Convergence Behavior of Iteratively Decoded Parallel Concatenated Codes

Stephan ten Brink,*Member, IEEE*

Abstract— **Mutual information transfer characteristics for soft in/soft out decoders are proposed as a tool to better understand the convergence behavior of iterative decoding schemes. The exchange of extrinsic information is visualized as a decoding trajectory in the Extrinsic Information Transfer Chart (EXIT chart). This allows the prediction of turbo cliff position and bit error rate after an arbitrary number of iterations. The influence of code memory, code polynomials as well as different constituent codes on the convergence behavior is studied for parallel concatenated codes. The applicability to other than Gaussian channels is illustrated. A code search based on the EXIT chart technique has been performed which yielded new recursive systematic convolutional constituent codes exhibiting turbo cliffs at low signal–to–noise ratios not attainable by previously known constituent codes. With these results the gap to Shannon's capacity limit on the Gaussian channel narrows to 0.3dB for code rates of 1:2 and 1:3.**

Keywords— **Iterative decoding, turbo codes, convergence, mutual information.**

I. Introduction

TYPICALLY , bit error rate (BER) charts of iterative decoding schemes can be divided into three regions: 1. the region of low E_b/N_0 with negligible iterative BER reduction, 2. the turbo cliff region (also referred to as "waterfall"–region) with persistent iterative BER reduction over many iterations, and 3. the BER floor region for moderate to high E_b/N_0 in which a rather low BER can be reached after just a few number of iterations. While good analytical bounding techniques have been found for moderate to high E_b/N_0, e. g. [1], [2], the turbo cliff has not yet attracted a comparable amount of interest, owing to the limitations of the commonly used bounding techniques in that region. This paper proposes extrinsic information transfer characteristics based on mutual information to describe the flow of extrinsic information through the soft in/soft out constituent decoders. This proves to be particularly useful in the region of low E_b/N_0. A decoding trajectory visualizes the exchange of extrinsic information between constituent decoders in the Extrinsic Information Transfer Chart (EXIT chart).

In [3], [4] the EXIT chart was introduced as a novel tool to provide design guidelines for mappings and signal constellations of an iterative demapping and decoding scheme (IDEM). IDEM can be regarded as a serial concatenation of two codes (SCC). In this paper the method of [3], [4] is applied to iterative decoding of parallel concatenated codes (PCC). We do not claim to present a rigorous proof of stability and convergence of iterative decoding; however, simulation results suggest that the EXIT chart predicts the best possible convergence behavior of the iterative decoder for large interleaving depth.

The paper is organized as follows: Section II introduces extrinsic information transfer characteristics for the constituent decoders. Section III explains the EXIT chart as a novel description of the iterative decoder, complementary to BER charts. The applicability to other than Gaussian channels is shown in Section IV for the case of a Rayleigh channel. Code search results based on the EXIT chart technique are presented in Section V yielding new constituent codes which are optimized with respect to the turbo cliff position. Finally, Section VI renders some conclusions.

S. ten Brink is with the Institute of Telecommunications, Room 2.333, Dep. 0408, University of Stuttgart, Pfaffenwaldring 47, 70569 Stuttgart, Germany. Tel: +49 711 685 7937, Fax: +49 711 685 7929, E-mail: tenbrink@inue.uni-stuttgart.de . The paper was presented in part at the 3rd IEEE/ITG Symposium on Source and Channel Coding, Munich, Germany, January 2000. This work has been performed in a joint project with Bell Laboratories, Lucent Technologies.

II. Extrinsic Transfer Characteristics

A. Iterative Decoder for Parallel Concatenated Codes

The iterative decoder for PCC is shown in Fig. 1. For each iteration, the first constituent decoder (BCJR–algorithm [5], [6]) takes channel observations Z_1 on the systematic (information) bits i and respective parity bits p_1 and outputs soft values D_1. The extrinsic information on the systematic bits $E_1 = D_1 - A_1 - Z_1$ is passed through the bit interleaver to become the *a priori* input A_2 of the second decoder. The second decoder takes the permuted channel observations Z_2 on the systematic bits i and respective parity bits p_2 and feeds back extrinsic information $E_2 = D_2 - A_2 - Z_2$ which becomes the *a priori* knowledge A_1 of the first decoder. The variables Z_1, A_1, D_1, E_1, Z_2, A_2, D_2 and E_2 denote log–likelihood ratios (L–values [7]).

For the received discrete–time signal $z = x + n$ from the AWGN–channel, the conditional probability density function (PDF) writes as

$$p(z\,|X = x) = \frac{e^{-\frac{(z-x)^2}{2\sigma_n^2}}}{\sqrt{2\pi}\sigma_n}. \tag{1}$$

The binary random variable X denotes the transmitted bits with realizations $x \in \{\pm 1\}$; for brevity of notation, we will not distinguish between X and x in the following (only where needed for clarification). The corresponding L–values Z are calculated as

$$Z = \ln \frac{p(z\,|x = +1)}{p(z\,|x = -1)} \tag{2}$$

which can be simplified to

$$Z = \frac{2}{\sigma_n^2} \cdot z. \tag{3}$$

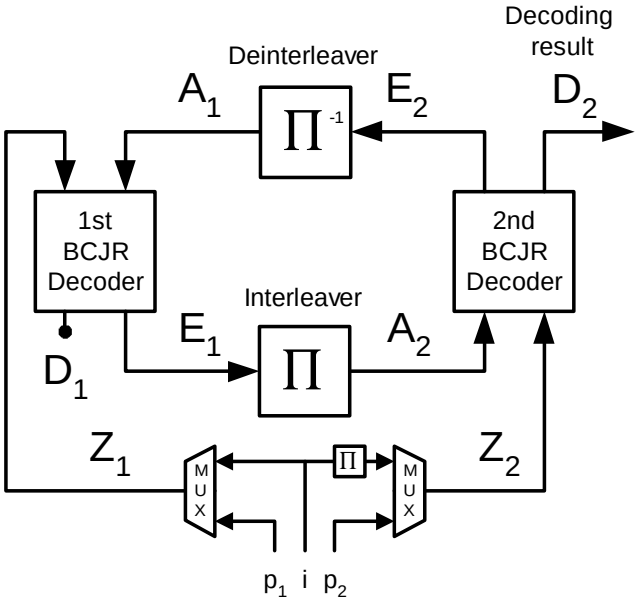

Fig. 1. Iterative decoder for parallel concatenated codes.

The variable n is Gaussian distributed with mean zero and variance $\sigma_n^2 = N_0/2$ (double–sided noise power spectral density). Note that Z can also be formulated as

$$Z = \mu_Z \cdot x + n_Z \tag{4}$$

with

$$\mu_Z = \frac{\sigma_Z^2}{2} \tag{5}$$

and n_Z being Gaussian distributed with mean zero and variance $\sigma_Z^2 = 4/\sigma_n^2$. This decomposition will turn out to be useful for modeling *a priori* knowledge in the next sub–section.

