Using the mean reliability as a design and stopping criterion for turbo codes

Land, Ingmar; Hoeher, Peter

Published in:
Proc. IEEE Inform. Theory Workshop, Cairns, Australia, Sept. 2001

Publication date:
2001

Document Version
Publisher's PDF, also known as Version of record

Using the Mean Reliability as a Design and Stopping Criterion for Turbo Codes

Ingmar Land and Peter A. Hoeher
Information and Coding Theory Lab
Faculty of Engineering
University of Kiel, Germany
e-mail: {il,ph}@tf.uni-kiel.de

Abstract — **By means of the *mean reliability* (defined as the mean of the absolute values of log-likelihood ratios), a new design of parallel concatenated "turbo" codes is proposed. This criterion allows us to describe the behavior of the constituent decoders and, furthermore, to predict the behavior of the iterative decoder for large block lengths. The mean reliability can also be used as a stopping criterion.**

I. Introduction

Parallel or serially concatenated codes are usually decoded by means of the well-known iterative "turbo" algorithm, in which soft-in soft-out decoders for the constituent codes exchange extrinsic information [1]. After a certain stopping criterion is fulfilled or a maximum number of iterations is reached, the sent bits are estimated by a hard decision of the final soft outputs.

When the logarithmic a posteriori probability (LogAPP) algorithm [2] is employed for decoding the constituent codes, the turbo decoder processes log-likelihood ratios (LLRs). The LLR $L(U) \triangleq \ln[P(U{=}{+}1)/P(U{=}{-}1)]$ of a bit $U \in \{\pm 1\}$ can be separated into the hard decision $\hat{u}$, where $\hat{u} = +1$ for $L(U) \geq 0$ and $\hat{u} = -1$ else, and the reliability $|L(U)|$ of this decision, which is given by the absolute value of the LLR, i.e., $L(U) = \hat{u} \cdot |L(U)|$.

Since the aim of iterative decoding is an improvement of the reliabilities of the hard-decided bits, we propose to characterize the quality of a sequence of LLRs by the mean absolute value of the LLRs, $\overline{|L(U)|}$, referred to as the *mean reliability* in the following.

Based on this measure, we present a method for designing the constituent codes optimized with respect to iterative decoding, and we define a stopping criterion for convergence detection. A formal reasoning for the "correctness" of this and related design criteria [3, 4, 5] is presented. As an application, we consider the optimization of partially systematic turbo codes (PSTC), which are a generalization of conventional turbo codes due to not only puncturing the parity bits, but also some of the systematic bits [6].

II. Encoder and Decoder

The encoder for a PSTC is depicted in Fig. 1. The info word $\boldsymbol{u}$ of length K is encoded by the first recursive systematic convolutional (RSC) encoder with the generator $[1; g(D)]$ onto its code word comprising both the info word $\boldsymbol{u}$ and the parity word $\boldsymbol{p}_1$. The interleaved info word is encoded by the second RSC encoder with generator $g(D)$; its code word consists only of the parity word $\boldsymbol{p}_2$. Both RSC codes are terminated by means of post-interleaver flushing[1] [7]. The resulting (mother) code word $\boldsymbol{c} = (\boldsymbol{u}, \boldsymbol{p}_1, \boldsymbol{p}_2)$ of rate $K/(3K + 3M) \approx 1/3$, where M denotes the RSC memory length, is punctured to obtain the code word $\boldsymbol{c}' = (\boldsymbol{u}', \boldsymbol{p}_1', \boldsymbol{p}_2')$ of overall rate $R > 1/3$.

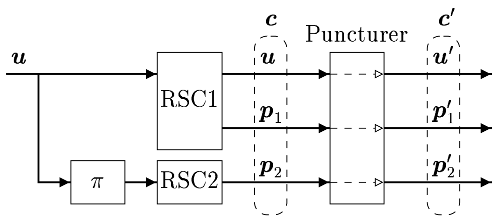

Figure 1: Encoder for a partially systematic turbo code.

The puncturing pattern is chosen such that (i) the overall code rate is R, (ii) the two parity words $\boldsymbol{p}_1$ and $\boldsymbol{p}_2$ are equally often punctured, and (iii) the ratio of the number of systematic bits after puncturing to the number before puncturing is $\rho_u \in [0; 1]$. Since the codes may not contain all of the systematic bits, they are called *partially systematic* turbo codes. $\rho_u = 1$ corresponds to the classical (systematic) turbo code [1] containing all of the systematic bits, whereas $\rho_u = 0$ corresponds to a non-systematic turbo code containing only parity bits. Smaller values of ρ_u improve the distance properties of the PSTC and therefore the high-SNR performance [6].

