Near-Capacity Coding for Discrete Multitone Systems with Impulse Noise

Masoud Ardakani,[1] Frank R. Kschischang,[2] and Wei Yu[2]

[1] *Department of Electrical and Computer Engineering, University of Alberta, ECERF Building, Edmonton, AB, Canada, T6G 2V4*

[2] *Department of Electrica and Computer Engineering, University of Toronto, 10 King's College Road, Toronto, ON, Canada M5S 3G4*

We consider the design of near-capacity-achieving error-correcting codes for a discrete multitone (DMT) system in the presence of both additive white Gaussian noise and impulse noise. Impulse noise is one of the main channel impairments for digital subscriber lines (DSL). One way to combat impulse noise is to detect the presence of the impulses and to declare an erasure when an impulse occurs. In this paper, we propose a coding system based on low-density parity-check (LDPC) codes and bit-interleaved coded modulation that is capable of taking advantage of the knowledge of erasures. We show that by carefully choosing the degree distribution of an irregular LDPC code, both the additive noise and the erasures can be handled by a single code, thus eliminating the need for an outer code. Such a system can perform close to the capacity of the channel and for the same redundancy is significantly more immune to the impulse noise than existing methods based on an outer Reed-Solomon (RS) code. The proposed method has a lower implementation complexity than the concatenated coding approach.

1. INTRODUCTION

The design of error control codes for discrete multitone (DMT) systems is of great interest for applications such as digital subscriber lines (DSL) [1–6]. In a DMT system, different constellations may be used in different tones and nonbinary high-order modulation formats are typically used. Current asymmetric DSL (ADSL) standards use a trellis-coded modulation scheme concatenated with an outer Reed-Solomon (RS) code. The inner trellis code provides a coding gain for an additive white Gaussian noise (AWGN) channel, and the outer RS code offers additional noise immunity, especially against impulse noise.

This paper is motivated by the phenomenal success of turbo and low-density parity-check (LDPC) codes in the past decade. It is now possible to design codes that perform within a fraction of a decibel (dB) from the Shannon limit in an additive white Gaussian noise channel. However, the use of turbo codes and LDPC codes in DMT systems is not yet widespread. This is in part due to the fact that the effect of impulse noise on turbo or LDPC codes has not yet been studied in depth. Impulse noise is one of the main channel impairments in DSL. The main focus of this paper is the design of LDPC codes for a DMT system in an impulse-noise environment.

In their original forms, both turbo and LDPC codes are binary codes. In the low signal-to-noise-ratio (SNR) regime, where binary modulation is spectrally efficient, binary turbo codes and binary low-density parity-check (LDPC) codes can effectively approach the capacity of many channels (e.g., [7–11]). A DMT system, however, often includes higher-order constellations that are not necessarily binary. This necessitates the use of nonbinary alphabets or the use of multilevel coding techniques. Theoretically, multilevel coding [12], when combined with capacity-achieving binary codes, is capable of achieving the capacity of higher-order modulation. In practice, schemes such as bit-interleaved coded modulation (BICM) [13] can offer a performance very close to the capacity of higher-order modulation.

The use of multilevel coding for DMT systems has been studied by various authors for DSL and power-line communication channels [2, 4–6]. For example, in [5] a regular high-rate LDPC code is used for error correction in a DSL transmission system, achieving a coding gain of 6 dB at a symbol error rate (SER) of 10^{-7}. In [2], turbo codes have been used for ADSL and a coding gain of 6 dB at an SER of 10^{-6} (equivalent to approximately 6.8 dB at an SER of 10^{-7}) is reported. In [6], the idea of using LDPC codes together with coded modulation is used, although the issue of code optimization is not addressed. In [4], irregular LDPC

codes—optimized for Gaussian channels—are used for data transmission over power-line channels and an average coding gain of about 7.5 dB at an SER of 10^{-7} is achieved. These coding gains can be compared with a 5.5 dB gain which can be achieved by a 512-state trellis code on an AWGN channel at an SER of 10^{-7} [14].

The above-mentioned systems all assume an additive Gaussian noise model. The effect of impulsive noise is either ignored, or not considered in the code design. However, impulse noise is a major channel impairment in digital subscriber lines. A large impulse often causes an entire DMT symbol to be corrupted [15] which can seriously deteriorate overall system performance. Current DSL standards rely on extensive interleaving and an outer RS code to provide impulse protection. Much work has been done (e.g., see [15–20] and the references therein) on the characteristics of impulse noise and the use of RS codes for impulse protection.

One way to combat the effect of impulse noise is the idea of impulse detection and erasure. As an impulse is typically a large signal that occurs for an extended period of time and across a large number of frequency tones, reliable detection of impulses is often possible. When an impulse is detected, an erasure can then be declared. The declaration of erasure provides side information to the decoder of the error-correcting code and effectively increases its error-correcting capability. For example, in a system with an RS code, the error-correcting capability can be increased by a factor of two when erasures are declared [18, 19]. Under the erasure model, the impulse channel is essentially a concatenation of an additive noise channel and an erasure channel.

LDPC codes have excellent performance on both additive noise channels (e.g., AWGN) and the binary erasure channel (BEC) alone. However, the design and performance of these codes over channels with both additive noise and erasures have not—to the best of our knowledge—been studied. This motivates us to study the design and the analysis of LDPC codes over a channel model seen in DMT systems in impulsive environments.

In this study, we rigorously analyze an impulse-noise channel model and design an irregular LDPC code specifically for the impulse noise channel. Our main contribution is a design methodology that enables the design of capacity-approaching LDPC codes for channels with both additive Gaussian noise and erasures. Such an optimized LDPC code, when combined with bit-interleaved coded modulation, outperforms both conventional systems with RS codes and systems with LDPC codes optimized for AWGN channel alone in a DMT system. In fact, such a design can effectively handle both the additive Gaussian noise and the erasures introduced by the DMT channel, thus eliminating the need for an outer code. This results in hardware—and software—complexity savings as well as higher code rates.

