

Research Article

Differentially Encoded LDPC Codes—Part I: Special Case of Product Accumulate Codes

Jing Li (Tiffany)

Electrical and Computer Engineering, Lehigh University, Bethlehem, PA 18015, USA

Correspondence should be addressed to Jing Li (Tiffany), jingli@ece.lehigh.edu

Received 19 November 2007; Accepted 6 March 2008

Recommended by Yonghui Li

Part I of a two-part series investigates product accumulate codes, a special class of differentially-encoded low density parity check (DE-LDPC) codes with high performance and low complexity, on flat Rayleigh fading channels. In the coherent detection case, Divsalar's simple bounds and iterative thresholds using density evolution are computed to quantify the code performance at finite and infinite lengths, respectively. In the noncoherent detection case, a simple iterative differential detection and decoding (IDDD) receiver is proposed and shown to be robust for different Doppler shifts. Extrinsic information transfer (EXIT) charts reveal that, with pilot symbol assisted differential detection, the widespread practice of inserting pilot symbols to terminate the trellis actually incurs a loss in capacity, and a more efficient way is to separate pilots from the trellis. Through analysis and simulations, it is shown that PA codes perform very well with both coherent and noncoherent detections. The more general case of DE-LDPC codes, where the LDPC part may take arbitrary degree profiles, is studied in Part II Li 2008.

Copyright © 2008 Jing Li (Tiffany). This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

1. INTRODUCTION

The discovery of turbo codes and the rediscovery of low-density parity-check (LDPC) codes have renewed the research frontier of capacity-achieving codes [1, 2]. They also revolutionized the coding theory by establishing a new soft-iterative paradigm, where long powerful codes are constructed from short simple codes and decoded through iterative message exchange and successive refinement between component decoders. Compared to turbo codes, LDPC codes boast a lower complexity in decoding, a richer variety in code construction, and not being patented.

One important application of LDPC codes is wireless communications, where sender and receiver communicate through, for example, a no-line-of-sight land-mobile channel that is characterized by the Rayleigh fading model. It is well-recognized that LDPC codes perform remarkably well on Rayleigh fading channels, that is, assuming the carrier phase is perfectly synchronized and coherent detection is performed; but what if otherwise?

It should be noted that, due to practical issues like complexity, acquisition time, sensitivity to tracking errors,

and phase ambiguity, coherent detection may become expensive or infeasible in some cases. In the context of noncoherent detection, the technique of differential encoding becomes immediately relevant. Differential encoding admits simple noncoherent differential detection which solves phase ambiguity and requires only frequency synchronization (often more readily available than phase synchronization). Viewed from the coding perspective, performing differential encoding is essentially concatenating the original code with an accumulator, or, a recursive convolutional code in the form of $1/(1 + D)$.

In this series of two-part papers, we investigate the theory and practice of LDPC codes with differential encoding. We start with a special class of differentially encoded LDPC (DE-LDPC) codes, namely, product accumulate (PA) codes (Part I), and then we move to the general case where an arbitrary (random) LDPC code is concatenated with an accumulator (Part II) [3].

Product accumulate codes, proposed in [4] and depicted in Figure 1, are a class of serially concatenated codes, where the inner code is a differential encoder, and the outer code is a parallel concatenation of two branches of single-parity

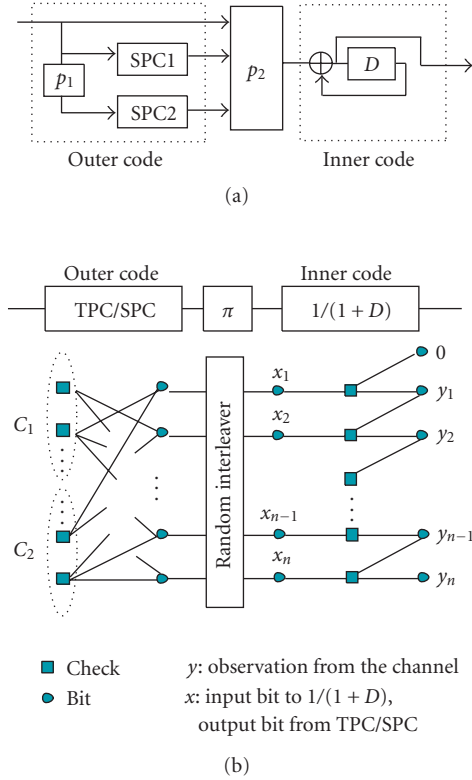


FIGURE 1: PA codes (a), code structure (b). Graph representation.

check (SPC) codes or a structured LDPC code comprising degree-1 and degree-2 variable nodes. Since the accumulator can also be described using a sparse bipartite graph, a PA code is, overall, an LDPC code. Alternatively, it may also be regarded as a differentially-encoded LDPC code, to emphasize the impact of the inner differential encoder. The reasons to study PA codes are multifold. First, PA codes exhibit an interesting threshold property and remarkable performance, and are well established as a class of “good” codes with rates $\geq 1/2$ and performance within a few tenths of a dB from the Shannon limit [4]. Here, “good” is in the sense defined by MacKay [2]. Second, PA codes are desirable for their simplicity. They are simple to describe, simple to encode and decode, and simple enough to allow rigorous theoretical analysis [4]. Comparatively, a random LDPC code can be expensive to describe and expensive to implement in VLSI (due to the difficulty of routing and wiring). Finally, PA codes are intrinsically differentially encoded, which naturally permits noncoherent differential detection without needing additional components.

The primary interest is the noncoherent detection case, but for completeness of investigation and for comparison, we also include the case of coherent detection. Under the assumption that phase information is known, we compute Divsalar’s simple bounds to benchmark the performance of PA codes at finite code lengths [5], and we evaluate iterative thresholds using density evolution (DE) to benchmark the performance of PA codes at infinite code lengths. The asymptotic thresholds reveal that PA codes are about from

0.6 to 0.7 dB better than regular LDPC codes, but 0.5 dB worse than optimal irregular LDPC codes (whose maximal left degree is 50) on Rayleigh fading channels with coherent detection. Simulations of fairly long block lengths show a good agreement with the analytical results.

When phase information is unavailable, the decoder/detector will either proceed without phase information (completely blind), or entails some (coarse) estimation and compensation in the decoding process. We regard either case as noncoherent detection. The presence of a differential encoder in the code structure readily lands PA codes to noncoherent differential detection. Conventional differential detection (CDD) operates on two symbol intervals and recovers the information by subtracting the phase of the previous signal sample from the current signal sample. It is cheap to implement, but suffers as much as from 4 to 5 dB in bit error rate (BER) performance [6]. Closing the gap between CDD and *differentially encoded coherent detection* generally requires the extension of the observation window beyond two symbol intervals. The result is multisymbol differential detection (MSDD), exemplified by maximum-likelihood (ML) multisymbol detection, trellis-based multisymbol detection with per-survivor processing, and their variations [7, 8]. MSDD performs significantly better than CDD, at the cost of a considerably higher complexity which increases exponentially with the window size. To preserve the simplicity of PA codes, here we propose an efficient *iterative differential detection and decoding* (IDDD) receiver which is robust against various Doppler spreads and can perform, for example, within 1 dB from coherent detection on fast fading channels.

We investigate the impact of pilot spacing and filter lengths, and we show that the proposed PA IDDD receiver requires very moderate number of pilot symbols, compared to, for example, turbo codes [6]. It is quite expected that the percentage of pilots directly affects the performance especially on very fast fading channels, but much less expected is that how these pilot symbols are inserted also makes a huge difference. Through extrinsic information transfer (EXIT) analysis [9], we show that the widespread practice of inserting pilot symbols to periodically terminate the trellis of the differential encoder inevitably [6, 7] incurs a loss in code capacity. We attribute this to what we call the “trellis segmentation” effect, namely, error events are made much shorter in the periodically terminated trellis than otherwise. We propose that pilot symbols be separated from the trellis structure, and simulation confirms the efficiency of the new method.

From analysis and simulation, it is fair to say that PA codes perform well both with coherent and noncoherent detection. In Part II of this series of papers, we will show that conventional LDPC codes, such as regular LDPC codes with uniform column weight of 3 and optimized irregular ones reported in literature, actually perform poorly with noncoherent differential detection. We will discuss why, how, and how much we can change the situation.

The rest of the paper is organized as follows. Section 2 introduces PA codes and the channel model. Section 3 analyzes the coherently detected PA codes on fading channels

using Divsalar's simple bounds and iterative thresholds. Section 4 discusses noncoherent detection and decoding of PA codes and performs EXIT analysis. Finally, Section 5 summarizes the paper.

2. PA CODES AND CHANNEL MODEL

2.1. Channel model

We consider binary phase shift-keying (BPSK) signaling ($0 \rightarrow +1, 1 \rightarrow -1$) over flat Rayleigh fading channels. Assuming proper sampling of the outputs from the matched filter, the received discrete-time baseband signal can be modeled as $r_k = \alpha_k e^{j\theta_k} s_k + n_k$, where s_k is the BPSK-modulated signal, n_k is the i.i.d. complex AWGN with zero mean and variance $\sigma^2 = N_0/2$ in each dimension. The fading amplitude α_k is modeled as a normalized Rayleigh random variable with $E[\alpha_k^2] = 1$ and pdf $p_A(\alpha_k) = 2\alpha_k \exp(-\alpha_k^2)$ for $\alpha_k > 0$, and the fading phase θ_k is uniformly distributed over $[0, 2\pi]$.

