Optimization of LDPC codes for UEP channels

Charly Poulliat, David Declercq and Inbar Fijalkow
ETIS laboratory
UMR 8051- ENSEA/University of Cergy-Pontoise/CNRS
E-mail:{poulliat,declercq,fijalkow}@ensea.fr

I. INTRODUCTION

The framework of the paper is the optimization of LDPC codes for Unequal Error Protection (UEP) transmission schemes. Such transmission schemes are used to take into account the different error sensitivities of the source bitstream. For example, semantic information like headers have to be almost error free to avoid the crash of the source decoding, whereas the data symbols can tolerate a given bit error rate. UEP can enhance significantly the performance compared to an equal error protection scheme since the overall redundancy is allocated in order to provide more protection to the most sensitive parts of the bitstream.

Apart from multilevel coded modulations, there are commonly two ways to provide unequal protection depending on the transmission scheme we consider. First, for adaptive transmissions, one can adapt the protection level by adaptively changing the code rate through puncturing. RCPC [7], RCPT [1] or punctured IRA [8] and LDPC [6] codes are some examples. These scheme are implemented using a unique mother code, which is punctured depending on the required protection level. The second way is to design codes providing inherent unequal error protection within a codeword. This was early studied in ([2], [10]) to design linear UEP codes. For such codes, the bits within a codeword do not have the same protection level and error correction capability. Thus for a given source codeword, the most sensitive bits are associated to information bit positions with the highest protection level. Irregular LDPC codes can provide inherent unequal error capability due to the different connection degrees of the data nodes. As irregularly, it appears that highly connected nodes are "more" protected than weakly connected ones. This inherent irregularity can be used to provide UEP, in the sense that high sensitivity bits are associated with high connection degrees data nodes [5]. We choose therefore to deepen UEP with LDPC codes.

The optimization of the LDPC irregularity has already been proposed for different channels as the Binary Erasure Channel [9] or the Additive White Gaussian Noise (AWGN) channel [4]. However, the existing optimization techniques do not ensure that the codes would achieve the best UEP. We show in this paper that by modeling the UEP transmission scheme as a specific channel, it is possible to optimize the LDPC irregularity, in order to get the best UEP capability. The transmission scheme consisting in a UEP-coded bitstream sent over a channel X will be called UEP-X channel. In this paper, only the UEP-AWGN channel is considered. We will show that, for a given check node distribution, the UEP-optimized LDPC code has better performance, in terms of bit error probability associated with a given error sensitivity at a finite number of iterations, than the code optimized for the AWGN channel.

The paper is organized as follows. In section II, we describe the considered UEP-AWGN channel and give a general description of the LDPC optimization problem. In section III, we explain the used notations and propose an optimization method based on a hierarchical optimization with respect to the different sensitivity classes. Simulation results for finite length codes are given in section IV and conclusions and perspective are drawn in section V.

II. UEP CHANNEL DESCRIPTION AND OPTIMIZATION STRATEGY

A. The UEP channel

Let a channel codeword of a rate $R$ LDPC code divided into $N_c$ classes ordered in decreasing order of their error sensitivity. Thus considering the set of $N_c$ classes $\{C_k/k = 1 \ldots N_c\}$, $C_1$ will be associated with the highest protected level and $C_{N_c}$ with the lowest. The redundancy bits of the channel codeword are associated with class $C_{N_c}$ and the information bits are associated with the $(N_c-1)$ remaining classes. Let the proportions $\alpha = \{\alpha_k/k = 1 \ldots N_c-1\}$ be the normalized lengths of each class corresponding to the information bits with $\sum_{k=1}^{N_c-1} \alpha_k = 1$. Supposing that $\alpha$ is known, provided by a data partitioning mechanism for example, the proportions distribution of the bits in the channel codeword belonging to the different classes $\{C_k/k = 1 \ldots N_c\}$ is given by $p = \{\alpha_1R, \ldots, \alpha_{N_c-1}R, (1-R)\}$.

