Design of Low–Density Parity–Check Codes for Modulation and Detection

Stephan ten Brink, Gerhard Kramer, and Alexei Ashikhmin

Abstract— **A coding and modulation technique is studied where the coded bits of an irregular low–density parity–check (LDPC) code are passed directly to a modulator. At the receiver, the variable nodes of the LDPC decoder graph are connected to detector nodes, and iterative decoding is performed by viewing the variable and detector nodes as one decoder. The code is optimized by performing a curve–fitting using extrinsic information transfer (EXIT) charts. Design examples are given for additive white Gaussian noise (AWGN) channels, as well as multi–input, multi–output (MIMO) fading channels where the receiver, but not the transmitter, knows the channel. For the MIMO channels, the technique operates within 1.25dB of capacity for various antenna configurations, and thereby outperforms a scheme employing a standard parallel concatenated (turbo) code by wide margins when there are more transmit than receive antennas.**

Keywords— **Fading, iterative decoding, low–density parity–check codes, MIMO detection, mutual information**

I. Introduction

Iterative decoding of low–density parity–check (LDPC) codes is a powerful method for approaching capacity on noisy channels [1]–[7]. We consider two problems associated with LDPC codes. The first is how to combine the code with a modulator and detector. The second is how to design the code, i.e., how to choose good degree distributions for the modulator, channel and detector under consideration.

We approach the first problem by mapping the coded bits of an irregular LDPC code directly onto a modulation set. At the receiver, we consider the standard graphical representation of an LDPC decoder [3] and connect the LDPC variable nodes to detector nodes. We show that this structure is flexible enough to closely approach capacity for many communication systems.

We solve the second problem by using a *curve–fitting* procedure together with extrinsic information transfer (EXIT) charts [8]. The curve–fitting might be possible using other chart techniques, see e.g. [5]–[7], [9], [10]; we refer to [11] for a comparison of these tools. Our design methodology is illustrated for two types of channels: additive white Gaussian noise (AWGN) channels with binary phase shift keying (BPSK), and multi–input, multi–output (MIMO) fading channels with quadrature phase shift keying (QPSK). The MIMO code design can be be extended in a straightforward way to other modulators, channels and detectors.

This paper is organized as follows. In Section II we develop the curve–fitting procedure for designing LDPC codes for the AWGN channel. In Section III we describe how to extend this technique to other communication problems, and in particular to MIMO fading channels where the receiver, but not the transmitter, knows the channel. We design LDPC codes for ergodic fading, and compare their performance to a scheme employing a Universal Mobile Telecommunications System (UMTS) standard turbo code. The LDPC codes are shown to perform substantially better for channels having more transmit than receive antennas. Section IV summarizes our results.

The authors are with Bell Laboratories, Lucent Technologies, 600 Mountain Ave, Murray Hill, NJ. E–mail: {stenbrink,gkr,aea}@bell-labs.com.

II. Code Design for AWGN Channels

Consider a LDPC code of length n and (design) rate $R = k/n$ [5]. An iterative decoder for this code has n variable nodes, an edge interleaver, and $n-k$ check nodes [3]. The ith variable node represents the ith bit of the codeword. This bit is involved in $d_{v,i}$ parity checks so that its node has $d_{v,i}$ edges going into the edge interleaver. The edge interleaver connects the variable nodes to the check nodes, each of which represents a parity–check equation. The ith check node checks $d_{c,i}$ bits so that it has $d_{c,i}$ edges. The sets of variable and check nodes are referred to as the variable node decoder (VND) and check node decoder (CND), respectively. Iterative decoding is performed by passing messages between the VND and CND.

The decoder structure in Fig. 1 is similar to that of an iterative decoder for a serially concatenated code based on a mixture of inner repeat codes and a mixture of outer single parity check codes. This observation illustrates the close relation between LDPC decoding and other iterative schemes such as turbo decoding [12]. However, a serially concatenated decoder exchanges only *one* extrinsic message per outer codeword bit (or one message per information bit of the inner code), whereas the decoder in Fig. 1 exchanges *several* messages per codeword bit.

A. EXIT Charts

An *a posteriori* probability (APP) decoder converts channel and *a priori* log–likelihood ratios (or L–values) into *a posteriori* L–values. However, when embedded in an iterative decoding structure, the APP decoder passes on *extrinsic* L–values. We refer to [13] for an introduction to extrinsic information processing. An EXIT chart characterizes the decoder's operation. We use the notation of [14] and write I_A for the average mutual information between the bits on the decoder graph edges (which are the bits about which extrinsic information is passed) and the *a priori* L–values. Similarly, we write I_E for the average mutual information between the bits on the edges and the

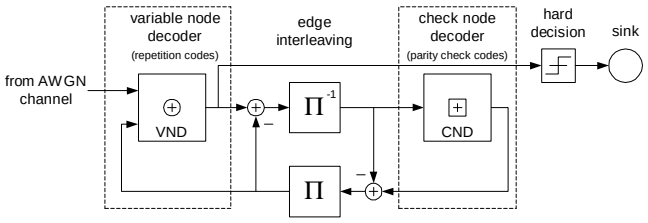


Fig. 1. Iterative decoder for a LDPC code.

extrinsic L–values. We refer to [8], [14] for further details on how to interpret these quantities.

B. EXIT Curve of the Inner Variable Node Decoder

A variable node of degree d_v has $d_v + 1$ incoming messages, d_v from the edge interleaver and one from the channel. The variable node decodes by adding L–values. More precisely, we have

$$L_{i,out} = L_{ch} + \sum_{j \neq i} L_{j,in} \tag{1}$$

where $L_{j,in}$ is the jth *a priori* L–value going into the variable node, $L_{i,out}$ is the ith extrinsic L–value coming out of the variable node, and L_{ch} is the L–value coming from the channel. Consider the AWGN channel with BPSK (± 1) modulation and noise variance σ_n^2. We define the normalized signal–to–noise ratio as $E_b/N_0 = 1/(2R\sigma_n^2)$. The variance of the channel L–value is thus (see the Appendix)

$$\sigma_{ch}^2 = 4/\sigma_n^2 = 8R \cdot E_b/N_0. \tag{2}$$

We assume that the $L_{j,in}$ are Gaussian distributed. The EXIT function of a degree–d_v variable node is then

$$I_{E,VND}\left(I_A, d_v, E_b/N_0, R\right) = J\left(\sqrt{(d_v - 1)\left[J^{-1}\left(I_A\right)\right]^2 + \sigma_{ch}^2}\right) \tag{3}$$

where the functions $J(\cdot)$ and $J^{-1}(\cdot)$ are given in the Appendix (see also [14], [15]).