For fully interleaved channels, α_k 's and θ_k 's are independent for different time indexes k . For insufficiently interleaved channels, they are correlated. We use the Jakes' isotropic scattering land mobile Rayleigh channel model to describe the correlated Rayleigh process which has autocorrelation $\mathcal{R}_k = (1/2)\mathcal{J}_0(2k\pi f_d T_s)$, where $f_d T_s$ is the normalized Doppler spread, and $\mathcal{J}_0(\cdot)$ is the 0th order Bessel function of the first kind.

Throughout the paper, θ_k is assumed known perfectly to the receiver/decoder in the coherent detection case, and unknown (and needs to be worked around) in the noncoherent detection case. Further, the receiver is said to have channel state information (CSI) if α_k known (irrespective of θ_k), and no CSI otherwise.

2.2. PA codes and decoding analysis

A product accumulate code, as illustrated in Figure 1(a), consists of an accumulator (or a differential encoder) as the inner code, and a parallel concatenation of 2 branches of single-parity check codes as the outer code. PA codes are decoded through a soft-iterative process where soft extrinsic information is exchanged between component decoders conforming to the turbo principle. The outer code, modeled as a structured LDPC code, is decoded using the message-passing algorithm. The inner code, taking the convolutional form of $1/(1+D)$, may be decoded either using the trellis-based BCJR algorithm, or a graph-based message-passing algorithm. The latter, thanks to the cycle-free code graph of $1/(1+D)$, performs as optimally as the BCJR algorithm, but consumes several times less of complexity [4, 10]. Thus, the entire code can be efficiently decoded through a unified message-passing algorithm, driven by the initial log-likelihood ratio (LLR) values extracted from the channel [4]. For Rayleigh fading channels with perfect CSI, that is, α_k is known $\forall k$, the initial channel-LLRs are computed using

$$L_{\text{ch}}^{\text{CSI}}(s_k) = \frac{4\alpha_k}{N_0} r_k, \quad (1)$$

and for Rayleigh fading channels without CSI,

$$L_{\text{ch}}^{\text{NCSI}}(s_k) = \frac{4E[\alpha_k]}{N_0} r_k, \quad (2)$$

where $E[\alpha] = \sqrt{\pi}/2$ is the mean of α . Due to the space limitation, we omit the details of the overall message-passing algorithm, but refer readers to [4].

3. COHERENT DETECTION

This section investigates the coherent detection case on Rayleigh fading channels. We employ Divsalar's simple bounds and the iterative threshold to analyze the ensemble average performance of PA codes, and simulate individual PA codes at short and long lengths.

3.1. Simple bounds

Union bounds are simple to compute, but are rather loose at low SNRs. Divsalar's simple bound is possibly one of the best closed-form bounds [5]. Like many other tight bounds, the simple bound is based on the second Gallager's bounding techniques [1]. By using numerical integration instead of a Chernoff bound and by reducing the number of codewords to be included in the bound, Divsalar was able to tighten the bound to overcome the cutoff rate limitation. Since the simple bound requires the knowledge of the distance spectrum, a hard-to-attain property especially for concatenated codes, it has not seen wide application. Here, the simplicity of PA codes permits an accurate computation of the ensemble-average distance spectrum (whose details can be found in [4]), and thus enables the exploitation of the simple bound.

The technique of the simple bound allows for the computation of either a maximum likelihood (ML) threshold in the asymptotic sense [4, 5], or a performance upper bound with respect to a given finite length. Divsalar derived the general form of the simple bound on independent Rayleigh fading channels with perfect CSI. Following a similar line of reasoning, below we extend it to the case of non-CSI.

3.1.1. Gallager's second bounding technique

Gallager's second bounding technique sets the base for many tight bounds including the simple bounds [1]. It states that

$$\Pr(\text{error}) \leq \Pr(\text{error}, \mathbf{r} \in \mathfrak{R}) + \Pr(\mathbf{r} \notin \mathfrak{R}), \quad (3)$$

where $\mathbf{r} = \gamma \boldsymbol{\alpha} \mathbf{s} + \mathbf{n}$ is the received codeword (N -dimensional noise-corrupted vector), \mathbf{s} is the transmitted codeword vector, \mathbf{n} is the noise vector whose components are i.i.d. Gaussian random variables with zero mean and unit variance, γ is the known constant (in modulation), $\boldsymbol{\alpha}$ is the $N \times N$ matrix containing fading coefficients ($\boldsymbol{\alpha}$ is an identity matrix for AWGN channels), and \mathfrak{R} denotes a region in the observed space around the transmitted codeword. To get a tight bound, optimization and integration are usually needed to determine a meaningful \mathfrak{R} .

3.1.2. Divsalar's simple bound for independent rayleigh fading channels with CSI

For Rayleigh fading channels, the decision metric is based on the minimization of the norm $\|\mathbf{r} - \gamma\boldsymbol{\alpha}\mathbf{s}\|$, where \mathbf{s} , \mathbf{r} , and $\boldsymbol{\alpha}$ are the transmitted signal, received signal, and the fading amplitude in vector form, respectively, and γ is the amplitude of the transmitted signal such that $\gamma^2/2 = E_s/N_0$.

For a good approximation of the error using (3), and for computational simplicity, the decision region \mathfrak{R} was chosen as an N -dimensional hypersphere centered at $\eta\gamma\boldsymbol{\alpha}\mathbf{s}$ and with radius $\sqrt{N}\mathcal{R}$, where η and \mathcal{R} are the parameters to be optimized [5].

When perfect CSI is available, the effect of fading can be compensated through a linear transformation on $\gamma\boldsymbol{\alpha}\mathbf{s}$. In particular, a rotation $e^{j\varphi}$ and a rescaling ζ have shown to yield a good and analytically feasible solution [5]

$$\mathfrak{R} = \{\mathbf{r} \mid \|\mathbf{r} - \zeta e^{j\varphi} \gamma \boldsymbol{\alpha} \mathbf{s}\|^2 \leq N\mathcal{R}^2\}, \quad (4)$$

which leads to the upper bound of the error probability of an (N, K, R) code [5]

$$P(e) \leq \sum_{h=2}^{2\sqrt{N-K+1}} \min \left\{ e^{-N\mathbb{E}(c, \delta, \rho, \beta, \kappa, \phi)}, e^{N\gamma_N(\delta)} \frac{1}{\pi} \int_0^{\pi/2} \left[\frac{\sin^2 \theta}{\sin^2 \theta + c} \right]^h d\theta \right\}, \quad (5)$$

where

$$\begin{aligned} \mathbb{E}(c, \delta, \rho, \beta, \kappa, \phi) &= -\rho\gamma_N(\delta) + \frac{\rho}{2} \log \frac{\beta}{\rho} + \frac{1-\rho}{2} \log \frac{1-\beta}{1-\rho} \\ &\quad + \rho\delta \log(1+c(1-2\kappa\phi)) \\ &\quad + \rho(1-\delta) \log \left[1+c \left(1-2\kappa\phi - \frac{(1-\kappa^2)\rho}{\beta} \right) \right] \\ &\quad + (1-\rho) \log \left[1+c \left(\frac{1-\rho(1-2\kappa\phi)}{1-\rho}, -\frac{(1-\rho(1-\kappa))^2}{(1-\rho)(1-\beta)} \right) \right], \end{aligned} \quad (6)$$

$$c = \frac{\gamma^2}{2} = \frac{E_s}{N_0} = R \frac{E_b}{N_0}, \quad (7)$$

$$\delta = \frac{h}{N}, \quad (8)$$

$$\begin{aligned} \gamma(\delta) &= \gamma_N \left(\frac{h}{N} \right) \\ &= \begin{cases} \frac{1}{N} \log(A_h), & \text{for word error rate,} \\ \frac{1}{N} \log \left(\sum_w \frac{w}{K} A_{w,h} \right), & \text{for bit error rate.} \end{cases} \end{aligned} \quad (9)$$

