Designing Iterative Decoding Schemes with the Extrinsic Information Transfer Chart

Stephan ten Brink

Abstract Since the discovery of parallel concatenated (turbo) codes, iterative decoding has become a vital field of research in digital communications. Applications of the “turbo principle” to many detection and decoding problems have been found. While most studies have focused on designing code concatenations with respect to maximum likelihood decoding performance, the convergence properties of iterative decoding schemes have gained a considerable amount of interest just recently. In this paper we use the extrinsic information transfer chart (EXIT chart) to illuminate the convergence properties of bit-interleaved coded modulation, equalization, and serially concatenated binary codes. The EXIT chart leads to new design ideas like inner and outer doping, facilitating the construction of code concatenations with iterative decoders operating close to the capacity limit.

Keywords Convergence, Iterative Decoding, Turbo Codes, Serially Concatenated Codes, Equalization

1. Introduction

Code concatenation is a way to construct long powerful codes which can achieve big coding gains while keeping the decoding complexity manageable. It was first introduced in [1] for the case of serially concatenated codes (SCC), with an “inner” and an “outer” code used in cascade. The discovery of parallel concatenated “turbo” codes (PCC) [2] considerably improved the performance gains by separating component codes through interleavers and using iterative decoding to further reduce the bit error rate (BER). The iterative processing techniques are not limited to traditional concatenated coding schemes, and the “turbo principle” [3] is more generally applicable to several other algorithms that can be found in modern digital communications, e.g. equalization [4], [5], spectrally efficient modulation [6], [7] and multiuser detection [8], [9].

So far, research has been focusing on optimizing concatenated coding schemes with respect to asymptotic slopes of error probability curves for moderate to high signal to noise ratios. Recently, people have started to investigate the convergence behavior of iterative decoding. In [10] the authors propose a density evolution algorithm to calculate convergence thresholds for randomly constructed irregular low-density parity-check (LDPC) codes on the additive white Gaussian noise (AWGN) channel. In [11], [12] the convergence of iterative decoders is studied based on signal to noise ratio measures. Mutual information to describe the flow of extrinsic information through soft in/soft out decoders was proposed in [13]; the exchange of extrinsic information between constituent codes is visualized as a decoding trajectory in the extrinsic information transfer chart (EXIT chart) for both parallel [14] and serially concatenated codes [15].

The purpose of this paper is to present the EXIT chart as a versatile tool for designing different types of serial code concatenations. The convergence behavior of iterative decoding for bit-interleaved coded modulation, equalization, and serially concatenated binary codes is studied in one common framework, illustrated by numerous figures and examples. The paper is organized as follows: In Section 2 we introduce mutual information transfer characteristics for the case of bit-interleaved coded modulation. Section 3 studies the performance of iterative equalization and decoding in the EXIT chart. The design of iterative decoders for serially concatenated codes is shown in Section 4.

2. Iterative Demapping and Decoding

2.1 System model

![Diagram of encoder, channel, and iterative decoder for bit-interleaved coded modulation with iterative decoding.](image)

Fig. 1. Encoder, channel and iterative decoder for bit–interleaved coded modulation with iterative decoding.

At the transmitter, the binary source signal is convolutionally encoded, bit-interleaved and mapped onto a complex QAM–signal constellation by a mapping device. One constellation symbol comprises a number of \( M \) outer coded bits. On the channel white Gaussian noise is added, with component–wise noise power \( \sigma^2 = N_0 \) (double–sided noise spectral density). At the receiver the signal is iteratively decoded by mutually exchanging soft (reliability) information between inner demapper and outer decoder. The demapper takes channel observations \( Z \) and \( a \...
priori knowledge \( A_1 \) on the inner unmapped bits and computes channel and extrinsic information \( E_1 \) for each of the \( M \) coded bits per constellation symbol. The extrinsic output \( E_1 \) is deinterleaved to become the a priori input \( A_2 \) to the outer soft in/soft out decoder (MAP-, APP-, BCJR-algorithm [16]) which calculates extrinsic information \( E_2 \) on the outer coded bits. \( E_2 \) is re–interleaved and fed back as a priori knowledge \( A_1 \) to the inner demapper where it is exploited to reduce the bit error rate (BER) in further iterative decoding steps. The variables \( A_1, D_1, E_1, A_2, D_2, E_2 \) denote log–likelihood ratio values (L–values [17]).