The rest of the paper is organized as follows. In Section 2, the impulse-noise channel model is introduced. In Section 3, a review of multilevel coding and density evolution analysis for LDPC codes is provided. In Section 4, a coding structure based on BICM and LDPC coding is established; the characteristics of the effective "bit-channel" are studied; and
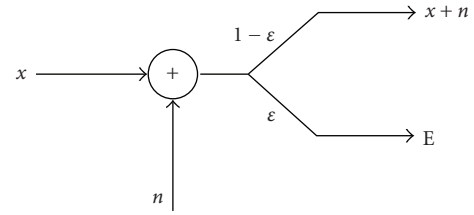

FIGURE 1: Channel model: the concatenation of an AWGN channel with an erasure channel.

an LDPC code optimization technique is described. The optimization technique is capable of finding LDPC codes that perform close to the capacity of the effective "bit-channel." Section 5 provides simulation results and comparison with existing solutions. Finally, conclusions are drawn in Section 6.

2. CHANNEL MODEL AND PROBLEM STATEMENT

Throughout this work we assume that the DMT symbols that are corrupted by impulsive noise are known to the receiver, and can therefore be replaced by erasures. In [19] a process called erasing is used to detect the impulse noise and to indicate the location of the potentially corrupted data in an RS block. This way, the error-correcting capability of the RS code is increased by a factor of two. There are many different methods for detecting the impulse noise at the receiver, for example, [18–20]. The detection of impulse noise is usually based on the observation that when a noise burst occurs, the squared distance between several of the received signals in a DMT symbol and the constellation points exceeds a threshold. Therefore, using square distance-based methods, the impulsive noise can be replaced with an erasure in the channel.

Figure 1 shows the channel model that we consider in this study. The input to this channel, x, is a point selected from a constellation $A \subset \mathbb{C}$ of size 2^l. The output, $y \in \mathbb{C} \cup \{E\}$, of this channel is either erasure E with probability ϵ or $x + n$ with probability $1 - \epsilon$, where n is assumed to be a sample of a complex zero-mean circularly-symmetric white Gaussian noise with variance $2\sigma^2$. The erasure output of this channel model reflects the effect of the impulse noise and the additive noise reflects the effect of the AWGN channel.

Erasures can be naturally incorporated in the message-passing decoding of LDPC codes (see Section 3.2). However, it is not clear that a code optimized for an AWGN channel would also remain optimized for a channel with both AWGN and erasures. One of the objectives of this paper is to design LDPC codes specifically for such channels.

Since in this work we use multilevel coding, the transmission of the signal will be modelled as the transmission of the bits of the binary labels used for labelling different constellation points. Thus, we are most interested in the characteristics of the channels seen by the label bits. We call these channels bit-channels, and will study them after a brief review of multilevel coding and a description of our proposed system.

3. BACKGROUND

In this section we briefly review the idea of multilevel coding, the structure of LDPC codes, and their analysis techniques.

3.1. Multilevel coding

The main idea of multilevel coding [12] is to label each point of a nonbinary constellation $\mathbf{A} = \{a_0, a_1, \ldots, a_{N-1}\}$ of size $N = 2^l$ with a binary address $\mathbf{b} = (b_0, b_1, \ldots, b_{l-1})$ and use binary codes to protect each address bit b_i by an individual binary code $\mathcal{C}^i$ at level i [21].

Let A and Y represent the random variables corresponding to the transmitted and received signals, respectively. Also consider $(b_0, b_1, \ldots, b_{l-1})$ as the vector of random variables corresponding to the address bits. Since a one-to-one mapping between the address vector and the constellation points exists, we have

$$I(Y; A) = I(Y; b_0, b_1, \ldots, b_{l-1}) \qquad (1)$$

which, using the mutual information chain rule, can be written as

$$\begin{aligned} I(Y; A) = I(Y; b_0) + I(Y; b_1 \mid b_0) \\ + \cdots + I(Y; b_{l-1} \mid b_0, b_1, \ldots, b_{l-2}). \end{aligned} \qquad (2)$$

This decomposition of the mutual information implies that transmission of the signal over the physical channel can be thought of as parallel transmission of individual bits. Due to the labelling scheme, each address bit sees an effective channel (bit-channel) whose capacity can be different from the capacity of other bit-channels. An individual code should be used for each bit-channel, and it is known that if each code achieves the capacity of its bit-channel, the total capacity of the channel is achieved [21]. A complete study of multilevel coding and the structure of the encoder and decoder can be found in [22, 23].

One drawback of capacity-approaching systems using multilevel coding is that they require different codes with different rates for each bit-channel. This problem can be solved using BICM. In BICM, after using gray labelling, the address bits of the labels are interleaved. Hence, instead of using individual codes at different levels, a single code can be used for all the address bits. BICM can approach the channel capacity very closely [13], but, unlike multilevel coding, cannot achieve capacity. What makes BICM attractive is its lower software/hardware complexity at the expense of very minor performance loss.

3.2. LDPC codes

An LDPC code is a linear block code with a sparse parity-check matrix that can be represented by a bipartite graph [24, 25]. Figure 2 shows one such graphical representation.

A variable node v_j (represented by a circle) is a binary variable from the alphabet $\{0, 1\}$ and a check node c_i (represented by a square) is an even parity constraint on its neighboring variable nodes, so that in a valid codeword we
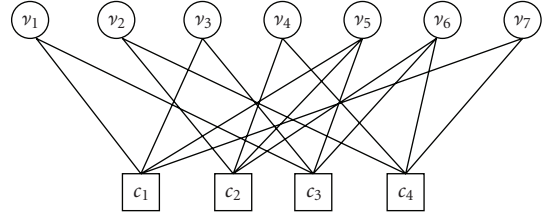

Figure 2: A bipartite graph representing a parity-check code.

have, for all the checks c_i,

$$\bigoplus_{j: v_j \in n(c_i)} v_j = 0, \qquad (3)$$

where $n(c_i)$ is the set of all the neighbors of c_i and $\oplus$ represents modulo-two sum.

LDPC codes, depending on their structure, are classified as being *regular* or *irregular*. In a regular code, all variable nodes are of a fixed degree and all the check nodes are of another fixed degree. Irregular LDPC codes can significantly outperform regular codes [26]. By carefully designing the irregularity in the graph, codes which perform very close to the capacity can be found [9, 11, 26, 27].

