Low-Density Parity-Check Codes and Their Rateless Relatives

Nicholas Bonello, Sheng Chen, and Lajos Hanzo

***Abstract*—This survey guides the reader through the extensive open literature that is covering the family of low-density parity-check (LDPC) codes and their rateless relatives. In doing so, we will identify the most important milestones that have occurred since their conception until the current era and elucidate the related design problems and their respective solutions.**

***Index Terms*—Low-density parity-check (LDPC) codes, rateless codes and codes-on-graph.**

I. INTRODUCTION

LOOKING back over the last six decades or so, one can reasonably surmise that the family of low-density parity-check codes (LDPC) [1] and that of turbo codes [2], constitute the two most practical realizations of Shannon's theory [3], which have revolutionized the field of error correction coding [4].

It was precisely the year 1948, when Claude E. Shannon, who at that time was a researcher at Bell Labs, published one of the most important theories, which inspired the research community for many years to come. At that time, his theories disproved the widely supported belief that increasing the amount of information-carrying bits transmitted over the channel per second, imposes an increase in the probability of error. Shannon demonstrated that it is possible to transmit information arbitrarily reliably over any unreliable channel, provided that the information transmission rate is lower than the capacity of the channel [3]. Therefore, the channel capacity sets the bound on how much information we can transmit over a channel.

Shannon's claim can be realized by a technique referred to as forward error correction. The basic idea is that of incorporating redundant bits, or check bits, thus creating what is known as a codeword. If the check bits are introduced in an "appropriate manner" so as to make each codeword sufficiently distinct from each other, the receiver will then become capable of determining the most likely codeword that has been transmitted. The channel capacity will determine the exact amount of redundancy that has to be incorporated by the encoder in order to be able to correct the errors imposed by the channel.

However, Shannon's theory only quantifies the maximum attainable rate, but refrains from specifying the means of achieving it. This triggered widespread research efforts resulting in diverse extensions, deeper interpretations and practical realizations of Shannon's original work, which reached its pinnacle in the definition of LDPC and turbo codes.

A. Fixed-rate Versus Rateless Forward Error Correction

In the majority of the research literature, fixed-rate and rateless codes are generally treated separately and hence the reader inevitably gets the impression that these channel codes are strikingly dissimilar or even perhaps unrelated. By contrast, in this survey we will be treating them jointly and thus endeavor to portray both the differences and the similarities of fixed-rate and rateless codes.

Let us commence by outlining the differences. In simplistic terms, *fixed-rate* error correction codes, also referred to as fixed-rate channel codes, incorporate a *fixed amount* of redundant bits and thus may be deemed to possess a *fixed code-rate*. For example, a half-rate channel code will be outputting twice the number of input information bits for the whole duration of our transmission. Therefore, a fixed-rate code having a rate R_x, can be carefully designed in order to attain a performance that is close to the capacity target $C(\psi_x)$ at a specific channel signal-to-noise (SNR) value of ψ_x dB, for which it was originally contrived for. However, having a fixed-rate will impose two limitations. Firstly, if the channel SNR encountered is actually higher than ψ_x dB, the fixed-rate channel code essentially becomes an inefficient channel code, albeit it exhibits a good performance at ψ_x dB, since the code incorporates more redundancy than the actual channel conditions require. Secondly, if on the other hand, the channel SNR encountered becomes lower than the SNR value of ψ_x dB, then the link is said to be in outage for the simple reason that the fixed-rate channel code under consideration is failing to supply sufficient redundancy to cope with the channel conditions encountered. The fixed-rate channel code can be modified in order to become more suitable or more efficient for employment in channels of higher or lower quality by using the so-called code puncturing [5] or code extension techniques [6]. Code puncturing involves removing some of the codeword bits and thus creating a code having a rate that is higher than the original rate R_x, whilst code extension is used to incorporate additional parity bits and thus for reducing the code-rate.

On the other hand, *rateless codes* solve this problem from a slightly different perspective. By delving into their fundamental principles and thus portraying their philosophical differences, rateless codes do not fix their code-rate, or equivalently, the amount of redundancy, before transmission. This

Manuscript received 6 May 2009; revised 25 October and 11 November 2009.

The authors are with the School of ECS, University of Southampton, SO17 1BJ, United Kingdom. (e-mail: {nb06r, sqc, lh}@ecs.soton.ac.uk), http://www-mobile.ecs.soton.ac.uk

The financial support of both the EPSRC U.K., and that of the EU under the auspices of the Optimix project is gratefully acknowledged.

Digital Object Identifier 10.1109/SURV.2011.040410.00042

is essentially the interpretation of the terminology ‘rateless’. A rateless code will progressively transmit more redundant bits, until the receiver can correctly recover the transmitted data. Therefore, their code-rate can only be determined by taking into account the total redundancy that had to be transmitted in order to achieve correct reception. Rateless codes were also intended to be employed in situations, where channel state information is unavailable at the transmitter. However, we particularly emphasize that this does not automatically imply that rateless codes do not require a feedback channel; on the contrary, it is still necessary to have a reliable low-rate feedback channel for the receiver to acknowledge the correct recovery of the data by sending its acknowledgment flag and thus to allow for the next codeword’s transmission to start. Another significant characteristic of rateless codes, which makes them eminently suitable for employment on time-varying channels is their inherent flexibility and practicality, when it comes to the calculation of the transmitted codeword.

As we briefly mentioned at the beginning of this subsection, it is equally important to appreciate the strong similarities between the two code families. In order to make our arguments conceptually appealing, we progress by saying that the analogy between rateless and fixed-rate channel codes may be viewed in the same way, as the correspondence between the continuous and the discrete representation of the *same* signal or mathematical function. A fixed-rate channel code will then correspond to a discrete signal in our simplified analogy, because provided it is designed appropriately, it is well-capable of attaining a near-capacity performance at the (single) SNR value of ψ_x dB. By contrast, a rateless code may be deemed to be constructed from an infinite number of fixed-rate channel codes (of different rates) and thus are capable of attaining a near-capacity performance over a wider range of channel SNR values. Interestingly enough, this analogy also applies to their underlying code construction. In fact, we will also see in the forthcoming sections that a good number of rateless codes’ constructions may be viewed as being instances of other fixed-rate channel code constructions.

B. Organization of this Survey

In this survey, we will only focus our attention on LDPC codes and their rateless relatives. We will guide the reader through the extensive literature, commencing from the conception of both code-families and portraying their evolution, including the current state-of-the-art. We will commence our discourse by introducing the related preliminary terminology and definitions. We will then proceed to provide further insights on the most pertinent issues related to LDPC codes, such as their code constructions, their encoding and decoding techniques, their performance metrics, the convergence of their decoding as well as the associated design techniques. Subsequently, we will also outline a range of hardware-implementation-related issues and detail a variety of current research endeavors. We will then progress further by outlining some important milestones in the history of rateless coding as well as discuss the related design problems and identify their solutions. The list of abbreviations that have been used in this survey is summarized in Table I.

TABLE I
LIST OF ABBREVIATIONS USED IN THIS SURVEY

ALT	approximate lower triangular
ARA	accumulate-repeat-accumulate
ARAA	accumulate-repeat-accumulate-accumulate
AWGN	additive white Gaussian noise
BCH	Bose-Chaudhuri-Hocquenghem
BEC	binary erasure channel
BER	bit error ratio
BF	bit-flipping
BLER	block error ratio
BP	belief propagation
BWBF	bootstrapped weighted bit-flipping
CND	check node decoder
CSS	Calderbank-Shor-Steane
EXIT	extrinsic information transfer
FG	finite geometry
GA	genetic algorithm
GF	Galois field
GLDPC	generalized low-density parity-check
HARQ	hybrid automatic repeat request
LUT	look-up table
LT	Luby transform
LDGM	low-density generator matrix code
LDPC	low-density parity-check
MIMO	multiple-input multiple-output
MWBF	modified weighted bit-flipping
QC	quasi-cyclic
RA	repeat-accumulate
RS	Reed-Solomon
SPA	sum product algorithm
UMP	universally most-powerful
VND	variable node decoder
WBF	weighted bit-flipping

II. PRELIMINARIES

In this section, we will strive to explain the basic principles and the LDPC code related terminology in a simple and concise manner. Our discourse will be limited to the following topics:

- The basic principles of linear block codes;
- Their generator and parity-check matrices as well as the associated graphical representation and
- Some important graph-theoretic properties.

Each point will be treated separately in the forthcoming subsections. Those readers who are familiar with the above-mentioned topics, might like to proceed directly to Section III. On the other hand, we would like to direct the attention of those readers, who wish to delve into further detail, to some excellent magazine papers and textbooks such as [7]–[16], amongst others.

A. Basic Principles of Linear Block Codes

LDPC codes form part of a larger family of codes, which are typically referred to as linear block codes. A code is termed a block code, if the original information bit-sequence can be segmented into fixed-length *message blocks*, hereby denoted by $\mathbf{u} = u_1, u_2, \ldots, u_K$, each having K information digits. This implies that there are 2^K possible distinct message blocks. For the sake of simplicity, we will here be giving examples for binary LDPC codes, i.e., the codes are associated with the logical symbols/bits of $(1, 0)$. The elements $(1, 0)$ are said to constitute an *alphabet* or a *finite field*, where the latter is typically referred to as Galois field (GF). Using this terminology, a GF containing q elements is denoted by GF(q) and correspondingly, the binary GF is represented as GF(2).

The LDPC encoder is then capable of transforming each input message block $\mathbf{u}$ according to a predefined set of rules into a distinct N-tuple (N-bit sequence) $\mathbf{z}$, which is typically referred to as the *codeword*. The codeword length N, where $N > K$, is then referred to as the *block-length*. Again, there are 2^K distinct legitimate codewords corresponding to the 2^K message blocks. This set of the 2^K codewords is termed as a $\mathbb{C}(N, K)$ linear *block code*. The word *linear* signifies that the modulo-2 sum of any two or more codewords in the code $\mathbb{C}(N, K)$ is another valid codeword. The number of nonzero symbols of a codeword $\mathbf{z}$ is called the *weight*, whilst the number of bit-positions in which two codewords differ is termed the *distance*. For instance, the distance between the codewords $\mathbf{z}_1 = (1101001)$ and $\mathbf{z}_2 = (0100101)$ is equal to three. Subsequently, codewords that have a low number of binary ones are referred to as *low-weight* codewords. The *minimum distance* of a linear code, hereby denoted by d_{min}, is then determined by the weight of the non-zero codeword/s in the code $\mathbb{C}(N, K)$, which has/have the minimum weight.

B. Generator and Parity-Check Matrices

The unique and distinctive nature of the codewords implies that there is a one-to-one mapping between a K-bit information sequence $\mathbf{u}$ and the corresponding N-bit codeword $\mathbf{z}$ described by the set of rules of the encoder. Clearly, if both K and N are small, then the 2^K distinct message blocks and the corresponding codewords can be stored in a look-up table (LUT). However, for large K and N, the N-entry LUT encoder will be prohibitively complex. This complexity is significantly reduced by the fact that LDPC codes are linear codes and thus the codeword $\mathbf{z}$ can be calculated by multiplying the input message sequence $\mathbf{u}$ with a $(K \times N)$-element matrix $\mathbf{G}$, which is referred to as the *generator matrix*. So, if we consider the simple example of having a four-bit input message sequence $\mathbf{u}$ and assume that the i^{th} column of $\mathbf{G}$ is given by $[1101]^{\text{T}}$, then the i^{th} bit of the codeword $\mathbf{z}$ will be equal to the modulo-2 sum of the first, second and fourth bit of $\mathbf{u}$.

We also note that $\mathbf{G}$ can also be transformed into what is referred as the *systematic matrix form*, i.e., to $\mathbf{G} = [\mathbf{I}_K \; \mathbf{A}]$, where $\mathbf{I}_K$ is a $(K \times K)$-element identity matrix and $\mathbf{A}$ has $K \times (N-K)$-elements. This transformation is carried out by using the so-called row and column operations, which include permutations of the rows (columns), multiplication of a row (column) with a nonzero scalar and the addition of a scalar multiple of one row to another. When $\mathbf{G}$ is expressed in its systematic form, the resultant N-bit codeword $\mathbf{z}$ can be divided into two parts. The first K bits of $\mathbf{z}$ constitute of the information segment $\mathbf{u}$ of the code; whilst the second segment consists of the $(N-K)$ redundant *parity-check* bits which are calculated by means of the previously described modulo-2 addition.

There is another useful matrix associated with a linear block code. This matrix is referred to as the *parity-check matrix*, which is typically denoted by $\mathbf{H}$ and contains $(N - K) \times N$ elements. If the generator matrix is in the systematic matrix form, then the $\mathbf{H}$ matrix of the code is given by $\mathbf{H} = \left[-\mathbf{A}^{\text{T}} \; \mathbf{I}_{N-K}\right]$, where $\mathbf{I}_{N-K}$ is an identity matrix of dimension $(N-K) \times (N-K)$. A characteristic of the $\mathbf{H}$ matrix of LDPC codes is that it is sparse, i.e., there are fewer ones than there are zeros. As a result, their $\mathbf{H}$ matrix is said to have a 'low-density' - hence the terminology of low-density parity-check codes. If all the rows of $\mathbf{H}$ matrix are linearly independent, then the rate of the code becomes $R = K/N = 1 - (N-K)/N$. The $\mathbf{H}$ matrix is also said to be the generator matrix of the so-called *dual code* $\mathbb{C}^{\perp}$.

TABLE II

THE CODEWORDS FOR THE CODE $\mathbb{C}(7,4)$ AND ITS DUAL CODE $\mathbb{C}^{\perp}(7,3)$, GIVEN THE GENERATOR MATRIX AND PARITY-CHECK MATRIX REPRESENTED IN (2) AND (3), RESPECTIVELY

$\mathbf{z} \in \mathbb{C}$	$\mathbf{z}^{\perp} \in \mathbb{C}^{\perp}$
0 0 0 0 0 0 0	0 0 0 0 0 0 0
0 0 0 1 0 1 1	1 1 1 1 0 0 1
0 0 1 0 1 1 0	0 1 1 1 0 1 0
0 0 1 1 1 0 1	1 0 1 0 0 1 1
0 1 0 0 1 1 1	1 1 1 0 1 0 0
0 1 0 1 1 0 0	0 0 1 1 1 0 1
0 1 1 0 0 0 1	1 0 0 1 1 1 0
0 1 1 1 0 1 0	0 1 0 0 1 1 1
1 0 0 0 1 0 1	
1 0 0 1 1 1 0	
1 0 1 0 0 1 1	
1 0 1 1 0 0 0	
1 1 0 0 0 1 0	
1 1 0 1 0 0 1	
1 1 1 0 1 0 0	
1 1 1 1 1 1 1	

We will provide a simple example in order to illustrate our discourse. Let a $(7,4)$ code be described by means of the generator matrix $\mathbf{G}$ given by

$$\mathbf{G} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 0 & 0 & 0 & 1 & 0 & 1 \\ 1 & 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 1 \end{bmatrix}. \tag{1}$$

The generator matrix seen in (1) can be converted to its standard form with the aid of the previously described row and column operations which results in

$$\mathbf{G} = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 \end{bmatrix}. \tag{2}$$

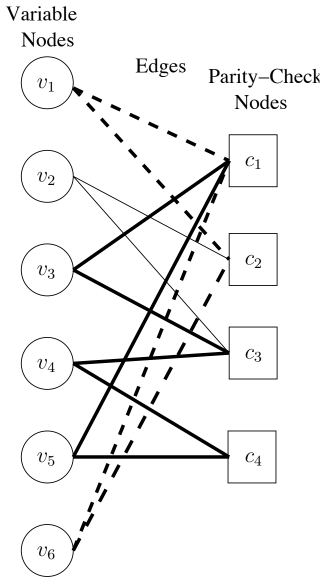
The $\mathbf{H}$ matrix is then given by

$$\mathbf{H} = \begin{bmatrix} 1 & 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 & 1 \end{bmatrix}. \tag{3}$$

The resultant codewords corresponding to the linear $(7,4)$ block codes and its dual code $\mathbb{C}^{\perp}(7,3)$ are subsequently shown in Table II, which were generated according to $\mathbf{z} = \mathbf{u}\mathbf{G}$. Observe in Table II that the first four bits of a codeword are the systematic information bits, followed by three parity bits, each of which checks the parity of the specific information bits as determined by the generator matrix represented in (2).

The $\mathbf{H}$ matrix can also be represented graphically by what is known as a *bipartite graph*, as exemplified in Figure 1. Let

$$\mathbf{H} = \begin{array}{c} \left[\begin{array}{cccccc} 1 & 0 & 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 & 0 & 1 \\ 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 \end{array}\right] \begin{array}{c} c_1 \\ c_2 \\ c_3 \\ c_4 \end{array} \\ \begin{array}{cccccc} v_1 & v_2 & v_3 & v_4 & v_5 & v_6 \end{array} \end{array}$$



(a) (b)

Fig. 1. (a) A parity-check matrix (b) The bipartite graph having girth of four and corresponding to the parity-check matrix of (a). A cycle of six (represented by the continuous bold edges) and a cycle of four (represented by dashed bold edges) are shown.

us consider as an example the LDPC code having $N = 6$, associated with the **H** matrix shown in Figure 1(a). The corresponding graph is then illustrated in Figure 1(b). It can be observed that this graph can be divided in two parts (and hence the name bipartite), whereby the right-hand side of the graph shows the so-called *parity-check nodes*, which correspond to a row of **H**, whilst the left-hand side (LHS) contains the *variable nodes*, which relate to the columns of **H**. A variable node is essentially a transmitted bit in the codeword **z**. The ones in the **H** matrix of Figure 1(a) represent the edges that interconnect the parity-check nodes and the variable nodes located on the graph of Figure 1(b). For example, one can observe from Figure 1(b) that the first parity-check node c_1 is checking the result of the modulo-2 sum (called the *parity*) of v_1, v_3, v_5 and v_6, which is also seen in the first row of the corresponding **H** matrix; i.e., if the transmitted bits represented by v_1, v_3, v_5 and v_6 are received correctly, then the value of $v_1 \oplus v_3 \oplus v_5 \oplus v_6 \oplus c_1 = 0$, where '$\oplus$' denotes the modulo-2 sum.

C. Important Graph Theoretical Properties

Let us once again focus our attention on the bipartite graph illustrated in Figure 1(b). The bipartite graph representing an LDPC code is also said to be *undirected* since its edges do not posses any sense of direction. Following this, the term *chain* is used to refer to the series of successive edges that form a continuous curve passing from one node to another located on an undirected graph. A *cycle* in a graph refers to a particular chain of nodes forming a closed loop, where the initial and final node are the same and no edge is used more than once. The number of edges in a cycle is then called the *length* of the cycle and the shortest cycle-length of the graph corresponds to what is referred to as the *girth*. The girth in a bipartite graph is always even and its smallest value is four. The graph depicted in Figure 1(b) has a girth of four and the corresponding cycle of four is shown by the dashed bold edges. A cycle of six is also shown marked by the continuous bold edges. An LDPC code is also said to be *regular*, if it is associated with an **H** matrix having a fixed row and column weight. Let the row and column weight of the **H** matrix be denoted by ρ and γ, respectively. Subsequently, an LDPC code is said to be regular if every parity-check node contained in its underlying graph is connected to ρ variable nodes, whilst every variable node is connected to γ parity-check nodes. If this is not the case, the code (and its associated graph) are termed to be irregular. For example, the graph shown in Figure 1(b) can be described as being left-regular, since all the variable nodes located in the graph have the same degree.

III. IMPORTANT MILESTONES IN THE HISTORY OF LOW-DENSITY PARITY-CHECK CODES

Following this rudimentary introduction to the related terminology, we will now proceed with a glimpse of history. LDPC codes were conceived by Gallager in his doctoral dissertation in 1962 [1], [17]. However, having limited computing resources prevented him from proving the near-capacity operation of these codes and from finding rigorous performance bounds of the decoding algorithm. In addition to this, the introduction of Reed-Solomon (RS) codes a few years earlier [18], and the widely accepted belief that concatenated RS and convolutional codes [19] were perfectly suited for practical error-control coding resulted in Gallager's work becoming neglected by researchers for approximately 30 years. Exceptions to this which are worth mentioning are the work of Zyablov, Pinsker and Margulis from the Russian school [20]–[22] and by Tanner [23]. Margulis proposed a structured regular construction for a half-rate Gallager code based on the Cayley graph, which is nowadays known as the 'Margulis' code [22]. The algebraic construction rules for LDPC codes given by Margulis were still found to be valid and applicable by Rosenthal and Vontobel [24] 20 years later, who proposed a similar code known as the 'Ramanujan-Margulis' code. Later, MacKay and Postol [25] discovered the existence of near-codewords in the Margulis codes and the presence of low-weight codewords in Ramanujan-Margulis codes.