The parallel decoder of Fig. 1 is a symmetric arrangement: The situation for the second decoder with respect to Z_2, A_2, E_2 is essentially the same as for Z_1, A_1, E_1. Long sequence lengths make sure that tail effects (open/terminated trellises of convolutional codes) can be neglected. Hence, it is sufficient to focus on the first decoder for the remainder of Section II. To simplify notation the decoder index "1" is omitted in the following.

B. Transfer Characteristics of Constituent Decoders

The idea is to predict the behavior of the iterative decoder by solely looking at the input/output relations of individual constituent decoders. Since analytical treatment of the BCJR–decoder is difficult, we make use of the following observations obtained by simulation: 1. For large interleavers the *a priori* values A remain fairly uncorrelated from the respective channel observations Z over many iterations. 2. The probability density functions of the extrinsic output values E (*a priori* values A for the next decoder respectively) approach Gaussian distributions with increasing number of iterations.

An explanation for the first observation can be found by looking at the decoder output $D = Z + A + E$. For soft in/soft out decoding with the BCJR–algorithm the extrinsic information E_k of the bit at time instance k is not influenced by the channel observations Z_k or *a priori* knowledge A_k [8]. A large interleaver further contributes to reduce correlations and to get a better "separation" of both decoders. Possible reasons for the second observation are a) the use of a Gaussian channel model, and b) that sums over many values are involved in the L–value calculation of E which typically leads to Gaussian–like distributions.

Observations 1 and 2 suggest that the *a priori* input A to the constituent decoder can be modeled by applying an independent Gaussian random variable n_A with variance σ_A^2 and mean zero in conjunction with the known transmitted systematic bits x.

$$A = \mu_A \cdot x + n_A \tag{6}$$

Since A is supposed to be an L–value based on Gaussian distributions, as in the case of (4), (5), the mean value μ_A must fulfill

$$\mu_A = \frac{\sigma_A^2}{2}. \tag{7}$$

With (7) the conditional probability density function belonging to the L–value A is

$$p_A(\xi|X=x) = \frac{e^{-\frac{(\xi - \frac{\sigma_A^2}{2}\cdot x)^2}{2\sigma_A^2}}}{\sqrt{2\pi}\,\sigma_A}. \tag{8}$$

To measure the information contents of the *a priori* knowledge, mutual information $I_A = I(X;A)$ [9], [10] between transmitted systematic bits X and the L–values A is used.

$$I_A = \frac{1}{2} \cdot \sum_{x=-1,1} \int_{-\infty}^{+\infty} p_A(\xi|X=x) \times \mathrm{ld} \frac{2 \cdot p_A(\xi|X=x)}{p_A(\xi|X=-1) + p_A(\xi|X=1)}\, d\xi \tag{9}$$

$$0 \le I_A \le 1 \tag{10}$$

With (8), equation (9) becomes

$$I_A(\sigma_A) = \int_{-\infty}^{+\infty} \frac{e^{-\frac{(\xi - \sigma_A^2/2)^2}{2\sigma_A^2}}}{\sqrt{2\pi}\,\sigma_A} \cdot \left(1 - \mathrm{ld}\left[1 + e^{-\xi}\right]\right) d\xi\,. \tag{11}$$

For abbreviation we define

$$J(\sigma) := I_A(\sigma_A = \sigma) \tag{12}$$

with

$$\lim_{\sigma \to 0} J(\sigma) = 0,\ \lim_{\sigma \to \infty} J(\sigma) = 1,\ \sigma > 0 \tag{13}$$

The function $J(\sigma)$ cannot be expressed in closed form. As it turns out from numerical integration, $J(\sigma)$ is monotonically increasing and thus reversible.

$$\sigma_A = J^{-1}(I_A) \tag{14}$$

Mutual information is also used to quantify the extrinsic output $I_E = I(X;E)$.

$$I_E = \frac{1}{2} \cdot \sum_{x=-1,1} \int_{-\infty}^{+\infty} p_E(\xi|X=x) \times \mathrm{ld} \frac{2 \cdot p_E(\xi|X=x)}{p_E(\xi|X=-1) + p_E(\xi|X=1)} \, d\xi \tag{15}$$

$$0 \le I_E \le 1 \tag{16}$$

Viewing I_E as a function of I_A and the E_b/N_0–value, the extrinsic information transfer characteristics are defined as

$$I_E = T(I_A, E_b/N_0) \tag{17}$$

or, for fixed E_b/N_0, just

$$I_E = T(I_A). \tag{18}$$

To calculate $T(I_A, E_b/N_0)$ for the desired $(I_A, E_b/N_0)$–input combination, the distributions p_E in (15) are most conveniently determined by means of Monte Carlo simulation. For this, the independent Gaussian random variable of (6) is applied as *a priori* input to the constituent decoder of interest. Note that a certain value of I_A is obtained by appropriately choosing the parameter σ_A according to (14). Sequence lengths of 10^4 systematic bits sufficiently suppress tail effects.

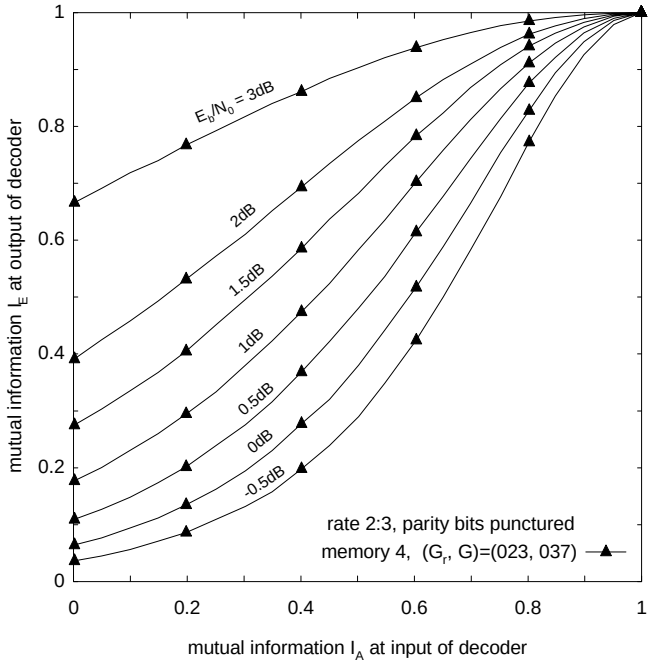

Fig. 2. Extrinsic information transfer characteristics of soft in/soft out decoder for rate 2:3 convolutional code, E_b/N_0 of channel observations as parameter to curves.

