LDPC Code Design for Half-Duplex Decode-and-Forward Relaying

Arnab Chakrabarti, Alexandre de Baynast, Ashutosh Sabharwal, and Behnaam Aazhang
Department of Electrical and Computer Engineering
Rice University, Houston, Texas 77005
Email: \{arnychak,debaynas,ashu,aaz\}@rice.edu

Abstract

We propose LDPC code designs for the half-duplex relay channel. Our designs mimic the information theoretic random coding scheme for decode-and-forward relaying. An important advantage of our scheme is that it is built entirely of single-user codes that can be decoded by belief propagation. The optimization of relay LDPC code profiles presents unique challenges, which are met by using the density evolution algorithm with additional constraints for relaying. To speed up our optimization, we use a Gaussian approximation of density evolution that converts the infinite dimensional code profile optimization into a simple linear programming problem. The thresholds of the discovered relay code profiles are within 0.4 dB of the achievable lower bound for decode-and-forward relaying.

1 Introduction

We propose LDPC code designs for the half-duplex relay channel. Our work is motivated by the observation that half-duplex relaying - where the relay cannot simultaneously transmit and receive in the same band - yields substantially higher rate than both direct and two-hop communication with the same average power constraint. We use LDPC codes as component codes because they have several advantages: provably good performance, efficient decoding using belief propagation, and parallelizable decoder structure.

Our code designs are based on the information theoretically optimal random-coding scheme, and provide insight into relay coding. We use component LDPC codes of appropriate rates in our designs. Although the relay channel is commonly visualized as a combination of a broadcast and a multiple-access channel, we show that the achievable rate of decode-and-forward relaying can be approached by using single-user codes decoded with single-user receivers. The single-user decodability of our codes supports the practicality of half-duplex relaying.

Even though the relay coding strategy proposed by us is built entirely on single-user codes, the problem of optimizing LDPC code profiles does not follow trivially from what is already known for single-user codes. Connections between the factor graphs of the component LDPC codes introduces unique challenges for relay LDPC code profile optimization. We address these challenges by imposing additional constraints on the profile optimization problem involving density evolution. We also use the Gaussian Approximation scheme proposed in [1] with suitable changes to yield a linear-programming formulation for rapidly estimating approximate noise thresholds and discovering good relay code profiles.

The information theoretic relay channel was first studied by van der Meulen [2] in 1971. A few years later, several fundamental capacity results on relaying were published in [3]. After the initial interest, however, the idea of relaying received little attention for nearly two decades. Recent years

*This work was supported by Nokia.
have seen a renewed interest in relaying in the context of wireless networks. In [4], the feasibility of user cooperation in a wireless network was demonstrated by an information theoretic exposition of the gains, and a practical CDMA implementation. New information theoretic results were found in [5, 6, 7, 8]. The study of relay protocols and their outage analysis in fading environments was done in [9, 10]. As a practical alternative to conventional decode-and-forward and amplify-and-forward relay protocols, in [11, 12], the authors investigated relaying through coded cooperation. It is worth noting that several of the above research contributions have been based on the premise of half-duplex relaying [7, 8, 11, 10, 9].

The rest of this paper is organized as follows. Section 2 describes our setup and assumptions. In Section 3, we briefly review information theoretic rates for direct, two-hop and relay channels. Section 4 introduces LDPC codes, then moves on to the relay LDPC code structure. Numerically determined thresholds and simulation results are presented in Section 5. Finally, we conclude in Section 6.

![Figure 1: The relay channel](image)

**2 System description**

The relay channel is shown in Fig. 1 (a). The source (S) sends data to the destination (D), aided by the relay (R). In the particular case of a half-duplex relay channel, the relay cannot transmit and receive simultaneously in the same band. Communication takes place over two slots of time fractions $t$ and $(1-t)$. In the first slot, S transmits information, which is received by R and D. We will call this the BC (broadcast) mode of communication. In the second slot, both S and R transmit information to D. This, we will refer to as the MAC (multiple access) mode. These two modes are depicted in Fig. 1 (b).

We compare half-duplex relaying with both direct and two-hop communication. For fair comparison, the middle node is assumed to operate in a half-duplex fashion in the two-hop channel. In the context of a two-hop channel also, we use the terms BC and MAC mode to denote the phases in which the relay receives and transmits respectively.