Fig. 2 plots several variable node curves when $R = 1/2$ and $E_b/N_0 = 1$dB. The quantity

$$I_{E,VND}(0, d_v, E_b/N_0, R) = J(\sigma_{ch}) \tag{4}$$

is the capacity of the channel at the σ_{ch} that is being considered, which in our case is the capacity of the AWGN channel with BPSK modulation (see the Appendix). This result will be useful later on.

C. EXIT Curve of the Outer Check Node Decoder

The decoding of a degree d_c check node corresponds to the decoding of a length d_c (or rate $(d_c - 1)/d_c$) single parity check code. The output L–values are thus (see [13, Sec. II.A])

$$L_{i,out} = \ln \frac{1 - \prod_{j \neq i} \frac{1 - e^{L_{j,in}}}{1 + e^{L_{j,in}}}}{1 + \prod_{j \neq i} \frac{1 - e^{L_{j,in}}}{1 + e^{L_{j,in}}}}. \tag{5}$$

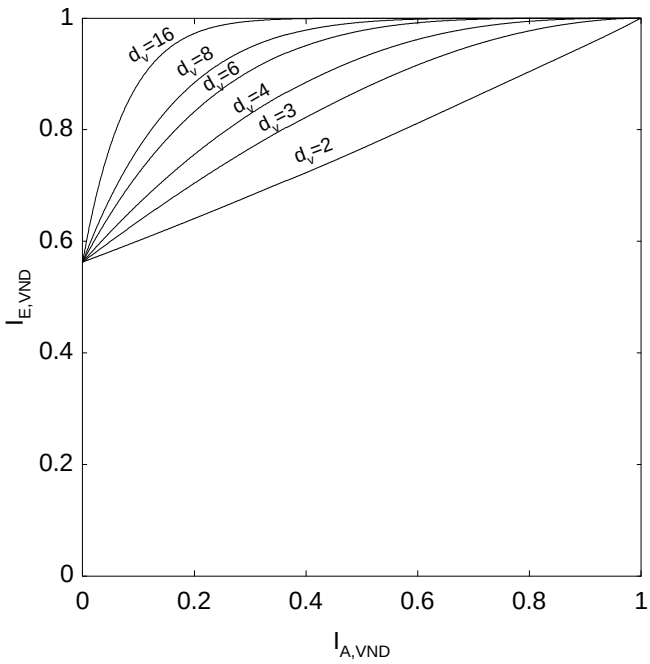


Fig. 2. VND EXIT curves for $E_b/N_0 = 1$dB and $R = 1/2$.

In the L–value notation of [13] this can also be written as a "box–plus" operation:

$$L_{i,out} = \sum_{j \neq i} \boxplus L_{j,in}. \tag{6}$$

For code design we will later assume that the $L_{j,in}$ (but not the $L_{i,out}$) are Gaussian distributed. We point out that for numerical stability it is better to compute (5) recursively in the logarithmic domain rather than directly.

The check node EXIT curves can be computed by simulation. Alternatively, for the binary erasure channel a duality property exists [7], [15] that gives the EXIT curve $I_{E,SPC}(\cdot)$ of the length d_c single parity check code in terms of the EXIT curve $I_{E,REP}(\cdot)$ of the length d_c (or rate $1/d_c$) repeat code, i.e.,

$$I_{E,SPC}(I_A, d_c) = 1 - I_{E,REP}\left(1 - I_A, d_c\right). \tag{7}$$

Simulations show that this property is in fact an accurate approximation for Gaussian *a priori* inputs $L_{j,in}$. We thus approximate the check node EXIT curves as

$$I_{E,CND}(I_A, d_c) \approx 1 - I_{E,REP}\left(1 - I_A, d_c\right) = 1 - J\left(\sqrt{d_c - 1} \cdot J^{-1}\left(1 - I_A\right)\right) \tag{8}$$

where the second step follows by using (3) with $\sigma_{ch}^2 = 0$. It is further useful to express (8) in terms of its inverse function, i.e.,

$$I_{A,CND}(I_E, d_c) \approx 1 - J\left(\frac{J^{-1}\left(1 - I_E\right)}{\sqrt{d_c - 1}}\right). \tag{9}$$

Fig. 3 plots several check node curves. Observe that the curves are similar to the VND curves of Fig. 2 except that they all start from the origin.

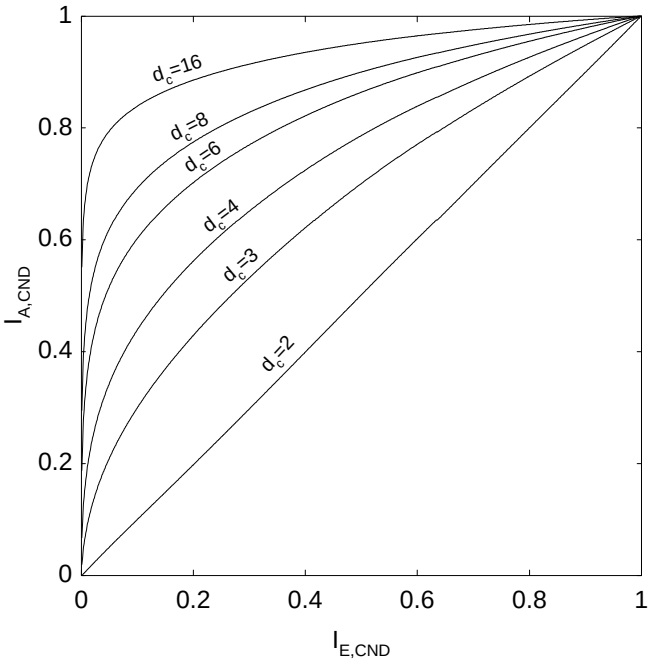


Fig. 3. CND EXIT curves. Observe that the *a priori* and extrinsic axes are swapped as compared to Fig. 2.

