Iterative Decoding Trajectories of Parallel Concatenated Codes

Stephan ten Brink (tenbrink@inue.uni-stuttgart.de)
Institute of Telecommunications, Stuttgart University, Germany
in a joint project with Bell Laboratories, Lucent Technologies

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. This allows the prediction of turbo cliff position and bit error rate after an arbitrary number of iterations. The influence of code memory, generator polynomials as well as different constituent codes on the convergence behavior is studied for parallel concatenated codes.**

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], the turbo cliff region 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. The exchange of extrinsic information between constituent codes is visualized in the Extrinsic Information Transfer Chart (EIT chart). The EIT chart predictions on code performance assume large interleavers and can be regarded as asymptotic with respect to interleaving depth rather than with respect to the E_b/N_0–value.

The paper is organized as follows: Section II introduces extrinsic information transfer characteristics for the constituent decoders. Section III presents the EIT chart as a novel description of the iterative decoder, complementary to BER charts. Section IV presents some results and illustrates the use of the EIT chart for different code combinations. Finally, section V renders some conclusions.

II. Extrinsic Transfer Characteristics

In [2] the EIT chart was introduced as a novel method 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 [2] is applied to iterative decoding of parallel concatenated codes (PCC).

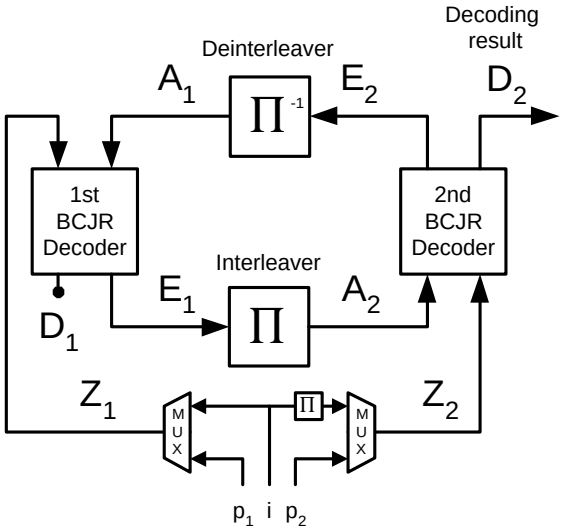

Fig. 1: Iterative decoder for parallel concatenated codes.

The iterative decoder for PCC is shown in Fig. 1. For each iteration, the first constituent decoder (BCJR–algorithm [4]) 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 for 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 for 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 [3]).

For the received signal $z = x + n$ from the AWGN-channel, the L–values Z are calculated according to

$$Z = \frac{2}{\sigma_n^2} \cdot z \tag{1}$$

with the transmitted bits $x \in \{\pm 1\}$, n being Gaussian distributed with variance $\sigma_n^2 = N_0/2$ (double–sided noise power spectral density).

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. For long sequence lengths tail effects (open/terminated trellises of convolutional codes) can be

Proceedings of 3rd IEEE/ITG Conference on Source and Channel Coding, Munich, Germany, January 2000, pp. 75-80

neglected. Hence, it is sufficient to focus on the first decoder for the remainder of this section. To simplify notation the decoder index "1" is omitted in the following.

The idea is to predict the behavior of the iterative decoder by solely looking at the input/ouput relations of individual constituent decoders. For this, 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 (PDF) 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 [5]. 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 modelled by applying an independent Gaussian random variable n_A with zero mean and variance σ_A^2.

$$A = \mu_A x + n_A \tag{2}$$

Since A is an L–value based on Gaussian distributions it can be shown [3] that μ_A must fulfill

$$\mu_A = \frac{\sigma_A^2}{2}. \tag{3}$$

With (3) the conditional probability density function belonging to the L–value A is

$$p_A(\xi|X=x) = \frac{e^{-\frac{(\xi - \frac{\sigma_A^2}{2}\cdot x)^2}{2\sigma_A^2}}}{\sqrt{2\pi}\,\sigma_A} \tag{4}$$

with $x \in \{\pm 1\}$.

