The 31st Symposium on Information Theory
and its Applications (SITA2008)
Kinugawa, Tochigi, Japan, Oct. 7–10, 2008

On Designing Irregular LDPC Codes using Accurate Densities for Gilbert-Elliott Channel

Manabu KOBAYASHI * Hideki YAGI † Toshiyasu MATSUSHIMA ‡
Shigeichi HIRASAWA §

Abstract— In this paper we investigate the design of low-density parity-check (LDPC) codes for the Gilbert-Elliott (GE) channel. In the design method proposed by Eckford et al., the input probability density function (PDF) of the extrinsic information transfer (EXIT) chart is approximated by the Gaussian distribution. However, the generated EXIT chart is not so accurate since the PDF of the channel messages is not Gaussian for the GE channel. Therefore, we propose two methods to get the accurate PDFs. First method can obtain the approximate PDFs by utilizing two density-evolution (DE) steps for the Gaussian distribution. Second one generates the accurate EXIT chart by executing the exact DE (EDE) with the result obtained by first method. Consequently, we can design the good LDPC codes by using these methods.

Keywords— Irregular low-density parity-check (LDPC) codes, density evolution, Gilbert-Elliott channel, Gaussian approximation, code optimization.

1 Introduction

Low-density parity-check (LDPC) codes are a class of linear codes with very sparse parity-check matrices [1]. To analyze the performance of LDPC codes, the density-evolution (DE) algorithm has been proposed by Richardson et al. in [2]. This DE analysis provides performance thresholds which the decoder converges to zero error probability. Furthermore, irregular LDPC codes exhibit the performance extremely close to the Shannon limit for memoryless channels [3]. S.-Y. Chung et al. introduced an one-dimensional (1-D) analysis of LDPC codes on additive white Gaussian noise (AWGN) [4]. Assuming a Gaussian distribution for messages in message-passing decoding, this method can reduce the complexity of the design of irregular LDPC codes. Moreover, Ardakani et al. have proposed the design method which assumes a Gaussian distribution only for messages from variable nodes to check nodes [5]. Consequently, good irregular LDPC codes can be designed since this method generates the accurate extrinsic information transfer (EXIT) chart.

For the Gilbert-Elliott (GE) channel or more general Markov channels, several message-passing decoding algorithms of LDPC codes have been proposed. These algorithms result in significantly improved performance for channels with memory. Furthermore, Eckford et al. have analyzed performance of LDPC codes over the GE channel using DE, under an estimation-decoding strategy, in which intermediate results from the iterative decoding algorithm are used to estimate a channel state [6]. Recently, Eckford et al. proposed a design method of irregular LDPC codes using approximate DE for Markov channels [7]. In the design method proposed by Eckford et al, the probability density function (PDF) of the messages from variable nodes to check nodes is approximated by the Gaussian distribution. However, the generated EXIT chart is not so accurate since the PDFs of the channel messages are not Gaussian for the GE channel.

In this paper, we propose the method to get accurate PDFs of the messages from variable nodes to check nodes compared with the Gaussian distribution. First method can obtain the approximate PDFs by utilizing two density-evolution (DE) steps for the Gaussian distribution. Second one generates the accurate EXIT chart by executing the exact DE (EDE) with the result obtained by first method. Consequently, we can design the good LDPC codes by using these methods.

2 Design of LDPC Codes for GE Channel

Definition 1 Let $\mathcal{C}$ be a binary $(\lambda(x), \rho(x))$-irregular LDPC code of length n, where $\lambda(x)$ and $\rho(x)$ are defined as $\lambda(x) = \sum_{i=2}^{v_{\max}} \lambda_i x^{i-1}$ and $\rho(x) = \sum_{i=2}^{c_{\max}} \rho_i x^{i-1}$, respectively, and λ_i and ρ_i are the probabilities that a given edge in an LDPC subgraph is connected to a variable node and check node of degree i, respectively.

