Design of serially concatenated systems depending on the block length

Michael Tüchler

Parts of this work have been published in the proceedings of the Conference for Information Sciences and Systems 2002 in Princeton, U.S.A., and have been submitted to the International Conference on Communications 2003 in Anchorage, U.S.A.

The author is with the Institute for Communications Engineering, Munich University of Technology, Arcisstr. 21, 80290 Munich, Germany, phone:+49-89-289-23479, email: micha@lnt.ei.tum.de.

Abstract

We study the convergence behavior of iterative decoding for a number of serially concatenated systems, such as a serially concatenated code, coded data transmission over an inter-symbol interference channel, bit-interleaved coded modulation, or trellis-coded modulation. We rederive an existing analysis technique called EXIT chart, simplify its construction, and construct simple irregular codes to improve the convergence of iterative decoding. An efficient and optimal optimization algorithm yields systems, which approach information theoretic limits very closely. However, these systems exhibit their performance only for very long block lengths. To overcome this problem, we optimize the decoding convergence after a fixed, finite amount of iterations yielding systems, which perform very well for short block lengths, too. As an example, optimal system configurations for communication over an additive white Gaussian noise channel are presented.

Keywords

Turbo decoding, bit-interleaved coded modulation, trellis-coded modulation, Turbo equalization, concatenated codes

I. INTRODUCTION

Berrou et al. showed in 1993 that a concatenated error-correction code (ECC) code can be decoded almost optimally with low computational burden using iterative decoding [1]. This decoding principle was extended later to other communication systems, such as coded data transmission over an inter-symbol interference (ISI) channel [2], bit-interleaved coded modulation (BiCM) [3, 4], or trellis coded modulation (TCM) [5, 6], to name only a few. The concept of iterative decoding can be generalized to message-passing on a graph [7], which was used to analyze the family of the low-density parity-check (LDPC) codes [8] originally described in [9]. A major question for understanding iterative decoding is to explain the regions in signal-to-noise ratio (SNR), where performance improvement over the iterations occurs or not at all. These regions are separated by a transition called *waterfall*. A successful approach to this problem called *density evolution* was pioneered in [10, 11], which investigates the probability density functions (PDFs) of the communicated information within the iterative decoding algorithm. Based on the evolution of these PDFs, a design algorithm [11] and convergence thresholds [10] for LDPC codes have been obtained. Other

approaches are simpler, e.g., by assuming that the PDFs considered above are Gaussian [12] or by observing only a single parameter of them [13–16]. Among the latter are the extrinsic information transfer (EXIT) charts [14, 17], which have been applied successfully to various concatenated systems [3, 18, 19], due its high accuracy reported in [20].

This paper aims on the following. We devise rules to design a concatenated system based on the EXIT chart analysis. A new class of irregular codes with managable encoding and decoding complexity is proposed, which is broader and more flexible than similar classes [21–23]. We derive a simple optimization algorithm using EXIT charts to design systems approaching information theoretic performance limits as close as possible. The applied optimization tools are similar to those for LDPC codes [11] or for repeat-accumulate (RA) codes [24]. A new optimization criterion is proposed to optimize decoding convergence after a fixed number if iterations. Using this criterion yields systems, which perform well for short block lengths, too, and even surpass the systems optimized for convergence close to channel capacity. Reasons for this behavior are discussed. Results are presented for a serially concatenated system designed for communication over an additive white Gaussian noise (AWGN) channel. We consider only serial concatenations in this paper and note that the derived techniques can also be applied to parallel concatenated systems [25].

II. A SERIALLY CONCATENATED SYSTEM

Fig. 1 depicts a serially concatenated system utilizing iterative decoding in the receiver. The encoder I is usually a ECC. Depending on the encoder II, the system or its receiver algorithm, respectively, is referred to as Turbo code (encoder II is a ECC), Turbo equalization (ISI channel), Turbo BiCM (mapper), or Turbo TCM (modulation code) in the literature. A block of K data bits is encoded with the rate-R_{I} *outer* encoder I to L code bits c_n. The set of all code words $\mathbf{c} = (c_1, ..., c_n, ..., c_L)$ is denoted $\mathcal{C}$. The interleaver, a fixed permutation on L elements, permutes the bits in $\mathbf{c}$ to $\mathbf{x} = (x_1, ..., x_L)$. The deinterleaver reverses the interleaver permutation. The x_n are encoded with the rate-R_{II} *inner* encoder II to N code bits y_n. The alphabet of the c_n, x_n, and y_n is $\{0, 1\}$ and operations on them are in $\mathbb{F}_2$. The total rate of the concatenated system is

$$R = K/N = R_{\mathrm{I}} \cdot R_{\mathrm{II}}. \tag{1}$$

A Q-tuple $\mathbf{y}_k = (y_{Qk+1}, ..., y_{Qk+Q})$ of bits y_n is mapped to a symbol $S(\mathbf{y}_k) \in \mathbb{C}$ from the 2^Q-ary signal constellation $\mathcal{S}$. The average power of the symbols in $\mathcal{S}$ is P. All results in this paper assume an AWGN channel specified by the noise variance σ^2. The noise samples w_k are independent and identically distributed (i.i.d.) with $f_w(w) = 1/(\pi\sigma^2) \cdot \exp(-|w|^2/\sigma^2)$, $w \in \mathbb{C}$. Besides the plain AWGN channel, where $z_k = S(\mathbf{y}_k) + w_k$ is received, we briefly mention Fading and ISI channels in Sec. VII. The signal-to-noise ratio (SNR) at the receiver input is defined as

$$\frac{E_b}{N_0} = \frac{E_s}{N_0} \cdot \frac{1}{R \cdot Q} = \frac{P}{\sigma^2} \cdot \frac{1}{R \cdot Q}.$$

The two decoders communicate log-likelihood ratios (LLRs) [25] on the code bits c_n or x_n. Input to decoder II are the *a-priori* LLRs $L(x_n) = \ln\left(P(x_n = 0)/P(x_n = 1)\right)$ and z_k. Using bit-based a-posteriori probability (APP) decoding, e.g., the BCJR algorithm [26], decoder II outputs

$$\lambda_{\mathrm{II},n} = \ln \frac{P(x_n = 0 \mid z_0, ..., z_{N/Q-1},\ L(x_1), ..., L(x_L))}{P(x_n = 1 \mid z_0, ..., z_{N/Q-1},\ L(x_1), ..., L(x_L))} - L(x_n). \tag{2}$$

After deinterleaving, $\lambda_{\mathrm{II},n}$ is considered a-priori LLR $L(c_n) = \ln(P(c_n = 0)/P(c_n = 1))$ on c_n by decoder I, which outputs $\lambda_{\mathrm{I},n}$ and estimates of the transmitted data, e.g.,

$$\lambda_{\mathrm{I},n} = \ln \frac{P(c_n = 0 \mid L(c_1), ..., L(c_L))}{P(c_n = 1 \mid L(c_1), ..., L(c_L))} - L(c_n),$$

using APP-based decoding. The LLRs $\lambda_{\mathrm{I},n}$ and $\lambda_{\mathrm{II},n}$ are often called *extrinsic* information in the literature [25]. Performing iterative decoding, the LLRs $\lambda_{\mathrm{I},n}$ are interleaved and regarded as *a-priori* LLR $L(x_n)$ for decoder II, where $L(x_n) = 0$ for all x_n is assumed for the first iteration.

III. Convergence prediction using EXIT charts

The LLRs communicated between the two decoders can be modelled as outcomes of the random variables (r.v.'s) Λ_{I} and Λ_{II} modeling the LLRs $\lambda_{\mathrm{I},n}$ and $\lambda_{\mathrm{II},n}$, respectively. The outcomes of Λ_{I} and Λ_{II} are distributed with the PDF $f_{\mathrm{I}}(l|c)$ conditioned on $c_n = c$ and $f_{\mathrm{II}}(l|x)$ conditioned on $x_n = x$, respectively, which both vary with the iterations. Analyzing these PDFs allows to predict the

behavior of the decoding algorithm, but this is unfortunately extremely difficult for most system configurations. A substantial simplification is to observe only a single parameter of the PDFs [13–16], e.g., the mutual information between Λ_{I} and the r.v. C [14],

$$I(\Lambda_{\mathrm{I}};C)=\frac{1}{2}\sum_{c\in\{0,1\}}\int_{-\infty}^{\infty}f_{\mathrm{I}}(l|c)\cdot\log_2\frac{2\,f_{\mathrm{I}}(l|c)}{f_{\mathrm{I}}(l|0)+f_{\mathrm{I}}(l|1)}\,\mathrm{d}l. \tag{3}$$