Tanner [23] was first to propose the previously described graphical representation of LDPC codes using bipartite graphs. Tanner also introduced the min-sum as well as the sum-product decoding algorithms and demonstrated their convergence on cycle-free graphs. It was Wiberg [26]–[28] who first referred to these graphs as 'Tanner graphs' and extended them to include trellis codes. Forney [29] called these graphs Tanner - Wiberg - Loeliger graphs. Another contribution related to that of Tanner [23] was later made by Kschischang *et al.* [30], when they introduced the so-called factor graphs. The natural association of factor graphs with the sum-product algorithm (SPA) was also discussed. The forward/backward algorithm [29], the Viterbi algorithm and the Kalman filter were also considered as instances of the SPA. The work of [30] can also be considered as an alternative approach to that taken by Ali and McEliece [31], in which they view various algorithms as generalized message passing algorithms[1] and grouped them under the term of 'generalized distributive law'. Forney [32] later extended the concept of factor graphs to normal graphs.

The excellent performance of turbo codes reported during the mid-1990s [2], [33], [34] demonstrated the benefits of using low-complexity constituent codes and iterative decoding, but since they were patented, this rekindled the community's interest in LDPC codes [35]. Sipser and Spielman [36], [37] analyzed LDPC codes in terms of various code-construction expansions and introduced a sub-class of LDPC codes based on the so-called expander graphs which were appropriately referred to as 'expander codes' and decoded them with the aid of what is known as Gallager's 'Algorithm A', devised by Gallager [1], [17]. An encoder for these expander graphs was designed in [38].

The advantages offered by linear block codes having low-density parity-check matrices were rediscovered by MacKay and Neal, who proposed the MacKay-Neal [39] codes and showed that pseudo-randomly constructed LDPC codes can perform within about 1.2 dB of the theoretical upper bound of the Shannon limit [40]–[42]. Alon and Luby [43] made the first attempt to design an LDPC code capable of correcting erasures. A more practical algorithm based on cascaded random bipartite graphs was then devised in [44]. It is important to note that up to this point in time the understanding of LDPC codes was mostly limited to the regular codes. The understanding of both regular and irregular graphs was further deepened in [45]–[47] and it was demonstrated that the performance of the latter may be superior to that exhibited by the former. In [48], Luby *et al.* devised a new probabilistic tool, which significantly simplified the analysis of the probabilistic decoding algorithm proposed by Gallager [1], [17]. Richardson *et al.* further improved the results of [47] by using a technique referred to as density evolution [49] for analysing the behavior of irregular LDPC codes. Discrete density evolution was used by Chung *et al.* [50] in order to simulate a half-rate code having a block length of 10^7 exhibiting a performance within 0.04 dB of the Shannon limit at a bit error ratio (BER) of 10^{-6}.

A. Code Constructions

Broadly speaking, the parity-check matrix associated with an LDPC code can be constructed in either an unstructured [1], [42] or in a structured manner [51]. Table III lists some noteworthy examples of both structured as well as of unstructured LDPC code classifications. We note that the construction of LDPC codes has been a highly active research area in the last decade or so, and therefore Table III represents only a small fraction of the body of attractive designs available in the open literature. We also note that this classification of structured and unstructured constructions is in itself very broad.

A specific class of unstructured constructions is constituted by the pseudo-random constructions, which are typically distinguished by what is called an ensemble [1]. This defines the group of pseudo-random constructions that are governed by the same constraints. Typical constraints can be the block length N, the row and column weights of the $\mathbf{H}$ matrix, the girth etc. Richardson and Urbanke [52] showed that any constituent code can be used to approximate the average performance of the entire ensemble. There is also another class of unstructured LDPC codes, where the codes are constructed by means of a search algorithm, typically attempting to increase the girth of the underlying Tanner graph. These techniques were for example proposed by Mao and Banihashemi [53], Campello *et al.* [54], [55], Hu *et al.* [56], [57] as well as by Asamov and Aydin [58]. The underlying philosophy of these constructions is generally based upon the intuition that the presence of short cycles (i.e., having a graph with a low girth) severely violates the independence assumption between the messages exchanged between the left and right vertices of the graph, potentially propagating errors at a faster rate than they can be corrected.

The unstructured LDPC codes do not impose any implementation-related constraints on their corresponding parity-check matrix or underlying graph and they typically exhibit a performance that is close to the best achievable error correction performance [42], [50]. Hence, these unstructured constructions are often considered to be the baseline benchmarkers in BER or block error ratio (BLER) performance assessments. As we will outline in more detail in the forthcoming subsections, the excellent error-correction capabilities of unstructured LDPC codes are however achieved at the expense of a relatively high encoding and decoding complexity. Therefore, structured (sometimes referred to as deterministic) constructions may be regarded as attractive design alternatives, especially when considering their increased flexibility and adaptability, their lower cost and simpler implementation as well as their reduced encoding/decoding latency. Various structured constructions have been investigated in the literature, such as for example those using geometric approaches [60] or combinatorial designs [74]. The latter family includes the classes referred to as balanced incomplete block designs [70], such as the Steiner and Kirkman triple systems [62], [64], Bose designs [68], mutually orthogonal Latin rectangles [65] and the so-called anti-Pasch techniques [66].

In our preliminary section, we have limited our discourse to binary LDPC codes, for the sake of simplifying our analysis in the hope of capturing a wider audience. However, LDPC codes may also be classified as binary or nonbinary. Nonbinary LDPC codes were first proposed and investigated by Davey and Mackay [75], who demonstrated that nonbinary LDPC codes constructed over higher-order Galois fields may achieve a superior performance in comparison to binary codes for transmission over binary symmetric channels and binary Gaussian channels. The achievable performance improvement may be attributed to two main factors; the reduced probability of forming short cycles when compared to their binary counterparts and to the increased number of nonbinary check and variable nodes, which ultimately improves the achievable decoding performance. However, nonbinary LDPC codes suffer from the disadvantage of having an increased number of values, which potentially renders the classification of symbols more complex and hence naturally increases the

---

[1] In this context, it is worth mentioning that LDPC decoding algorithms are referred by a number of names, the most common being the SPA, the message passing algorithm and the belief propagation algorithm.

TABLE III
CLASSIFICATION OF THE LDPC CODES' CONSTRUCTIONS TOGETHER WITH SOME OF THEIR EXEMPLARS

Construction	Construction Example
Structured	Designs based on finite geometries [59], [60]
	Balanced incomplete block designs [61]–[70]
	Geometry-based designs [71], [72]
	Turbo-structured designs [73]
Unstructured	Gallager's construction [17]
	MacKay's ensembles [42]
	Lin and Costello's technique for random construction [10]
	Bit-filling and extended bit-filling [54], [55]
	Progressive edge growth [56], [57]
	Successive edge growth [58]

decoding complexity imposed. Nonbinary codes have been applied for transmission over nondispersive Rayleigh fading channels [76], over frequency selective channels [77] and multiple-input multiple-output (MIMO) channels [78]–[81]. The results in [75] were also substantiated by Hu *et al.* [57], who proposed a construction for irregular nonbinary LDPC codes defined over GF(q) constructed using the so-called progressive edge growth algorithm. It was also demonstrated that the performance of these codes improves upon increasing the Galois field size.

Lentmaier *et al.* [82] as well as Boutros *et al.* [83] proposed a more generalized version of the classic LDPC codes of Gallager [1], [17], which were referred to as generalized low-density codes (sometimes also known as generalized LDPC (GLDPC) codes). Instead of having each check node corresponding to a single-parity check equation as in the conventional LDPC codes proposed by Gallager [1], [17], the check nodes of GLDPC codes are associated with more powerful codes such as Hamming codes,[2] Bose-Chaudhuri-Hocquenghem (BCH) codes [84], [85] and RS codes [86]. GLDPC codes have been investigated, for instance in [87]–[92]. Irregular GLDPC codes have also been proposed by Liva *et al.* [93].[3] Recently, Wang *et al.* [95] proposed the doubly-GLDPC, which represent a wider class of codes than those GLDPC codes proposed in [82], [83], where linear block codes can be used as component codes for both the check and variable nodes. The investigation of doubly-GLDPC codes for transmission over the binary erasure channel (BEC) was carried out by Paolini *et al.* [96]. Further developments on GLDPC and doubly-GLDPC codes were provided recently in [97], [98].

B. Encoding of Low-Density Parity-Check Codes

The encoding operation requires the calculation of the generator matrix $\mathbf{G}$ from the parity-check matrix $\mathbf{H}$ by Gaussian elimination, whose complexity is a cubic function of the number of rows of the $\mathbf{H}$ matrix. The message block $\mathbf{u}$ is then multiplied by $\mathbf{G}$ in order to calculate the transmitted codeword. The complexity of this process is a quadratic function of the number of rows of the parity-check matrix $\mathbf{H}$. This may be viewed as a disadvantage of LDPC codes, when compared to turbo codes, considering that the latter have a lower encoding complexity.

Several authors have proposed complexity reduction measures in order to address this issue. For example, Luby *et al.* [99], [100] investigated the performance of cascaded graphs instead of bipartite graphs for transmission over the BEC. Careful selection of the number of cascaded graph stages as well as of the size of each stage may result in codes, which are encodable (and decodable) at a complexity that is a linear function of the block length. Likewise, Spielman [36], [37] promoted the employment of another concatenated scheme employing expander codes. However, in both cases, the performance exhibited by the resultant codes based on cascaded graphs appeared to be inferior to that of standard LDPC codes,[4] since the block length of each stage of the cascaded code is lower than that of the overall length of the standard LDPC code. MacKay *et al.* [101] suggested that the parity-check matrix must be constrained to be in an approximate lower triangular (ALT) form depicted in Figure 2 which guarantees a linear increase of the encoding complexity. Richardson and Urbanke [52] proved that in general, the encoding complexity increases nearly linear with the block length, being quadratic only in a small term $\underline{g}^2$, where $\underline{g}$ is referred to as the gap [102], which is a measure of the 'distance' [102] between the parity-check matrix and the lower triangular matrix as shown in Figure 2. For example, a regular LDPC code associated with an $\mathbf{H}$ matrix having a column weight of $\gamma = 3$ and row weight of $\rho = 6$ has a gap of $\underline{g} = 0.017$. There are many LDPC code families with the gap of $\underline{g} = 0$. For a more detailed discussion on the topic, we would like to refer the interested reader to Section 4 of [102].

Haley *et al.* [103] described a method, which performs LDPC encoding using an iterative matrix inversion technique.

---

[2] Hamming codes are considered to be a very efficient class of short codes having a minimum distance equal to 3. The resultant GLDPC codes constituted from Hamming component codes, are characterized by a relatively high minimum distance. This conjecture was verified in [83].

[3] Liva *et al.* [93], [94] also refer to these codes as doped LDPC codes due to the presence of more powerful (doped) nodes created by replacing any node by a linear block code.

[4] By 'standard' code, we are referring to those codes that can only be encoded by using the conventional encoded method [1], [17].

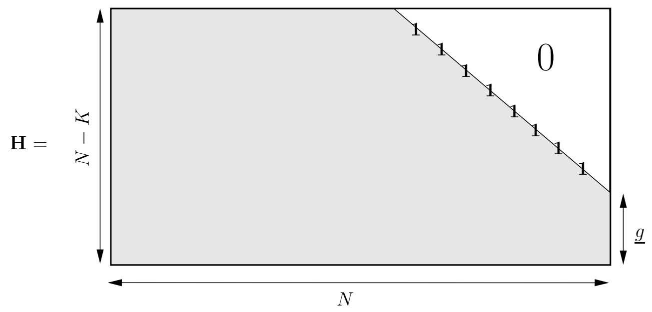


Fig. 2. A pictorial representation of a parity-check matrix in the approximate lower triangular (ALT) form. The parameter g denotes the so-called gap [102], which is a measure of the 'distance' [102] between the parity-check matrix and the lower triangular matrix.

It was shown in [103] that if the matrix satisfies certain conditions, then the proposed iterative encoding algorithm will converge after a finite number of iterations and more importantly, the resultant codes exhibits no performance loss when compared to the corresponding classic LDPC codes. This was only verified for regular LPDC codes. In [57], Hu *et al.* constructed parity-check matrices having a lower triangular form using the aforementioned progressive edge growth algorithm, and thus creating code that have a linear block-length dependent complexity. Burshtein *et al.* [104] proposed the ALT-LDPC code ensemble, which has an inherent tradeoff between the gap size (and hence the encoding complexity) as well as the achievable performance for any given block length.

Another class of codes, which attracted the attention of many researchers due to having linearly increasing block-length-dependent encoding complexity is that of the repeat accumulate (RA) codes, first proposed Divsalar *et al.* [105], which encompass the attractive characteristics of both LDPC codes and serial turbo codes. In the RA encoder, the source message is repeated a given d_v-number of times and then passed through an interleaver. The parameter d_v would then correspond to what is known as the variable node degree. The interleaved bits are then grouped into groups of d_c bits, where d_c denotes the so-called check node degree, and the modulo-2 sum of each group is then calculated. The resultant bits, corresponding to the modulo-2 sum of each group of interleaved and repeated source bits, are then passed through a rate-1 encoder, which is also referred to as an accumulator (or a recursive systematic convolutional code). Jin *et al.* [106] also extended the concept of RA codes to the family of irregular repeat-accumulate codes, where the bits of the information block are repeated in an irregular manner and where the interleaved bits are grouped into sets of different sizes. In [107], Roumy *et al.* demonstrated that these codes exhibit a near-capacity performance and have a linearly block-length-dependent encoding complexity. Abbasfar *et al.* [108] have also proposed the further enhanced accumulate-repeat-accumulate (ARA) which may be considered to be a pre-coded RA code. Divsalar *et al.* [109] extended these concepts to accumulate-repeat-accumulate-accumulate (ARAA) codes, which are basically punctured ARA codes concatenated with another accumulator. Both ARA and ARAA codes enjoy the benefits of having low-complexity encoding due to the sparse matrix multiplication based encoder and fast decoding due to their appropriately structured graph construction.

The class of algebraically constructed codes [110] may also be encoded at a complexity, which increases linearly as a function of the block length, which is a benefit of the *cyclic* or *quasi-cyclic* (QC) nature of their parity-check matrix. A QC code is defined as that code in which any cyclic shift of a constituent codeword by x number of bits is also a codeword. For a cyclic code, we have $x = 1$. For instance, each row of the parity-check matrix of a cyclic code, such as the LDPC codes based on balanced incomplete block designs [68], [111], [112], is constituted by a cyclic shift of the previous row and the first row is the cyclic shift of the last row. We also define a *circulant matrix* as a square matrix, where each row is constructed from a single right cyclic shift of the previous row, and the first row is obtained by a single right cyclic shift of the last row [14]. A QC code, such as those proposed in [113]–[118] has a parity-check matrix, which is constituted from circulant sub-matrices. For example, Figure 3 shows the $\mathbf{H}$ matrix of a quarter-rate QC LDPC code constituted from circulant matrices of size 5. For a cyclic or a QC code, the generator matrix is also cyclic/QC and hence only the first row of the each circulant will be stored, while successive rows can be generated by a shift register generator. The encoding of QC codes was detailed by Li *et al.* [119]–[121]. Another class of algebraically constructed, cyclic or QC codes is constituted by the family of finite geometry (FG)-based LDPC codes, which were rediscovered by Kou [60]. The parity-check matrix of FG-LDPC codes does have some redundant checks (similar to MacKay's constructions [42]) and the row as well as the column weights tend to be higher than those of other LDPC codes. This implies that although FG-LDPC codes benefit from the same linearly block-length-dependent encoding complexity of cyclic or QC codes, they achieve their relatively high performance at the price of a higher decoding complexity owing to their increased logic depth.

C. BER/BLER Performance Metrics

The performance of any channel code is typically assessed by means of plots of the BER/BLER versus the channel's SNR or versus the ratio of the energy-per-bit to the noise power spectral density, commonly denoted by E_b/N_0. The overall BER/BLER versus SNR performance of an LDPC code is generally described by two different regions and a threshold.

The first region is commonly referred to as the 'waterfall' or the 'turbo-cliff' region, which corresponds to the low-to-medium SNR region of the BER/BLER versus SNR plot. By contrast, the error floor is located at the bottom of the 'waterfall'-shaped curve, where it can be observed that the BER/BLER no longer exhibits the rapid improvement as in the 'waterfall' region. More often than not, the error floor is not explicitly visible in the corresponding BER/BLER plot, since it is below the BERs readily generated by the simulation performed. There is also the parlance of 'turbo-cliff' SNR or the convergence SNR threshold, above which the BER/BLER performance improves rapidly upon increasing the SNR. The word 'cliff' is again another figure of speech used

$$\begin{array}{ccccc|ccccc|ccccc|ccccc}
\hline
1&0&0&0&0&1&0&0&0&0&1&0&0&0&0&1&0&0&0&0\\
0&1&0&0&0&0&1&0&0&0&0&1&0&0&0&0&1&0&0&0\\
0&0&1&0&0&0&0&1&0&0&0&0&1&0&0&0&0&1&0&0\\
0&0&0&1&0&0&0&0&1&0&0&0&0&1&0&0&0&0&1&0\\
0&0&0&0&1&0&0&0&0&1&0&0&0&0&1&0&0&0&0&1\\
\hline
1&0&0&0&0&0&1&0&0&0&0&0&1&0&0&0&0&0&1&0\\
0&1&0&0&0&0&0&1&0&0&0&0&0&1&0&0&0&0&0&1\\
0&0&1&0&0&0&0&0&1&0&0&0&0&0&1&1&0&0&0&0\\
0&0&0&1&0&0&0&0&0&1&1&0&0&0&0&0&1&0&0&0\\
0&0&0&0&1&1&0&0&0&0&0&1&0&0&0&0&0&1&0&0\\
\hline
1&0&0&0&0&0&0&1&0&0&0&0&0&0&1&0&1&0&0&0\\
0&1&0&0&0&0&0&0&1&0&1&0&0&0&0&0&0&1&0&0\\
0&0&1&0&0&0&0&0&0&1&0&1&0&0&0&0&0&0&1&0\\
0&0&0&1&0&1&0&0&0&0&0&0&1&0&0&0&0&0&0&1\\
0&0&0&0&1&0&1&0&0&0&0&0&0&1&0&1&0&0&0&0\\
\hline
\end{array}$$

Fig. 3. The parity-check matrix of a quarter-rate LDPC code constituted from circulant matrices of size 5.

to signify that the SNR threshold occurs at that point where the 'waterfall'-shaped BER/BLER curve exhibits a rapid drop.

The SNR threshold phenomenon was first observed by Gallager [1], [17], when using regular graph constructions and by Luby *et al.* [46] for randomly constructed irregular graphs. Richardson and Urbanke [52] generalized these observations and argued that LDPC codes will exhibit a decoding threshold phenomenon, regardless of the channels encountered and the iterative decoders considered.[5] An arbitrarily small BER/BLER can be achieved with the aid of a high-girth LDPC code provided that the noise level is lower than this SNR threshold, as the block length tends to infinity. This SNR threshold can be determined using either the density evolution technique [49], [50] or by minimizing the area of the open extrinsic information transfer (EXIT) tunnel between the check node decoder (CND) and variable node decoder (VND) EXIT chart curves.[6] Both techniques assume an infinite block length, a high-girth and an infinite number of decoder iterations.

The achievable BER/BLER performance in the 'waterfall' region is predetermined by the girth. As we have briefly described in Section II-C, short cycles prevent the decoder from gleaning independent parity-check information. Therefore, the higher the girth, the faster the iteration-aided BER/BLER improvement. This is in fact the reason why we find quite a number of LDPC constructions [51], [55], [58], [60], [62], [71]–[73], [118], [122], [123], which attempt to maximize the girth[7] of the bipartite graph. One of the most attractive example is the aforementioned progressive edge growth algorithm proposed by Hu *et al.* [56], [57], [124] since they have excellent error correction capabilities, especially for codes having short block lengths.

The performance in the error floor region depends on three main factors, namely (a) on $d_{\min}$ as well as the presence of particular graphical structures in the underlying graph, which are referred to as (b) stopping sets and (c) trapping sets.[8] We will continue our discourse by discussing each of these factors in more detail.