A codeword $\boldsymbol{X} \in \mathcal{C}$ is transmitted to the channel and the receiver receives a channel output $\boldsymbol{Y} \in \{0,1\}^n$ such that

$$\boldsymbol{Y} = \boldsymbol{X} \oplus \boldsymbol{Z}, \quad (1)$$

where $\boldsymbol{Z} \in \{0,1\}^n$ is a noise sequence, and $\oplus$ denotes the componentwise modulo-2 addition. The noise sequence $\boldsymbol{Z}$ arises from a two-states hidden Markov process at the GE channel. Given the state space $\mathcal{S} = \{G, B\}$ and a state sequence $\boldsymbol{S} \in \mathcal{S}^n$, η_s denotes the crossover probability at the state $s \in \mathcal{S}$, i.e., $\eta_G = \Pr(Z_i = 1|S_i = G)$ and $\eta_B = \Pr(Z_i = 1|S_i = B)$. The state transition probabilities are denoted by $g = \Pr(S_{i+1} = G|S_i = B)$ and $b = \Pr(S_{i+1} = B|S_i = G)$. □

This probabilistic model results in a factor graph formed by connecting the GE Markov chain factor graph and the LDPC factor graph so that edges are created to con-

* Department of Information Science, Shonan Institute of Technology, Kanagawa, Japan
† Graduate School of Electro-Communications, The University of Electro-Communications, Tokyo, Japan
‡ School of Fundamental Science and Engineering, Waseda University, Tokyo, Japan
§ School of Creative Science and Engineering, Waseda University, Tokyo, Japan

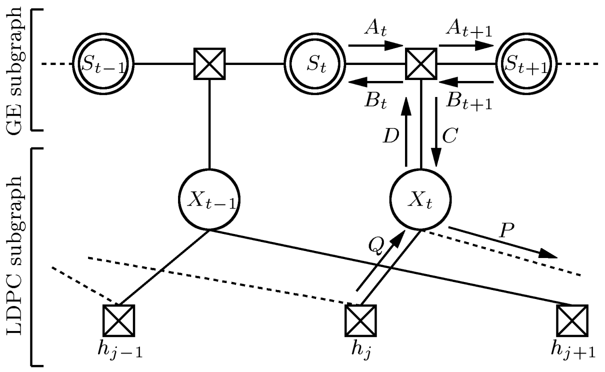

Figure 1: A GE-LDPC decoder graph and the massage flow through the Markov subgraph.

nect the appropriate symbol-variable nodes and channel factor nodes. For this factor graph, the message-passing decoder utilizing sum-product algorithm (SPA) can effectively decode the GE channel noise [6]. This decoder will be referred to as the GE-LDPC decoder [6]. The GE-LDPC decoder is depicted graphically in Fig.1.

Let P be a message from a variable node to a check node, and let Q be a message from a check node to a variable node. In Fig.1, the GE-LDPC decoder calculates the check node message Q by using the incoming messages from the neighbor variable nodes. Next, the extrinsic information D is calculated by using the check node messages from the neighbor check nodes. Furthermore, the decoder calculates the next forward message A_{t+1} by using the current forward message A_t, the channel output Y and the extrinsic information D as follows:

$$A_{t+1} = \frac{(1-b)T_G A_t + gT_B(1-A_t)}{T_G A_t + T_B(1-A_t)}, \qquad (2)$$

where $T_s = \eta_s + \gamma(D,Y)(1-2\eta_s)$ and $\gamma(D,Y) = \frac{1}{2}\left[1+\phi(Y)\tanh\left(\frac{D}{2}\right)\right]$, $\phi(Y) = (-1)^Y$, $Y \in \{0,1\}$. The forward message A_t indicates the marginal probability of the state G estimated by using the extrinsic information from the start to the current time. Similarly, the current backward message B_t is calculated by using the backward message B_{t+1} at the next time, the channel output Y and the extrinsic information D. The backward message B_t indicates the marginal probability of state G estimated by using the extrinsic information from the current time to the end. Using these forward and backward messages, the channel log-likelihood ratio (LLR) message C is calculated as follows:

$$C = \phi(Y)\log\frac{(1-\eta_G)UA_t + (1-\eta_B)(1-A_t)}{\eta_G UA_t + \eta_B(1-A_t)}, \qquad (3)$$

where $U = \frac{b+(1-2b)B_{t+1}}{1-g-(1-2g)B_{t+1}}$. Finally, the variable message P is calculated by using the channel message C and the check node messages.