Transfer characteristics $I_E = T(I_A, E_b/N_0)$ are shown in Fig. 2. The *a priori* input I_A is on the abscissa, the extrinsic output I_E on the ordinate. The E_b/N_0–value serves as a parameter to the curves. The BCJR–algorithm is applied to a rate 1:2 recursive systematic convolutional code of memory 4; the parity bits are punctured to obtain a rate 2:3 constituent code. This will lead to a rate 1:2 PCC in Section III. The code polynomials are $(G_r, G) = (023, 037)$. G_r stands for the (recursive) feedback polynomial; the values are given in octal, with the most significant bit (MSB) corresponding to the generator connection on the very left (input) side of the shift register. Note that E_b/N_0–values are given with respect to the rate 1:2 parallel concatenated code.

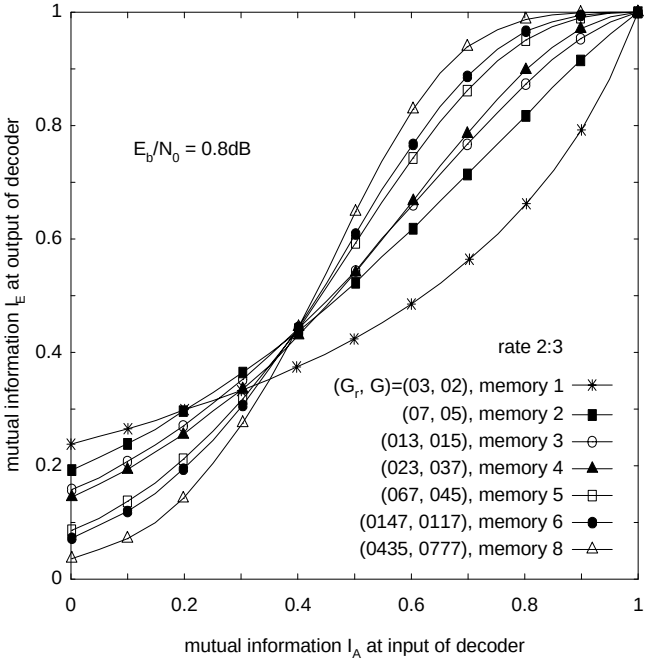

Fig. 3. Extrinsic information transfer characteristics of soft in/soft out decoder for rate 2:3 convolutional code, $E_b/N_0 = 0.8$dB, different code memory.

Transfer characteristics for different code memory at fixed $E_b/N_0 = 0.8$dB are depicted in Fig. 3. The code polynomials are taken from [11].

Fig. 4 shows the influence of different code polynomials for the prominent case of a memory 4 code. The $(023, 011)$–code provides good extrinsic output at the beginning, but returns diminishing output for higher *a priori* input. For the $(023, 035)$–code it is the other way round. The constituent code of the classic rate 1:2 PCC of [12] with polynomials $(037, 021)$ has good extrinsic output for low to medium *a priori* input.

From Fig. 2–4 it can be seen that the characteristics $I_E = T(I_A)$ are monotonically increasing in I_A, $0 \le I_A \le 1$, and thus the inverse function

$$I_A = T^{-1}(I_E) \tag{19}$$

exists on

$$T(0) \le I_E \le T(1). \tag{20}$$

Although no analytical proof can be given here, it is plausible that increased *a priori* input provides bigger extrinsic output.

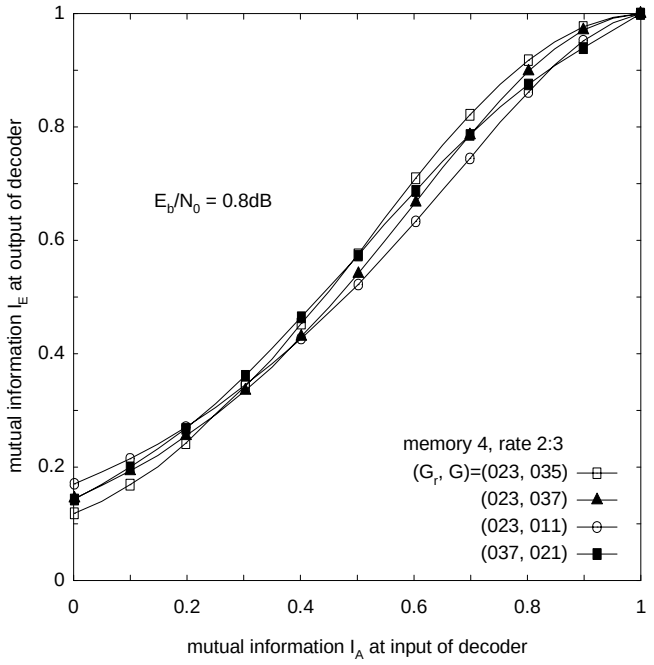

Fig. 4. Extrinsic information transfer characteristics of soft in/soft out decoder for rate 2:3 convolutional code, $E_b/N_0 = 0.8$dB, memory 4, different code polynomials.

The following properties make mutual information the preferred measure in comparison to other quantities based on mean and variance values: 1. *The robustness*: The shape of the probability density functions is accounted for which makes transfer characteristics based on mutual information quite robust against changes in the shape of the *a priori* input distributions p_A; this robustness has already been acknowledged for the entropy measure in [9], [10]. In an iterative decoder the actual distributions p_A (p_E respectively) differ significantly from Gaussian for the very first few iterations. However, the Gaussian approximation seems to be sufficiently close since the transfer characteristics remain essentially unchanged if the actual distributions are applied. 2. *The information–theoretic interpretation*: According to Shannon's channel coding theorem, mutual information between channel input and output variable directly relates to the maximal possible information rate for which reliable (i. e. error–free) transmission is achievable. 3. *The value range and logarithmic scaling*: Mutual information per binary symbol stays within the interval [0..1] which makes it convenient to plot in a chart with equally scaled abscissa and ordinate.

III. Extrinsic Information Transfer Chart

A. Trajectories of Iterative Decoding

To account for the iterative nature of the sub–optimal decoding algorithm, both decoder characteristics are plotted into a single diagram. However, for the transfer characteristics of the second decoder the axes are swapped.

This diagram is referred to as Extrinsic Information Transfer Chart (EXIT chart) since the exchange of extrinsic information can be visualized as a decoding trajectory. Provided that independence and Gaussian assumptions hold for modeling extrinsic information (*a priori* information respectively), the transfer characteristics of Section II should approximate the true behavior of the iterative decoder. Moreover, the decoding trajectory that can be graphically obtained by simply drawing a zigzag–path into the EXIT chart (bounded by the decoder transfer characteristics) should match with the trajectory computed by simulations.