D. EXIT Curves for Code Mixtures

We will consider only *check–regular* LDPC codes, i.e., all check nodes have degree d_c (in [16] these are called *right–regular* codes). After choosing d_c, the remaining LDPC design involves specifying the variable node degrees $d_{v,i}$, $i = 1, \ldots, n$. Let D be the number of different variable node degrees, and denote these degrees by $\widetilde{d}_{v,i}$, $i = 1, \ldots, D$. The average variable node degree is

$$\overline{d}_v = \sum_{i=1}^{D} a_i \cdot \widetilde{d}_{v,i} \tag{10}$$

where a_i is the fraction of *nodes* having degree $\widetilde{d}_{v,i}$. Note that the a_i must satisfy $\sum_i a_i = 1$. Since the number of edges at the VND and CND are the same, we have $n\,\overline{d}_v = (n-k)\,d_c$ or

$$\overline{d}_v = (1-R) \cdot d_c. \tag{11}$$

Let b_i be the fraction of *edges* involved with variable nodes having degree $\widetilde{d}_{v,i}$. There are in total $(n\,a_i)\,\widetilde{d}_{v,i}$ edges involved with such nodes, so we have

$$b_i = \frac{n\,a_i\,\widetilde{d}_{v,i}}{n\,\overline{d}_v} = \frac{\widetilde{d}_{v,i}}{(1-R)\,d_c} \cdot a_i. \tag{12}$$

Note that the b_i must satisfy $\sum_i b_i = 1$. In [15], [17] it is shown that the EXIT curve of a mixture of codes is an average of the component EXIT curves. We must here average using the b_i (and not the a_i) because it is the *edges* that carry the extrinsic messages. The effective VND transfer curve is thus given by

$$I_{E,VND}\left(I_A, E_b/N_0, R\right) = \sum_{i=1}^{D} b_i \cdot I_{E,VND}\left(I_A, \widetilde{d}_{v,i}, E_b/N_0, R\right) \tag{13}$$

i.e., (13) is a weighted sum of the curves of Fig. 2. Note that only $D-2$ edge fractions can be adjusted because we must enforce (11) and $\sum_i b_i = 1$. Thus, in order to have any flexibility we must choose $D \geq 3$. We shall see that $D = 3$ already gives surprisingly good results.

E. Design Example

In [15] (see also [17]) it is shown that to approach capacity on binary erasure channels one must match the VND and CND transfer curves. Experience shows that the same is true for other channels. We perform the curve matching by recalling that $I_{E,VND}(0, E_b/N_0, R)$ is the channel capacity for the specified σ_{ch} (see Section II-B). We design a code with $R = 1/2$, and choose $E_b/N_0 = 0.5$dB so that $I_{E,VND}(0, 0.5\text{dB}, 1/2) \approx 0.524$ is slightly larger than R. We further choose $d_c = 8$ so that the CND transfer curve has a reasonable distance from the y–axis at $I_{A,CND} = 1/2$. This approach simplifies finding a VND curve with $I_{E,VND}(0, 0.5\text{dB}, 1/2) \approx 0.524$ that lies above the CND curve. Furthermore, this gives the decoder a good "head start" at the first iteration.

For simplicity we restrict the VND to have only three different variable node degrees ($D = 3$). This means that only one b_i can be chosen freely, and Fig. 4 shows a manual curve fit whose variable node parameters are as follows:

$$\begin{array}{lll} \widetilde{d}_{v,1} = 2, & a_1 = 0.508, & b_1 = 0.254, \\ \widetilde{d}_{v,2} = 4, & a_2 = 0.419, & b_2 = 0.419, \\ \widetilde{d}_{v,3} = 18, & a_3 = 0.073, & b_3 = 0.327. \end{array}$$

For this simple example we obtain a convergence threshold of about 0.5dB while the capacity is at 0.19dB (see also [7]). A simulation with $n = 10^5$, a random edge interleaver, and 60 iterations shows a turbo cliff at about 0.55dB (we measure the turbo cliff at a bit error rate (BER) of 10^{-4}). After 200 iterations the turbo cliff is at 0.5dB, which verifies the accuracy of the EXIT chart and our approximations.

We remark that by choosing a larger D and using sophisticated curve fits one can match the VND curve more closely to the CND curve. One can also model the *a priori* knowledge more carefully than as Gaussian random variables. However, any improvement over the approach outlined above can only be marginal in the sense of lowering the E_b/N_0 where the turbo cliff appears. We also remark that the code design is just as easy for other rates. For instance, suppose that $R = 3/4$ so that solving $I_{E,VND}(0, E_b/N_0, 0.75) = 0.75$ for E_b/N_0 gives the best possible signal-to-noise ratio. Fig. 3 suggests that $d_c = 16$ can yield a good curve fit. We would now choose the b_i to ensure that the $I_{E,VND}(\cdot)$ curve lies above but close to the $I_{A,CND}(\cdot) = I_{E,CND}^{-1}(\cdot)$ curve.

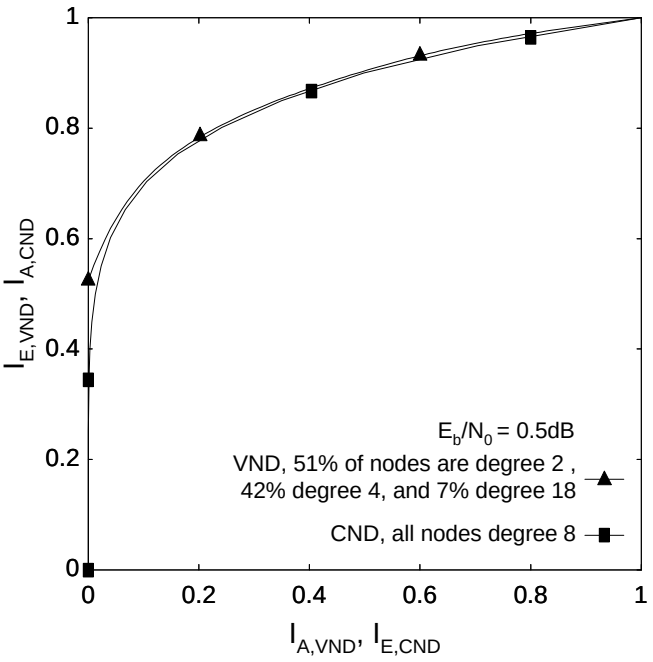


Fig. 4. Curve fit for a LDPC code with $R = 1/2$.