Next, we briefly describe the DE algorithm proposed by Eckford et al. [6]. Let $f_P(x)$ and $f_Q(x)$ be the probability density functions (PDFs) corresponding to the random variables P and Q, respectively. Furthermore, let $f_{A_t}(x)$, $f_{B_t}(x)$, $f_C(x)$ and $f_D(x)$ be the PDFs corresponding to A_t, B_t, C and D, respectively. The DE algorithm iterates to update these PDFs from the previous PDFs along the message-passing of the decoder as follows: (i) calculate $f_P(x)$ from $f_C(x)$, $f_Q(x)$ and $\lambda(x)$, (ii) calculate $f_Q(x)$ from $f_P(x)$ and $\rho(x)$, and calculate $f_D(x)$ from $f_Q(x)$ and $\lambda(x)$, (iii) calculate $f_{A_{t+1}}(x)$ from $f_{A_t}(x)$ and $f_D(x)$, and likewise calculate $f_{B_t}(x)$ from $f_{B_{t+1}}(x)$ and $f_D(x)$, and (iv) calculate $f_C(x)$ from $f_{A_t}(x)$ and $f_{B_{t+1}}(x)$. In this algorithm, we refer to the update functions of each step as $f_P = F_P(f_C, f_Q, \lambda)$, $f_Q = F_Q(f_P, \rho)$, $f_D = F_D(f_Q, \lambda)$, $f_{A_{t+1}} = F_A(f_{A_t}, f_D)$, $f_{B_t} = F_B(f_{B_{t+1}}, f_D)$ and $f_C = F_C(f_{A_t}, f_{B_{t+1}})$, respectively. Hereafter, we refer to the algorithm which iterates above steps until the error of f_P converges as an exact DE (EDE).

Recently, Eckford et al. proposed a design method of irregular LDPC codes using approximate DE for Markov channels [7]. For fixed $f_D(x)$ we consider the sequence $f_{A_t,f_D}, t = 0,1,\ldots,$ such that $f_{A_{t+1},f_D} = F_A(f_{A_t,f_D}, f_D)$. We assume that we can obtain the PDF $f^*_{A,f_D} := \lim_{t\to\infty} f_{A_t,f_D}$. Likewise, for fixed $f_D(x)$ we define $f^*_{B,f_D} := \lim_{t\to\infty} f_{B_t,f_D}$. Then we can obtain $f^*_{C,f_D} = F^*_C(f_D) := F_C(f^*_{A,f_D}, f^*_{B,f_D})$, and f^*_{C,f_D} is based only on f_D [7]. In this way, for various f_D we calculate f^*_{C,f_D}, where f_D are approximated by the Gaussian distribution $N(m, 2m)$ with mean m and variance $2m$. We refer to the design method proposed by Eckford et al. as the conventional design method.

[Conventional Design Method][7]

1) (Precalculation) Let $\boldsymbol{\nu}_D = \{\nu_{D,1}, \nu_{D,2}, \cdots, \nu_{D,n_D}\}$ be a set of n_D quantization points for the interval $(0, 0.5]$. For all $\nu_{D,i} \in \boldsymbol{\nu}_D$, execute the following calculations.
   (a) Let $m_i := \{\mathrm{erfc}^{-1}(2\nu_{D,i})\}^2$ and $f_D := N(m_i, 2m_i)$.
   (b) Obtain $f^*_{C,\nu_{D,i}} := F^*_C(f_D)$.
2) (Initialization) Assume that $\rho(x)$ is given. Let $l := 0$. Let $\lambda^{(0)}(x)$ be an initial variable degree sequence.
3) (EXIT chart calculation) Let $\boldsymbol{\nu}_P = \{\nu_{P,1}, \nu_{P,2}, \ldots, \nu_{P,n_P}\}$ be a set of n_P quantization points for the interval $(0, p_0]$, where p_0 is the average channel error probability. For all $\nu_{P,i} \in \boldsymbol{\nu}_P$, execute the following calculations.
   (a) Let $m_i := \{\mathrm{erfc}^{-1}(2\nu_{P,i})\}^2$ and $f_P := f_{N,m_i}$.
   (b) Obtain $f_Q := F_Q(f_P, \rho)$ and $f_D := F_D(f_Q, \lambda^{(l)})$.
   (c) Choose $\nu_{D,j}$ closest to $P_e(f_D)$, i.e.,

$$\nu_{D,j} := \arg\min_{\nu_{D,j}\in\boldsymbol{\nu}_D} |P_e(f_D) - \nu_{D,j}|,$$

   where for a PDF $f(x)$, $P_e(f)$ is defined as $P_e(f) = \int_{-\infty}^{0} f(x)\mathrm{d}x$.
   (d) For each $k = 2, 3, \ldots, v_{\max}$, obtain[1] $\psi_k(\nu_{P,i}) := P_e(F_P(f^*_{C,\nu_{D,j}}, f_Q, x^{k-1}))$.