We normalize transmitted power with noise power, which is assumed to be the same at each receiver. An average global transmission power constraint is imposed on the nodes, denoted by $\Theta$,

$$\Theta : tP_{S_{BC}} + (1-t)(P_{S_{MAC}} + P_{R_{MAC}}) \leq P \quad (1)$$

where $P$ represents the total system transmission power \(^1\), $P_{S_{BC}}$ is source transmission power in BC mode, $P_{S_{MAC}}$ is source transmission power in MAC mode, and $P_{R_{MAC}}$ is relay transmission power in MAC mode. In the two-hop case, the same power constraint holds with $P_{S_{MAC}} = 0$. For the direct link, the above power constraint is true with $t = 1$.

\(^1P\) is also the equivalent relay channel SNR in our plots.
All channels are assumed to be 1-D AWGN channels. The distance between S and D is normalized to unity, and R is assumed to lie on the straight line joining S and D. The relay position, denoted \(d\), represents its distance from the source. The collinearity of S,R and D is not used in deriving any of our results, but it enables a simple 1-D characterization of the relay position. The SD channel gain is given by \(\gamma_{SD} = 1\), the SR gain is \(\gamma_{SR} = \frac{1}{d}\), and the RD gain is \(\gamma_{RD} = \frac{1}{(1-d)^\lambda}\), where \(\lambda\) is the channel attenuation exponent. We use \(\lambda = 2\) in this paper.

3 Review of information theoretic results

For brevity, we omit the capacity and achievable rate expressions for direct, two-hop, and half-duplex relay channels. The reader will find a summary of relevant results in [13, 14]. We only mention the achievable rate of decode-and-forward half-duplex relaying, since we will refer to it often.

\[
R \geq \sup_{0 \leq t \leq 1} \min \{tI(X_1;V_1) + (1-t)I(X_2;Y_2|W_2), tI(X_1;Y_1) + (1-t)I(X_2,W_2;Y_2)\} \tag{2}
\]

where \(X,W\) are respectively source and relay transmitted signals and \(V,Y\) are relay and destination received signals (shown in Fig. 1). Subscripts 1 and 2 denote BC and MAC mode respectively. There are other achievable rates corresponding to amplify and forward and estimate and forward (also called compress and forward) protocols, but we will not consider them in this paper.

The capacities of the direct link and two-hop channels are compared with the achievable rate of the half-duplex relay channel in Fig. 3(a). Rates are plotted for Gaussian as well as BPSK signaling. The relay position is \(d = 0.5\). At low SNR, the performance of relaying is close to that of two-hop communication, whereas at high SNR, relaying is closer to direct communication. In the intermediate SNR range, half-duplex relaying affords sizeable gains. A maximal relaying gain of nearly 40% over both direct and two-hop links is obtained at 3dB SNR with Gaussian signaling. With BPSK, a maximum 35% relaying gain is obtained at -1dB. \(^3\)

The reader will observe from Fig. 3(a) that relaying gain is most significant in the SNR range around 3 dB, the exact range being dependent on relay position and other parameters. At high

\(^2\)An extension to circularly symmetric AWGN is straightforward.

\(^3\)Note that the rates in Fig. 3 correspond to normalized distances not exceeding unity. Translated to a practical network, the same rates would correspond to higher and more realistic SNRs.
SNR, the ratio of relay throughput to direct link throughput approaches one. We also observe that relaying with BPSK can outperform both direct and two-hop Gaussian capacities at SNRs as high as 4dB. This encourages us to consider binary codes for the half-duplex relay channel.

We assume that the two rate terms in the achievable rate expression (2) are equal at the point where the achievable rate is maximized, i.e.

$$tI(X_1;V_1) + (1-t)I(X_2;Y_2|W_2) = tI(X_1;Y_1) + (1-t)I(X_2,W_2;Y_2).$$

We have found the above to be empirically true, but a proof is difficult due to the nature of our sum power constraint. Even if this is not true, it is easy to see that we can find rates $$R_1 \leq I(X_1;V_1), R_2 \leq I(X_2;Y_2|W_2), R_3 \leq I(X_1;Y_1),$$ and $$R_4 \leq I(X_2,W_2;Y_2)$$ satisfying

$$\text{Achievable rate} = tR_1 + (1-t)R_2 = tR_3 + (1-t)R_4.$$  

The proposed coding scheme then remains the same with the mutual information terms replaced by the corresponding rates.