B. Using irregularity of LDPC codes for UEP-AWGN channels

First, we focus on infinite length LDPC codes. LDPC codes exhibit a threshold behavior depending on the channel signal to noise ratio $E_b/N_0$; above a given $E_b/N_0$ threshold $\delta$, the word error probability $P_w$ is zero as the word length $N$ tends to infinity. In that sense, there is no possibility of UEP in a LDPC codeword, since the error probability of each bit within a codeword is zero. However, if we consider a finite number of decoding iterations, we can exhibit the inherent UEP capability.
of LDPC codes. Let us consider the bit error probability at the \( l^{th} \) decoding iteration for data nodes with a connection degree \( i \) under gaussian approximation [4]:

\[
P_i^{(l)} = \tilde{\lambda}_i Q \left( \sqrt{\frac{2 + \imath m^{(l)}_i}{2}} \right)
\]

with \( \sigma^2 \) the additive gaussian noise variance, \( m^{(l)}_i \) the mean of messages from check to data nodes at the \( l^{th} \) iteration [4] and \( \tilde{\lambda}_i \) the proportion of degree \( i \) data nodes within a codeword. Above the threshold \( \delta \), \( m^{(l)}_i \) is strictly increasing with \( l \). \( Q(.) \) being a strictly decreasing function, equation (1) shows that, at a given iteration \( l \), the more connected a data node is, the more "protected" it is, in the sense that it has a smaller error probability than a data node with smaller connection degree.

So, as proposed in [5], a general strategy could be to associate information bits provided by the source coder with data nodes with the higher connection degrees, according to their relative error sensitivity. Let consider a rate \( R = 1/2 \) code optimized for an AWGN channel ([3], [4]). We assume that the UEP channel parameters (i.e. the average proportion distribution \( p \) of the \( N_c \) sensitivity classes of the channel codewords) are given. An easy way to allocate data nodes of the LDPC codewords to information bits, with respect to the degree distribution of the data nodes, is the following: we assign the information bits belonging to the first class to the information bits, with respect to the degree distribution of the data nodes, is the following: we assign the information bits belonging to the first class to the information bits provided by the source coder with data nodes with the higher connection degrees, according to their relative error sensitivity. Let consider a rate \( R = 1/2 \) code optimized for an AWGN channel ([3], [4]). We assume that the UEP channel parameters (i.e. the average proportion distribution \( p \) of the \( N_c \) sensitivity classes of the channel codewords) are given. An easy way to allocate data nodes of the LDPC codewords to information bits, with respect to the degree distribution of the data nodes, is the following: we assign the information bits belonging to the first class to the first class most connected data nodes, the information bits of the class \( C_2 \) to the \( \alpha_2 R \) most connected remaining data nodes, and so on up to the \( N_c^{-1} - 1 \) class. Finally, the redundancy bits are associated to the remaining \((1 - R)\) data nodes. This is illustrated in figure 1. The data node degree distribution for the rate \( R = 1/2 \) AWGN optimized LDPC codes and the mapping of information bits according to their sensitivity class are illustrated for a given proportion distribution \( c_2 = (0.1, 0.9) \) and \( p = (0.05, 0.45, 0.5) \). There is still a question to solve: how can we conclude that the code resulting from the bit allocation in figure 1 provides a "good" UEP. Is this the best profile to provide UEP? This question underlines the need for a comparison tool of UEP capabilities. We give thereafter a definition of what we expect to be a "good"

**UEP-AWGN optimized code.**

**Definition 1:** Let \( C_1 \) and \( C_2 \) be two LDPC codes with the same coding rate \( R \). The UEP channel is composed of \( N_c \) classes. Let \( P_1^{C_1}(l) \) and \( P_2^{C_2}(l) \) be the error probabilities of the class \( C_k \) at the \( l^{th} \) iteration for respectively \( C_1 \) and \( C_2 \). Then, if

\[
\exists k \in \{1, \ldots, N_c - 1\}, P_k^{C_1}(l) > P_k^{C_2}(l)
\]

and

\[
\forall k', \, k' \leq k, \, P_k^{C_{1'}}(l) \geq P_k^{C_{2'}}(l)
\]

\( C_1 \) is considered as a better code than \( C_2 \) for the considered UEP channel when \( l \) decoding iterations are performed.