The outcomes of C are the bits c_n, which are assumed to be equally likely 0 or 1. Similarly is defined $I(\Lambda_{\mathrm{II}};X)$ on $f_{\mathrm{II}}(l|x)$, where the outcomes of the r.v. X are the bits x_n. The evolution of $I(\Lambda_{\mathrm{I}};C)$ and $I(\Lambda_{\mathrm{II}};X)$ with the iterations is called *trajectory* of the decoding algorithm and can be depicted in a chart with two axes labeled with $I(\Lambda_{\mathrm{I}};C)$ and $I(\Lambda_{\mathrm{II}};X)$. In the following we focus on $I(\Lambda_{\mathrm{I}};C)$ to show how these parameters can be obtained efficiently. The integral in (3) can be evaluated numerically using a histogram of L LLRs $\lambda_{\mathrm{I},n}$ [14]. Calculating $I(\Lambda_{\mathrm{I}};C)$ becomes much simpler if the PDF $f_{\mathrm{I}}(l|c)$ is both *symmetric*, i.e. $f_{\mathrm{I}}(l|0)=f_{\mathrm{I}}(-l|1)$, and *consistent*, i.e. $f_{\mathrm{I}}(l|0)=f_{\mathrm{I}}(-l|0)\cdot\exp(l)$. For example, an APP-based decoder for a linear code produces symmetric output PDFs if the input PDF (of the *a-priori* LLRs $L(x_n)$ or $L(c_n)$) is symmetric and, when the decoder processes the received z_k, the communication channel is symmetric, e.g., $f_w(w)=f_w(-w)$. Since the output of one decoder is input to the next one, symmetry is retained over the iterations for this example. The consistency constraint is fulfilled when the output LLRs are *valid* log-likelihood ratios. For example, consider all LLRs $\lambda_{\mathrm{I},n}$ at indices n for which $c_n=0$. These LLRs are distributed with $f_{\mathrm{I}}(l|0)$. A hard decision of $\lambda_{\mathrm{I},n}$ on the value of c_n is given by $c_n=0$ if $\lambda_{\mathrm{I},n}\geq 0$ and $c_n=1$ if $\lambda_{\mathrm{I},n}<0$, where latter decision is wrong. From the LLR definition follows that the probability $p=P(\lambda_{\mathrm{I},n}\geq 0)$ of a correct decision is $\exp(|\lambda_{\mathrm{I},n})|/(1+\exp(|\lambda_{\mathrm{I},n}|))$ and that of a wrong decision is $1-p$. Recalling that the $\lambda_{\mathrm{I},n}$ are outcomes of Λ_{I}, we find that a $\lambda_{\mathrm{I},n}$ with a magnitude $|\lambda_{\mathrm{I},n}|$ in the small bin $[l,l+\epsilon]$, $l\geq 0$, of width ϵ is positive with probability $q=f_{\mathrm{I}}(l|0)/(f_{\mathrm{I}}(-l|0)+f_{\mathrm{I}}(l|0))$ or negative with probability $1-q$ for $\epsilon\to 0$. Clearly, the ratio of positive $\lambda_{\mathrm{I},n}$ to negative $\lambda_{\mathrm{I},n}$ should be equal to the ratio of correct and wrong decisions, i.e., $q/(1-q)=p/(1-p)$ in order that the $\lambda_{\mathrm{I},n}$ with such a magnitude are valid LLRs. Since $p=\exp(l)/(1+\exp(l))$ for these $\lambda_{\mathrm{I},n}$, we find that $f_{\mathrm{I}}(l|0)/f_{\mathrm{I}}(-l|0)=\exp(l)$ must hold, the consistency constraint for a particular

l. The corresponding constraint on the distribution $f_{\mathrm{I}}(l|1)$ is $f_{\mathrm{I}}(l|1)/f_{\mathrm{I}}(-l|1) = \exp(-l)$. Since each decoder in an iterative decoding algorithm assumes that the input LLRs are valid, it is an important design task to make sure that, disregarding the decoding principle, output LLRs with a consistent distribution are produced. We note that the this constraint was defined earlier, e.g., in [27] or in [11] (called symmetry constraint), but there it is only used to simplify analysis or to improve the accuracy of simulation results, respectively. In contrast, we consider this constraint as a fundamental property of decoders to be used for iterative decoding. Applying both constraints, the calculation of $I(\Lambda_{\mathrm{I}}; C)$ simplifies greatly,

$$I(\Lambda_{\mathrm{I}}; C) = 1 - \int_{-\infty}^{\infty} f_{\mathrm{I}}(l|0) \cdot \log_2(1 + e^{-l})\, \mathrm{d}l = E_{C=0}(1 - \log_2(1 + e^{-\Lambda_{\mathrm{I}}})). \tag{4}$$

This expectation over $f_{\mathrm{I}}(l|0)$ is arbitrarily closely approximated with the time average

$$I(\Lambda_{\mathrm{I}}; C) \approx 1 - \frac{1}{L} \sum\nolimits_{n=1}^{L} \log_2(1 + \exp(-m(c_n) \cdot \lambda_{\mathrm{I},n})), \tag{5}$$

where $m(\cdot)$ is defined as $m(0) = 1$ and $m(1) = -1$. Here, we again use the symmetry constraint to average over *all* output LLRs, not just those for which $c_n = 0$.

To predict the behavior of the decoding algorithm without actually running the algorithm, both decoders are analyzed separately via their *transfer functions*. For example, decoder I maps LLRs $L(c_n)$ distributed with $f_{\mathrm{II}}(l|c)$ to LLRs $\lambda_{\mathrm{I},n}$ distributed with $f_{\mathrm{I}}(l|c)$. Since $f_{\mathrm{II}}(l|c)$ is not known, i.i.d. LLRs $L(c_n)$ distributed with

$$g(l|c) = \exp\left(-(l - m(c) \cdot 2\gamma)^2/(8\gamma)\right) / \sqrt{8\pi\gamma} \tag{6}$$

conditioned on $c_n = c$ are artificially generated. We refer to [13, 14] to explain this choice of $g(l|c)$. Each $\gamma \in [0, \infty)$ is assigned the mutual information

$$I_{in}(\gamma) = \frac{1}{2} \sum_{c \in \{0,1\}} \int_{-\infty}^{\infty} g(l|c) \cdot \log_2 \frac{2\, g(l|c)}{g(l|0) + g(l|1)}\, \mathrm{d}l = 1 - \int_{-\infty}^{\infty} g(l|0) \cdot \log_2(1 + \exp(-l))\, \mathrm{d}l,$$

with $I_{in}(0) = 0$ (no a-priori information) and $I_{in}(\gamma \to \infty) = 1$ (exact knowledge of the associated bit) as extremal values. For each $I_{in} \in [0, 1]$, decoder I produces LLRs $\lambda_{\mathrm{I},n}$ with $I_{out} = I(\Lambda_{\mathrm{I}}; X)$

yielding the transfer function $I_{out} = T_{\mathrm{I}}(I_{in})$. Similarly is defined the transfer function $I_{out} = T_{\mathrm{II}}(I_{in})$ for decoder II, where in addition the bits x_n are transmitted over the channel yielding received symbols z_k. The analysis based on these transfer functions is accurate when the decoders output the same I_{out} when fed with LLRs distributed with the true PDF $f_{\mathrm{I}}(l|c)$ (and $f_{\mathrm{II}}(l|x)$) or the Gaussian PDF $g(l|c)$ at the same I_{in}. A comparative study observing 6 different parameters revealed that mutual information is one of the most accurate and robust ones [20]. Another simplification is that the input LLRs $L(c_n)$ (and $L(x_n)$) are independent when they are artificially generated using $g(l|x)$. This is not true with $\lambda_{\mathrm{I},n}$ and $\lambda_{\mathrm{II},n}$ in the real system, but it is plausible for large interleaver block lengths, at least for several iterations, in a neighborhood around each $\lambda_{\mathrm{I},n}$ and $\lambda_{\mathrm{II},n}$.

IV. Convergence properties of serially concatenated systems

A. Recursive encoding

Fig. 2 depicts $T_{\mathrm{I}}(I_{in})$ of an APP-based decoder I for a rate-$1/2$ convolutional code with generator $(1{+}D^2\ 1{+}D{+}D^2)$, $T_{\mathrm{II}}(I_{in})$ of decoder II for 8-PSK at 2.7 dB E_b/N_0 using a Gray or an Anti-Gray mapping with a trivial (memory-less) encoder II ($y_n = x_n$), and the trajectory of a real system for $K \in \{250, 2500\}$ using S-random interleaving (S $\in \{15, 25\}$) [28]. Using a Gray mapping yields good performance after one iteration but no gain over the iterations, since $T_{\mathrm{II}}(1) \approx T_{\mathrm{II}}(0)$. Using an Anti-Gray mapping causes a lower $T_{\mathrm{II}}(0)$, but a significant gain over the iterations even surpassing the performance using Gray mapping. This apparent compromise between $T_{\mathrm{II}}(0)$ and $T_{\mathrm{II}}(1)$ was investigated in [3]. We note that the findings in the EXIT chart directly apply to the BER performance. Comparing trajectory and transfer functions reveals that the analysis is accurate only for a small number of iterations, which increases with K. This number specifies that part of the decoding process, where the independence assumptions on $\lambda_{\mathrm{I},n}$ and $\lambda_{\mathrm{II},n}$ actually hold.

For the considered memory-less encoder II we find that $T_{\mathrm{II}}(I_{in} = 1) < 1$ holds regardless of the mapping $S(\cdot)$. At this I_{in}, all $x_1, ..., x_L$ are known to decoder II or $|L(x_n)| \to \infty$ for all n, respectively. Assuming APP-based demapping, it follows from (2) that the output LLRs $\lambda_{\mathrm{II},n}^{(\infty)} =$

$\lambda_{\mathrm{II},n}\big|_{|L(x_n)|\to\infty,\forall n}$ for this case are given by

$$\lambda_{\mathrm{II},n}^{(\infty)} = \ln \frac{P(x_n=0 \,|\, \mathbf{z}, x_1, ..., x_{n-1}, x_{n+1}, ..., x_L)}{P(x_n=1 \,|\, \mathbf{z}, x_1, ..., x_{n-1}, x_{n+1}, ..., x_L)} = \ln \frac{-|z_k - S(\mathbf{y}_k(n,0))|^2/\sigma^2}{-|z_k - S(\mathbf{y}_k(n,1))|^2/\sigma^2},$$

where $\mathbf{y}_k(n,b) = (x_{Qk+1}, ..., x_{n-1}, b, x_{n+1}, ..., x_{Qk+Q})$, $b \in \{0,1\}$, and $k = (n-1) \bmod Q$. Clearly, these $\lambda_{\mathrm{II},n}^{(\infty)}$ cause $T_{\mathrm{II}}(1) < 1$ and, thus, an error floor after decoding convergence, which will not disappear even for $K \to \infty$ [3]. Designing an encoder II such that $T_{\mathrm{II}}(1) = 1$ is (nearly) achieved by decoder II avoids this floor. Previous work reveals that a recursive encoder, where y_n depends not only on x_n and past versions of it, i.e. $x_{n-1}, x_{n-2}, ...$, but also on past versions of y_n [5, 29], provides this property. Such encoders were utilized in [30, 31] in the context of TCM and BiCM, and in [32, 33] in the context of Turbo equalization. It is one task of this paper to analyze and design low-complexity implementations of such encoders.

