Code Optimization for Lossless Turbo Source Coding

Nicolas Dütsch
Institute for Communications Engineering (LNT)
Munich University of Technology (TUM)
Arcisstraße 21, D-80290 Munich, Germany
E-mail: Nicolas.Duetsch@tum.de

***Abstract*— A novel source coding scheme based on turbo codes was presented in [1]. Lossless data compression is thereby achieved by puncturing data encoded with a turbo code while checking the integrity of the reconstructed information during compression. In this paper we apply code optimization tools to serially concatenated turbo source codes. The goal of the optimization is to minimize the area of the tunnel in the modified EXIT chart as it is proportional to the gap between source coding rate and entropy. We show that compression rates close to the Shannon limit can be obtained by irregular repeat accumulate codes.**

I. Introduction

One of the most remarkable milestone in the field of channel coding during the last decades has been the introduction of turbo codes [2] and low-density parity check (LDPC) codes [3]. Their common success of achieving near-to-optimal performance lies in the use of a probability-based message-passing algorithm at the decoder. It is well known that source coding and channel coding are basically dual problems. The latter is a sphere packing problem, whereas the former is a sphere covering problem. Thus it was just a natural step to apply the above mentioned class of channel codes to source coding problems.

In [4], [5] the authors presented a lossless data compression method based on error correcting codes. They used a library of LDPC codes of different rates to form the syndrome of a source message. By iterative doping together with Belief Propagation decoding it is possible to reconstruct the original message perfectly.

Another source coding approach based on turbo codes was published in [6]. Compression was accomplished by puncturing turbo-encoded data heavily. A turbo decoder was used to fill all gaps of the punctured bits. However the drawback of this method is that only near-lossless source coding is feasible as the decoder will fail in restoring all source data if too many bits are discarded. By adjusting the puncturing rate to the result of an integrity test at the compressor lossless turbo source coding is also attainable [1].

In this paper we elaborate how to improve the compression efficiency of turbo source codes. The corresponding optimization problem basically is a curve fitting problem of transfer functions visualized in the modified extrinsic information transfer (EXIT) chart. As both characteristic curves of a classical parallel concatenated turbo code depend on the puncturing rate, curve fitting is hardly possible and compression cannot be enhanced further. Thus we restrict our thoughts to serially concatenated turbo codes with one curve being independent of the puncturing. By dint of an optimization algorithm we are able to match the component codes in order to yield compression rates close to the entropy. Specially constructed irregular repeat accumulate codes are shown to outperform previous code constructions based on the parallel concatenation of two recursive systematic convolutional component codes.

The rest of this paper is organized as follows. In section II we describe the subject of turbo source coding, characterize the application of serially concatenated turbo codes to this problem and dwell on different compression rate adjustment strategies in order to perform noiseless source coding. Section III covers modified EXIT charts and presents code construction methods for the class of irregular repeat accumulate codes. Further on an area property of the tunnel in the EXIT chart is stated. Finally, section IV shows some numerical comparisons of our proposed compression scheme with parallel concatenated turbo compression and standard compression methods.

II. Turbo Source Coding

A. Problem Statement

Let U be a binary, memoryless source drawn i.i.d. from the alphabet $\mathcal{U} = \{-1, +1\}$ with $P(U = +1) = p$. The optimal compression rate for this source is given by its entropy $H(U) = H_b(p)$, whereas $H_b(p)$ is the binary entropy function. In this section we specify how to encode a source block $\boldsymbol{U}$ of length N resulting in a code word $\boldsymbol{Y}$ of length K with its elements $Y \in \{-1, +1\}$. As long as the rate of the source code, defined as $R_{comp} = K/N$, is larger than the entropy of the source, the source block can be reconstructed perfectly (source coding theorem). In the following we state how to generate the codewords using turbo codes, how to guarantee lossless source coding, i.e. perfect reconstruction of the source block at the decompressor, and how to attain a source coding rate close to the entropy.