In the following the concepts of iterative demapping and decoding (IDEM) are illustrated for a simple 8–ASK (amplitude shift keying) example. 8–ASK is a multi–amplitude modulation, with real symbol amplitudes \( Y_1 \in \{ \pm 1, \pm 3, \pm 5, \pm 7 \} \). The mapping (or “labeling”) associates the \( M = 3 \) bits of the inner unmapped bit–vector \( \mathbf{X}_1 = (X_1^0, X_1^1, X_1^2) \) with one of the \( N_A = 8 \) signal amplitudes, \( Y_1 = \text{map}(X_1) \). \( Y_1 \) is also referred to as “constellation symbol”. In this paper, all amplitude levels are assumed to be equiprobable, that is, \( P_n = P[Y_1 = a_n] = 1/8 \). The extension to complex (two–dimensional) signal constellations is straightforward. Obviously, the inner mapping does not add redundancy, and thus we can define an inner “code” rate of \( R_1 = 1 \).

2.2 Characterizing Mappings by Mutual Information

2.2.1 Symbol–wise mutual information

The mutual information between transmitted constellation symbol \( Y_1 \) and received AWGN channel output \( Z \) is given by

\[
I(Y_1; Z) = \frac{1}{N_A} \sum_{n=1}^{N_A} \int_{-\infty}^{+\infty} p(z|Y_1 = a_n) \times \log_2 \left( \frac{p(z|Y_1 = a_n)}{p(z)} \right) dz \tag{1}
\]

with conditional probability density function (PDF)

\[
p(z|Y_1 = a_n) = \frac{1}{\sqrt{2\pi} \sigma} \exp \left( -\frac{(z - a_n)^2}{2\sigma^2} \right) \tag{2}
\]

and

\[
p(z) = \frac{1}{N_A} \sum_{n=1}^{N_A} p(z|Y_1 = a_n) \tag{3}
\]

We refer to the symbol–wise mutual information \( I(Y_1; Z) \) as the capacity \( C_{8\text{ASK}} \) of the modulation scheme, keeping in mind that we fixed the a priori probabilities to \( P_n = 1/N_A \). Determining the actual capacity would require a maximization of \( I(Y_1; Z) \) over the a priori probabilities \( P_n \) (“signal shaping”). The capacity \( C_{8\text{ASK}} \) is dependent on the \( E_b/N_0 \)–value of the channel; for \( C_{8\text{ASK}} \geq M \cdot R \) reliable (i.e. error–free) communication is possible (channel coding theorem [18]). The average symbol energy is \( E_s = 1/N_A \sum_{n=1}^{N_A} a_n^2 \), with \( E_b/N_0 = 1/(R \cdot M) \cdot E_s/N_0 \). Evaluating (1) for an outer code rate of \( R_2 = 1/2 \) yields \( E_s/N_{0,\text{Sch}} \approx 4.24 \text{dB} \) (“Shannon limit”) which is the lowest \( E_b/N_0 \)–value for which reliable communication is possible. It serves as the ultimate performance limit of our iterative demapping and decoding scheme.

For a given signal constellation (e.g. 8–ASK), the symbol–wise mutual information \( I(Y_1; Z) \) is independent of the applied mapping, and therefore it is not a suitable measure for characterizing different mappings.

2.2.2 Bitwise mutual information

With the chain rule of mutual information [18] it can be shown that the symbol–wise mutual information \( I(X_1; Z) = I(Y_1; Z) \) can be decomposed into a sum of \( M \) bitwise mutual informations \( I_L \).

\[
I(X_1; Z) = \sum_{L=0}^{M-1} I_L = C_{8\text{ASK}} \tag{4}
\]

\( I_L \) is a short–hand notation of

\[
I_L = I(X_1^m; Z | L \text{ other bits known}) \tag{5}
\]

\( 0 \leq m < M; 0 \leq I_L \leq 1 \tag{6} \)

A number of \( L \) “other” unmapped bits \( X_1^m, n \neq m \) of the mapping are perfectly known, whereas no knowledge is available for the remaining \( M - 1 - L \) bits. The bar in (5) indicates that \( I_L \) is averaged a) over bitwise mutual information with respect to all bits of the mapping, \( X_1^m, 0 \leq m < M \), b) over all possible \( \binom{M-1}{L} \) combinations to choose \( L \) known bits out of the total of \( M - 1 \) other bits of the mapping, and c) over all \( 2^L \) bit vector realizations thereof. The vector channel with mutual information \( 0 \leq I(X_1; Z) \leq M \) carrying \( M \) bits can be viewed as being composed of \( M \) parallel sub–channels with mutual information \( 0 \leq I_L \leq 1 \), each carrying a single bit (concept of equivalent channels [19]).

The chain rule in formulation (4) allows an interesting interpretation: The mapping only influences the partitioning of the total amount of mutual information \( C_{8\text{ASK}} \) among the different conditional sub–channels \( I_L \), whereas the sum \( \sum I_L \) always adds up to the constant value \( C_{8\text{ASK}} \), independently of the applied mapping. Hence, the quantities \( I_L \) are well suited for characterizing different mappings.