Before we suggest a practical coding scheme, there is one important challenge to be met. In MAC mode, the source and relay signals are, in general, correlated to maximize the achievable rate. This optimum correlation between source and relay signals can take any value between 0 and 1. However, it is difficult to build pairs of codebooks with arbitrary correlation. To that end, the authors have shown in [13] that there is negligible rate loss when we intelligently choose either $$\rho = 0$$ or 1, depending on which one gives a better rate, instead of the optimum $$\rho$$. This is demonstrated in Fig. 3(b), which plots the achievable rates for both BPSK and Gaussian signaling for optimum $$\rho$$ as well as for $$\rho = 0, 1$$. Therefore, we will propose codes only for $$\rho = 0, 1$$.

4 LDPC code design for half-duplex relaying

We first present a brief introduction to LDPC coding. Then we describe the structure of half-duplex relay LDPC codes.

4.1 A brief introduction to binary LDPC codes

A binary LDPC code is a linear block code with a sparse parity-check matrix. This $$n \times m$$ matrix is equivalently represented by a bipartite graph with $$n$$ variable nodes corresponding to rows (bits in the codeword) and $$m$$ check nodes corresponding to columns (parity check equations). A one in the parity check matrix corresponds to an edge between the corresponding variable and check node in the graph, whereas a zero indicates no edge.

A regular LDPC code is one in which all nodes of the same type have the same degree. In an irregular code, there is no such constraint. An LDPC code is characterized by a pair of generating functions

$$\lambda(x) = \sum_{i=2}^{d_v} \lambda_i x^{i-1} ; \quad \rho(x) = \sum_{i=2}^{d_c} \rho_i x^{i-1}$$

where $$\lambda_i$$ ($$\rho_i$$) denotes the fraction of edges connected to a variable (check) node of degree $$i$$, and $$d_v$$ ($$d_c$$) is the maximum number of edges connected to any variable (check) node. For example, a (3,6) regular LDPC would have $$\lambda(x) = x^2$$ and $$\rho(x) = x^5$$. All LDPC codes with the same $$\lambda$$ and $$\rho$$ are said to belong to the same ensemble. In their seminal work [15], Richardson and Urbanke showed that in the limit of infinite block length, nearly all codes in an ensemble have near identical error
performance, thereby justifying this way of representing entire ensembles instead of individual codes. The design rate of the code is given in terms of $\lambda(x)$ and $\rho(x)$ by

$$R = 1 - \frac{m}{n} = 1 - \frac{\int_0^1 \rho(x)dx}{\int_0^1 \lambda(x)dx}$$

(6)

4.2 Decoding algorithms

LDPC codes can be decoded by a variety of message passing algorithms, of which we will consider only belief propagation [15]. In a message passing decoder, messages are passed iteratively from variable node to check node and back until decoding is successful or a maximum number of iterations is reached. A message sent from a node on one of its edges is calculated based on messages received by that node on all edges except the one on which the current message is being sent. In belief propagation, the message sent on an edge represents the a posteriori log likelihood ratio (LLR) of the variable bit connected to that edge based on received messages. Iterations start at variable nodes, with the messages initialized to LLRs based on channel received values. The belief propagation algorithm calculates the a posteriori LLRs for each variable exactly if the graph is cycle free. Although all interesting parity check matrices have cycles, the LLRs produced by belief propagation are exact up to $l$ iterations if the length of the smallest cycle is greater than $2l$. For the sake of analysis, we assume the graph to be free of cycles that compromise the accuracy of belief propagation.

Let $v$ denote the message passing from a variable to a check node and $u$ denote the message in the opposite direction. Therefore,

$$v = \log \frac{p(y|x = 1)}{p(y|x = -1)}$$

(7)

where $x$ is the bit value of the variable node and $y$ denotes all received information up to the present iteration on edges other than the one carrying $v$. Similarly,

$$u = \log \frac{p(y'|x' = 1)}{p(y'|x' = -1)}$$

(8)

where $x'$ is the bit value and $y'$ denotes information from all other edges up to the current iteration. The update rule for $v$ is given by

$$v = u_0 + \sum_{i=1}^{j-1} u_i$$

(9)

where $u_0$, the initial message, is the LLR conditioned on the channel output, the $u_i$ are the incoming LLRs, and $j$ is the number of check nodes connected to the variable node in question. Similarly, the update rule for $u$ is

$$\tanh \frac{u}{2} = \frac{k-1}{\prod_{i=1}^{k} \tanh \frac{v_i}{2}}$$

(10)

where $v_i$ $i = 1, ..., k-1$ are the incoming LLRs and $k$ is the number of edges incident on the check node. Using an equivalent sign-magnitude representation, where positive is 0 and negative is 1, (10) is written as

$$\text{sign}(u) = \left( \sum_{i=1}^{k-1} \text{sign}(v_i) \right) \mod(2); \quad \log \left| \tanh \frac{u}{2} \right| = \sum_{i=1}^{k-1} \log \left| \tanh \frac{v_i}{2} \right|$$

(11)

Since the update rules use addition of the incoming messages, the outgoing message probability density function is a convolution of the incoming message densities. Using these update rules, it is
possible to track message densities and predict the number of erroneous bits at different iterations - a process known as density evolution.