Let n be the iteration index, E_b/N_0 fixed. For $n = 0$ the iteration starts at the origin with zero *a priori* knowledge $I_{A_1,0} = 0$. At iteration n, the extrinsic output of the first decoder is $I_{E_1,n} = T_1(I_{A_1,n})$. $I_{E_1,n}$ is forwarded to the second decoder to become $I_{A_2,n} = I_{E_1,n}$ (ordinate). The extrinsic output of the second decoder is $I_{E_2,n} = T_2(I_{A_2,n})$, which is fed back to the first decoder to become the *a priori* knowledge $I_{A_1,n+1} = I_{E_2,n}$ (abscissa) of the next iteration. Note that interleaving does not change mutual information.

The iteration proceeds as long as $I_{E_2,n+1} > I_{E_2,n}$. With $I_{E_2,n+1} = T_2(T_1(I_{E_2,n}))$ this can be formulated as $T_1(I_{E_2,n}) > T_2^{-1}(I_{E_2,n})$. The iteration stops if $I_{E_2,n+1} = I_{E_2,n}$, or equivalently, $T_1(I_{E_2,n}) = T_2^{-1}(I_{E_2,n})$, which corresponds to an intersection of both characteristics in the EXIT chart.

It is interesting to recognize that an intersection of both transfer characteristics also relates to minimizing cross–entropy between consecutive extrinsic output distributions p_E. This is in agreement with the termination criterion of minimizing cross–entropy as introduced by [7] and [13]. However, in our work we are not interested in finding a termination criterion, i. e. receiver algorithm assuming unknown transmitted bits x, but rather in establishing an engineering tool for designing iterative decoding schemes.

Fig. 5 shows trajectories of iterative decoding at $E_b/N_0 = 0.1$dB and 0.8dB obtained by simulations of the iterative decoder (code parameters are those of Fig. 2). For $E_b/N_0 = 0.1$dB the trajectory (lower left corner) gets stuck after two iterations since both decoder characteristics do intersect. For $E_b/N_0 = 0.8$dB the trajectory has just managed to "sneak through the bottleneck". After six passes through the decoder, increasing correlations of extrinsic information start to show up and let the trajectory deviate from its expected zigzag–path. As it turns out, for larger interleavers the trajectory stays on the characteristics for some more passes through the decoder.

Fig. 6 depicts the EXIT chart with transfer characteristics over a set of E_b/N_0–values and two decoding trajectories at 0.7dB and 1.5dB as obtained from simulations of the iterative decoder (interleaving depth 10^6 systematic bits). The curves in between 0dB and 1dB are in steps of 0.1dB. Note that in the graphical representation the decoder characteristics are only plotted up to their first intersection, in order not to overload the two–dimensional graph. An opening for the trajectory at 0.7dB can clearly be seen, which corresponds to the turbo cliff position in the BER chart. It is remarkable how closely the simulated trajectories match with the characteristics: Large interleavers en-

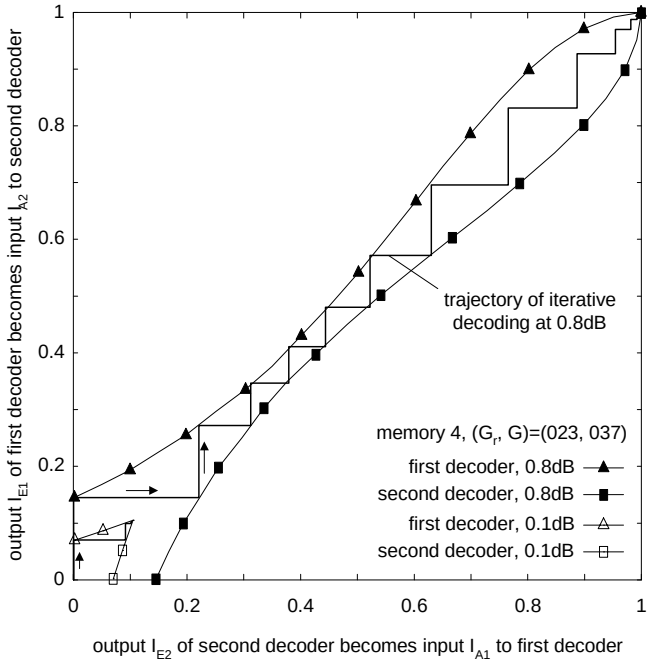

Fig. 5. Simulated trajectories of iterative decoding at $E_b/N_0 = 0.1$dB and 0.8dB (symmetric PCC rate 1:2, interleaver size 60000 systematic bits).

sure that the independence assumption of (6) holds over many iterations; the robustness of the mutual information measure allows to overcome non–Gaussian distributions of *a priori* information during the first few iterations. It should be emphasized, however, that the *decoding trajectory* is a simulation result of the iterative decoder, purely based on measurements of mutual information taken from the output of the respective constituent decoder "as is". Only to calculate the *transfer characteristics* of individual constituent decoders we were sought to impose the Gaussian and independence assumption on the *a priori* input A. The observation that simulated decoding trajectories and transfer characteristics match that well simply verifies that our model is good, and that mutual information is an appropriate measure.

Considering the behavior of the decoding trajectory in the EXIT chart we can come up with a more vivid nomenclature for the three typical regions of the BER chart as mentioned in the introductory section: 1. The region of low E_b/N_0 with negligible iterative BER reduction can also be referred to as the *pinch–off region* with the decoder transfer characteristics intersecting at low mutual information (corresponding to high BER) and the trajectory getting stuck. 2. The turbo cliff region is now the *bottleneck region* with the decoding trajectory just managing to sneak through a narrow tunnel; convergence towards low BER is slow, but possible since both decoder characteristics do not intersect anymore. 3. The BER floor region can also be regarded as the *wide–open region* with fast convergence. Likewise, the acronym EXIT can be understood as the opportunity to see at what E_b/N_0 the decoding trajectory succeeds in "exiting" the pinch–off region through the bottleneck to converge towards low BER.

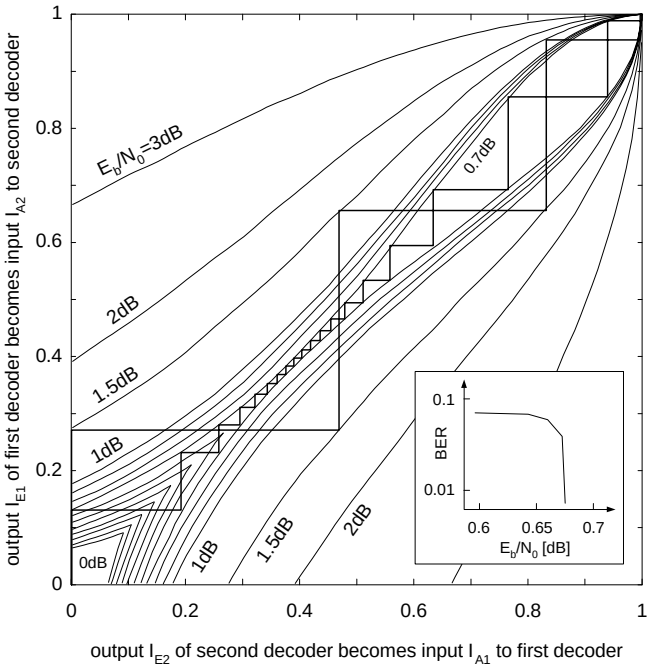

Fig. 6. EXIT chart with transfer characteristics for a set of E_b/N_0–values; two decoding trajectories at 0.7dB and 1.5dB (code parameters as in Fig. 5, PCC rate 1:2); interleaver size 10^6 systematic bits.