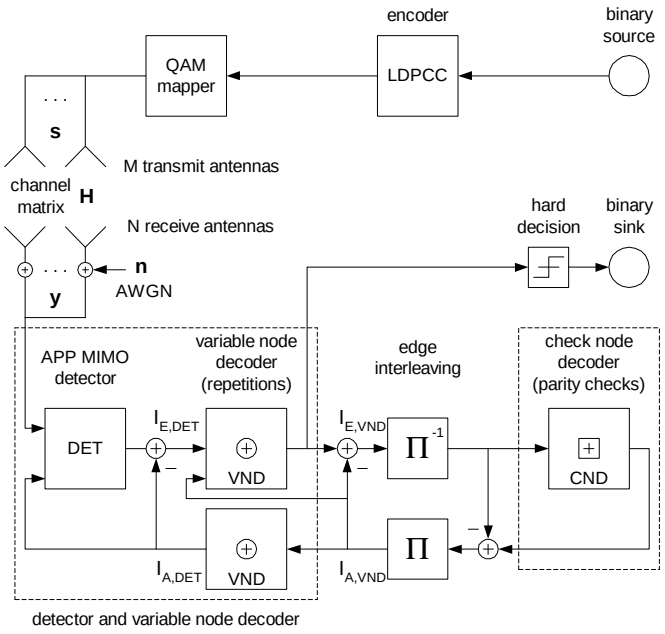


Fig. 5. The MIMO model.

III. Code Design for MIMO Channels

We next turn to multi–antenna modulation and detection. The techniques described here can be applied to many other modulators, channels, and detectors.

A. MIMO Channel Model

For MIMO fading one often distinguishes between three practical cases based on channel knowledge. The first case is that both the transmitter and receiver know the channel, the second that only the receiver knows the channel, and the third that neither terminal knows the channel [18, pp. 2627–2629]. The first case can be dealt with in a manner similar to the AWGN channel, with the addition of "water pouring" over the antennas and time [18, pp. 2627–2628]. The third case seems to be the most difficult and is not dealt with here (see [19]). We will consider primarily the second case.

Consider the setup of Fig. 5, where there are M transmit and N receive antennas. Each transmitter symbol is thus an $M \times 1$ vector $\mathbf{s} = [s_1, ..., s_M]^T$ whose entries take on complex values in a constellation set. We consider constellations of size 2^{M_c} so that each vector symbol carries $M \cdot M_c$ coded bits. For example, for quadrature phase–shift keying (QPSK) we have $M_c = 2$. The average energy per transmit symbol is limited to E_s, and we assume that $E[||s_m||^2] = E_s/M$ for $m = 1, 2, \ldots, M$.

The receiver sees $N \times 1$ vectors $\mathbf{y} = \mathbf{Hs} + \mathbf{n}$ where $\mathbf{H}$ is the $N \times M$ channel matrix and $\mathbf{n}$ is a $N \times 1$ noise vector. We assume the entries of $\mathbf{n}$ to be independent, complex, zero–mean, Gaussian random variables with independent real and imaginary parts each having variance $\sigma^2 = N_0/2$.

We define the normalized signal–to–noise ratio E_b/N_0 as

$$\left.\frac{E_b}{N_0}\right|_{\text{dB}} = \left.\frac{E_s}{N_0}\right|_{\text{dB}} + 10\log_{10}\frac{N}{RMM_c}. \tag{14}$$

The value N appears in (14) by convention; this is done simply to keep the $E_b/N_0|_{\text{dB}}$ at capacity close to each other for different N. We assume that $\mathbf{H}$ is known to the receiver only, and consider a Rayleigh fading channel so that the entries of $\mathbf{H}$ are independent, complex, zero–mean, Gaussian random variables with independent real and imaginary parts each having variance $1/2$ [20],[21, Sec. 4]. For a *quasistatic* channel the matrix $\mathbf{H}$ remains unchanged over long time intervals, while for an *ergodic* channel $\mathbf{H}$ changes for every symbol $\mathbf{s}$. We consider only the ergodic model whose capacity is (see [20], [21])

$$C = E\left[\log_2 \det\left(\mathbf{I} + \frac{E_s}{N_0}\frac{1}{M}\mathbf{H}\mathbf{H}^\dagger\right)\right] \tag{15}$$

where $\mathbf{I}$ is the identity matrix and $\mathbf{H}^\dagger$ is the complex–conjugate transpose of $\mathbf{H}$. One achieves capacity by using Gaussian distributed symbols $\mathbf{s}$. However, we here consider only Gray–mapped QPSK. The capacities for some such modulations are given in [22].

B. EXIT Curve of the MIMO Detector

We proceed by showing how to compute EXIT curves for the MIMO detector. We then describe a combined demodulation/decoding structure that is flexible enough to approach MIMO capacity. The structure automatically specifies how the encoding/modulation is done. Finally, we show how to match the resulting two EXIT curves.

The MIMO detector performs APP bit detection by considering all 2^{MM_c} possible hypotheses on $\mathbf{s}$ (see [22]–[26]).

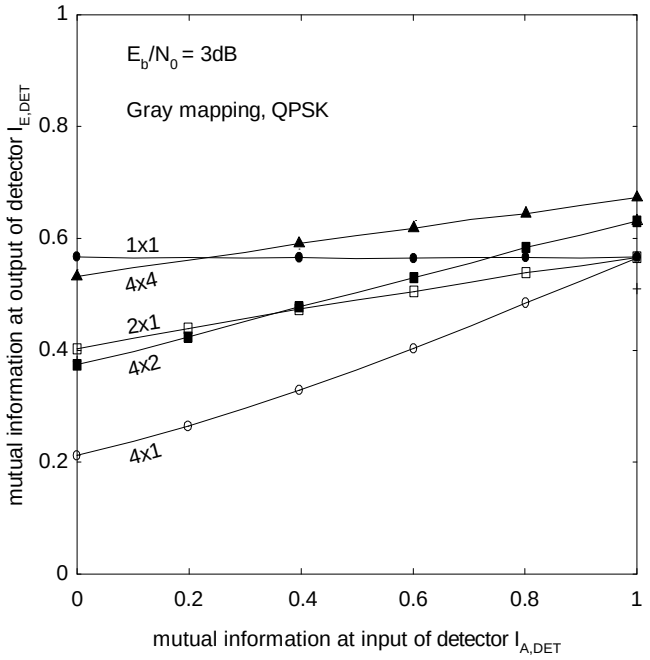


Fig. 6. MIMO detector EXIT curves at $E_b/N_0 = 3\text{dB}$ and $R = 1/2$.