Classic coding theory has always placed strong emphasis on trying to design codes that have a large $d_{\min}$, which is clearly justified when one recalls the fact that a code can correct up to $\lfloor (d_{\min} - 1)/2 \rfloor$ errors using a bounded distance decoder, where $\lfloor x \rfloor$ denotes the floor function, returning the largest integer less than or equal to x. Tanner [23] derived the lower bounds on the achievable $d_{\min}$ of an LDPC code and demonstrated that this increases with both the parity-check matrix column weight as well as with the girth of the underlying graph. According to these bounds, a regular LDPC code having a girth of 10 and with a $\gamma = 3$ will attain a $d_{\min} \geq 10$, whilst that code having the same girth but with a $\gamma = 4$ will attain a $d_{\min} \geq 17$. Moreover, a regular LDPC code having the same $\gamma = 4$ but with a higher girth of 12 will achieve a $d_{\min} \geq 26$. However, the relationship between these parameters is quite intricate, since whilst increasing the girth or the column weight of the associated parity-check matrix improves the minimum distance, an increase in the column weight will degrade the girth. Hence, if we consider two LDPC codes having the same rate but different column weights, the code having the highest column weight will exhibit a lower error floor owing to its higher $d_{\min}$, but a worse BER/BLER in the 'waterfall' region due to its lower girth.

A code having a small $d_{\min}$ is characterized by the presence of low-weight codewords. These will cause the so-called undetected errors, which occur when the decoding process will find a valid codeword that satisfies all the parity-check nodes, but it is not the originally transmitted codeword. However, given the fact that $d_{\min}$ of most LDPC codes increases linearly with N, undetected errors are relatively uncommon,[9] unless

---

[5] The observation was generalized to include a wide range of binary-input channels, including the binary erasure, the binary symmetric, the Laplace as well as the additive white Gaussian noise (AWGN) channels, when employing various message passing decoding algorithms [52].

[6] The EXIT chart will be explained in more detail in Section III-E.

[7] These techniques are collectively referred to by the term *girth conditioning*.

[8] We remark that failures on the erasure channel are characterized by means of stopping sets, whilst trapping sets play an analogous role for AWGN and binary symmetric channels.

[9] This is in contrast with turbo codes, which do not possess a large $d_{\min}$ and therefore their error floor is largely contributed by the low-weight codewords [4].

the block-length is short (less than a few hundred bits) or the code-rate is high. Nonetheless, it is was shown in [125] that it is computationally complex to directly design codes having a high $d_{\min}$.

An indirect way of increasing $d_{\min}$ is to increase the girth of the bipartite graph. However rather than using the conventional girth conditioning techniques, which only focus on increasing the shortest cycle length, Tian *et al.* [125] revealed that it is also important to consider the specific connectivity of the cycles with the other parts of the bipartite graph, rather than only the length of the cycles. This is because not all cycles are equally harmful - those which are well-connected to the rest of the graph are acceptable, whilst poorly connected long cycles may be more detrimental. This technique, which is commonly referred to as cycle conditioning - as opposed to girth conditioning - requires the identification of the so-called stopping sets,[10] which are a particular group of variable nodes that is connected to a group of neighboring parity-check nodes more than once. One example of a stopping set exemplified in Figure 1(b) is constituted by the variable nodes v_2, v_3 and v_6, because all the neighboring parity-check nodes c_1, c_2 and c_3 is connected to this variable node set twice. If the underlying graph does not contain any degree-one variable nodes, then it will not be possible to locate any cycle-free stopping set in that graph. Furthermore, most stopping sets are constituted by multiple cycles, unless the variable nodes in the stopping set have a degree of 2. This can also be verified from the previously mentioned stopping-set example containing v_2, v_3 and v_6 in the graph of Figure 1(b), which only contains one cycle of six. By means of avoiding small stopping sets, the technique of Tian *et al.* [125] succeeded in significantly reducing the error floor of irregular LDPC codes, whilst only suffering from a slight BER degradation in the waterfall region.

The so-called trapping sets also have a direct influence on the error floor of LDPC codes. A trapping set (a, b) refers to that particular set of a variable nodes in the associated bipartite graph which are connected to b odd-degree and an arbitrary number of even-degree parity-check nodes. For example, a trapping set $(5, 2)$ can be observed in the bipartite graph of Figure 1(b) constituted by the variable nodes v_1, v_2, v_3, v_4 and v_6 and the parity-check nodes c_2 and c_3. When the values of a and b are relatively small, the variable nodes in the trapping set are not well-connected to the rest of the graph and therefore the corresponding bits have a weak protection. In some research literature [25], [127], trapping sets are described as *near-codewords*, because when the parameters a and b are relatively small, an incorrectly decoded codeword may only be slightly different from that transmitted. We emphasize that the errors resulting from the presence of small trapping sets as well as small stopping sets are *detected* by the decoder; i.e., the decoder will be aware that the no legitimate codeword was found owing to having some unsatisfied (nonzero-valued) parity-check nodes after the affordable maximum number of decoding iterations. The problems that arise from the presence of trapping sets/near-codewords can be mitigated by either altering the parity-check matrix [128] (without changing the actual code) or by modifying the decoder [129], [130].

Carefully designed irregular LDPC codes can attain a lower ‘turbo-cliff’ SNR than regular codes of the same rate; i.e., their exhibited BER/BLER starts to rapidly decrease at a lower SNR value and hence their BER/BLER performance is superior in the ‘waterfall’ region. The reason for this phenomenon lies in the conflicting (ideal) requirements of the variable and parity-check nodes, whereby the variable nodes benefit from having large degrees, which strongly protects them. By contrast, a parity-check node should have a low degree to prevent error propagation, when it is corrupted. In this regard, irregular codes are well-capable to compromise between these seemingly competing variable and parity-check node requirements. We note however that the superior BER/BLER performance of irregular LDPC codes is achieved at the expense of a potentially increased implementational complexity.

Previously, we have emphasized that irregular LDPC codes must be ‘carefully designed’ for two main reasons. Firstly, the design of irregular codes necessitates the use of sophisticated techniques such as the aforementioned density-evolution or else EXIT charts, both of which can predict the value of the ‘turbo-cliff’ SNR. Both density-evolution and EXIT charts can also provide the actual (nonuniform) distributions for the row and column weights of the irregular parity-check matrix. Secondly, the BER/BLER performance exhibited by irregular LDPC codes is inferior to that exhibited by regular LDPC codes in the error floor region, unless we employ the previously outlined techniques, which attempt to reduce the error floor. In fact, the achievable BER performance of relatively unconditioned irregular LDPC codes will show an error floor at slightly below 10^{-6}, which is higher than that exhibited by their regular counterparts.

For the case of irregular LDPC codes, especially for those having a high proportion of degree-2 and 3 check-nodes, the construction is more difficult, since having large girths does not automatically result in a good distance properties. Chen *et al.* [131] provides an insightful example that flipping all the variable nodes in a cycle which are constituted of only degree-2 variable nodes will still leave the checks all satisfied and will therefore lead to an undetected error. Therefore, the d_{min} value of this code would be equal to the number of degree-2 variable nodes in that cycle. This observation led some authors [132], [133] to suggest that irregular codes should preferably have no degree-2 variable nodes.

Another important design aspect that has to be considered at an early stage of the LDPC construction is the issue of having a random (or more precisely pseudo-random) versus a more structured construction. It is widely accepted that in general, the former construction achieves a better performance in the waterfall region than structured LDPC codes having comparable parameters. However, we have already seen in Section III-B that structured constructions, such as for example, cyclic or QC codes, have lower-complexity encoding than most pseudo-random codes. The fact that the BER/BLER

---

[10] The study of stopping sets gained importance when Di *et al.* [126] managed to derive exact analytical BER performance curves for the LDPC-coded transmission over the BEC in terms of the distribution of the stopping set sizes. It is an often quoted result that the size of the smallest stopping set in the graph, which is called the stopping number or stopping distance, lower bounds the minimum distance of the code and essentially corresponds to the smallest number of erasures which cannot be recovered under iterative decoding.

performance exhibited by carefully designed structured LDPC codes can be comparable to that of pseudo-random constructions has been shown in a number of publications, for example in [114], [134]–[137].

D. Iterative Decoding Techniques for Low-Density Parity-Check Codes

The underlying principle of the different decoding techniques used for LDPC codes is that of having messages exchanged between the left and right nodes of the Tanner graph representing the code. The first decoding algorithm was introduced by Gallager [1], [17] and is commonly referred to as the bit-flipping (BF) algorithm. This hard-decoding technique was later improved by Kou *et al.* [60], who proposed a similar algorithm, referred to as the weighted bit-flipping (WBF) algorithm, which further exploits the bit-reliability information whilst still retaining the appealing conceptual and implementational simplicity of the BF algorithm. The BER performance and decoding complexity of the WBF algorithm were later improved by Nouh and Banihasehemi, using the so-called bootstrapped WBF (BWBF) algorithm [138]. The basic principle of the BWBF algorithm is to identify the symbols, which are less reliable than some predefined threshold (i.e., spotting the 'unreliable symbols') and then estimate their values as well as their corresponding reliabilities by exchanging information both with the more 'reliable' symbols and with the check nodes.[11] Inaba and Ohtsuki [139] investigated the performance of LDPC decoding using the BWBF technique for transmission over fast fading channels.

The WBF algorithm of [60] was also improved by Zhang and Fossorier [140] using a technique which is different from the BWBF solution of [138], by considering both the parity information supplied by the check nodes and that gleaned from the variable nodes. Their algorithm, which is referred to as the modified WBF (MWBF), was invoked for the decoding of LDPC codes based on FGs. Liu and Pados [141] modified the check node output in the decoding algorithm of [140]. Guo and Hanzo [142] improved the algorithm of [141] by using a reliability-based ratio and without relying on any off-line preprocessing. The BER performance exhibited by the bootstrap version of the MWBF was characterized by Inaba and Ohtsuki [143], where it was shown that the bootstrap MWBF (BMWBF) is capable of outperform the WBF, the MWBF and the BWBF algorithms, despite its lower decoding complexity.

As previously mentioned in Section III, soft decoding of LDPC codes is typically performed using the SPA, which achieves a better performance than hard decoding using the BF algorithm, at the expense of an increased complexity. We have also mentioned in Section III that the SPA comes under a number of different names, largely due to its independent discovery by different researchers. Its use has not been limited to the decoding of LDPC codes, it has also found employment in solving inference problems in artificial intelligence, in computer vision and in statistical physics.

The first soft decoding method proposed for LDPC codes was also introduced by Gallager [17] and was referred to as the probabilistic decoding method (please refer to Section 5.3 of [17]). In principle, this method is identical to Pearl's belief propagation (BP) [144], which was proposed in 1988 in the context of belief networks for solving inference problems. Although it gained popularity within the artificial intelligence community, it remained unknown to information theorists until it was employed by MacKay and Neal [39] as well as by McEliece *et al.* [145]. The latter work [145] created the link between turbo decoding and Pearl's belief propagation algorithm. Kschischang *et al.* [30] demonstrated that the SPA constitutes an instance of Pearl's BP operating on a factor graph [146].

Other researches focused their attention on reducing the complexity of the SPA. One of these reduced complexity algorithm is the min-sum algorithm introduced by Wiberg [26], which is very much related to the Viterbi algorithm and to Tanner's 'Algorithm B' [23]. A few years later, Fossorier *et al.* [147] proposed the universally most-powerful (UMP) - BP technique, which reduces the complexity of the check-to-source bit message passing by using a combination of hard- and soft-decisions. The normalised BP technique was later introduced by Chen and Fossorier [148], which improves the accuracy of soft values of the UMP-BP by multiplying the log-likelihood ratios during the check-to-source bit message exchange with a normalization factor. A genetic algorithm (GA) [149] based decoder designed for the LDPC codes was detailed by Scandurra *et al.* [150]. In contrast to the SPA decoder, the proposed GA-based decoder does not require the SNR value.[12] Its BER performance and its computational complexity can be readily modified by optimizing the GA's fitness function and the other GA's parameters.

Improving the performance of the conventional BP algorithm was also the focus of the contribution of Yedidia *et al.* [152] who introduced the generalized BP algorithm. The achievable performance improvement can be attributed to the fact that the generalized BP focuses its efforts on the messages exchanged by a group nodes rather than single nodes. Wang *et al.* [153] introduced the 'plain shuffled' and the 'replica shuffled' BP algorithm, as reduced-latency variants of the conventional BP and investigated their performance using both density evolution and EXIT charts. Further efforts were invested by Fossorier [154], who suggested the combination of ordered statistical decoding and the SPA for the decoding of LDPC codes. The output of the decoder is reprocessed using ordered statistical decoding in an attempt to bridge the gap between the performance exhibited by the SPA and the optimum maximum likelihood decoding, which has a potentially excessive complexity.

E. Convergence of the Iterative Decoding

The structure of the LDPC decoder is essentially constituted by a serial concatenation of two decoders; a VND and a CND separated by the so-called edge interleaver, as portrayed in

---

[11] A 'reliable' check node is defined as the check node, which is only connected to one 'unreliable' bit node [138], [139].

[12] The independence of the performance exhibited by an LDPC code on the assumed and actual noise level was investigated by MacKay and Hesketh [151] both for the binary symmetric and Gaussian channel.

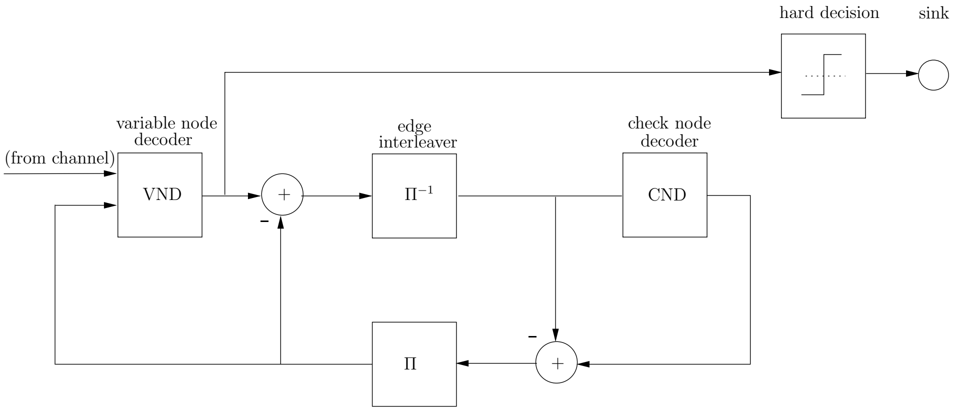


Fig. 4. The LDPC decoder consisting of a serial concatenation of the variable node decoder (VND) and check node decoder (CND) separated by an edge interleaver.

Figure 4. In parlance, the VND is referred as being the *inner* decoder, since it is the nearest to the communications channel, whilst the CND is referred to the outer decoder. Elaborating slightly further, each decoder can be mathematically described by a so-called EXIT function, which describes the average extrinsic mutual information of the respective decoder. The performance of the decoder can be then characterized by monitoring the exchange of extrinsic information between the two component decoders, which is pictorially represented by EXIT charts. EXIT charts were introduced by ten Brink [155] and became a popular tool for determining the convergence behavior[13] of any iterative decoding scheme.

An example of an EXIT chart is shown in Figure 5, which portrays the EXIT chart for a half-rate regular LDPC code that is associated with a parity-check matrix having a column weight of $\gamma = 3$ and a row weight of $\rho = 6$. We also assume binary phase shift keying (BPSK) modulated transmissions over the AWGN channel at E_b/N_0 = 2 dB. In Figure 5, we have explicitly marked the two EXIT curves, which correspond to the aforementioned EXIT function of the respective inner or outer constituent decoder, and the corresponding EXIT trajectory. The trajectory gives an estimate of the number of decoding iterations that are required to reach the perfect convergence to a vanishingly low BER, which corresponds to the $(1, 1)$ point of the EXIT chart. A single decoding iteration will correspond to one step on the corresponding EXIT trajectory.

Assuming this EXIT chart-based framework, there are three basic requirements to be satisfied in order to design a near-capacity system. Firstly, it is required that both the inner as well as the outer decoder's EXIT curves should reach the $(1, 1)$ point on the EXIT chart, in order to attain near-error-free decoding. Secondly, the inner decoder's curve should always be above the outer decoder's curve and hence should never intersect. This will result in an a so-called open tunnel between the two EXIT curves. If the two EXIT curves intersect and therefore no open tunnel will be available, the EXIT trajectory will fail to reach the error-free $(1, 1)$ point of the EXIT chart. Consequently, the resultant BER/BLER performance will exhibit high error floors.

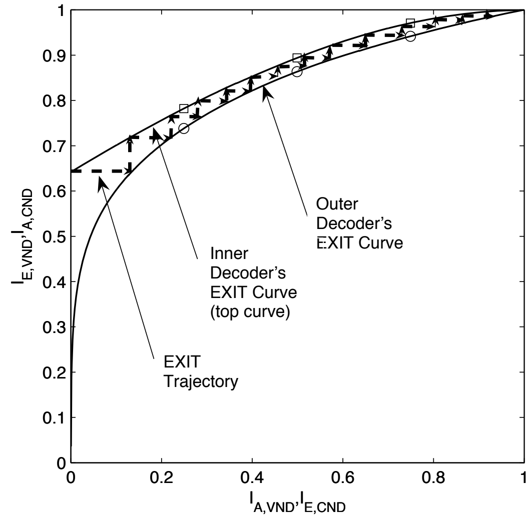


Fig. 5. The EXIT chart for a half-rate regular LDPC code, associated with a parity-check matrix having a column weight of $\gamma = 3$ and a row weight of $\rho = 6$. We also assume BPSK modulated transmission over the AWGN channel at E_b/N_0 = 2 dB. The a-priori information input and the extrinsic information output for the CND and VND are denoted by $I_{\mathrm{A,CND}}$, $I_{\mathrm{E,CND}}$, $I_{\mathrm{A,VND}}$ and $I_{\mathrm{E,CND}}$, respectively.

Thirdly, in order to maximize the achievable throughput, the two constituent decoder curves must match as accurately as possible, thus resulting in an infinitesimally low EXIT-chart-tunnel area. Indeed, a code that operates close to capacity has EXIT curves, which have a similar shape, as it was demonstrated for a variety of channels such as the BEC [156], single-input single-output as well as MIMO Gaussian channels [157], [158], for dispersive channels imposing inter-symbol interference [159] and for partial response [160] channels. As a consequence, it was also shown [156] that the area between the two EXIT curves is proportional to the SNR distance from capacity.[14] In this context, irregular codes allow for more flexibility in the design of their degree distribution and so, their corresponding EXIT curves can be better matched in order to attain a near-capacity performance. This can also be verified from Figures 6(a) and 6(b), which portray the

---

[13] The convergence behavior of a code can also be analyzed by means of the aforementioned density evolution [49].

[14] The EXIT curve matching can be very easily obtained using linear programming [161].

EXIT chart for a half-rate regular and irregular LDPC code, respectively. It can be observed that the open-tunnel area in the EXIT chart of the irregular code is significantly smaller than that of the corresponding regular counterpart. However, it is worth mentioning that the decoding complexity of the irregular LDPC code will be higher, since it requires more decoding iterations to reach the near-error-free $(1, 1)$ point of the EXIT chart.

Zheng *et al.* [162] discovered that there is only a 0.01 dB difference between the results predicted by using EXIT chart analysis in comparison to those determined by density evolution. However, EXIT chart analysis may be deemed to be more convenient, especially when considering that no Fourier and inverse Fourier transform computations are necessary. In the same paper [162], the EXIT chart analysis provided for LDPC codes was also extended to flat uncorrelated Rayleigh flat fading channels. Jian and Ashikhmin [163] utilize EXIT charts in order to determine the convergence SNR threshold for LDPC coded systems transmitting over flat Rayleigh fading channels and exploiting the knowledge of the channel impulse response. In Section III-C, we have mentioned that the convergence SNR threshold can be determined by finding the minimum SNR, at which the two EXIT curves no longer intersect and thus create a marginally open tunnel. In this context, we can observe from Figures 6(a) and 6(b) that the convergence SNR threshold of the regular and irregular LDPC code is equal to -1.71 dB (i.e., E_b/N_0 = 1.3 dB) and -2.51 dB (i.e., E_b/N_0 = 0.5 dB), respectively. The lower SNR threshold of the irregular code reaffirms our previous argument, namely that irregular LDPC codes are capable of attaining a superior performance in the waterfall region over their corresponding regular counterparts.