For decoding, the iterative decoding algorithm according to Fig. 2 is applied[2]. Each of the constituent decoders computes extrinsic LLRs for the info bits based on the channel LLRs ($L^-(U')$, $L^-(P_1')$, $L^-(P_2')$) of its respective code bits and on the extrinsic LLRs provided by the other constituent decoder as a priori LLRs: decoder 1 uses $[L_2^e(U_k)]_{[1..K]}$ to compute $[L_1^e(U_k)]_{[1..K]}$, and vice versa. After the last iteration, the sums of the extrinsic LLRs $L^+(U_k) \triangleq L_1^e(U_k) + L_2^e(U_k)$, $k = 1, 2, \ldots, K$, are hard decided.

For the examples in this paper, the parameters $g(D) = (1 + D^2)/(1 + D + D^2)$, info word length $K = 16384$, and overall code rate $R = 1/2$ were used. The code bits are

[1]After encoding of the K info bits the constituent encoders are driven back to the zero-state independently.

[2]The superscripts '−'/'+' denote prior/after decoding.

transmitted over a binary-input additive white Gaussian noise channel. The signal-to-noise ratio (SNR) per info bit is denoted as E_b/N_0. A maximum number of 60 iterations was allowed and the stopping criterion which will be introduced later was applied.

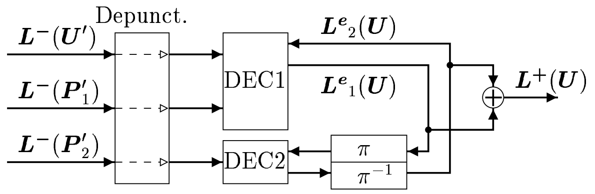

Figure 2: Decoder for a partially systematic turbo code.

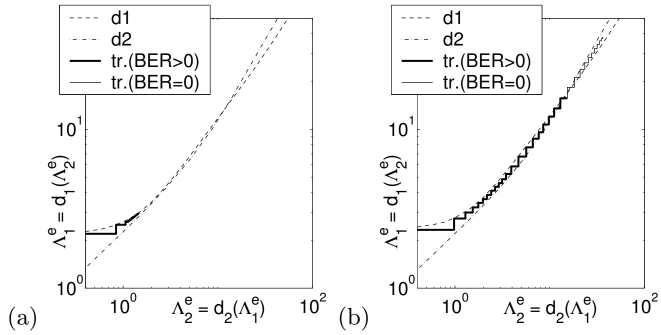

Figure 3: Decoding functions and trajectories (tr.) of a wrongly decoded word (a) ($E_b/N_0 = 0.4$ dB) and of a correctly decoded word (b) ($E_b/N_0 = 0.6$ dB), both for $\rho_u = 3/4$.

III. Mean Reliability

A single LogAPP decoder can be interpreted as a filter for LLRs [8]. In the same way, iterative decoding can be interpreted as successive filtering of extrinsic LLRs. To characterize the "filtering behavior" of a single constituent decoder, the (extrinsic) inputs are modeled as Gaussian distributed LLRs and the corresponding distribution of the extrinsic output LLRs is computed by simulation. Then, both the input and the output distribution are each represented by one variable (such as the mean value, variance, mutual information, or mean absolute value). The mapping of input values onto their corresponding output values is called the *decoding function* in the following.

In this paper, the "quality" of the inputs and outputs of a single constituent (LogAPP) decoder is measured by means of their mean reliabilities. We define the *mean reliability* Λ_i^e of the extrinsic LLRs $L_i^e(U_k)$, $k = 1, 2, \ldots, K$, computed by constituent decoder i as

$$\Lambda_i^e = \frac{1}{K} \sum_{k=1}^{K} |L_i^e(U_k)| \,,$$

where $i = 1, 2$. With this definition, the statistical input-output behavior of each constituent decoder i can be described by its decoding function $d_i(.)$, i.e.,