The main contribution of the EXIT chart to the understanding of iterative decoding is the advantage that only simulations of individual constituent decoders are needed to obtain the desired transfer characteristics. These can then be used in any combination in the EXIT chart to describe the behavior of the corresponding iterative decoder, asymptotic with respect to the interleaver size. No resource–intensive BER simulations of the iterative decoding scheme itself are required. Moreover, for turbo decoders with the same constituent codes ("symmetric turbo codes") as in the case of [12], transfer characteristics of only one constituent decoder are sufficient to predict the performance of iterative decoding. This further speeds up the evaluation of new parallel code concatenations. For symmetric turbo codes the straight line from (0,0) to (1,1), also referred to as "first diagonal of the EXIT chart", becomes significant since all intersections of both decoder characteristics will happen there.

B. Obtaining BER from EXIT Chart

The EXIT chart can be used to obtain an estimate on the BER after an arbitrary number of iterations. For both constituent decoders, the soft output on the systematic bits can be written as $D = Z + A + E$. For the sake of deriving a simple formula on the bit error probability P_b, we assume the *a priori* knowledge A and extrinsic output E to be Gaussian distributed. Consequently, the decoder soft output D is supposed to be Gaussian distributed with variance σ_D^2 and mean value $\mu_D = \sigma_D^2/2$, compare to (5), (7). With the complementary error function, the bit error probability

writes as

$$P_b \approx \frac{1}{2}\mathrm{erfc}\left(\frac{1}{\sqrt{2}}\frac{\mu_D}{\sigma_D}\right) = \frac{1}{2}\mathrm{erfc}\left(\frac{\sigma_D}{2\sqrt{2}}\right) . \qquad (21)$$

Assuming independence it is

$$\sigma_D^2 = \sigma_Z^2 + \sigma_A^2 + \sigma_E^2. \qquad (22)$$

With (3) and

$$\frac{E_b}{N_0} = \frac{1}{2R\sigma_n^2} \qquad (23)$$

we obtain σ_Z^2 as

$$\sigma_Z^2 = \left(\frac{2}{\sigma_n^2}\cdot\sigma_n\right)^2 = \frac{4}{\sigma_n^2} = 8R\cdot\frac{E_b}{N_0}. \qquad (24)$$

Applying (14), the variances σ_A^2 and σ_E^2 are calculated.

$$\sigma_A^2 \approx J^{-1}(I_A)^2,\ \sigma_E^2 \approx J^{-1}(I_E)^2 \qquad (25)$$

Finally, with (21), (22), (24) and (25) the result is

$$P_b \approx \frac{1}{2}\mathrm{erfc}\left(\frac{\sqrt{8R\cdot\frac{E_b}{N_0} + J^{-1}\left(I_A\right)^2 + J^{-1}\left(I_E\right)^2}}{2\sqrt{2}}\right) \qquad (26)$$

Fig. 7 shows transfer characteristics and respective simulated decoding trajectory of an asymmetric PCC with memory 2 and memory 6 constituent codes at 0.8dB: $(G_{r,1}, G_1) = (07, 05)$, $(G_{r,2}, G_2) = (0147, 0117)$. Note that the transfer characteristics are just taken from Fig. 3; the characteristic of the second decoder (memory 6) is mirrored at the first diagonal of the EXIT chart. Additionally, the BER scaling according to (26) is given as a contour plot. Table I compares BER–results obtained from the EXIT chart up to the 7th pass through the iterative decoder: (s) stands for the result obtained by simulation, right next to it the BER as calculated with (26). The table shows that the EXIT chart in combination with the Gaussian approximation of (26) provides reliable BER predictions down to 10^{-3}, that is, in the region of low E_b/N_0. It is not useful for determining BER floors. For this bounding techniques like [1] which include the interleaving depth are more suited.

TABLE I
COMPARISON OF BER PREDICTIONS FROM EXIT CHART. COLUMNS OF SIMULATION RESULTS ARE MARKED WITH (S).

	1st decoder		2nd decoder	
pass	BER (s)	BER (26)	BER (s)	BER (26)
1.	1.10e-1	1.10e-1	8.96e-2	8.97e-2
2.	7.86e-2	7.88e-2	6.85e-2	6.88e-2
3.	6.18e-2	6.21e-2	5.37e-2	5.39e-2
4.	4.78e-2	4.80e-2	3.94e-2	3.95e-2
5.	3.30e-2	3.31e-2	2.32e-2	2.32e-2
6.	1.65e-2	1.62e-2	7.63e-3	7.08e-3
7.	3.65e-3	2.95e-3	7.0e-4	4.4e-4

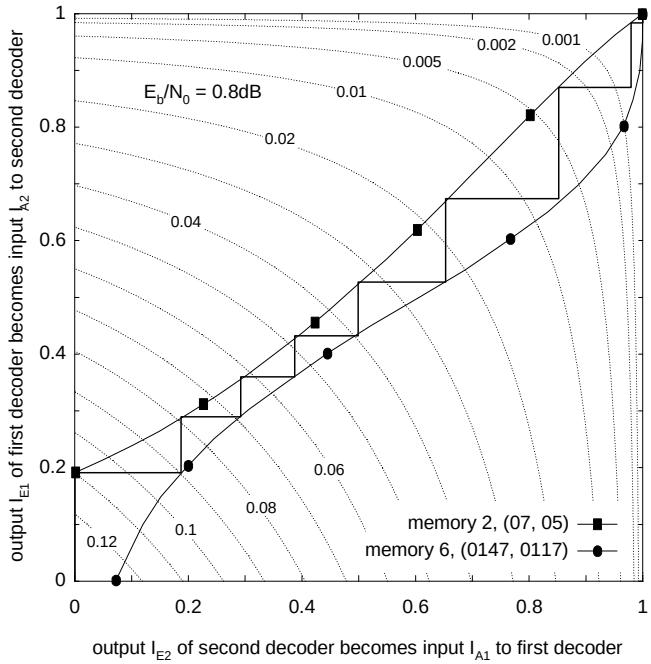

Fig. 7. Simulated trajectory of iterative decoding at $E_b/N_0 = 0.8$dB with BER scaling as contour plot (PCC rate 1:2, interleaver size 200000 systematic bits).