Typically, the variable-to-check and check-to-variable node information, as well as the channel's output messages are assumed to be Gaussian distributed [155], [157], [158], [164]–[166]. However, in practice this is not an accurate assumption for the check-to-variable node messages. The reason is essentially due to the fact that the check-node is performing a *tanh* operation and hence, the magnitude of the log-likelihood ratios at the output of the check node is typically smaller than that of the incoming messages at the CND. Thus, one can argue that the CND is producing the minimum soft value. This effectively makes the probability density function of the check-to-variable node messages skewed towards the origin, thus rendering their distribution non-Gaussian, especially at low SNR [167], [168]. However, according to Chung *et al.* [169], this approximation produces accurate result for codes having a code-rate between $R = 0.5$ and $R = 0.9$, provided that the variable nodes have degrees less than or equal to 10. Ardakani and Kschischang [167], [168] prefer to use the true histogram-based probability density function for the messages arriving from the check nodes and hence to produce a more accurate EXIT chart analysis. The same authors in [170] consider a general code design for achieving a specific desired convergence behavior and to provide the necessary as well as sufficient conditions satisfied by the EXIT chart of the highest rate LDPC code.

EXIT charts were also employed in the design of systems amalgamating coded modulation schemes and LDPC codes have been investigated in [171], [172]. The latter work by Francheschini *et al.* [172] presents a novel bound and design criterion, which directly links the EXIT chart analysis to the achievable BER performance, where the decoding convergence behavior has been characterized as a function of the LDPC code's degree distributions. This design criterion of [172] also provides a bound for the degree distribution coefficients, which must be satisfied in order to attain convergence within a specified number of iterations. Both density evolution and EXIT chart analysis were extended to the case of nonbinary LDPC codes by Rathi and Urbanke [173] as well as by Byers *et al.* [174], respectively.[15]

F. Hardware Implementation of Low-Density Parity-Check Codes

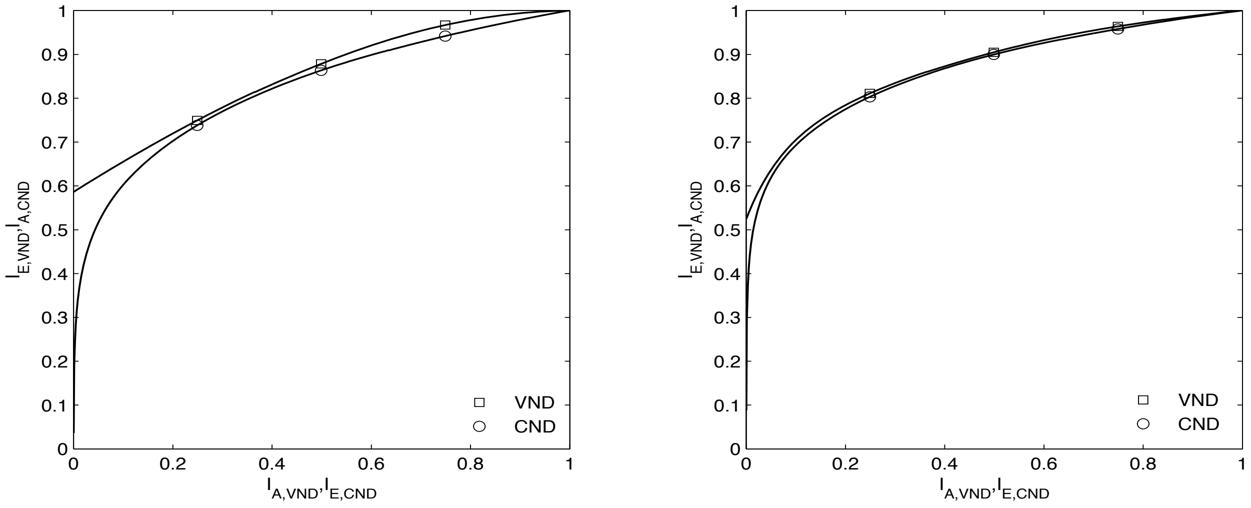
The hardware implementation of any channel code is typically orders of magnitude faster than their software-based counterparts, which results in a higher achievable bit rate. Hence it is desirable that the LDPC construction can be conveniently implemented in hardware. Several LDPC hardware implementations have been proposed, for example in [175]–[182], with many of them exploiting the speed and flexibility of field programmable gate arrays and of digital signal processors.

Whilst it can never be denied that pseudo-random codes such as the classic regular MacKay LDPC codes [42] and conditioned irregular codes [50], [125] exhibit an excellent BER/BLER performance, the random selection of the connections between their parity-check and variable nodes makes it particularly hard to create a convenient description for the code. Hence their implementation often results in either inflexible hardwired interconnections or large inefficient lookup tables. On the other hand, structured codes [51] benefit from simplified descriptions as well as from facilitating efficient read and write operations from/to memory. This underlines the argument that the design of an LDPC code construction has to maintain a good BER/BLER performance as well as to benefit from hardware-friendly implementations.

The primary factor which substantially affects the ease (or difficulty) of building an LDPC encoder is the description complexity, i.e., the amount of memory required to store the LDPC code's description, which is directly proportional to the number of nonzero bits in the **H** matrix or the number of edges in the corresponding Tanner Graph. For the case codes having a pseudo-random parity-check matrix, this simply means that the locations of all the nonzero bits of the **H** matrix must be enumerated. This is an important aspect to take into consideration, especially for those encoders that will be positioned in a remote location with limited source of power, for example in deep space [183]. In Section III-B, we have discussed the issue of the encoding complexity of LDPC codes, in particular, we referred to the work of Richardson and Urbanke [52], which demonstrated that in general, LDPC codes have a near-linearly block-length-dependent encoding complexity. Therefore it becomes evident that a desirable characteristic is to have a small gap factor. Preferably, the code construction will consist of circulant permutation matrices,

---

[15] Rathi and Urbanke [173] only considered transmission over the BEC.



(a) Regular LDPC code

(b) Irregular LDPC code

Fig. 6. The EXIT chart for (a) a half-rate regular LDPC code, associated with a parity-check matrix having a column weight of $\gamma = 3$ and a row weight of $\rho = 6$ at E_b/N_0 = 1.3 dB and (b) a half-rate irregular LDPC code at E_b/N_0 = 0.5 dB. The parity-check matrix for this irregular code follows the design of [158] and possesses 51% of the columns have a column weight of $\gamma = 2$, 42% of the columns have $\gamma = 4$ and 7% of the columns have $\gamma = 2$. All the rows of this irregular parity-check matrix have a row weight of $\rho = 8$. We also assume BPSK modulated transmission over the AWGN channel.

which makes it possible to carry out the encoding operation using shift registers.

The main challenge which has to be tackled, when implementing the SPA in hardware is that of effectively managing the exchange of extrinsic messages between the check and variable nodes. Howland and Blanskby [181] suggest two possible hardware architectures, namely a hardware-sharing and a parallel decoder architecture. After contrasting the two architectures, the authors opt for advocating the parallel decoder architecture, mainly for the reasons of its lower power dissipation and the reduced amount of control logic required, as well as owing to the inherent suitability of the architecture for the SPA. Andrews *et al.* [183] argue that the so-called protograph LDPC codes structured on a base protograph having a low number[16] of edges E^{b} are well-suited to semi-parallel hardware architectures. In fact, Lee *et al.* [184] proposed a hardware architecture, which is capable of simultaneously processing E^{b} edges per cycle, and therefore requiring $2J$ cycles per iteration, where J is the number of base protographs in the resultant protograph LDPC code. This implementation has the added advantage that the size of the protograph can also be tailored to match the available hardware.

In this context, it is worth mentioning that the task of designing an LDPC code that achieves a good BER/BLER performance and yet possesses implementational benefits is not at all simple. In [185], we have outlined the intricate dependencies that exist between the design attributes of LDPC codes and advocated code design techniques that aim for achieving the highest number of desirable attributes, rather than closely approaching the ultimate bounds, which hence tend to possess impractical hardware implementations. Constructions of LDPC codes using this design philosophy have been proposed in [136], [137], [186], [187], amongst others. Further insights related to the hardware implementation of LDPC codes are provided in [188].

G. Co-located versus Distributed Coding

A research area that has recently received substantial research attention lately is 'cooperative communications', which was originally referred to as 'cooperation diversity' [189]–[192]. The design of cooperative systems was motivated by the widely accepted fact that diversity is the most effective strategy of mitigating the effects of time-varying multipath fading in a wireless communication system. In practical terms, this directly implies that multiple antennas must be employed at the transmitter and the receiver, thus creating a MIMO system. One of the main benefits of MIMO systems is the linear increase in capacity with the number of transmitting antennas [193]–[196], provided that the number of receiver antennas matches this number. A further benefit of MIMOs is that they are capable of reducing the interference among different transmissions, they increase the diversity gain, the array and the spatial multiplexing gain. However, while employing multiple antennas at cellular base stations is practically realizable, it might be less feasible for the mobile terminals due to their limited size, battery power consumption and hardware complexity constraints.

This dilemma prompted researchers to move a further step away from having *co-located* MIMO elements to having *distributed* MIMO elements [197], [198]. This prompted a similar idea, which is now known as distributed coding. The most of the commonly used concatenated coding schemes are constituted by a number of constituent encoders/decoders. In this light, we may view traditional concatenated coding schemes as being a code having co-located components, since its constituent encoders/decoders are literally located within the same transmitter/receiver. On the other hand, a distributed

---

[16] Andrews *et al.* [183] suggest that the number of edges in the base protograph, hereby denoted by E^{b}, should be less than 300.

code involves having constituent components allocated to a number of geographically dispersed transmitters/receivers. For example, Zhao and Valenti [199] investigated a distributed turbo coded system, which effectively emulates a parallel concatenated convolutional code by encoding the data twice, first at the source and then at the relay (after interleaving). The data is then iteratively decoded at the destination by means of a classic turbo decoder.

In 2005, Bao and Li [200]–[203] proposed a solution that may be viewed as the first distributed LDPC code. Their strategy was in fact based on systematic low-density generator matrix (LDGM) based codes and on LDPC codes associated with lower triangular parity-check matrices. These two families of LDPC codes possess an **H** matrix that is comprised of the horizontal concatenation of a sparse matrix and a lower triangular (or in the case of systematic LDGM codes, an identity) matrix. In [200], [203], Bao and Li related these two matrices to two transmission phases of a cooperative communication system, whereby the first phase consists of what is known as the broadcast phase, whilst the second phase corresponds to the so-called relaying phase. In doing so, the authors allocated the function of the check-combiner to the relay, rather than being also performed by the original transmitter. However, Bao and Li do not portray their system as being a distributed LDPC coded system, rather they make the interesting proposal of representing the cooperative network by a Tanner graph, and in so doing, a code-on-graph [32] such as an LDPC code may be viewed in the above-mentioned context as 'network-on-graph' [200]–[203].[17] Subsequently, the information theoretic analysis of network-on-graphs was carried out in [204], [205]. Interestingly enough, the principles underlying networks-on-graph can be traced back to the roots of network coding [206]. The employment for LDPC codes for transmission over relay-aided channels was also suggested by Razaghi and Yu [207], Chakrabarti *et al.* [208] as well as by Hu and Duman [209], amongst many others.

H. Quantum Error Correction Codes

In the last decade or so, we have witnessed the emergence of what is now known as quantum information theory and quantum error correction [210]–[213]. It was Feyman who originally proposed the idea of processing information by means of quantum systems. A fundamental problem that arises is that of protecting the fragile quantum states from unwanted evolutions, whilst guaranteeing the robust implementation of the quantum processing devices. This phenomenon, referred to as decoherence, can be reduced by what is now known as quantum error correction.[18] Following the landmark papers of Shor [215] in 1995 and Steane [216], it was Calderbank and Shor [217] who provided the proof of existence of 'good' quantum error correction codes, even though they did not provide any explicit guidelines for their construction. These codes are often referred to as Calderbank-Shor-Steane (CSS) codes. These contributions further motivated researchers to construct interesting quantum codes based on classic binary codes, such as those proposed in [218]–[220]. Other quantum codes were based on the family of algebraic-geometric codes (see [221]–[224] amongst others).

In 2001, Postol proposed the first quantum CSS code constructed from classic FG-based LDPC codes [60]. This contribution was followed by MacKay *et al.* [225], who proposed quantum LDPC codes constructed with the aid of cyclic matrices. Camara *et al.* [226] presented two methods for constructing quantum LDPC codes and adopted the message passing algorithm for employment in generic quantum LDPC codes. Recently, Hagiwara and Imai [227] realized a CSS code with the aid of quantum QC LDPC codes. The first non-CSS quantum LDPC code was then proposed by Tan and Li in [228]. Recently, Djordjevic also proposed quantum LDPC codes based on balanced incomplete block designs [229] as well as quantum LDPC encoders and decoders for employment in an all-optical implementation [230].

IV. IMPORTANT MILESTONES IN THE HISTORY OF RATELESS CODING

Rateless codes were originally contrived for erasure channels and hence they were sometimes referred to as erasure-filling codes or simply, erasure codes. The foundation of erasure codes can be traced back to the proposal of the BEC in 1955 by Elias [231]. The encoded symbols transmitted over this channel can either be correctly received or completely erased with a probability of $(1 - P_\mathrm{e})$ and P_e, respectively. It was also demonstrated that a vanishingly low probability of error can be attained if random linear codes with rates close to $(1 - P_e)$ are decoded using an maximum likelihood decoder. The encoding and decoding complexity is at most a quadratic function of the block length.

However, research focusing on codes designed for the BEC remained dormant until the Internet became used on a large-scale basis during the mid-1990s. The only codes which can be regarded as being erasure-filling codes are the popular RS codes proposed in 1960 [86] and their relatives, such as the BCH codes [84], [85] as well as redundant residue number system codes [232]–[234]. Nonetheless, their employment for transmission over the BEC modeling the Internet channel has been hampered by the fact that a priori estimation of the channel's erasure probability has to be known and hence the code-rate has to be fixed before the actual transmission commences.

The quest for more efficient erasure-filling codes was initiated by Alon *et al.* [43], [235] and was first realized in the form of erasure-filling block codes designed on irregular bipartite graphs, which were termed as Tornado codes [44]. Their performance is however dependent on the validity of the assumption that the erasures are independent, which is not always true, especially when taking into account the binary erasures of the Internet channel imposed by statistical multiplexing-induced Internet protocol packet loss events. Moreover, their rate is still fixed like that of RS codes and hence, they cannot be used to serve multiple users communicating over channels having different qualities. Another effective erasure code was proposed by Rizzo [236] based on a class of generator matrix

---

[17] These networks-on-graph were commonly referred to as adaptive network coded cooperation or progressive network coding.

[18] The interested reader is referred to [214] for a thorough discussion on quantum error correction.

based codes, where the generator matrix was constructed to inherit the structure of the Vandermonde matrix [237].

Luby transform (LT) codes [238], proposed by Luby in 2002, can be considered as the first practical rateless code family, which are reminiscent of the ideal digital fountain code concept advocated by Byers *et al.* [239], [240]. Metaphorically speaking, a fountain code can be compared to an abundant water supply capable of sourcing a potentially unlimited number of encoded packets (water-drops) [241]. The receiver is capable of recovering K out of the N transmitted packets on a BEC, if N is sufficiently larger than K.

The encoding and decoding process of an LT code is conceptually appealing. Assume a message consisting of K input (source) symbols $\mathbf{v} = [v_1\, v_2 \ldots v_K]$, where each symbol contains an arbitrary number of bits.[19] The LT encoded symbol c_j, $j = 1, \ldots, N$, is simply the modulo-2 sum of d_c distinct input symbols, chosen uniformly at random. The actual degree of each symbol to be encoded is then chosen from a predefined distribution, which is typically either the robust soliton distribution or the so-called truncated Poisson 1 distribution. Given the nature of this encoding scheme, there is no limit on the possible number of encoded symbols that can be produced and for this reason, fountain codes such as LT codes are described as being rateless codes. LT codes also benefit from having a low encoding and decoding cost, avoiding an excessive complexity upon increasing the source's codeword length. Due to these characteristics, LT codes are considered to be universal in the sense that they are near-optimal and thus applicable for every type of erasure channels.

Similarly to the previously described LDPC codes, the connection between the input and output symbols can also be diagrammatically represented by means of a bipartite graph, which is commonly referred to as a Tanner [23] or a factor graph [30], as shown in Figure 7. In this context, an input source symbol can be treated as a variable node, whilst an LT encoded symbol can be regarded as a check node. The terminology of input/output symbols, source/LT-encoded symbols and variable/check nodes is interchangeably used in the literature.

The decoding process as detailed by Luby [238] commences by locating a self-contained symbol, i.e., a so-called degree-one input symbol which is not combined with any other. The decoder will then add (modulo-2) the value of this symbol to all the LT-encoded symbols relying on it and then removes the corresponding modulo-2 connections. The decoding procedure will continue in an iterative manner, each time commencing from a degree-one symbol. If no degree-one symbol is present at any point during the decoding process, the decoding operation will abruptly halt. However, a carefully designed degree distribution, such as the robust soliton distribution [238], guarantees that this does not occur more often than a predefined probability of decoding failure. This LT decoding process is illustrated in Figure 2 of [100]. Clearly, using this decoding technique for LT codes designed for transmission over noisy channels constitutes an additional challenge, since a single corrupted symbol will produce uncontrolled error propagation. This have led the authors in [242] to formalize the concept of a 'wireless erasure'. A cyclic redundancy check sequence is appended to a block of LT encoded symbols and are consequently declared to be erased if the cyclic redundancy check fails. In such a manner, the noisy channel can be effectively treated as a block erasure channel. A superior decoding strategy designed for LT codes transmitted over channels such as the binary symmetric and the AWGN channel is to allow the exchange of soft information between the source and LT-encoded symbols [242]–[244] in a fashion akin to that used for the decoding of LDPC codes.

A. *Other Rateless Codes And Their Performance Over Noisy Channels*

Palanki and Yedidia [244], [245] were the first to document the achieved performance of LT codes for transmission over the binary symmetric and the binary-input AWGN channels. More particularly, it was demonstrated that the BER and BLER performance of LT codes over these channels exhibit high error floors [244], [245]. For this reason, LT codes used for transmission over noisy channels have always been concatenated with other forward error correction schemes, such as iteratively detected bit-interleaved coded modulation [246], generalized LDPC [247], convolutional and turbo codes [242], [248], [249]. In the literature, the concatenation of LT codes with turbo codes was referred to as the turbo fountain [249] code.

Recently, we have also witnessed the emergence of Raptor codes [250], [251], which do not share the error floor problem of their predecessors. In fact, the results published in [244], [245], [252]–[258] attest near-capacity performance and 'universal-like' attributes on a variety of noisy channels. Note that our emphasis is on the phrase 'universal-like'; since it has been shown in [252] that Raptor codes are not exactly universal on symmetric channels, since their degree distribution is in fact dependent on the channel statistics. The benefits provided by Raptor codes were then exploited in a number of practical scenarios, such as for wireless relay channels [259]–[261] as well as for multimedia transmission [262]–[267]. Other types of rateless codes proposed in the literature are the systematic LT codes [268]–[271], the online codes [272], [273], the codes based on linear congruential recursions [274] as well as the LDPC-like Matrioshka codes [275], [276]. The latter codes were proposed as a solution to the Slepian-Wolf problem [277]. Caire *et al.* [243] delved into the applicability of rateless coding for variable-length data compression.

From another point of view, we can consider the family of rateless codes for the provision of incremental redundancy [278]–[281]; for example in the context of adaptive-rate schemes or as an instance of the so-called type-II hybrid automatic repeat-request (HARQ) [10], [282], [283] schemes. In such schemes, the transmitter continues to send additional incremental redundancies of a codeword until a positive ACK is received or all redundancy available for the current codeword was sent. If the latter case happens, i.e., the decoding is still unsuccessful after all the parity-bits have been sent, the codeword is either discarded or rescheduled for retransmission. The forward error correction codes that

---

[19] The terminology used in [238] refers to the original data message as a 'file'.