Thus, \( C_1 \) is a better UEP-code than \( C_2 \) if there exits at least one class \( C_k \) for which the performance of \( C_1 \) are better than those of \( C_2 \). Moreover, the performance of the more sensitive classes have to be comparable for \( C_1 \) and \( C_2 \). This definition tries to take into account that the decoding of less sensitive classes are not useful if data of greater error sensitivity are erroneous. Indeed, a sequential source decoding is usually used and due to the sequential nature of the decoding process, some errors in a class can impact on the decoding of the following ones. Therefore, the first classes need to be protected as much as possible. In the following, we propose a method to optimize LDPC codes for a given UEP-AWGN channel according to definition 1 and we will show that the UEP optimized code has a better behavior than the AWGN optimized one.

![Fig. 1. Association of classes with data node degree distribution: a direct approach](image-url)
III. A HIERARCHICAL APPROACH FOR UEP-LDPC OPTIMIZATION

In this section, we present a general method to optimize LPDC codes for a given UEP channel. First, we briefly give the notations we will use in the following. Based on the previous observations, we show that the optimization problem can be achieved through a hierarchical process, each step consisting in the optimization of the average data node degree in a class subject to a maximized minimum degree and some constraints provided by previous steps. Each step can be achieved by linear programming. Finally, we give the hierarchical optimization algorithm.

A. Notations

We consider the UEP-AWGN channel defined by $N_c$ classes within a codeword. The distribution of class proportions is given by $q = \{\alpha_k/k = 1 : N_c - 1\}$. The redundancy is associated with the $N_c$-th class. Considering a rate $R$ LDPC code, the distribution of data among the $N_c$ classes is given by $\rho = \{\alpha_1, R, \ldots, \alpha_{N_c - 1}, R, (1 - R)\}$.

Let $t_{c_{\text{max}}}$ and $\rho(x) = \sum_{j=2}^{t_{c_{\text{max}}}} \rho_j x^{j-1}$ be respectively the maximum check node connection degree and the generating function of check nodes degree distribution [12]. We assume that $\rho(x)$ is the same for each class. Let $t_{c_{\text{max}}}$ be the maximum data node connection degree in the class $C_k$. For each class $C_k$, we define $\lambda^{(C_k)}(x) = \sum_{i=2}^{t_{c_{\text{max}}}} \lambda_i^{(C_k)} x^{i-1}$ and $\bar{\lambda}^{(C_k)}(x) = \sum_{i=2}^{t_{c_{\text{max}}}} \bar{\lambda}_i^{(C_k)} x^{i-1}$ the generating function of the degree distribution for the data nodes and the dual generating function where $\bar{\lambda}_i^{(C_k)}$ is the fraction of degree-$i$ data nodes. We note $\bar{\lambda}(C_k)$ the average data node degree of class $C_k$. The following equalities hold

$$\sum_{k=1}^{N_c} \lambda^{(C_k)}(1) = 1 \quad \text{and} \quad \sum_{k=1}^{N_c} \bar{\lambda}^{(C_k)}(1) = 1$$

$$\forall k = 1 \ldots N_c - 1, \sum_{i=2}^{t_{c_{\text{max}}}} \bar{\lambda}_i^{(C_k)} = \alpha_k R \quad \text{and} \quad \sum_{i=2}^{t_{c_{\text{max}}}} \bar{\lambda}_i^{(N_c)} = (1 - R)$$

The relation between $\lambda_i^{(C_k)}$ and $\bar{\lambda}_i^{(C_k)}$ is given by

$$\bar{\lambda}_i^{(C_k)} = \frac{\lambda_i^{(C_k)}}{\sum_{i'} \lambda_i^{(C_k)}}$$ (2)

