

Near-Capacity Coding in Multicarrier Modulation Systems

Masoud Ardakani, *Student Member, IEEE*, Tooraj Esmailian, and Frank R. Kschischang, *Senior Member, IEEE*

Abstract—We apply irregular low-density parity-check (LDPC) codes to the design of multilevel coded quadrature amplitude modulation (QAM) schemes for application in discrete multitone systems in frequency-selective channels. A combined Gray/Ungerboeck scheme is used to label each QAM constellation. The Gray-labeled bits are protected using an irregular LDPC code with iterative soft-decision decoding, while other bits are protected using a high-rate Reed–Solomon code with hard-decision decoding (or are left uncoded). The rate of the LDPC code is selected by analyzing the capacity of the channel seen by the Gray-labeled bits and is made adaptive by selective concatenation with an inner repetition code. Using a practical bit-loading algorithm, we apply this coding scheme to an ensemble of frequency-selective channels with Gaussian noise. Over a large number of channel realizations, this coding scheme provides an average effective coding gain of more than 7.5 dB at a bit-error rate of 10^{-7} and a block length of approximately 10^5 b. This represents a gap of approximately 2.3 dB from the Shannon limit of the additive white Gaussian noise channel, which could be closed to within 0.8–1.2 dB using constellation shaping.

Index Terms—Discrete multitone systems, low-density parity-check (LDPC) codes, multilevel-coded modulation, frequency-selective channels.

I. INTRODUCTION

IN DISCRETE multitone (DMT) systems, there are many sub-channels with different signal-to-noise ratios (SNRs). Designing a coding system that meets the requirements of these channels has drawn much attention [1]–[3]. In standard asymmetric digital subscriber lines, a trellis-coded-modulation scheme in concatenation with a Reed–Solomon code is used [4], providing approximately 5-dB coding gain at a symbol-error rate (SER) of 10^{-7} .

The two principal classes of codes for the high-SNR regime are lattice codes and trellis codes. The SNR gap between uncoded baseline performance of quadrature-amplitude modulation (QAM) and Shannon limit is 9 dB at an SER of 10^{-6} . At this SER, the coding gain of the Leech lattice in dimension 24 is less than 4 dB [5]. With a 512-state trellis code, an effective

coding gain of 5.5 dB can be achieved, but it seems that approaching a coding gain close to 6 dB with trellis codes is very difficult [5]. All of the coding gains that we refer to are only due to coding. Using proper shaping techniques, a shaping gain of up to 1.53 dB can be added independent of the coding gain. However, even with a 512-state trellis code and achieving the ultimate shaping gain, there is a gap of 2 dB from the Shannon limit.

The goal of this paper is to provide near-capacity coding techniques for the high-SNR regime and more generally for DMT systems. At low SNR, the problem of near-capacity coding has been studied extensively, and it has been shown [6]–[9] that turbo codes and low-density parity-check (LDPC) codes can approach the capacity of many channels with practical complexity. For high-SNR channels, however, multilevel modulation is required. The main problem of using multilevel symbols in these codes is that large alphabet sizes create prohibitively large decoding complexity. To overcome this problem, one can use multilevel coding, which allows one to apply binary codes to multilevel modulation schemes. However, in DMT systems, dealing with subchannels whose SNR is different and use different constellation size is a challenge.

In [10], a regular high-rate LDPC code is used for error correction in a digital subscriber line (DSL) transmission system. The maximum reported coding gain at a bit-error rate (BER) of 10^{-7} is 6.2 dB, which, compared to the maximum possible coding gain (8.3 dB for the additive white Gaussian noise (AWGN) channel), shows a gap of more than 2 dB from the Shannon limit. The goal of [10] is to provide a coding system for DSL transmission systems with practical complexity and suitable structure for hardware implementation, so highly efficient irregular codes, which are more difficult to implement, were not considered. The idea of using LDPC codes together with coded modulation is used in [11] as well. In [2], a turbo code has 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. This amounts to a 1.5-dB gap from the Shannon limit for the AWGN channel.

In this paper, we address the problem of coding for DMT systems and propose a coding scheme based on a combination of irregular LDPC codes and multilevel coding, which provides an average coding gain of more than 7.5 dB at a message error rate of 10^{-7} . This is equivalent to a gap of approximately 0.8 dB from the Shannon limit for the AWGN channel. The decoding complexity in our system is comparable with a 512-state trellis code.

The main difference between our approach and other work on use of turbo codes/LDPC codes for DMT systems is that our

Paper approved by W. E. Ryan, the Editor for Modulation, Coding, and Equalization of the IEEE Communications Society. Manuscript received June 19, 2003; revised March 3, 2004 and May 26, 2004.

M. Ardakani and F. R. Kschischang are with the Edward S. Rogers Sr. Department of Electrical and Computer Engineering, University of Toronto, Toronto, ON M5S 3G4, Canada (e-mail: masoud@comm.utoronto.ca; frank@comm.utoronto.ca).

T. Esmailian was with the Edward S. Rogers Sr. Department of Electrical and Computer Engineering, University of Toronto, Toronto, ON M5S 3G4, Canada. He is now with the Research and Development Group, Edge-water Computer Systems, Inc., Kanata, ON K2K 3G6, Canada (e-mail: tooraj@eecg.utoronto.ca).

Digital Object Identifier 10.1109/TCOMM.2004.836560

main goal is to maximize the coding gain for a practical decoding complexity. For this purpose, motivated by [12] and [13], after labeling symbols with binary sequences, we model an effective channel—which will be referred to as a bit-channel—for each bit level of the label. Then, for different constellation sizes, we compute the capacity of the bit-channels that each label bit sees. This allows us to choose an efficient error-correcting code for each level bit whose code rate accounts for the capacity of the associated bit-channel.

Another difference is that we avoid coding all of the label bits with the same code. Although it has been shown that bit-interleaved coded modulation together with a Gray-labeling can perform very close to multilevel coding with set partitioning labeling [14], and hence the performance loss in this strategy is not an issue, we use the powerful LDPC coding only for those label bits that in fact need this high-complexity coding scheme. As a result, we propose a quasi-multilevel-coding system. We use a single LDPC code for the least significant bits of the label, but higher level bits are coded with less complex codes.

Another benefit of this strategy is that we avoid high-rate LDPC codes. High-rate irregular LDPC codes need a very high check degree in order to be able to perform near capacity. High-degree check nodes introduce a large number of short cycles in the Tanner graph of the code, which adversely affects the performance in the case of finite-length codes.

Moreover, instead of choosing from a set of codes with different rates at each channel condition, we employ a fixed-rate LDPC code that is selectively concatenated with a repetition code to provide a flexible overall coding rate for the system. This is necessary, since for some realizations of the channel the capacity of the bit-channels assigned to the LDPC code can be less than the rate of the LDPC code. The repetition code is used only for bit-channels whose capacity is very low.