Throughout this paper we will use log-likelihood values with the natural logarithm. In particular we use the so called

- source state information (SSI), defined as $L_S = \log \frac{p}{1-p}$ and the
- channel state information (CSI) $L(y|x) = \log \frac{p(y|x=+1)}{p(y|x=-1)}$ $= L_C \cdot y$, whereas y is a noisy version of x.

Recall that L_C is ∞ resp. 0 for the binary erasure channel (BEC).

B. Serially Concatenated Turbo Source Codes

First we dwell on the generation of the compressed code word $\boldsymbol{Y}$. Fig. 1 illustrates the coding scheme for turbo source coding. The source block $\boldsymbol{U}$ is encoded using a serially concatenated turbo code resulting in a code word $\boldsymbol{X}$ with the $(R_{out}R_{in})^{-1}$-fold length L of the source message. Compression is now achieved by puncturing the turbo encoded bits X randomly. Thus the output of the turbo source encoder consists of K non-discarded bits forming the compressed code word $\boldsymbol{Y}$ of variable length. It is worth to mention that the interleaved code word $\boldsymbol{V}$ of length $M = NR_{out}^{-1}$ obtained by the outer encoder can be interpreted as the output of a binary memoryless source also drawn i.i.d. with entropy $H(V)$.

The decompressed source message $\hat{\boldsymbol{U}}$ is finally attained by a turbo decoder. Note that compression through puncturing is equivalent to transmission over a BEC. Thus the inner APP-decoder is fed with the source code word $\boldsymbol{Y}$ weighted with the CSI as commented in the previous subsection. According to the turbo principle both component decoders exchange extrinsic information and support each other in filling the gaps of the punctured bits. Furthermore the turbo decoder also benefits from the non-uniformly distributed source bits by utilizing this additional information in terms of the SSI.

C. Puncturing Rate Adjustment and Perfect Reconstruction

Completing the section describing turbo source coding we have a closer look at the puncturing scheme. The algorithm called *decremental redundancy* [1], which regulates the compression rate, is outlined next. Notice that this method was proposed for a parallel concatenated turbo code of rate $1/3$, i.e. the output of the turbo encoder are systematic bits, parity bits determined by the first component encoder and those calculated by the second encoder.

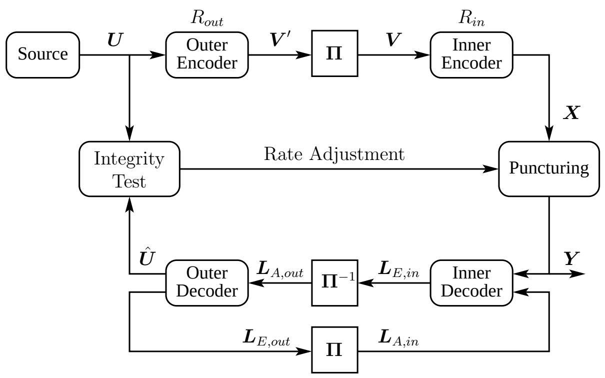

Fig. 1. Serially concatenated turbo compressor.

In order to obtain a compression rate of at least one, all systematic bits and half of the parity bits are randomly removed initially. Then the remaining non-punctured bits are used for decoding. If the integrity test is passed successfully, meaning the restored source block does not differ from the original source message, more parity bits are punctured and the turbo decoder tries to fill all gaps of the erased bits again. Redundancy is decreased in this way step by step as long as the integrity test fails for the first time, i.e. the source block can not be recovered perfectly. Finally the non-discarded parity bits of the former puncturing step, which led to successful decoding, are stored as compressed code word $\boldsymbol{Y}$.

To rebuild the source sequence at the decompressor, the randomly chosen puncturing pattern has to be saved. Accordingly, the side information is as long as the source block and thus compression is not possible. An equivalent puncturing strategy was proposed in [1]: each parity sequence is interleaved, written line by line into a square matrix and the bits are erased column by column as long as the reconstructed message is identical to the source message. Consequently, only the index of the last punctured parity segment and no longer the puncturing pattern has to be known to reconstruct the source block. For the serial concatenation we maintain the puncturing principle specified above. Only the following modification of the puncturing matrix has to be introduced which is visualized in Fig. 2. Recall that the serial turbo encoder determines a code word of length $L = N(R_{out}R_{in})^{-1}$ for each source sequence. Thus we also suggest to interleave this code word, but store it in a rectangular matrix of size $\delta(R_{out}R_{in})^{-1} \times N/\delta$. The encoded bits are erased column by column and deinterleaved before decoding. The step size of the puncturing rate accordingly is $\Delta R_{comp} = \delta(NR_{out}R_{in})^{-1}$, and can be adjusted by the design parameter δ.