[1] x^{k-1} is equal to $\lambda(x)$ in the case where $\lambda_k = 1$.

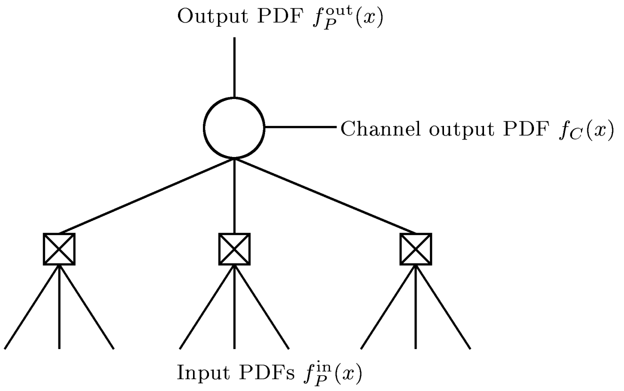

Figure 2: An example factor graph of DE for GE-LDPC decoder.

4) (Optimization) Solve the following optimization problem by linear programming[2]:
   Maximize $\sum_{k=2}^{v_{\max}} \lambda_k/k$
   Subject to $\sum_{k=2}^{v_{\max}} \lambda_k \psi_k(\nu_{P,i}) < \nu_{P,i}$ $(\forall \nu_{P,i} \in \boldsymbol{\nu}_P)$, $\lambda_k \geq 0$ $(k \in [2, v_{\max}])$, $\lambda_2 \leq \lambda_2^{\mathrm{sc}}$ and $\sum_{k=2}^{v_{\max}} \lambda_k = 1$.
   Let $\lambda^{(l+1)}(x)$ be the output of linear programming.
5) (Termination condition) If $l \geq l_{\max}$ or $\sum_{k=2}^{v_{\max}} (\lambda_k^{(l+1)} - \lambda_k^{(l)})^2 < \epsilon$ for a small ϵ, then output $\lambda^{(l+1)}(x)$ and stop, otherwise let $l := l + 1$, and go to step(3). □

In [7], furthermore, Eckford et al. have proposed the reduced complexity methods compared with above one. In this paper, we omit the description of those methods.

3 Design Method using More Accurate Densities

Figure 2 shows an example of the input-output relation of DE for GE-LDPC decoder, where for given λ and ρ, $f_P^{\mathrm{out}} = F_P(f_C, F_Q(f_P^{\mathrm{in}}, \rho), \lambda)$.

In the conventional design method, the input PDFs $f_P^{\mathrm{in}}(x)$ are approximated by Gaussian distribution. We refer to this DE as semi-Gaussian approximation (SGA) [7]. In Figures 3, 4 and 5, we show the output PDFs $f_P^{\mathrm{out}}(x)$ of the EDE and the SGA, where $P_e(f_P^{\mathrm{in}}) =$ 0.0896, 0.0796 and 0.05, respectively. In these figures, we set $(g, b, \eta_G, \eta_B) = (0.01, 0.01, 0.01, 0.22)$ as parameters of GE channel, $\lambda(x) = 0.25x + 0.424687x^2 + 0.208265x^9 + 0.117048x^{10}$ and $\rho(x) = x^6$.

Figure 5 shows that enough acculacy has been accomplished by the SGA in the case where $P_e(f_P^{\mathrm{in}}) = 0.05$. Meanwhile mismatches of output PDFs between the EDE and the SGA in Figures 3 and 4 are not negligible. In fact, if we execute the EDE with the variable degree sequence optimized by the conventional design method for the given GE channel parameters, in a lot of cases it does not succeed [7]. Then in [7], the average check degree is decreased until the EDE is successful.