An ensemble of irregular codes can be defined by its variable and check-node-degree distributions. The degree distribution at the variable/check side represents the percentage of edges incident on variable/check nodes with different degrees. As the block length of a randomly chosen code approaches infinity, the performance of the code depends only on its degree distributions [9]; therefore, code design problem translates to the problem of finding good variable and check-degree distributions. In practice, when a finite-length code is used, the larger the block length, the closer the performance of the code to the predicted asymptotic behavior. In the remainder of this paper, we refer to variable-node-degree distribution as $\lambda = \{\lambda_2, \lambda_3, \ldots, \lambda_{d_v}\}$, where λ_i is the fraction of edges incident on variable nodes of degree i and d_v is the maximum variable-node degree of the code. Similarly we use $\rho = \{\rho_2, \rho_3, \ldots, \rho_{d_c}\}$ for the check-degree distribution.

The decoding of LDPC codes is done by iterative message-passing algorithms. Different types of messages exist, but log-likelihood-ratio (LLR) messages are the most common ones. The LLR for a bit b is defined as

$$\text{LLR}(b) = \ln \frac{\Pr(b = 0)}{\Pr(b = 1)}. \qquad (4)$$

Notice that in the case of an erasure, $\text{LLR}(b) = 0$.

Many different message-passing algorithms exist [9, 25], among which the sum-product algorithm is the most accurate one [25]. Assuming that LLR messages are used, and following the notation of [25], the update rules of the sum-product algorithm at a check node c and a neighboring
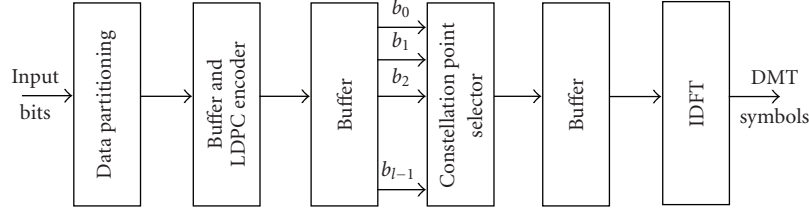
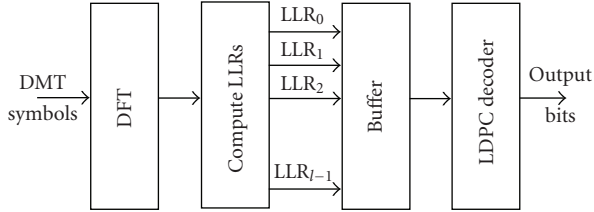FIGURE 3: The block diagram of the encoder of the proposed system.

FIGURE 4: The block diagram of the decoder of the proposed system.

variable node v are

$$\mu_{c \to v} = 2\tanh^{-1}\left(\prod_{h \in n(c) - \{v\}} \tanh\frac{\mu_{h \to c}}{2}\right),$$

$$\mu_{v \to c} = \mu_{ch \to v} + \sum_{y \in n(v) - \{c\}} \mu_{y \to v}, \qquad (5)$$

where $\mu_{h \to c}$ represents a message from a variable node h to the check node c, $\mu_{y \to v}$ represents a message from a check node y to the variable node v, and $\mu_{ch \to v}$ is the channel message to the variable node v. In this paper, we assume that our LDPC decoder uses the sum-product algorithm.

Erasure LLR messages are zero and are processed like other messages in the decoder. Hence, they are naturally incorporated in the decoding, requiring no modification to the decoder.

3.3. LDPC code analysis and design

The analysis of the decoder's performance can be done by density evolution [10] or discrete density evolution [27]. The main idea of the analysis is to track the evolution of the probability density function (PDF) of the messages in the decoder. The inputs to this algorithm are the code variable-node- and check-node-degree distributions, the update rules at the variable and check nodes, and the PDF of LLR messages from the channel. The output of the algorithm is the PDF of LLR messages after each iteration.

One of the pillars of density evolution analysis of LDPC codes for binary transmission is the assumption that the all-zero codeword (all-$\{+1\}$ channel-word) is transmitted. This assumption is valid because the performance of the code is independent of the transmitted codeword [10]. Therefore, under this assumption we can associate an error rate to a message PDF, that is, the negative tail of the PDF. Thus, one can investigate whether or not the error rate of messages (the

negative tail of the LLR density) approaches zero as the number of iterations goes to infinity.

For higher-order modulation, however, assuming that the all-zero codeword is transmitted is not valid. This is because, different constellation points might have different error rates. In Section 4.3 we will show this problem can be tackled and an LLR density whose negative tail corresponds to the message error rate can be found, without the need for an all-zero codeword assumption.

Using density evolution as a probe, one can search for variable-node- and check-node-degree distributions that provide performance approaching capacity [27]. The inputs to this optimization program are the message update rules of the decoder at the variable and check nodes as well as the channel LLR PDF and the output is the optimized degree-distribution.

There are many papers on such degree optimizations for various channel models. Probably the one most related to our study is [28], where component LDPC codes are designed in a multilevel coding structure. Our work is different in two ways: first, we use BICM and hence we need to design one single LDPC code; second, we consider the effect of impulse noise. Using one LDPC code has many practical benefits, for example, for the same delay, it allows for a longer block-length (hence better performance); it also reduces hardware and software complexity.

4. SYSTEM STRUCTURE

Figures 3 and 4 show the block diagrams of the proposed system at the transmitter and the receiver, respectively. At the transmitter, the constellation mapper uses l bits from the output of LDPC encoder to select a point from the constellation for each subchannel of the DMT system. After assigning data to all subchannels, this complex vector is mapped to a DMT symbol using an inverse Fourier transform (IDFT).

At the receiver, a DMT symbol is first converted to a complex vector using the discrete Fourier transform (DFT). Then, for each tone, l LLR values associated with the l transmitted bits are computed and buffered. Once all LLR values are ready, they are passed to the LDPC decoder.

Here, we assume that the length of the LDPC code is an integer multiple of the length of the DMT symbols. Even if we relax this assumption (due to the time-varying number of bits loaded into the DMT symbols), we can simply put as many DMT symbols as possible in the LDPC structure and fill its remaining portion with zeros. Typically, the

length of an LDPC codeword is much longer than the length of a DMT symbol. Thus, filling the remaining portion of the LDPC code with zeros has no significant effect on the system performance.

4.1. Channel capacity

The physical channel model for a DMT system in an impulse environment is as shown in Figure 1. In this section, we consider instead the transmission of the bits of the constellation labels and study the resulting bit-channel. Since the binary LDPC decoder receives its initial LLRs from these bit-channels, we first investigate the distribution of the LLRs of the bit-channels. We then characterize the bit-channel capacity, which provides a metric with which the actual achieved code rates may be compared.