The detector EXIT curve cannot be described in closed form so we measure it by Monte–Carlo simulation [27]. We denote the detector by DET and its EXIT curve by

$$I_{E,DET}\left(I_{A,DET}, E_b/N_0, R\right). \tag{16}$$

Fig. 6 shows simulated EXIT curves of MIMO detectors for different numbers of transmit and receive antennas. The modulation used is Gray–mapped QPSK. Observe that most of the curves resemble straight lines. The 1×1–curve is in fact the horizontal line

$$I_{E,DET}\left(I_{A,DET}, E_b/N_0, R\right) = E\left[J\left(\sqrt{8R\,E_b/N_0 \cdot |H|^2}\right)\right] \tag{17}$$

where H is a unit variance, complex Gaussian random variable, and where we have used (4). Note that any $1 \times N$ curve is a horizontal line. We also remark that the $M \times 1$ curves meet at $I_{A,DET} = 1$, and that $I_{E,DET}$ decreases with M for fixed $I_{A,DET}$ when $N = 1$. Similarly, one can show analytically that any $M \times N$ curve meets the $1 \times N$ curve at $I_{A,DET} = 1$, and that $I_{E,DET}$ decreases with M for fixed $I_{A,DET}$ and N.

C. EXIT Curve of the Combined MIMO Detector and Variable Node Decoder

We combine the MIMO detector and the LDPC variable node decoder as shown in Fig. 5. The detector consists of $n/(MM_c)$ individual detectors (or detector nodes) that are each connected to MM_c variable nodes. Consider the three boxes in Fig. 5 corresponding to one of these individual detector/VNDs. For simplicity, we denote the EXIT curve of this structure by $I_{E,VND}(\cdot)$ and assume that the *a priori* L–values are Gaussian distributed. We further choose all the MM_c variable nodes to have the same degree d_v.

We remark that a similar structure has been considered by others, e.g. [28], but only for regular LDPC codes. We show that the scheme can in fact approach capacity by designing LDPC degree distributions that are well matched to the MIMO detector. Note that we are designing codes for individual modulations, which is similar to *trellis coded modulation* [29]. Other coding approaches are *bit–interleaved coded modulation* [22], [30], or *space–time block coded modulation* [31], [32] (see also [33] and the references therein).

Continuing our analysis, we use (3) and find that the lower VND in Fig. 5 maps $I_{A,VND}$ into

$$I_{A,DET}(I_{A,VND}, d_v) = J\left(\sqrt{d_v} \cdot J^{-1}\left(I_{A,VND}\right)\right). \tag{18}$$

We next consider the transfer curves (16) and approximate them by third order polynomials. For example, the EXIT curve of the 4×1 MIMO detector with QPSK at $E_b/N_0 = 7\text{dB}$ and $R = 1/2$ is well–approximated as

$$\begin{aligned} I_{E,DET}\left(I_{A,DET}, 7\text{dB}, 1/2\right) = &-0.171 \cdot I_{A,DET}^3 \\ &+ 0.271 \cdot I_{A,DET}^2 + 0.393 \cdot I_{A,DET} + 0.269. \end{aligned} \tag{19}$$

The polynomial (19) lets us express the combined detector/VND EXIT curve in closed form, which is convenient for the curve fitting described below.

The third step is to consider the upper VND in Fig. 5. We approximate the EXIT curve of the combined MIMO detector and VND as

$$\begin{aligned} &I_{E,VND}\left(I_{A,VND}, I_{E,DET}, d_v\right) \\ &= J\left(\sqrt{(d_v - 1)\left[J^{-1}\left(I_{A,VND}\right)\right]^2 + \left[J^{-1}\left(I_{E,DET}\right)\right]^2}\right) \end{aligned} \tag{20}$$

where we have used (3) with $\sigma_{ch}^2 = \left[J^{-1}\left(I_{E,DET}\right)\right]^2$. Note that in (20) we use the subscript "VND" rather than, say, "VND&DET"; this is done solely to avoid unwieldy notation. Finally, inserting (16), (18), and the detector polynomial (19) into (20) we obtain the desired transfer curve in the form

$$I_{E,VND}\left(I_{A,VND}, d_v, E_b/N_0, R\right). \tag{21}$$

Fig. 7 shows some combined detector and VND transfer curves. Observe that by setting $d_v = 1$ in (21) we recover the "pure" detector transfer curve of Fig. 6.

D. Design Examples

We design check–regular LDPC codes by matching the curve (21) to the CND curve. Again, we restrict ourselves to just three different variable node degrees ($D = 3$) so that the curve fit can be performed manually. This simple approach gives surprisingly powerful coded modulations. Table I shows the parameters of our curve fitting for several MIMO channels. Fig. 8 plots the corresponding EXIT chart for a 4×1 MIMO channel.

Fig. 9 shows simulation results for LDPC codes with QPSK, $R = 1/2$, $n = 10^5$, a random edge interleaver, and

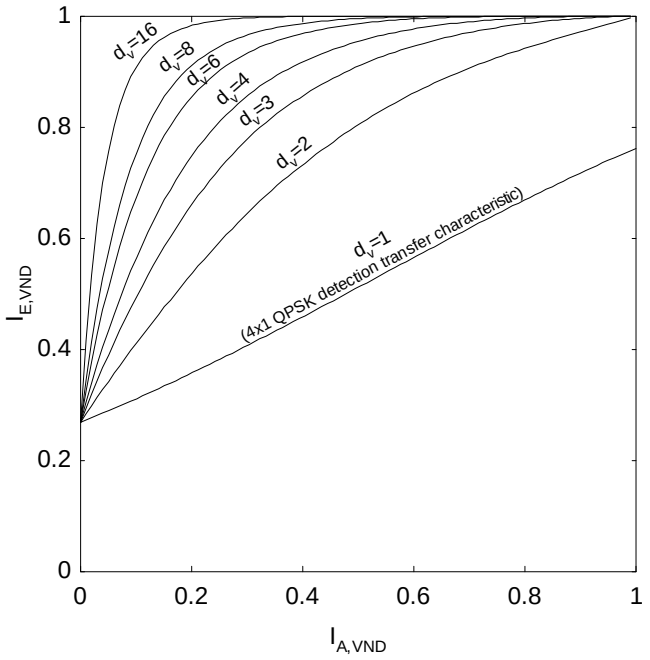


Fig. 7. Combined 4×1–detector and VND EXIT curves at $E_b/N_0 =$ 7dB and $R = 1/2$.

200 decoder iterations. All schemes operate within 1.25dB of their respective capacity limits (see Table I). We remark that the gaps to capacity in Fig. 9 (or Table I) can be narrowed further by using $D > 3$ and better curve fits.