![Fig. 2. 8–ASK signal constellation with 5 different mappings.](image)
From the many possible 8-ASK mappings we chose the five different mappings depicted in Fig. 2 for further consideration. For each mapping the number of changing digits in between neighboring signal amplitudes is given. With Gray mapping, the unmapped bit–vectors \( \mathbf{X}_1 \) of neighboring signal amplitudes differ by one binary digit which keeps the BER small if the demapper is implemented as a simple slicer; a slicer provides hard decision estimates \( \hat{a} \) on the received symbol amplitude by selecting that \( a_n \) which minimizes \( | \hat{z} - a_n | \), \( 0 \leq n < N_A \). This is the reason why traditionally Gray mapping is used with spectral efficient modulation. However, from simulations we observed that Gray mapping does not perform well in an IDEM scheme, and other, formerly less popular mappings like anti Gray mapping turn out to be more suited.

### Table 1. Bitwise conditional mutual information \( I_{\ell} \) for different 8-ASK mappings at \( E_b/N_0 = 6 \text{dB} \) (code rate 1/2).

<table>
<thead>
<tr>
<th>8-ASK mappings</th>
<th>( I_0 )</th>
<th>( I_1 )</th>
<th>( I_2 )</th>
<th>( I_{L-2} - I_{L-1} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gray</td>
<td>0.561</td>
<td>0.580</td>
<td>0.593</td>
<td>1.74</td>
</tr>
<tr>
<td>natural</td>
<td>0.448</td>
<td>0.599</td>
<td>0.699</td>
<td>1.74</td>
</tr>
<tr>
<td>“d2”</td>
<td>0.312</td>
<td>0.517</td>
<td>0.911</td>
<td>1.74</td>
</tr>
<tr>
<td>“d2”</td>
<td>0.253</td>
<td>0.536</td>
<td>0.951</td>
<td>1.74</td>
</tr>
<tr>
<td>anti Gray</td>
<td>0.159</td>
<td>0.707</td>
<td>0.874</td>
<td>1.74</td>
</tr>
</tbody>
</table>

A first hint to understanding this seemingly counterintuitive result can be found in Table 1. It shows the conditional mutual information \( I_{L} \) for the mappings of Fig. 2 at \( E_b/N_0 = 6 \text{dB} \) (values in bit per constellation symbol). For Gray mapping the information transfer through the demapper does not change much with increasing a priori knowledge (i.e. number of \( L \) known bits of the mapping), and \( I_{L-2} - I_{L-1} \) stays fairly small. For the other mappings the span between \( I_0 \) and \( I_2 \) is much bigger. Note that conditioning increases mutual information [18], \( I_{L} \geq I_{L-1} \).

Apparently, the partitioning of mutual information among the conditional sub–channels \( I_{\ell} \) has a strong impact on the behavior of the particular mapping in an iterative decoding scheme. For a more comprehensive study of the iterative decoding process we introduce demapper transfer characteristics based on mutual information.

### 2.3 Transfer Characteristics of the Demapper

In the previous Section we have seen that the information transfer through the demapper can be controlled by the amount of available a priori knowledge. The characteristic behavior is determined by the choice of the mapping. However, so far we were only considering “discrete” levels of a priori knowledge, that is, the a priori knowledge was quantized to a certain number of \( L \) known, and \( M - 1 - L \) unknown other bits of the mapping. To perform the transition to “continuous” a priori knowledge we refrain from further distinguishing between particular bits \( X_1^{(\ell)} \) of the mapping, but rather focus on the resulting bit–channel with mutual information \( I(X_1; E_1) \) between unmapped bits \( X_1 \) and demapped channel and extrinsic L–values \( E_1 \).

The inputs to the demapper are the noise–corrupted channel observations \( Z \) and the a priori knowledge \( A_1 \) on the unmapped bits. The demapper outputs channel and extrinsic information \( E_1 \). The calculation of the demapper output \( L–values \) \( E_1 \) can be found, e. g., in [7]. From simulations of the iterative decoder we observed that the extrinsic information \( L–values \) \( E_2 \) (i.e. \( A_1 \)) as fed back from the outer decoder are almost Gaussian distributed. Additionally, large interleavers keep the a priori \( L–values \) \( A_1 \) fairly uncorrelated over many iterations. Hence, it seems appropriate to model the a priori input \( A_1 \) by an independent Gaussian random variable \( n_A \) with variance \( \sigma^2_{A_1} \) and mean zero. In conjunction with the known transmitted unmapped bits \( x_1 \in \{ \pm 1 \} \) we write

\[
A_1 = \mu_{A_1} \cdot x_1 + n_{A_1}.
\]