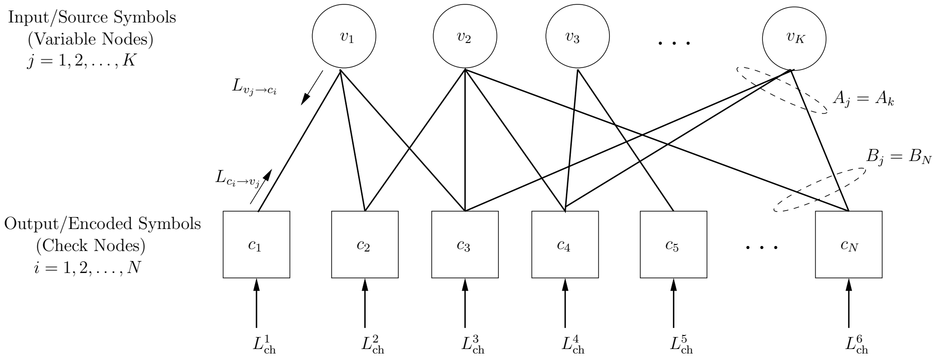


Fig. 7. A Tanner graph based description of LT code showing the source symbols (variable nodes) and the LT-encoded symbols (check nodes). The symbols are of an arbitrarily size.

are employed in conjunction with incremental redundanc are typically referred to as rate-compatible codes [284]. Th techniques applied in order to design rate-compatible cod either use puncturing [284]–[286] of the parity bits from low rate mother code in order to obtain higher rate codes employ code extension [6] for concatenating additional pari bits to a high-rate code in order create a low-rate code. Bo methods have their own limitations and typically a comb nation of both techniques is generally preferred [6], [287 The striking similarities of rateless coding with HARQ we first exploited by Soljanin *et al.* [288], [289], who compare the performance of Raptor codes as well as punctured LDP codes for transmission over the binary-input AWGN channe Their results demonstrated that the family of Raptor cod represents a more suitable alternative than punctured LDPC for covering an extensive range of channel SNRs (and thu rates).

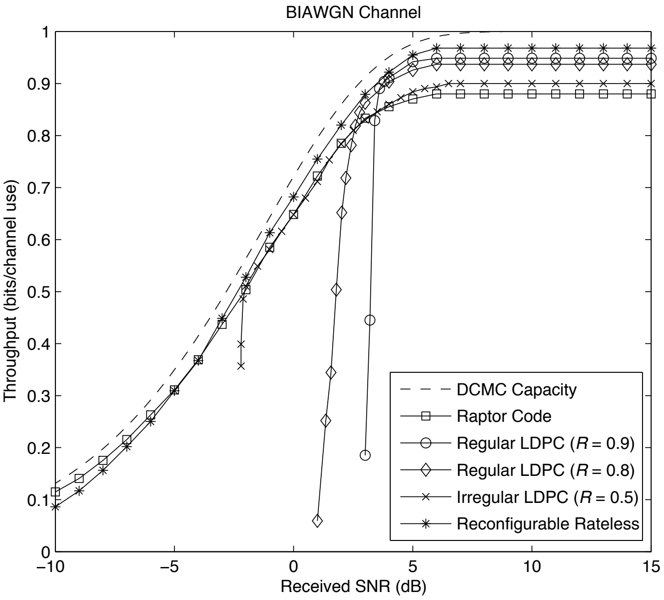


Fig. 8. Average throughput (bits/channel use) performance for transmission over the binary-input AWGN channel versus SNR (dB) using the proposed reconfigurable rateless codes as well as for the Raptor code [251] and the incremental-redundancy-based HARQ schemes employing punctured regular LDPC codes having $R = 0.8$ and 0.9 and an optimized punctured half-rate irregular LDPC code. The Raptor code and the punctured LDPC benchmarker codes followed the design presented in [288], [289]. The decoder employed the SPA and was limited to a maximum of 200 iterations. The number of information bits used for all the simulated schemes was set to 9500 bits.

The state-of-the-art rateless codes employ a fixed degre distribution [238]; i.e., the degree distribution used for coinin the degree d_c for each transmitted bit is time invariant and thus channel-independent. Consequently, such rateless codes, can only alter the number of bits transmitted (i.e., the code-rate) in order to cater for the variations of the channel conditions encountered. However, it was shown in [290] that a degree distribution designed for rateless coded transmissions over time-varying noisy channels will depend on the underlying channel characteristics, and therefore a fixed degree distribution can never be optimal[20] at all code rates. Motivated by this, the so-called reconfigurable rateless codes were proposed in [291]. These codes are capable of not only varying the block length (and thus the rate) but also adaptively modify their encoding strategy according to the prevalent channel conditions. Figure 8 compares the achievable throughput of the reconfigurable rateless codes with that of Raptor codes [251] and with punctured regular as well as with optimized irregular LDPC codes. It can be observed that reconfigurable rateless codes perform approximately 1 dB away from the discrete-input continuous-output memoryless channel's capacity over a diverse range of channel SNRs. Moreover, it can be verified that the performance of the proposed rateless reconfigurable codes is superior to that of punctured regular and irregular LDPC codes at all SNRs, and superior to that of the Raptor codes for all SNRs higher than -4 dB. Rateless codes sharing a somewhat similar philosophy were also explored in [292], in the context of MIMO transmit preprocessing systems.

Similarly to the case of LDPC codes, rateless codes have also been advocated in cooperative networks. Castura and Mao [259] proposed a half-relaying protocol using Raptor codes that naturally allows for their extension to multiple antennas and relays. A different approach was also suggested by Molisch *et al.* [293], [294]. Puducheri *et al.* proposed what are known at the time of writing as distributed LT codes, when considering a scenario, where the data is independently encoded from multiple sources and then combined at a common

---

[20] In this context, we use the adjective 'optimal' in terms of attaining a near-capacity performance.

relay. The authors proposed the degree selection distribution to be employed at the source to ensure that the resultant packet stream at the common relay has a degree distribution that approximates that of a conventional LT code.

B. Rateless Codes versus their Fixed-Rate Counterparts

In Section I-A, we have presented simplified arguments, which helped us to create a link between the well-understood fixed-rate coding and rateless coding families. In this context, it is worth elaborating slightly further by noting that some rateless code families are very closely related to their fixed-rate counterparts. For instance, an LT code [238] is analogous to a nonsystematic LDGM-based code [295], having a generator matrix that is calculated online (and thus allowing adaptive-rate configuration for diverse channel conditions) and where the LT encoded codeword corresponds to a sequence of repeated parity-check equation values, each checking the parity of d_c information bits. We remark that LDGM codes are essentially the dual codes of LDPC codes, where the latter codes where defined in Section II-B.

Similarly, we can regard Raptor codes [251] as a serial concatenation of a (typically) high-rate LDPC code as the outer code combined with a rateless LDGM code as the inner code. Both the LT as well as Raptor codes are decoded using the classic BP algorithm, in a similar fashion to the decoding of LDPC codes. However, in contrast to fixed-rate codes, code-design optimization techniques such as the often used girth-conditioning [57] or cycle-connectivity analysis [125] are inapplicable since the parity-check connections between the information and parity bits are determined "on-the-fly". Nonetheless, this is advantageous in terms of memory requirements, since there is no need to store the code description (e.g. the parity-check or the generator matrix).

V. CONCLUSIONS AND FUTURE DIRECTIONS

A. Summary of the Paper

In this article, we have provided a comprehensive survey of the associated open literature that is related to LDPC codes and their rateless relatives. We have commenced our discourse by outlining the related basic terminology and definitions in Section II. We have limited our elaborations to the basic principles of linear block codes, to the description of their generator and parity-check matrices as well as to their graphical representation. We have also touched upon some basic graph theoretical foundations. Following this preliminary foundation, we proceeded to provide a brief historical overview of LDPC codes. More specifically, in Section III-A, we provide a somewhat general discussion on the various LDPC code constructions that have been previously proposed in the literature. In Section III-B, we focused our attention on the literature concerning the encoding of LDPC codes. We stated that the encoding of conventional LDPC codes has a complexity that increases as a quadratic function of the block length. Subsequently, we detailed the proposed solutions, which mitigate these specific problems. In Section III-C, we outlined the BER/BLER performance metrics of LDPC codes and associated these metrics with the LDPC construction attributes. In Section III-D, we have summarized the majority of the previously presented LDPC decoding algorithms and discussed their complexity versus performance tradeoffs. The iterative decoding convergence was then discussed in Section III-E, and we outlined the basic principles of code design tools, such as the EXIT chart. In Sections III-G and III-H, we have focused our attention on current research topics related to distributed coding in cooperative communications as well as to the employment of LDPC codes in quantum error correction. Finally, in Section IV, we have outlined the most important milestones in the history of rateless coding and discussed some of the related design problems as well as their respective solutions.

B. Potential Future Research Directions

In this survey, we have offered a glimpse of six decades of research pertaining to LDPC codes as well of the more recent efforts concentrated on rateless coding. Beyond any doubt, LDPC and rateless codes will find employment in a myriad of other potential applications and be included in the forthcoming standards. Against this backdrop, we will also shed some light on some potential future research directions, which are outlined in the points below:

- From a practical perspective, we expect that research efforts will be shifted from that of solely focusing on attaining further (minute) gains in their attainable BER/BLER performance (or the achievable throughput in the case of rateless codes) to a more holistic approach, which attempts to strike the best balance between the associated design tradeoffs. A stronger focus on the cost minimization of the error correction codes is certainly to be expected.
- In Section III-C, we described the stopping sets and trapping sets as well as their role on the determination of the error floor region of the LDPC-coded BER/BLER performance. From a more theoretical viewpoint, contemporary research is also focusing on the effects of the so-called pseudo-codewords [296], [297], instantons [298], [299] and absorbing sets [300] on the error floor region of the LDPC-coded BER/BLER performance. Although exact analytical performance curves for the LDPC transmission over the BEC has already been formulated in terms of the distribution of the stopping set sizes [126], the exact nature of the relationship between these attributes and the achievable performance of LDPC-coded transmission over AWGN and fading channels remains to be found.
- We also foresee further developments on the design of short codes as well as on the exact formulation of the delay-limited performance bounds. Related work includes that of Lee and Blahut [301]–[303] as well as of Tüchler [304] for turbo codes, and that of [126], [305]–[307] for LDPC codes. However, we note that the emphasis of the latter was mostly placed on communications over the BEC.
- We also anticipate the exact formulation of complexity-limited performance bounds and the associated design of codes that attempt to perform close to derive bounds. Related work includes that of Yu *et al.* [308].
- Apart from the further exploitation of such codes in the quantum domain, we also predict more developments in

the employment of error control at the network layer. In this context, these advances will be expedited by a better understanding of the associated performance bounds as well as by the extension of the well-understood code-design-related tools to these upper layers. Future research might also consider the codes discussed in this treatise, for example when creating distributed error-correction codes exchanging extrinsic information between ad-hoc networks nodes [309], such as in inter-Vehicle communications [310], or in optical networks [311], in cognitive radio applications [312] and in space Internet [313].

REFERENCES

[1] R. G. Gallager, "Low-density parity-check codes," *IRE Trans. Inf. Theory*, vol. 45, pp. 21–28, Jan. 1962.

[2] C. Berrou, A. Glavieux, and P. Thitimajshima, "Near Shannon limit error-correcting coding and decoding: Turbo-codes," in *Proc. IEEE International Conf. Commun., Geneva Technical Program*, vol. 2, (Geneva, Switzerland), pp. 1064–1070, May 23–26, 1993.

[3] C. E. Shannon, "A mathematical theory of communication," *Bell System Technical J.*, vol. 27, pp. 379–423, 1948.

[4] G. D. Forney Jr. and D. J. Costello Jr., "Channel coding: The road to channel capacity," in *Proc. IEEE*, vol. 95, pp. 1150–1177, June 2007.

[5] J. Hagenauer, "Rate-compatible punctured convolutional codes (RCPC codes) and their applications," *IEEE Trans. Commun.*, vol. 36, pp. 389–400, Apr. 1988.

[6] J. Li and K. R. Narayanan, "Rate-compatible low density parity check codes for capacity-approaching ARQ schemes in packet data communications," in *Proc. International Conf. Commun. Internet Inf. Technol.*, (U.S. Virgin Islands), Nov. 2002.

[7] B. Sklar and F. J. Harris, "The ABCs of linear block codes," *IEEE Signal Process. Mag.*, vol. 21, pp. 14–35, July 2004.

[8] E. R. Berlekamp, *Algebraic Coding Theory*. Aegean Park Press, 1984.

[9] R. Hill, *A First Course in Coding Theory*. Oxford University Press Inc., NY, US, 1999.

[10] S. Lin and D. J. Costello Jr., *Error Control Coding*. Prentice Hall, Englewood Cliffs, NJ, Apr. 2004.

[11] J. H. van Lint, *Introduction to Coding Theory*. Springer-Verlag, Berlin, Germany, Third ed., 1999.

[12] F. J. MacWilliams and N. J. A. Sloane, *The Theory of Error-Correcting Codes*. Elsevier Science, 1977.

[13] R. J. McEliece, *The Theory of Information and Coding*. Cambridge University Press, 2002.

[14] W. Peterson and E. J. W. Jr., *Error Correcting Codes*. Cambridge, MA.: MIT Press, 2002.

[15] D. J. C. MacKay, *Information Theory, Inference and Learning Algorithms*. Cambridge, UK: Cambridge University Press, 2003.

[16] T. J. Richardson and R. L. Urbanke, *Modern Coding Theory*. Cambridge, UK: Cambridge University Press, Mar. 2008.

[17] R. G. Gallager, *Low-density Parity-check Codes*. Cambridge, MA: MIT Press, 1963.

[18] I. S. Reed and G. Solomon, "Polynomial codes over certain finite fields," *J. Society Industrial Applied Mathematics*, vol. 8, pp. 300–304, June 1960.

[19] G. D. Forney Jr., *Concatenated codes*. Cambridge, MA: MIT Press, 1966.

[20] V. V. Zyablov, "An estimation of the complexity of constructing binary linear cascade codes," *Problems Inf. Transmission*, vol. 7, pp. 3–10, 1971.

[21] V. V. Zyablov and M. S. Pinsker, "Estimation of error-correction complexity of Gallager low-density codes," *Problems Inf. Transmission*, vol. 11, pp. 18–28, 1976.

[22] G. A. Margulis, "Explicit construction of graphs without short cycles and low-density codes," *Combinatorica*, vol. 2, pp. 71–78, 1982.

[23] R. Tanner, "A recursive approach to low complexity codes," *IEEE Trans. Inf. Theory*, vol. 27, pp. 533–547, Sept. 1981.

[24] J. Rosenthal and P. O. Vontobel, "Construction of LDPC codes using Ramanujan graphs and ideas from Margulis," in *Proc.* 38th *Annual Allerton Conf. Commun., Control Comput.*, (Monticello, IL), pp. 248–257, Oct. 4–6 2000.

[25] D. J. C. MacKay and M. Postol, "Weaknesses of Margulis and Ramanujan-Margulis low-density parity-check codes," *Electron. Notes Theoretical Comput. Science*, vol. 74, no. 8, pp. 1–8, 2003.

[26] N. Wiberg, "Codes and decoding on general graphs," *PhD thesis, Linköping University, Department Electrical Engineering, Sweden*, 1996.

[27] N. Wiberg, H.-A. Löeliger, and R. Kotter, "Codes and iterative decoding on general graphs," in *European Trans. Telecommun.*, pp. 513–525, Sept. 1995.

[28] N. Wiberg, H.-A. Löeliger, and R. Kotter, "Codes and iterative decoding on general graphs," in *Proc. IEEEInternational Symp. Inf. Theory*, p. 468, 1995.

[29] G. D. Forney Jr., "The forward-backward algorithm," in *Proc.* 34th *Allerton Conf. Commun., Control Comput.*, (Monticello,IL), pp. 432–446, Oct. 1996.

[30] F. R. Kschischang, B. J. Frey, and H.-A. Loeliger, "Factor graphs and the sum-product algorithm," *IEEE Trans. on Inf. Theory*, vol. 47, pp. 498–519, Feb. 2001.

[31] S. M. Aji and R. J. McEliece, "The generalised distributed law," *IEEE Trans. Inf. Theory*, vol. 46, pp. 325–343, Mar. 2000.

[32] G. D. Forney Jr., "Codes on graphs: normal realizations," *IEEE Trans. Inf. Theory*, vol. 47, pp. 520–548, Feb. 2001.

[33] C. Berrou and A. Glavieux, "Near optimum error correcting coding and decoding: Turbo codes," *IEEE Trans. Commun.*, vol. 44, pp. 1261–1271, Oct. 1996.

[34] S. L. Goff, A. Glavieux, and C. Berrou, "Turbo-codes and high spectral efficiency modulation," in *Proc. IEEEInternational Conf. Commun.*, (New Orleans, LA), pp. 645–649, Oct. 1994.

[35] T. J. Richardson and R. L. Urbanke, "The renaissance of Gallager's low-density parity-check codes," *IEEE Commun. Mag.*, vol. 41, pp. 121–131, 2003.

[36] M. Sipser and D. A. Spielman, "Expander codes," in *Proc.* 35th *Annual IEEE Conf. Foundations Computer Science*, pp. 566–576, Nov. 1994.

[37] M. Sipser and D. A. Spielman, "Expander codes," *IEEE Trans. Inf. Theory*, vol. 42, pp. 1660–1686, Nov. 1996.

[38] D. A. Spielman, "Linear-time encodable and decodable error-correcting codes," *IEEE Trans. Inf. Theory*, vol. 42, pp. 1723–1731, Nov. 1996.

[39] D. J. C. MacKay and R. M. Neal, "Good codes based on very sparse matrices," in *Proc.* 5th *IMA Conf. Cryptography Coding*, Dec. 1995.

[40] D. J. C. MacKay and R. M. Neal, "Near Shannon limit performance of low density parity check codes," *Electron. Lett.*, vol. 32, pp. 1645–1646, Mar. 1996.

[41] D. J. C. MacKay and R. M. Neal, "Near Shannon limit performance of low density parity check codes," *Electron. Lett. (Reprint)*, vol. 33, pp. 457–458, Mar. 1997.

[42] D. J. C. MacKay, "Good error-correcting codes based on very sparse matrices," *IEEE Trans. Inf. Theory*, vol. 45, pp. 399–431, Mar. 1999.

[43] N. Alon and M. G. Luby, "A linear time erasure-resilient code with nearly optimal recovery," *IEEE Trans. Inf. Theory*, vol. 42, pp. 1732–1736, 1996.

[44] M. G. Luby, M. Mitzenmacher, M. A. Shokrollahi, D. A. Spielman, and V. Stemann, "Error-resilient codes," in *Proc. 29th Symp. Theory Comput.*, pp. 150–159, 1997.

[45] M. G. Luby, M. Mitzenmacher, M. A. Shokrollahi, and D. Spielman, "Analysis of low-density codes and improved designs using irregular graphs," in *Proc.* 30th *Annual Symp. Theory Comput.*, (San Francisco, CA), pp. 249–258, May 1998.

[46] M. G. Luby, M. Mitzenmacher, M. A. Shokrollahi, and D. Spielman, "Analysis of low-density codes and improved low-density parity-check codes using irregular graphs and belief propagation," in *Proc. IEEE International Symp. Inf. Theory*, (Boston, USA), p. 111, Aug. 1998.

[47] M. G. Luby, M. Mitzenmacher, M. A. Shokrollahi, and D. A. Spielman, "Improved low-density parity-check codes using irregular graphs," *IEEE Trans. Inf. Theory*, vol. 47, pp. 585–598, Feb. 2001.

[48] M. G. Luby, M. Mitzenmacher, and M. A. Shokrollahi, "Analysis of random processes via the And-Or tree evaluation," in *Proc.* 9th *Annual ACM-SIAM Symp. Discrete Algorithms*, (Dallas, TX), pp. 364–373, 1998.

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

[50] S. Y. Chung, G. D. Forney Jr., T. J. Richardson, and R. L. Urbanke, "On the design of low-density parity-check codes within 0.0045 dB of the Shannon limit," *IEEE Commun. Lett.*, vol. 5, pp. 58–60, Feb. 2001.

[51] J. M. F. Moura, J. Lu, and H. Zhang, "Structured low-density parity-check codes," *IEEE Signal Process. Mag.*, vol. 21, pp. 42–55, Jan. 2004.

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

[53] Y. Mao and A. H. Banihashemi, “A heuristic search for good LDPC codes at short block lengths,” in *Proc. IEEEInternational Conf. Commun.*, vol. 1, (Helsinki, Finland), pp. 41–44, Nov. 2001.

[54] J. Campello, D. S. Modha, and S. Rajagopalan, “Designing LDPC codes using bit-filling,” in *Proc. IEEE International Conf. Commun.*, (Helsinki, Finland), 2001.