B. Transfer function integrals

Let A_{I} and $\bar{A}_{\mathrm{I}}$ be the areas under $T_{\mathrm{I}}(i)$ and its inverse $T_{\mathrm{I}}^{-1}(i)$, $i \in [0,1]$, respectively:

$$A_{\mathrm{I}} = \int_0^1 T_{\mathrm{I}}(i)\,\mathrm{d}i, \qquad \bar{A}_{\mathrm{I}} = \int_0^1 T_{\mathrm{I}}^{-1}(i)\,\mathrm{d}i = 1 - A_{\mathrm{I}}.$$

Similar areas A_{II} and $\bar{A}_{\mathrm{II}}$ are defined for $T_{\mathrm{II}}(i)$ and $T_{\mathrm{II}}^{-1}(i)$, respectively. The equivalences $\bar{A}_{\mathrm{I}} = 1 - A_{\mathrm{I}}$ and $\bar{A}_{\mathrm{II}} = 1 - A_{\mathrm{II}}$ follow from the definition of the EXIT chart. It has been observed in [34, 35] that an APP-based decoder for a rate-R_{I} code satisfies the property

$$\bar{A}_{\mathrm{I}} \approx R_{\mathrm{I}}. \tag{7}$$

Equality was proven in [34] when the $L(c_n)$ fed into decoder I are artificially generated using $\tilde{g}(l|c) = \epsilon \cdot \delta(l) + (1-\epsilon) \cdot \delta(l - m(c) \cdot L)$ for $L \to \infty$, where $\delta(l)$ is the Dirac delta function. This PDF follows when the c_n are transmitted over a binary erasure channel with erasure probability ϵ. Calculating $I_{in}(\epsilon) = 1 - \epsilon$ and even $T_{\mathrm{I}}(i)$ for some classes of linear codes becomes now managable [34] in contrast to using the PDF $g(l|c)$. We use (7) even when $g(l|c)$ is used to obtain $T_{\mathrm{I}}(I_{in})$, since the transfer functions are almost identical in both cases. We note that the EXIT chart analysis itself is based on such an assumption, since the true LLR distribution $f_{\mathrm{I}}(l|c)$ is only approximated with

$g(l|c)$. We say that a concatenated system converges whenever

$$T_{\mathrm{II}}(0)>0,\ T_{\mathrm{II}}(1)=1, \quad \text{and} \quad T_{\mathrm{II}}(i)^{-1}<T_{\mathrm{I}}(i),\ \forall i\in[0,1). \tag{8}$$

In [11, 24], decoding convergence is defined by analyzing the fixed point of the recurrence

$$\mu_i=T_{\mathrm{I}}(T_{\mathrm{II}}(\mu_{i-1})),\quad i=1,2,...,\quad \mu_0=0. \tag{9}$$

Convergence is achieved when $\mu_i\to 1$ for $i\to\infty$, which implies that $T_{\mathrm{I}}(T_{\mathrm{II}}(i))>i$ must hold for all $i\in[0,1)$. Applying $T_{\mathrm{I}}^{-1}(\cdot)$ on both sides yields $T_{\mathrm{II}}(i)>T_{\mathrm{I}}^{-1}(i)$, and taking the inverse on both sides yields $T_{\mathrm{II}}^{-1}(i)<T_{\mathrm{I}}(i)$, where we reasonably assume that $T_{\mathrm{I}}(i)$ and $T_{\mathrm{II}}(i)$ are strictly monotonically increasing. Attaining the point $T_{\mathrm{I}}(1)=1$ corresponds to a zero BER, which requires an infinite amount of iterations and an infinite block length K.

An obvious convergence threshold is the bound $\bar{A}_{\mathrm{I}}<A_{\mathrm{II}}$, which is attained when $T_{\mathrm{I}}(i)$ is identical to $T_{\mathrm{II}}^{-1}(i)$. From this bound, (1), and (7) follows that

$$R_{\mathrm{I}}<A_{\mathrm{II}} \qquad \text{as well as} \qquad R<R_{\mathrm{II}}\cdot A_{\mathrm{II}}. \tag{10}$$

We made the observation that A_{II} is approximately Q times the uniform input capacity C_{UI} of the communication channel in bits per transmitted symbol [35], i.e.,

$$Q\cdot A_{\mathrm{II}}\approx C_{\mathrm{UI}} \tag{11}$$

for any bijective rate-1 encoder II regardless of $S(\cdot)$. An explanation for this fact was presented in [36]. A proof was presented in [34] for the case that the LLRs $L(x_n)$ fed into decoder II are generated using $\tilde{g}(l|x)$. As for property (7), we justify the use of (11) since the transfer functions $T_{\mathrm{II}}(i)$ obtained using $g(l|x)$ or $\tilde{g}(l|x)$ to generate the LLRs $L(x_n)$ look almost alike.

An important design rule follows from (11) when C_{UI} is close to its maximum Q, i.e., for large E_s/N_0. To assure that the overall code rate R is close to C_{UI}, it has to approach 1. From (1) or (10) and the obvious bound $A_{\mathrm{II}}\leq 1$ follows that $R\leq R_{\mathrm{II}}$. Consequently, we should set R_{II} to 1. This rule holds not only for large E_s/N_0, since also for medium E_s/N_0, when C_{UI} is significantly smaller than Q, the achievable rate R of a system spending redundancy on encoder II is below the one using a rate-1 encoder [35].

Eq. (8) states the conditions for decoding convergence assuming an infinite block length K and number of performed iterations. Latter assumptions are of course not practical at all and design rules for a finite K and/or number of iterations are desired. In this paper we want to study how the concatenated system can be optimized when only a finite number F of iterations is performed. Precisely, we want to maximize the mutual information μ_F of the output LLRs $\lambda_{\mathrm{I},n}$ of decoder I after F iterations. Fig. 2 shows that this also allows to evaluate the performance for finite K, since the trajectory *does* follow the transfer functions for a small F.

In the following, we show how to design serially concatenated systems according to the rules stated in Sections IV-A and IV-B, i.e., to find an encoder II such that $T_{\mathrm{II}}(1) = 1$ holds and to find an encoder I such that $T_{\mathrm{I}}(I_{in})$ fits to $T_{\mathrm{II}}^{-1}(I_{in})$ according to optimization criteria to be derived.

V. Design of serially concatenated systems

We start with the design of rate-$R_{\mathrm{II}} = 1$ encoders II, i.e. $L = N$, for which $T_{\mathrm{II}}(1) = 1$ holds. We assume that encoder II is a finite-state machine (FSM) with memory M, which processes Q bits $\mathbf{x}_k = (x_{Qk+1}, ..., x_{Qk+Q})$ to output $\mathbf{y}_k$ for $k = 0, 1, ..., N/Q-1$. This assures that encoder II and decoder II work with the same 2^M-state trellis describing the FSM. The evolution of the encoder state $\mathbf{s}_k$, a length M row vector, is in general described with the state-space equations

$$\mathbf{s}_{k+1} = \mathbf{s}_k\mathbf{A} + \mathbf{x}_k\mathbf{B} \quad \text{and} \quad \mathbf{y}_k = \mathbf{s}_k\mathbf{C}^{\mathrm{T}} + \mathbf{x}_k\mathbf{D}.$$

where $\mathbf{A}$ is an $M \times M$ matrix, $\mathbf{B}$ and $\mathbf{C}$ are $Q \times M$ matrices, and $\mathbf{D}$ is a $Q \times Q$ matrix, all with elements from $\mathbb{F}_2$. The Q rows of $\mathbf{B}$ of length M are denoted $\mathbf{b}_j$, $j = 1, ..., Q$. The initial encoder state is $\mathbf{s}_0 = \mathbf{0}$, i.e., the transfer function using the delay operator D is given by

$$\mathbf{y}(D) = \mathbf{x}(D)\mathbf{G}(D) = \mathbf{x}(D) \cdot (\mathbf{B}(D^{-1}\mathbf{I}_M + \mathbf{A})^{-1}\mathbf{C}^{\mathrm{T}} + \mathbf{D}),$$

where $\mathbf{I}_M$ is the $M \times M$ identity matrix and $\mathbf{x}(D) = \mathbf{x}_0 + \mathbf{x}_1 D + \mathbf{x}_2 D^2 + ...$ Clearly, the rate-1 encoder II must be bijective and, thus, $\mathbf{G}(D)$ be invertible. We observe the trellis to analyze the output LLR $\lambda_{\mathrm{II},n}^{(\infty)}$ when $I_{in} = 1$. Decoder II knows $\mathbf{s}_0$, but not the final state $\mathbf{s}_{N/Q}$. Even if the trellis is terminated to $\mathbf{s}_{N/Q+t} = \mathbf{0}$ for some $t > 0$, determining $\mathbf{s}_{N/Q}$ requires the use of noisy symbols

z_k, since decoder II does not know the termination bits $x_{n'}$, $n' > N$, which yield $\mathbf{s}_{N/Q+t} = \mathbf{0}$. In general, a particular state $\mathbf{s}_m$ at time step $m = 0, ..., N/Q-1$ is given by

$$\mathbf{s}_m = \mathbf{s}_0\mathbf{A}^m + \sum_{l=0}^{m-1} \mathbf{x}_l\mathbf{B}\mathbf{A}^{m-l-1} = \sum_{l=0}^{m-1} \mathbf{x}_l\mathbf{B}\mathbf{A}^{m-l-1}.$$