Since \( A_1 \) is supposed to be an \( L–value \) based on Gaussian distributions it can be shown [17] that \( \mu_{A_1} \) must fulfill \( \mu_{A_1} = \sigma^2_{A_1} / 2 \), and thus the conditional probability density function is

\[
p_{A_1}(\xi|X_1 = x_1) = \frac{1}{\sqrt{2\pi \sigma_{A_1}}} e^{-\frac{(\xi - \mu_{A_1})^2}{2\sigma^2_{A_1}}}.
\]

To measure the information content of the a priori knowledge, mutual information \( I_{A_1} = I(X_1; A_1), 0 \leq I_{A_1} \leq 1 \), between transmitted unmapped bits \( X_1 \) and the \( L–values \) \( A_1 \) is used [18].

\[
I_{A_1} = \frac{1}{2} \sum_{x_1 = -1, 1} \int_{-\infty}^{+\infty} p_{A_1}(\xi|X_1 = x_1) \times \frac{2 \cdot p_{A_1}(\xi|X_1 = x_1)}{p_{A_1}(\xi|X_1 = -1) + p_{A_1}(\xi|X_1 = 1)} d\xi
\]

With (8), equation (9) becomes

\[
I_{A_1} = 1 - \int_{-\infty}^{\infty} \frac{1}{\sqrt{2\pi \sigma_{A_1}}} e^{-\frac{(\xi - \mu_{A_1})^2}{2\sigma^2_{A_1}}} \cdot \ln [1 + e^{-\xi}] d\xi.
\]

For abbreviation we define

\[
J(\sigma) := I_{A_1}(\sigma_{A_1} = \sigma)
\]

\[
\lim_{\sigma \to 0} J(\sigma) = 0, \quad \lim_{\sigma \to \infty} J(\sigma) = 1, \quad \sigma > 0
\]

The function \( J(\sigma) \) cannot be expressed in closed form. It is monotonically increasing and thus reversible.

Mutual information is also used to quantify the extrinsic output \( I_{E_1} = I(X_1; E_1) \); it is computed according to (9), but now using the extrinsic output PDF \( p_{E_1} \). Viewing \( I_{E_1} \) as a function of \( I_{A_1} \) and the \( E_b/N_0–value \), the inner extrinsic information transfer characteristics are defined as

\[
I_{E_1} = T_1(I_{A_1}, E_b/N_0).
\]

To calculate the characteristic \( T_1(I_{A_1}, E_b/N_0) \) for a desired \( (I_{A_1}, E_b/N_0)–input \) combination, the distributions \( p_{E_1} \) are most conveniently determined by means of Monte
Carlo simulation. For this, the independent Gaussian random variable of \( \sigma_{A_1} = J^{-1}(I_{A_1}) \) is applied as a priori input to the demapper of interest.

![Demapper transfer characteristics at E_b/N_0 = 6dB](image)

Fig. 3. Demapper transfer characteristics of 8-ASK mappings at \( E_b/N_0 = 6dB \).

In Fig. 3 the corresponding demapper transfer characteristics are given. The values \( I_0, I_2 \) of Table 1 show up as \( I_0 \approx I_E(I_{A_1} = 0) \) and \( I_2 \approx I_E(I_{A_1} = 1) \). Different mappings result in transfer characteristics of different slopes. However, the mapping with lowest \( I_0 \) does not necessarily exhibit the highest \( I_2 \) (see anti Gray mapping).

2.4 Transfer Characteristics of the Outer Decoder

The outer extrinsic transfer characteristic is

\[
I_{E_0} = T_2(I_{A_2})
\]

(14)

and describes the input/output relationship between outer coded input \( A_2 \) and outer coded extrinsic output \( E_2 \). It is not dependent on the \( E_b/N_0 \)-value. It can be computed by assuming \( A_2 \) to be Gaussian distributed and applying the same equations as presented for the demapper characteristic \( T_1 \) in the previous Section.

Fig. 4 shows extrinsic transfer characteristics of some outer rate 1/2 recursive systematic convolutional (RSC) codes. The generator polynomials (feedback polynomial \( G_r \), feedforward polynomial \( G \)) are given in octal numbers, with the most significant bit denoting the very left (input) connection to the shift register. Note that the axes are swapped: The input \( I_{A_1} \) is on the ordinate, the output \( I_{E_0} \) on the abscissa.