In the sequel, we note $\Delta(C_k) = [\lambda_2^{(C_k)}, \ldots, \lambda_{t_{c_{\text{max}}}}^{(C_k)}]^T$ and $\bar{\lambda}$ the vectors associated with $\lambda^{(C_k)}(x)$ and $\rho(x)$. $1$ is a one valued vector and $^T$ is used for the transpose vector. We assume that $t_{c_{\text{max}}} = \max(t_{c_{\text{max}}}, \forall k = 1, \ldots, N_c$. We also set $1/t_k = [1/2, \ldots, 1/t_{c_{\text{max}}}]^T$, $1/t_c = [1/2, \ldots, 1/t_{c_{\text{max}}}]^T$ and $\bar{\Delta} = [\bar{\Delta}(C_1), \ldots, \bar{\Delta}(C_{N_c})]^T$. With these notations, a LDPC code irregularity is parameterized by $(\Delta, \bar{\lambda}, \rho, \theta)$.

B. The optimization of a class as a conditioned linear programming problem

In this section, we only focus on the optimization of a single class $C_k$, i.e. the optimization of the irregularity for the part of the codeword associated with this class. We assume that all the optimizations for classes $\{C_k', k' < k\}$ have been already performed. Let $P_{C_k}^{(l)}$ be the bit error probability for the class $C_k$ defined as

$$P_{C_k}^{(l)} = \frac{1}{\alpha_k R} \sum_{i=t_{c_{\text{min}}}^{(k)}}^{t_{c_{\text{max}}}^{(k)}} \bar{\lambda}_i^{(C_k)} Q \left( \frac{2}{\sigma^2} + \frac{im\theta^{(l)}}{2} \right) \forall k = 1 \ldots N_c - 1$$ (3)

We are interested in the classes with information bits only. $t_{c_{\text{max}}}$ and $t_{c_{\text{min}}}^{(k)}$ are respectively the maximum and the minimum data node degrees present in the class $C_k$. From equation (3), the following upper bounds hold

$$P_{C_k}^{(l)} \leq \frac{1}{R} Q \left( \sqrt{\frac{2}{\sigma^2} + \frac{t_{c_{\text{min}}}^{(k)} m\theta^{(l)}}{2}} \right) \forall k = 1 \ldots N_c - 1$$ (4)

According to equation (3) and (4), the bit error probability of a class is closely related to the minimum degree $t_{c_{\text{min}}}^{(k)}$ of the class. To optimize the irregularity of the class $C_k$, we propose to maximize the average data node degree when the minimum data node degree is as high as possible.

According to ([12], [4]), we will consider the LDPC codes that converge to a vanishing bit error probability at a given $E_b/N_0$. The target $E_b/N_0$ will be denote the code threshold. The chosen threshold for the UEP-AWGN channel optimization
is set to $E_b/N_0 = \delta + \epsilon$, where $\delta$ is the threshold for the AWGN optimized LDPC code for a given $\rho(x)$ and $\epsilon$ a small constant. We choose a value of $E_b/N_0$ different but close to the LDPC-AWGN threshold $\delta$ for two reasons: first we want to ensure almost the same global performance, and secondly, we want to increase the LDPC irregularities set in which we will choose the best UEP-LDPC code. The proposed optimization is performed by maximizing the average data node degree of the class $C_k$ for a decreasing $t_{c_{\min}}^{(k)}$ from $t_{c_{\max}}$ to 2. The iterative procedure is stopped when a solution of a LDPC code is found which converges at the target $E_b/N_0 = \delta + \epsilon$. The average data node degree cost function tries to take into account the fact that the error probability of a class is closely related to the distribution of data node degrees and that the higher the degrees are, the lower the error probability is expected.