For any $m > k$, this sum can be split into

$$\mathbf{s}_m = x_{Qk+j}\mathbf{b}_j\mathbf{A}^{m-k-1} + \mathbf{r}_m(k,j), \qquad \mathbf{r}_m(k,j) = \left(\sum_{l=1:l\neq j}^{Q} x_{Qk+l}\mathbf{b}_l\mathbf{A}^{m-k-1}\right) + \left(\sum_{l=0:l\neq k}^{m-1} \mathbf{x}_l\mathbf{B}\mathbf{A}^{m-l-1}\right),$$

where $j = 1, 2, ..., Q$. To compute $\lambda_{\mathrm{II},n}^{(\infty)}$ for a particular $n = Qk+j$, all $x_{n'}$, $n' \neq n$, are known and, thus, the difference $\mathbf{r}_m(k,j) = \mathbf{s}_m - x_{Qk+j}\mathbf{b}_j\mathbf{A}^{m-k-1}$ is uniquely determined. The state $\mathbf{s}_m$ is either known exactly for $m < k$ or it takes one of the at most two values $\mathbf{r}_m(k,j) + x \cdot \mathbf{b}_j\mathbf{A}^{m-k-1}$, $x \in \mathbb{F}_2$, for $m > k$. It follows that

$$\lambda_{\mathrm{II},n}^{(\infty)} = \ln \frac{\prod_{l=k+1}^{N/Q-1} p(z_l \mid S(\mathbf{y}_l(k,j,0)))}{\prod_{l=k+1}^{N/Q-1} p(z_l \mid S(\mathbf{y}_l(k,j,1)))} = \frac{1}{\sigma^2} \cdot \sum_{l=k+1}^{N/Q-1} |z_l - S(\mathbf{y}_l(k,j,1))|^2 - |z_l - S(\mathbf{y}_l(k,j,0))|^2,$$

where $\mathbf{y}_l(k,j,x) = \mathbf{r}_l(k,j)\,\mathbf{C}^{\mathrm{T}} + x \cdot \mathbf{b}_j\mathbf{A}^{l-k-1}\mathbf{C}^{\mathrm{T}} + \mathbf{x}_l\mathbf{D}$. The expectation of $|\lambda_{\mathrm{II},n}^{(\infty)}|$ over $f_w(w)$ is $E(|\lambda_{\mathrm{II},n}^{(\infty)}|) = \frac{1}{\sigma^2}\sum_{l=k+1}^{N/Q-1} |S(\mathbf{y}_l(k,j,1)) - S(\mathbf{y}_l(k,j,0))|^2$. For a finite σ^2, $E(|\lambda_{\mathrm{II},n}^{(\infty)}|)$ grows unbounded with N/Q if the number of non-zero $1 \times Q$ vectors $\mathbf{b}_j\mathbf{A}^{l-k-1}\mathbf{C}^{\mathrm{T}}$, $l = k+1, k+2, ...$, is unbounded, since only non-zero $\mathbf{b}_j\mathbf{A}^{l-k-1}\mathbf{C}^{\mathrm{T}}$ yield a non-zero difference $S(\mathbf{y}_l(k,j,1)) - S(\mathbf{y}_l(k,j,0))$. However, this number is bounded if $\mathbf{b}_j\mathbf{A}^{l-k-1}$ is zero for any $l = k+1, ...$ Since there are at most 2^M different $\mathbf{b}_j\mathbf{A}^{l-k-1}$, it suffices to check whether $\mathbf{b}_j\mathbf{A}^{2^M-1}$ is zero. When $\mathbf{b}_j\mathbf{A}^{2^M-1}$ is non-zero, the number of non-zero $\mathbf{b}_j\mathbf{A}^{l-k-1}\mathbf{C}^{\mathrm{T}}$ is unbounded when there is at least one non-zero $\mathbf{b}_j\mathbf{A}^{l'}\mathbf{C}^{\mathrm{T}}$, $l' = 0, 1, ..., 2^M-1$, for all $j = 1, ..., Q$. The necessary und sufficient conditions on encoder II such that decoder II approaches $T_{\mathrm{II}}(I_{in} = 1) = 1$ for large N/Q are therefore

$$(\mathbf{1})\ \mathbf{G}(D) \text{ is invertible} \quad (\mathbf{2})\ \mathbf{b}_j\mathbf{A}^{2^M-1} \neq \mathbf{0},\ \forall j, \quad \text{and} \quad (\mathbf{3})\ \exists\, \mathbf{b}_j\mathbf{A}^{l}\mathbf{C}^{\mathrm{T}} \neq \mathbf{0},\ l = 0, ..., 2^M-1,\ \forall j.$$

Even though $E(|\lambda_{\mathrm{II},n}^{(\infty)}|) \to \infty$ holds only for $N \to \infty$, i.e., a decoder II satisfying $T_{\mathrm{II}}(1) = 1$ exists only for $N \to \infty$, we say that $T_{\mathrm{II}}(1) = 1$ is achieved already when the three constraints above are satisfied. A non-recursive encoder, where $\mathbf{A}$ is lower triangular, cannot satisfy the second

constraint, since $\mathbf{A}^M$ is the all-zero matrix. On the other hand, simple 2-state ($M=1$) encoders yielding $T_{\rm II}(1)=1$ can be found for any Q by choosing $\mathbf{A}=1$, $\mathbf{B}=[1\,1...1]$, and $\mathbf{C}\neq\mathbf{0}$. Fig. 3 shows $T_{\rm II}(I_{in})$ of decoder II at 2.7 dB E_b/N_0 for seven rate-1, 2-state encoders for 8-PSK using Gray mapping, $\mathbf{D}=\mathbf{I}_3$, and $\mathbf{C}\in\{[100],[010],...,[111]\}$. Also included is $T_{\rm II}(I_{in})$ for a memoryless mapper $y_n=x_n$ and $T_{\rm I}(I_{in})$ for the rate-$1/2$ code from Section IV-A. As predicted, $T_{\rm II}(1)=1$ holds and, furthermore, $3\cdot A_{\rm II}$ is equal to $C_{\rm UI}=1.76$ in bits per channel use for all encoders II.

To approach $C_{\rm UI}$ in the example above, we need an outer code whose $T_{\rm I}(I_{in})$ fits to $T_{\rm II}^{-1}(I_{in})$. Fig. 3 shows that convergence is possible using the encoder II with $\mathbf{C}=[100]$, but not with $\mathbf{C}=[111]$. In fact, since $A_{\rm II}=1.76/3=0.587$, convergence is possible with a rate-0.587 code if $T_{\rm I}(I_{in})$ is suitably chosen. However, there is little knowledge how $T_{\rm I}(I_{in})$ looks like for a given code $\mathcal{C}$ and optimizing $T_{\rm I}(I_{in})$ usually requires to scan through many possible codes. This search procedure can be reduced and improved by selecting a family of subcodes $\mathcal{C}_k$, $k=1,2,...$, which are used to construct a target code $\mathcal{C}$, whose code words consist of fractions of code words from the subcodes. Such a code is usually referred to as *irregular* code, e.g., irregular LDPC codes [11] or RA codes [24]. There, simple parity check and/or repetition constraints on the code bits of $\mathcal{C}$ are used as subcodes. The approaches in [22, 23] optimize $T_{\rm II}(I_{in})$ by encoding fractions of the code bits x_n with different encoders II. We propose to keep $T_{\rm II}(I_{in})$ constant and, instead, optimize $T_{\rm I}(I_{in})$ by encoding fractions of the K information bits using a general set of P subcodes $\mathcal{C}_k$.

To keep en- and decoding of $\mathcal{C}$ simple, we propose to select a rate-r_1 convolutional mother code $\mathcal{C}_1$ and obtain the $P-1$ other subcodes of rate $r_k>r_1$ by puncturing. Thus, en- and decoding of all $\mathcal{C}_k$ can be performed on the mother code trellis. Using $\mathcal{C}_k$ to encode a fraction of $\alpha_k r_k L$ information bits to $\alpha_k L$, $\alpha_k\in[0,1]$, code bits c_n, the α_k have to satisfy

$$1=\sum_{k=1}^{P}\alpha_k \quad \text{and} \quad R_{\rm I}=\sum_{k=1}^{P}\alpha_k r_k \tag{12}$$

given the target rate $R_{\rm I}\in[0,1]$. For example, a family of $P=17$ subcodes constructed from a systematic, rate-$1/2$, memory 4 mother code defined by the generator matrix $\mathbf{G}(D)=1/g_0\cdot[g_0\;\;g_1]$, $g_0=1+D+D^4$, $g_1=1+D^2+D^3+D^4$, is used in this paper. Higher rates are obtained by puncturing,

lower rates are obtained by adding more generators and using puncturing while maximizing the free distance. Latter optimization requires only two new generators, $g_2 = 1+D+D^2+D^4$ and $g_3 = 1+D+D^3+D^4$. We denote the 17 subcodes with the tuples $\{r_k, [w_0, w_1, ...], l_k, [p_0, p_1, ...]\}$, $k = 1, ..., 17$, where w_i, $i = 0, 1, 2, 3$, denotes how often g_i occurs in the generator matrix, l_k is the puncturing period, and p_i is the octal representation of the puncturing pattern associated to g_i:
$\{0.10, [1, 4, 4, 1], 1, [1, 1, 1, 1]\}$, $\{0.15, [1, 3, 2, 1], 3, [7, 7, 7, 3]\}$, $\{0.20, [1, 2, 1, 1], 1, [1, 1, 1, 1]\}$,
$\{0.25, [1, 1, 1, 1], 1, [1, 1, 1, 1]\}$, $\{0.30, [1, 1, 1, 1], 3, [7, 7, 7, 1]\}$, $\{0.35, [1, 1, 1], 7, [177, 177, 077]\}$,
$\{0.40, [1, 1, 1], 2, [3, 3, 1]\}$, $\{0.45, [1, 1, 1], 9, [777, 777, 021]\}$, $\{0.50, [1, 1], 1, [1, 1]\}$,
$\{0.55, [1, 1], 11, [3777, 2737]\}$, $\{0.60, [1, 1], 3, [7, 3]\}$, $\{0.65, [1, 1], 13, [17777, 05253]\}$,
$\{0.70, [1, 1], 7, [177, 025]\}$, $\{0.75, [1, 1], 3, [7, 1]\}$, $\{0.80, [1, 1], 4, [17, 1]\}$,
$\{0.85, [1, 1], 17, [377777, 010101]\}$, and $\{0.90, [1, 1], 9, [777, 1]\}$.