To measure the information contents of the *a priori* knowledge, mutual information $I_A = I(X;A)$ [6, 7] between transmitted systematic bits X and the L–values A is used.

$$I_A = \frac{1}{2} \cdot \sum_{x=-1,1} \int_{-\infty}^{+\infty} p_A(\xi|X=x) \times \mathrm{ld} \frac{2 \cdot p_A(\xi|X=x)}{p_A(\xi|X=-1) + p_A(\xi|X=1)}\, d\xi \tag{5}$$

$$0 \le I_A \le 1 \tag{6}$$

With (4), equation (5) becomes

$$I_A(\sigma_A) = \int_{-\infty}^{+\infty} \frac{e^{-\frac{(\xi - \sigma_A^2/2)^2}{2\sigma_A^2}}}{\sqrt{2\pi}\,\sigma_A} \cdot \left(1 - \mathrm{ld}\left[1 + e^{-\xi}\right]\right)\, d\xi \tag{7}$$

For abbreviation we define

$$J(\sigma) := I_A(\sigma_A = \sigma) \tag{8}$$

with

$$\lim_{\sigma \to 0} J(\sigma) = 0, \ \lim_{\sigma \to \infty} J(\sigma) = 1 \tag{9}$$

$$\sigma > 0 \tag{10}$$

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.

$$\sigma_A = J^{-1}(I_A) \tag{11}$$

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

$$I_E = \frac{1}{2} \cdot \sum_{x=-1,1} \int_{-\infty}^{+\infty} p_E(\xi|X=x) \times \mathrm{ld} \frac{2 \cdot p_E(\xi|X=x)}{p_E(\xi|X=-1) + p_E(\xi|X=1)}\, d\xi \tag{12}$$

$$0 \le I_E \le 1 \tag{13}$$

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

$$I_E = T(I_A, E_b/N_0) \tag{14}$$

or, for fixed E_b/N_0, just

$$I_E = T(I_A). \tag{15}$$

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 (12) are most conveniently determined by means of Monte Carlo simulation. For this, the independent Gaussian random variable of (2) is applied in conjunction with the known transmitted bits x. Note that a certain value of I_A is obtained by appropriately choosing the parameter σ_A according to (11).

Transfer characteristics based on mutual information prove to be quite robust against changes in the shape of the input distributions of A, owing to the robustness of the entropy measure [6, 7]. In an iterative decoder the actual distributions p_A 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. This property of the mutual information measure is even more needed in an IDEM scheme like [2] where the demapper output distributions can be quite asymmetric with sharp edges. There, transfer characteristics based on mean or variance values would fail.

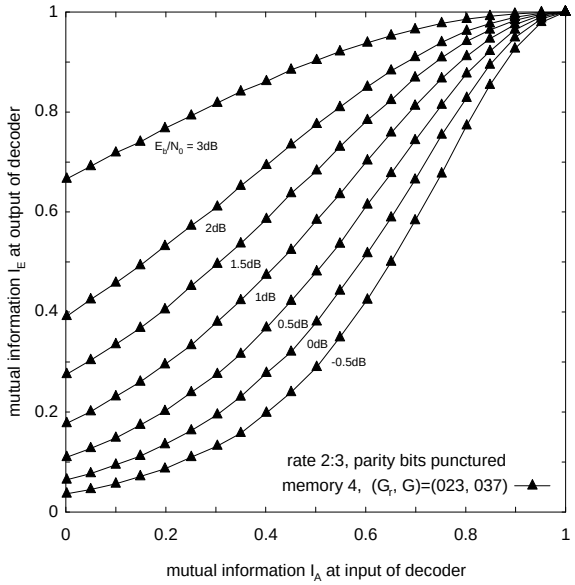

Fig. 2: Extrinsic information transfer characteristics of soft in/soft out decoder for rate 2:3 convolutional code, E_b/N_0 of channel observations as parameter to curves.

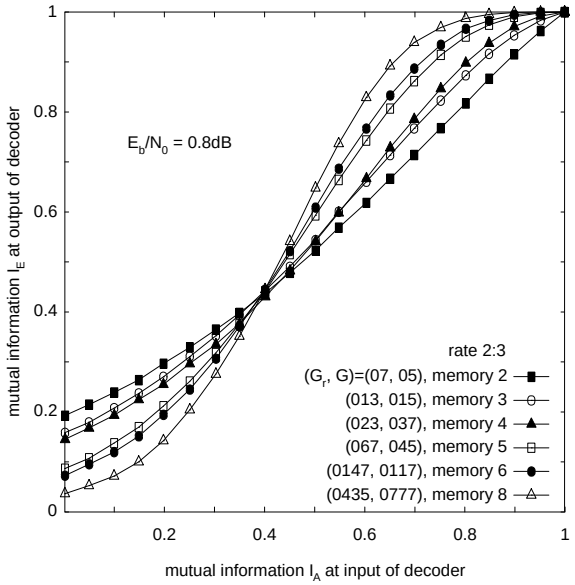

Fig. 3: Extrinsic information transfer characteristics of soft in/soft out decoder for rate 2:3 convolutional code, $E_b/N_0 = 0.8$dB, different code memory.