IV. EXIT Chart for the Rayleigh Channel

The EXIT chart technique is not limited to the Gaussian channel. In this section we explain the changes for a coherently detected, fully interleaved Rayleigh channel when perfect channel state information is available at the receiver. The received discrete–time signal is

$$z_c = a_c \cdot x + n_c \qquad (27)$$

with the transmitted binary symbols $x \in \{\pm 1\}$ and the complex additive noise $n_c = n_I + jn_Q$. The n_I, n_Q are realizations of independent Gaussian random variables with variance $\sigma_n^2 = N_0/2$. The complex fading coefficient is $a_c = a_I + ja_Q$. The a_I, a_Q are Gaussian distributed; their variance is normalized to $\sigma_a^2 = 1/2$ such that the magnitude $a = \sqrt{a_I^2 + a_Q^2}$ is Rayleigh distributed with $E[a^2] = 1$.

Assuming perfectly known a_c at the receiver the channel L–values are calculated as

$$Z = \ln\frac{p\left(z_c\,|a_c, x=+1\right)}{p\left(z_c\,|a_c, x=-1\right)} \qquad (28)$$

with

$$p\left(z_c\,|a_c, x\right) = \frac{1}{2\pi\sigma_n^2}\exp\left[\frac{\|z_c - a_c\cdot x\|^2}{2\sigma_n^2}\right] \qquad (29)$$

which results in

$$Z = \frac{2}{\sigma_n^2}\cdot\mathrm{Re}\left\{a_c^*\cdot z_c\right\} = \frac{2}{\sigma_n^2}\cdot\left(a^2\cdot x + a\cdot n\right). \qquad (30)$$

It is easy to show that n is Gaussian distributed with variance σ_n^2 and mean zero.

A. Extrinsic Transfer Characteristics

We noticed that the shapes of the extrinsic density functions at the decoder output are very similar to those of the Gaussian channel case. Hence, following the same arguments as in Section II-B, we model the *a priori* knowledge A as an independent Gaussian random variable according to (6). Likewise, the extrinsic information transfer characteristics $I_E = T_R(I_A, E_b/N_0)$ for the Rayleigh channel are computed by Monte Carlo simulation applying (9) and (15) in combination with the (no longer Gaussian distributed) channel L–values Z of (30). As for the Gaussian channel case, we found a very good agreement of transfer characteristics and simulated decoding trajectories.

B. Obtaining BER from EXIT Chart

The probability density function of the channel L–values Z can be calculated from (29) and (30) using integral [14], (3.325).

$$p_Z(t\,|x) = \frac{\sigma_n^2}{2\sqrt{1+2\sigma_n^2}} \cdot \exp\left[\frac{x\cdot t}{2} - \sqrt{1+2\sigma_n^2}\cdot\frac{|t|}{2}\right] \quad (31)$$

As in Section III-B, the decoder soft output D can be written as a sum of independent random variables $D = Z + A + E$, with the *a priori* values A and extrinsic output values E being approximated by independent Gaussian random variables. Thus, the sum $A + E$ itself is Gaussian distributed with variance $\sigma_{AE}^2 = \sigma_A^2 + \sigma_E^2$, mean value $\mu_{AE} = \sigma_{AE}^2/2$ and probability density function

$$p_{AE}(t\,|x) = \frac{1}{\sqrt{2\pi}\sigma_{AE}} \cdot \exp\left[\frac{1}{2\sigma_{AE}^2}\cdot\left(t - \frac{\sigma_{AE}^2}{2}\cdot x\right)^2\right]. \quad (32)$$

The PDF p_D of the sum $Z+A+E$ is calculated through convolution of (31) and (32)

$$p_D(t\,|x) = p_Z(t\,|x) * p_{AE}(t\,|x). \quad (33)$$

With integral [14], (3.322, 2.) we obtain

$$\begin{aligned} p_D(t\,|x) = {} & \frac{\sigma_n^2}{4\sqrt{1+2\sigma_n^2}} \cdot \exp\left(\frac{\sigma_n^2\sigma_{AE}^2}{4}\right) \\ & \times\left[\exp\left(\frac{1+\sqrt{1+2\sigma_n^2}}{2}\cdot x\cdot t\right)\right. \\ & \quad \times\operatorname{erfc}\left(\frac{1}{\sqrt{2}\sigma_{AE}}\cdot x\cdot t + \frac{\sigma_{AE}\cdot\sqrt{1+2\sigma_n^2}}{2\sqrt{2}}\right) \\ & + \exp\left(\frac{1-\sqrt{1+2\sigma_n^2}}{2}\cdot x\cdot t\right) \\ & \quad \left.\times\operatorname{erfc}\left(-\frac{1}{\sqrt{2}\sigma_{AE}}\cdot x\cdot t + \frac{\sigma_{AE}\cdot\sqrt{1+2\sigma_n^2}}{2\sqrt{2}}\right)\right]. \end{aligned} \quad (34)$$

The bit error probability is calculated from (34) by integration

$$P_b = \int_{-\infty}^{0} p_D(t\,|x=1)\cdot dt. \quad (35)$$

Applying integral [14], (6.282, 1.) gives rise to the following closed–form result

$$\begin{aligned} P_b = {} & \tfrac{1}{2}\operatorname{erfc}\left(\frac{\sigma_{AE}}{2\sqrt{2}}\right) \\ & -\tfrac{1}{2}\operatorname{erfc}\left(\frac{\sigma_{AE}\sqrt{1+2\sigma_n^2}}{2\sqrt{2}}\right)\cdot\frac{1}{\sqrt{1+2\sigma_n^2}}\cdot\exp\left[\frac{\sigma_n^2\sigma_{AE}^2}{4}\right]. \end{aligned} \quad (36)$$

With (23), (36) and $\sigma_{AE}^2 \approx J^{-1}(I_A)^2 + J^{-1}(I_E)^2$ an estimate on the BER can be obtained from the EXIT chart. For the simulated decoding trajectory of Fig. 8 the same constituent code combination as in Fig. 7 is used. Additionally, we chose $E_b/N_0 = 2.6$dB such that the distance to the capacity limit is comparable to the Gaussian channel case of Fig. 7: For rate 1:2, the capacity limit of the Rayleigh channel is at 1.83dB, and thus the distance is 2.6dB − 1.73dB = 0.77dB; for the Gaussian channel the capacity limit is at 0.19dB and the distance is 0.8dB − 0.19dB = 0.61dB. The BER obtained from (36) is given as contour lines. As indicated in Table II, the BER predictions are quite accurate down to 10^{-3}.