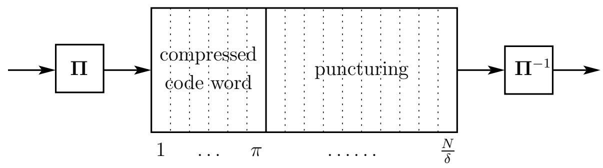

Fig. 2. Puncturing scheme to fine tune the compression rate.

Instead of starting with a completely filled puncturing matrix and remove the redundancy gradually until decoding fails for the first time, we propose to begin with puncturing index $\pi = \lfloor NH_b(p)R_{out}R_{in}\delta^{-1}\rfloor$ and increment redundancy as long as the integrity test is passed successfully for the first time. This initial puncturing index determines the minimum number of bits needed to reconstruct the source block perfectly. As a result the number of decoding trials can be reduced dramatically as the compression rate we expect is very close to the entropy of the source.

III. Code Optimization

A. Modified EXIT Charts

In order to analyze the convergence behavior of an iterative decoding scheme such as our turbo compressor we use the EXIT chart. This helpful tool is based on characteristic curves which interpret the extrinsic output information as a function of the a priori input information for each component decoder. Plotted together in one chart one can predict the convergence behavior of the turbo decoder. As such a decoder is used to verify the integrity of the reconstructed source information we are able to anticipate the compression rate of our lossless turbo source coder.

Again we assume a binary source U drawn i.i.d. from the alphabet $\mathcal{U} = \{-1, +1\}$ with $P(U = +1) = p$. Anymore we suppose an encoded sequence transmitted through a channel with probability density function $f(y|U = u)$, where y denotes the channel output. Thus we can write the mutual information between the discrete non-uniformly distributed random variable U and the continuous valued random variable Y as

$$I(U;Y) = \sum_{u=\pm 1} P(U=u) \int_{-\infty}^{+\infty} f(y|U=u) \cdot \log_2 \frac{f(y|U=u)}{p \cdot f(y|U=+1) + (1-p) \cdot f(y|U=-1)} \, dy. \quad (1)$$

When exploiting the ergodicity of the source this expression can be simplified to

$$\begin{aligned} I(U;Y) &= H(U) - \mathrm{E}\left\{\log_2(1 + e^{-u \cdot L(u|y)})\right\} \\ &\simeq H(U) - \frac{1}{N} \sum_{n=1}^{N} \log_2 \left(1 + e^{-u_n \cdot L(u_n|y_n)}\right), \quad (2) \end{aligned}$$

where the expectation is taken over all possible observations. This measure is now used to calculate the set of mutual information $I(V; L_{A,in})$, $I(V; L_{E,in})$, $I(U; L_{A,out})$ and $I(U; L_{E,out})$, which we will abbreviate with $I_{A,in}$, $I_{E,in}$, $I_{A,out}$ and $I_{E,out}$, respectively.

Now we are able to analyze the component decoders of the turbo code by exploiting their exchange of extrinsic information. Therefore we express the extrinsic mutual information as a function of the a priori mutual information $I_{E,i} = T_i(I_{A,i})$ for each component decoder i. This transfer function which is also called characteristic curve is obtained by evaluating Eq. (2) with the a priori and extrinsic L-values, respectively. The former are generated by passing a source sequence through an a priori channel which we model as a BEC with erasure probability ϵ. Notice that we must also use the SSI to calculate the mutual information.