Transfer characteristics $I_E = T(I_A, E_b/N_0)$ 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 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 generator 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.

Transfer characteristics for different code memory at fixed $E_b/N_0 = 0.8$dB are depicted in Fig. 3. The generator polynomials are taken from [8].

Fig. 4 shows the influence of different generator 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 classic rate 1:2 PCC of [9] with generators $(037, 021)$ has good extrinsic output for low to medium *a priori* input.

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

$$I_A = T^{-1}(I_E) \tag{16}$$

exists on

$$T(0) \le I_E \le T(1). \tag{17}$$

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

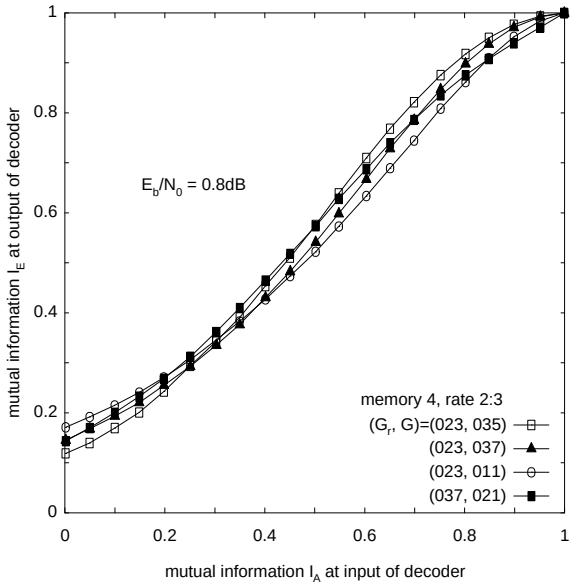

Fig. 4: Extrinsic information transfer characteristics of soft in/soft out decoder for rate 2:3 convolutional code, $E_b/N_0 = 0.8$dB, memory 4, different generator polynomials.

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 (EIT chart), since the exchange of extrinsic information can be visualized as a decoding trajectory. Provided that independence and Gaussian assumptions hold for modelling 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 EIT 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 EIT chart.

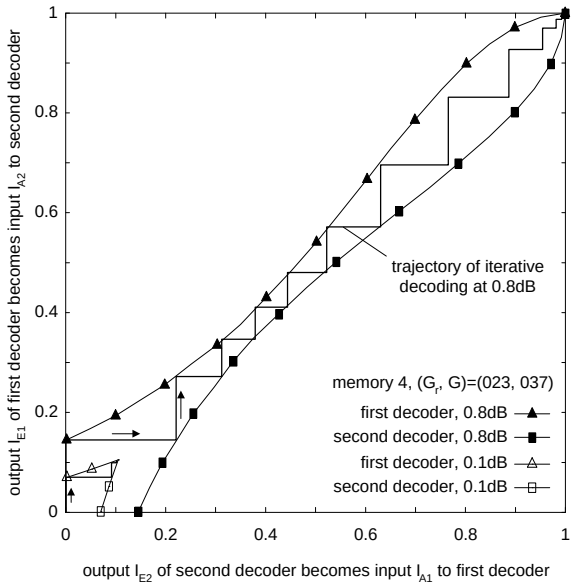

Fig. 5: Simulated trajectories of iterative decoding at $E_b/N_0 = 0.1$dB and 0.8dB (PCC rate 1:2, interleaver size 60000 information bits).

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.

It is remarkable how closely the simulated trajectory matches with the characteristics: Large interleavers ensure that the independence assumption of (2) 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.

Simulation results suggest that the EIT chart predicts the best possible convergence behavior of the iterative decoding scheme for large interleaving depth. For short interleavers the trajectory tends to diverge from the characteristics towards smaller extrinsic output after a few iterations, owing to increasing correlation of extrinsic information. A similar effect can be observed for soft in/soft out decoders like the SOVA [5]. There, the extrinsic information remains correlated with the channel observations; the simulated decoding trajectory does not keep to the characteristics.