As shown in [29], the mutual information between the input and the output of an AWGN channel, when the input signals are equally likely chosen from a constellation A of size 2^l, is

$$C_A = \log_2\left(2^l\right)$$
$$- \frac{1}{2^l}\sum_{k=0}^{2^l-1}\mathbb{E}\left\{\log_2\sum_{i=0}^{2^l-1}\exp\left[-\frac{|a_k+w-a_i|^2-|w|^2}{2\sigma^2}\right]\right\},$$
(6)

where a_k is the kth point on the constellation, and w represents samples of a complex Gaussian noise with variance $2\sigma^2$.

Our channel model is a concatenation of such an AWGN channel with an erasure channel whose erasure rate is ϵ. Therefore the channel capacity is

$$C_{A,\epsilon} = (1-\epsilon)C_A. \tag{7}$$

This capacity should be normalized per bit to reflect the average capacity of bit-channels. Thus, (7) results in bit-channels whose average capacity C_b is

$$C_b = \frac{1}{l}(1-\epsilon)C_A. \tag{8}$$

In this work we use BICM. Although, BICM cannot achieve the capacity (8), it can approach the capacity very closely, as shown in [13]. At high SNRs the capacities of BICM and multilevel coding are almost equal.

Figure 5 shows the plot of C_b versus SNR. It also shows the capacity of BICM. It can be seen that for SNRs exceeding 6 dB, the gap from the capacity of BICM to the capacity of multilevel coding is minor. It should be mentioned that here SNR is defined as $10\log_{10}(E_s/\sigma^2)$, where E_s is the average energy of constellation per real dimension and $2\sigma^2$ is the variance of the complex Gaussian noise. The erasure rate of the channel is assumed to be 0.06, hence at high SNR the value of C_b approaches $1 - \epsilon = 0.94$.

4.2. Density of bit-channel LLRs

While an erasure in the actual channel translates to an erasure in all bit-channels, a Gaussian noise on the actual channel does not necessarily translate to Gaussian noise on the
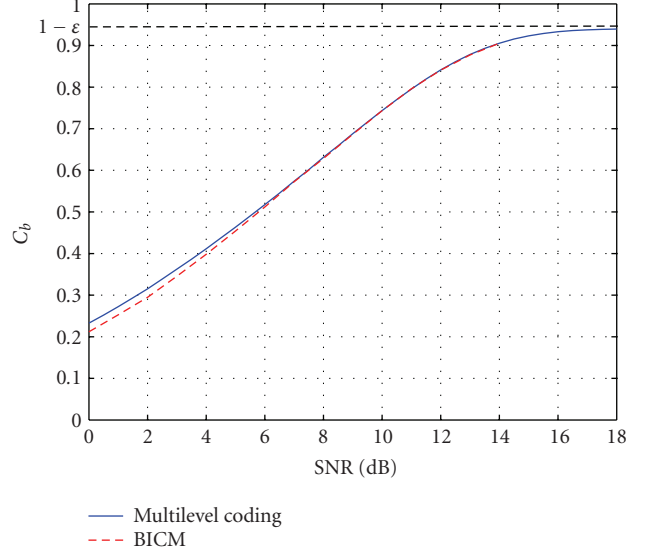

FIGURE 5: The average bit-capacity for a gray-labelled 16-QAM at different SNRs with an erasure rate of $\epsilon = 0.06$.

bit-channels. The effective noise of each bit-channel highly depends on the constellation labelling.

We can easily find the LLRs for different bit-channels. The LLR value for b_i, can be computed as

$$\text{LLR}(b_i) = \ln\frac{\Pr\left(b_i = 0\mid y\right)}{\Pr\left(b_i = 1\mid y\right)}$$

$$= \begin{cases} 0 & \text{if } y = \text{E}, \\ \ln\dfrac{\sum_{a_j\in A\,(b_i=0)}e^{-\|a_j-y\|^2/2\sigma^2}}{\sum_{a_j\in A\,(b_i=1)}e^{-\|a_j-y\|^2/2\sigma^2}} & \text{otherwise}, \end{cases} \tag{9}$$

where y is the received signal, a_j represents a point of the transmitted constellation, $2\sigma^2$ is the Gaussian noise variance, and $A\,(b_i = 0)$ represents the subconstellation of A with the address bit b_i equal to zero.

In BICM, since the bits are interleaved, the density of LLRs that the LDPC code sees is a mixture of LLR densities for different bit-channels. For a gray-labelled 16-QAM constellation this density is shown in Figure 6. In this figure, the erasure probability is 0.06 and the Gaussian channel SNR is 10 dB. Notice that since $\Pr\left(y = \text{E}\right) = \epsilon$, the density of LLRs has an impulse of weight ϵ at zero. Another interesting observation is that the density is not symmetric with respect to the vertical axis. To see why, consider the gray-labelled 16-QAM constellation of Figure 7 and notice that the value of $\text{LLR}(b_0)$ can grow unboundedly large (for received signals far to the right or to the left), but approaches its minimum for the received signals close to the vertical axis. Therefore, the LLR PDF of b_0 (and as a result the mixture of LLR densities) is not symmetric with respect to the vertical axis.

4.3. All-zero codeword assumption

For the design and analysis of binary LDPC codes over symmetric binary-input channels, it is usually assumed that the
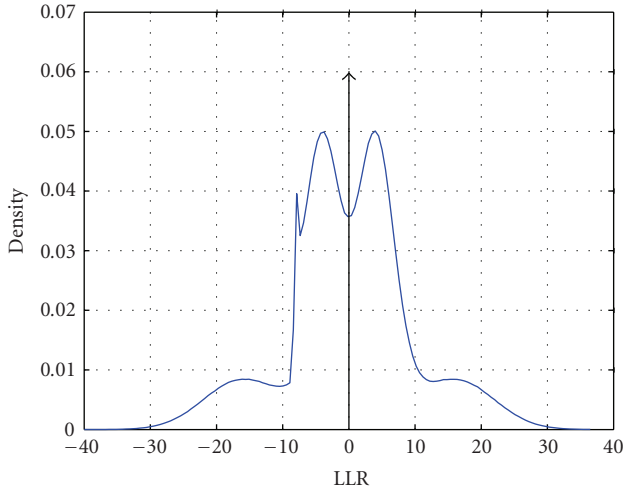
FIGURE 6: The density of LLRs for a 16-QAM gray-labelled constellation at SNR = 10 dB.
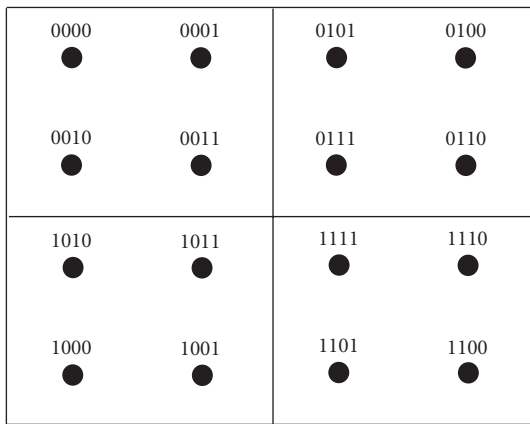