3.1.3. Extension of the simple bound to the case of No CSI

Another simple and reasonable choice of the decision region is an ellipsoid centered at $\eta\gamma\mathbf{s}$, which can be obtained by

rescaling each coordinate of \mathbf{r} so as to compensate for the effect of fading

$$\mathfrak{R} = \{\mathbf{r} \mid \|\boldsymbol{\alpha}^{-1}\mathbf{r} - \eta\gamma\mathbf{s}\|^2 \leq N\mathcal{R}^2\}, \quad (10)$$

where η and \mathcal{R} are optimized. For independent Rayleigh channels without CSI, since accurate information on $\boldsymbol{\alpha}$ is unavailable, we resort to the expectation of the fading coefficient $\boldsymbol{\alpha}^{-1} \approx E[\boldsymbol{\alpha}^{-1}] = (1/0.8862)\mathbf{I}$ in (10), where \mathbf{I} is an identity matrix. By replicating the computations described in [5], we obtain the upper bound of the bit error rate for independent Rayleigh channels without CSI:

$$P(e) \leq \sum_{h=2}^{2\sqrt{N-K+1}} \min \left\{ e^{-N\mathbb{E}(c, \delta, \rho)}, \exp \left(\frac{h\nu N\gamma_N(\delta)}{c} \right) \times \left(1 - \frac{2}{\sqrt{1+2/\nu+1}} \right)^h \right\}, \quad (11)$$

where

$$\begin{aligned} \mathbb{E}(c, \delta, \rho) &= -\frac{1}{2} \log(1-\rho + \rho e^{2\gamma_N(\delta)}) \\ &\quad + c \left(1 + \frac{1-\delta}{\delta} \left(1 + \frac{1-\rho}{\rho} e^{-2\gamma_N(\delta)} \right) \right)^{-1}, \\ \rho &= \left(1 + \frac{1-\beta}{\beta} e^{2\gamma_N(\delta)} \right)^{-1}, \\ \beta &= \left\{ 2c \frac{1-\delta}{\delta} (1 - e^{-2\gamma_N(\delta)})^{-1} \right. \\ &\quad \left. + \left(\frac{1-\delta}{\delta} \right)^2 [(1+c)^2 - 1] \right\}^{1/2} - (1+c) \frac{1-\delta}{\delta}, \end{aligned} \quad (12)$$

$$c = E^2[\alpha] \frac{\gamma^2}{2} = 0.8862^2 R \frac{E_b}{N_0},$$

$$\nu = \sqrt{(\gamma^2/2)^2 - 1} = \sqrt{(RE_b/N_0)^2 - 1},$$

and δ and $\gamma_N(\delta)$ are the same as in (8) and (9).

Please note that the aforesaid extension to the fading case with no CSI slightly loosens the simple bound, but it preserves the computational simplicity. It is possible for a more sophisticated transformation to yield tighter bounds but not necessarily a feasible analytical expression.

Figure 2 plots the simulated BER performance and the simple bound of a (1024,512) PA code on independent Rayleigh fading channels with and without CSI. Since an optimal ML decoder is assumed, and the ensemble average distance spectrum is used, in the computation, the simple bound represents the best ensemble average performance, and may not accurately reflect the individual PA code being simulated. Nevertheless, we see that the bound is fairly tight. It provides a useful indication of the code performance at SNRs below the cut-off rate, and, at high SNRs, it joins with the union bound to predict the error floor.

3.2. Threshold computation via the iterative analysis

The ML performance bound evaluated in the previous subsection factors in the finite length of a PA code ensemble,

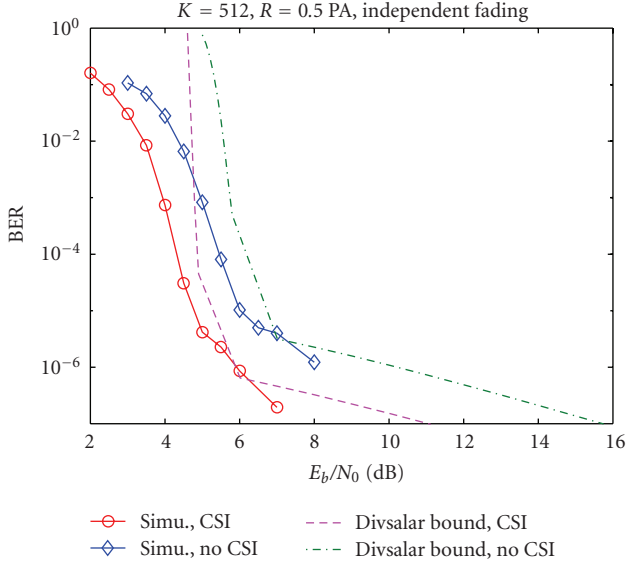


FIGURE 2: Divsalar simple bounds for $R = 0.5$ PA codes.

but the assumption of an ML decoder may be optimistic. Below we account for the iterative nature of the practical decoder and compute an asymptotic iterative threshold using the renowned method of density evolution [11].

A useful tool for analyzing the iterative decoding process of sparse-graph codes, density evolution examines the probability density function (pdf) of exchanging messages in each step and can, literally speaking, track the entire decoding process. In general, we are more interested in the asymptotic SNR thresholds, η , which are defined as the critical channel condition that is required for the decoding process to converge unanimously to the correct decision:

$$\eta(\text{dB}) = \min_{\text{SNR}} \left\{ \lim_{l \rightarrow \infty} \int_{-\infty}^0 y f_{L_y}^{(l)}(\zeta) d\zeta = 0 \right\}, \quad (13)$$

where $y = \pm 1$ is the BPSK modulated signals, and $f_{L_y}^{(l)}$ denotes the pdf of LLR information on y after the l th decoding iteration.

Tracking the density of the messages requires the computation of the initial pdf of the LLR messages from the channel, and the transformation of the message pdf's in each step of the decoding process. Although Gaussian approximation is reported to incur only very little inaccuracy on AWGN channels [12, 13], the deviation is larger on fading channels, since the pdf of the initial LLRs from a fading channel looks different from a Gaussian distribution. Hence, exact density evolution is used to preserve accuracy.

3.2.1. Initial LLR pdf from the channel

Hou et al. showed in [14] that the pdf of the LLRs from independent Rayleigh channels with perfect CSI is given

by (assuming BPSK signaling and the all-zero sequence is transmitted)

$$\begin{aligned} f_{L_{ch,y}}^{\text{CSI}}(\zeta) &= \int_0^\infty \mathcal{N}\left(\frac{4\alpha^2}{N_0}, \frac{8\alpha^2}{N_0}\right) \cdot p(\alpha) d\alpha, \\ &= \sqrt{\frac{N_0}{4\pi}} \exp\left(-\frac{\zeta(\sqrt{N_0+1}-1)}{2}\right) \\ &\quad \times \int_0^\infty \exp\left(-\frac{((\zeta N_0/4\alpha) - \alpha\sqrt{N_0+1})^2}{N_0}\right) d\alpha. \end{aligned} \quad (14)$$

Using integrals from [15], we further simplify (14) to

$$f_{L_{ch,y}}^{\text{CSI}}(\zeta) = \frac{N_0}{4\sqrt{1+N_0}} \cdot \exp\left(\frac{\zeta - |\zeta|\sqrt{1+N_0}}{2}\right). \quad (15)$$

For the case when CSI is not available to the receiver, we assume that the Rayleigh-faded and AWGN-corrupted signals follow a Gaussian distribution in the most probable region. The pdf of the initial messages is then derived as

$$f_{L_{ch,y}}^{\text{NCSI}}(\zeta) = \frac{\Delta^2 \sqrt{N_0} \kappa^{N_0}}{\pi} \left(\kappa + \sqrt{2} \Delta \zeta Q\left(-\frac{\Delta \zeta}{\sqrt{\pi}}\right) \right), \quad (16)$$

where $\Delta = \sqrt{N_0/2(N_0+1)}$, $\kappa = \exp(-\Delta^2 \zeta^2/2\pi)$, and $Q(x) = (1/\sqrt{2\pi}) \int_x^\infty e^{-z^2/2} dz$.

3.2.2. Evolution of LLR pdf in the decoder

To track the evolution of the pdf's along the iterative process can either employ Monte Carlo simulation, or, more accurately and more efficiently, to proceed analytically through discretized density evolution. The latter is possible due to the simplicity in the code structure and in the decoding algorithm of PA codes. As a self-contained discussion, we summarize the major steps of the discretized density evolution of PA codes in the Appendix, but for details, please refer to [4].

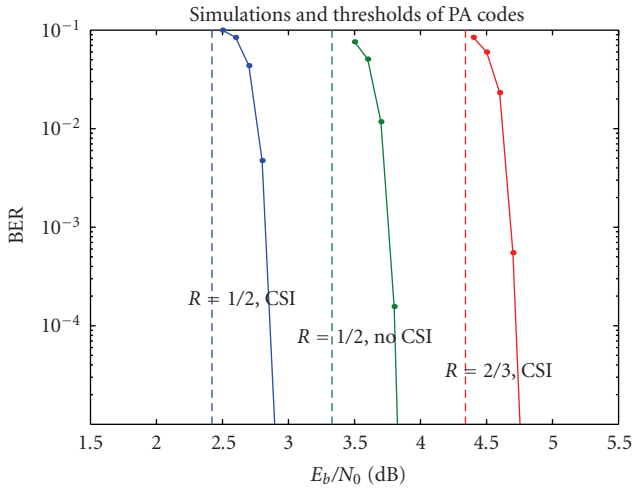
Using (15) for perfect CSI case or (16) for no CSI case (i.e., substituting them in (A.4) and (A.5) in the Appendix), the thresholds of PA codes on Rayleigh channels can be computed through (A.3) to (A.12) in the Appendix. The computed thresholds are a good indication of the performance limit as the code length and the number of iterations increase without bound.

Figure 3 plots the thresholds as well as the simulation results of PA codes on independent Rayleigh channels with and without CSI. We see that the analytical results are consistent with the simulation results for fairly large block sizes. Here, simulations are evaluated after the 50th iteration. As the block size and the number of iterations continue to increase, we expect the actual performance to converge to the thresholds.

Table 1 compares the thresholds of PA codes with those of LDPC codes for several code rates. The ergodic capacity of the independent Rayleigh fading channel is also listed as reference. We see that the thresholds of PA codes are about 0.6 dB from the channel capacity, and simulations of fairly

TABLE 1: Thresholds (E_b/N_0 in dB) of PA codes on Rayleigh channels ($(3, \rho)$ LDPC data by courtesy of Hou et al. [14]).