2.5 Extrinsic Information Transfer Chart

To visualize the exchange of extrinsic information we plot demapper and decoder characteristics into a single diagram which is referred to as Extrinsic Information Transfer Chart. On the ordinate, the inner channel and extrinsic output \( I_{E_i} \) becomes the outer a priori input \( I_{A_i} \) (interleaving does not change mutual information). On the abscissa, the outer extrinsic output \( I_{E_0} \) becomes the inner a priori input \( I_{A_1} \). Provided that independence and Gaussian assumptions hold for modelling extrinsic information (a priori information respectively), the transfer characteristics of the individual demapping/decoding blocks should approximately describe the true behavior of the iterative decoder. To verify the EXIT chart predictions we use the trajectory of iterative decoding which is a simulation result of the compound "free–running" iterative decoder.

Fig. 5 shows EXIT charts of the "d21"–mapping in combination with an outer memory 2 code. The transfer characteristics are just taken from Figs. 3, 4. At \( E_b/N_0 = 5dB \) the demapper and decoder transfer characteristics do intersect, and the decoding trajectory gets stuck after about three iterations at low mutual information, corresponding to high BER. With increasing \( E_b/N_0 \)-value, the demapper transfer characteristic is raised, opening a narrow tunnel ("bottleneck") at about6dB to allow for convergence of iterative decoding towards low BER. This turbo cliff effect is verified in the BER chart of Fig. 7. For higher \( E_b/N_0 \)-values it just takes less iterations to reach a low BER. A closer examination based on demapper transfer characteristics yields a pinch–off limit at \( E_b/N_0 |_{\text{off}} \approx 5.64dB \) (Table 2) where both transfer characteristics are just about to intersect. Simulation results suggest that the iterative decoder converges for \( E_b/N_0 \)-values above the pinch–off limit, provided that the interleaving depth is sufficiently large.

For non–recursive inner “codes”, like the mapping in an IDEM scheme, the BER floor is dominated by the in-
tersection of demapper and decoder transfer characteristic on the right hand side of the EXIT chart. For recursive inner codes, like used in the serially concatenated codes of Section 4, the inner transfer characteristics go up to $(I_A, I_E) = (1, 1)$, and the interleaver size becomes the most crucial parameter [20].

The simulated trajectories match with the characteristics very well, owing to the large interleaver which ensures that the independence assumption of (7) holds over many iterations; in addition to that, the robustness of the mutual information measure allows to overcome non–Gaussian distributions of a priori information. It should be emphasized, however, that the decoding trajectory is a simulation result purely based on measurements of mutual information as taken from the output of the respective demapping/decoding block. Only to calculate the transfer characteristics of demapper and decoder we were sought to impose the Gaussian and independence assumption on the a priori inputs $A_1, A_2$.

Table 2. Pinch–off limits $E_b/N_0$ [dB] of some IDEM schemes.

<table>
<thead>
<tr>
<th>8–ASK mappings</th>
<th>outer codes $(G_r, G)$</th>
<th>memory 1</th>
<th>memory 2</th>
<th>memory 4</th>
<th>memory 8</th>
</tr>
</thead>
<tbody>
<tr>
<td>“d21”</td>
<td>$(00, 02)$</td>
<td>4.80</td>
<td>5.64</td>
<td>6.15</td>
<td>6.98</td>
</tr>
<tr>
<td>“d23”</td>
<td>$(01, 05)$</td>
<td>4.83</td>
<td>6.22</td>
<td>6.85</td>
<td>7.79</td>
</tr>
<tr>
<td>anti Gray</td>
<td>$(00, 02)$</td>
<td>4.64</td>
<td>6.21</td>
<td>7.93</td>
<td>8.88</td>
</tr>
</tbody>
</table>

2.6 Bit Error Rate Chart

Fig. 7 shows the BER chart of iterative demapping and decoding for 8–ASK with the 5 different mappings of Fig. 2 in combination with an outer memory 2 code. For Gray mapping, almost no BER improvement can be achieved by performing iterative demapping, owing to the weak slope of the demapper transfer characteristic: Perfect a priori knowledge $I_{A_1}$ does not significantly improve $I_{E_1}$ at the demapper output. Therefore, only the BER curve of Gray mapping for no iterations is plotted. For the other mappings, a strong BER reduction can be triggered by iterative demapping, depending on the slope of the demapper characteristic: The steeper the slope, the later (in terms of $E_b/N_0$–value) the turbo cliff in the BER chart, but, typically, the lower the BER floor. At a BER of $10^{-5}$ the
advantage of the “d21”/memory 2 combination in comparison to the Gray/memory 2 system is more than 7dB; the distance to the Shannon limit is about 1.8dB. The BER curve of the “d23”/memory 1 combination further illustrates the trade–off “early turbo cliff versus low BER floor”: The EXIT chart of Fig. 6 shows that the characteristic of the “d23”–mapping is a good match to the outer memory 1 decoder curve. Consequently, the pinch–off limit is rather low \((E_b/N_0)_{\text{off}} \approx 4.83\text{dB}\), but the BER floor is raised.