[2] λ_2^{sc} is an certain constant which satisfies the stability condition [3].

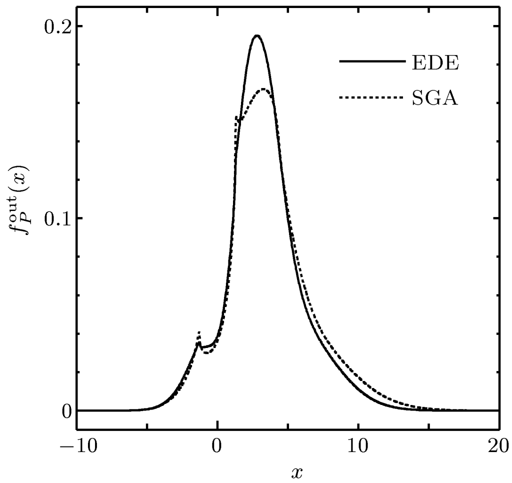

Figure 3: $f_P^{\mathrm{out}}(x)$ of the EDE and the SGA in the case where $P_e(f_P^{\mathrm{in}}) = 0.0896$.

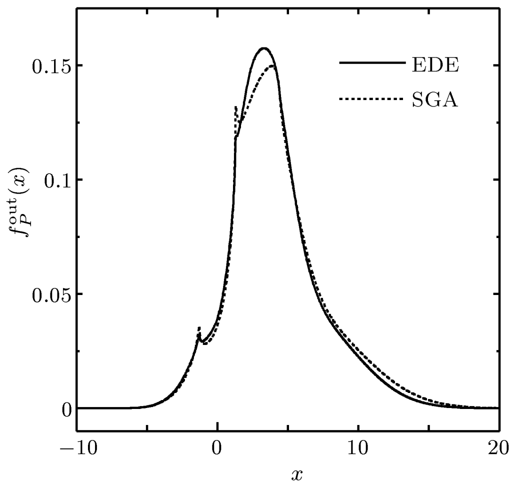

Figure 4: $f_P^{\mathrm{out}}(x)$ of the EDE and the SGA in the case where $P_e(f_P^{\mathrm{in}}) = 0.0796$.

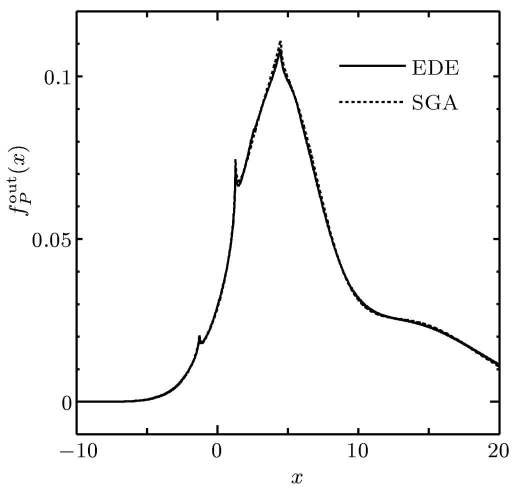

Figure 5: $f_P^{\mathrm{out}}(x)$ of the EDE and the SGA in the case where $P_e(f_P^{\mathrm{in}}) = 0.5$.

In this section, we will present two design methods using more accurate densities. First method utilizes Gaussian Approximation (GA), and second one executes the EDE for the degree sequence obtained by first method,

3.1 Iterative Density Approximation

As mentioned above, mismatches of output PDFs between the EDE and the SGA in the high error probability region are not negligible for the degree sequence optimization. However, if the output PDFs of the SGA are used as the next input PDFs $f_P^{\text{in}}(x)$, it may derive more accurate output densities. More precisely, for various m we can calculate $f_P^{\text{in}} = F_P(f_C, F_Q(N(m, 2m), \rho), \lambda)$ and $f_P^{\text{out}} = F_P(f_C, F_Q(f_P^{\text{in}}, \rho), \lambda)$, where we assume that λ and ρ are given. This is the approximation of input-output densities by executing DE twice. Hereafter, we refer to this approximation as the iterative density approximation (IDA).