In terms of hardware implementation, compared to the approach in [10], which uses highly structured LDPC codes in a multilevel coding system, our system is more complex because we use constellations of size 2^l where l can be an even or odd integer, and our LDPC code is irregular. Although our decoding complexity is similar, the encoding complexity in our system is higher because we have not used codes with algebraic construction.

All of the above differences have allowed us to approach capacity much more closely. Moreover, our system works for a very wide range of channel conditions because it has a flexible code rate.

This paper is organized as follows. In Sections II and III, we briefly review some aspects of multilevel coding and LDPC codes. In Section IV, our channel model is briefly discussed. In Section V, we explain the structure of our proposed system. In Section VI, we present the specifications of our system for an ensemble of frequency-selective channels encountered in power-line channels [15]. Section VII shows some simulation results, and, finally, we conclude the paper in Section VIII.

II. MULTILEVEL CODING

The idea of multilevel coding was introduced in 1977 by Imai and Hirakawa [16]. Like Ungerboeck's idea for trellis-coded

modulation [17], the idea of multilevel coding is based on the concept of set partitioning.

Assume that each point of a constellation $\mathbf{A} = \{a_0, a_1, \dots, a_{M-1}\}$ of $M = 2^l$ points is labeled with a binary address $\mathbf{b} = (b_0, b_1, \dots, b_{l-1})$. The idea of multilevel coding is to protect each address bit b_i of the constellation point by an individual binary code C^i at level i [12].

Since the mapping between addresses and constellation points is one-to-one, if A and Y represent the random variables corresponding to the transmitted and received signals, respectively, then the mutual information $I(Y; A)$ is equal to the mutual information between address vector and the received signal. Considering $(b_0, b_1, \dots, b_{l-1})$ as the vector of random variables corresponding to the address bits, we have

$$\begin{aligned} I(Y; A) &= I(Y; b_0, b_1, \dots, b_{l-1}) \\ &= I(Y; b_0) + I(Y; b_1 | b_0) \\ &\quad + \dots + I(Y; b_{l-1} | b_0, b_1, \dots, b_{l-2}). \end{aligned} \quad (1)$$

Hence, transmission of address vector $(b_0, b_1, \dots, b_{l-1})$ over the physical channel can be separated into parallel transmission of individual digits. Notice that each address bit sees an effective channel (bit-channel) whose capacity can be different from the other bit-channels due to the labeling scheme.

At the decoder, according to (1), the mutual information $I(Y; b_0)$ is used in order to decode b_0 . Then the conditional mutual information $I(Y; b_1 | b_0)$ is used to decode b_1 . That is to say, the capacity of the second bit-channel has to be defined conditioned on the first address bit. In general, for the i th bit-channel, the capacity is defined conditioned on b_0 to b_{i-1} . If set partitioning labeling, e.g., an Ungerboeck labeling, is used for label bits, then conditioned on b_0 , bit b_1 sees a higher capacity bit-channel. Hence, a higher rate code can be used for it. At the next stage, conditioned on both b_0 and b_1 , bit b_2 sees an even higher capacity bit-channel, and so on. Therefore, label bits must be protected by different rate codes. It is known that, if each code achieves the capacity of its bit-channel, the total capacity of the channel is achieved [12]. Fig. 1 shows the net capacity of 4-QAM and 8-QAM signaling together with the capacity of each bit-channel as a function of SNR assuming that an Ungerboeck labeling is used. We refer the reader to [18] and [19] for a complete study of these systems and a more detailed structure of the encoder and decoder for such systems.

III. LDPC CODES

In the case of binary modulation, LDPC codes are capable of approaching the capacity of many different types of channels with a practical decoding complexity [8], [9], [20]. For the binary erasure channel, LDPC codes can, in fact, achieve the capacity under message-passing decoding [21].

Decoding LDPC codes is based on iterative message-passing algorithms. There are many different message-passing algorithms, among which the sum-product algorithm [22] appears to be the most accurate. The main idea in all message-passing algorithms is that each message on each edge carries a belief about the value of the adjacent variable node. This can be,

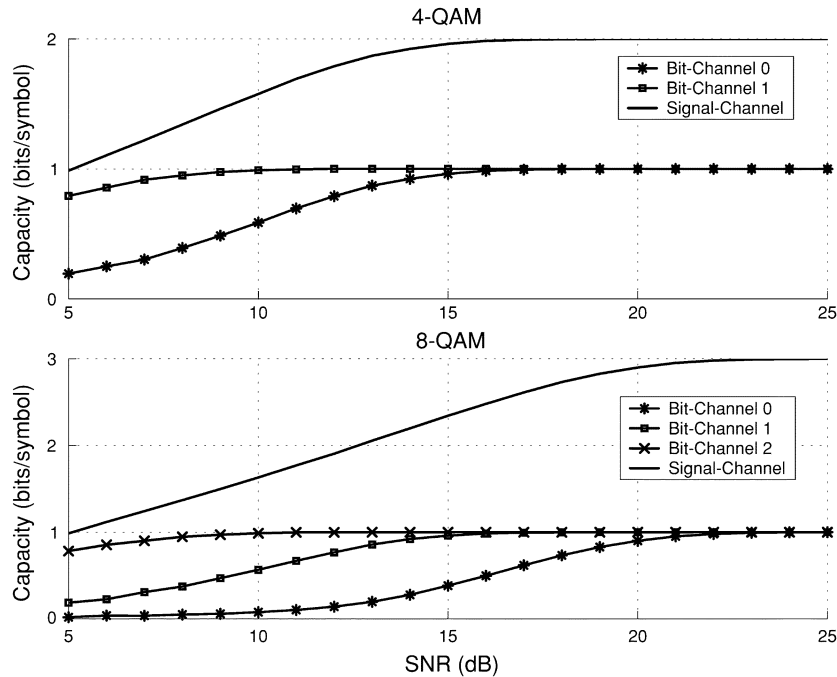


Fig. 1. Net capacity and capacities associated with bit-channels (bit-channel 0: $I(Y; b_0)$; bit-channel 1: $I(Y; b_1 | b_0)$; bit-channel 2: $I(Y; b_2 | b_0, b_1)$) for 4-QAM/8-QAM signaling when an Ungerboeck labeling is used.

for example, a hard decision binary value, a soft probability measure such as $p(0)$, or the log-likelihood ratio (LLR) of probabilities $\log[p(0)/p(1)]$. In each iteration, all of the variable nodes send their messages to all adjacent check nodes and then all of the check nodes send their updated messages back. The update rules depend on the decoding algorithm. In [22], different decoding algorithms are discussed.

Depending on their structure, LDPC codes are said to be “regular” or “irregular.” In regular LDPC codes, all variable nodes have a fixed degree and all check nodes have another fixed degree. In the irregular case, variable nodes and check nodes of different degree are used. It has been shown that irregular codes can have a better performance than regular ones [8], [23].

