$ on the error of parameters, the PPE is formulated as

$$p(|\delta a_1| < \alpha_1, \ldots, |\delta a_N| < \alpha_N) = c, \tag{37}$$

where $c \in [0, 1]$ and $\delta a_i$ represents the error in the computation of the parameters. Using this notation, we can rewrite the BER integral from Eq. (4) as follows:

$$\begin{aligned} P_e &+ \Delta P_e(\delta a_1, \ldots, \delta a_N) \\ &= \int_0^\infty \exp\left[-\sum_{i=0}^{N}(a_i + \delta a_i)y^i\right] \mathrm{d}y && (38) \\ &= \int_0^\infty \exp\left(-\sum_{i=0}^{N} a_i y^i\right) \exp\left(-\sum_{i=0}^{N} \delta a_i y^i\right) \mathrm{d}y && (39) \\ &\simeq \int_0^\infty \exp\left(-\sum_{i=0}^{N} a_i y^i\right)\left(1 - \sum_{i=0}^{N} \delta a_i y^i\right) \mathrm{d}y && (40) \end{aligned}$$

$$= \int_0^\infty \exp\left(-\sum_{i=0}^N a_i y^i\right) \mathrm{d}y - \int_0^\infty \exp\left(-\sum_{i=0}^N a_i y^i\right) \sum_{i=0}^N \delta a_i y^i \,\mathrm{d}y \tag{41}$$

$$= P_e - \sum_{i=0}^N \delta a_i \int_0^\infty y^i \exp\left(-\sum_{i=0}^N a_i y^i\right) \mathrm{d}y \tag{42}$$

$$= P_e - \sum_{i=0}^N m_i \delta a_i, \tag{43}$$

where

$$m_i = \int_0^\infty y^i \exp\left(-\sum_{i=0}^N a_i y^i\right) \mathrm{d}y. \tag{44}$$

The above approximation holds as $|\sum_{i=0}^N \delta a_i y^i|$ has small values in the range of interest of the integration (where $\exp(-\sum_{i=0}^N a_i y^i)$ has significant values) (note that, as already mentioned, $a_i$'s, and consequently $\delta a_i$'s, have small values). It can be seen that $\Delta P_e$, the error in the BER estimation, may be approximated by a linear combination of $m_i$'s, which can be estimated during the procedure of the moment computation by considering the positive samples only. Recalling the PPE statement (37) for the parameters, and noting that $\Delta P_e$ is a linear combination of $\delta a_i$'s, we can present a similar statement for the BER as follows:

$$p(|\Delta P_e(\delta a_1, \ldots, \delta a_N)| < \Delta P_e(\alpha_1, \ldots, \alpha_N)) = c. \tag{45}$$

This analysis enables us to make PPE statements on the estimated BER in terms of the PPEs for the model parameters.

## 5. Numerical results

A turbo-code of length 100 and rate $\frac{1}{2}$ constructed by parallel concatenation of two recursive convolutional encoders is employed to perform the simulations. The short code is selected for simplicity of the numerical analysis. To demonstrate the possibility of applying the proposed method to larger codes, another example for a code of length 12,000 is also provided later in this section. In Table 1, variances of the BER estimations are computed for both the proposed method and MC simulations. The number of samples and the variance to the mean ratio of the BER are denoted as $n(.)$ and $v(.)$, respectively. The variance of the MC method can be computed analytically (refer to Appendix A.2), although this analysis is very complex for the proposed method and we need to estimate

Table 1
Comparison of the proposed method and the MC simulation, where the variances are computed as described in Section 5

| $E_b/N_0$ (dB) | BER | $v$(new) | $n$(new) | $v$(MC) | $n$(MC) | $G$ |
|---|---|---|---|---|---|---|
| 1 | $3.81 \times 10^{-2}$ | $6.78 \times 10^{-5}$ | $10^4$ | $9.51 \times 10^{-5}$ | $10^4$ | 1.4 |
| 2 | $4.95 \times 10^{-3}$ | $1.46 \times 10^{-5}$ | $10^4$ | $9.90 \times 10^{-5}$ | $10^4$ | 6.8 |
| 3 | $1.76 \times 10^{-4}$ | $4.95 \times 10^{-6}$ | $10^5$ | $9.99 \times 10^{-6}$ | $10^6$ | 20.2 |
| 4 | $3.51 \times 10^{-6}$ | $2.30 \times 10^{-8}$ | $10^6$ | $1.00 \times 10^{-8}$ | $10^8$ | 43.5 |