FIGURE 8: The density of modified LLRs for a 16-QAM gray-labelled constellation at SNR = 10 dB.

FIGURE 7: A gray-labelled 16-QAM constellation ($b_3 b_2 b_1 b_0$).

all-zero codeword (equivalent to all-$\{+1\}$ channel-word) is transmitted. In fact, the density of LLR messages associated with the all-$\{+1\}$ channel-word, is essential for the design procedures which are based on density evolution.

In our case, however, assuming that the all-zero codeword is transmitted is not valid because the points of the constellation are not all equivalent. To tackle this problem, we first notice that for an LDPC code of length N and dimension K, an information sequence of K bits, when K is large, is a typical Bernoulli sequence with $p = 1/2$. While not strictly true, it is reasonable to assume that the encoded sequence is also a typical Bernoulli sequence. Therefore, in the encoded codeword, all binary sequences of length l appear with the same probability. Hence, for one typical LDPC codeword, all the constellation points are transmitted the same number of times. As a result, we can assume that almost all the LDPC codewords are received with the same quality at the receiver. That is to say, the performance of the decoder is independent of the codeword, as long as a typical codeword is transmitted.
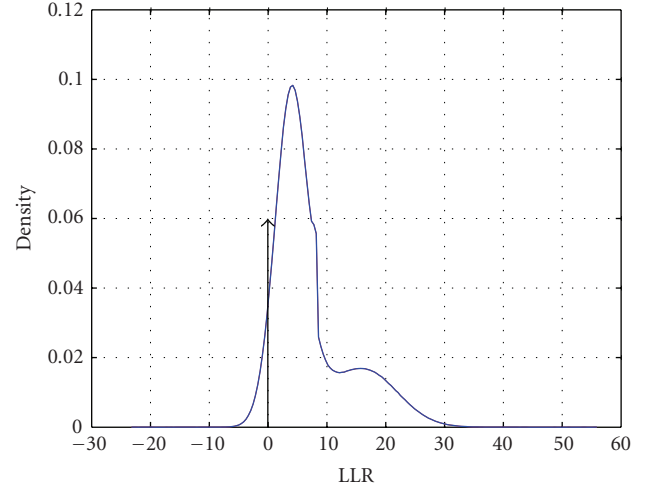
Notice that we can always add a random dither to the codewords to make them typical Bernoulli sequences, but it seems reasonable that such a dither is not required in practice. Indeed, we will see later that the codes designed based on the assumption that all constellation points are transmitted equally likely perform very close to the predicted behavior without the need for a random dither.

Now that all the codewords are assumed to be received with the same quality at the receiver, the performance of the LDPC decoder is independent of the codeword. Therefore, we only need to translate each LLR to its equivalent value corresponding to the all-$\{+1\}$ channel-word. In other words, for every bit which is a "1," we define the LLR to be $\ln(\Pr(1)/\Pr(0))$ and for every bit which is a "0" we use the conventional definition of $\ln(\Pr(0)/\Pr(1))$. This way, a positive LLR carries a correct belief, and (once more) the negative tail of the LLR density shows the LLR error rate. This equivalent LLR density may be computed for all bit-channels and for every constellation point. We may then mix these densities uniformly to produce a modified LLR which is suitable for code optimization.

The density of these modified LLRs is shown in Figure 8. A negative LLR here reflects an error in a hard decision decoder (which was not the case in Figure 6). For a given constellation, a given labelling scheme, and a given noise variance, we can find such a modified LLR density to be used as input to a density evolution program for degree distribution optimization.

4.4. Code design

So far, we have discussed the use of BICM and LDPC codes for DMT systems in an impulsive environment. We now focus on the design of capacity-approaching LDPC codes on such channels. As mentioned before, the optimization algorithm should take the density of LLR messages and the update rules of the decoder as the input and should give an

optimized degree distribution. Thus, given the channel conditions, that is, SNR of the Gaussian part of the channel and ϵ of the erasure part, the purpose of our optimization is to find the highest rate LDPC code whose convergence is guaranteed at these channel conditions.

Different methods are used for finding optimized degree distributions [10, 27]. In this work, we use a method which is based on linear programming and reflects our goal of finding the maximum rate for a given channel. We start with a degree distribution for which convergence to zero error rate is guaranteed. In the density evolution program, at each iteration, we save p_{in}, that is, the error rate of the LLR messages at the input of the iteration (the input messages to the check nodes) and also $p_{out,i}$, that is, the error rate of the LLR messages at the output of variable nodes of degree i (for all values of i). A plot of $p_{out,i}$ versus p_{in} for all values of i is equivalent to the elementary $p_{in} - p_{out}$ extrinsic information transfer (EXIT) charts of [30], that is, EXIT charts corresponding to different variable degrees. For more information on EXIT charts, please see [31].

It is shown in [30] that for a fixed check-degree distribution, a linear combination of elementary $p_{in} - p_{out}$ EXIT charts with a set of weights $\lambda = \{\lambda_2, \ldots, \lambda_{d_v}\}$, such that for all i, $\lambda_i \geq 0$ and $\sum_i \lambda_i = 1$ results in the EXIT chart of an irregular code whose degree distribution is determined by λ, that is, for all p_{in},

$$p_{out} = f(p_{in}) = \sum_i \lambda_i f_i(p_{in}), \tag{10}$$

where $f(p_{in})$ is the EXIT chart of the irregular code and $f_i(p_{in})$ is the elementary EXIT chart associated with variable nodes of degree i.