Fig. 9 also plots the BER curves of a bit-interleaved coded modulation employing a UMTS standard turbo code, denoted PCC for "parallel concatenated code", with memory 3 constituent codes (feedback and feedforward generator polynomials $1 + D^2 + D^3$ and $1 + D + D^3$, respectively). The modulation is again QPSK and the code has $R = 1/2$, $n = 10^5$, and a random edge interleaver. The iterative processing is done by performing 20 internal decoder iterations per detector/decoder iteration, and 4 detector/decoder iterations. It turns out that more internal or detector/decoder iterations hardly improve the bit error rate.

Observe that the LDPC scheme outperforms the turbo coded scheme by wide margins for $M > N$ (the gain is over 5dB for 4×1 MIMO). The reason this happens is that the turbo code EXIT curve is almost a horizontal line, as will be the case for any strong code for the AWGN channel. The turbo code EXIT curve is therefore poorly matched to a steep detector EXIT curve [27].

E. Fading Model and Finite Modulation Sets

We discuss two practical issues. First, the ergodic channel model assumes that the transmitted codeword sees multiple independent realizations of $\mathbf{H}$, and that the receiver can accurately estimate these realizations. This occurs in practice when there is block fading (such as with time-division multiple-access, or with multicarrier modulation experiencing fast fading and employing time/frequency interleaving) and one can afford to code over many blocks.

TABLE I
PARAMETERS FOR $M \times N$ MIMO USING QPSK WITH $R = 1/2$ LDPC CODES. OVERALL RATE: M BITS/CHANNEL USE

2×1		
Capacity E_b/N_0 at 3.25dB		
Curve fit at 3.50dB		
BER 10^{-4} at 4.00dB (PCC at 5.00dB)		
$d_c = 8$		
$\widetilde{d}_{v,i}$	a_i [%]	b_i [%]
2	48.04	24.02
3	45.52	34.14
26	6.44	41.84

4×1			4×2		
Capacity E_b/N_0 at 6.65dB			Capacity E_b/N_0 at 2.95dB		
Curve fit at 7.00dB			Curve fit at 3.30dB		
BER 10^{-4} at 7.90dB (PCC at 13.8dB)			BER 10^{-4} at 3.60dB (PCC at 5.40dB)		
$d_c = 7$			$d_c = 6$		
$\widetilde{d}_{v,i}$	a_i [%]	b_i [%]	$\widetilde{d}_{v,i}$	a_i [%]	b_i [%]
2	94.15	53.80	2	69.78	46.52
16	5.48	25.05	3	26.12	26.12
200	0.37	21.15	20	4.10	27.36

4×3			4×4		
Capacity E_b/N_0 at 1.97dB			Capacity E_b/N_0 at 1.47dB		
Curve fit at 2.30dB			Curve fit at 1.80dB		
BER 10^{-4} at 2.45dB (PCC at 3.20dB)			BER 10^{-4} at 1.95dB (PCC at 2.40dB)		
$d_c = 7$			$d_c = 8$		
$\widetilde{d}_{v,i}$	a_i [%]	b_i [%]	$\widetilde{d}_{v,i}$	a_i [%]	b_i [%]
2	49.49	28.28	2	33.83	16.91
3	43.88	37.61	3	59.80	44.85
18	6.63	34.11	24	6.37	38.24

However, in some cases the delay constraints are too severe to code over many blocks, in which case *outages* cannot be avoided [21]. For such quasistatic cases one can still apply the curve–fitting procedure described here, but one now shapes the detector/VND EXIT curve to minimize outage probability. The question of how to do this precisely for given fading statistics is an interesting open problem.

Second, the capacities of both the ergodic and quasistatic models are achieved with a *Gaussian* input distribution. However, in practice one does not use such a distribution. Some reasons for this are: the transmitter and receiver amplifiers have peak energy constraints, the amplifiers are linear over limited ranges of inputs, the receiver must acquire and maintain phase synchronization, and the APP detector complexity grows exponentially with the number of bits per input symbol. These reasons make small modulation sets such as QPSK attractive for wireless communication. Of course, the capacities of such modulation sets are smaller than for Gaussian inputs. We thus compare the performance of our codes to the finite modulation set capacities, which is the fairest benchmark.

F. Comparison with Space–Time Block Codes

Space–time block codes are attractive because they improve *diversity* and are easy to encode and decode [33]. However, the diversity gain comes at the expense of re-

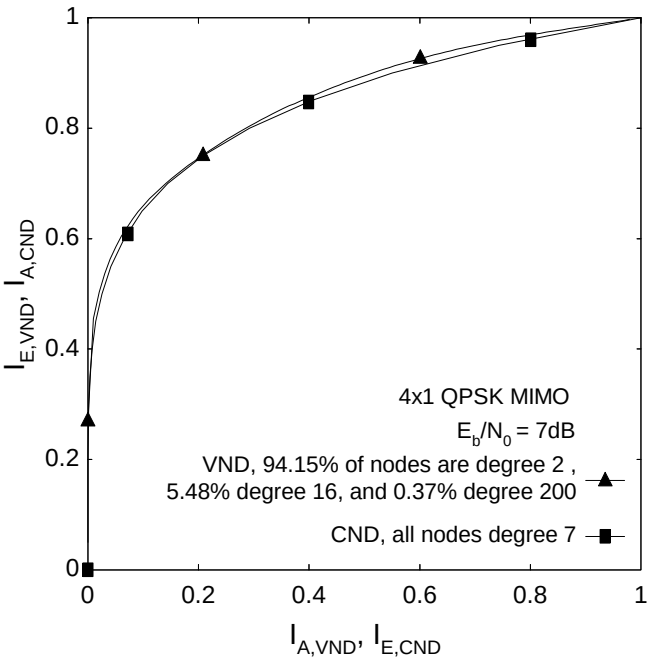


Fig. 8. Curve fit for the 4×1 scheme at $E_b/N_0 = 7$dB and $R = 1/2$.