LDPC codes demonstrate a threshold phenomenon, i.e. in the limit of infinite block length and infinite iterations, LDPC codes within a given ensemble can be decoded with probability 1 if and only if the noise variance is below a threshold. This was first observed by Gallager [16]. This threshold can be determined numerically using density evolution for a broad class of channels including the BEC, BSC and BAWGN channels [15]. Consequently, it is possible to find good codes using density evolution, i.e. search for a code profile for a given design rate, which will yield a high noise threshold.

Unfortunately, tracking entire densities over thousands of iterations is computationally expensive. To reduce the computational burden of threshold determination, Chung et al. [1] show that by approximating the message densities as Gaussians, a fair approximation to density evolution is obtained. Gaussian approximation reduces the infinite dimensional problem of tracking entire densities to a one-dimensional problem of tracking means. We use the Gaussian approximation to find good code profiles and to obtain approximate thresholds for relay channel codes.

A Gaussian is completely specified by its mean and variance. Moreover, it was pointed out in [17] that message densities satisfy a symmetry condition in density evolution given by \( f(x) = f(-x)e^x \), where \( f(x) \) is the density of the LLR message. For a Gaussian density with mean \( m \) and variance \( \sigma^2 \), this condition yields the relation \( \sigma^2 = 2m \), which means that we only need to track the mean at every iteration. For brevity, we present the procedure for determining the threshold without the steps in its derivation, which can be found in [1]. For a given \( \rho(x) \) and \( \lambda(x) \), and channel noise variance \( \sigma_c^2 \), decoding succeeds in the limit of infinite blocklength and infinite iterations if

\[
 r > h(s, r) \quad \forall r \in (0, \phi(s)) \quad ; \quad h(s, r) = \sum_{i=2}^{d_1} \lambda_i \phi \left( s + (i - 1) \sum_{j=2}^{d_1} \rho_j \phi^{-1}(1 - (1 - r)^{j-1}) \right) \tag{12}
\]

with \( s = 2/\sigma_c^2 \), where \( \sigma_c^2 \) is the channel noise variance, and

\[
 \phi(x) = \begin{cases} 
 1 - \frac{1}{\sqrt{4\pi}} \int_R \tanh \frac{u}{4x} du & \text{if } x > 0 \\
 1 & \text{if } x = 0 
\end{cases} \tag{13}
\]

For numerical purposes, we use the following approximations [1] for the function \( \phi \).

\[
 \phi(x) \approx \begin{cases} 
 e^{(-0.4527x^{0.86} + 0.0218)} & \text{if } x \in [0, 10] \\
 \sqrt{\frac{x}{2e}} \left( 1 - \frac{20}{x^2} \right) & \text{if } x > 10 
\end{cases} \tag{14}
\]

We use (12) to find the largest \( \sigma_c^2 \) for which decoding is successful.

Finding the optimum degree distribution for a given rate is a search for the \((\lambda(x), \rho(x))\) pair that yields the largest noise threshold. For single-user codes, it has been empirically observed that good check node distributions are concentrated, i.e. all parity check nodes should have equal or nearly equal degrees. This has also been argued through analysis in [18, 1]. The variable node distribution, on the other hand, is not as easy to characterize. Therefore, the search for a good \((\lambda(x), \rho(x))\) pair for a single user code is carried out in practice by choosing several concentrated \( \rho(x) \) and finding the best \( \lambda(x) \) for each, followed by picking the \( \rho(x) \) for which the best noise threshold was obtained. An advantage of this approach is that the best \( \lambda(x) \) for a given \( \rho(x) \) can be found by linear programming. The following is a brief description of this procedure.