As the extrinsic mutual information of one decoder is the a priori mutual information of the other decoder and vice versa in iterative decoding we can plot $I_{E,out} = T_{out}(I_{A,out})$ against a mirrored version of $I_{E,in} = T_{in}(I_{A,in})$. The width of the tunnel between both characteristic curves depends on the compression rate as well as on the used component decoders. If the tunnel is opened the source sequence can be reconstructed perfectly.

B. Repeat Accumulate Codes

Very simple serially concatenated codes we used for turbo compression are repeat accumulate codes. This class of codes was used by ten Brink to nearly reach the Shannon limit in channel coding [7]. As channel coding and source coding are dual problems we assumed that repeat accumulate codes are also very eligible for turbo compression and thus compression rates very close to the entropy should be obtained.

Repeat accumulate codes are composed of an outer repetition code and an inner accumulator of rate $R_{in} = 1$. The two main advantages of this code class is that on the one hand the characteristic curve of the repetition code can be calculated analytically as shown below and on the other hand the entropy of the interleaved code bits V is given by $H(V) = H(U)$. As the inner APP-decoder works on the noisy and interleaved code bits of the outer repetition code, the SSI utilized by the inner decoder is also $L_S = \log \frac{p}{1-p}$ and there is no need to recalculate it. Thus the transfer function of the inner accumulate code can be measured according to Eq. (2).

Following the ideas and derivations of [9] the a priori mutual information $I_{A,rep}$ of a rate $R_{rep} = \frac{1}{n}$ repetition code can be calculated as

$$I_{A,rep} = (1 - \epsilon) H_b(p) \,, \quad (3)$$

if we assume the a priori channel to be a BEC with erasure probability ϵ. Furthermore the extrinsic mutual information $I_{E,rep}$ of this code class is obtained by

$$I_{E,rep} = (1 - \epsilon^{n-1}) H_b(p) \,. \quad (4)$$

Substituting (3) in (4) we finally get the transfer function

$$\frac{I_{E,rep}}{H_b(p)} = T_{rep}\left(\frac{I_{A,rep}}{H_b(p)}\right) = 1 - \left(1 - \frac{I_{A,rep}}{H_b(p)}\right)^{n-1} \,. \quad (5)$$

Note that if we assume the input bits of this code to be binary and uniformly distributed we get the same results as presented in [9].

The transfer function of a rate $R_{rep} = \frac{m}{n}$ repetition code can also be derived analytically. It is easy to show that such a code is equivalent to m repetition codes of rate $R_\mu = \frac{1}{n_\mu}$ with $\sum_{\mu=1}^{m} n_\mu = n$. According to this the characteristic curve of the rate $R_{rep} = \frac{m}{n}$ repetition code is obtained by

$$\frac{I_{E,rep}}{H_b(p)} = 1 - \frac{1}{m} \sum_{\mu=1}^{m} \left(1 - \frac{I_{A,rep}}{H_b(p)}\right)^{n_\mu - 1} \,. \quad (6)$$

C. Area Property

Let us define the area under the transfer function as $\mathcal{A} = \int_0^{H_b(p)} T(I_A) \, dI_A$. From [9] it is known that this area can also be expressed by entropy measures as

$$\mathcal{A} = H_b^2(p) \left[1 - \frac{H(\boldsymbol{W}|\boldsymbol{Y})}{\sum_{i=1}^{\omega} H(W_i)}\right] \quad (7)$$