Note that for all BER charts in this paper we simulated \(10^7\) information bits per \(E_b/N_0\)-value.

### 3. Iterative Equalization and Decoding

Some channels, like the wireless communication channel, show effects of multipath propagation. The superposition of delayed replicas of the transmitted signal (echoes) at the receiver causes intersymbol interference which limits the minimal symbol duration, and thus the maximal achievable bit rate. At the receiver, equalization is used to compensate for the effects of multipath propagation. Iterative equalization and decoding of multipath channels was proposed in [4], [5].

The multipath channel can be modeled as a tapped delay line (Fig. 8) with \(n_{ch}\) delay elements; the parameter \(n_{ch}\) is also referred to as channel memory. The tapped delay line model closely resembles the structure of a convolutional encoder; however, arithmetic additions rather than modulo–2 additions are used. The tap coefficients \(h_i\), \(i=0, \ldots, n_{ch}\), are complex–valued and normalized to one, \(\sum_{i=0}^{n_{ch}} |h_i|^2 = 1\).

For the transmitter, the AWGN–channel and the receiver we use the system configuration of Fig. 1, with the following changes: 1. The multipath channel is viewed as being part of the transmitter, replacing the mapping device, and 2. the demapper at the receiver is substituted by the corresponding channel equalizer. The multipath channel does not add redundancy, and thus we set the inner “code” rate to \(R_{1}=1\).

The equalizer calculates a posteriori \(L\)-values on the transmitted symbols \(x_1\). It is based on the BCJR–algorithm and performs a trellis processing over \(2^n_{ch}\) states, using the logarithmic metric increment

\[
\gamma_k = -\frac{1}{2\sigma^2} \left| z_k - \sum_{i=0}^{n_{ch}} h_i \cdot \hat{x}_{1,k-i} \right|^2
\]  

with received signal \(z_k\) and hypothesized transmitted outer coded bits \(\hat{x}_{1,k-i}\) according to the current state in the equalization trellis. In the following we further study the iterative equalization of multipath channels with tap coefficients according to Table 3. For simplicity we chose time–invariant channels with real coefficients \(h_i\) and a symmetric discrete–time impulse response of triangular shape.

### 3.1 Transfer Characteristics of Equalizers

![Transfer characteristics of APP equalizer for multipath channels](image)

Corresponding to the demapper of Section 2.3, we can describe the APP equalizer by a mutual information transfer characteristic. This provides further insights into the behavior of the iterative equalization and decoding algorithm. Transfer characteristics of APP equalizers for the multipath channels of Table 3 are depicted Fig. 9. With increasing channel memory \(n_{ch}\), the transmitted bits \(x_1\) get more and more tight up by the convolution with the channel impulse response \(h_i\), and the beginning of the

<table>
<thead>
<tr>
<th>(n_{ch})</th>
<th>(h_0)</th>
<th>(h_1)</th>
<th>(h_2)</th>
<th>(h_3)</th>
<th>(h_4)</th>
<th>(h_5)</th>
<th>(h_6)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>(2/3)</td>
<td>(-1/2)</td>
<td>(1/2)</td>
<td>(1/2)</td>
<td>(1/2)</td>
<td>(-1/2)</td>
<td>(-1/2)</td>
</tr>
<tr>
<td>2</td>
<td>(\sqrt{2}/2)</td>
<td>(-\sqrt{2}/2)</td>
<td>(\sqrt{2}/2)</td>
<td>(\sqrt{2}/2)</td>
<td>(-\sqrt{2}/2)</td>
<td>(-\sqrt{2}/2)</td>
<td>(-\sqrt{2}/2)</td>
</tr>
<tr>
<td>3</td>
<td>(\sqrt{3}/3)</td>
<td>(-\sqrt{3}/3)</td>
<td>(\sqrt{3}/3)</td>
<td>(\sqrt{3}/3)</td>
<td>(-\sqrt{3}/3)</td>
<td>(-\sqrt{3}/3)</td>
<td>(-\sqrt{3}/3)</td>
</tr>
<tr>
<td>4</td>
<td>(\sqrt{4}/4)</td>
<td>(-\sqrt{4}/4)</td>
<td>(\sqrt{4}/4)</td>
<td>(\sqrt{4}/4)</td>
<td>(-\sqrt{4}/4)</td>
<td>(-\sqrt{4}/4)</td>
<td>(-\sqrt{4}/4)</td>
</tr>
<tr>
<td>5</td>
<td>(\sqrt{5}/5)</td>
<td>(-\sqrt{5}/5)</td>
<td>(\sqrt{5}/5)</td>
<td>(\sqrt{5}/5)</td>
<td>(-\sqrt{5}/5)</td>
<td>(-\sqrt{5}/5)</td>
<td>(-\sqrt{5}/5)</td>
</tr>
<tr>
<td>6</td>
<td>(\sqrt{6}/6)</td>
<td>(-\sqrt{6}/6)</td>
<td>(\sqrt{6}/6)</td>
<td>(\sqrt{6}/6)</td>
<td>(-\sqrt{6}/6)</td>
<td>(-\sqrt{6}/6)</td>
<td>(-\sqrt{6}/6)</td>
</tr>
</tbody>
</table>