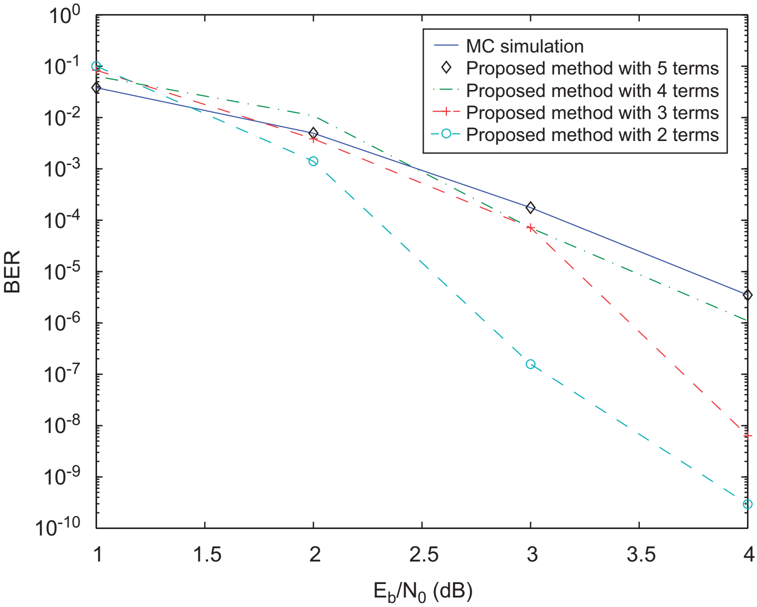

Fig. 1. BER curves for turbo-code of the length 100 and rate $\frac{1}{2}$ in comparison with the MC simulation.

the variances with numerical methods. The variance of the proposed method can be computed by repeating the experiment for $J$ times (generating $J$ independent sets of moments), and computing the variance of the resulting sequence of the BER values, denoted as $p_i$, $i = 1, \ldots, J$, as follows:

$$E[\hat{P}_e] = \frac{1}{J}\sum_{i=1}^{J} p_i \tag{46}$$

and

$$\mathrm{var}[\hat{P}_e] = -(E[\hat{P}_e])^2 + \frac{1}{J}\sum_{i=1}^{J} p_i^2. \tag{47}$$

In the computations of Table 1, we set $J = 1000$ to obtain a good approximation, and, at the same time, render the analysis feasible in terms of the required time.

We use the relative gain $G$ in Table 1 as a measure to compare the two methods. To incorporate both the variance reduction and the sample reduction advantage of the new method, and noting that $v(\mathrm{MC})$ is inversely proportional[3] to $n(\mathrm{MC})$, we define $G$ as follows:

$$G = \frac{v(\mathrm{MC})}{v(\mathrm{new})} \cdot \frac{n(\mathrm{MC})}{n(\mathrm{new})}. \tag{48}$$

Simulation results are shown in Fig. 1, where we have used the same number of samples as indicated in Table 1. It is evident that increasing the number of moments (the order of approximation) from two to five significantly improves the approximation.

[3] Refer to Appendix A.2.