Rate	Flat Rayleigh CSI			Flat Rayleigh no CSI		
	Capacity (dB)	PA (dB)	LDPC (dB)	Capacity (dB)	PA (dB)	LDPC (dB)
0.5	1.8	2.42	3.06	2.6	3.33	4.06
0.6	3.0	3.56	—	3.8	4.48	—
2/3	3.7	4.34	4.72	4.4	5.15	5.74

FIGURE 3: Thresholds computed using density evolution and simulations (data block size $K = 64$ K).

large block sizes are about 0.3-0.4 dB from the thresholds. Compared to the thresholds of LDPC codes reported in [14], rate 1/2 PA codes are from about 0.6-0.7 dB better (asymptotically) than $(3,6)$ -regular LDPC codes, but are about 0.5 dB worse (asymptotically) than irregular LDPC codes. It should be noted that these irregular LDPC codes are specifically optimized for Rayleigh fading channels and have maximum variable node degree of 50. It is fair to say that PA codes perform on par with LDPC codes (using coherent detection).

3.3. Simulation with coherent detection

To benchmark the performance of coherently detected PA codes, several PA configurations are simulated on correlated and independent Rayleigh fading channels. In each global iteration (i.e., iteration between the inner decoder and the outer decoder), two local iterations of the outer decoding are performed. This scheduling is found to strike the best trade-off between complexity and performance (with coherent detection).

3.3.1. Coherent BPSK on independent rayleigh channels

Figure 4 shows the performances of rate 1/2 PA codes on independent Rayleigh fading channels with and without channel state information, respectively. Bit error rates after 20, 30, and 50 (global) iterations are plotted, and data

block sizes from short to large (512, 1 K, 4 K, and 64 K) are evaluated to demonstrate the interleaving gain. For comparison purpose, the corresponding channel capacities are also shown. The simulated performance degradation due to the lack of CSI is about 0.9 dB, which is consistent with the gap between the respective channel capacities.

Compared to the $(3, 6)$ -regular LDPC codes reported in [14], the performance of this rate 1/2, codeword length $N = 128 \times 1024 = 1.3 \times 10^5$ PA code is about 0.4 and 0.25 dB better than regular LDPC codes of length $N = 10^5$ and 10^6 on independent Rayleigh channels. It is possible that optimized irregular LDPC codes will outperform PA codes (as indicated by their thresholds), but for regular codes, PA codes seem one of the best.

3.3.2. Coherent BPSK on correlated rayleigh channels

Figure 5 shows the performance of PA codes on correlated fading channels. Perfect CSI is assumed available to the receiver, and an interleaver exists between the PA code and the channel (to partially break up the correlation between the neighboring bits). Short PA codes with rate 1/2 and 3/4 are simulated on two common fading scenarios with normalized Doppler spreads $f_d T_s = 0.01$ and 0.001 , respectively. As expected, the performance deteriorates rapidly as $f_d T_s$ decreases, since slower Doppler rate brings smaller diversity order. Due to the interleaver between the PA code and the channel, the impact of slow Doppler rate is less severe for larger block sizes than for smaller ones. Whereas $K = 1$ K PA code loses about 7 dB at $\text{BER} = 10^{-4}$ as $f_d T_s$ changes from 0.01 to 0.001, the loss with $K = 4$ K PA code is less than 5 dB.

To illuminate how well short PA codes perform on correlated channels, we compare them with turbo codes (which are the best-known codes at short code lengths) in Figure 5. The comparing turbo code has 16-state component convolutional codes whose generator polynomial is $(1, 35/23)_{\text{oct}}$ and which are decoded using log-domain BCJR algorithm. Code rate is 0.75, data block size is 4 K, and S-random interleavers are used in both codes to lower the possible error floors. Curves plotted are for PA codes at the 10th iteration and turbo codes at the 6th iteration. We observe that turbo codes perform about 0.6 and 0.7 dB better than PA codes for $f_d T_s = 0.001$ and 0.01, respectively. However, it should be noted that this performance gain comes at a price of a considerably higher complexity. While the message-passing decoding of a rate-0.75 PA code at the 10th iteration requires about 267 operations per data bit [4], the log-domain BCJR decoding of a rate-0.75 turbo code at the 6th iteration requires as many as 9720 operations per data

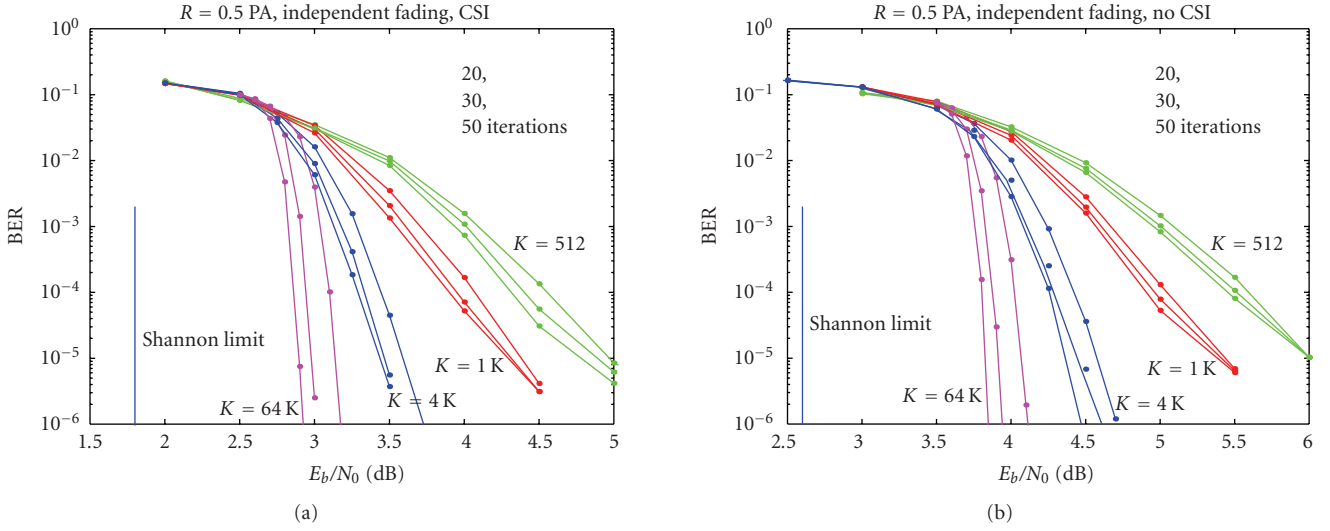


FIGURE 4: Performance of PA codes on independent Rayleigh fading channels. Code rate 0.5, data block size 512, 1 K, 4 K, 64 K. (a) With CSI; (b) without CSI.

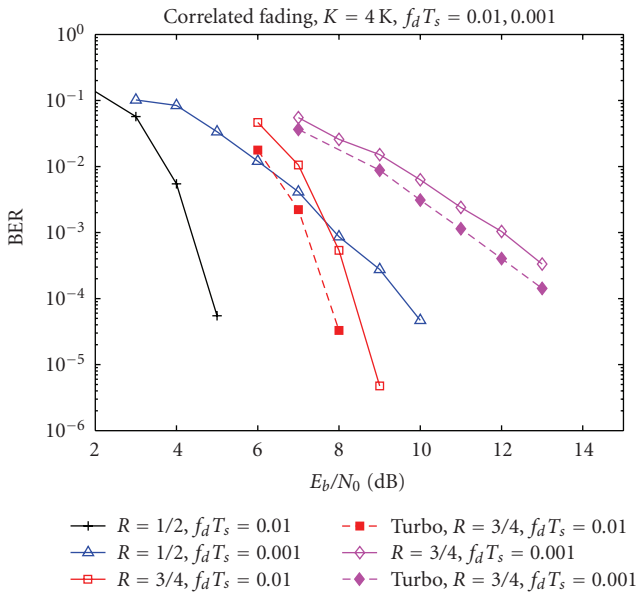


FIGURE 5: Performance of PA codes on correlated Rayleigh fading channels with CSI. Data block length 4 K, normalized Doppler rate $f_d T_s = 0.01, 0.001$, rate of PA codes 0.5 and 0.75, rate of turbo codes 0.75, component codes of the turbo code $(1, 35/23)_{\text{Oct}}$, 10 iterations for PA codes, and 6 iterations for turbo codes.

bit, a complexity 35 times larger. Hence, PA codes are still attractive for providing good performance at low cost.

4. NONCOHERENT DETECTION OF PA CODES

This section considers noncoherent detection. The channel model of interest is a Rayleigh fading channel with correlated fading coefficients.

4.1. Iterative differential detection and decoding

PA codes are inherently differentially encoded which makes it convenient for noncoherent differential detection. Although multiple symbol differential detection is possible, for complexity concerns, we consider a simple iterative differential detection and decoding receiver, whose structure is shown in Figure 6. The IDDD receiver consists of a conventional differential detector with 2-symbol observation window (the current and the previous), a phase tracking filter and the original PA decoder (that used in coherent detection [4]). Trellis structure is employed to assist the detection and decoding of the inner differential code $1/(1 + D)$, but unlike the case of multiple symbol detection, the trellis is not expanded and has 2 states only. Soft information is passed back and forth among different parts of the receiver conforming to the turbo principle. Let x denote the input to the inner differential encoder or the output from the outer code, and let y denote the output from the differential encoder or the symbol to be put on the channel (see Figure 6). The differential encoder implements $y_k = x_k y_{k-1}$ for $x_k, y_k \in \{\pm 1\}$ (BPSK signal mapping $0 \rightarrow +1, 1 \rightarrow -1$). The channel reception is given by $r_k = \alpha_k e^{j\theta_k} y_k + n_k$, where the channel amplitudes (α_k 's) and phases (θ_k 's) are correlated, and the complex white Gaussian noise samples (n_k 's) are independent.