Fig. 6 depicts the EIT chart for a set of E_b/N_0–values. The curves in between 0dB and 1dB are in 0.1dB steps. 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.

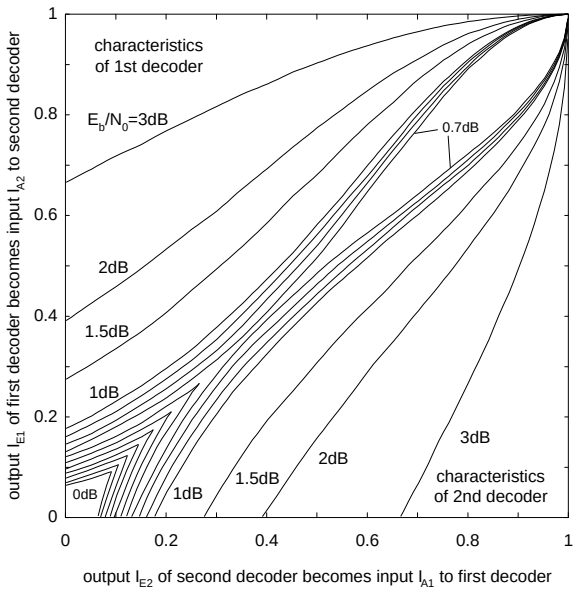

Fig. 6: EIT chart with transfer characteristics for a set of E_b/N_0–values (code parameters as in Fig. 5, PCC rate 1:2).

The main contribution of the EIT chart to the analysis of iterative decoding is the advantage that only simu-

lations of individual decoders are needed to obtain the desired transfer characteristics. These can then be used in any combination in the EIT chart to describe the behavior of the corresponding iterative decoder, asymptotic with respect to the interleaver size. No 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 [9], 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 EIT chart", becomes significant since all intersections of both decoder characteristics will happen there.

B. Obtaining BER from EIT chart

The EIT chart can be used to obtain an estimate of the BER after an arbitrary number of iterations. For both constituent decoders, the soft output on the systematic bits can be written as $D = Z + A + E$. In this section we derive a formula for the bit error probability P_b. The soft output D is assumed to be Gaussian distributed with variance σ_D^2 and mean $\mu_D = \sigma_D^2/2$, compare to (2), (3).

$$P_b \approx Q\left(\frac{\mu_D}{\sigma_D}\right) = Q\left(\frac{\sigma_D}{2}\right) \tag{18}$$

Assuming independence it is

$$\sigma_D^2 = \sigma_Z^2 + \sigma_A^2 + \sigma_E^2. \tag{19}$$

With (1) and

$$\frac{E_b}{N_0} = \frac{1}{2R\sigma_n^2} \tag{20}$$

we obtain σ_Z^2 as

$$\sigma_Z^2 = \left(\frac{2}{\sigma_n^2}\cdot\sigma_n\right)^2 = \frac{4}{\sigma_n^2} = 8R\cdot\frac{E_b}{N_0}. \tag{21}$$

Applying (11) the variances σ_A^2 and σ_E^2 are calculated.

$$\sigma_A^2 \approx J^{-1}(I_A)^2,\ \sigma_E^2 \approx J^{-1}(I_E)^2 \tag{22}$$

Finally, with (18), (19), (21) and (22) the result is

$$P_b \approx Q\left(\frac{\sqrt{8R\cdot\frac{E_b}{N_0} + J^{-1}\left(I_A\right)^2 + J^{-1}\left(I_E\right)^2}}{2}\right). \tag{23}$$

With (23) an estimate on the bit error probability can be calculated from the EIT chart. Fig. 7 shows transfer characteristics and respective simulated decoding trajectory at 0.8dB for an asymmetric PCC with memory 2 and memory 6 constituent codes: $(G_{r,1}, G_1) = (07, 05)$, $(G_{r,2}, G_2) = (0147, 0117)$. 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 EIT chart. Additionally, the BER scaling according to (23) is given as a contour plot. Table I compares BER–results obtained from the EIT 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 (23). The table shows that the EIT chart in combination with the Gaussian approximation of (23) can provide 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.