Table 2
Relation between $n$ and PPE at $E_b/N_0 = 2\,\mathrm{dB}$ for the new method and the MC simulation where $\theta = \Delta P_e(\alpha|a_1|, \ldots, \alpha|a_N|)$, and $p1, p2$ are equal to $p(|\Delta P_e| < \theta)$ for new method and MC, respectively

| $n$ | $\alpha$ | $\theta$ | $p1$ | $p2$ |
|---|---|---|---|---|
| $10^4$ | 0.742 | 0.0058 | 0.97 | 0.67 |
| $10^5$ | 0.742 | 0.0058 | $\simeq 1$ | 0.70 |
| $10^6$ | 0.742 | 0.0058 | $\simeq 1$ | 0.96 |
| $10^4$ | 0.251 | 0.0020 | 0.95 | 0.66 |
| $10^5$ | 0.251 | 0.0020 | 0.98 | 0.70 |
| $10^6$ | 0.251 | 0.0020 | $\simeq 1$ | 0.96 |
| $10^4$ | 0.075 | 0.0005 | 0.93 | 0.33 |
| $10^5$ | 0.075 | 0.0005 | 0.94 | 0.68 |
| $10^6$ | 0.075 | 0.0005 | 0.97 | 0.95 |

Note that $\simeq 1$ means that the corresponding value is larger than 0.99.

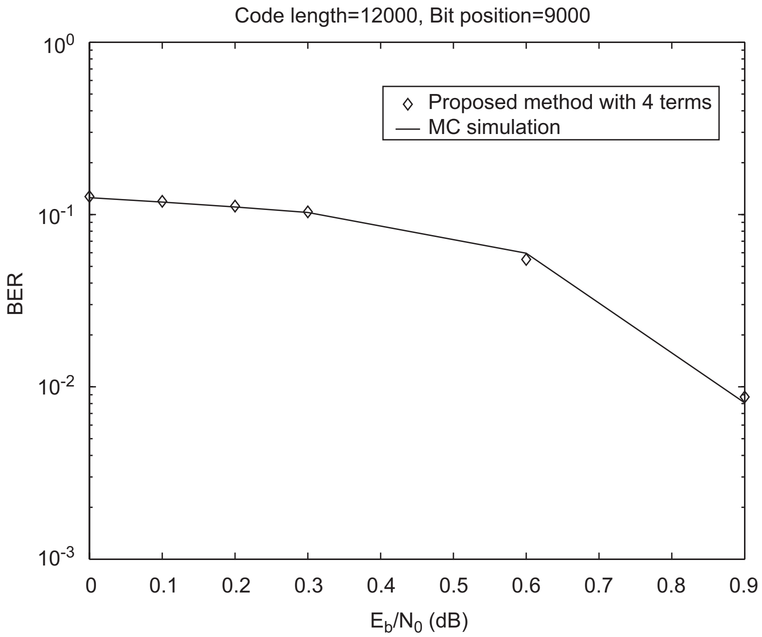

Fig. 2. BER curves for turbo-code of the length 12,000 and rate $\frac{1}{2}$ in comparison with the MC simulation.

In addition, we compute the PPEs by using the proposed method in Section 4 for this example. This PPE is closely related to $n$, the number of samples used to compute it. In Table 2, this relationship is presented for three different values of $n$ at $E_b/N_0 = 2\,\mathrm{dB}$. We compute[4] $p(|\Delta P_e| < \theta)$ for the different values of $n$, where $\theta = \Delta P_e(\alpha|a_1|, \ldots, \alpha|a_N|)$. When we compare the proposed method with the MC simulation in Tables 1 and 2, the number of samples required for the BER calculations indicates a significant reduction for our

[4] Refer to Appendix A.3 for more details on the PPE for the MC simulation.

method. It can be seen that the proposed method is more accurate than the MC simulation even by using significantly fewer samples.

Another example for a code of length 12,000 is presented in Fig. 2 to demonstrate the applicability of the proposed method to very long codes.

## 6. Concluding remarks

In this paper, we have proposed a new and low-complexity method for the performance evaluation of turbo-like codes. This method is applicable to a wide variety of codes and channels as long as a set of mild symmetry conditions, as defined in [26], are satisfied. The problem of finding the BER in high-signal-to-noise-ratio regions can be solved with this method, since the MC simulation may not be feasible. The moment matching method is employed to find the density with the maximum entropy which satisfies the moment constraints. A simple method is introduced to make PPE statements both for the parameters of the model and the BER value. It is demonstrated that significantly fewer samples, compared to those required in the MC simulation, are necessary to compute the statistical moments that are accurate enough. The complexity of the method scales with the block length of the code similar to the MC simulation with the difference that fewer samples are required in our case. In other words, the proposed method does not reduce the complexity of the iterative decoding algorithms, while it reduces the number of required samples for accurate performance evaluation.