An ensemble of irregular codes can be defined by their variable and check degree distributions. A degree distribution at the variable/check side defines the portion of edges incident to variable/check nodes with different degrees. For instance, if the variable degree distribution of a code is $\Lambda = \{\lambda_2 = 0.2, \lambda_5 = 0.5, \lambda_9 = 0.3\}$, it means that 20% of edges are connected to degree-2 variable nodes, 50% of edges are connected to degree-5 variable nodes and 30% of them are connected to degree-9 variable nodes. Similarly, the check nodes have their own degree distribution. As the block length of a randomly chosen code approaches infinity, the performance of the code depends only on its degree distributions [8]. Hence, the design of an ensemble of irregular LDPC codes is equivalent to finding good variable and check degree distributions. The main goal of design is to find the maximum-rate ensemble that guarantees some required convergence behavior for a given channel condition, in the limit of long block length.

In practice, when a finite-length code is used, the performance of the code depends on the block length as well: the larger the block length, the closer the performance of the code to the predicted asymptotic behavior. For instance, with a block length of

10^5 , there is a gap of about 0.2 dB to the predicted performance, and with a block length of 10^6 this gap is reduced to 0.05 dB [9].

IV. CHANNEL MODEL

Although the approach of this paper in terms of providing a near-capacity coding system for DMT systems is quite general, we consider in-building power-line channels as an example, and our focus in the design will be this class of channels.

A detailed study of characteristics of in-building power-line channels has been conducted in [15], and a stochastic ensemble of test channels described. The parameters of the test channels are based on limitations placed on wiring configurations by the National Electric Code, by the size and type of building in which the power lines are located, by the expected load impedances, and by experimental measurement of background and impulsive noise.

We use the channel model provided in [15] to generate our sample test channels. We also use the distribution of channel SNR provided in [15] for our system design. To show the success of our design, we present results for different size buildings and different realizations of the channel. Fig. 2 shows the magnitude of the frequency response of three representative test channels. For more details about the parameters of these channels, see [15].

The application to power-line channels is intended to be representative only, and we expect the results of this paper to apply to broad classes of frequency-selective Gaussian channels.

V. SYSTEM STRUCTURE

Equation (1) is one of many different ways that one can partition the total capacity to the capacity of single bits. As discussed before, this suggests using an Ungerboeck labeling together

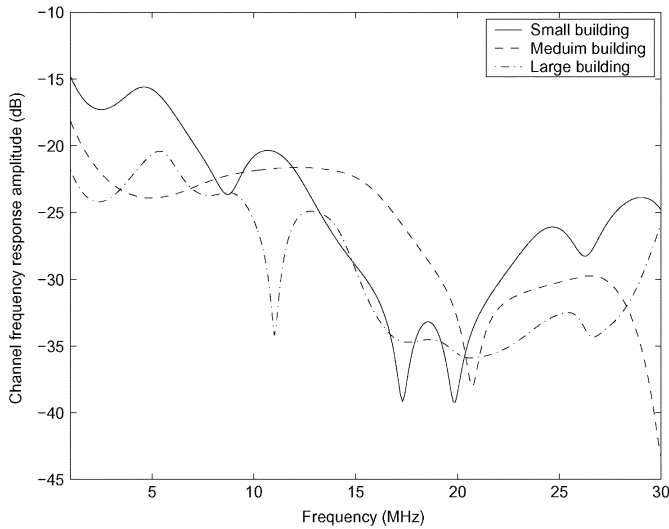


Fig. 2. Magnitude of the frequency response of three test channels.

with separate codes for each level of the label bits. Constellation labeling and decoding stages strongly depend on the way that one partitions the total capacity.

Since we wish to perform near the capacity of the channel, we consider using LDPC codes for different levels of the label bits. However, we avoid using a single code for each address bit because decoding higher level bits is not possible without finishing the decoding for lower level bits. As a result, to avoid a long decoding delay, the length of the employed LDPC codes should be relatively small which hinders the performance of the code.

An alternative solution for our system is bit-interleaved coded modulation, i.e., to use a single code for all label bits without partitioning the total capacity to sum of bit-channel capacities. Bit-interleaved coded modulation, together with a Gray labeling, can approach the channel capacity very closely [14], but unlike multilevel coding cannot achieve it. The capacity of Gray-coded channels for various constellation sizes and number of coded bits is analyzed in [25].

Assigning all label bits to one LDPC code has some disadvantages. For instance, it requires a high-rate LDPC code which performs close to capacity only with high-degree check nodes. High-degree check nodes introduce short cycles in the factor graph of the code and hinder the performance. Another problem with bit-interleaved coded modulation is that in terms of decoding complexity it may not be an efficient solution. Notice that the capacity of higher bit-channels under set partitioning labeling can be very close to one (typically more than 0.95 b/symbol). Hence, in some bits, no coding is required and other bits can be protected effectively by a low complexity code such as a Reed–Solomon code.

Considering these facts, we propose using a single code for only the two least significant bits. This way, we can double the length of the code for the same delay and hence have a better performance. Notice that the LDPC decoding complexity scales linearly with the length of the code. Hence, the complexity per bit of information is independent of the code length. Encoding and decoding these two bits together is equivalent to bit-interleaved code modulation for these two bits. Hence, a Gray labeling must be used in order to approach the

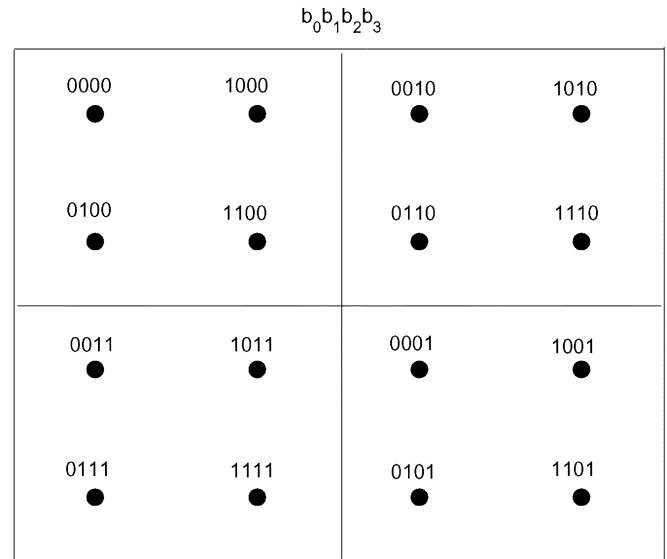


Fig. 3. Labeling of a 16-QAM constellation in our system.

capacity very closely [14]. In other words, these two address bits will have equal minimum Euclidean distance and almost equal error probability, which allows use of a single LDPC code for them.

We use Ungerboeck labeling for higher level bits to further increase the capacity for even higher address bits. This provides us with some bit-channels whose raw error rate is very low and can be coded with low-complexity high-rate codes and some bit-channels whose raw error rate is better than the target error rate of the system and hence do not need any coding. This will further reduce the overall decoding complexity of the system. In this paper, only b_2 and b_3 need protection and higher address bits are not coded. Fig. 3 shows the labeling used for a 16-QAM constellation in our system. Notice that for b_2 and b_3 an Ungerboeck labeling is used.