Maximizing the average data node degree of the class $C_k$ can be written as the maximization of the following expression:

$$\overline{\lambda(C_k)} = \sum_{i=2}^{t_{c_{\max}}} \lambda_i^{(C_k)} / \bar{t}$$

(5)

Inserting equation (2) in equation (5) and considering that the code rate $R$ is constant (and so, $\sum_k \sum_{i'} \lambda_{i'}^{(C_k)} / i'$ is constant), the maximization (5) is equivalent to the maximization of

$$\sum_{i=2}^{t_{c_{\max}}} \lambda_i^{(C_k)}$$

(6)

For a given threshold $\delta + \epsilon$ (and then a given noise power $\sigma^2$) and a check node degree distribution $\rho(x)$, the optimization of the class $C_k$ can be stated as a linear programming problem subject to three types of constraints as follows:

1) Initialization: $t_{c_{\min}}^{(k)} = t_{c_{\max}}$

2) While optimization failure (Any constraint is not fulfilled):
   a) Optimize

   $$\max_{\lambda} \lambda^{(C_k)}^T 1$$

   (7)

   under the constraints
   - Global constraints:
     $[C_1]$ Rate constraint:
     $$\sum_k \lambda^{(C_k)}^T 1/t_c = (1 - R)^{-1} \rho^T 1/t_r$$

     $[C_2]$ Proportion distribution constraints:
     \begin{align*}
     (i) & \sum_k \lambda^{(C_k)}^T 1 = 1 \hfill \\
     (ii) & \forall k \in \{1, \ldots, N_c - 1\}, \lambda^{(C_k)}^T 1/t_c = \alpha_c R 1/t_r \end{align*}

     $[C_3]$ Convergence constraints:
     $$F(\lambda, x, \sigma^2) > x$$

     $[C_4]$ Stability condition:
     $$\sum_k \lambda_2^{(C_k)} < e^{1/2\sigma^2} / \sum_{j=2} \rho_j (j - 1)$$

   - Local constraint:
     $[C_5]$ Minimum data node degree constraint:
     $$\forall i < t_{c_{\min}}^{(k)}, \lambda_i^{(C_k)} = 0$$

   - Conditional constraints:
     $[C_6]$ Previous optimizations constraints:
     $$\forall k' < k, \lambda_{i'}^{(C_{k'})}$$ is fixed

   b) $t_{c_{\min}} = t_{c_{\min}}^{(k)} - 1$

end
In constraint \([C_1]\), the function \(F(\lambda, x, \sigma^2)\) is the EXIT Chart of the LDPC code and \(x\) is the mutual information associated to the messages in the graph [3]. The cost function used in equation (7) only depends on \(\lambda^{(C_k)}\), which is a local optimization cost function. The optimization results are however the vectors \(\{\lambda^{(C_{k'})}, \forall k' \geq k\}\) which are involved in the global constraints. The conditions \([C_1]\) to \([C_4]\) are global constraints related to code convergence, rate and proportion distribution constraints. \([C_5]\) is related to the local constraint of the minimum data node degree of the class \(C_k\) and finally \([C_6]\) takes into account the optimized irregularities of the previous classes.

C. Hierarchical optimization algorithm

Based on definition 1, we propose a hierarchical approach for the global optimization problem. We will then start to optimize the most sensitive classes and perform the hierarchical optimization in decreasing order of sensitivity. Assuming that \(t_{c_{\text{max}}}\) and \(\rho(x)\) are given, the following algorithm illustrates this hierarchical approach.

**Hierarchical UEP Optimization algorithm:**

1) Fix \(E_b/N_0 = \delta + \epsilon\)

2) for \(k = 1 \ldots N_c - 1:\)
   a) Find \(\lambda_{opt}^{(C_k)}\) and \(t_{c_{\text{min, opt}}}^{(k)}\) with the optimization procedure described in section III-B.
   b) Compute the constraints of the next step with \(\{\lambda_{opt}^{(C_k)}, t_{c_{\text{min, opt}}}^{(k)}\}, \forall k' \leq k\}

In the following section, we will apply this algorithm with different values of \(\epsilon\) and compare to the UEP-AWGN code.