## Appendix A

### *A.1. Covariance matrix of the moments*

The covariance matrix of the sample moments can be computed as follows:

$$\begin{aligned}
\mathrm{cov}(\tilde{\mu}_k, \tilde{\mu}_m) &= E[(\tilde{\mu}_k - \mu_k)(\tilde{\mu}_m - \mu_m)] && \text{(A.1)}\\
&= E\left[\left(\frac{1}{n}\sum_{i=1}^{n} y_i^k - \mu_k\right)\left(\frac{1}{n}\sum_{j=1}^{n} y_j^m - \mu_m\right)\right] && \text{(A.2)}\\
&= \frac{1}{n^2}\sum_{i=1}^{n}\sum_{j=1}^{n} E[y_i^k y_j^m] - \mu_k\mu_m && \text{(A.3)}\\
&= \frac{1}{n^2}[n\mu_{k+m} + (n^2 - n)\mu_k\mu_m] - \mu_k\mu_m && \text{(A.4)}\\
&= \frac{1}{n}(\mu_{k+m} - \mu_k\mu_m) && \text{(A.5)}\\
&\simeq \frac{1}{n}(\tilde{\mu}_{k+m} - \tilde{\mu}_k\tilde{\mu}_m). && \text{(A.6)}
\end{aligned}$$

This is used as $\tilde{\mathbf{U}}$ in Eq. (34).

### *A.2. Variance of MC simulation*

Let us consider the situation of transmitting a bit $b_i$ and decoding $\hat{b}_i$, for $i = 1, \ldots, n$, where $n$ is the number of samples used for the MC simulation. Let us define the following random variable:

$$e_i = \begin{cases} 1, & b_i \neq \hat{b}_i, \\ 0 & \text{otherwise.} \end{cases} \tag{A.7}$$

An error event is represented by $e_i$. We can find the BER by averaging the following random variable, $\hat{P}_e$, i.e., $P_e = E[\hat{P}_e]$:

$$\hat{P}_e = \frac{1}{n}\sum_{i=1}^{n} e_i. \tag{A.8}$$

To compute the variance of $\hat{P}_e$, we need to use the definition of variance as proved in [37]:

$$\mathrm{var}[\hat{P}_e] = \frac{P_e}{n}(1 - P_e). \tag{A.9}$$

In practice, an estimation of var$[\hat{P}_e]$ is obtained by substituting $P_e$ with $\hat{P}_e$ in (A.9).

### A.3. Computing PPEs on MC simulation

Let us define the $c \in [0, 1]$ PPE for MC, denoted as $\alpha$, as follows:

$$p(|P_e - \hat{P}_e| < \alpha) = c, \tag{A.10}$$

where $P_e, \hat{P}_e$ are the true and the estimated values of the BER. Following the same notation and definitions as in Appendix A.2, for a large $n$ and some integers $m$ and $a$, we can represent $P_e$ and $\alpha$ as $m/n$ and $a/n$, respectively. We can find the PPE for the MC simulation as follows:

$$p\left(|P_e - \hat{P}_e| < \frac{a}{n}\right) = p\left(\frac{m-a}{n} < \frac{1}{n}\sum_{i=1}^{n} e_i < \frac{m+a}{n}\right) \tag{A.11}$$

$$= p\left(\sum_{i=1}^{n} e_i < m + a\right) - p\left(\sum_{i=1}^{n} e_i \leqslant m - a\right) \tag{A.12}$$

$$= \sum_{j=0}^{m+a-1} p(j \text{ errors among } n \text{ bits}) - \sum_{j=0}^{m-a} p(j \text{ errors among } n \text{ bits}) \tag{A.13}$$

$$= \sum_{j=0}^{m+a-1} \binom{n}{j} P_e^j (1 - P_e)^{n-j} - \sum_{j=0}^{m-a} \binom{n}{j} P_e^j (1 - P_e)^{n-j} \tag{A.14}$$

$$= \sum_{j=m-a+1}^{m+a-1} \binom{n}{j} P_e^j (1 - P_e)^{n-j}. \tag{A.15}$$

In practice, an estimation of Eq. (A.11) is obtained by substituting $P_e$ with $\hat{P}_e$ in Eq. (A.15).