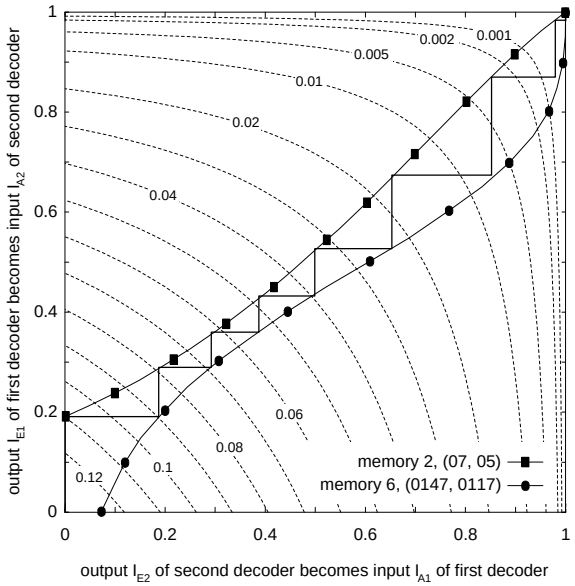

Fig. 7: Simulated trajectory of iterative decoding at $E_b/N_0 = 0.8$dB with BER scaling as contour plot (PCC rate 1:2, interleaver size 200000 information bits).

TABLE I
Comparison of BER predictions from EIT chart.
Columns of simulation results are marked with (s).

	1st decoder		2nd decoder	
pass	BER (s)	BER (23)	BER (s)	BER (23)
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

IV. Further Results

The results of Fig. 2–4 can be re–interpreted in the EIT 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. As it turns out, the trajectory of iterative decoding would find a fairly opened tunnel to creep through for symmetric PCC 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 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, 8.

Asymmetric turbo codes with constituent codes of different memory seem to be particularly attractive to design "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). Fig. 8 shows the EIT chart of the memory 2, 6–combination of Fig. 7 for a set of E_b/N_0–values. The curves below 1dB are in steps of 0.1dB. Individual BER scales according to (23) are added to the characteristics.

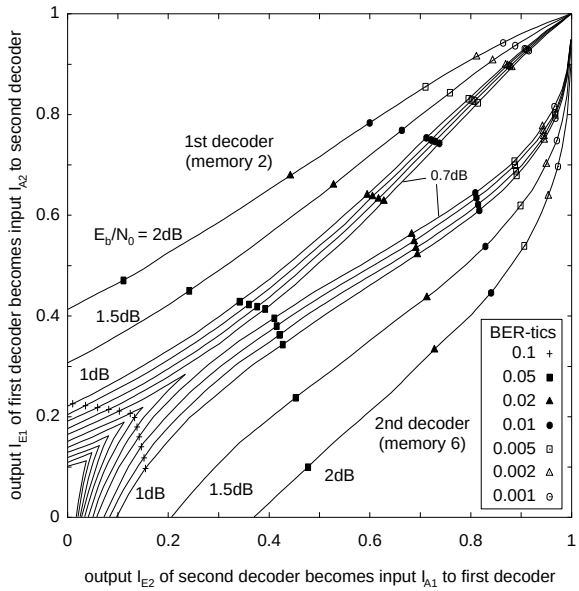

Fig. 8: EIT chart of asymmetric PCC (rate 1:2) with BER scaling on transfer characteristics.

Clearly, the BER chart for this code will have its turbo cliff at 0.7dB. For smaller E_b/N_0–values no stable convergence can be achieved, even for very large interleavers. For greater E_b/N_0–values convergence is possible and the BER floor is only determined by the interleaver size [1].

It is worth mentioning that the EIT chart of the $(037, 021)$–code of [9] (in this paper only single characteristic shown at 0.8dB, see Fig. 4) indicates convergence at 0.6dB. However, the BER chart in [9] shows a cliff at 0.7dB (interleaver size 65536 bits). Motivated by this contradiction a simulation with an interleaver size of 10^6 systematic bits was performed, and indeed, convergence was found at 0.6dB. After simulating 100 blocks (10^8 bits) the BER was $2 \cdot 10^{-7}$ (25 iterations per block). As expected from the EIT chart, no convergence could be achieved at 0.5dB.

V. Conclusion

Mutual information transfer characteristics have been shown to be particularly useful for the description of soft in/soft out decoders in the region of low E_b/N_0. The Extrinsic Information Transfer Chart was introduced as a tool to facilitate the design of parallel concatenated codes. In the case of symmetric parallel concatenated codes the computation of transfer characteristics for only one constituent decoder is sufficient to predict the performance of the corresponding iterative decoder.

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

[3] 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

[4] 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

[5] 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

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

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

[8] 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

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