For example, the last tuple $\{0.90, [1, 1], 9, [777, 1]\}$ denotes that only g_0 and g_1 are used where the bits belonging to g_0 are not punctured at all and those of g_1 are punctured in 8 out of 9 trellis sections ($p_1 \triangleq 000.000.001$) yielding the rate $r_{17} = 1/2 \cdot 18/10 = 0.9$. We note that at the beginning of each block of $\alpha_k r_k L$ trellis sections corresponding to $\mathcal{C}_k$, the puncturing pattern should be applied from the beginning. Trellis termination is necessary only after all K information bits have been encoded. The output LLRs $\lambda_{\mathrm{I},n}$ of decoder I are distributed with $f_{\mathrm{I},k}(l|c)$ in the fraction of the trellis belonging to $\mathcal{C}_k$. To each $f_{\mathrm{I},k}(l|c)$ corresponds a transfer function $T_{\mathrm{I},k}(I_{in})$. Assuming that the trellis fractions do not significantly interfere with each other, which might change the transfer characteristic, the transfer function $T_{\mathrm{I}}(I_{in})$ of the target code $\mathcal{C}$ is the weighted superposition of the $T_{\mathrm{I},k}(I_{in})$ when all $f_{\mathrm{I},k}(l|c)$ are symmetric and consistent, i.e. $T_{\mathrm{I}}(I_{in}) = \sum_{k=1}^{P} \alpha_k T_{\mathrm{I},k}(I_{in})$, which follows from (4). The task of fitting $T_{\mathrm{I}}(I_{in})$ to a given $T_{\mathrm{II}}^{-1}(I_{in})$ while satisfying (8) can be formulated as a linear or quadratic programming problem [37]:

$$\left| \begin{array}{l} \textbf{Minimize } J(\boldsymbol{\alpha}) = \frac{1}{2} \cdot \boldsymbol{\alpha}\mathbf{Q}\boldsymbol{\alpha}^{\mathrm{T}} + \boldsymbol{\alpha}\mathbf{P} \\ \textbf{subject to } \sum_{k=1}^{P} \alpha_k T_{\mathrm{I},k}(i) > T_{\mathrm{II}}^{-1}(i), \ \forall\, i \in [0, 1), \\ \boldsymbol{\alpha}\mathbf{C} = [\,1 \ R_{\mathrm{I}}\,], \quad \alpha_k \in [0, 1], \ \ k = 1, 2, ..., P, \end{array} \right| \qquad \mathbf{C} = \begin{bmatrix} 1 & 1 & ... & 1 \\ r_1 & r_2 & ... & r_P \end{bmatrix}^{\mathrm{T}}, \tag{13}$$

where $\boldsymbol{\alpha} = [\alpha_1 \ ... \ \alpha_P]$, $\mathbf{Q}$ is a positive semi-definite $P \times P$ matrix, and $\mathbf{P}$ is a $P \times 1$ matrix. Using

the constraints on $\boldsymbol{\alpha}$ alone yields irregular codes $\mathcal{C}$ for which decoding convergence is possible assuming an infinite K and F. When many valid $\boldsymbol{\alpha}$ are possible, appropriate cost functions $J(\boldsymbol{\alpha})$ can be used to find codes $\mathcal{C}$ satisfying or optimizing additional constraints, for instance:

C1 Penalize subcodes with high rates.

C2 Minimize the total square error $\int_0^1 (T_{\mathrm{I}}(i) - T_{\mathrm{II}}^{-1}(i))^2 \,\mathrm{d}i$.

C3 Minimize the total absolute error $\int_0^1 (T_{\mathrm{I}}(i) - T_{\mathrm{II}}^{-1}(i)) \,\mathrm{d}i$.

C4 Maximize the mutual information μ_F after F iterations.

The criteria C1-C3 are chosen ad-hoc, but they can be expressed with appropriate $\mathbf{Q}$ and $\mathbf{P}$, e.g., $\mathbf{Q} = \mathbf{0}$, $\mathbf{P} = [1^p\, 2^p\, 3^p ... P^p]$, $p = 2, 3, ...$, for C1. The criterion C4 cannot be solved with (13). Appendix A devises an iterative solution to this optimization problem.

Suppose we seek a code $\mathcal{C}$ of highest rate R_{I}, such that decoding convergence is possible at 2.7 dB E_b/N_0 using two rate-1, 2-state encoders with $\mathbf{A} = 1$, $\mathbf{B} = [111]$, $\mathbf{C} \in \{[100], [110]\}$, and $\mathbf{D} = \mathbf{I}_3$ for 8-PSK using Gray mapping. The largest rate for convergence is $A_{\mathrm{II}} = 0.587 = C_{\mathrm{UI}}/3$. For this application, a solution $\boldsymbol{\alpha}$ satisfying the constraints in (13) is sought, only. This is usually a single vector, i.e., imposing any cost function is not necessary. The two uppermost plots in Fig. 4 depict the result including $T_{\mathrm{II}}(I_{in})$ for the two encoders, $T_{\mathrm{I},k}(I_{in})$ of the subcodes, and $T_{\mathrm{I}}(I_{in})$ of the irregular code $\mathcal{C}$. The highest rate for convergence is $R_{\mathrm{I}} = 0.559$ using the encoder II with $\mathbf{C} = [100]$, but it is $R_{\mathrm{I}} = 0.583$ using the one with $\mathbf{C} = [110]$.

Another application is to fix the rate of $\mathcal{C}$ for example to $R_{\mathrm{I}} = 1/2$, and to find a $T_{\mathrm{I}}(I_{in})$ aiming on *fast* decoding convergence, which we define as a quick increase in μ_i over the first iterations. This could be desired, e.g., for complexity reasons or given a finite K (see Fig. 2). The last 8 plots of Fig. 4 show the resulting $T_{\mathrm{I}}(I_{in})$ of irregular codes $\mathcal{C}$ optimized using the criteria C1 ($p = 2$), C2, C3, and C4 ($F = 6$) and the expected system trajectory after 6 iterations. Non of the ad-hoc criteria C1-C3 give satisfactory results for both encoders II. Only the criterion C4, which is optimal with respect to the stated problem formulation, always provides a useful code $\mathcal{C}$. However, latter optimization is somewhat complicated and yields a different $\mathcal{C}$ depending on the type of encoder II, E_b/N_0, R_{I}, and F. Heuristically, we found that for the chosen family of subcodes, encoders II

yielding a large $T_{\rm II}(0)$ achieve the largest μ_F for small $F = 1, ..., 10$ when μ_F closely approaches 1. This situation is well-explained in Fig. 4, where the encoder with $\mathbf{C} = [100]$ outperforms that one with $\mathbf{C} = [110]$ for 2.7 dB E_b/N_0 and $R_{\rm I} = 0.5$. In fact, the encoder with $\mathbf{C} = [100]$ yields the largest $T_{\rm II}(0)$ among all 2-state encoders for 8-PSK. In contrast, if we allow a large number F of iterations to be performed and a large K, the encoder with $\mathbf{C} = [110]$ outperforms all others in achieving the highest possible rate $R_{\rm I}$ for convergence at that E_b/N_0.

VI. Results

We derived design rules for two criteria: to approach the channel capacity as close as possible or to achieve a maximal gain in μ_F (or a low BER) at the output of decoder I after F iterations. Addressing the first criterion, we searched among all rate-1, 2-state encoders II for QPSK, 8PSK, 16QAM, 32QAM, and 64QAM using a Gray mapping yielding the smallest E_b/N_0 for which convergence is possible with the irregular codes from Sec. V. This E_b/N_0, denoted $E_b/N_0|_{th}$, is referred to as *threshold* [11, 24, 30, 31]. The signal constellations and mappings are defined in Figs. 2 and 5. Table I shows optimized codes $\mathcal{C}$ and encoders II for several throughputs $T = R{\cdot}Q = R_{\rm I}{\cdot}Q$ in bits per channel use. Also shown is the SNR $E_b/N_0|_{min}$ for which $C_{\rm UI} = T$ when 64QAM modulation is used. Fig. 6 shows that with a single family of subcodes we can closely approach $C_{\rm UI}$ over a wide range of throughputs just by selecting the weights $\boldsymbol{\alpha}$ and the signal constellation appropriately. We stated in Sec. V that decoders II yielding a large $T_{\rm II}(0)$ improve the convergence speed, i.e., they help to maximize μ_F for small F. We found that among all rate-1, 2-state encoders with $\mathbf{D} = \mathbf{I}_Q$, those with $\mathbf{A} = 1$, $\mathbf{B} = [11...1]$, and $\mathbf{C} = [10...0]$ maximize $T_{\rm II}(0)$. Encoders with more memory, i.e. $M > 1$, or a more general $\mathbf{D}$ do not improve $T_{\rm II}(0)$ significantly.

We want to illustrate the results of this paper with an example. We seek to transmit $T = 1.5$ bit per channel use over an AWGN channel using 8PSK. Table I shows that at least 1.0 dB E_b/N_0 is required to transmit these 1.5 bit reliably. An irregular rate-1/2 code constructed with the weights $\boldsymbol{\alpha}$ = (0, 0, 0, 0.05, 0.24, 0, 0.15, 0, 0.26, 0, 0, 0.11, 0.05, 0.07, 0, 0, 0.07) and an encoder II with $\mathbf{A} = 1$, $\mathbf{B} = [111]$, $\mathbf{C} = [110]$, and $\mathbf{D} = \mathbf{I}_3$ optimizes decoding convergence. The convergence

threshold is 1.3 dB E_b/N_0. We also find that the rate-$1/2$ code constructed from $\boldsymbol{\alpha}$ = (0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0) and an encoder II with $\mathbf{A} = 1$, $\mathbf{B} = [111]$, $\mathbf{C} = [100]$, and $\mathbf{D} = \mathbf{I}_3$ optimizes μ_6, the mutual information $I(\Lambda_{\rm I}; C)$ at the output of decoder I after $F = 6$ iterations.