[55] J. Campello and D. S. Modha, “Extended bit-filling and LDPC code design,” in *Proc. IEEE Global Telecommun. Conf.*, vol. 2, (San Antonio, TX), pp. 985–989, Nov. 25–29, 2001.

[56] X.-Y. Hu, E. Eleftheriou, and D.-M. Arnold, “Progressive edge-growth Tanner graphs,” in *Proc. IEEE Global Telecommun. Conf.*, vol. 2, (San Antonio, TX), pp. 995–1001, Nov. 25–29, 2001.

[57] X.-Y. Hu, E. Eleftheriou, and D. M. Arnold, “Regular and irregular progressive edge-growth Tanner graphs,” *IEEE Trans. Inf. Theory*, vol. 51, pp. 386–398, Jan. 2005.

[58] T. Asamov and N. Aydin, “LDPC codes of arbitrary girth,” *IEEE Trans. Inf. Theory*, vol. 1, pp. 498–519, Nov. 2002.

[59] Y. Kou, S. Lin, and M. P. C. Fossorier, “Low density parity-check codes: Construction based on finite geometries,” in *Proc. IEEE Global Telecommun. Conf.*, vol. 2, (San Franciso, CA), pp. 825–829, July 2000.

[60] Y. Kou, S. Lin, and M. P. C. Fossorier, “Low-density parity-check codes based on finite geometries: a rediscovery and new results,” *IEEE Trans. Inf. Theory*, vol. 47, pp. 2711–2736, Nov. 2001.

[61] D. J. C. MacKay and M. Davey, “Evaluation of Gallager codes for short block length and high rate applications.” [Online]. Available: wol.ra.phy.cam.ac.uk/mackay/.

[62] S. J. Johnson and S. R. Weller, “Regular low-density parity-check codes from combinatorial designs,” in *Proc. IEEE International Telecommun. Workshop*, (Cairns, Qld.), pp. 90–92, Sept. 2–7, 2001.

[63] S. J. Johnson and S. R. Weller, “Construction of low-density parity-check codes from Kirkman triple systems,” in *Proc. IEEE Global Telecommun. Conf.*, vol. 2, (San Antonio, TX), pp. 970–974, Nov. 25–29, 2001.

[64] B. Vasic, “Structured iteratively decodable codes based on Steiner systems and their application in magnetic recording,” in *Proc. IEEE Global Telecommun. Conf.*, vol. 5, (San Antonio, TX), pp. 2954–2960, Nov. 25–29, 2001.

[65] B. Vasic, E. M. Kurtas, and A. V. Kuznetsov, “LDPC codes based on mutually orthogonal Latin rectangles and their application in perpendicular magnetic recording,” *IEEE Trans. Magnetics*, vol. 38, pp. 2346–2348, Sept. 2002.

[66] B. Vasic, “High-rate low-density parity-check codes based on anti-Pasch affine geometries,” in *Proc. IEEE International Conf. Commun.*, vol. 3, pp. 1332–1336, Apr. 28–May 2, 2002.

[67] B. Vasic, E. M. Kurtas, and A. V. Kuznetsov, “Kirkman systems and their application in perpendicular magnetic recording,” in *IEEE Trans. Magnetics*, vol. 38, pp. 1705–1710, July 2002.

[68] B. Ammar, B. Honary, Y. Kou, J. Xu, and S. Lin, “Construction of low-density parity-check codes based on balanced incomplete block designs,” *IEEE Trans. Inf. Theory*, vol. 50, pp. 1257–1269, June 2004.

[69] B. Ammar, “Error protection and security for data transmission,” *PhD thesis, University of Lancaster*, 2004.

[70] I. F. Blake and R. Mullin, *The Mathematical theory of Coding*. NY, USA: Academic, 1975.

[71] H. Zhang and J. M. F. Moura, “Large-girth LDPC codes based on graphical models,” in *Proc. IEEE Signal Process. Wireless Commun.*, (Rome, Italy), July 15–18, 2003.

[72] H. Zhang and J. M. F. Moura, “Structured regular LDPC with large girth,” in *Proc. IEEE Global Telecommun. Conf.*, vol. 2, (San Francisco, CA), Dec. 2003.

[73] J. Lu and J. M. F. Moura, “Turbo design for LDPC codes with large girth,” in *Proc. IEEE Signal Process. Wireless Commun. Workshop*, (Rome, Italy), July 15–18, 2003.

[74] C. J. Colbourn and H. Dinitz, *The CRC Handbook of Combinatorial Designs*. Boca Raton, FL: CRC, 1996.

[75] M. C. Davey and D. J. C. MacKay, “Low density parity check codes over GF(q),” *IEEE Commun. Lett.*, vol. 2, pp. 165–167, June 1998.

[76] R. Peng and R.-R. Chen, “Application of nonbinary LDPC codes for communication over fading channels using higher order modulations,” in *Proc. IEEE Global Telecommun. Conf.*, (San Francisco, CA, USA), pp. 1–5, Nov. 2006.

[77] J. J. Boutros, A. Ghaith, and Y. Yuan-Wu, “Non-binary adaptive LDPC codes for frequency selective channels: code construction and iterative decoding,” in *Proc. IEEE Inf. Theory Workshop*, (Chengdu, China), pp. 184–188, Oct. 2006.

[78] J. J. Boutros, A. Ghaith, and Y. Yuan-Wu, “Nonbinary and concatenated LDPC codes for multiple-antenna transmission,” in *Proc.* 7th *Africon Conf. Africa*, (Gaborne, Botswana), pp. 83–88, Sept. 15–17, 2004.

[79] F. Guo and L. Hanzo, “Low complexity non-binary LDPC and modulation schemes communicating over MIMO channels,” in *Proc. IEEE* 60th *Veh. Technol. Conf.*, vol. 2, pp. 1294–1298, Sept. 26–29, 2004.

[80] R. H. Peng and R.-R. Chen, “Design of nonbinary LDPC codes over GF(q) for multiple-antenna transmission,” in *Proc. IEEE Military Commun. Conf.*, (Washington, DC), pp. 1–7, Oct. 2006.

[81] O. Alamri, F. Guo, M. Jiang, and L. Hanzo, “Turbo detection of symbol-based non-binary LDPC-coded space-time signals using sphere packing modulation,” in *Proc. IEEE* 62nd *Veh. Technol. Conf.*, vol. 1, pp. 540–544, Sept. 28–25, 2005.

[82] M. Lentmaier and K. S. Zigangirov, “On generalized low-density parity-check codes based on hamming component codes,” *IEEE Commun. Lett.*, vol. 3, pp. 248–250, Aug. 1999.

[83] J. J. Boutros, O. Pothier, and G. Zemor, “Generalized low density (Tanner) codes,” in *Proc. IEEE International Conf. Commun.*, (Vancouver,Canada), pp. 441–445, June 1999.

[84] A. Hocquenghem, “Codes correcteurs d’erreurs,” *Chiffres*, vol. 2, pp. 147–156, Sept. 1959.

[85] R. C. Bose and D. K. Ray-Chaudhuri, “On a class of error correcting binary group codes,” *Information and Control*, vol. 3, pp. 68–79, Mar. 1960.

[86] I. S. Reed and G. Solomon, “Polynomial codes over certain finite fields,” *J. Society Industrial Applied Mathematics*, vol. 8, pp. 300–304, June 1960.

[87] J. Chen and R. M. Tanner, “A hybrid coding scheme for the gilbert-elliot channel,” in *Proc. Allerton Conf. Commun., Control Comput.*, (Monticello, USA), Sept. 2004.

[88] N. Miladinovic and M. P. C. Fossorier, “Generalized LDPC codes with Reed-Solomon and BCH codes as component codes for binary channels,” in *Proc. IEEE Global Telecommun. Conf.*, (St. Louis, USA), pp. 1239–1244, Dec. 2005.

[89] S. Abu-Surra, G. Liva, and W. E. Ryan, “Low-floor tanner codes via hamming-node or rscc-node doping,” in *Proc.* 16th *Symp. Applied Algebra, Algebraic Algorithms Error Correcting Codes*, (Las Vegas, NV, USA), Oct. 2006.

[90] E. Paolini, M. P. C. Fossorier, and M. Chiani, “Analysis of generalized LDPC codes with random component codes for the binary erasure channel,” in *Proc.* 16th *Symp. Applied Algebra, Algebraic Algorithms Error Correcting Codes*, (Seoul, Korea), Dec. 2006.

[91] G. Liva, W. E. Ryan, and M. Chiani, “Quasi-cyclic generalized LDPC codes with low error floors,” *submitted to IEEE Trans. Commun.*.

[92] A. Moinian, B. Honary, and E. M. Gabidulin, “Generalized quasi-cyclic LDPC codes for wireless data transmission,” in *Proc. IET International Conf. Wireless Mobile Multimedia*, (Hangzhou, China), Nov. 6–9 2006.

[93] G. Liva and W. E. Ryan, “Short low-error-floor Tanner codes with Hamming nodes,” in *Proc. IEEE Military Commun. Conf.*, pp. 208–213, Oct. 17–20, 2005.

[94] G. Liva, S. Song, L. Lan, Y. Zhang, S. Lin, and W. E. Ryan, “Design of LDPC codes: A survey and new results,” *J. Commun. Software Syst.*, vol. 2, pp. 191–211, 09 2006.

[95] Y. Wang and M. P. C. Fossorier, “Doubly-generalized low-density parity-check codes,” in *Proc. IEEE International Symp. Inf. Theory*, (Seattle, USA), July 2006.

[96] E. Paolini, M. P. C. Fossorier, and M. Chiani, “Analysis of doubly-generalized LDPC codes with random component codes for the binary erasure channel,” in *Proc. Allerton Conf. Commun., Control Comput.*, (Monticello, USA), Sept. 2006.

[97] N. Miladinovic and M. P. C. Fossorier, “Generalized LDPC codes and generalized stopping sets,” *IEEE Trans. Commun.*, vol. 56, pp. 201–212, Feb. 2008.

[98] E. Paolini, M. P. C. Fossorier, and M. Chiani, “Doubly-generalized LDPC codes: Stability bound over the BEC,” *IEEE Trans. Inf. Theory*, vol. 55, pp. 1027–1046, Mar. 2009.

[99] M. G. Luby, M. Mitzenmacher, M. A. Shokrollahi, D. Spielman, and V. Stemann, “Practical loss resilient codes,” in *Proc.* 29th *Annual ACM Symp. Theory Comput.*, (Seattle, WA), pp. 150–159, 1997.

[100] M. G. Luby, M. Mitzenmacher, M. A. Shokrollahi, and D. A. Spielman, “Efficient erasure correcting codes,” *IEEE Trans. Inf. Theory*, vol. 47, pp. 569–584, Feb. 2001.

[101] D. J. C. MacKay, S. T. Wilson, and M. C. Davey, “Comparison of constructions of irregular gallager codes,” in *Proc.* 36th *Allerton Conf. Commun., Control Comput.*, (Monticello, IL, USA), Sept. 23–25 1998.

[102] T. J. Richardson and R. L. Urbanke, “Efficient encoding of low-density parity check codes,” *IEEE Trans. Commun.*, vol. 47, pp. 808–821, Feb. 2001.

[103] D. Haley, A. Grant, and J. Buetefuer, "Iterative encoding of low-density parity-check codes," in *Proc. IEEE Global Telecommun. Conf.*, vol. 2, (Taipei, Taiwan), pp. 1289–1293, Nov. 17–21, 2002.

[104] D. Burshtein, S. Freundlich, and S. Litsyn, "Approximately lower triangular ensembles of LPDC codes with linear encoding complexity," in *Proc. IEEE International Symp. Inf. Theory*, (Seattle, WA, USA), pp. 821–825, July 9–14, 2006.

[105] D. Divsalar, H. Jin, and R. J. McEliece, "Coding theorems for "Turbo-Like" codes," in *Proc.* 36th *Annual Allerton Conf. Commun., Control Comput.*, pp. 201–210, Sept. 1998.

[106] H. Jin, A. Khandekar, and R. J. McEliece, "Irregular repeat-accumulate codes," in *Proc.* 2nd *International Symp. Turbo Codes Related Topics*, (Brest, France), pp. 1–8, Sept. 2000.

[107] A. Roumy, S. Guemghar, G. Caire, and S. Verdu, "Design methods for irregular repeat-accumulate codes," *IEEE Trans. Inf. Theory*, vol. 50, pp. 1711–1727, Aug. 2004.

[108] A. Abbasfar, D. Divsalar, and K. Yao, "Accumulate repeat accumulate coded modulation," in *Proc. Military Commun. Conf.*, vol. 1, pp. 169–174, Oct. 31–Nov. 3, 2004.

[109] D. Divsalar, S. Dolinar, and J. Thorpe, "Accumulate-repeat-accumulate-accumulate-codes," in *Proc. IEEE* 60th *Veh. Technol. Conf.*, vol. 3, pp. 2292–2296, Sept. 26–29, 2004.

[110] R. E. Blahut, *Algebraic Codes for Data Transmission*. Cambridge University Press, jul 2002.

[111] N. Hamada, "On the *p*-rank of the incidence matrix of a balance or partial balanced incomplete block designs and its application to error correcting codes," *Hiroshima Mathematical J.*, vol. 3, pp. 153–226, 1973.

[112] B. Ammar, B. Honary, Y. Kou, and S. Lin, "Construction of low density parity check codes: a combinatoric design approach," in *Proc. IEEE International Symp. Inf. Theory*, (Lausanne, Switzerland), p. 311, June 30–July 5, 2002.

[113] S. Lin, L. Chen, J. Xu, and I. Djurdjevic, "Near Shannon limit quasi-cyclic low-density parity-check codes," in *Proc. IEEE Global Telecommun. Conf.*, vol. 4, pp. 2030–2035, Dec. 1–5, 2003.

[114] L. Chen, J. Xu, I. Djurdjevic, and S. Lin, "Near-Shannon-limit quasi-cyclic low-density parity-check codes," *IEEE Trans. Commun.*, vol. 52, pp. 1038–1042, July 2004.

[115] H. Tang, J. Xu, Y. Kou, S. Lin, and K. Abdel-Ghaffar, "On algebraic construction of Gallager low density parity check codes," in *Proc. IEEE International Symp. Inf. Theory*, (Lausanne, Switzerland), p. 482, June 30–July 5, 2002.

[116] H. Tang, J. Xu, Y. Kou, S. Lin, and K. Abdel-Ghaffar, "On algebraic construction of Gallager and circulant low-density parity-check codes," *IEEE Trans. Inf. Theory*, vol. 50, pp. 1269–1279, June 2004.

[117] J. L. Fan, "Array codes as low-density parity-check codes," in *Proc.* 2nd *International Symp. Turbo Codes*, vol. 3, (Brest, France), pp. 543–546, 2000.

[118] M. P. C. Fossorier, "Quasi-cyclic low-density parity-check codes from circulant permutation matrices," *IEEE Trans. Inf. Theory*, vol. 50, pp. 1788–1793, Aug. 2004.

[119] Z. Li, L. Chen, L. Zeng, S. Lin, and W. Fong, "Efficient encoding of quasi-cyclic low-density parity-check codes," in *Proc. IEEE Global Telecomm. Conf.*, vol. 3, Nov. 28–Dec. 2, 2005.

[120] Z. Li, L. Chen, L. Zeng, S. Lin, and W. Fong, "Efficient encoding of quasi-cyclic low-density parity-check codes," *IEEE Trans. Commun.*, vol. 53, pp. 71–81, Nov. 2005.

[121] Z. Li, L. Chen, L. Zeng, S. Lin, and W. H. Fong, "Efficient encoding of quasi-cyclic low-density parity-check codes," *IEEE Trans.Commun.*, vol. 54, pp. 71–81, Jan. 2006.

[122] F. Zhang, Y. Xu, X. Mao, and W. Zhou, "High girth LDPC codes construction based on combinatorial design," in *Proc. IEEE* 61st *Veh. Technol. Conf.*, vol. 1, pp. 591–594, May 30–June 1, 2005.

[123] Y. Mao and A. H. Banihashemi, "Design of good LDPC codes using girth distribution," in *Proc. IEEE International Symp. on Inf. Theory*, (Sorrento, Italy), June 25–30, 2000.

[124] X.-Y. Hu, E. Eleftheriou, and D.-M. Arnold, "Irregular progressive edge-growth (PEG) Tanner graphs," in *Proc. IEEE International Symp. Inf. Theory*, (Lausanne, Switzerland), p. 480, June 30–July 5, 2002.

[125] T. Tian, C. R. Jones, J. D. Villasenor, and R. D. Wesel, "Selective avoidance of cycles in irregular LDPC code construction," *IEEE Trans. Commun.*, vol. 52, pp. 1242–1247, Aug. 2004.

[126] C. Di, D. Proietti, I. E. Telatar, T. J. Richardson, and R. L. Urbanke, "Finite-length analysis of low-density parity-check codes on the binary erasure channel," *IEEE Trans. Inf. Theory*, vol. 48, pp. 1570–1579, June 2002.

[127] T. J. Richardson, "Error floors of LDPC codes," in *Proc. the* 41st *Annual Allerton Conf. Commun., Control Comput.*, (Urbana-Champaign, US), pp. 1426–1435, Oct. 2003.

[128] C. C. Wang, "On the exhaustion and elimination of trapping sets: Algorithms & the suppressing effect," in *Proc. IEEE International Symp. Inf. Theory*, (Nice, France), pp. 2271–2275, June 24–29, 2007.

[129] A. I. V. Casado, M. Griot, and R. D. Wesel, "Improving LDPC decoders via informed dynamic scheduling," in *Proc IEEE Inf. Theory Workshop*, (Lake Tahoe, CA), pp. 208–213, Sept. 2–6, 2007.

[130] Y. Han and W. E. Ryan, "LDPC decoder strategies for achieving low error floors," in *Proc. Inf. Theory Appl. Workshop*, (San Diego, CA), pp. 277–286, Jan. 27–Feb. 2 2008.

[131] J. Chen, R. M. Tanner, J. Zhang, and M. P. C. Fossorier, "Construction of Irregular LDPC Codes by Quasi-cyclic Extension," *IEEE Trans. Inf. Theory*, vol. 53, pp. 1479–1483, Apr. 2007.

[132] T. J. Richardson, M. A. Shokrollahi, and R. L. Urbanke, "Design of capacity-approaching irregular low-density parity-check codes," *IEEE Trans. Inf. Theory*, vol. 47, pp. 619–637, Feb. 2001.

[133] M. Yang and W. E. Ryan, "Lowering the error-rate floors of moderate-length high-rate irregular LDPC codes," in *Proc. IEEE International Symp. Inf. Theory*, (Yokohama, Japan), p. 237, June 29–July 4, 2003.

[134] S. Lin, J. Xu, I. Djurdjevic, and H. Tang, "Hybrid construction of LDPC codes," in *Proc.* 40th *Annual Conf. Commun., Control Comput.*, (Monticello, IL), pp. 1149–1158, Oct. 2002.

[135] J. Xu and S. Lin, "A combinatoric superposition method for constructing low-density parity-check codes," in *Proc. IEEE International Symp. Inf. Theory*, vol. 30, (Yokohama,Japan), June 2003.

[136] N. Bonello, S. Chen, and L. Hanzo, "Construction of regular quasi-cyclic protograph LDPC codes based on Vandermonde matrices," *IEEE Trans. Veh. Technol.*, vol. 57, pp. 2583–2588, July 2008.

[137] N. Bonello, S. Chen, and L. Hanzo, "Multilevel structured low-density parity-check codes for AWGN and Rayleigh channels." submitted to the IEEE Trans. Veh. Technol..

[138] A. Nouh and A. H. Banihashemi, "Bootstrap decoding of low-density parity-check codes," *IEEE Commun. Lett.*, vol. 6, pp. 391–393, Sept. 2002.

[139] Y. Inaba and T. Ohtsuki, "Performance of low density parity check (LDPC) codes with bootstrap decoding algorithm on a fast fading channel," in *Proc. IEEE* 59th *Veh. Technol. Conf.*, vol. 1, pp. 333–337, May 17–19, 2004.

[140] J. Zhang and M. P. C. Fossorier, "A modified weighted bit-flipping decoding of low-density parity-check codes," *IEEE Commun. Lett.*, vol. 8, pp. 165–167, Mar. 2003.

[141] Z. Liu and D. A. Pados, "A decoding algorithm for finite geometry LDPC codes," *IEEE Trans. Commun.*, vol. 53, pp. 415–421, Mar. 2005.