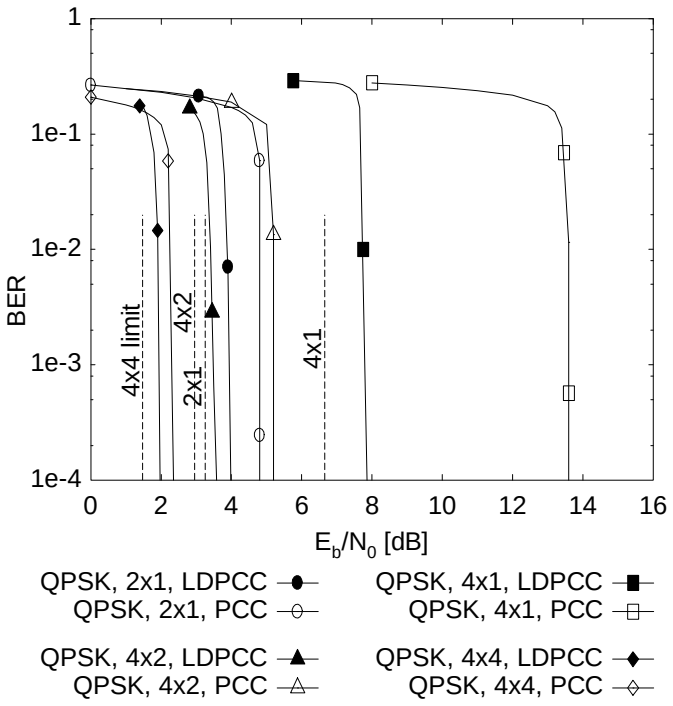


Fig. 9. Bit error rates for LDPC and turbo codes (PCC). All schemes have a total rate of M bits per channel use.

ducing spectral efficiency. For example, consider the 2×1 MIMO channel with QPSK and a rate $1/2$ LDPC code. The spectral efficiency is 2 bits per channel use. Suppose next that we use Alamouti's code [31] as an inner code. This scheme obtains a diversity gain with what is in effect a rate $1/2$ repeat code. Thus, to achieve 2 bits per channel use with a rate $1/2$ outer code one must compensate for the inner code rate loss by changing the constellation from QPSK to 16–QAM. But recall that 16–QAM complicates amplification, synchronization, and detection.

These issues will be even more pronounced for 4×1 MIMO channels. For example, the scheme proposed here achieves a spectral efficiency of 4 bits per channel use with QPSK modulation and a rate $1/2$ LDPC code. A corresponding space–time block code [32], which is in effect a rate $1/4$ repeat code, would have to use 256–QAM to achieve 4 bits per channel use with a rate $1/2$ outer code. Furthermore, no rate-$1/4$ orthogonal space–time block code exists for the 4×1 MIMO channel [32]. These considerations suggest that diversity is best achieved by designing the coding and modulation together rather than separately.

IV. Summary

We described a method for combining an irregular LDPC code with a modulator and detector. We further introduced a pragmatic method for designing LDPC degree distributions that are well matched to MIMO detectors. The design was based on EXIT charts. Simulations verified that all our codes operate within 1.25dB of their respective capacity limits at a BER of 10^{-4}.

Appendix
The $J(\cdot)$–Function

Suppose we have $Y = X + N$, where $\Pr(X = +1) = \Pr(X = -1) = 1/2$ and N is zero–mean, Gaussian noise with variance σ_n^2. We define the L–value

$$L_{ch}(y) := \log \frac{p(y|x=+1)}{p(y|x=-1)} = \frac{2}{\sigma_n^2}\, y. \tag{22}$$

Note that $L_{ch}(Y)$ conditioned on $X = \pm 1$ is Gaussian with mean $\mu_{ch} = \pm 2/\sigma_n^2$ and variance $\sigma_{ch}^2 = 4/\sigma_n^2$. We thus have

$$\mu_{ch} = \pm \sigma_{ch}^2/2.$$

Let $J(\sigma_{ch})$ be the mutual information $I(X; L_{ch}(Y))$. We compute

$$\begin{aligned} J(\sigma_{ch}) &= H(X) - H(X|L_{ch}(Y)) \\ &= 1 - \int_{-\infty}^{\infty} \frac{e^{-\frac{\left(\xi - \sigma_{ch}^2/2\right)^2}{2\sigma_{ch}^2}}}{\sqrt{2\pi\sigma_{ch}^2}} \cdot \log_2\left[1 + e^{-\xi}\right] d\xi \end{aligned} \tag{23}$$

where $H(X)$ is the entropy of X and $H(X|L_{ch}(Y))$ is the entropy of X conditioned on $L_{ch}(Y)$. Note that $I(X; L_{ch}(Y))$ is the same as $I(X; Y)$. The capacity of our $Y = X + N$ channel with the restriction $X \in \{\pm 1\}$ is therefore $J(\sigma_{ch}) = J(2/\sigma_n)$.

A. Approximating the $J(\cdot)$–function

For computer implementation we used the following approach. We split $J(\cdot)$ into two parts corresponding to the two intervals $0 \le \sigma \le \sigma^*$ and $\sigma^* < \sigma < \infty$ where $\sigma^* = 1.6363$. We used a polynomial fit for the left interval and an exponential fit for the right interval. We applied the

non–linear least–squares (NLLS) Marquardt–Levenberg algorithm (see [34]) to obtain

$$J(\sigma) \approx \begin{cases} a_{J,1}\cdot\sigma^3 + b_{J,1}\cdot\sigma^2 + c_{J,1}\cdot\sigma, & 0 \le \sigma \le \sigma^* \\ 1 - \exp\left[a_{J,2}\cdot\sigma^3 + b_{J,2}\cdot\sigma^2 + c_{J,2}\cdot\sigma + d_{J,2}\right], & \sigma^* < \sigma < 10 \\ 1, & \sigma \ge 10 \end{cases}$$

where

$$\begin{aligned} a_{J,1} &= -0.0421061, \quad b_{J,1} = 0.209252, \\ c_{J,1} &= -0.00640081 \\ a_{J,2} &= 0.00181491, \quad b_{J,2} = -0.142675, \\ c_{J,2} &= -0.0822054, \quad d_{J,2} = 0.0549608. \end{aligned}$$

B. Approximating $J^{-1}(\cdot)$

For the inverse $J(\cdot)$–function we split the curve into two intervals at $I^* = 0.3646$. Our approximation is

$$J^{-1}(I) \approx \begin{cases} a_{\sigma,1}\cdot I^2 + b_{\sigma,1}\cdot I + c_{\sigma,1}\cdot\sqrt{I}, & 0 \le I \le I^* \\ -a_{\sigma,2}\cdot\ln\left[-b_{\sigma,2}\cdot(I-1)\right] - c_{\sigma,2}\cdot I, & I^* < I < 1 \end{cases}$$

where

$$\begin{aligned} a_{\sigma,1} &= 1.09542, \quad b_{\sigma,1} = 0.214217, \quad c_{\sigma,1} = 2.33727 \\ a_{\sigma,2} &= 0.706692, \quad b_{\sigma,2} = 0.386013, \quad c_{\sigma,2} = -1.75017. \end{aligned}$$

References

[1] R. G. Gallager, "Low–Density Parity–Check Codes," Sc.D. thesis, Mass. Inst. Tech., Cambridge USA, Sept. 1960.