While one of our main reasons for bundling b_0 and b_1 together was to permit a longer blocklength for the LDPC code, we avoid including b_2 in this bundle. This is because many subchannels do not use constellation sizes more than 4-QAM. Thus, including b_2 in the LDPC code increases the complexity of the system, but they do not contribute much in the length of the LDPC code to provide a better performance.

Based on the above discussion, we rewrite (1) as

$$I(Y; b_0, b_1, \dots, b_{l-1}) = I(Y; b_0, b_1) + I(Y; b_2 | b_0, b_1) + \dots + I(Y; b_{l-1} | b_0, b_1, \dots, b_{l-2}). \tag{2}$$

For b_0 and b_1 , the average bit-channel capacity can be computed as

$$C_{b_0 b_1} = 1 - \frac{1}{2N} \sum_{b_0=0}^1 \sum_{b_1=0}^1 \sum_{k=0}^{N/4-1} \cdot E \left\{ \log_2 \left[\frac{\sum_{i=0}^N \exp \left(\frac{-1}{2\sigma^2} |a_{b_0 b_1}^k + w - a^i|^2 \right)}{\sum_{i=0}^{N/4-1} \exp \left(\frac{-1}{2\sigma^2} |a_{b_0 b_1}^k + w - a_{b_0 b_1}^i|^2 \right)} \right] \right\} \tag{3}$$

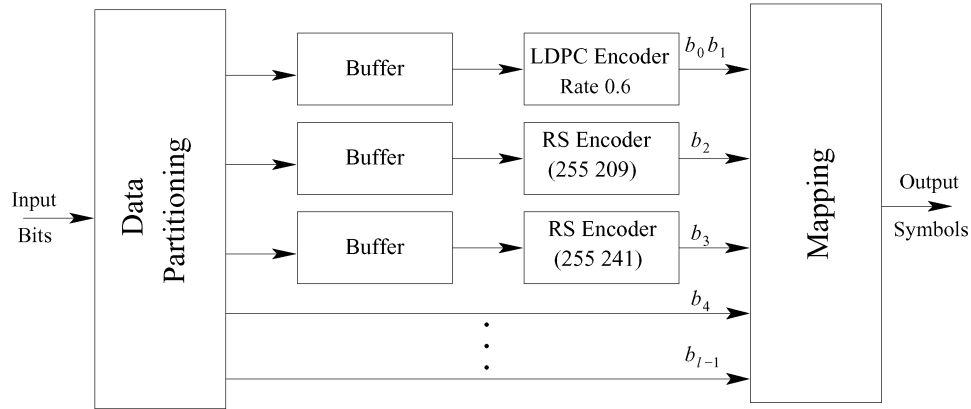


Fig. 4. Block diagram of the encoder of the proposed system.

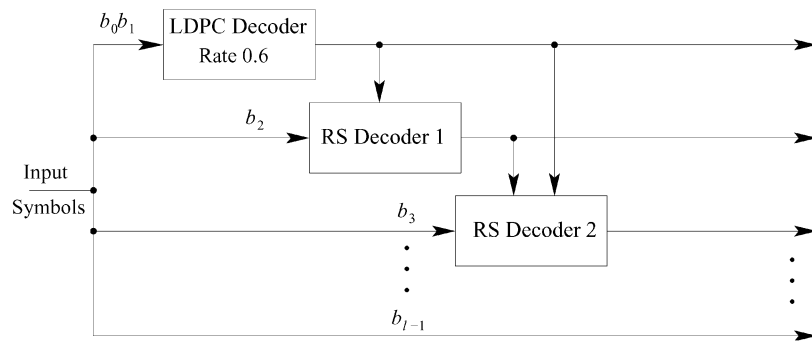


Fig. 5. Block diagram of the decoder of the proposed system.

where N is the number of points in the constellation, a^i is the i th constellation point, $a_{b_0 b_1}^k$ is the k th point on the subconstellation with fixed b_1 and b_0 , and w represents samples of a complex Gaussian noise with variance $2\sigma^2$.

Since the rate of the irregular LDPC code that we use is fixed and very close to the bit-channel capacity, but the characteristics of the channel vary with time and frequency, we use an inner repetition code to take care of cases that the capacity of the bit-channels assigned to the LDPC code is less than the rate of the LDPC code. The repetition code is used only for bit-channels whose capacity is very low. In other words, our code (LDPC code in concatenation with the repetition code) has an adaptive rate to guarantee convergence over all channel realizations. Another advantage of the repetition code is that it allows use of low-capacity bit-channels. This in turn allows us to switch to higher constellation sizes at lower SNRs compared to previous work. As a result, the system can perform even closer to the capacity.

Figs. 4 and 5 show the block diagram of the proposed system at the transmitter and the receiver, respectively. At the transmitter, the constellation mapper uses two bits from the output of LDPC encoder as the lowest address bits. For higher address bits it uses the output of other encoders plus the uncoded bits. In the case of 2-PAM, only one bit from the LDPC encoder is used. Using these address bits, a point from the constellation is chosen for each subchannel of the DMT system. After assigning data to all subchannels, the inverse fast Fourier transform (IFFT) is used to map this complex vector to a DMT symbol. At the receiver, the DMT symbol is converted to a complex vector using the FFT.

Our LDPC decoder first computes the LLR values for each subchannel. The LLR value for b_0 is computed as

$$\begin{aligned} \text{LLR}(b_0) &= \ln \frac{P(b_0 = 0 | z)}{P(b_0 = 1 | z)} \\ &= \ln \frac{\sum_{a_i \in A(b_0=0)} e^{-\|a_i g_i - z\|^2 / 2\sigma^2}}{\sum_{a_i \in A(b_0=1)} e^{-\|a_i g_i - z\|^2 / 2\sigma^2}} \end{aligned} \quad (4)$$

in which z is the received signal, a_i represents a point of transmitted constellation, g_i is the subchannel gain, σ is the Gaussian noise variance, and $A(b_0 = 0)$ represents a subconstellation of A with the address bit b_0 equal to zero. Similarly, the LLR value for b_1 can be computed. Once all LLR values are ready, they will be decoded in the LDPC decoder. These decoded values are used to decode higher level address bits like any other multistage decoder.

VI. SYSTEM SPECIFICATIONS

In this system, we use a rate-0.6 irregular LDPC code. The rate of the LDPC code is chosen according to the expected value of $C_{b_0 b_1}$ over all realizations of the channel. Fig. 6 depicts $C_{b_0 b_1}$, the capacity of the bit-channel that b_2 sees and the distribution of subchannels as a function of SNR.

The system delay is mainly due to the following:

- buffer-fills in the LDPC encoder/decoder, which can be computed as shown in the equation at the bottom of the next page;
- decoding delay: $t_d = \frac{(\text{LDPC code length})}{(\text{decoder throughput})}$.