[142] F. Guo and L. Hanzo, "Reliability ratio based weighted bit-flipping decoding for LDPC codes," in *Proc. IEEE Veh. Technol. Conf.*, (Stockholm, Sweden), pp. 415–421, May 2005.

[143] Y. Inaba and T. Ohtsuki, "Bootstrapped modified weighted bit flipping decoding of low density parity check codes," *IEICE Trans. Fundamentals*, vol. 1E89-A, pp. 1145–1149, May 17–19, 2006.

[144] J. Pearl, *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. San Mateo, CA: Morgan Kaufman, 1988.

[145] R. J. McEliece, D. J. C. MacKay, and J.-F. Cheng, "Turbo decoding as an instance of Pearl's belief propagation algorithm," *IEEE J. Sel. Areas Commun.*, vol. 16, pp. 140–152, Feb. 1998.

[146] H.-A. Loeliger, J. Dauwels, J. Hu, S. Korl, L. Ping, and F. R. Kschischang, "The factor graph approach to model-based signal processing," in. *Proc. IEEE*, vol. 95, pp. 1295–1322, June 2007.

[147] M. P. C. Fossorier, M. Mihaljevic, and H. Imai, "Reduced complexity iterative decoding of low density parity-check codes based on belief propagation," *IEEE Trans. Inf. Theory*, vol. 47, pp. 673–680, May 1999.

[148] J. Chen and M. P. C. Fossorier, "Near optimum Universal Belief Propagation based decoding of low-density parity-check codes," *IEEE Trans. Commun.*, vol. 50, pp. 406–414, Mar. 2002.

[149] D. E. Goldberg, *Genetic Algorithms in Search, Optimization, and Machine Learning*. NY: Addison-Wesley, 1989.

[150] A. G. Scandurra, A. L. D. Pra, L. Arnone, L. Passoni, and J. C. Moreira, "A genetic-algorithm based decoder for low density parity check codes," *Latin American Applied Research*, vol. 36, pp. 169–172, Sept. 2006.

[151] D. J. C. MacKay and C. P. Hesketh, "Performance of low density parity check codes as a function of actual and assumed noise levels," *Electron. Notes Theoretical Comput. Sci.*, vol. 74, pp. 1–8, 2003.

[152] J. S. Yedidia, W. T. Freeman, and Y. Weiss, "Generalized belief propagation," in *Advances Neural Inf. Proc. Syst. (NIPS)*, pp. 689–695, MIT Press, 2001.

[153] Y. Wang, J. Zhang, M. P. C. Fossorier, and J. S. Yedidia, "Reduced latency iterative decoding of LDPC codes," in *Proc. IEEE Global Telecommun. Conf.*, vol. 3, 2005.

[154] A. Roumy, S. Guemghar, G. Caire, and S. Verdu, "Iterative reliability based decoding of low-density parity check codes," *IEEE J. Sel. Areas Commun.*, vol. 19, pp. 908–917, May 2001.

[155] S. ten Brink, "Convergence of iterative decoding," *IEE Electron. Lett.*, vol. 35, pp. 806–808, May 13, 1999.

[156] A. Ashikhmin, G. Kramer, and S. ten Brink, "Extrinsic information transfer functions: A model and two properties," *IEEE Trans. Inf. Theory*, vol. 50, pp. 2657–2673, Nov. 2004.

[157] S. ten Brink and G. Kramer, "Design of repeat-accumulate codes for iterative detection and decoding," *IEEE Trans. Signal Process.*, vol. 51, pp. 2764–2772, Nov. 2003.

[158] S. ten Brink, G. Kramer, and A. Ashikhmin, "Design of low-density parity-check codes for modulation and detection," *IEEE Trans. Commun.*, vol. 52, pp. 670–678, Apr. 2004.

[159] M. Franceschini, G. Ferrari, , and R. Raheli, "EXIT chart-based design of LDPC codes for inter-symbol interference channels," in *Proc. IST Mobile Wireless Commun. Summit*, (Dresden, Germany), June 19–23, 2005.

[160] H. Song, J. Liu, and V. Kumar, "Convergence analysis of iterative soft decoding in partial response channels," *IEEE Trans. Magnetics*, vol. 39, pp. 2552–2554, Sept. 2003.

[161] Y. Yang, X. Changqing, and Z. Haibin, "Design of low-density parity-check codes using linear programming for modulation and detection," in *Proc. IEEE* 62$^{\mathrm{nd}}$ *Veh. Technol. Conf.*, vol. 1, pp. 532–535, Sept. 28–25, 2005.

[162] H. Zheng, X. Jin, and H. Hu, "More accurate performance analysis for BP decoding of LDPC codes using EXIT trajectories," in *Proc. IEEE International Symp. Commun. Inf. Technol.*, vol. 2, (Beijing , China), pp. 1392–1395, Oct. 12–14, 2005.

[163] Y. Jian and A. Ashikhmin, "LDPC codes for flat Rayleigh fading channels with channel side information," *submitted to IEEE Trans. Commun.*.

[164] A. Ashikhmin, G. Kramer, and S. ten Brink, "Extrinsic information transfer functions: model and erasure channel properties," *IEEE Trans. Inf. Theory*, vol. 50, pp. 2657–2673, Nov. 2004.

[165] F. Lehman and G. M. Maggio, "An approximate analytical method of the message passing decoder of LDPC codes," in *IEEE International Symp. Inf. Theory*, (Lausanne, Switzerland), p. 31, July 2002.

[166] S. R. Kollu and H. Jafarkhani, "On the EXIT chart analysis of low-density parity-check codes," in *Proc. IEEE Global Telecommun. Conf.*, vol. 3, pp. 1131–1136, Nov. 28–Dec. 2, 2005.

[167] M. Ardakani and F. R. Kschischang, "Designing irregular LPDC codes using EXIT charts based on message error rate," in *Proc. IEEE International Symp. Inf. Theory*, (Lausanne, Switzerland), p. 454, June 30–July 5, 2002.

[168] M. Ardakani and F. R. Kschischang, "A more accurate one-dimensional analysis and design of irregular LDPC codes," *IEEE Trans. Commun.*, vol. 52, pp. 2106–2114, Dec. 2004.

[169] 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 Trans. Inf. Theory*, vol. 47, pp. 657–670, Feb. 2001.

[170] M. Ardakani, T. H. Chan, and F. R. Kschischang, "EXIT-chart properties of the highest-rate LDPC code with desired convergence behavior," *IEEE Commun. Lett.*, vol. 9, pp. 52–54, Jan. 2005.

[171] N. Huaning and J. Ritcey, "Threshold of LDPC coded BICM for Rayleigh fading: Density evolution and EXIT chart," in *Proc. IEEE Wireless Commun. Netw. Conf. (WCNC))*, vol. 4, (Atlanta GA, USA), pp. 2422–2427, Mar. 21–25 2004.

[172] M. Franceschini, G. Ferrari, and R. Raheli, "LDPC-coded modulation: Performance bounds and a novel design criterion," in *Proc. Turbo Coding Symp.*, (Munich, Germany), Apr. 3–7, 2006.

[173] V. Rathi and R. L. Urbanke, "Density evolution, thresholds and the stability condition for non-binary LDPC codes," *IEE Proc. Commun.*, vol. 152, pp. 1069–1074, Dec. 9, 2005.

[174] G. J. Byers and F. Takawira, "EXIT charts for non-binary LDPC codes," in in *Proc. IEEE International Conf. Commun.*, vol. 1, pp. 652–657, May 16–20, 2005.

[175] B. Levine, R. R. Taylor, and H. Schmit, "Implementation of near Shannon limit error-correcting codes using reconfigurable hardware," in *Proc. IEEEField-Programmable Custom Comput. Machines*, (Napa Valley, CA), pp. 217–226, Apr. 17–19, 2000.

[176] T. Zhang, Z. Wang, and K. K. Parhi, "On finite precision implementation of low density parity check codes decoder," in *Proc. IEEE Circuits Syst. ISCAS*, vol. 4, (Sydney, NSW), pp. 202–205, May 6–9, 2001.

[177] T. Zhang and K. K. Parhi, "VLSI implementation-oriented $(3, k)$-regular low-density parity-check codes," in *Proc. IEEE Workshop Signal Process. Syst.*, (Antwerp, Belgium), pp. 25–36, Sept. 26–28, 2001.

[178] T. Zhang and K. K. Parhi, "A 54 Mbps (3,6)-regular FPGA LDPC decoder," in *Proc. IEEE Workshop Signal Process. Syst.*, pp. 127–132, Oct. 16–18, 2002.

[179] H. Zhong and T. Zhang, "Design of VLSI implementation-oriented LDPC codes," in *Proc. IEEE* 58$^{\mathrm{th}}$ *Veh. Technol. Conf.*, vol. 1, pp. 670–673, Oct. 6–9, 2003.

[180] T. Bhatt, K. R. Narayanan, and N. Kehtarnavaz, "Fixed-point dsp implementation of low-density parity check codes," in *Proc. IEEE Digital Signal Process. Workshop*, 2000.

[181] C. J. Howland and A. J. Blanksby, "Parallel decoding architectures for low density parity check codes," in *Proc. IEEE International Symp. Circuits Syst.*, vol. 4, pp. 742–745, 2001.

[182] C. J. Howland and A. J. Blanksby, "A 220mW 1 Gbps 1024-Bit rate-1/2 low-density parity-check code decoder," in *Proc. IEEE Custom Integrated Circuit Conf.*, pp. 293–296, 2001.

[183] K. Andrews, S. Dolinar, D. Divsalar, and J. Thorpe, "Design of low-density parity-check codes LDPC codes for deep-space applications," IPN Progress Report 42-159, Jet Propulsion Laboratory, Nov. 2004. [Online]. Available: $\mathrm{http://ipnpr.jpl.nasa.gov/progress_report/42-159/159K.pdf}$.

[184] J. K.-S. Lee, B. Lee, J. Thorpe, K. Andrews, S. Dolinar, and J. Hamkins, "A scalable architecture of a structured LDPC decoder," in *Proce. IEEE International Symp. Inf. Theory*, p. 292, June 27–July 2, 2004.

[185] N. Bonello, S. Chen, and L. Hanzo, "On the design of low-density parity-check codes." revised for the *IEEE Commun. Mag.*.

[186] Y. Zhu and C. Chakrabarti, "Architecture-aware LDPC code design for software defined radio," in *Proc. IEEE Workshop Signal Process. Syst. Des. Implementation*, pp. 405–410, Oct. 2–4, 2006.

[187] D. Kania and W. Sulek, "Code construction algorithm for architecture aware LDPC codes with low-error-floor," in *Proc. IEEE Region 8 International Conf. Computational Technol. Electrical Electron. Eng.*, (Novosibirsk,Russia), pp. 1–6, July 21–25, 2008.

[188] K. Gracie and M. H. Hamon, "Turbo and turbo-like codes: Principles and applications in telecommunications," in *Proc. IEEE*, vol. 95, pp. 1228–1254, June 2007.

[189] A. Serdonaris, E. Erkip, and B. Aazhang, "Increasing uplink capacity via user cooperation diversity," in *Proc. IEEE International Symp. Inf. Theory*, (Boston, USA), Aug. 1998.

[190] A. Serdonaris, E. Erkip, and B. Aazhang, "User cooperation diversity - part I: System description," *IEEE Trans. Commun.*, vol. 51, pp. 1927–1938, Nov. 2003.

[191] A. Serdonaris, E. Erkip, and B. Aazhang, "User cooperation diversity - part II: System description," *IEEE Trans. Commun.*, vol. 51, pp. 1939–1948, Nov. 2003.

[192] J. N. Laneman, *Cooperative diversity in wireless networks: Algorithms and Architectures*. PhD thesis, MIT, Cambridge, MA, Aug. 2002.

[193] G. J. Foschini and M. J. Gans, "On limits of wireless communications in a fading environment when using multiple antennas," *Wireless Personal Commun.*, no. 6, pp. 315–335, 1998.

[194] I. E. Telatar, "Capacity of multi-antenna Gaussian channels," *European Trans. Commun.*, vol. 10, pp. 585–596, Nov. 1999.

[195] D. Gesbert, M. Shafi, D.-S. Shiu, P. J. Smith, and A. Naguib, "From theory to practice: an overview of MIMO space-time coded wireless systems," *IEEE J. Sel. Areas Commun.*, vol. 21, pp. 281–302, Apr. 2003.

[196] P. Stoica, Y. Jiang, and J. Li, "On MIMO channel capacity: an intuitive discussion," *IEEE Signal Process. Mag.*, vol. 22, pp. 83–84, May 2005.

[197] H. Dai, "Distributed versus co-located MIMO systems with correlated fading and shadowing," in *Proc. IEEE International Conf. Acoustics, Speech Signal Process. (ICASSP)*, vol. 4, (Toulouse), May 14–19, 2006.

[198] J. Mietzner, *Spatial Diversity in MIMO communication systems with distributed or co-located antennas*. PhD thesis, Christian-Albrechts, University of Kiel, Kiel, Germany, Dec. 2006.

[199] B. Zhao and M. C. Valenti, "Distributed turbo coded diversity for the relay channel," *IEE Electron. Lett.*, vol. 39, pp. 786–787, May 2003.

[200] X. Bao and J. Li, "Matching code-on-graph with network-on-graph: Adaptive network coding for wireless relay networks," in *Proc. Allerton Conf. Commun., Control Comput.*, (Monticello, IL), Sept. 28–30, 2005.

[201] X. Bao and J. Li, "Progressive network coding for message-forwarding in ad-hoc wireless networks," in *Proc.* 3$^{\mathrm{rd}}$ *Annual IEEE Commun. Society Sensor Ad Hoc Commun. Netw.*, vol. 1, pp. 207–215, Sept. 28–28, 2006.

[202] X. Bao and J. Li, "A unified channel-network coding treatment for user cooperation in wireless ad-hoc networks," in *Proc. IEEE International Symp. Inf. Theory*, (Seattle, WA, USA), pp. 202–206, July 9–14, 2006.

[203] X. Bao and J. Li, "Adaptive network coded cooperation (ANCC) for wireless relay networks: matching code-on-graph with network-on-graph," *IEEE Trans. Wireless Commun.*, vol. 7, pp. 574–583, Feb. 2008.

[204] X. Bao and J. Li, "On the outage properties of adaptive network coded cooperation (ANCC) in large wireless networks," in *Proc. IEEE International Conf. Acoustics, Speech Signal Process.*, vol. 4, May 14–19, 2006.

[205] X. Bao and J. Li, "An information theoretic analysis for adaptive-network-coded-cooperation (ANCC) in wireless relay networks," in *Proc. IEEE International Symp. Inf. Theory*, (Seattle, WA, USA), pp. 2719–2723, July 9–14, 2006.

[206] R. Ahlswede, C. Ning, S.-Y. R. Li, and R. W. Yeung, "Network information flow," *IEEE Trans. Inf. Theory*, vol. 46, pp. 1204–1216, July 2000.

[207] P. Razaghi and W. Yu, "Bilayer LDPC codes for the relay channel," in *Proc. IEEE International Conf. Commun.*, vol. 4, pp. 1574–1579, June 2006.

[208] A. Chakrabarti, A. de Baynast, A. Sabharwal, and B. Aazhang, "Low density parity check codes for the relay channel," *IEEE J. Sel. Areas Commun.*, vol. 25, pp. 280–291, Feb. 2007.

[209] J. Hu and T. M. Duman, "Low density parity check codes over wireless relay channels," *IEEE Trans. Wireless Commun.*, vol. 6, pp. 3384–3394, Sept. 2007.

[210] H.-K. Lo, T. Spiller, and S. Popescu, *Introduction to Quantum Computation and Information*. World Scientific, 1998.

[211] G. P. Berman, *The Physics of Quantum Information*. Springer, 2000.

[212] G. P. Berman, *Introduction to Quantum Computers*. World Scientific, 1998.

[213] M. Nielson and I. Chuang, *Quantum Computation and Quantum Information*. Cambridge: Cambridge University Press, 2000.

[214] A. R. Calderbank, E. M. Rains, P. W. Shor, and N. J. A. Sloane, "Quantum error correction via codes over $GF(4)$," *IEEE Trans. Inf. Theory*, vol. 44, pp. 1369–1387, July 1998.

[215] P. W. Shor, "Scheme for reducing decoherence in quantum computer memory," *Phys. Rev. A*, vol. 52, pp. 2493–2496, Oct. 1995.

[216] A. Steane, "Multiple particle interference and quantum error correction," in *Proc. Royal Society London - A*, vol. 452, p. 2551, 1996.

[217] A. R. Calderbank and P. W. Shor, "Good quantum error-correcting codes exist," *Phys. Rev. A*, vol. 54, pp. 1098–1105, 1996.

[218] G. Cohen, S. Encheva, and S. Litsyn, "On binary construction of quantum codes," *IEEE Trans. Inf. Theory*, vol. 45, pp. 2495–2498, Nov. 1999.

[219] M. Grassl, W. Geiselmann, and T. Beth, "Quantum Reed Solomon codes," in *Proc.* 16^{th} *International Symp. Applied Algebra, Algebraic Algorithms Error-Correcting* (M. P. C. Fossorier, H. Imai, S. Lin, and A. Poli, eds.), vol. 1719, pp. 231–244, NY: Springer-Verlag, May 24–25, 1999.

[220] A. M. Steane, "Enlargement of Calderbank-Shor-Steane quantum codes," *IEEE Trans. Inf. Theory*, vol. 45, pp. 2492–2495, Nov. 1999.

[221] A. Ashikhmin, S. Litsyn, and M. A. Tsfasman, "Asymptotically good quantum codes," *Phys. Rev. A*, vol. 63, p. 032311, Feb 2001.

[222] H. Chen, S. Ling, and C. Xing, "Asymptotically good quantum codes exceeding the Ashikhmin-Litsyn-Tsfasman bound," *IEEE Trans. Inf. Theory*, vol. 47, pp. 2055–2058, July 2001.

[223] H. Chen, "Some good quantum error-correcting codes from algebraic-geometric codes," *IEEE Trans. Inf. Theory*, vol. 47, pp. 2059–2061, July 2001.

[224] R. Matsumoto, "Improvement of Ashikhmin-Litsyn-Tsfasman bound for quantum codes," *IEEE Trans. Inf. Theory*, vol. 48, pp. 2122–2124, July 2002.

[225] D. J. C. Mackay, G. Mitchison, and P. L. Mcfadden, "Sparse-graph codes for quantum error-correction," *IEEE Trans. Inf. Theory*, vol. 50, pp. 2315–2330, 2003.

[226] T. Camara, H. Ollivier, and J.-P. Tillich, "Constructions and performance of classes of quantum LDPC codes." [Online]. Available: http://arxiv.org/abs/quant-ph/0502086.

[227] M. Hagiwara and H. Imai, "Quantum quasi-cyclic LDPC codes," in *Proc. IEEE International Symp. Inf. Theory*, (Nice, France), pp. 806–810, June 24–29, 2007.

[228] P. Tan and J. Li, "On construction of two classes of efficient quantum error-correction codes," in *Proc. IEEE International Symp. Inf. Theory*, (Nice, France), pp. 2106–2110, June 24–29, 2007.

[229] I. B. Djordjevic, "Quantum LDPC codes from balanced incomplete block designs," *IEEE Commun. Lett.*, vol. 12, pp. 389–391, May 2008.

[230] I. B. Djordjevic, "Photonic quantum dual-containing LDPC encoders and decoders," *Accepted for future publication in IEEE Photonics Technol. Lett.*.

[231] P. Elias, "Coding for two noisy channels," in *Proc. 3rd London Symp. Inf. Theory, Buttersworth's Scientific Publications*, pp. 61–76, Sept. 1955.

[232] F. Taylor, "A single modulus complex ALU for signal processing," *IEEE Trans. Acoustics, Speech Signal Process.*, vol. 33, pp. 1302–1315, Oct. 1985.

[233] H. Krishna, K.-Y. Lin, and J.-D. Sun, "A coding theory approach to error control in redundant residue number systems - part I: Theory and single error correction," *IEEE Trans. Circuits Syst. II: Analog Digital Signal Process.*, vol. 39, pp. 8–17, Jan. 1992.

[234] J.-D. Sun and H. Krishna, "A coding theory approach to error control in redundant residue number systems - part II: Multiple error detection and correction," *IEEE Trans. Circuits Syst. II: Analog Digital Signal Process.*, vol. 39, pp. 18–34, Jan. 1992.

[235] N. Alon, J. Edmonds, and M. G. Luby, "Linear time erasure codes with nearly optimal recovery (extended abstract)," in *IEEE Symp. Foundations Comput. Sci.*, pp. 512–519, 1995.