Fig. 7 shows the BER performance of a concatenated system using these two parameter sets for $K \in \{250, 2500, 25000\}$ using S-random interleavers with $S \in \{15, 25, 40\}$. The receiver performs $F \in \{10, 20, 40\}$ iterations. Reference is a BiCM system with a memory-less encoder II using the Gray or the Anti-Gray mapping in Fig. 2. The code $\mathcal{C}$ optimizing decoding convergence shows indeed very early convergence starting at 1.4 dB E_b/N_0 for $K = 25000$. However, the performance improvement over the iterations declines for shorter K and is actually worst among all systems for $K = 250$. This is caused by the fractions of poorly protected bits in a code word $\mathbf{c}$ from $\mathcal{C}$ due to high-rate subcodes $\mathcal{C}_k$. We introduced the criterion C1, which attempts to avoid fractions of high-rate codes in $\mathcal{C}$, for this reason. The second system using the code optimizing μ_6 shows an excellent performance for all K except that it converges at a slightly larger E_b/N_0. Since $T_{\rm II}(1) = 1$ holds for both example systems, they do not suffer from an error floor as the BiCM systems do.

VII. Conclusions

An iteratively decoded concatenated system achieving decoding convergence close to the channel capacity must use irregular codes to match the transfer functions of the two decoders in the receiver. These irregular codes contain fractions of high-rate codes to fit $T_{\rm I}(I_{in})$ well to $T_{\rm II}^{-1}(I_{in})$ yielding poorly protected information bits in the code words more prone to bit errors. The result are significant error floors for small K. We conclude that there is a fundamental trade-off in early convergence and poor finite-length performance of an iteratively decoded system. Analyzing the above-mentioned floors is cumbersome for two reasons. First, bounds on the BER performance, e.g., using the distance spectrum of the global code consisting of both encoders and assuming Maximum Likelihood (ML) decoding, are hard to obtain for the given system structure (recursive encoder II). Second, iterative decoding performs significantly worse than ML decoding for short K, i.e., bounds on the ML decoding performance are not meaningful. A treatment of this issue

is omitted here, also because of space limitation. The EXIT chart analysis underlying the design rules assumes a very large K and optimized codes performing well for large K unfortunately fail for short K. We solved this dilemma by introducing an optimization criterion for which the made assumptions approximately hold also for short K - the first F iterations of iterative decoding.

We note that the derived design rules also apply for parallel concatenated systems and for other communication channels with possibly unknown and/or time-varying parameters. For example, in presence of ISI in the channel, decoder II performs APP-based or linear equalization [18]. Using encoders II with memory causes usually no complexity overhead using APP equalization [32, 33], but the extension of linear equalizers is more troublesome. We emphasize that property (11) holds, too. Thus, applying the design rules from this paper, the uniform input capacity C_{UI} of channels such as ISI or Fading channels can be approached [35, 36]. A drawback is the fact that the transmitter must know $T_{\text{II}}(I_{in})$ precisely, which depends on channel parameters such as the SNR, the fading amplitude, or the impulse response. When these parameters are quickly time-varying, using a large K averages over the possible channel states and decoder II exhibits an *ergodic* transfer function $T_{\text{II}}(I_{in})$ depending only on a few parameters such as the SNR, which could be communicated to the transmitter via a feedback path. However, the faster the channel is varying, the more difficult channel estimation becomes in the receiver. For slowly varying parameters, a feedback path is even more important. Without feedback, the transmitter might design a robust outer code $\mathcal{C}$, e.g., using criterion C4, which tolerates uncertainties in $T_{\text{II}}(I_{in})$. We conclude that the closer a concatenated system approaches a particular channel capacity, the more precise the transmitter needs to know the channel parameters.

APPENDIX A

We wish to find the weight vector $\boldsymbol{\alpha}_{\text{opt}} = \text{argmax}_{\forall \boldsymbol{\alpha} \in \mathcal{A}} \mu_F(\boldsymbol{\alpha})$, where $\mathcal{A}$ is the set of all $\boldsymbol{\alpha}$ satisfying $\boldsymbol{\alpha}\mathbf{C} = [1 \; R_{\text{I}}]$ and $\alpha_k \in [0,1]$ for all k. The first constraint in (13) is not considered, since there might be solutions $\boldsymbol{\alpha}$ maximizing μ_F with $T_{\text{I}}(i) < T_{\text{II}}^{-1}(i)$ for some $i > \mu_F$. From (9) follows that

$$\nabla \mu_F = \frac{\partial \mu_F(\boldsymbol{\alpha})}{\partial \boldsymbol{\alpha}} = \left[\frac{\partial \mu_F}{\partial \alpha_1} \;\; \frac{\partial \mu_F}{\partial \alpha_2} \;\; \ldots \;\; \frac{\partial \mu_F}{\partial \alpha_P} \right]^{\text{T}},$$

the gradient of μ_F with respect to $\boldsymbol{\alpha}$, is also given by a recurrence,

$$\frac{\partial \mu_i}{\partial \alpha_k} = T_k(\mu_{i-1}) + \frac{\partial \mu_{i-1}}{\partial \alpha_k} \cdot \sum_{j=1}^{P} \alpha_j T_j'(\mu_{i-1}), \quad i = 2, 3, ..., F,$$

where $T_k(i) = T_{\mathrm{I},k}(T_{\mathrm{II}}(i))$, $T_k'(i) = \mathrm{d}T_k(i)/\mathrm{d}i$, $i \in (0,1)$, and $\partial\mu_1/\partial\alpha_k = T_k(0)$ for all k. We can find a local maximum of $\mu_F(\boldsymbol{\alpha})$ for all $\boldsymbol{\alpha}$ in the domain $\alpha_k \in \mathbb{R}$ using the steepest descent approach

$$\boldsymbol{\alpha}_{j+1} = \boldsymbol{\alpha}_j + s_j \nabla \mu_F(\boldsymbol{\alpha}_j), \quad j = 0, 1, ..., \quad \boldsymbol{\alpha}_j = [\alpha_{j,1} ... \alpha_{j,P}], \tag{14}$$

where $\boldsymbol{\alpha}_0 \in \mathcal{A}$ is an initial guess. We obtained $\boldsymbol{\alpha}_0$ using (13) and the criterion C2. However, applying (14) may yield solutions $\boldsymbol{\alpha}_j$ outside $\mathcal{A}$. Projecting each intermediate $\boldsymbol{\alpha}_j$ onto the convex set $\mathcal{A}$ yields a local maximum of $\mu_F(\boldsymbol{\alpha})$ for $\boldsymbol{\alpha}$ being in $\mathcal{A}$. This projection is carried out by projecting iteratively onto the set of all $\boldsymbol{\alpha}$ satisfying $\boldsymbol{\alpha}\mathbf{C} = [1\ R_{\mathrm{I}}]$ and onto the set of all $\boldsymbol{\alpha}$ satisfying $\alpha_k \in [0,1]$ for all k. We are not able to show whether the obtained local maximum is equal to the global maximum $\boldsymbol{\alpha}_{\text{opt}}$. In fact, $\mu_F(\boldsymbol{\alpha})$ exhibits multiple local maxima. We assume that with the chosen $\boldsymbol{\alpha}_0 \in \mathcal{A}$, the local maximum found with (14) is a satisfactory solution to the stated optimization problem. An appropriate step size s_j is $(\mu_F(\boldsymbol{\alpha}_{\text{opt}}) - \mu_F(\boldsymbol{\alpha}_j))/\|\nabla\mu_F(\boldsymbol{\alpha}_j))\|^2$ [38], which requires the knowledge of $\boldsymbol{\alpha}_{\text{opt}}$. We used the upper bound $\mu_F(\boldsymbol{\alpha}_{\text{opt}}) \leq 1$ on $\mu_F(\boldsymbol{\alpha}_{\text{opt}})$, i.e., $s_j = (1 - \mu_F(\boldsymbol{\alpha}_j))/\|\nabla\mu_F(\boldsymbol{\alpha}_j))\|^2$, for which we always achieved convergence. The devised algorithm is summarized in the following:

1. To initialize, set $\mu_0 = 0$, $l = 0$, $g_{1,k} = T_k(0)$, $k = 1, 2, ..., P$, and choose $\boldsymbol{\alpha}_0 = [\alpha_{0,1} ... \alpha_{0,P}]$.
2. Compute gradient and update $\boldsymbol{\alpha}_l$:

$$\begin{aligned}
\mu_i &= \sum\nolimits_{j=1}^{P} \alpha_{l,j} T_k(\mu_{i-1}), \quad i = 1, 2, ..., F, \\
g_{i,k} &= T_k(\mu_{i-1}) + g_{i-1,k} \cdot \sum\nolimits_{j=1}^{P} \alpha_{l,j} T_j'(\mu_{i-1}), \quad i = 2, 3, ..., F, \quad k = 1, 2, ..., P, \\
\boldsymbol{\alpha}_l &= \boldsymbol{\alpha}_{l-1} + (1 - \mu_F) / \left(\sum\nolimits_{k=1}^{P} g_{F,k}^2 \right) \cdot [g_{F,1}\, g_{F,2} \cdots g_{F,P}],
\end{aligned}$$

3. Project $\boldsymbol{\alpha}_l$ onto $\mathcal{A}$:

$$\begin{aligned}
\boldsymbol{\alpha}' &= \boldsymbol{\alpha}_l, \\
\alpha_{l,k} &= 0 \text{ when } \alpha_{l,k} < 0 \text{ and } \alpha_{l,k} = 1 \text{ when } \alpha_{l,k} > 1, \quad k = 1, 2, ..., P, \\
\boldsymbol{\alpha}_l &= \boldsymbol{\alpha}_l - \mathbf{C}(\mathbf{C}^{\mathrm{T}}\mathbf{C})^{-1}\mathbf{C}^{\mathrm{T}}(\boldsymbol{\alpha}_l - \boldsymbol{\alpha}_0),
\end{aligned}$$

4. Go to step (3) while $||\boldsymbol{\alpha}' - \boldsymbol{\alpha}_l||_2 > \epsilon$.

5. Set l to $l+1$. Go to step (2) while $l < l_{max}$.

To find the irregular codes in this paper, we chose $l_{max} = 1000$ and $\epsilon = 10^{-8}$.

REFERENCES

[1] C. Berrou, A. Glavieux, and P. Thitimajshima, "Near shannon limit error-correcting coding and decoding: Turbo codes," in *Proc. IEEE Intern. Conf. on Comm., Geneva*, May 1993.