![Transfer characteristics of APP equalizer for multipath channels](image)

Fig. 9. Transfer characteristics of APP equalizer for multipath channels of memories \(n_{ch}=1, \ldots, 6\) at \(E_b/N_0 = 4\text{dB}\) (code rate 1/2).
equalizer transfer characteristic $I_{E_1}(0)$ is lowered. Independently of the channel memory, the equalizer transfer characteristics approach the capacity $C_G \approx 0.79$ ($E_b/N_0 = 4$ dB) of the perfectly equalized AWGN channel for $I_{A_1} \to 1$. Note that for computing the equalizer transfer characteristics, the channel coefficients $h_i$ are assumed to be known at the receiver.

Recalling the EXIT chart interpretation of the iterative demapping and decoding scheme we realize that a small slope $I_{E_1}(1) - I_{E_1}(0)$ of the equalizer characteristic (i.e. small channel memory) corresponds to a weak potential performance improvement through iterative equalization, resulting in a less pronounced turbo cliff in the BER chart. Conversely, a bad channel with big slope $I_{E_1}(1) - I_{E_1}(0)$ (i.e. big channel memory) offers a significant performance improvement.

### 3.2 Extrinsic Information Transfer Chart

![EXIT chart of APP equalizer for multipath channel of memory 4 in combination with an outer convolutional decoder of rate $1/2$.](image)

Fig. 10. EXIT chart of APP equalizer for multipath channel of memory 4 in combination with an outer convolutional decoder of rate $1/2$.

An EXIT chart for iterative equalization with decoding trajectory at $E_b/N_0 = 4$ dB is shown in Fig. 10. The equalizer transfer characteristic at 4 dB is taken from Fig. 9, the outer decoder characteristic from Fig. 4. The trajectory is a simulation result of the compound iterative decoding scheme. The pinch–off limit turns out to be at $E_b/N_0 \approx 3.58$ dB which is in good agreement with the respective BER curve of Fig. 11. Additionally, the transfer characteristic of an outer memory 1 decoder is given in Fig. 10 as a dashed line. In comparison to the equalizer characteristic, the slope of the memory 1 decoder characteristic is steeper, which removes bottleneck effects; however, the BER floor is higher, as the trajectory gets stuck at lower mutual information.

The channel is given by the environment (e.g. wireline, wireless) and the signaling rate (symbol duration), and thus the memory of the communication channel is not a design parameter. However, by proper choice of the outer decoder we can optimize the receiver with respect to the performance of iterative equalization.

### 3.3 Bit Error Rate Chart

![BER curves of iterative equalization and decoding for different channel memories; interleaver size $2 \cdot 10^5$ outer coded bits.](image)

Fig. 11. BER curves of iterative equalization and decoding for different channel memories; interleaver size $2 \cdot 10^5$ outer coded bits.

The BER curves of Fig. 11 show the achievable performance improvements through iterative equalization and decoding. Note that “no iteration” corresponds to one pass through equalizer and decoder. Considering that no change of the coding scheme on the transmitting side is required, the advantage turns out to be quite significant: At a BER of $10^{-5}$ the gain is about 2 dB for a memory 1 channel, 3 dB for a memory 2 channel, 5 dB for a memory 3 channel, and 6.5 dB for the case of a memory 4 channel.

### 4. Iterative Decoding of Serially Concatenated Codes

For serially concatenated codes, the mapper of Fig. 1 is replaced by a binary encoder of rate $R_1$, and the demapper by the respective APP decoder.

### 4.1 Extrinsic Information Transfer Charts

In this Section we present some examples for rate $1/3$ code designs, using different rate splittings between inner and outer code.

Fig. 12 shows the encoder structure of a serially concatenated code with outer rate $R_2 = 1/2$ code in combination with an inner rate $R_1 = 2/3$ code. For both component decoders the same rate $1/2$ memory 2 mother RSC code is used; however, for the inner code every other...
parity bit (i.e., output from the recursive shift register) is punctured. The corresponding EXIT chart is given in Fig. 13. The outer decoder transfer characteristic is taken from Fig. 4. The pinch–off limit for this code concatenation is at $E_b/N_{0\text{off}} \approx 0.28$ dB.