We are interested in finding \( \lambda(x) \) which maximizes the noise threshold \( \sigma_c^2 \) for a given rate and \( \rho(x) \). Instead, we solve the equivalent problem of finding \( \lambda(x) \) which will maximize the rate for a given noise variance and \( \rho(x) \), and slowly increasing the variance until the design rate is barely achievable.
From (6), we see that for fixed $\rho$, maximizing rate is equivalent to maximizing $\int_0^1 \lambda(x)dx = \sum_{i=2}^{d_0} \frac{\lambda_i}{i}$. The constraints are $\lambda(1) = 1$, and (12), which are both linear. This linear program with inequality constraints can be solved quickly and accurately using available optimization tools.

We conclude our brief summary of LDPC codes, density evolution and Gaussian approximation here. The interested reader will find extensive material on these in [16, 19, 18, 15, 17, 1].

4.3 LDPC code structure for half-duplex relaying

As mentioned before, we design LDPC codes for source-relay correlations of 0 and 1. We will explain both designs in this section. We assume that the sum length of BC and MAC mode codewords is $N$ bits.

When $\rho = 1$, S and R transmit identical signals in MAC mode. For this case, we propose the following scheme. In BC mode, S encodes information bits using a code $\text{LDPC}_{SR-BC}$ to yield a codeword of length $tN$ bits. This codeword can be successfully decoded by R, but not by D. In the beginning of MAC mode, the $tN$ variable bits from BC mode are compressed. Compression is done at both S and R, by multiplying with the same parity matrix. These compressed bits, acting as parity together with the parity bits of $\text{LDPC}_{SR-BC}$ form a composite code $\text{LDPC}_{SD-BC}$ that can be decoded at D at the end of MAC mode. In order to communicate the compressed bits to D reliably, S and R treat them as information bits for MAC mode, and re-encode them using a code $\text{LDPC}_{MAC}$ to yield a codeword of length $(1-t)N$, which is then transmitted synchronously from S and R with appropriate powers.

For $\rho = 1$, decoding is performed as follows. R decodes $\text{LDPC}_{SR-BC}$ using belief propagation like any single-user LDPC code. D waits for both BC and MAC mode signals to arrive before it commences decoding. $\text{LDPC}_{MAC}$ is decoded like a single-user LDPC code, from which side information in the form of additional parity bits is obtained about the BC mode signal. Using knowledge of the single-user BC mode source-relay code, and with the help of these additional parity bits, $\text{LDPC}_{SD-BC}$ is decoded. This final decoding also is performed using belief propagation.

For $\rho = 0$, the BC mode is the same as before. In MAC mode, however, S and R transmit independent (therefore uncorrelated) information using codes $\text{LDPC}_{SD-MAC}$ and $\text{LDPC}_{RD-MAC}$ respectively. As before, R compresses the information bits received in BC mode to produce additional parity bits, which serve as relay information bits in MAC mode. These bits are re-encoded by R using $\text{LDPC}_{RD-MAC}$ to yield $(1-t)N$ coded bits. The source, in MAC mode, sends bits of new information encoded using $\text{LDPC}_{SD-MAC}$ to yield another set of $(1-t)N$ coded bits. Thus, $(1-t)N$ coded bits each from S and R are transmitted simultaneously with appropriate power allocation, so that the two (uncorrelated) signals appear superimposed at D.

For $\rho = 0$, decoding proceeds as follows. R decodes the BC mode signal like a single-user LDPC code. D waits for both BC and MAC mode signals. In MAC mode, the rates for SD and RD channels correspond to one of the corner points of the MAC capacity region, for which it is well known [20] that capacity can be achieved by successive decoding (onion-peeling). The MAC signal is successively decoded to first reveal the relay codeword, treating both noise and interference from S as noise. Next, the relay codeword is subtracted out to reveal the source codeword in the presence of noise alone, which is then decoded. The MAC mode source information is new information, whereas the relay information provides additional parity bits to aid in decoding the BC mode codeword.

The main challenge is the design of codes $\text{LDPC}_{SD-BC}$ and $\text{LDPC}_{SR-BC}$, which must be jointly optimized, since the factor graph of the latter is a subgraph of the factor graph of the former. The reader should note that these codes are of different rates, and although the received codeword is same for both R and D, the received SNRs are different. To avoid confusion, we would like to mention

\footnote{We are unable to disclose certain details at this time due to intellectual property considerations.}
that neither S, nor R actually uses $LDP_{SD-BC}$ to encode information. It is merely a convenience to visualize the side information received by D in MAC mode as extra parity bits in addition to the actual parity bits of $LDP_{SR-BC}$, and call the composite a code $LDP_{SD-BC}$. The design of the MAC mode LDPC codes poses no new challenge, therefore we will not discuss them here. The optimization of code profiles is performed using a modification of the density evolution algorithm. In the implementation of density evolution, the messages have been approximated as Gaussians to speed up the optimization, the cost being usually small inaccuracy in threshold determination.