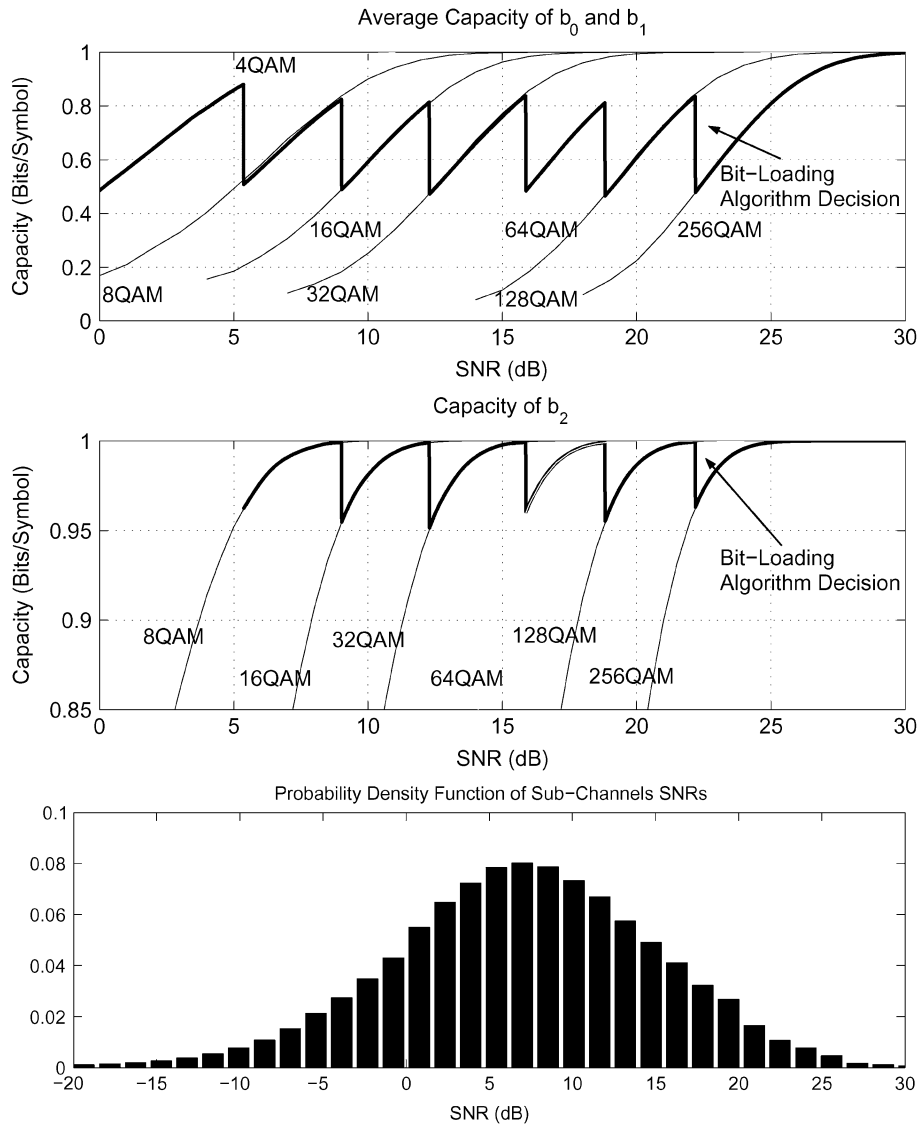


Fig. 6. $C_{b_0b_1}$, the bit-channel capacity for b_2 and the PDF of subchannels as a function of SNR.

Compared to above-mentioned delays, encoding delay of the LDPC code, transmission delay, and other delays due to higher level codes are negligible. Hence, the total delay can be approximated as $2t_{bf} + t_d$.

In our system, the length of the LDPC code is 100 000 and its rate is 0.6. Assuming a typical throughput of 100 Mb/s for the decoder, a minimum bit rate of 1 Mb/s and assuming that 2/3 of bits are being passed to the LDPC code, the overall delay is approximated as 181 ms. Hence, the actual delay is not more than 190 ms.

About 95% of this delay is due to buffer-fills in the LDPC encoder/decoder. For higher bit rates, say 10 Mb/s, this delay is much shorter (approximately 19 ms) and the overall delay does not exceed 30 ms. Here we assume that the channel has the same ratio of coded to uncoded bits and the higher bit rate is due to

a higher channel bandwidth. A longer delay is expected, if the higher bit rate is due to (or partially due to) a higher SNR for the channel.

From Fig. 6, it can be seen that b_2 sees a channel whose capacity is very close to one, which means that a high-rate code can be used. The expected value of this capacity is about 0.986 b/symbol and for b_3 it is more than 0.999 b/symbol. The capacity for higher levels is even greater. There are many different types of codes which can be used for these high-capacity channels. Depending on the decoding complexity that one can afford, the rate of these codes can be close to the capacity. One simple solution is to use Reed–Solomon codes. A (255, 209) Reed–Solomon code, with a rate more than 0.819, can guarantee a frame error rate less than 10^{-7} for b_2 and a (255, 241) Reed–Solomon code, with a rate more than 0.945

$$t_{bf} = \frac{(\text{LDPC code length}) \times (\text{LDPC code rate})}{(\text{system bit rate}) \times (\text{portion of bits assigned to the LDPC code})}$$

can guarantee the same frame error rate for b_3 . Hence, the gap from capacity in these bit-channels is small. Also notice that, due to their low SNR, many subchannels do not have b_2 and b_3 , so some rate loss in these bit-channels does not dramatically impact the overall performance of the system.

The bit-loading algorithm, i.e., the algorithm that chooses the constellation size at a given SNR, affects the capacity of bit-channels. The bit-loading algorithm chooses the constellation size in order to approach the total capacity as closely as possible. The zigzag form of bit-channel capacities in Fig. 6 is due to the bit-loading algorithm, which switches to larger constellations as the SNR increases. At each switch, the capacity of all address bits drops, however the net capacity increases. This can be understood from Fig. 1. The capacity of “bit-channel 0” and “bit-channel 1” in the case of 4-QAM is always greater than that of 8-QAM. However, in the case of 8-QAM, there is an extra bit-channel with a very high capacity which makes the net capacity of 8-QAM greater than the net capacity of 4-QAM.

As a result, the bit-loading algorithm, on the one hand, wishes to switch to higher constellation sizes to approach capacity more closely and, on the other hand, wishes to avoid higher constellation sizes as they may lead to bit-channels with a very poor capacity. Higher constellation sizes may also increase complexity. Hence, the bit-loading algorithm must trade off between complexity and performance. Pseudocode for the bit-loading algorithm that we have used is as follows.

Bit-Loading Algorithm

1. Input: SNR, Threshold
2. set Constellation = MaximumConstellationSize
3. while $\{C(\text{Constellation}, \text{SNR}) - C(\text{Constellation}/2, \text{SNR}) \leq \text{Threshold}\}$
 set Constellation = Constellation/2
4. Output: Constellation

Here the function $C(\text{Constellation}, \text{SNR})$ returns the capacity of a QAM constellation of the given size at the given channel SNR. In our setting, the maximum constellation size is 256-QAM, the threshold is 0.25 b/symbol, and we allow all constellation sizes of the form 2^l down to 2-PAM. As a result, our bit-loading algorithm assigns, at each SNR, the minimum number of bits to each subchannel in a way so that the constellation capacity is less than 0.25 b away from the capacity of the constellation one level higher. In this way, we make sure that we are quite close to the channel capacity with the minimum complexity in terms of constellation size.