In theory, differential decoding does not require pilot symbols. In practice, however, pilot symbols are inserted periodically even with multiple symbol detection, to avoid catastrophic error propagation in differential decoding. This is particularly so for the fast fading case where phases (θ_k) are changing rapidly (will show later). Hence, some of the r_k 's (and y_k 's) in the received sequence are pilot symbols.

We use L to denote the LLR information, superscript (q) to denote the q th (global) iteration, and subscript i, o, ch , and e to denote the quantities associated with the inner

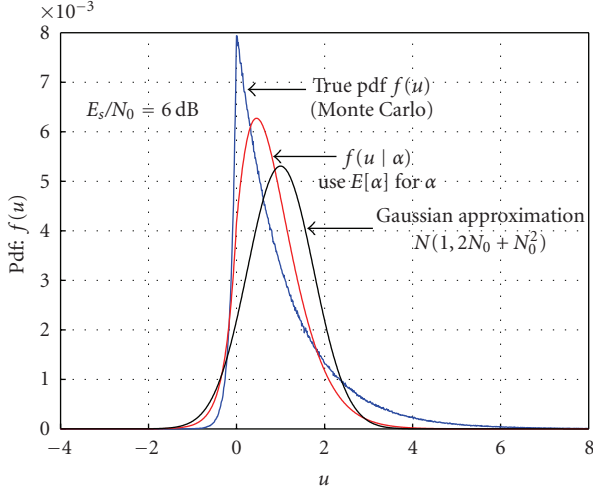


FIGURE 7: Distribution of $u_k = \text{Re}\{y_k y_{k-1}^*\}$ in a conventional differential detection (assume “+1” transmitted).

be used, which makes the first iteration exactly the same as subsequent iterations. This third option then leads to pilot symbol assisted modulation (PSAM), which has slightly higher complexity than using differential detection in the first iteration.

To see how accurate the above treatments are, we plot in Figure 7 several curves approximating the pdf of u_k . From the most sharp and asymmetric to the least sharp and symmetric, these curves denote the exact pdf of $f_{U|X}(u | x = +1)$ from Monte Carlo simulations (histogram, can be regarded as the numerical evaluation of (18)), the “mean- α approximated” pdf from (19) and the Gaussian approximated pdf from (21). From the figure, the Gaussian approximation does not reflect the true pdf well, but this inaccuracy turns out not severely affecting the overall IDDD performance. As shown later in Figure 13, all the three treatments (Gaussian approximation, mean- α approximation, and PSAM) result in very similar decoding performance. We attribute this to the fact that the inaccuracy affects mostly the first iteration, and subsequent iterations can help mitigate the loss. Thus, Gaussian approximation still presents itself as a simple and viable approach for noncoherent differential decoding.

4.1.3. Channel estimator

The channel estimator in the IDDD receiver (Figure 6) may be implemented in several ways. Here we use a linear filter of $(2L + 1)$ taps to estimate α_k 's and θ_k 's in the q th iteration

$$\hat{\alpha}_k^{(q)} e^{j\hat{\theta}_k^{(q)}} = \sum_{l=-L}^L p_l \hat{y}_{k-l}^{(q-1)} r_{k-l}, \quad (23)$$

where p_l denotes the coefficient of the l th filter tap, and $\hat{y}_k^{(q-1)}$ denotes the estimate on y_k from the feedback of the previous iteration. For soft feedback, $\hat{y}_k^{(q-1)}$ is computed using $\hat{y}_k^{(q-1)} = \tanh((L_{e,i}^{(q-1)}(y_k))/2)$, and for hard feedback, $\hat{y}_k^{(q-1)} = \text{sign}(L_{e,i}^{(q-1)}(y_k))$. The LLR message $L_{e,i}^{(q-1)}(y_k)$ is generated toge-

ther with $L_{e,i}^{(q-1)}(x_k)$ by the inner decoder in the $(q - 1)$ th decoding iteration (please refer to [4] for the step-by-step message-passing decoding algorithm of $1/(1 + D)$ code). In the first iteration, $L_{e,i}^{(0)}(y_k)$'s are initiated as zeros for coded bits and a large positive number (i.e., $+\infty$) for pilot symbols.

Regarding the choice of the filter, we take a Wiener filter, since it is known to be optimal for estimating channel gain in the minimum mean-square-error (MMSE) sense, when the correlation of the fading process, \mathcal{R}_k 's, are known [18]. The filter coefficients, $p_{-L}, p_{-L+1}, \dots, p_L$, are obtained from the Wiener-Hopf equation

$$\begin{pmatrix} \mathcal{R}_0 - N_0 & \mathcal{R}_1 & \cdots & \mathcal{R}_{L-1} \\ \mathcal{R}_1 & \mathcal{R}_0 - N_0 & \cdots & \mathcal{R}_{L-2} \\ \cdots & \cdots & \cdots & \cdots \\ \mathcal{R}_{L-1} & \mathcal{R}_{L-2} & \cdots & \mathcal{R}_0 - N_0 \end{pmatrix} \cdot \begin{pmatrix} p_{-L} \\ p_{-(L-1)} \\ \cdots \\ p_L \end{pmatrix} = \begin{pmatrix} \mathcal{R}_{-L} \\ \mathcal{R}_{-L-1} \\ \cdots \\ \mathcal{R}_L \end{pmatrix}, \quad (24)$$

where $\mathcal{R}_k = (1/2)\mathcal{J}_0(2k\pi f_d T_s)$. Since the computation of p_l 's from (24) involves an inverse operation on a matrix (one-time job), it may not be computable when the matrix becomes (near) singular, which occurs when the channel is very slow fading. In such cases, a low-pass filter, or a simple “moving average” can be used [6].

4.2. Analysis of pilot insertion through EXIT charts

4.2.1. EXIT charts

We perform EXIT analysis [9] to generate further insights into PA codes and the proposed noncoherent IDDD receiver. In EXIT charts, the exchange of extrinsic information is visualized as a decoding/detection trajectory, allowing the prediction of the decoding convergence and thresholds [9]. Several quantities, like the bit error rate, the mean of the extrinsic LLR information, and the equivalent SNR value, were previously used to depict the characteristics and relations of the component decoders, but the mutual information is shown to be the most robust among all [9]. The mutual information between the binary bit y_k and its corresponding LLR values is defined as

$$\begin{aligned} I(Y, L(Y)) &\triangleq \frac{1}{2} \sum_{y=\pm 1} \int_{-\infty}^{\infty} f_{L(y)}(\eta | Y = y) \\ &\quad \cdot \log_2 \frac{2f_{L(y)}(\eta | Y = y)}{f_{L(y)}(\eta | Y = +1) + f_{L(y)}(\eta | Y = -1)} d\eta \\ &= \int_{-\infty}^{\infty} f_{L(y)}(\eta | Y = +1) \\ &\quad \cdot \log_2 \frac{2f_{L(y)}(\eta | Y = +1)}{f_{L(y)}(\eta | Y = +1) + f_{L(y)}(-\eta | Y = +1)} d\eta \\ &= 1 - \int_{-\infty}^{\infty} f_{L(y)}(\eta | Y = +1) \cdot \log_2(1 + e^{-\eta}) d\eta, \end{aligned} \quad (25)$$

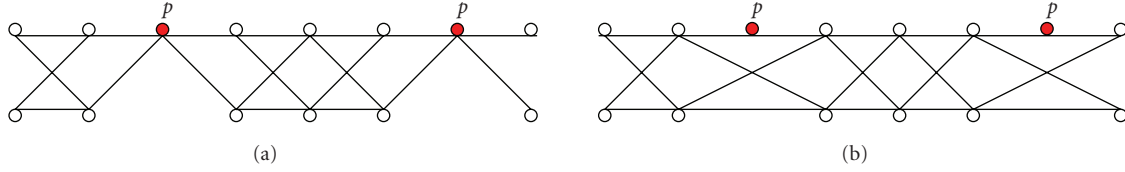


FIGURE 8: Trellis diagram of binary differential PSK with pilot insertion. (a) Pilot symbols periodically terminate the trellis. (b) Pilot symbols are separated from the trellis structure.

where $L(Y)$ is either the *a priori* information $L_a(Y)$ or the extrinsic information $L_e(Y)$, and $f_{L(Y)}(\eta | Y = y)$ is the conditional pdf. The second equality holds when the channel is output symmetric such that $f_{L(Y)}(\eta | Y = -y) = f_{L(Y)}(-\eta | Y = y)$, and the third equality holds when the received messages satisfy the consistency condition (also known as the symmetry condition): $f_{L(Y)}(\eta | Y = y) = f_{L(Y)}(-\eta | Y = y)e^{y\eta}$ [11]. Note that the consistency condition is an invariant in the message-passing process on a number of channels including the AWGN channel and the independent Rayleigh fading channel with perfect CSI; but it is not preserved on fading channels without CSI or with estimated (thus imperfect) CSI, since the initial density function evaluated in the latter cases is but an approximation of the actual pdf of the LLR messages. Thus, (25) should be used to compute the mutual information in those cases. We use the X -axis to represent the mutual information to the inner code (*a priori*) or from the outer code (extrinsic), denoted as $I_{a,i}/I_{e,o}$, and the Y -axis to represent the mutual information from the inner code or to the outer code, denoted as $I_{e,i}/I_{a,o}$.