$$\Lambda_1^e = d_1(\Lambda_2^e) \,, \quad \Lambda_2^e = d_2(\Lambda_1^e) \,.$$

Fig. 3 shows simulation results for the PSTC defined above with $\rho_u = 3/4$ (i.e., a quarter of the systematic bits is punctured) at two different SNR. The decoding functions d_1 and d_2 of the respective constituent decoders are crossing in a point, which can be identified as the point of convergence. For illustration, the figures show also actual decoding trajectories. The parts with remaining bit errors are plotted with thick lines, whereas the error-free parts are plotted with thin lines. In our experience, a trajectory follows the way predicted by the decoding functions as long as there are remaining bit errors. A few iterations before the word becomes error-free, the trajectory starts to cross the decoding functions and converges "behind" their intersection (see Fig. 3(b)).

In related publications, the SNR [3, 4] and the mutual information [5], respectively, are used to measure the quality of the extrinsic values exchanged by the constituent decoders. In our opinion, the *mean reliability* measure is a suitable alternative due to the following reasoning:

1. The absolute values of the a posteriori LLRs (which are the outputs of a LogAPP decoder) are sufficient to estimate the bit error probability [9]. This motivates to apply a measure based on the absolute values of the LLRs.

2. The mean value μ and the variance σ^2 of Gaussian distributed LLRs fulfill the property $2\mu = \sigma^2$ [9].

   Therefore, the input LLR distribution of the constituent decoders, which is modeled as Gaussian distributed, is completely determined by μ *or* σ^2. Moreover, it is completely determined by every reversible function of μ and/or σ^2. Thus, it is sufficient to describe the distribution with any arbitrary reversible function of μ and/or σ^2, preferably one which is easy to compute (like the mean reliability).

   One the other hand, since the output LLR distribution is not Gaussian, it cannot be described by just one variable. Therefore, from this point of view the SNR [3, 4], the mutual information [5], and the mean reliability are equally meaningful. However, given these measures, the mean reliability requires the lowest computational effort.

3. The mean reliability does not depend on the info bits actually sent, and it can be computed in a real decoder. Furthermore, it can be used to define a stopping criterion.

IV. Code Design

Since the intersection of the decoding functions corresponds to the point of convergence, these curves allows us to predict the asymptotic decoding behavior of the iterative decoder for (infinitely) long code words. The computational complexity of computing the decoding functions of the constituent codes is very low compared to that of simulating the concatenated code, which involves iterative decoding. Therefore, this method can be applied as an efficient tool to investigate the convergence behavior

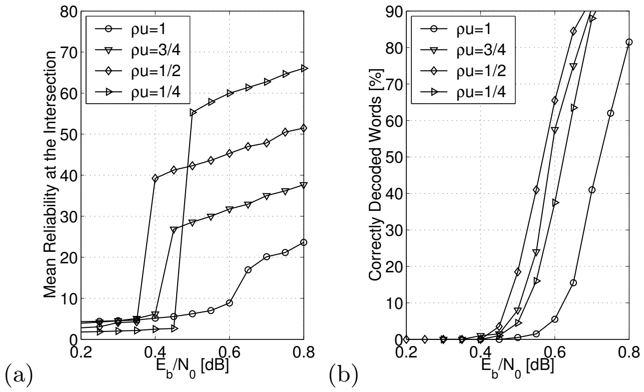

Figure 4: Sum $\Lambda_1^e + \Lambda_2^e$ of the mean extrinsic reliabilities at the intersections of the decoding functions (a) and ratio of correctly decoded words (b) vs. the SNR.

for different constituent codes or, like in our examples, for puncturing patterns according to different values of ρ_u. The main interest is on the lowest SNR at which iterative decoding of a very long code will succeed.

In Fig. 4(a), the sum $\Lambda_1^e + \Lambda_2^e$ of the mean reliabilities associated with the intersection is plotted against the SNR for several values of ρ_u. At a certain SNR the mean reliability increases very quickly from small to large values. This point can be identified with the SNR necessary for convergence or, equivalently, with the "waterfall region".

The actual amounts of correctly decoded words are depicted in Fig. 4(b). When this figure is compared to Fig. 4(a), the critical SNR values turn out to be quite similar. The remaining differences are due to two reasons: (i) the actual distributions of the extrinsic LLRs are only approximately Gaussian; (ii) the code actually used is of finite length. Nevertheless, the relations between different values of ρ_u hold also for finite word lengths.