Fig. 6 also shows that the $C_{b_0 b_1}$ varies approximately between 0.5 and 0.9 b/symbol for different SNRs. Its average value over all realizations of the channel is about 0.68 b/symbol. Hence, a rate 0.6 LDPC code is used.

In any given channel realization, each subchannel can be viewed as a random sample from the given distribution of SNRs in Fig. 6. Since the number of subchannels is relatively large, the average value of $C_{b_0 b_1}$ for one realization is close to the expected value on all realizations. However, with a

small probability, some realizations of channel can result in an average $C_{b_0 b_1}$ less than 0.6 b/symbol. This is possible since each subchannel, independent of the other subchannels, can have a capacity less than 0.6 b/symbol. To handle these cases with the same fixed-rate LDPC code, we use an inner repetition rate-1/2 code for some subchannels with very low capacity.

For a given realization of the channel, we first check the average of $C_{b_0 b_1}$. If it is less than 0.66, we decide to employ the repetition inner code because the LDPC code may not converge on its own. In every subchannel, if $C_{b_0 b_1}$ is less than a threshold value c_0 (typically 0.5 b/symbol), we repeat those bits. From the LDPC code point of view, this is equivalent to sending a bit over a cleaner channel with a capacity almost doubled. Making the simplifying assumption that the capacity in those subchannels is actually doubled, we compute the average $C_{b_0 b_1}$ over all subchannels again, and if it exceeds 0.7 we reduce the threshold value c_0 ; if it is less than 0.66, we increase the threshold value c_0 ; otherwise we proceed. This repetition algorithm makes the average capacity over all subchannels to be between 0.66–0.7 b/symbol. In this way, we maintain a reasonably small gap from capacity for rate-0.6 LDPC code.

The degree distribution of the LDPC code that we have used in these simulations is $\{\lambda_2 = 0.229, \lambda_3 = 0.234, \lambda_6 = 0.217, \lambda_{15} = 0.320\}$. The code is regular at the check side with check nodes of degree 10. The code is designed under a Gaussian assumption on the messages and using extrinsic information transfer (EXIT) charts based on message error rate [24].

We note that we cannot make the length of the LDPC code an integer multiple of the length of the DMT symbols, because for many channels that change with time (for example, DSL or power-line channels), the length of the DMT symbols vary. To overcome this problem, we simply put as many DMT symbols as possible in the LDPC structure and fill the remaining portion of the LDPC code with zeros. This has no significant effect on the system performance, because the typical size of a DMT symbol (a few hundred bits), compared to the length of the code (100 000 bits), is very small. One may use the fact that the code is zero-padded in order to perform a more efficient decoding, but the improvement is minor.

A. System Complexity

Since the encoding/decoding complexity of the rather long LDPC code dominates the overall complexity of the system, it will be our focus on this section.

The encoding complexity, when the code has some structure, is relatively small in comparison with the decoding complexity as the decoder needs a large number of iterations (about 100 in our case). For example, if the degree two variable nodes are chained to form a structure similar to RA codes [26], the number of operations for encoding will be approximately equal to the number of edges (400 000 in our case), while the decoder needs more than this many operations per iterations. So the main focus of this complexity analysis will be on the decoding complexity.

Following the notation of [22], the update rules of the sum-product algorithm at a check node c and a neighboring variable node v are as follows:

$$\mu_{c \rightarrow v} = 2 \tanh^{-1} \left(\prod_{h \in n(c) \setminus \{v\}} \tanh(\mu_{h \rightarrow c}/2) \right) \quad (5)$$

$$\mu_{v \rightarrow c} = \mu_{ch \rightarrow v} + \sum_{y \in n(v) \setminus \{c\}} \mu_{y \rightarrow v}. \quad (6)$$

In (5), $n(c) \setminus \{v\}$ is the set of all of the neighbors of c except v ; similarly, in (6), $n(v) \setminus \{c\}$ is the set of all of the neighbors of v except c . Here $\mu_{h \rightarrow c}$ represents a message from a variable node h to the check node c , $\mu_{y \rightarrow v}$ represents a message from a check node y to the variable node v , and $\mu_{ch \rightarrow v}$ is the channel message to the variable node v .

From the above equations, it can be shown that at a variable node of degree d_v , $2d_v$ operations are enough to compute all the output messages. Notice that one can add all $d_v + 1$ input messages in d_v operations and then, for every outgoing message, one subtraction is required. Thus, another d_v operations should be done. As a result, the number of operations at the variable side of the code is $2 \sum_v d_v$ or equivalently $2E$, where E is the number of edges in the graph.

Similarly, at a check node of degree d_c , a total of $2(d_c - 1)$ operations is needed. Hence, at the check side of the code the number of operations is $2(E - C)$, where C is the number of check nodes. Notice that (5) can be written as

$$\tanh(\mu_{c \rightarrow v}/2) = \prod_{h \in n(c) \setminus \{v\}} \tanh(\mu_{h \rightarrow c}/2). \quad (7)$$

Hence, having a lookup table or a circuit to map a message m to $\tanh(m/2)$ and back, the situation is similar to a variable node. As a result, a total of $4E - 2C$ operations per iteration is required at the decoder. Considering a maximum of 100 iterations, the number of edges (400 000) and the number of check nodes (40 000), the overall complexity of decoding is 152 Mega operations for 60 000 information bits. That is, 2534 operations per bit. Considering complexities involved with other parts of the system, we estimate an overall complexity of less than 2700 operations per bit.

In our proposed system, we have four coded bits whose overall rate is about $3/4$. The complexity of our system is comparable with the complexity of Viterbi decoding on a 512-state trellis with an underlying rate- $3/4$ code. Such a trellis has 512 state and eight branches leaving each state. At each stage of the trellis, the decoding involves one computation and addition per branch and seven comparisons per state of the trellis. That is 7680 operations per three information bits or 2560 operations per bit.

It has to be mentioned that the hardware complexity of the LDPC decoder is higher than that of the Viterbi decoder as the LDPC decoder needs more memory. However, such comparisons are highly implementation-dependant.

VII. SIMULATION RESULTS

In our simulations, we have assumed that synchronization and channel knowledge at the transmitter and receiver sides are perfect.