4.2.2. Pilot symbol insertion

A practicality issue about noncoherent detection is pilot insertion. The number of pilot symbols inserted should be sufficient to attain a reasonable track of the channel, but not in excess. Many researchers have reported that excessive pilot symbols not only cause wasteful bandwidth expansion, but actually degrade the overall performance, since the energy compensation for the rate loss due to excessive pilot more than outweighs the gain that can be obtained by a finer channel tracking. This trade-off issue has long been noted in literature, but little attention has been given to another issue of no less importance, namely, how pilots should be inserted when differential encoding or other trellis-based coding/modulation front-end is used.

There exist at least two ways to insert pilot symbols in a differential encoder. The widespread approach is to periodically terminate the trellis [6, 7], as shown in Figure 8(a), such that pilot symbols are used to estimate the channel and at the same time participate in the trellis decoding. Seemingly plausible, this turns out to be a bad strategy, since segmenting the trellis into small chunks significantly increases the number of short error events, and consequently incurs a loss in performance.

The negative effect of trellis segmentation is best illustrated by the EXIT chart in Figure 9. EXIT curves corre-

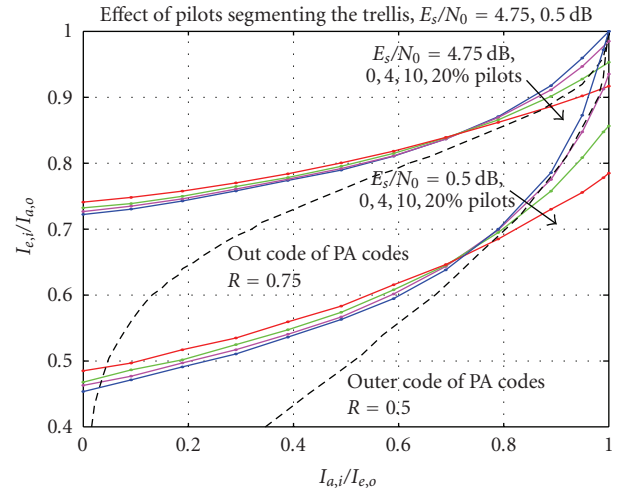


FIGURE 9: The effect of pilot symbols segmenting the trellis on the performance of the differential decoder. Normalized Doppler rate $f_d T_s = 0.01$, $E_s/N_0 = 4.75$ dB and 0 dB, perfect CSI.

sponding to the differential decoder with 0%, 4%, 10%, and 20% pilot insertion are plotted for two different SNR values. To eliminate the impact of other factors, the four curves in each SNR set are given the same energy per transmitted symbol and perfect knowledge on the fading phase and amplitude is provided to all the decoders (irrespective of the number of pilot symbols). Thus the difference between the curves in each family is only due to the difference in pilot spacing. At the left end of the curves (when input mutual information is small), a larger number of pilot symbols correspond to a better performance (a higher output mutual information). This is because when little information is provided from the outer code, pilot symbols become the primary contributor to *a priori* information. However, the situation is completely reversed toward the right end of the EXIT curves. We see that more pilot symbols actually degrade the performance, the reason being, given sufficient information provided by the outer code, pilot symbols no longer constitute the key source of *a priori* information; on the other hand, they segment the trellis and shorten error events, rendering an opposite effect to spectrum thinning and thus deteriorating the performance. The performance loss is more severe when more pilot symbols are inserted and when the code is operating at a relatively low SNR level. It is worth noting, for example, with 20% of pilot insertion (pilot spacing is 5), even provided with a perfect

mutual information from the outer code ($I_{a,i} = 1$, but the channel remains noisy), the trellis decoder nevertheless fails to produce sufficient output mutual information $I_{e,i}$. As such, the inner EXIT curve is bound to intersect the outer EXIT curve at a rather early stage of the iterative process, causing the iterative decoder to fail at a high BER level (not to mention this EXIT curve has 20% more of energy consumption than the no-pilot case).

The implication of this EXIT analysis is that the widespread approach of inserting pilot symbols as part of the trellis could cause deficiency for differential encoding (and other serial concatenated schemes with inner trellis codes). Specifically, unless the outer code is itself a capacity-achieving code at some SNR, the inner and outer EXIT curves will intersect, result in convergence failure and cause error floors. We observe that the more the pilot symbols, the higher the error floor; and the lower the code rate (lower SNR), the more severe the impact. It is therefore particularly important to keep the number of pilot symbols in such schemes minimal, so that error floors do not occur too early. This analysis also suggests an alternative, and potentially better-performing, way of pilot insertion, namely, separating pilots from the trellis and thus not affecting error events; see Figure 8(b).

It should be pointed out, that the level of the impact caused by trellis segmentation may be very different for different outer codes. Many (outer) codes, including single parity check codes, block turbo codes (i.e., turbo product codes) and convolutional codes, will see a large impact, since these (outer) codes require sufficient input information in order to produce perfect output information, or, put another way, these codes alone are not “good” codes (good in the sense as MacKay defined in [2]). However, “good” codes like LDPC codes will likely see a much smaller impact. This is because an ideal LDPC code has an EXIT curve shaping like a box (e.g., see [3, Figure 3]) which can produce perfect output information as long as the input information is above some threshold (without requiring $I_{a,i} = 1$). Alternatively, one may also interpret it as: ideal LDPC codes have large minimum distances and are capable of correcting short error events including those caused by the segmentation effect.

To verify the analytical results, we simulate the performance of a rate 1/2, data block size $K = 32$ K PA code with different strategies of pilot insertion; see Figure 10. The normalized Doppler spread is $f_d T_s = 0.01$, and error rates evaluated after 10 decoding iterations. Solid lines represent the cases where perfect channel knowledge is known to the receiver, and dashed lines represent the case where noncoherent detection is used. Comparing solid curves, we see a drastic performance gap results from different strategies of pilot insertion. In this specific case, by segmenting the trellis every 10 symbols, trellis-segmented pilot insertion losses more than 3 dB at BER of 10^{-4} than otherwise. The dashed curve corresponds to the same PA code noncoherently-detected via the IDDD receiver discussed before, where 10% of pilot symbols are inserted using the strategy in Figure 8(b) and where an 81-tap wiener filter is used to estimate the channel. It is interesting to note that if one overlooks the impact of pilot insertion strategies, one might arrive

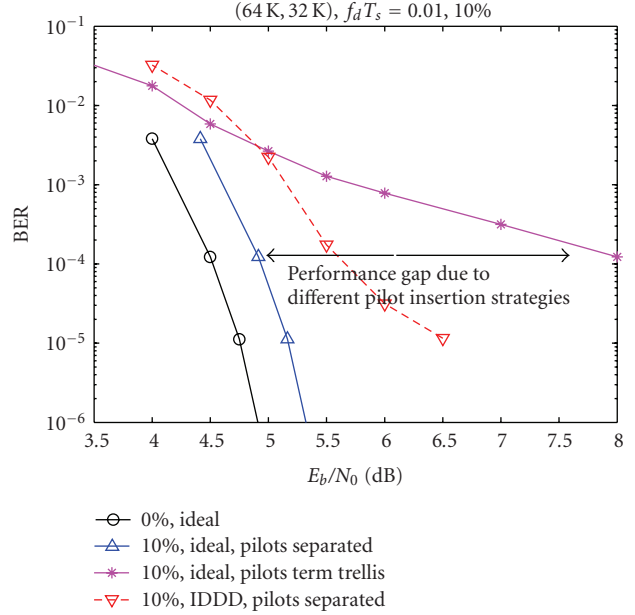


FIGURE 10: Performance of PA codes with different pilot insertion strategies. Normalized Doppler rate $f_d T_s = 0.01$, code rate 0.5, data block size 32 K, 0% or 10% pilot insertion, 10 iterations.

at a paradox result that noncoherent detection (dashed line) performs (noticeably) better than coherent detection (rightmost solid line)!