For the design of PSTCs, two effects have to be considered: (a) smaller values of ρ_u lead to better distance properties of the code and therefore to lower error rates in the "flattening region" [6]; (b) the results of the above investigations show that the iterative decoder needs a certain amount of systematic bits to provide convergence at low SNR, i.e., the value of ρ_u should not be too small. Due to this trade-off, the value of ρ_u has to be adapted to the SNR at which the code is to be used. For best power efficiency about half of the systematic bits ($\rho_u \approx 1/2$) should be punctured.

V. Stopping Criterion

Since the mean reliability is independent of the bits actually sent, it can also be used as a stopping criterion. In our experience, after a certain number of iterations the mean reliability of the extrinsic LLRs does not significantly change if either the constituent codes are terminated by post-interleaver flushing or if none of the constituent codes is terminated (which should be avoided due to the well-known poor distance properties). For the case that only one of the codes is terminated and for the case that both codes are terminated but all of the termination bits are encoded by both encoders, the mean reliability will keep growing even for already error-free words.

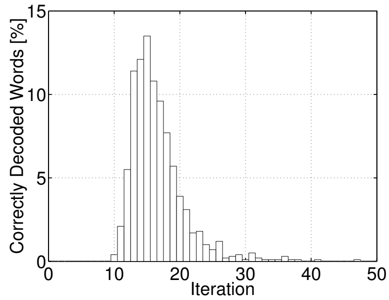

Figure 5: Percentage of words correctly decoded after each iteration vs. iteration number ($\rho_u = 1/2$, $E_b/N_0 = 0.7$ dB).

A word is considered as converged, when both mean reliabilities Λ_1^e and Λ_2^e do not change in two subsequent iterations. When this occurs, decoding is terminated. Simulations indicate that a precision of 10^{-2} is appropriate. As opposed to many other stopping criteria, the proposed one does not depend on the SNR. In our experience, PSTCs with small values of ρ_u (codes with fewer systematic bits) require more iterations than those with larger values of ρ_u.

Although most of the words are error-free after a small number of iterations, some words need up to 40 iterations or even more. As an example, the percentage of words correctly decoded within a certain iteration step is depicted in Fig. 5. Thus, a relatively high maximum number of iterations is necessary for minimizing the word error rate. At the same time, the average number of iterations will be much smaller when the proposed stopping criterion is applied.

Acknowledgments

The authors would like to thank Dr. Ulrich Sorger of the Darmstadt University of Technology, Germany, for inspiring discussions.

References

[1] C. Berrou and A. Glavieux, "Near optimum error correcting coding and decoding: Turbo-codes," *IEEE Trans. Commun.*, vol. 44, no. 10, pp. 1261–1271, Oct. 1996.

[2] P. Robertson, P. Hoeher, and E. Villebrun, "Optimal and suboptimal maximum a posteriori algorithms suitable for turbo decoding," *Europ. Trans. Telecommun.*, vol. 8, no. 2, pp. 119–125, Mar.-Apr. 1997.

[3] H. El Gamal and A.R. Hammons, Jr., "Analyzing the turbo decoder using the Gaussian approximation," *IEEE Trans. Inform. Theory*, vol. 47, no. 2, pp. 671–686, Feb. 2001.

[4] D. Divsalar, S. Dolinar, and F. Pollara, "Low complexity turbo-like codes," *Int. Symp. on Turbo Codes and Rel. Topics*, pp. 73–80, Sept. 2000.

[5] S. ten Brink, "Convergence of iterative decoding," *IEE Electronics Letters*, vol. 35, no. 13, pp. 806–808, June 1999.

[6] I. Land and P. Hoeher, "Partially systematic rate 1/2 turbo codes," *Int. Symp. on Turbo Codes and Rel. Topics*, pp. 287–290, Sept. 2000.

[7] D. Divsalar and F. Pollara, "Turbo codes for deep-space communications," *TDA Progress Report 42-120*, pp. 29–39, Feb. 1995.

[8] I. Land, P. Hoeher, and U. Sorger, "On the interpretation of the APP algorithm as an LLR filter," *IEEE Int. Symp. on Inform. Theory*, p. 415, June 2000.

[9] P. Hoeher, I. Land, and U. Sorger, "Log-likelihood values and Monte Carlo simulation - some fundamental results," *Int. Symp. on Turbo Codes and Rel. Topics*, pp. 43–46, Sept. 2000.