Therefore, a linear program similar to that of [30] can be used to find an optimized degree distribution. The linear program of [30] takes a set of elementary EXIT charts, a desired convergence behavior (given by a curve $h(p_{in})$), and finds the variable-node-degree distribution λ which maximizes the code rate and for all p_{in} keeps the p_{out} of the irregular code less than or equal to $h(p_{in})$. Here, we have to make sure that the change made in the degree distribution is small, since the elementary EXIT charts found here are valid only for the degree distribution for which the density evolution program is executed.

Assuming that making a small change in the degree distribution of the code will not significantly change the elementary EXIT charts, our design procedure is as follows. First, we find $f(p_{in})$, the current EXIT chart of the code (for which convergence was observed). Then, we make a small change in this EXIT chart to make it closer to the line $p_{out} = p_{in}$ and feed it back as $h(p_{in})$ to the linear program to find another degree distribution whose rate is maximum and whose convergence behavior is better than $h(p_{in})$. It is shown in [30] that by moving towards $h(p_{in}) = p_{in}$ we achieve higher code rates. Using the updated degree distribution, we run the density evolution program to update the elementary EXIT charts and repeat the linear program to update the degree distribution. This optimization is iterated until the desired convergence behavior is $h(p_{in}) \simeq p_{in}$.

TABLE 1: Code design for 16-QAM signalling.

Degree sequence	Code 1	Code 2
d_1, λ_{d_1}	2, 0.1448	2, 0.1561
d_2, λ_{d_2}	3, 0.1767	3, 0.1790
d_3, λ_{d_3}	7, 0.1434	7, 0.0786
d_4, λ_{d_4}	8, 0.0819	8, 0.1404
d_5, λ_{d_5}	21, 0.1015	22, 0.0004
d_6, λ_{d_6}	22, 0.0256	23, 0.0309
d_7, λ_{d_7}	24, 0.0606	24, 0.1656
d_8, λ_{d_8}	25, 0.0273	100, 0.2490
d_9, λ_{d_9}	80, 0.2382	—
d_c	22	13
Channel conditions: $\epsilon = 0.06$, SNR =	10 dB	7 dB
Rate	0.740	0.566
Approx. gap to the Shannon limit	0.09 dB	0.08 dB

5. SIMULATION RESULTS

To show the success of our proposed system, we have designed two irregular codes for 16-QAM signaling at two different channel conditions. The first code is designed at SNR = 10 dB, and $\epsilon = 0.06$ and the second one at SNR = 7 dB and $\epsilon = 0.06$. At these channel conditions, the capacity of BICM is almost 0.744 bits/symbol (per bit-channel) and 0.570 bits/symbol, respectively. Our codes, whose degree distributions can be found in Table 1, have rates of 0.740 bits/symbol and 0.566 bits/symbol, respectively. The maximum variable degree allowed in these designs is 80 for code 1 and 100 for code 2, and their check degree is chosen to be regular. The capacity (normalized per bit-channel) of BICM with 16-QAM signaling on our channel model with $\epsilon = 0.06$ is 0.740 at about 9.91 dB and is 0.566 at about 6.92 dB, hence in both cases a gap of about only 0.08-0.09 dB from the Shannon limit exists.

We have computed the threshold of a code of rate 0.74 optimized for the AWGN channel (with $d_c = 24$ and maximum $d_v = 100$) on our channel. The degree distribution of the code is selected from the optimized codes reported in [32]. We first verified the reported threshold of this code on a Gaussian channel. Then, assuming an erasure rate of $\epsilon = 0.06$ and 16-QAM signalling, we found the threshold of convergence of this code to be at about 10.2 dB or almost 0.29 dB away from the capacity.

It can be concluded that although the performance of codes designed for the AWGN is fairly good on our channel model, further improvement in the order of 0.2 dB is possible. Similar results have been observed in [33], where irregular codes are designed for the Rayleigh channel. In fact, the performance of the codes designed for the AWGN channel on a Rayleigh channel is about 0.2 dB worse than those designed for the Rayleigh channel.
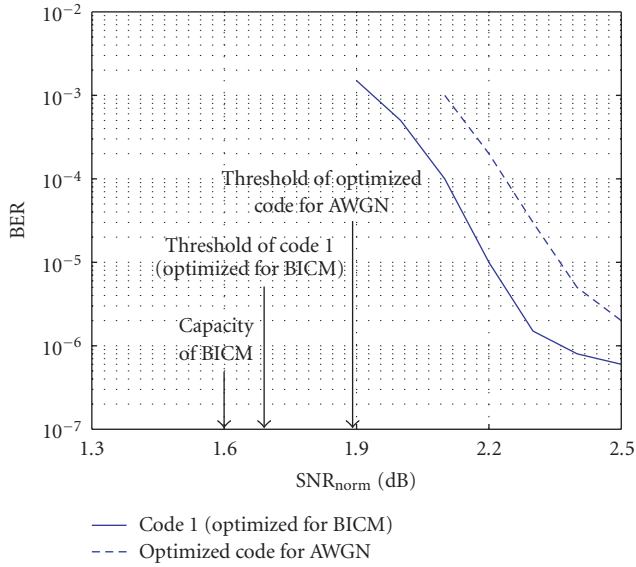
FIGURE 9: Performance comparison of LDPC codes in a system based on 16-QAM signaling and BICM in the presence of impulse noise.

We have also simulated the performance of our rate 0.74 code and have compared it with the above-mentioned rate 0.74 code optimized for AWGN. Both codes have a block length of 30 000. Figure 9 shows the results of our comparison. The erasure rate for simulation is $\epsilon = 0.06$ and a maximum of 100 iterations is allowed. As expected, our code has a performance that is about 0.2 dB better than the code designed for AWGN channel. The code designed for AWGN is slightly more complex because it has a higher check degree and hence its decoding complexity is about 10% higher. The decoding complexity for both codes is comparable with a 512-state trellis code (see [4] for a detailed study of complexity and comparison with the complexity of 512-state trellis code).