5 Results

We will present numerically calculated thresholds, as well as simulated relay code performance comparisons in this section.

![Rate vs $E_b/N_0$. Theoretical limits and LDPC performance based on thresholds.](image1)

![BER performance of LDPC constituent codes in relay coding scheme.](image2)

(a) Rate vs. $E_b/N_0$. Theoretical limits and LDPC performance based on thresholds.

(b) BER vs. $E_b/N_0$ for LDPC codes which take part in the relay coding scheme ($P = 1, \rho = 1$).

As mentioned before, the main challenge is the optimization of $LDP_{SR-BC}$ and $LDP_{SD-BC}$ profiles. We restrict the maximum variable degree to 25, and the maximum check degree to 50 for both codes. In addition, to reduce the search space, we impose the following restriction on check degrees $\rho_{SR-BC}$ and $\rho_{SD-BC}$. We will search for good codes with $\rho_{SR-BC}$ concentrated at a single degree $j$ and $\rho_{SD-BC}$ supported on two degrees $i$ and $j$ with $i < j$. Thus,

$$\rho_{SR-BC}(x) = x^{j-1}$$

$$\rho_{SD-BC}(x) = \frac{ax^{i-1} + bx^{j-1}}{(a + b)}$$

$$a = i(I(X_1; V_1) - I(X_1; Y_1)); b = j(1 - I(X_1; V_1)).$$

There is no theoretical justification for reducing our search space in this way, but the thresholds of the codes that we discover show that good code pairs can be obtained with this constraint.

Relay coding makes use of several single-user component LDPC codes. For $\rho = 1$, the codes are $LDP_{SR-BC}$, $LDP_{SD-BC}$ and $LDP_{MAC}$, and for $\rho = 0$, the codes are $LDP_{SR-BC}$, $LDP_{SD-BC}$, $LDP_{SD-MAC}$, and $LDP_{RD-MAC}$. The MAC mode codes are all single-user codes, for which we use the best profile available in [21]. If codes of the exact rate that we need are unavailable, then we use codes of the next higher rate. The same is true for the single-user codes that we compare with in the context of direct and two-hop channels. Fig. 5(a) shows how far the codes are from their corresponding theoretical bounds. In this figure, we fix the rates of the codes,
and calculate $E_b/N_0$ from the numerically calculated thresholds of the LDPC codes. For the relay channel, the achievable $E_b/N_0$ values are less than 0.4 dB away from the theoretical minimum. Note that the thresholds for the relay channel are not thresholds of any single code, but a function of the thresholds of all component codes.

Fig. 5(b) plots the BER performance of LDPC codes that constitute the relay coding scheme. The total power is $P = 1$ and correlation $\rho = 1$ in this case. The codes $LDPC_{SR-BC}$ and $LDPC_{MAC}$ are about 1dB away from the threshold for a BER of $10^{-4}$ whereas $LDPC_{SR-BC}$ is about 2dB away from the threshold. All BER performances are based on codes of blocklength 10000. The codes are randomly constructed and we perform no cycle removal with the exception of removing double edges. The simulated BER performance is likely to improve considerably if the codes are carefully constructed and have much larger blocklengths. Efforts are currently being made in this direction.

In all our simulations, we assume that the all-zero codeword is transmitted, with no loss of generality. Even in the case of $\rho = 0$, this is permissible, with the exception of modeling MAC interference. A simple, yet accurate way to simulate MAC interference is to assume that the RD codeword is the all-zero codeword, whereas the interference from S is a random bit stream in which 0’s and 1’s occur with equal probability (since the codebook has equal proportion of 0’s and 1’s).

6 Conclusions and future directions

We have discovered LDPC code designs and a profile optimization technique for the half-duplex relay channel. The thresholds determined using Gaussian approximation are close to theoretical limits. In future, we propose to discover better relay code profiles by removing the restrictions that we imposed on check degrees to speed up the discovery of good profiles. Efforts are also underway to perform optimization using an exact implementation of density evolution, since the Gaussian approximation is sometimes inaccurate for low rate codes, and also for codes with large fractions of variable nodes with degrees 2, 3. We are also trying to discover codes with BER performance much closer to the thresholds than the currently published results by more careful construction of parity matrices. Finally, our techniques will be extended to the design of relay codes for other relay protocols also.

References