## References

[1] A. Abedi, A.K. Khandani, A new method for performance evaluation of turbo-codes, in: Proceedings of the 22nd Biennial Symposium on Communications, Kingston, Canada, June 2004, pp. 66–68.

[2] A. Abedi, A.K. Khandani, A new method for performance evaluation of bit decoding algorithms using statistics of the log likelihood ratio, in: The 4th International Symposium on Turbo-Codes, Munich, Germany, April 2006.

[3] C. Berrou, A. Glavieux, P. Thitimajshima, Near Shannon limit error-correcting coding and decoding: turbo-codes (1), in: Proceedings of the IEEE International Conference on Communications, Geneva, Switzerland, May 1993, pp. 1064–1070.

[4] T.M. Duman, M. Salehi, New performance bounds for turbo codes, IEEE Trans. Commun. 46 (6) (1998) 717–723.

[5] I. Sason, S. Shamai, Improved upper bounds on the ML decoding error probability of parallel and serial concatenated turbo codes via their ensemble distance spectrum, IEEE Trans. Inf. Theory 46 (1) (2000) 24–47.

[6] D. Divsalar, S. Dolinar, F. Pollara, Iterative turbo decoder analysis based on density evolution, IEEE J. Sel. Areas Commun. 19 (5) (2001) 891–907.

[7] H.A. Loeliger, A posteriori probabilities and performance evaluation of trellis codes, in: Proceedings of the IEEE International Symposium on Information Theory (ISIT 1994), Trondheim, June 1994, p. 335.

[8] L.R. Bahl, J. Cocke, F. Jelinek, J. Raviv, Optimal decoding of linear codes for minimizing symbol error rate, IEEE Trans. Inf. Theory 20 (1974) 284–287.

[9] H. Yoshikawa, Theoretical analysis of bit error probability for maximum a posteriori probability decoding, in: Proceedings of the IEEE International Symposium on Information Theory (ISIT 2003), Yokohama, Japan, 2003.

[10] M. Lentmaier, D.V. Truhachev, K.Sh. Zigangirov, Analytic expressions for the bit error probabilities of rate-$\frac{1}{2}$ memory 2 convolutional encoders, IEEE Trans. Inf. Theory 50 (2004) 1303–1311.

[11] M.R.D. Rodrigues, J.E. Mitchell, I. Darwazeh, J.J. O'Reilly, Error probability evaluation with a limited number of moments, in: Proceedings of the IEEE International Symposium on Information Theory (ISIT 2003), Yokohama, Japan, July 2003.

[12] B. Aiazzi, L. Alparone, S. Baronti, Estimation based on entropy matching for generalized gaussian pdf modeling, IEEE Signal Process. Lett. 6 (6) (1999) 138–140.

[13] N.C. Beaulieu, An investigation of Gaussian tail and Rayleigh tail density functions for importance sampling digital communication system simulation, IEEE Trans. Commun. 38 (9) (1990) 1288–1292.

[14] K.S. Shanmugam, P. Balaban, A modified Monte-Carlo simulation technique for the evaluation of error rate in digital communication systems, IEEE Trans. Commun. 28 (11) (1980) 1916–1924.

[15] M.C. Jeruchim, On the application of importance sampling to the simulation of digital satellite and multi-hop links, IEEE Trans. Commun. 32 (10) (1984) 1088–1092.

[16] Q. Wang, V.K. Bhargava, On the application of importance sampling to BER estimation in the simulation of digital communication systems, IEEE Trans. Commun. 35 (11) (1987) 1231–1233.