With this code length and code rate, and assuming an ADSL system with a bit rate of 1 Mbits/s, the buffer-fill delay is

$$t_{bf} = \frac{30\,000 \times 0.74}{1\,000\,000} = 22.2 \text{ milliseconds.} \qquad (11)$$

Therefore, neglecting the encoding/decoding delay and also the transmission delay, the total delay of the system can be approximated to be about 45 milliseconds. For ADSL systems with higher data bit rates, longer block-length codes can be used and hence, performance closer to the threshold can be expected.

Although SNR_{norm}, defined as $SNR_{norm} = SNR/(2^R - 1)$ [14], is usually a better measure than SNR, this paper uses SNR because our design goal is to maximize the code rate for a given channel condition, and prior to the design the code rate is not known (however, Figure 9 uses SNR_{norm}, since the rate is known).

The conventional solution based on trellis codes and outer RS codes is known to have a gap of a few dB from the Shannon limit. Also notice that a (255, 239) RS code (of a rate a little bit less than 0.94) can correct a maximum of 16 erased bytes. Assuming that each byte has at most one erased bit and that the bit erasure probability is ϵ, we can define $p = 1 - (1 - \epsilon)^8$ as the byte erasure probability. To guarantee an error rate of less than 10^{-6}, p should take a value for which

$$\sum_{i=17}^{255} \binom{255}{i} p^i (1 - p)^{(255-i)} < 10^{-6}. \qquad (12)$$

This results in an error-correction capability of $p < 0.0159$ and hence $\epsilon < 2.003 \times 10^{-3}$. In other words, about 0.2% error can overwhelm an RS code that introduces more than 6% rate loss to the system. Our code, however, can correct up to 6% erasures. This erasure correction capability is achieved by a rate loss of 6% introduced by the channel model whose capacity at high SNR is 0.94.

6. CONCLUSIONS

In this work, we have shown that a carefully optimized LDPC code is a promising candidate coding approach in DMT systems with impulsive noise. To illustrate the potential of this approach, we have shown how to optimize an LDPC code at a fixed SNR and for a fixed constellation size. Further research is required to address practical DMT systems that require flexible coding to accommodate a variety of SNRs and hence a variety of constellation sizes at different tones. Indeed, it is shown in [4] that a single LDPC code can perform effectively in a DMT system if a fixed number of bits from each constellation is assigned to it.

From Figure 9, it can be seen that both LDPC codes show an error floor at bit error rates near 10^{-6}. Error floors in LDPC codes are mainly due to the presence of short cycles in the graph structure of the code. This will be a less serious problem for long block-length codes. Many solutions have been proposed to handle the error floor issue. One easy solution is to use a very high-rate, low-complexity outer BCH code (e.g., a double-error correcting BCH code). However, there also exist LDPC codes which are carefully structured to push the error floor to a much lower BER (10^{-8} and lower) [34, 35]. Using such LDPC codes, the need for an outer code is completely eliminated.

REFERENCES

[1] T. N. Zogakis, J. T. Aslanis Jr., and J. M. Cioffi, "Analysis of a concatenated coding scheme for a discrete multitone modulation system," in *Proceedings of IEEE Military Communications Conference (MILCOM '94)*, vol. 2, pp. 433–437, Fort Monmouth, NJ, USA, October 1994.

[2] L. Zhang and A. Yongacoglu, "Turbo coding in ADSL DMT systems," in *Proceedings of IEEE International Conference on Communications (ICC '01)*, vol. 1, pp. 151–155, Helsinki, Finland, June 2001.

[3] Z. Cai, K. R. Subramanian, and L. Zhang, "DMT scheme with multidimensional turbo trellis code," *Electronics Letters*, vol. 36, no. 4, pp. 334–335, 2000.

[4] M. Ardakani, T. Esmailian, and F. R. Kschischang, "Near-capacity coding in multicarrier modulation systems," *IEEE Transactions on Communications*, vol. 52, no. 11, pp. 1880–1889, 2004.

[5] E. Eleftheriou and S. Ölçer, "Low-density parity-check codes for digital subscriber lines," in *Proceedings of IEEE International Conference on Communications (ICC '02)*, vol. 3, pp. 1752–1757, New York, NY, USA, April—May 2002.

[6] T. Cooklev, M. Tzannes, and A. Friedman, "Low-density parity-check coded modulation for ADSL," Temporary Document BI-081, ITU-Telecommunication Standardization Sector, Geneva, Switzerland, October 2000.

[7] C. Berrou, A. Glavieux, and P. Thitimajshima, "Near Shannon limit error-correcting coding and decoding: Turbo-codes. 1," in *Proceedings of IEEE International Conference on Communications (ICC '93)*, vol. 2, pp. 1064–1070, Geneva, Switzerland, May 1993.

[8] D. Divsalar and F. Pollara, "On the design of turbo codes," TDA Progr. Rep. 42-123, Jet Propulsion Laboratory, California Institute of Technology, Pasadena, Calif, USA, 1995.

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

[10] T. J. Richardson, M. A. Shokrollahi, and R. L. Urbanke, "Design of capacity-approaching irregular low-density parity-check codes," *IEEE Transactions on Information Theory*, vol. 47, no. 2, pp. 619–637, 2001.

[11] A. Shokrollahi, "Capacity-achieving sequences," in *Codes, Systems, and Graphical Models*, B. Marcus and J. Rosenthal, Eds., IMA Volumes in Mathematics and Its Applications, no. 123, pp. 153–166, Springer, New York, NY, USA, 2000.

[12] H. Imai and S. Hirakawa, "A new multilevel coding method using error-correcting codes," *IEEE Transactions on Information Theory*, vol. 23, no. 3, pp. 371–377, 1977.

[13] G. Caire, G. Taricco, and E. Biglieri, "Bit-interleaved coded modulation," *IEEE Transactions on Information Theory*, vol. 44, no. 3, pp. 927–946, 1998.

[14] G. D. Forney Jr. and G. Ungerboeck, "Modulation and coding for linear Gaussian channels," *IEEE Transactions on Information Theory*, vol. 44, no. 6, pp. 2384–2415, 1998.

[15] W. Yu, D. Toumpakaris, J. M. Cioffi, D. Gardan, and F. Gauthier, "Performance of asymmetric digital subscriber lines in an impulse noise environment," *IEEE Transactions on Communications*, vol. 51, no. 10, pp. 1653–1657, 2003.

[16] M. Barton, "Impulse noise performance of an asymmetric digital subscriber lines passband transmission system," *IEEE Transactions on Communications*, vol. 43, no. 234, pp. 1337–1340, 1995.