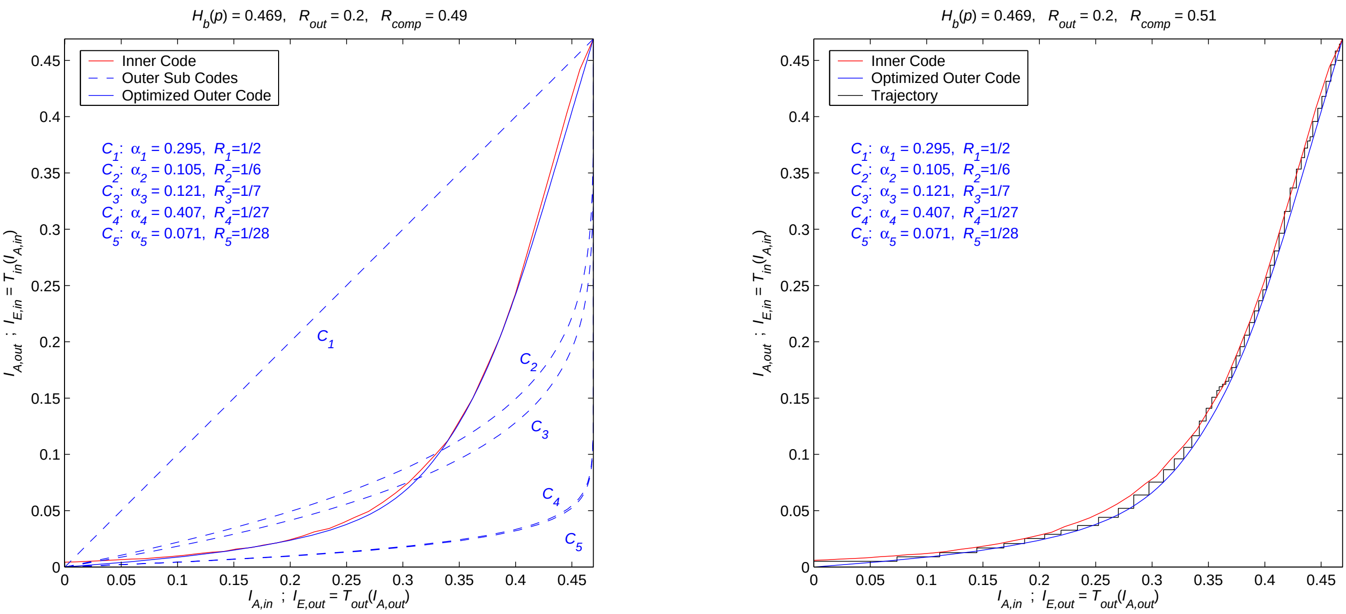

Fig. 3. EXIT charts of optimized irregular repeat accumulate codes. Lossless compression is possible at a coding rate of 0.51 as indicated by the trajectory.

with $\boldsymbol{W}$ of length ω being the input of the a priori channel and $\boldsymbol{Y}$ being the communication channel output. Eq. (7) is now applied to both component codes of the repeat accumulate compression scheme.

Assuming the outer code being a repetition code we have $\boldsymbol{W} = \boldsymbol{V}'$, $\omega = M$ and $H(\boldsymbol{W}|\boldsymbol{Y}) = H(\boldsymbol{V}')$ as no communication channel is available. Thus the area under the transfer function of this code is obtained by

$$\mathcal{A}_{out} = H_b^2(p)\,[1 - R_{out}] \; . \tag{8}$$

Note that this results is also derived by the integration of Eq. (5). In the case the inner code is a rate one accumulator we have $\boldsymbol{W} = \boldsymbol{V}$ and $\omega = M$. Because of the one-to-one mapping $\boldsymbol{V}$ can be exchanged by $\boldsymbol{X}$. Accordingly $H(\boldsymbol{W}|\boldsymbol{Y}) = H(\boldsymbol{X}|\boldsymbol{Y}) = H(\boldsymbol{V}) - I(\boldsymbol{X};\boldsymbol{Y}) = M(H(V) - I(X;Y))$. The last step follows as the elements of $\boldsymbol{V}$ are i.i.d. (see section II-B) and as the communication channel is memoryless. Finally the area is given by

$$\mathcal{A}_{in} = H_b(p) R_{out} R_{comp} \; . \tag{9}$$

It is easy to show that the area between both characteristic curves is $H_b(p)R_{out}(R_{comp} - H_b(p))$ for the regarded code class. Different from the results derived for channel coding, where the area of the tunnel is exactly the rate loss [9], this area is only proportional to the gap between entropy and compression rate. Anyway if we minimize the tunnel we should obtain compression rates close to the Shannon limit.