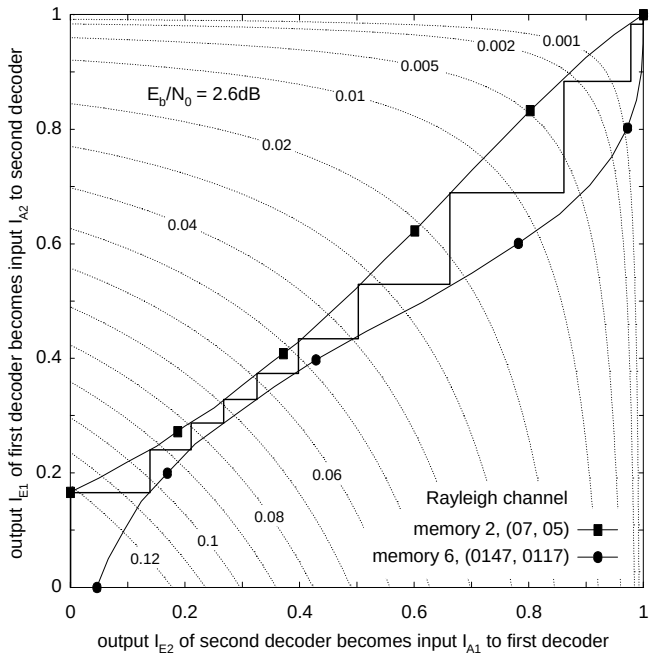

Fig. 8. Simulated trajectory of iterative decoding at $E_b/N_0 = 2.6$dB with BER scaling as contour plot (PCC rate 1:2, interleaver size 200000 systematic bits); Rayleigh channel.

V. Applications of the EXIT Chart

The results of Fig. 2–4 can be re–interpreted in the EXIT chart. Obviously, with respect to Fig. 3, a big code memory hurts at the beginning of the iteration, but helps for reaching a low BER floor. Assuming symmetric PCC one can draw a straight line from (0,0) to (1,1) into Fig. 3 to determine intersections with the decoder characteristics at $E_b/N_0 = 0.8$dB and consequently gaining insight into the turbo cliff position of the code concatenation of interest. For example, the trajectory of iterative decoding would find a fairly open tunnel to creep through for symmetric PCC

TABLE II
COMPARISON OF BER PREDICTIONS (RAYLEIGH CHANNEL). COLUMNS OF SIMULATION RESULTS ARE MARKED WITH (S).

	1st decoder		2nd decoder	
pass	BER (s)	BER (36)	BER (s)	BER (36)
1.	1.23e-1	1.22e-1	1.05e-1	1.03e-1
2.	9.44e-2	9.30e-2	8.60e-2	8.48e-2
3.	8.00e-2	7.92e-2	7.42e-2	7.33e-2
4.	6.95e-2	6.89e-2	6.41e-2	6.33e-2
5.	5.90e-2	5.89e-2	5.31e-2	5.25e-2
6.	4.81e-2	4.74e-2	3.95e-2	3.91e-2
7.	3.32e-2	3.26e-2	2.34e-2	2.22e-2
8.	1.63e-2	1.48e-2	7.24e-3	6.21e-3
9.	3.86e-3	2.42e-3	9.5e-4	4.05e-4

of memory 2, 3 and 4; for a memory 5 code the convergence is just about to become possible at $E_b/N_0 = 0.8$dB; for constituent codes of memory 1, 6 and 8 the trajectory would get stuck after a few iterations. Note, however, that the memory 6 code in combination with a memory 2 code does converge at 0.8dB, as shown in Fig. 7.

A. Search for "Early Convergent" Codes

There have been various code searches for parallel concatenated codes focusing on optimizing "effective" free distances of constituent codes and multiplicities thereof [11], [15]. While these quantities govern the code performance in the BER floor region, they have only little influence on the convergence properties in the turbo cliff region. The EXIT chart provides a new design criterion which can be included into code searches to find PCC with "early convergence", that is, with turbo cliff positions at low E_b/N_0. Table III shows the result of a code search over PCC of code rate 1:2 (rate 1:2 constituent codes punctured to rate 2:3) and rate 1:3 up to memory 6. We restricted ourselves to symmetric PCC. Hence it was sufficient to consider only single transfer characteristics to decide on the convergence of the iterative decoder: The extrinsic transfer characteristic of the respective constituent decoder must not intersect with the first diagonal of the EXIT chart, or equivalently, $I_E > I_A$ must be fulfilled for $0 < I_A < 1$.

We demanded convergence to low BER at 0.5dB for the rate 1:2 PCC (0.31dB away from Gaussian capacity limit 0.19dB), and at -0.2dB for the rate 1:3 PCC (0.3dB away from capacity limit -0.5dB). We relaxed the convergence–criterion and allowed intersections of transfer characteristics above $I_A > 0.9$ such that bit error rates of about 10^{-3} and better can be reached. No code with memory smaller than 4 could meet these requirements. For the rate 1:2 PCC we found one recursive systematic convolutional constituent code of memory 4, no memory 5 and two memory 6 codes exhibiting a turbo cliff lower than 0.5dB. For the rate 1:3 PCC, "early converging" codes with turbo cliffs lower than -0.2dB were found for memory 4, 5 and 6. From the work of [16] it appears very likely that codes with higher memory exist which have similarly low turbo cliffs.

As the code search results are based on the transfer characteristics of individual constituent codes we verified them

TABLE III
EXAMPLES OF SYMMETRIC PCC WITH EARLY CONVERGENCE. GAUSSIAN CHANNEL

	PCC rate 1:2		PCC rate 1:3	
ν	(G_r, G)	BER at 0.5dB	(G_r, G)	BER at -0.2dB
4	(022,037)	2e-5	(025,037)	3e-6
5	—	—	(044,073)	2e-5
6	(0102,0147)	2e-5	(0111,0113)	1e-6
6	(0110,0141)	1e-3	(0120,0117)	3e-3
6	—	—	(0120,0127)	2e-4

by simulating the respective iterative decoders using an interleaving depth of 10^6 systematic bits. The BER given in Table III is averaged over 10 blocks (i. e. 10^7 systematic bits, 40 iterations). Fig. 9 shows some decoding trajectories for rate 1:2 PCC at 0.5dB as obtained from simulations of the iterative decoder. Additionally, contour lines according to (21) indicate where a BER of 0.1, 0.01 and 0.001 is reached. The figure highlights yet again the prediction capabilities of the EXIT chart and illustrates how the convergence gets slowed down in the bottleneck region.