4.3. Impact of the pilot symbol spacing and filter length

We now investigate how the number of pilot symbols and the length of the estimation filter affect the performance of noncoherent detection. Figure 11 illustrates the impact of different pilot spacing on the BER performance of fast fading channels where the normalized Doppler spread takes $f_d T_s = 0.05, 0.02$ or 0.01 . We observe the following: (1) The IDDD receiver is rather robust for different Doppler rates. (2) Smaller pilot spacing, such as < 6 symbols, is undesirable, whose consumption of additional energy more than outweighs any gain it may bring. (3) The code performance at high Doppler rates is more sensitive to pilot spacing than that at lower Doppler rates. At the normalized Doppler rate of 0.01 (already fast fading), noncoherently detected PA codes tolerate pilot spacing as small as 6 symbols and as large as 45 to 50 symbols (put aside the bandwidth issue); but at very fast Doppler rate of 0.05, pilot spacing beyond 7–9 symbols will soon cause drastic performance degradation. For comparison, we also plot the case where pilot symbols periodically terminate the trellis (dashed line), which, due to trellis segmentation, experiences inferior performance when pilot spacing is small. Compared to differentially encoded turbo codes [6], PA codes appear to require fewer pilot symbols (we note that in the study of differentially encoded turbo codes in [6], the authors terminated the trellis periodically with pilot symbols, which may have made the

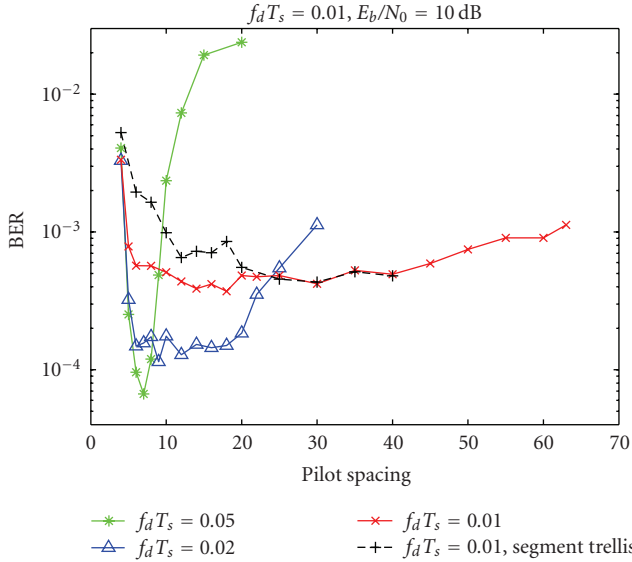


FIGURE 11: Effect of the number of pilot symbols on the performance of noncoherently detected PA codes on correlated Rayleigh channels with $f_d T_s = 0.01$. Code rate 0.75, data block size 1 K, filter length 65, 10 (global) iterations, 4 (local) iterations within the outer code of PA codes.

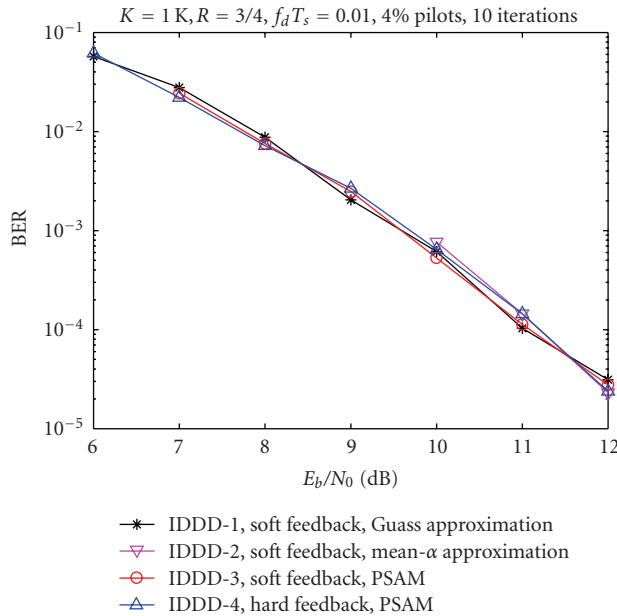


FIGURE 12: Comparison of BER performance for several noncoherent receiver strategies on correlated Rayleigh channels with $f_d T_s = 0.01$. Code rate 0.75, data block size 1 K, 4% of bandwidth expansion, filter length 65, 10 (global) iterations each with 4 (local) iterations for the outer decoding.

tolerant range of pilot spacing (at the small spacing end smaller than otherwise).

The impact of the length of the channel tracking filter is also studied. We observe that while the filter length affects the overall performance, the impact is limited compared to pilot spacing. This is consistent with what has been reported

in other studies [6] and is not a new discovery. Hence, we omit the plot.

4.4. Simulation results of noncoherent detection

The performance of noncoherently detected PA codes on fast Rayleigh fading channels are presented below. Unless otherwise indicated, the BER curves shown are after 10 global iterations, and in each global iteration, 4 to 6 local iterations of the outer code are performed. We have chosen these parameters on the basis of a set of simulations and trading-off between performance and complexity.

4.4.1. Noncoherent detection of PA codes with different receiver strategies

We compare the BER performance of 4 types of IDDD strategies for a $K = 1$ K, $R = 3/4$ PA code on a $f_d T_s = 0.01$ Rayleigh fading channel in Figure 12. “IDDD-1” uses the conventional differential detection with Gaussian approximation (22) to compute $L_{ch}(x_k)$ in the first iteration, and soft feedback of \hat{y}_k in all iterations to assist channel estimation; “IDDD-2” uses conventional differential detection with “mean- α ” approximation (20) in the first iteration and soft feedback in all iterations; “IDDD-3” is PSAM with soft feedback; and “IDDD-4” is PSAM with hard feedback. In all cases, 4% of pilot symbols are inserted and curves shown are after 10 iterations. Different decoding strategies in the first iteration does not affect the performance much, and the performance is not very sensitive to hard or soft feedback either. Although not shown, simulations of a long PA code ($K = 48$ K) of the same (high) rate ($R = 3/4$) reveal a similar phenomenon. It is possible, however, that other codes may be more sensitive to the difference in decoding strategies especially the difference in the feedback information [6].

4.4.2. Comparison of noncoherent detection with coherent detection

Figure 13 shows the performance of rate 3/4 PA codes after 10 iterations on fast Rayleigh fading channels with Doppler rate $T_s f_d = 0.01$. Short block size of 1 K and large block size of 48 K are evaluated. In each case, a family of 5 BER-versus- E_b/N_0 curves, accounting for rate loss due to pilot insertion, are plotted. The three leftmost curves are the ideal coherent case with knowledge of fading amplitudes and phases provided to the receiver, and the two right curves are the noncoherent case where IDDD is used to track amplitudes and phases. In both the coherent and the noncoherent case, trellis segmentation incurs a small performance loss, but since the pilot spacing is not very small (every 25 symbols), the effect is not as drastic as the case in Figure 10. The noncoherent cases are about 1 dB and 0.55 dB away from the ideal coherent case at BER of 10^{-4} for block sizes of 48 K and 1 K, respectively. This satisfying performance is achieved with only 4% of pilot insertion and a very low-complexity IDDD receiver.

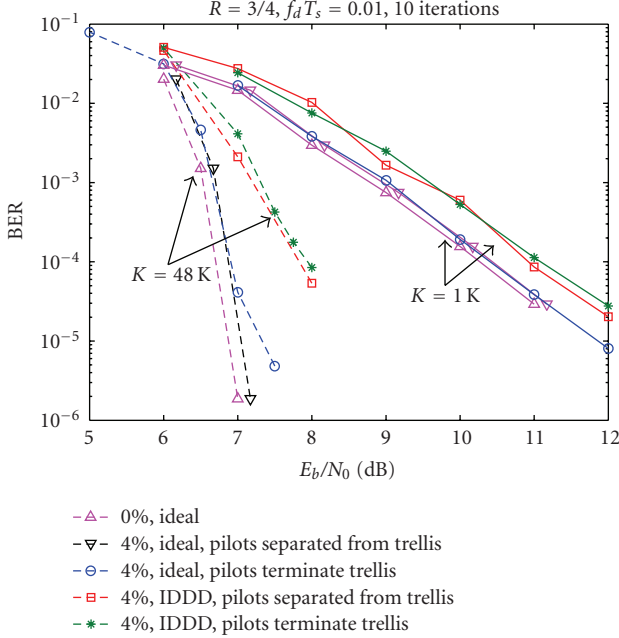


FIGURE 13: Comparison of BER performance for several transmission/reception strategies for PA codes of large and small block sizes on correlated Rayleigh channels with $f_d T_s = 0.01$. Code rate 0.75, data block size 48 K and 1 K, 4% of bandwidth expansion, filter length 65, 10 (global) iterations each with 4 (local) iterations for the outer decoding.

5. CONCLUSION

Previous work has established product accumulate codes as a class of provenly “good” codes on AWGN channels, with low linear-time complexity and performances close to the Shannon limit. This paper performs a comprehensive study of product accumulate codes on Rayleigh fading channels with both coherent and noncoherent detection. Useful analytical tools including Divsalar’s simple bounds, density evolution, and EXIT charts are employed, and extensive simulations are conducted. It is shown that PA codes not only perform remarkably well with coherent detection, but the embedded differential encoder makes them naturally suitable for noncoherent detection. A simple iterative differential detection and decoding (IDDD) strategy allows PA codes to perform only 1 dB away from the coherent case. Another useful finding reveals that the widespread practice of inserting pilot symbols to terminate the trellis actually incurs performance loss compared to when pilot symbols are inserted as separate parts from the trellis.

We conclude by proposing product accumulate codes as a promising low-cost candidate for wireless applications. The advantages of PA codes include (i) they perform very well with coherent and noncoherent detection (especially at high rates), (ii) the performance is comparable to turbo and LDPC codes, yet PA codes require much less decoding complexity than turbo codes and much less encoding complexity and memory than random LDPC codes, and (iii) the regular structure of PA codes makes it possible for low-cost implementation in hardware.