IV. Results

In this section, we present some simulation results to illustrate the performance of the method we propose. First, we analyze the results provided by the linear programming optimization in the case of infinite codeword length. Then, we focus on the performance in case of finite codeword size to validate our approach.

A. Optimization results

For the simulations, we consider a UEP channel with 3 classes within a codeword: \(C_1\) is the high error sensitivity information bits class, \(C_2\) the low error sensitivity information bits class and \(C_3\) is assigned to redundancy bits. The information bits proportion distribution is given by \(\alpha = (\alpha_1, \alpha_2)\), which can take different values. We will consider rate \(R = 1/2\) LDPC codes. We assume that \(t_{c_{\text{max}}}\) is fixed to \(t_{c_{\text{max}}} = 30\). \(\rho(x)\) is fixed to the value of the AWGN optimized LDPC code [3]. Thus, all performance of UEP-AWGN optimized LDPC codes will be compared to the performance of the AWGN optimized code for those parameters. \(\delta\) is the threshold in dB of the AWGN optimized code and \(\epsilon\) is the \(E_b/N_0\) offset. Figure 2 gives the minimum degree for the first class versus \(\epsilon\) for different information bits distribution \(\alpha = (\alpha_1, \alpha_2)\). As we can see, the minimum degree increases with increasing \(\epsilon\) values. Therefore, the UEP-LDPC code will exhibit better UEP behavior than the AWGN code. For low \(\epsilon\) values, the minimum degree converges to the minimum degree of the AWGN optimized code. We also remark that the increase of the minimum degree is less significant when \(\alpha\) converges to a uniform repartition.
B. Finite length simulation results

For the sake of space, we give the finite length results only for the case $\alpha = (0.2, 0.8)$. $\epsilon = 0.0512 \mathrm{dB}$, $\rho(x) = 0.0437x^7 + 0.9563x^8$ and $t_{\text{max}} = 30$. Figure 3 shows simulation results for UEP-LDPC codes of length $N = 2000$ at iteration $l = 6$ when 10000 Monte-Carlo runs are performed. The curves show an improvement of performance for the first class while maintaining the same performance for the two other classes. According to definition 1, the UEP-AWGN optimized code offer a better UEP than the AWGN optimized one. In figure 3, we can also observe that the two codes have almost the same global performance. Finally table I gives the degree distribution for the UEP-AWGN optimized LDPC code and the AWGN optimized one. The information bit mapping for the AWGN optimized code is performed as illustrated in figure 1.

V. CONCLUSION

In this paper we have proposed a general method to optimize LDPC codes for UEP channels. The method is based on a hierarchical optimization of the irregularity for each class within the codeword by maximizing the average data node degree while guaranteeing a minimum degree as high as possible. This method shows encouraging results, since the UEP-AWGN optimized codes show better UEP capabilities than the AWGN optimized one. Although we developed this method for AWGN channel, it could be easily extended to other types of channels. In future works, performance results for codeword partition with more than 3 classes will be investigated and a study in a joint source and channel decoding context will be done.

REFERENCES


Fig. 3. Bit error probability performance vs $E_b/N_0$

<table>
<thead>
<tr>
<th>C1</th>
<th>C2</th>
<th>C3</th>
<th>C4</th>
<th>C5</th>
<th>C6</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\lambda_{14}$</td>
<td>0.2078</td>
<td>0.1320</td>
<td>0.2127</td>
<td>0.1003</td>
<td>0.1764</td>
</tr>
<tr>
<td>$\lambda_{15}$</td>
<td>0.0017</td>
<td>0.0750</td>
<td>0.0161</td>
<td>0.2943</td>
<td>0.0541</td>
</tr>
<tr>
<td>$\lambda_{30}$</td>
<td>0.2215</td>
<td>0.1332</td>
<td>0.0870</td>
<td>0.0585</td>
<td>0.0017</td>
</tr>
</tbody>
</table>

TABLE I
COMPARISON OF DEGREE DISTRIBUTIONS FOR THE DIFFERENT CLASSES WITH $\epsilon = 0.0512$