[17] J.S. Sadowsky, A new method for Viterbi decoder simulation using importance sampling, IEEE Trans. Commun. 38 (9) (1990) 1341–1351.

[18] J.C. Chen, D. Lu, J.S. Sadowsky, K. Yao, On importance sampling in digital communications. I. Fundamentals, IEEE J. Sel. Areas Commun. 11 (3) (1993) 289–299.

[19] J.A. Bucklew, R. Radeke, On the Monte-Carlo simulation of digital communication systems in Gaussian noise, IEEE Trans. Commun. 51 (2) (2003) 267–274.

[20] M. Ferrari, S. Bellini, Importance sampling simulation of turbo-product codes, in: Proceedings of the IEEE International Conference on Communications (ICC 2001), vol. 9, June 2001, pp. 2773–2777.

[21] H. El-Gamal, A.R. Hammons Jr., Analyzing the turbo decoder using the Gaussian approximation, IEEE Trans. Inf. Theory 47 (2) (2001) 671–686.

[22] S.Y. Chung, T.J. Richardson, R.L. Urbanke, Analysis of sum–product decoding of low-density-parity-check codes using Gaussian approximation, IEEE Trans. Inf. Theory 47 (2) (2001) 657–670.

[23] N. Letzepis, A. Grant, Bit error rate estimation for turbo decoding, in: Proceedings of the IEEE International Symposium on Information Theory (ISIT), Yokohama, Japan, June/July 2003, p. 437.

[24] I. Land, P.A. Hoeher, U. Sorger, Log-likelihood values and Monte Carlo simulation: some fundamental results, in: Proceedings of the International Symposium on Turbo Codes and Related Topics, Brest, France, September 2000.

[25] I. Land, P.A. Hoeher, New results on Monte Carlo bit error simulation based on the a posteriori log-likelihood ratio, in: Proceedings of the International Symposium on Turbo Codes and Related Topics, Brest, France, September 2003.

[26] A. Abedi, A.K. Khandani, Invariance properties of binary linear block codes over a memoryless channel with discrete input, IEEE Trans. Inf. Theory 53 (3) (2007) 1215–1218.

[27] T.J. Richardson, M.A. Shokrollahi, R.L. Urbanke, Design of capacity approaching irregular low-density parity-check codes, IEEE Trans. Inf. Theory 47 (2) (2001) 619–637.

[28] A. Abedi, A.K. Khandani, A new method for performance evaluation of turbo-like codes using statistics of the log likelihood ratio, Technical Report UW-E&CE 2005-07, University of Waterloo, Ontario, Canada.

[29] E.T. Jaynes, On the rationale of maximum-entropy methods, Proc. IEEE 70 (9) (1982) 939–952 (invited paper).

[30] M. Grendar Jr., M. Grendar, Maximum entropy: clearing up mysteries, J. Entropy 3 (2001) 58–63.

[31] P.D. Grunwald, A.P. Dawid, Game theory, maximum generalized entropy, minimum discrepancy, robust Bayes and Pythagoras, in: Information Theory Workshop, Bangalore, India, October 2002, pp. 94–97.

[32] M. Kavehrad, M. Joseph, Maximum entropy and the method of moments in performance evaluation of digital communications systems, IEEE Trans. Commun. 34 (12) (1986) 1183–1189.

[33] R. Weinstock, Calculus of Variations with Applications to Physics and Engineering, Dover, New York, 1974.

[34] L.R. Mead, N. Papanicolaou, Maximum entropy in the problem of moments, J. Math. Phys. 25 (1984) 2404–2417.

[35] N. Agmon, Y. Alhassid, R.D. Levine, An algorithm for finding the distribution of maximal entropy, J. Comput. Phys. 30 (1979) 250.

[36] S. Kullback, Information Theory and Statistics, Wiley, New York, 1959.

[37] N.C. Beaulieu, A composite importance sampling technique for digital communications system simulation, IEEE Trans. Commun. 38 (1990) 393–396.