D. Irregular Repeat Accumulate Codes

According to the principle of irregular codes the source sequence $\boldsymbol{U}$ is separated into S fractions and each part is encoded with a different sub code $\mathcal{C}_s$ of rate R_s, $s = 1 \ldots S$. In detail the source fraction $\boldsymbol{U}_s$ of length $N_s = \alpha_s R_s M$ is mapped to an encoded sequence $\boldsymbol{V}'_s$ of length $\alpha_s M$, whereas the weighting coefficients are restricted according to

$$\sum_{s=1}^{S} \alpha_s = 1, \qquad \alpha_s \in [0, 1]. \tag{10}$$

In [8] it is shown that the code rate of the irregular code is $R_{ir} = \sum \alpha_s R_s$. Furthermore it is also indicated that the transfer function $T_{ir}(I_A)$ for the irregular code is obtained by

$$T_{ir}(I_A) = \sum_{s=1}^{S} \alpha_s \, T_s(I_A), \tag{11}$$

where $T_s(I_A)$ are the characteristic curves of the sub codes $\mathcal{C}_s$.

The goal of our work is to minimize the area of the tunnel in the EXIT chart as it is proportional to the gap between entropy and compression rate. For a given compression rate and thus a fixed inner transfer function we want to find a set of weighting coefficients α_s so that the characteristic curve of the outer irregular code matches well against the corresponding curve of the inner code. As base functions we use transfer functions of repetition codes of rate $R_{rep} = \frac{1}{n}$. The algorithm we used for this curve fitting problem minimizes the area between both transfer functions in a least square sense. Details on this optimization method are found in [8].

IV. Performance Evaluation

To evaluate the source coding rates of our turbo compression scheme and to illustrate the compression performance by means of EXIT charts, we present some numerical results. For all experiments we used a binary i.i.d. source with entropy $H_b(p) = 0.469$ bits/symbol. The block length of the source sequence was chosen to be $N = 2^{16}$. In order to measure the characteristic curves of the accumulator we simulated 10 blocks and have taken the average.

In the case of the repeat accumulate code we used the same code construction as in [7]. In comparison to the results of parallel concatenated turbo codes presented in [1] the obtained compression rates of the serially concatenated scheme are worse (not shown here). The reason is that indeed both, the repeat accumulate code and the parallel concatenated turbo code are equivalent in theory, but this serially concatenated system suffers from its tighter tunnel in the EXIT chart.

We also examined the influence of systematic doping. This doping is the key element to achieve the Shannon limit in channel coding as it opens the tunnel in the beginning of iterative decoding, when no a priori information is available. However in turbo source coding the tunnel is already open because of the SSI and according to this systematic doping does not increase the compression efficiency.

The other class of codes we studied are irregular repeat accumulate codes. Our serially concatenated turbo code consists of an inner scrambler with polynomial $1 + D^3/1 + D + D^3$ and an outer irregular repetition code of rate $R_{ir} = \frac{1}{5}$ with sub codes of rate $R_s = \frac{1}{n_s}$.

To illustrate the performance of irregular repeat accumulate codes the characteristic curves as well as a decoding trajectory are shown in Fig. 3. On the left hand side one sees the transfer function of the inner accumulator obtained by puncturing at a compression rate of $R_{comp} = 0.49$ bits/symbol. The optimized transfer curve of the outer repetition code, which fits the curve of the inner code, is the weighted sum of a small number of different sub codes $\mathcal{C}_s$. On the right hand side of Fig. 3 one identifies the same optimized transfer function of the outer code, but with an inner characteristic curve obtained by less puncturing. The result is a very tight, but still open tunnel and thus the source sequence can be recovered successfully as displayed by the decoding trajectory. Fig. 4 shows the histogram of compression rates achieved by turbo source coding and Huffman coding applied to 250 blocks of 2^{16} bits each. The Huffman code is implemented as specified in [10], which works on 256 different symbols. Thus we grouped the bit sequence resulting in a block of 2^{13} symbols and compressed them. It is obvious that the optimized turbo source code outperforms the parallel concatenation dramatically. Even the compression efficiency of Huffman coding is outmatched a little bit by irregular repeat accumulate codes in this parameter setup. Notice that the distance between the bars of the serial turbo source code histogram depends on the step size of the puncturing rate ΔR_{comp}. It can be reduced by the design of the puncturing matrix at the price of higher encoding complexity, as more integrity tests have to be passed.