A search over inner rate 1 codes up to memory 6 yields a matching pair of inner memory 2 code with polynomials $(G_r, G) = (06, 07)$ and outer rate 1/3 repetition code. The encoder structure is given in Fig. 15. Systematic doping is used to make the inner rate 1 code suitable for iterative decoding. With inner systematic doping, some of the coded bits at the output of the recursive shift register are substituted by their systematic counterpart. This opens up an entry point $I_{E_{1}}(0) > 0$ to initiate convergence of iterative decoding. Without systematic doping, the inner transfer characteristic would start at the origin $I_{E_{1}}(0) \approx 0$, preventing the iterative decoder from convergence.

The EXIT chart of this code concatenation is depicted in Fig. 16. With an inner systematic doping ratio of $r_d = 1/50$ (every 51st coded bit is replaced by its corresponding systematic bit) we achieve a pinch–off limit of $E_b/N_{0\text{off}} \approx -0.34$ dB. At $E_b/N_0 = -0.2$ dB we performed a simulation over $10^8$ information bits (interleaver length $3 \cdot 10^5$ bits, 100 iterations) without measuring a bit error. Owing to the small code memory, the iter-
itive decoder is of very low complexity. For comparison: The classic rate 1/3 PCC of [2] with constituent codes of memory 4, \((G_r, G) = (037, 021)\) has a pinch-off limit at \(E_b/N_0\text{off} \approx -0.13\text{dB}\), that is, no convergence is possible at \(-0.2\text{dB}\).

Another opportunity to initiate convergence is to alter the transfer characteristic of the outer decoder, to achieve \(I_{E_b}(0) > 0\). This can be accomplished by replacing some of the outer rate 1/3 repetition codewords by rate 1/2 repetition codewords and using the remaining redundancy as pilot symbols, with predetermined values \pm 1\) which must be known to the receiver. As the receiver has perfect knowledge of some of the outer coded bits, we refer to this method as perfect outer doping. The respective encoder structure is shown in Fig. 17 (corresponding EXIT chart not shown).

Fig. 17. Serially concatenated code consisting of outer rate 1/3 repetition code with perfect doping and inner rate 1 memory 2 recursive convolutional code.

Fig. 16. EXIT chart of inner rate 1 memory 2 code in combination with outer rate 1/3 repetition code; doping ratio \(r_d = 1/50\).

4.2 Bit Error Rate Chart

The BER chart of Fig. 18 compares the performance of two different rate 1/3 serially concatenated codes with the classic rate 1/3 parallel concatenated code of [2]; the codeword length is \(3 \cdot 10^6\) bits. Note that more iterations do not significantly improve the BER performance. The pinch-off limits are determined using the EXIT chart technique; they nicely agree with the BER curves. Increasing the codeword length results in steeper turbo cliffs, the positions of which being well approximated by the respective pinch-off limits.

4.3 Other Code Rates

The concept of inner systematic doping can be applied to other code rates as well. A rate 1/2 code consisting of an inner rate 1 memory 3 code \((G_r, G) = (017, 07)\) (inner doping ratio \(r_d = 1/100\)) with a pinch-off limit as low as 0.27dB (Shannon limit 0.19dB) was presented in [22]. For a code rate of \(R = 2/3\) we found an inner rate 1 memory 2 code with \((G_r, G) = (07, 03)\) which goes very well together with an outer rate 2/3 single parity check code. For an inner systematic doping ratio of \(r_d = 1/20\), the pinch-off limit is at \(E_b/N_0\text{off} \approx 1.13\text{dB}\) which is only 0.07dB away from the Shannon limit \(E_b/N_0\text{off} \approx 1.06\text{dB}\). Rather than using inner systematic doping, outer perfect doping can be applied to these codes as well.

5. Conclusions

The EXIT chart was presented as a simple yet powerful tool to study the convergence behavior of a variety of iterative decoding schemes, not restricted to binary concatenations of error correcting codes. Transfer characteristics based on mutual information can be used to verify if the soft in/soft out decoding blocks interact efficiently. This
allows to optimize concatenated coding schemes with respect to the performance of the corresponding iterative decoder.

Acknowledgement

This work was carried out in a joint project of Bell Laboratories and the Institute of Telecommunications (Nachrichtenübertragung), University of Stuttgart, Germany.

References


Stephan ten Brink was born in Backnang, Germany, in 1970. He received the Dipl.-Ing. degree in Electrical Engineering and Information Technology from the University of Stuttgart, Germany in 1997. From 1997 to 2000 he was a research assistant at the Institute of Telecommunications, working towards the doctoral degree. Since November 2000 he is with the Global Wireless Systems Research Department of Bell Laboratories, Lucent Technologies in Holmdel, USA. His research interests include error correcting coding and channel estimation for wireless communication systems.