[17] K. J. Kerpez and A. M. Gottlieb, "The error performance of digital subscriber lines in the presence of impulse noise," *IEEE Transactions on Communications*, vol. 43, no. 5, pp. 1902–1905, 1995.

[18] D. Toumpakaris, W. Yu, J. M. Cioffi, D. Gardan, and M. Ouzzif, "A byte-erasure method for improved impulse immunity in DSL systems using soft information from an inner code," in *Proceedings of IEEE International Conference on Communications (ICC '03)*, vol. 4, pp. 2431–2435, Anchorage, Alaska, USA, May 2003.

[19] D. Toumpakaris, J. M. Cioffi, D. Gardan, and M. Ouzzif, "A square distance-based byte-erasure method for reduced-delay protection of DSL systems from non-stationary interference," in *Proceedings of IEEE Global Telecommunications Conference (GLOBECOM '03)*, vol. 4, pp. 2114–2119, San Francisco, Calif, USA, December 2003.

[20] P. S. Chow, *Bandwidth optimized digital transmission techniques for spectrally shaped channels with impulse noise*, Ph.D. thesis, Department of Electrical Engineering, Stanford University, Stanford, Calif, USA, May 1993.

[21] U. Wachsmann, R. F. H. Fischer, and J. B. Huber, "Multilevel codes: theoretical concepts and practical design rules," *IEEE Transactions on Information Theory*, vol. 45, no. 5, pp. 1361–1391, 1999.

[22] G. J. Pottie and D. P. Taylor, "Multilevel codes based on partitioning," *IEEE Transactions on Information Theory*, vol. 35, no. 1, pp. 87–98, 1989.

[23] A. R. Calderbank, "Multilevel codes and multistage decoding," *IEEE Transactions on Communications*, vol. 37, no. 3, pp. 222–229, 1989.

[24] R. M. Tanner, "A recursive approach to low complexity codes," *IEEE Transactions on Information Theory*, vol. 27, no. 5, pp. 533–547, 1981.

[25] F. R. Kschischang, B. J. Frey, and H.-A. Loeliger, "Factor graphs and the sum-product algorithm," *IEEE Transactions on Information Theory*, vol. 47, no. 2, pp. 498–519, 2001.

[26] M. G. Luby, M. Mitzenmacher, M. A. Shokrollahi, and D. A. Spielman, "Efficient erasure correcting codes," *IEEE Transactions on Information Theory*, vol. 47, no. 2, pp. 569–584, 2001.

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

[28] J. Hou, P. H. Siegel, L. B. Milstein, and D. Pfister, "Multilevel coding with low-density parity-check component codes," in *Proceedings of IEEE Global Telecommunications Conference (GLOBECOM '01)*, vol. 2, pp. 1016–1020, San Antonio, Tex, USA, November 2001.

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

[30] M. Ardakani, T. H. Chan, and F. R. Kschischang, "EXIT-chart properties of the highest-rate LDPC code with desired convergence behavior," *IEEE Communications Letters*, vol. 9, no. 1, pp. 52–54, 2005.

[31] S. ten Brink, "Convergence behavior of iteratively decoded parallel concatenated codes," *IEEE Transactions on Communications*, vol. 49, no. 10, pp. 1727–1737, 2001.

[32] http://lthcwww.epfl.ch/research/ldpcopt/.

[33] J. Hou, P. H. Siegel, and L. B. Milstein, "Performance analysis and code optimization of low density parity-check codes on Rayleigh fading channels," *IEEE Journal on Selected areas in Communications*, vol. 19, no. 5, pp. 924–934, 2001.

[34] Flarion Inc., "Vector-LDPC Coding Solution Data Sheet", http://www.flarion.com/products/overviews/Vector-LDPC_Product_Overview.pdf.

[35] T. Tian, C. Jones, J. D. Villasenor, and R. D. Wesel, "Construction of irregular LDPC codes with low error floors," in *Proceedings of IEEE International Conference on Communications (ICC '03)*, vol. 5, pp. 3125–3129, Anchorage, Alaska, USA, May 2003.

Masoud Ardakani received the B.S. degree from Isfahan University of Technology in 1994, the M.S. degree form Tehran University in 1997, and the Ph.D. degree from the University of Toronto in 2004, all in electrical engineering. He was a Postdoctoral fellow at the University of Toronto from 2004 to 2005. From 1997 to 1999, he was with the Electrical and Computer Engineering Research Center, Isfahan, Iran. Currently, he is an Assistant Professor of electrical and computer engineering at the University of Alberta, where he holds an Informatics Circle of Research Excellence (iCore) Junior Research Chair in wireless communications. His research interests are in the general area of digital communications, codes defined on graphs associated with iterative decoding, and MIMO systems.

Frank R. Kschischang received the B.A.Sc. degree (with honors) from the University of British Columbia in 1985, and the M.A.Sc. and Ph.D. degrees from the University of Toronto in 1988 and 1991, respectively, all in electrical engineering. He is a Professor of electrical and computer engineering and Canada Research Chair in communication algorithms at the University of Toronto, where he has been a faculty member since 1991. During 1997-1998 he was a Visitor at the Massachusetts Institute of Technology, Cambridge, Mass, and in 2005 at the Swiss Federal Institute of Technology, Zurich, Switzerland. His research interests are focused on the area of coding techniques, primarily on soft-decision decoding algorithms, trellis structure of codes, codes defined on graphs, and iterative decoders. Dr. Kschischang is a recipient of the Ontario Premier's Research Excellence Award. From October 1997 to October 2000, he served as the Associate Editor for coding theory for the IEEE Transactions on Information Theory. He was Technical Program Cochair of the 2004 IEEE International Symposium on Information Theory held in Chicago.

Wei Yu received the B.A.Sc. degree in computer engineering and mathematics from the University of Waterloo, Waterloo, Ontario, Canada, in 1997, and M.S. and Ph.D. degrees in electrical engineering from Stanford University, Stanford, Calif, USA, in 1998 and 2002, respectively. Since 2002, he has been an Assistant Professor with the Electrical and Computer Engineering Department, University of Toronto, Toronto, Ontario, Canada, where he also holds a Canada Research Chair. His main research interests include multiuser information theory, coding, optimization, wireless communications, and broadband access networks.