[2] C. Douillard et al., "Iterative correction of intersymbol interference: Turbo equalization," *European Trans. on Telecomm.*, vol. 6, pp. 507–511, Sep-Oct 1995.

[3] S. ten Brink, J. Speidel, and R. Yan, "Iterative demapping and decoding for multilevel modulation," *Proc. IEEE Globecom Conf.*, pp. 579–584, Nov 1998.

[4] A. Chindapol and J. Ritcey, "Design, analysis, and performance evaluation for BICM-ID with square QAM constellations in rayleigh fading channels," *IEEE Journal on Sel. Areas on Comm.*, vol. 19, pp. 944–957, May 2001.

[5] S. Benedetto et al., "Serial concatenated trellis coded modulation with iterative decoding: design and performance," in *Proc. IEEE Global Telecomm. Conf.*, Nov 1997.

[6] X. Li and J. Ritcey, "Trellis-coded modulation with bit interleaving and iterative decoding," *IEEE Journal on Sel. Areas in Communications*, vol. 17, pp. 715–724, April 1999.

[7] N. Wiberg, *Codes and decoding on general graphs*. PhD thesis, Linköping University, 1996.

[8] D. MacKay and R. Neal, "Near Shannon limit performance of low-density parity-check codes," *IEEE Electronics Letters*, vol. 32, pp. 1645–1646, Aug 1996.

[9] R. Gallager, "Low density parity check codes," *IRE Trans. on Information Theory*, vol. 8, pp. 21–28, January 1962.

[10] T. Richardson and R. Urbanke, "The capacity of low density parity-check codes under message passing decoding," *IEEE Trans. on Information Theory*, vol. 47, pp. 599–618, Feb 2001.

[11] T. Richardson and R. Urbanke, "Design of capacity-approaching low density parity-check codes," *IEEE Trans. on Information Theory*, vol. 47, pp. 619–637, Feb 2001.

[12] S. Chung, T. Richardson and R. Urbanke, "Analysis of sum-product decoding of low-density-parity-check codes using a Gaussian approximation," *IEEE Trans. on Information Theory*, vol. 47, pp. 657–670, Feb 2001.

[13] H. El Gamal and A. Hammons, Jr., "Analyzing the Turbo decoder using the Gaussian approximation," *IEEE Trans. on Information Theory*, vol. 47, pp. 671–686, Feb 2001.

[14] S. ten Brink, "Convergence behaviour of iteratively decoded parallel concatenated codes," *IEEE Trans. on Comm.*, vol. 49, pp. 1727–1737, Oct 2001.

[15] D. Divsalar, S. Dolinar, and F. Pollara, "Serial Turbo trellis coded modulation with rate-1 inner code," in *Proc. Intern. Symp. on Information Theory*, p. 194, June 2000.

[16] P. Alexander, A. Grant, and M. Reed, "Iterative detection and code-division multiple-access with error control coding," *European Trans. on Telecomm.*, vol. 9, pp. 419–425, Sep-Oct 1998.

[17] S. ten Brink, "Code characteristic matching for iterative decoding of serially concatenated codes," *Annals of Telecommunications*, vol. 56, pp. 394–408, April 2001.

[18] M. Tüchler, R. Koetter, and A. Singer, "Turbo equalization: principles and new results," *IEEE Trans. on Comm.*, pp. 754–767, May 2002.

[19] B. Scanavino, G. Mondorsi, and S. Benedetto, "Convergence properties of iterative decoders working at bit and symbol level," in *Proc. IEEE Intern. Conf. on Comm.*, Nov 2001.

[20] M. Tüchler, S. ten Brink, and J. Hagenauer, "Measures for tracing convergence of iterative decoding algorithms," in *Proc. 4th IEEE/ITG Conf. on Source and Channel Coding, Berlin, Germany*, pp. 53–60, Jan 2002.

[21] B. Frey and D. MacKay, "Irregular turbo-like codes," in *Proc. 2nd Int. Symp. on Turbo codes, Brest*, pp. 67–72, Sep 2000.

[22] S. ten Brink, "Code doping for triggering iterative decoding convergence," in *Proc. Intern. Symp. on IT*, p. 235, Oct 2001.

[23] D. Divsalar, S. Dolinar, and F. Pollara, "Iterative turbo decoder analysis based on density evolution," *IEEE Journal on Sel. Areas on Comm.*, vol. 19, pp. 891–907, May 2001.

[24] H. Jin, A. Khandekar, and R. McEliece, "Irregular repeat-accumulate codes," in *Proc. 2nd Intern. Symp. on Turbo Codes and Related Topics*, pp. 1–8, Sep 2000.

[25] J. Hagenauer, E. Offer, and L. Papke, "Iterative decoding of binary block and convolutional codes," *IEEE Trans. on Information Theory*, pp. 429–445, March 1996.

[26] L.R. Bahl et al., "Optimal decoding of linear codes for minimizing symbol error rate," *IEEE Trans. on Information Theory*, vol. 20, pp. 284–287, March 1974.

[27] P. Hoeher, U. Sorger, and I. Land, "Log-likelihood values and monte carlo simulation - some fundamental results," in *Proc. Int. Symp. on Turbo Codes, Brest, France*, pp. 43–46, Sep 2000.

[28] C. Heegard and S. Wicker, *Turbo Coding*. Boston: Kluwer Academic Publishing, 1999.

[29] S. Benedetto et al., "Serial concatenation of interleaved codes: performance analysis design, and iterative decoding," *IEEE Trans. on Information Theory*, vol. 44, pp. 909–926, May 1998.

[30] D. Divsalar, S. Dolinar, and F. Pollara, "Serial concatenated trellis coded modulation with rate-1 inner code," in *Proc. Global Communications. Conf.*, 2000.

[31] H. Tullberg and P.Siegel, "Serial concatenated trellis coded modulation with inner rate-1 accumulate code," in *Proc. Global Communications Conf.*, Nov 2001.

[32] K. Narayanan, "Effect of precoding on the convergence of turbo equalization for partial response channels," *IEEE Journal on Sel. Areas in Comm.*, vol. 19, pp. 686–698, April 2001.

[33] I. Lee, "The effect of a precoder on serially concatenated coding systems with an ISI channel," *IEEE Trans. on Comm.*, vol. 49, pp. 1168–1175, July 2001.

[34] A. Ashikhmin, G. Kramer, S. ten Brink, "Extrinsic information transfer functions: A model and two properties," in *Proc. CISS, Princeton*, March 2002.

[35] M. Tüchler and J. Hagenauer, "Exit charts of irregular codes," in *Proc. CISS, Princeton*, March 2002.

[36] S. ten Brink, "Exploiting the chain rule of mutual information for the design of iterative decoding schemes," in *Proc. of the 39th Allerton Conf., Monticello, IL*, Oct 2001.

[37] T. Cormen et al., *Introduction to Algorithms, 2nd Ed.* Cambridge, U.S.A.: MIT Press, 2001.

[38] D. Bertsekas, A, Nedic, and A. Ozdaglar, *Convexity, Duality, and Langrange Multipliers*. Cambridge, U.S.A.: Lecture notes, MIT class 6291, 2001.

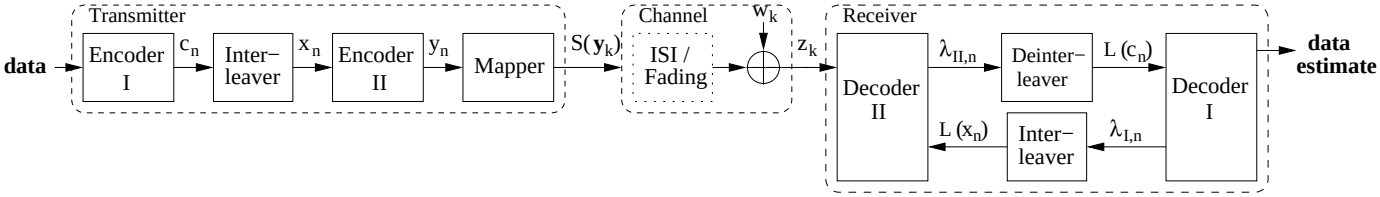

Fig. 1. A serially concatenated system.

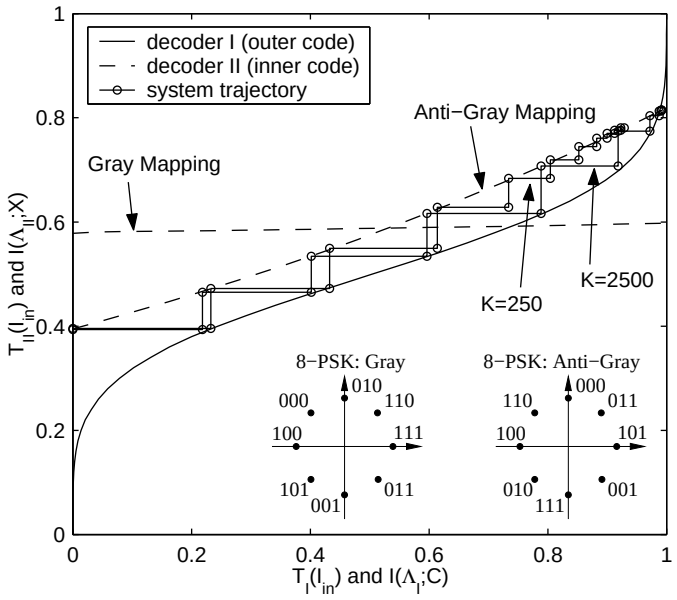

Fig. 2. Transfer functions and system trajectory of a serially concatenated system.