TABLE I
SIMULATION RESULTS ON SIX POWER-LINE CHANNELS

Channel	$C_{b_0 b_1}$	R_c	Decoding Iterations	l_b (dB)	ER (bits)	G (dB)
Small	0.69	0.60	90	0.58	0.41	7.51
Medium	0.71	0.60	70	0.68	0.43	7.15
Large	0.69	0.60	80	0.58	0.36	7.56
Medium 1	0.65	0.57	100	0.56	0.37	7.67
Medium 2	0.75	0.60	50	1.15	0.56	6.52
Medium 3	0.57	0.50	100	0.61	0.23	8.09

Table I shows the results of our simulations on six samples of power-line channels. In this table, $C_{b_0 b_1}$ is the average of (3) over all subchannels. R_c is the overall code rate for each specific channel realization. When no repetition is required, the overall rate is 0.6 b/symbol, but, when the average channel capacity is smaller than 0.66 b/symbol, the repetition code is active and the overall rate of the code is less than 0.6 b/symbol.

In this table, l_b represents the gap from the capacity of binary modulation. In other words, assuming a binary input AWGN channel when the capacity is $C_{b_0 b_1}$, l_b is the gap from the Shannon limit considering a rate R_c code is used. As another measure of performance, we define excess redundancy (ER). Excess redundancy in each subchannel is the difference between the capacity of that subchannel and the actual information bits which are sent in it. ER is the average excess redundancy on all subchannels which can be computed as

$$ER = \frac{1}{N} \sum_{i=1}^N \left[\log_2(1 + \text{SNR}_i) - \sum_{j=1}^{n_i} R_{b_{j,i}} \right] \quad (8)$$

where N is the number of subchannels, n_i is the number of bits transmitted in the i th subchannel, SNR_i is the SNR of the i th subchannel, and $R_{b_{j,i}}$ is the code rate used for the j th bit of i th subchannel. ER is a good measure of performance for a coding scheme, because it directly measures the gap from the capacity in bits/symbol.

In many other papers in the coded-modulation literature, coding gain is represented as a measure of performance. In our case, however, since subchannels with different SNRs are used, there is no standard way to compute the coding gain.

However, considering a DMT channel with N subchannels whose capacities are given by C_1 to C_N , and comparing it with a second DMT system whose subchannels all have the same capacity C , we have $C = \sum_{i=1}^N C_i / N$ for the two systems to have equal capacity. Now, assume SNR_i for subchannel i on the first system and SNR_{eff} for all of the subchannels of the second DMT system. Using $C = \log_2(1 + \text{SNR})$, we obtain

$$1 + \text{SNR}_{\text{eff}} = \sqrt[N]{\prod_{i=1}^N (1 + \text{SNR}_i)}.$$

When SNR values are sufficiently large, this can be simplified to

$$\text{SNR}_{\text{eff}} = \sqrt[N]{\prod_{i=1}^N \text{SNR}_i} \quad (9)$$

or, in other words, SNR_{eff} is the geometric mean of the SNR_i 's. Now, comparing an uncoded system with one using coding, if we denote the effective SNR after coding with $\text{SNR}_{\text{c,eff}}$ in decibels and if each subchannel has a coding gain of G_i in decibels, then the required SNR for i th subchannel is $\text{SNR}_i - G_i$ so we have

$$\text{SNR}_{\text{c,eff}} = \text{SNR}_{\text{eff}} - \frac{1}{N} \sum_{i=1}^N G_i$$

so our average coding gain in decibels is $G = (1/N) \sum_{i=1}^N G_i$. This is in decibels which in linear form would be the geometric average of the subchannels coding gain. To find the coding gain in each subchannel, we first compute a gap from the Shannon limit as

$$L_i = \frac{\text{SNR}_i}{2^{\sum_{j=1}^{n_i} R_{b,j,i}} - 1}. \quad (10)$$

The coding gain on this subchannel at a BER of 10^{-7} is $G_i = 10^{0.98}/L_i$, because the uncoded SNR_{norm} for QAM modulation at this BER is 9.8 dB. The effective coding gain is then $G = \sqrt[N]{\prod_{i=1}^N G_i}$. This effective coding gain is also reported in Table I.

It worth mentioning that at high SNRs there is a gap of about 1.53 dB between modulation capacity and the capacity of the Gaussian channel [5]. This can be reduced using shaping techniques [27]–[29]. The gap at lower SNRs is smaller. In a DMT system, which has subchannels with different SNRs, this gap is less than 1.53 dB on average.

In Table I, the first three channels are three power-line test channels derived from buildings with different sizes. The next three channels are random channels from medium-size buildings. The channel frequency response of the first three test channels of the table is shown in Fig. 2.

VIII. CONCLUSION

In this paper, we have proposed a high-performance error-correction system suitable for DMT systems. In our system, we label QAM constellation points using a combined Gray–Ungerboeck labeling scheme. Low-order bits are coded with an irregular LDPC code, which is selectively concatenated with a repetition code to provide an adaptive rate. The rate of the LDPC code is carefully chosen based on the capacity of the corresponding bit channels. Higher order bits are coded with Reed–Solomon codes or are left uncoded. We provide a practical bit-loading algorithm that maintains a balance between constellation complexity and coding efficiency. The decoding complexity of the system is comparable to that of a traditional trellis-coded modulation scheme with 512 states. Using our scheme, performance is achieved equivalent to a gap of approximately 2.3 dB from the Shannon limit without constellation shaping, on a class of channels encountered in power-line communication systems. This gap could be closed to about 1 dB with appropriate shaping methods. Our approach applies not only to power-line channels, but to general frequency-selective channels; hence, we believe that a similar

system can be considered for high SNR wireless channels or DSL channels.