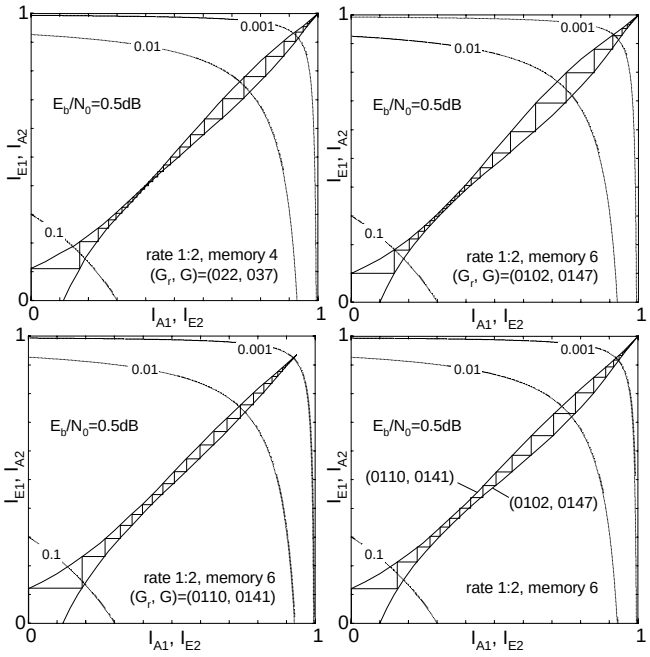

Fig. 9. Examples of decoding trajectories for three symmetric and one asymmetric rate 1:2 PCC with turbo cliff below 0.5dB (Gaussian channel); interleaver size 10^6 systematic bits.

B. Outlook to Asymmetric Turbo Codes

Asymmetric turbo codes with constituent codes of different memory (compare Fig. 7,8) or different code polynomials offer another degree of freedom in the design of parallel concatenated codes. They are particularly attractive to obtain "balanced" codes with a good trade–off between early turbo cliff (i. e. big extrinsic output for low to moderate I_A–input) and reasonable BER floor (i. e. big extrinsic output for high I_A–input).

From Fig. 9 we can see that the memory 6 code with polynomials (0110, 0141) has a fairly open bottleneck region at 0.5dB, but the trajectory gets stuck at a BER of about 10^{-3}, owing to the intersection of both characteristics before reaching $(I_A, I_E) = (1, 1)$. We can cure this problem by combining it with the memory 6, (0102, 0147)–code to obtain both a fairly open bottleneck region and an open tunnel up to $(I_A, I_E) = (1, 1)$ (see bottom right EXIT chart of Fig. 9). For an interleaver size of 10^6 systematic bits this asymmetric PCC achieves a BER of about 10^{-5}.

VI. Summary and Conclusion

Mutual information between transmitted systematic bits and extrinsic output of constituent decoders was found to be a useful measure for gaining insight into the convergence behavior of iterative decoding. Extrinsic information transfer characteristics have been shown to be particularly suited for the description of soft in/soft out decoders in the region of low E_b/N_0. The EXIT chart has been presented as an engineering tool for the design of iterative decoding schemes, in principle not limited to parallel concatenated codes. In the case of symmetric PCC the computation of transfer characteristics for only one constituent decoder turned out to be sufficient to predict the performance of the corresponding iterative decoder. Illustratively, the capabilities of the EXIT chart technique have been demonstrated by predicting the BER after up to 9 iterations for asymmetric PCC in Gaussian and Rayleigh channels. Additionally, extrinsic information transfer characteristics were included into a code search for symmetric PCC up to memory 6 which yielded new constituent codes optimized with respect to convergence at low E_b/N_0. The improvements to previously known constituent codes may be regarded as marginal; however, the EXIT chart technique is a first systematic way of optimizing code concatenations with respect to convergence of iterative decoding.

References

[1] S. Benedetto, G. Montorsi, "Unveiling Turbo Codes: Some Results on Parallel Concatenated Coding Schemes", *IEEE Trans. Inform. Theory*, vol. 42, no. 2, pp. 409–428, Mar. 1996

[2] T. Duman, R. Salehi "New performance bounds for turbo codes", *Proc. GLOBECOM*, pp. 634–638, Nov. 1997

[3] S. ten Brink, "Convergence of iterative decoding", *Electron. Lett.*, vol. 35, no. 10, pp. 806–808, May 1999

[4] S. ten Brink, "Iterative decoding for multicode CDMA", *Proc. IEEE VTC*, pp. 1876–1880, May 1999

[5] L. Bahl, J. Cocke, F. Jelinek, J. Raviv, "Optimal decoding of linear codes for minimizing symbol error rate", *IEEE Trans. Inform. Theory*, vol. 20, pp. 284–287, Mar. 1974

[6] P. Robertson, E. Villebrun, P. Hoeher, "A comparison of optimal and sub–optimal MAP decoding algorithms operating in the log domain", *Proc. ICC*, pp. 1009–1013, June 1995

[7] J. Hagenauer, E. Offer, L. Papke, "Iterative Decoding of Binary Block and Convolutional Codes", *IEEE Trans. Inform. Theory*, vol. 42, no. 2, pp. 429–445, Mar. 1996

[8] J. Hagenauer, P. Robertson, L. Papke, "Iterative ('Turbo') decoding of systematic convolutional codes with the MAP and SOVA algorithms", *Proc. ITG Symposium on Source and Channel Coding*, pp. 21–29, Munich, 1994

[9] T. M. Cover, J. A. Thomas, *Elements of Information Theory.* Wiley, New York, 1991.

[10] R. W. Hamming, *Coding and Information Theory.* Prentice–Hall, New Jersey, 1986.

[11] M. S. C. Ho, S. S. Pietrobon, T. Giles, "Improving the constituent codes of turbo encoders", *Proc. IEEE Globecom*, pp. 3525–3529, Sydney, Nov. 98

[12] C. Berrou, A. Glavieux, P. Thitimajshima, "Near Shannon limit error–correcting coding and decoding: Turbo–codes", *Proc. ICC*, pp. 1064–1070, May 1993

[13] M. Moher, T. A. Gulliver, "Cross–Entropy and Iterative Decoding", *IEEE Trans. Inform. Theory*, vol. 44, No. 7, pp. 3097–3104, Nov. 1998

[14] I. S. Gradshteyn, I. M. Ryzhik, *Tables of Integrals, Series and Products*, New York: Academic, 1980

[15] S. Benedetto, R. Garello, G. Montorsi, "A search for good convolutional codes to be used in the construction of turbo codes", *IEEE Trans. Comm.*, vol. 46, No. 9, pp. 1101–1105, Sep. 1998

[16] D. J. Costello, P. C. Massey, O. M. Collins, "Some Reflections on the Mythology of Turbo Codes" *Proc. 3rd IEEE/ITG Conference on Source and Channel Coding*, pp. 157–160, Munich, Jan. 2000