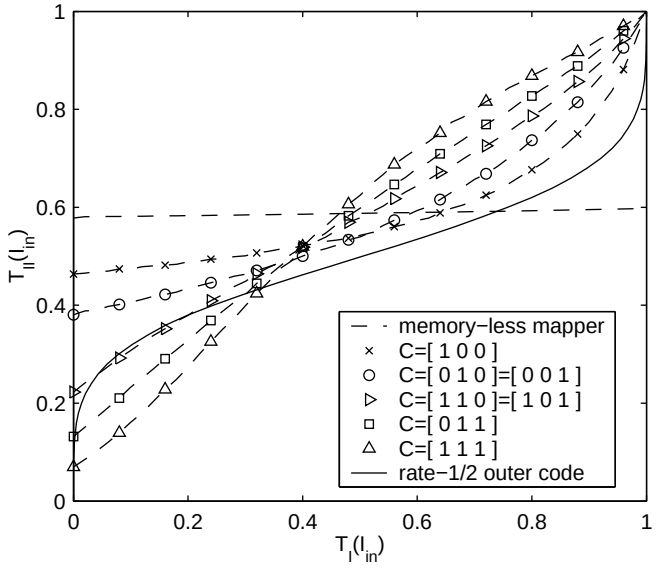

Fig. 3. Transfer functions of 2-state decoders for 8-PSK at 2.7 dB E_b/N_0.

T in bpc	$\frac{E_b}{N_0}\|_{min}$ in dB	$\frac{E_b}{N_0}\|_{th}$ in dB	**A**	**B**	**C**	**D**	$\alpha_1, \alpha_2, ..., \alpha_{17}$ in percent
0.5	-0.854	-0.515	1	11	10	$\mathbf{I}_2$	00 36 42 00 00 00 14 00 00 00 04 01 00 02 00 00 01
1.0	0.040	0.184	1	111	110	$\mathbf{I}_3$	00 10 25 26 00 00 22 00 04 00 05 01 00 04 00 01 02
1.5	1.000	1.163	1	1111	1010	$\mathbf{I}_4$	00 00 09 47 00 00 15 01 14 00 00 05 00 06 00 00 03
2.0	2.045	2.061	1	11111	11100	$\mathbf{I}_5$	00 26 00 16 00 18 00 00 15 06 00 00 08 05 00 00 06
2.5	3.141	3.167	1	11111	11100	$\mathbf{I}_5$	00 00 08 23 00 00 15 01 13 00 08 05 00 16 00 00 11
3.0	4.269	4.358	1	11111	11100	$\mathbf{I}_5$	00 00 00 00 18 10 00 00 17 04 05 00 07 18 00 00 21
3.5	5.435	5.656	1	11111	11100	$\mathbf{I}_5$	00 00 00 00 00 00 00 29 02 01 02 00 19 10 00 00 37
4.0	6.620	6.847	1	111111	001010	$\mathbf{I}_6$	00 00 00 00 00 00 00 00 35 12 00 00 10 21 00 00 22
4.5	7.838	8.125	1	111111	001010	$\mathbf{I}_6$	00 00 00 00 00 00 00 00 00 06 30 00 00 24 03 00 37
5.0	9.151	9.609	1	111111	001010	$\mathbf{I}_6$	00 00 00 00 00 00 00 00 00 00 00 00 00 33 01 00 66

TABLE I

OPTIMIZED IRREGULAR OUTER CODES $\mathcal{C}$ AND INNER RATE-1, 2-STATE ENCODERS FOR SEVERAL THROUGHPUTS T IN BITS PER CHANNEL USE.

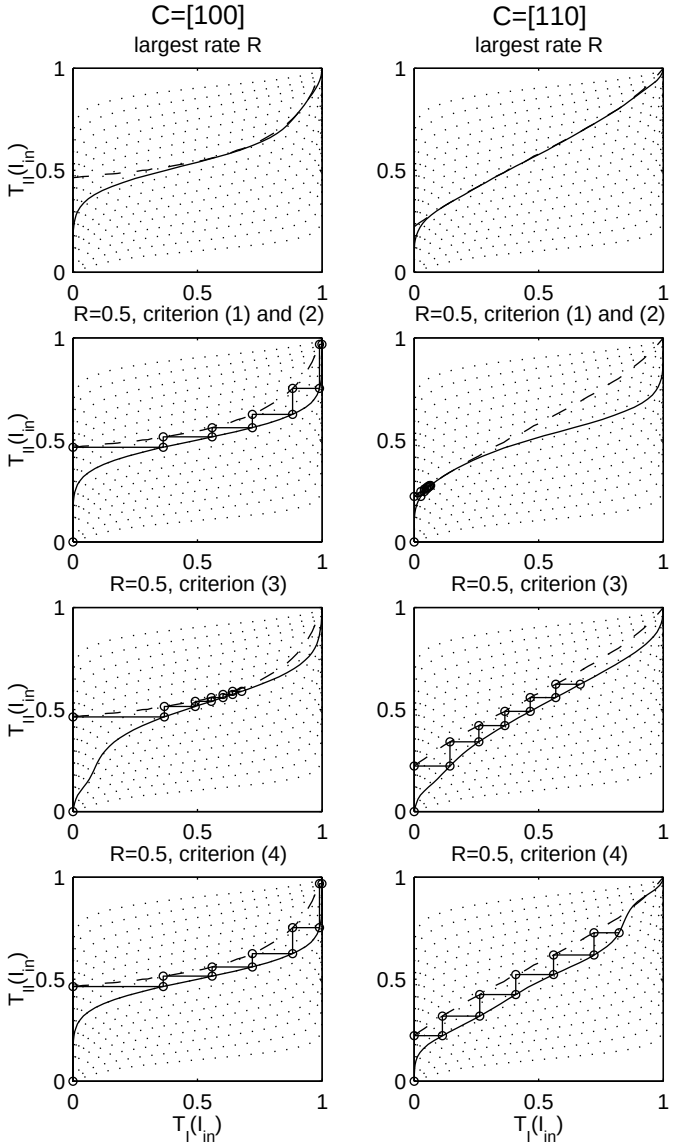

Fig. 4. Optimization of the outer code via the transfer function $T_{\rm I}(I_{in})$ of decoder I given the transfer function $T_{\rm II}(I_{in})$ of two inner encoders II with $\mathbf{A}=1$, $\mathbf{B}=[111]$, $\mathbf{C}\in\{[100],[110]\}$, and $\mathbf{D}=\mathbf{I}_3$ at 2.7 dB E_b/N_0.

. . . . $T_{{\rm I},k}(i)$ of the subcodes —— $T_{\rm I}(i)$ of the outer decoder I

– – $T_{\rm II}(i)$ of the inner decoder II –o– system trajectory after 6 iterations

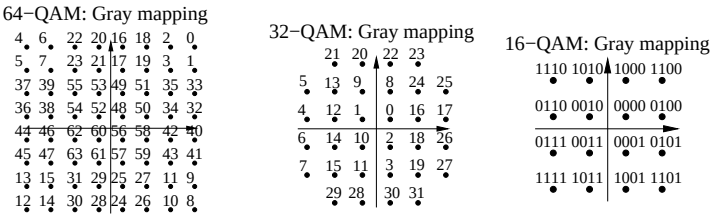

Fig. 5. Some higher-order signal alphabets $\mathcal{S}$ with Gray mapping. Decimal numbers denote a tuple $\mathbf{y}_k$ as follows: $3 \triangleq (1,1,0,0)$.

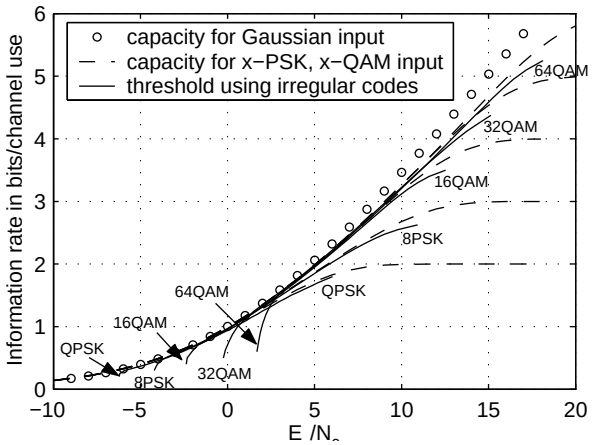

Fig. 6. Achievable throughputs using irregular codes.

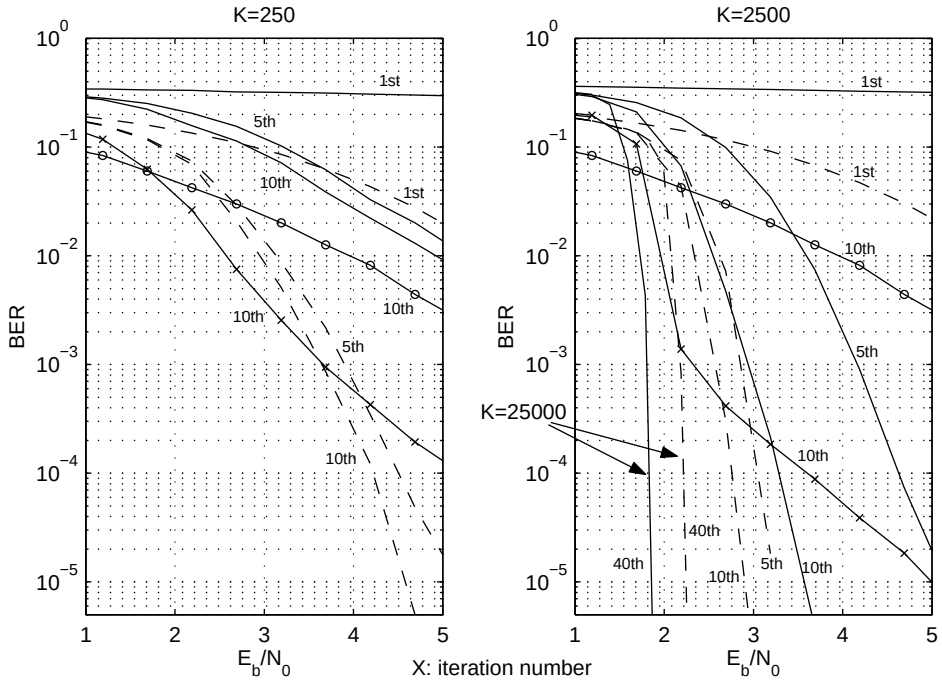

Fig. 7. BER performance of a serially concatenated system for different block lengths.
−o− memoryless mapper, Gray −x− memoryless mapper, Anti-Gray
—— optimized for early convergence – – optimized to maximize μ_6