[2] R. G. Gallager, "Low–density parity–check codes," *IEEE Trans. Inform. Theory*, vol. 8, pp. 21–28, Jan. 1962.

[3] M. G. Luby, M. Mitzenmacher, M. A. Shokrollahi, D. A. Spielman, and V. Stemann, "Practical Loss-Resilient Codes," in *Proc. 29th Annu. ACM Symp. Theory of Computing,* 1997, pp. 150-159.

[4] M. G. Luby, M. Mitzenmacher, M. A. Shokrollahi, and D. A. Spielman, " Efficient erasure correcting codes," *IEEE Trans. Inform. Theory*, vol. 47, no. 2, pp. 569–584, Feb. 2001.

[5] T. J. Richardson and R. L. Urbanke, "The capacity of low–density parity–check codes under message–passing decoding," *IEEE Trans. Inform. Theory*, vol. 47, no. 2, pp. 599–618, Feb. 2001.

[6] T. J. Richardson, A. Shokrollahi, and R. L. Urbanke, "Design of capacity–approaching low–density parity–check codes," *IEEE Trans. Inform. Theory*, vol. 47, no. 2, pp. 619–637, Feb. 2001.

[7] S. Y. Chung, G. D. Forney, T. J. Richardson, and R. Urbanke, "On the design of low–density parity–check codes within 0.0045dB of the Shannon limit," *IEEE Comm. Lett.*, vol. 5, no. 2, pp. 58–60, Feb. 2001.

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

[9] D. Divsalar, S. Dolinar, and F. Pollara, "Low complexity turbo–like codes", *Proc. 2nd Int. Symp. on Turbo Codes*, pp. 73–80, Sept. 2000.

[10] H. El Gamal and A. R. Hammons, Jr., "Analyzing the turbo decoder using the Gaussian approximation," *IEEE Trans. Inform. Theory*, vol. 47, no. 2, pp. 671–686, Feb. 2001.

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

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

[13] J. Hagenauer, E. Offer, and L. Papke, "Iterative decoding of binary block and convolutional codes," *IEEE Trans. Inform. Theory*, vol. 42, no. 2, pp. 429–445, Mar. 1996.

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

[15] A. Ashikhmin, G. Kramer, and S. ten Brink, "Extrinsic information transfer functions: A model and two properties," *36th Ann. Conf. on Inf. Sci. and Syst.*, Princeton, Mar. 2002.

[16] M. A. Shokrollahi, "New sequences of linear time erasure codes approaching the channel capacity," *Proc. 13th Conf. Applied Algebra, Error Correcting Codes, and Cryptography (Lecture Notes in Computer Science)*, Springer Verlag, Berlin Germany, 1999, pp. 65–76.

[17] M. Tuechler and J. Hagenauer, "EXIT charts and irregular codes," *36th Ann. Conf. on Inf. Sci. and Syst.*, Princeton, Mar. 2002.

[18] E. Biglieri, J. Proakis, and S. Shamai (Shitz), "Fading channels: information-theoretic and communications aspects," *IEEE Trans. Inform. Theory*, vol. 44, no. 2, pp. 2619–2692, Oct. 1998.

[19] T. L. Marzetta and B. M. Hochwald, "Capacity of a mobile multiple-antenna communication link in Rayleigh flat-fading," IEEE Transactions on Infomation Theory, vol. 45, no. 1, pp. 139–157, January 1999.

[20] G. J. Foschini, "Layered space–time architecture for wireless communication in a fading environment when using multi–element antennas," *Bell Labs. Tech. J.*, vol. 1, no. 2, pp. 41–59, 1996.

[21] I. E. Telatar, "Capacity of multi–antenna Gaussian channels," *Eur. Trans. Telecom.*, vol. 10, pp. 585–595, Nov. 1999.

[22] B. M. Hochwald and S. ten Brink, "Achieving near-capacity on a multiple-antenna channel," *IEEE Trans. Commun.*, vol. 51, no. 3, pp. 389–399, March 2003.

[23] A. M. Tonello, "Space–time bit–interleaved coded modulation with an iterative decoding strategy," *Proc. VTC*, pp. 473–478, Sept. 2000.

[24] A. van Zelst, R. van Nee, and G. Awater, "Turbo–BLAST and its performance," *Proc. VTC*, May 2001.

[25] A. Stefanov and T. Duman, "Turbo–coded modulation for systems with transmit and receive antenna diversity over block fading channels: system model, decoding approaches, and practical considerations," *IEEE J. Sel. Areas Comm.*, vol. 19, pp. 958–968, May 2001.

[26] C. Schlegel and A. Grant, "Concatenated space–time coding," *Proc. PIMRC*, San Diego, CA, Sept. 2001.

[27] S. ten Brink and B. M. Hochwald, "Detection thresholds of iterative MIMO processing," *Proc. Int. Symp. Inform. Theory*, p. 22, July 2002.

[28] B. Lu, X. Wang, and K. Narayanan, "LDPC–based space–time coded OFDM systems over correlated fading channels: performance analysis and receiver design," *IEEE Trans. Commun.*, vol. 50, pp. 74–88, Jan. 2002.

[29] G. Ungerboeck, "Channel coding with multilevel/phase signals," *IEEE Trans. Inform. Theory*, vol. 28, no. 1, pp. 55–67, Jan. 1982.

[30] G. Caire, G. Taricco, and E. Biglieri, "Bit-interleaved coded modulation," *IEEE Trans. Inform. Theory*, vol. 44, no. 3, pp. 927–946, May 1998.

[31] S. M. Alamouti, "A simple transmit diversity technique for wireless communications," *IEEE Journ. on Select. Areas in Commun.*, vol. 16, pp. 1451–1458, Oct. 1998.

[32] V. Tarokh, H. Jafarkani and A. R. Calderbank, "Space–time block codes from orthogonal designs," *IEEE Trans. Inform. Theory*, vol. 45, pp. 1456–1467, July 1999.

[33] H. El Gamal and M. O. Damen, "Universal space-time coding," *IEEE Trans. Inform. Theory*, vol. 49, no. 5, pp. 1097–1119, May 2003.

[34] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery, *Numerical Recipes in C*, Cambridge University Press, New York, 1997.