REFERENCES

- [1] T. N. Zogakis, J. T. Aslanis, and J. M. Cioffi, "Analysis of a concatenated coding scheme for a discrete multi-tone modulation," in *Proc. IEEE System Military Communications Conf.*, 1994, pp. 433–437.
- [2] L. Zhang and A. Yongacoglu, "Turbo coding in ADSL DMT systems," in *Proc. IEEE Int. Communications Conf.*, vol. 1, 2001, pp. 151–155.
- [3] Z. Cai, K. R. Subramanian, and L. Zhang, "DMT scheme with multidimensional turbo trellis code," *Electron. Lett.*, pp. 334–335, Feb. 2000.
- [4] *Asymmetric Digital Subscriber Line (ADSL) Metallic Interface*, 1995. ANSI Standard T1E1.4/95-007R2.
- [5] G. D. Forney and G. Ungerboeck, "Modulation and coding for linear Gaussian channels," *IEEE Trans. Inform. Theory*, vol. 44, pp. 2384–2415, Oct. 1998.
- [6] C. Berrou, A. Glavieux, and P. Thitimajshima, "Near Shannon limit error-correcting coding and decoding: Turbo-codes," in *Proc. IEEE Int. Communications Conf.*, 1993, pp. 1064–1070.
- [7] D. Divsalar and F. Pollara, "On the design of turbo codes," Jet Propulsion Lab., Pasadena, CA, TDA Progr. Rep., 1995.
- [8] T. J. Richardson and R. L. Urbanke, "The capacity of low-density parity-check codes under message passing decoding," *IEEE Trans. Inform. Theory*, vol. 47, pp. 599–618, Feb. 2001.
- [9] T. J. Richardson, A. Shokrollahi, and R. L. Urbanke, "Design of capacity-approaching irregular low-density parity-check codes," *IEEE Trans. Inform. Theory*, vol. 47, pp. 619–637, Feb. 2001.
- [10] E. Eleftheriou and S. Ölçer, "Low-density parity-check codes for digital subscriber lines," in *Proc. IEEE Int. Communications Conf.*, 2002, pp. 1752–1757.
- [11] T. Cooklev, M. Tzannes, and A. Friedman, "Low-density parity-check coded modulation for ADSL," ITU-Telecommun. Standardization Sector, Temp. Doc. BI-081, Oct. 2000.
- [12] U. Wachsmann, R. F. H. Fischer, and J. B. Huber, "Multilevel codes: Theoretical concepts and practical design rules," *IEEE Trans. Inform. Theory*, vol. 45, pp. 1361–1391, July 1999.
- [13] G. D. Forney, M. D. Trott, and S.-Y. Chung, "Sphere-bound-achieving coset codes and multilevel coset codes," *IEEE Trans. Inform. Theory*, vol. 46, pp. 820–850, May 2000.
- [14] G. Caire, G. Taricco, and E. Biglieri, "Bit-interleaved coded modulation," *IEEE Trans. Inform. Theory*, vol. 44, pp. 927–946, May 1998.
- [15] T. Esmailian, F. R. Kschischang, and P. G. Gulak, "In-building power lines as high speed communication channels: Channel characterization and a test channel ensemble," *Int. J. Commun. Syst.*, vol. 16, pp. 381–400, May 2003.
- [16] H. Imai and S. Hirakawa, "A new multilevel coding method using error correcting codes," *IEEE Trans. Inform. Theory*, vol. IT-23, pp. 371–377, May 1977.
- [17] G. Ungerboeck, "Channel coding with multilevel/phase signals," *IEEE Trans. Inform. Theory*, vol. IT-28, pp. 55–67, Jan. 1982.
- [18] G. J. Pottie and D. P. Taylor, "Multilevel codes based on partitioning," *IEEE Trans. Inform. Theory*, vol. 35, pp. 87–98, Jan. 1989.
- [19] A. R. Calderbank, "Multilevel codes and multistage decoding," *IEEE Trans. Commun.*, vol. 37, pp. 222–229, Mar. 1989.
- [20] D. J. C. MacKay, "Good error-correcting codes based on very sparse matrices," *IEEE Trans. Inform. Theory*, vol. 45, pp. 399–431, Mar. 1999.
- [21] A. Shokrollahi, "Capacity-achieving sequences," in *Codes, Systems, and Graphical Models*, no. 123 of IMA Volumes in Mathematics and Its Applications, B. Marcus and J. Rosenthal, Eds. New York: Springer-Verlag, 2000, pp. 153–166.
- [22] F. R. Kschischang, B. J. Frey, and H.-A. Loeliger, "Factor graphs and the sum-product algorithm," *IEEE Trans. Inform. Theory*, vol. 47, pp. 498–519, Feb. 2001.
- [23] M. G. Luby, M. Mitzenmacher, M. A. Shokrollahi, and D. A. Spielman, "Efficient erasure correcting codes," *IEEE Trans. Inform. Theory*, vol. 47, pp. 569–584, Feb. 2001.
- [24] M. Ardakani and F. R. Kschischang, "Designing irregular LDPC codes using EXIT charts based on message error rate," in *Proc. IEEE Int. Symp. Inform. Theory*, Lausanne, Switzerland, June 2002, p. 454.
- [25] E. Eleftheriou and S. Ölçer, "Low-density parity-check codes for multi-level modulation," in *Proc. IEEE Int. Symp. Inform. Theory*, Lausanne, Switzerland, June 2002, p. 442.

- [26] H. Jin, A. Khandekar, and R. McEliece, "Irregular repeat-accumulate codes," in *Proc. 2nd Int. Symp. on Turbo Codes and Related Topics*, Brest, France, Sept. 2000.
- [27] A. K. Khandani and P. Kabal, "Shaping multidimensional signal space—Part I: Optimum shaping, shell mapping," *IEEE Trans. Inform. Theory*, vol. 39, pp. 1799–1808, Nov. 1993.
- [28] F. R. Kschischang and S. Pasupathy, "Optimal nonuniform signaling for Gaussian channels," *IEEE Trans. Inform. Theory*, vol. 39, pp. 913–929, May 1993.
- [29] —, "Optimal shaping properties of the truncated polydisc," *IEEE Trans. Inform. Theory*, vol. 40, pp. 892–903, May 1994.



Masoud Ardakani (S'01) received the B.Sc. degree in electrical engineering from Isfahan University of Technology, Isfahan, Iran, in 1994, and the M.Sc. degree in electrical engineering from University of Tehran, Tehran, Iran, in 1997. He is currently working toward the Ph.D. degree at the University of Toronto, Toronto, ON, Canada.

From 1997 to 1999, he was with the Electrical and Computer Engineering Research Center, Isfahan, Iran. His research interests are in the general area of digital communications, error-control coding,

especially codes defined on graphs associated with iterative decoding, and MIMO systems.

Mr. Ardakani was the recipient of a number of scholarships and awards including the Edward S. Rogers Sr. Scholarship while at the University of Toronto.



Tooraj Esmailian received the B.Sc. degree from Isfahan University of Technology, Isfahan, Iran, in 1992, the M.Sc. degree from Sharif University of Technology, Tehran, Iran, in 1995, and the Ph.D. degree from the University of Toronto, Toronto, ON, Canada, in 2002, all in electrical engineering.

While at the University of Toronto, he was supported by a grant from Motorola and received an international student award from the University of Toronto in 1998, a University of Toronto open fellowship in 1999, and Ontario graduate scholarships

in 2000 and 2001. From 2002 to 2003, he was with the research and development group of Cogency Semiconductor, Inc., Kanata, ON, Canada, working on Home-Plug-based power line products. Since February 2004, he has been with the Research and Development Group, Edgewater Computer Systems, Inc., Kanata, ON, Canada. His research interests are in the general area of digital communications, especially multicarrier modulation, error control coding, and channel modeling.



Frank R. Kschischang (S'83–M'91–SM'00) received the B.A.Sc. degree (with honors) from the University of British Columbia, Vancouver, BC, Canada, in 1985 and the M.A.Sc. and Ph.D. degrees from the University of Toronto, Toronto, ON, Canada, 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 spent a sabbatical year as a Visiting Scientist at the Massachusetts Institute of Technology (MIT), Cambridge. 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. He has taught graduate courses in coding theory, information theory, and data transmission.

Dr. Kschischang was the recipient of the Ontario Premier's Research Excellence Award. From October 1997 to October 2000, he served as an Associate Editor for Coding Theory for the IEEE TRANSACTIONS ON INFORMATION THEORY. He also served as technical program co-chair for the 2004 IEEE International Symposium on Information Theory held in Chicago, IL.