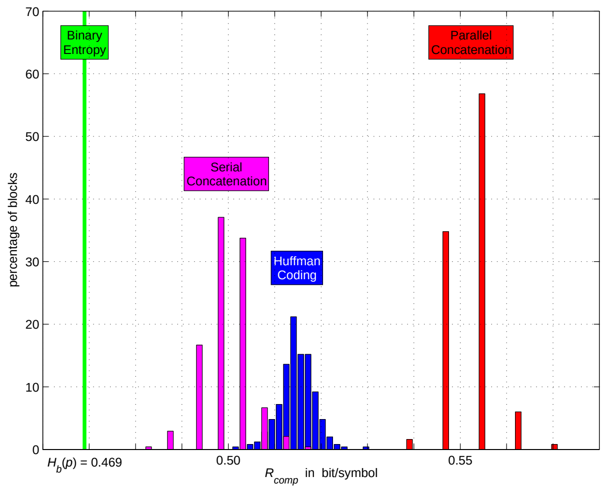

Fig. 4. Histogram showing the compression rates of different turbo source codes in comparison with Huffman coding.

V. Conclusions

We applied the principle of lossless turbo source coding to serially concatenated turbo codes. By means of modified EXIT charts we optimized irregular repeat accumulate codes. This code class is appropriate to turbo source coding as the binary entropy of the interleaved code word obtained by the repetition code is the same as the entropy of the source and thus the SSI has not to be changed. Another advantage is the analytical calculation of the characteristic curves of its sub codes. Numerical results have shown that these turbo source codes outperform the parallel concatenated scheme as well as standard compression tools. Furthermore it was indicated that compression rates close to the entropy are attainable.

Acknowledgments

The author gratefully acknowledges fruitful discussions with João Barros and Michael Tüchler and thanks Dmytro Chakhoyan for implementing the software.

References

[1] J. Hagenauer, J. Barros, and A. Schaefer, “Lossless turbo source coding with decremental redundancy,” in *Proc. International ITG Conference on Source and Channel Coding*, Erlangen, Germany, January 2004, pp. 333–339.

[2] C. Berrou, A. Glavieux, and P. Thitimajshima, “Near Shannon limit error-correcting coding and decoding: Turbo-codes,” in *Proc. IEEE International Conference on Communications*, vol. 2, Geneva, Switzerland, May 1993, pp. 1064–1070.

[3] R. G. Gallager, “Low-density parity-check codes,” *IRE Transactions on Information Theory*, pp. 21–28, January 1962.

[4] G. Caire, S. Shamai, and S. Verdú, “A new data compression algorithm for sources with memory based on error correcting codes,” in *Proc. IEEE Information Theory Workshop*, Paris, France, April 2003, pp. 291–295.

[5] ——, “Universal data compression with LDPC codes,” in *Proc. Interantional Symposium on Turbo Codes & Related Topics*, Brest, France, September 2003, pp. 55–58.

[6] J. García-Frías and Y. Zhao, “Compression of binary memoryless sources using punctured turbo codes,” *IEEE Communications Letters*, pp. 394–396, September 2002.

[7] S. ten Brink, “Rate one-half code for approaching the shannon limit by 0.1 dB,” *Electronics Letters*, pp. 1293–1294, July 2000.

[8] M. Tüchler and J. Hagenauer, “EXIT charts of irregular codes,” in *Proc. Conference on Information Sciences and Systems*, Princeton, NJ, USA, March 2002.

[9] A. Ashikhmin, G. Kramer, and S. ten Brink, “Extrinsic information transfer functions: Model and erasure channel properties,” *IEEE Transactions on Information Theory*, vol. 50, no. 11, pp. 2657–2673, November 2004.

[10] M. Nelson, *The Data Compression Book*. Prentice Hall, 1991, ch. 3, The Dawn Age: Minimum Redundancy Coding, pp. 29–80.