[236] L. Rizzo, "Effective erasure codes for reliable computer communication protocols," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 27, no. 2, pp. 24–36, 1997.

[237] M. Marcus and H. Minc, *A Survey of Matrix Theory and Matrix Inequalities*. Dover Publications, Apr. 1992.

[238] M. G. Luby, "LT codes," in *Proc. 43rd Annual IEEE Symp. Foundations Comput. Sci.*, pp. 271–280, Nov. 16–19, 2002.

[239] J. Byers, M. G. Luby, M. Mitzenmacher, and A. Rege, "A digital fountain approach to reliable distribution of bulk data," in *Proc. ACM SIGCOMM*, (Vancouver, Canada), Sept. 1998.

[240] J. Byers, M. G. Luby, and M. Mitzenmacher, "A digital fountain approach to asynchronous reliable multicast," *IEEE J. Sel. Areas Commun.*, vol. 20, Oct. 2002.

[241] D. J. C. MacKay, "Fountain codes," in *IEE Proc. Commun.*, vol. 152, pp. 1062–1068, Dec. 9, 2005.

[242] H. Jenkac, T. Mayer, T. Stockhammer, and W. Xu, "Soft decoding of LT-codes for wireless broadcast," in *Proc. IST Mobile Wireless Summit*, June 19–23, 2005.

[243] G. Caire, S. Shamai, M. A. Shokrollahi, and S. Verdu, "Universal variable-length data compression of binary sources using fountain codes," in *Proc. IEEE Inf. Theory Workshop*, (San Antonio, USA), pp. 123–128, Oct.24–29, 2004.

[244] R. Palanki, "Iterative decoding for wireless networks," *PhD Thesis, Caltech*, May 2005.

[245] R. Palanki and J. S. Yedidia, "Rateless codes on noisy channels," in *Proc. IEEE International Symp. Inf. Theory*, (Chicago, IL, USA), p. 37, June 27–July 2, 2004.

[246] R. Y. S. Tee, T. D. Nguyen, L. L. Yang, and L. Hanzo, "Serially concatenated Luby transform coding and bit-interleaved coded modulation using iterative decoding for the wireless Internet," in *Proc. IEEE* 63^{rd} *Veh. Technol. Conf.*, vol. 1, pp. 22–26, 2006.

[247] T. D. Nguyen, F. C. Kuo, L. L. Yang, and L. Hanzo, "Amalgamated generalized low-density parity-check and Luby transform codes for the wireless Internet," in *Proc. IEEE Veh. Technol. Conf.*, (Dublin, Ireland), pp. 2440–2444, Apr. 22–25, 2007.

[248] N. Dütsch, H. Jenkac, T. Mayer, and J. Hagenauer, "Joint source-channel-fountain coding for asynchronous broadcast," in *Proc. IST Mobile Wireless Summit*, June 19–23, 2005.

[249] H. Jenkac, J. Hagenauer, and T. Mayer, "The turbo-fountain," *European Trans. Telecommun. (ETT), Special Issue on "Next Generation Wireless Mobile Commun."*, vol. 17, pp. 337–349, May 2006.

[250] M. A. Shokrollahi, "Raptor codes," in *Proc. IEEE International Symp. Inf. Theory*, p. 36, 2004.

[251] M. A. Shokrollahi, "Raptor codes," *IEEE Trans. Inf. Theory*, vol. 52, pp. 2551–2567, June 2006.

[252] O. Etesami, M. Molkaraie, and M. A. Shokrollahi, "Raptor codes on symmetric channels," in *Proc. IEEE International Symp. Inf. Theory*, (Chicago, IL, USA), p. 38, June 27–July 2, 2004.

[253] J. Castura and Y. Mao, "Rateless coding over fading channels," *IEEE Commun. Lett.*, vol. 10, pp. 46–48, Jan. 2006.

[254] J. Castura, Y. Mao, and S. Draper, "On Rateless Coding over Fading Channels with Delay Constraints," in *Proc. IEEE International Symp. Inf. Theory*, (Seattle, WA, USA), pp. 1124–1128, July 9–14, 2006.

[255] Z. Yang and A. Host-Madsen, "Rateless coded cooperation for multiple-access channels in the low power regime," in *Proc. IEEE International Symp. Inf. Theory*, (Seattle, WA, USA), pp. 967–971, July 9–14, 2006.

[256] K. Hu, J. Castura, and Y. Mao, "Performance-complexity tradeoffs of Raptor codes over Gaussian channels," *IEEE Commun. Lett.*, vol. 11, pp. 343–345, Apr. 2007.

[257] C. Lee and W. Gao, "Rateless-coded hybrid ARQ," in *Proc.* 6$^{\text{th}}$ *International Conf. Inf., Commun. Signal Process.*, (Singapore), pp. 1–5, Dec. 10–13, 2007.

[258] J. Castura and Y. Mao, "A rateless coding and modulation scheme for unknown Gaussian channels," in *Proc.* 10$^{\text{th}}$ *Canadian Workshop Inf. Theory*, (Edmonton, Alta.), pp. 148–151, June 6–8, 2007.

[259] J. Castura and Y. Mao, "Rateless coding for wireless relay channels," in *Proc. IEEE International Symp. Inf. Theory*, (Adelaide, SA), pp. 810–814, Sept. 4–9, 2005.

[260] J. Castura and Y. Mao, "Rateless coding for wireless relay channels," *IEEE Trans. Wireless Commun.*, vol. 6, pp. 1638–1642, May 2007.

[261] J. Castura and Y. Mao, "Rateless coding and relay networks," *IEEE Signal Process. Mag.*, vol. 24, pp. 27–35, Sept. 2007.

[262] P. Cataldi, M. P. Shatarski, M. Grangetto, and E. Magli, "Implementation and performance evaluation of LT and Raptor codes for multimedia applications," in *Proc. International Conf. Intelligent Inf. Hiding Multimedia Signal Process.*, (Pasadena, CA, USA), pp. 263–266, Dec. 2006.

[263] M. G. Luby, T. Gasiba, T. Stockhammer, and M. Watson, "Reliable multimedia download delivery in cellular broadcast networks," *IEEE Trans. Broadcasting*, vol. 53, pp. 235–246, Mar. 2007.

[264] Q. Xu, V. Stankovic, and Z. Xiong, "Distributed joint source-channel coding of video using raptor codes," *IEEE J. Sel. Areas Commun.*, vol. 25, pp. 851–861, May 2007.

[265] N. Rahnavard, B. N. Vellambi, and F. Fekri, "Rateless codes with unequal error protection property," *IEEE Trans. Inf. Theory*, vol. 53, pp. 1521–1532, Apr. 2007.

[266] N. Rahnavard, B. N. Vellambi, and F. Fekri, "A fractional transmission scheme for efficient broadcasting via rateless coding in multi-hop wireless networks," in *Proc. IEEE Military Commun. Conf.*, (Orlando, FL, USA), pp. 1–7, Oct. 29–31, 2007.

[267] T. Schierl, S. Johansen, C. Hellge, T. Stockhammer, and T. Wiegand, "Distributed rate-distortion optimization for rateless coded scalable video in mobile ad hoc networks," in *Proc. IEEE International Conf. Image Process.*, vol. 6, (San Antonio, TX, USA), pp. 497–500, Sept. 16–Oct. 2007, 2007.

[268] T. D. Nguyen, L. L. Yang, and L. Hanzo, "Systematic Luby transform codes and their soft decoding," in *Proc. IEEE Workshop Signal Process. Syst.*, (Shanghai, China), pp. 67–72, Oct. 17–19, 2007.

[269] S. Argyropoulos, A. S. Tan, N. Thomos, E. Arikan, and M. G. Strintzis, "Robust transmission of multi-view video streams using flexible macro block ordering and systematic LT codes," in *Proc. 3DTV Conf.*, (Kos, Greece), pp. 1–4, May 7–9, 2007.

[270] A. W. Eckford, J. P. K. Chu, and R. S. Adve, "Low-complexity cooperative coding for sensor networks using rateless and LDGM codes," in *Proc. IEEE International Conf. Commun.*, vol. 4, pp. 1537–1542, June 2006.

[271] X. Yuan and L. Ping, "On systematic LT codes," *IEEE Commun. Lett.*, vol. 12, pp. 681–683, Sept. 2008.

[272] P. Maymounkov, "Online codes," Technical Report TR2002-833, New York University, NY, Nov. 2002. [Online]. Available: http://pdos.csail.mit.edu/ petar/papers/maymounkov-online.pdf.

[273] P. Maymounkov and D. Mazieres, "Rateless codes and big downloads," in *Proc.* 2$^{\text{nd}}$ *International Workshop Peer-to-Peer Syst.*, (Berkeley, CA, USA), Feb. 20–21, 2003.

[274] T. Eriksson and N. Goertz, "Rateless codes based on linear congruential recursions," *Electron. Lett.*, vol. 43, pp. 402–404, Mar. 29, 2007.

[275] A. W. Eckford and W. Yu, "Density evolution for the simultaneous decoding of LDPC-based Slepian-Wolf source codes," in *Proc. IEEE International Symp. Inf. Theory*, (Adelaide, SA), pp. 1401–1405, Sept. 4–9, 2005.

[276] A. W. Eckford and W. Yu, "Rateless Slepian-Wolf codes," in *Proc.* 39$^{\text{th}}$ *Asilomar Conf. Signals, Syst. Comput.*, pp. 1757–1761, Oct. 28–Nov. 1, 2005.

[277] D. Slepian and J. Wolf, "Noiseless coding of correlated information sources," *IEEE Trans. Inf. Theory*, vol. 19, pp. 471–480, July 1973.

[278] J. J. Metzner, "An improved broadcast retransmission protocol," *IEEE Trans. Commun.*, vol. 32, pp. 679–683, June 1984.

[279] S. Sesia, G. Caire, and G. Vivier, "Incremental redundancy hybrid ARQ schemes based on low-density parity-check codes," *IEEE Trans. Commun.*, vol. 52, pp. 1311–1321, Aug. 2004.

[280] M. Good and F. R. Kschischang, "Incremental redundancy via check splitting," in *Proc.* 23$^{\text{rd}}$ *Biennial Symp. Commun.*, (Kingston, Canada), pp. 55–58, May 29–June 1, 2006.

[281] R. Liu, P. Spasojevic, and E. Soljanin, "Incremental redundancy cooperative coding for wireless networks: Cooperative diversity, coding, and transmission energy gains," *IEEE Trans. Inf. Theory*, vol. 54, pp. 1207–1224, Mar. 2008.

[282] D. J. Costello Jr., J. Hagenauer, H. Imai, and S. B. Wicker, "Applications of error-control coding," *IEEE Trans. Inf. Theory*, vol. 44, pp. 2531–2560, Oct. 1998.

[283] J. Hámorsky and L. Hanzo, "Performance of the turbo hybrid automatic repeat request system type II," in *Proc. IEEE Inf. Technical Workshop*, (Metsovo, Greece), p. 51, June 27–July 1, 1999.

[284] H. Y. Park, J. W. Kang, K. S. Kim, and K. C. Whang, "Efficient puncturing method for rate-compatible low-density parity-check codes," *IEEE Trans. Wireless Commun.*, vol. 6, pp. 3914–3919, Nov. 2007.

[285] J. Ha, J. Kim, D. Klinc, and S. W. McLaughlin, "Rate-compatible punctured low-density parity-check codes with short block lengths," *IEEE Trans. Inf. Theory*, vol. 52, pp. 728–738, Feb. 2006.

[286] H. Pishro-Nik and F. Fekri, "Results on punctured low-density parity-check codes and improved iterative decoding techniques," *IEEE Trans. Inf. Theory*, vol. 53, pp. 599–614, Feb. 2007.

[287] G. Yue, X. Wang, and M. Madihian, "Design of rate-compatible irregular repeat-accumulate codes," *IEEE Trans. Commun.*, vol. 55, pp. 1153–1163, June 2007.

[288] E. Soljanin, N. Varnica, and P. Whiting, "Punctured vs rateless codes for hybrid ARQ," in *Proc. IEEE Inf. Theory Workshop*, (Punta del Este, Uruguay), pp. 155–159, 2006.

[289] E. Soljanin, N. Varnica, and P. Whiting, "Incremental redundancy hybrid ARQ with LDPC codes and raptor codes," *submitted to IEEE Trans. Inf. Theory*, Sept. 2005.

[290] O. Etesami and M. A. Shokrollahi, "Raptor codes on binary memoryless symmetric channels," *IEEE Trans. Inf. Theory*, vol. 52, pp. 2033–2051, May 2006.

[291] N. Bonello, R. Zhang, S. Chen, and L. Hanzo, "Reconfigurable rateless codes." *accepted in the IEEE Trans. Wireless Commun.*.

[292] N. Bonello, D. Yang, S. Chen, and L. Hanzo, "Generalized MIMO transmit preprocessing using pilot symbol assisted rateless codes." *accepted in the IEEE Trans. Wireless Commun.*.

[293] A. F. Molisch, N. B. Mehta, J. S. Yedidia, and J. Zhang, "Cooperative relay networks using fountain codes," in *Proc. IEEE Global Telecommun. Conf.*, pp. 1–6, Nov. 2006.

[294] A. F. Molisch, N. B. Mehta, J. S. Yedidia, and J. Zhang, "Performance of fountain codes in collaborative relay networks," *IEEE Trans. Wireless Commun.*, vol. 6, pp. 4108–4119, Nov. 2007.

[295] J. Garcia-Frias and W. Zhong, "Approaching Shannon performance by iterative decoding of linear codes with low-density generator matrix," *IEEE Commun. Lett.*, vol. 7, pp. 266–268, June 2003.

[296] R. Koetter and P. O. Vontobel, "Graph covers and iterative decoding of finite-length codes," in *Proc.* 3rd *International Symp. Turbo Codes Related Topics*, (Brest, France), pp. 75–82, Sept. 2003.

[297] S.-T. Xia and F.-W. Fu, "Minimum pseudoweight and minimum pseudocodewords of LDPC codes," *IEEE Trans. Inf. Theory*, vol. 54, pp. 480–485, Jan. 2008.

[298] V. Chernyak, M. Chertkov, M. G. Stepanov, and B. Vasic, "Error correction on a tree: an instanton approach," *Physics Rev. Lett.*, vol. 93, Nov. 2004. id. 198702.

[299] M. Stepanov and M. Chertkov, "Instanton analysis of low- density parity-check codes in the error-floor regime," in *Proc. IEEE International Symp. Inf. Theory*, (Nice, France), pp. 552–556, June 24–27 2006.

[300] L. Dolecek, Z. Zhang, V. Anantharam, M. Wainwright, and B. Nikolic, "Analysis of absorbing sets for array-based LDPC codes," in *Proc. IEEE International Conf. Commun.*, (Glasgow, Scotland), pp. 6261–6268, June 24–28 2007.

[301] J. W. Lee and R. E. Blahut, "A note on the analysis of finite length turbo decoding," in *Proc. IEEE International Symp. Inf. Theory*, (Lausanne, Switzerland), p. 83, June 30–July 5, 2002.

[302] J. W. Lee and R. E. Blahut, "Lower bound on BER of finite-length turbo codes based on EXIT characteristics," *IEEE Commun. Lett.*, vol. 8, pp. 238–240, Apr. 2004.

[303] J. W. Lee and R. E. Blahut, "Convergence analysis and BER performance of finite-length turbo codes," *IEEE Trans. Commun.*, vol. 55, pp. 1033–1043, May 2007.

[304] M. Tüchler, "Design of serially concatenated systems depending on the block length," *IEEE Trans. Commun.*, vol. 52, pp. 209–218, Feb. 2004.

[305] A. Amraoui, R. L. Urbanke, and A. Montanari, "Finite-length scaling of irregular LDPC code ensembles," in *Proc. IEEE Inf. Theory Workshop*, Aug. 29–Sept. 1, 2005.

[306] T. J. Richardson, M. A. Shokrollahi, and R. L. Urbanke, “Finite-length analysis of various low-density parity-check ensembles for the binary erasure channel,” in *Proc. IEEE International Symp. Inf. Theory*, (Lausanne, Switzerland), p. 1, June 30–July 5, 2002.

[307] A. Amraoui, R. L. Urbanke, A. Montanari, and T. J. Richardson, “Further results on finite-length scaling for iteratively decoded LDPC ensembles,” in *Proc. IEEE International Symp. Inf. Theory*, (Chicago, IL USA), p. 103, June 27–July 2, 2004.

[308] W. Yu, M. Ardakani, B. Smith, and F. Kschischang, “Complexity-optimized low-density parity-check codes for Gallager decoding algorithm B,” in *Proc. International Symp. Inf. Theory*, (Adelaide, SA), pp. 1488–1492, Sept. 2005.

[309] L. Junhai, Y. Danxia, X. Liu, and F. Mingyu, “A survey of multicast routing protocols for mobile ad-hoc networks,” *IEEE Commun. Surveys Tuts.*, vol. 11, no. 1, pp. 78–91, 2009.

[310] T. L. Willke, P. Tientrakool, and N. F. Maxemchuk, “A survey of inter-vehicle communication protocols and their applications,” *IEEE Commun. Surveys Tuts.*, vol. 11, no. 2, pp. 3–20, 2009.

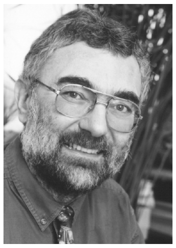
[311] M. Chamania and A. Jukan, “A survey of inter-domain peering and provisioning solutions for the next generation optical networks,” *IEEE Commun. Surveys Tuts.*, vol. 11, no. 1, pp. 33–51, 2009.

[312] T. Yücek and H. Arslan, “A survey of spectrum sensing algorithms for cognitive radio applications,” *IEEE Commun. Surveys Tuts.*, vol. 11, no. 1, pp. 116–130, 2009.

[313] R. Wang, T. Taleb, A. Jamalipour, and B. Sun, “Protocols for reliable data transport in space internet,” *IEEE Commun. Surveys Tuts.*, vol. 11, no. 2, pp. 21–32, 2009.

Nicholas Bonello received the B.Eng.(Hons) degree in Electrical Engineering from the University of Malta in 2004, the M.Sc. degree in Radio Frequency Communications Systems from University of Southampton, U.K., in 2006 and the Ph.D. degree in Wireless Communications from the University of Southampton, U.K, in 2009. His primary research interests include fixed-rate as well as rateless error correction techniques, signal processing and statistics.

Sheng Chen (M’90-SM’97-F’08) obtained a BEng degree from the East China Petroleum Institute, Dongying, China, in 1982, and a PhD degree from the City University, London, in 1986, both in control engineering. In 2005, he was awarded DSc from the University of Southampton, Southampton, UK. Since 1999 he has been with the School of Electronics and Computer Science, the University of Southampton, UK. He previously held research and academic appointments at the Universities of Sheffield, Edinburgh and Portsmouth, all in UK. Professor Chen’s recent research works include adaptive signal processing, wireless communications, modelling and identification of nonlinear systems, neural network and machine learning, finite-precision digital controller design, evolutionary computation methods, and optimization. He has published over 280 research papers. In the database of the world’s most highly cited researchers in various disciplines, compiled by Institute for Scientific Information (ISI) of the USA, Prof. Chen is on the list of the highly cited researchers in the engineering category, see http://www.ISIHighlyCited.com.



Lajos Hanzo FREng, FIEEE, FIET, DSc received his degree in electronics in 1976 and his doctorate in 1983. During his 31-year career in telecommunications he has held various research and academic posts in Hungary, Germany and the UK. Since 1986 he has been with the School of Electronics and Computer Science, University of Southampton, UK, where he holds the chair in telecommunications. He has co-authored 17 books on mobile radio communications totalling in excess of 10 000 pages, published in excess of 800 research papers, acted as TPC Chair of IEEE conferences, presented keynote lectures and been awarded a number of distinctions. Currently he is directing an academic research team, working on a range of research projects in the field of wireless multimedia communications sponsored by industry, the Engineering and Physical Sciences Research Council (EPSRC) UK, the European IST Programme and the Mobile Virtual Centre of Excellence (VCE), UK. He is an enthusiastic supporter of industrial and academic liaison and he offers a range of industrial courses. He is also an IEEE Distinguished Lecturer as well as a Governor of both the IEEE ComSoc and the VTS. He is the acting Editor-in-Chief of the IEEE Press. For further information on research in progress and associated publications please refer to http://www-mobile.ecs.soton.ac.uk.