APPENDIX

DISCRETIZED DENSITY EVOLUTION FOR PA CODES

Using message-passing decoding, the relevant operations on the messages (in LLR form) include the sum in the real domain and the tanh operation (also known as the *check* operation or \boxplus operation). For independent messages to add together, the resulting pdf of the sum is the discrete convolution (denoted by $*$) of the component pdf’s which can be efficiently implemented using a fast Fourier transform (FFT). For the tanh operation on messages, define:

$$\gamma = \alpha \boxplus \beta \triangleq Q(2 \tanh^{-1}(\tanh(\alpha/2) \tanh(\beta/2))),$$

where α , β , and γ are quantified messages, and Q defines the quantization operation. The pdf of γ , f_γ , can be computed using

$$f_\gamma[k] = \sum_{(i,j):k\Delta=i\Delta\boxplus j\Delta} f_\alpha[i] \cdot f_\beta[j], \quad (\text{A.1})$$

where Δ is the quantization interval. To simplify the notation, we denote this operation (A.1) as $f_\gamma = \mathcal{R}(f_\alpha, f_\beta)$, and using induction on the above equation, we further denote

$$\mathcal{R}^k(f_\alpha) \triangleq \underbrace{\mathcal{R}(f_\alpha, (\mathcal{R}(f_\alpha, \dots, \mathcal{R}(f_\alpha, f_\alpha) \dots))}_{k-1}. \quad (\text{A.2})$$

The following notations are also used:

- (i) $f_{L_{ch},y}$: pdf of the messages of the received signals y obtained from the channel (see Figure 1(b)),
- (ii) $f_{L_{o,x}}^{(k)}$: pdf of the (*a priori*) messages of the input x to the inner $1/(1+D)$ code in the k th iteration (obtained from the outer code in the $k-1$ th iteration) (see Figure 1(b)),
- (iii) $f_{L_{e,x}}^{(k)}$: pdf of the (extrinsic) messages passed from the inner code to the outer code in the k th iteration,
- (iv) $f_{L_{e1},(\cdot)}^{(k)}$ and $f_{L_{e2},(\cdot)}^{(k)}$: pdf’s of the extrinsic information computed from the upper and lower branch of the outer code in the k th iteration, respectively. Subscripts d and p denote data and parity bit, respectively. Obviously, $f_{L_{e2,d}}^{(0)} = f_{L_{e2,p}}^{(0)} = \delta(0)$, the Kronecker delta function.

The discretized density evolution of a rate $t/(t+2)$ PA code can then be summarized as follows [4]:

$$\text{initialization: } f_{L_{o,x}}^{(0)} = f_{L_{e,y}}^{(0)} = f_{L_{e1,d}}^{(0)} = f_{L_{e2,d}}^{(0)} = \delta(0), \quad (\text{A.3})$$

$$\text{inner code: } f_{L_{e,y}}^{(k)} = \mathcal{R}(f_{L_{o,x}}^{(k-1)}, f_{L_{ch,y}} * f_{L_{e,y}}^{(k-1)}), \quad (\text{A.4})$$

$$f_{L_{e,x}}^{(k)} = \mathcal{R}^2(f_{L_{ch,y}} * f_{L_{e,y}}^{(k)}), \quad (\text{A.5})$$

$$\text{inner-to-outer: } f_{L_{o,d}}^{(k)} = f_{L_{e,x}}^{(k)}, \quad (\text{A.6})$$

$$f_{L_{o,p}}^{(k)} = f_{L_{e,x}}^{(k)}, \quad (\text{A.7})$$

$$\text{outer code: } f_{L_{e1},d}^{(k)} = \mathcal{R}\left(f_{L_{o,p}}^{(k)}, \mathcal{R}^{(t-1)}\left(f_{L_{o,d}}^{(k)} * f_{L_{e2},d}^{(k-1)}\right)\right), \quad (\text{A.8})$$

$$f_{L_{e1},p}^{(k)} = \mathcal{R}^t\left(f_{L_{o,d}}^{(k)} * f_{L_{e2},d}^{(k-1)}\right), \quad (\text{A.9})$$

$$f_{L_{e2},d}^{(k)} = \mathcal{R}\left(f_{L_{o,p}}^{(k)}, \mathcal{R}^{(t-1)}\left(f_{L_{o,d}}^{(k)} * f_{L_{e1},d}^{(k)}\right)\right), \quad (\text{A.10})$$

$$f_{L_{e2},p}^{(k)} = \mathcal{R}^t\left(f_{L_{o,d}}^{(k)} * f_{L_{e1},d}^{(k)}\right), \quad (\text{A.11})$$

$$\text{outer-to-inner: } f_{L_{o,x}}^{(k+1)} = \frac{t(f_{L_{e1},d}^{(k)} + f_{L_{e2},d}^{(k)})}{t+2} + \frac{f_{L_{e1},p}^{(k)} + f_{L_{e2},p}^{(k)}}{2t+2}. \quad (\text{A.12})$$

Although the outer code of PA codes can be viewed as an LDPC code, it is desirable to take a serial update procedure as described above rather than a parallel one as in a conventional LDPC code, since this allows the checks corresponding to the two SPC branches to take turns to update, which leads to a faster convergence [4].

ACKNOWLEDGMENTS

This research work is supported in part by the National Science Foundation under Grants no. CCF-0430634 and CCF-0635199, and by the Commonwealth of Pennsylvania through the Pennsylvania Infrastructure Technology Alliance (PITA).

REFERENCES

- [1] R. G. Gallager, *Low Density Parity Check Codes*, MIT Press, Cambridge, Mass, USA, 1963.
- [2] D. J. C. MacKay, "Good error-correcting codes based on very sparse matrices," *IEEE Transactions on Information Theory*, vol. 45, no. 2, pp. 399–431, 1999.
- [3] J. Li, "Differentially encoded LDPC codes—part II: general case and code optimization," to appear in *EURASIP Journal on Wireless Communications and Networking*.
- [4] J. Li, K. R. Narayanan, and C. N. Georghiades, "Product accumulate codes: a class of codes with near-capacity performance and low decoding complexity," *IEEE Transactions on Information Theory*, vol. 50, no. 1, pp. 31–46, 2004.
- [5] D. Divsalar and E. Biglieri, "Upper bounds to error probabilities of coded systems beyond the cutoff rate," in *Proceedings of the IEEE International Symposium on Information Theory (ISIT '00)*, p. 288, Sorrento, Italy, June 2000.
- [6] M. C. Valenti and B. D. Woerner, "Iterative channel estimation and decoding of pilot symbol assisted turbo codes over flat-fading channels," *IEEE Journal on Selected Areas in Communications*, vol. 19, no. 9, pp. 1697–1705, 2001.
- [7] P. Hoeher and J. Lodge, "'Turbo DPSK': iterative differential PSK demodulation and channel decoding," *IEEE Transactions on Communications*, vol. 47, no. 6, pp. 837–843, 1999.
- [8] M. Peleg and S. Shamai, "Iterative decoding of coded and interleaved noncoherent multiple symbol detected DPSK," *Electronics Letters*, vol. 33, no. 12, pp. 1018–1020, 1997.
- [9] S. T. Brink, "Convergence behavior of iteratively decoded parallel concatenated codes," *IEEE Transactions on Communications*, vol. 49, no. 10, pp. 1727–1737, 2001.
- [10] P. Robertson, E. Villebrun, and P. Hoeher, "A comparison of optimal and sub-optimal MAP decoding algorithms operating in the log domain," in *Proceedings of the IEEE International*

Conference on Communications (ICC '95), vol. 2, pp. 1009–1013, Seattle, Wash, USA, June 1995.

- [11] T. J. Richardson, M. A. Shokrollahi, and R. L. Urbanke, "Design of capacity-approaching irregular low-density parity-check codes," *IEEE Transactions on Information Theory*, vol. 47, no. 2, pp. 619–637, 2001.
- [12] S.-Y. Chung, T. J. Richardson, and R. L. Urbanke, "Analysis of sum-product decoding of low-density parity-check codes using a Gaussian approximation," *IEEE Transactions on Information Theory*, vol. 47, no. 2, pp. 657–670, 2001.
- [13] K. Xie and J. Li, "On accuracy of Gaussian assumption in iterative analysis for LDPC codes," in *Proceedings of IEEE International Symposium on Information Theory (ISIT 06)*, pp. 2398–2402, Seattle, Wash, USA, July 2006.
- [14] J. Hou, P. H. Siegel, and L. B. Milstein, "Performance analysis and code optimization of low density parity-check codes on Rayleigh fading channels," *IEEE Journal on Selected Areas in Communications*, vol. 19, no. 5, pp. 924–934, 2001.
- [15] I. S. Gradshteyn and I. M. Ryzhik, *Tables of Integrals, Series and Products*, Academic Press, New York, NY, USA, 1980.
- [16] G. L. Stuber, *Principles of Mobile Communications*, Kluwer Academic Publishers, Norwell, Mass, USA, 1996.
- [17] M. K. Simon and M.-S. Alouini, *Digital Communication over Fading Channels*, John Wiley & Sons, New York, NY, USA, 2000.
- [18] P. Hoeher, S. Kaiser, and P. Robertson, "Two-dimensional pilot-symbol-aided channel estimation by Wiener filtering," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP '97)*, pp. 1845–1848, Munich, Germany, April 1997.