In Figures 6 and 7, we show the output PDFs $f_P^{\text{out}}(x)$ of the EDE and the IDA in the case where $P_e(f_P^{\text{in}}) =$ 0.0896 and 0.0796, respectively. In these figures, the parameters of GE channel, $\lambda(x)$ and $\rho(x)$ are the same as those of Figures 3 and 4. From these figures, we see that the IDA derives very accurate densities as compared with Figures 3 and 4.

Here, we show the design method to obtain the degree sequence with high rate by using the IDA.

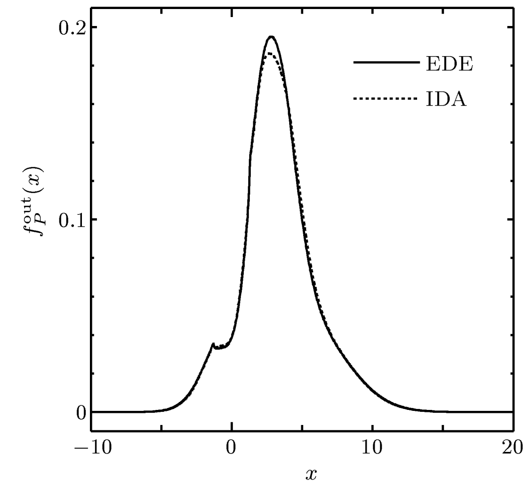

Figure 6: $f_P^{\text{out}}(x)$ of the EDE and the IDA in the case where $P_e(f_P^{\text{in}}) = 0.0896$.

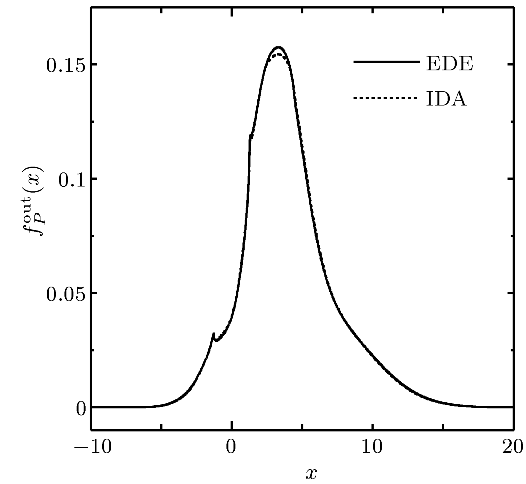

Figure 7: $f_P^{\text{out}}(x)$ of the EDE and the IDA in the case where $P_e(f_P^{\text{in}}) = 0.0796$.

[Design Method with IDA]

Steps 1),2) and 5) are the same as the conventional design method.

3) (EXIT chart calculation) Let $\boldsymbol{\nu}_P = \{\nu_{P,1}, \nu_{P,2}, \ldots, \nu_{P,n_P}\}$ be a set of n_P quantization points for the interval $(0, \hat{p}_0]$, where $\hat{p}_0$ is set to some constant larger than the average channel error probability. For all $\nu_{P,i} \in \boldsymbol{\nu}_P$, execute the following calculations.
   (a) Let $m_i := \{\text{erfc}^{-1}(2\nu_{P,i})\}^2$ and $f_P := f_{N,m_i}$.
   (b) Obtain $f_Q := F_Q(f_P, \rho)$ and $f_D := F_D(f_Q, \lambda^{(l)})$.
   (c) Choose $\nu_{D,j}$ closest to $P_e(f_D)$.
   (d) Calculate $f_P := F_P(f^*_{C,\nu_{D,j}}, f_Q, \lambda^{(l)})$ and let $\nu'_{P,i} := P_e(f_P)$.
   (e) Calculate steps (b) and (c) again.
   (f) For each $k = 2, 3, \ldots, v_{\max}$, obtain $\psi_k(\nu'_{P,i}) := P_e(F_P(f^*_{C,\nu_{D,j}}, f_Q, x^{k-1}))$.

4) (Optimization) Solve the following optimization problem by linear programming:
   Maximize $\sum_{k=2}^{v_{\max}} \lambda_k / k$
   Subject to $\sum_{k=2}^{v_{\max}} \lambda_k \psi_k(\nu'_{P,i}) < \nu'_{P,i}$ $(\forall \nu'_{P,i}, i \in [1, n_P])$, $\lambda_k \geq 0$ $(k \in [2, v_{\max}])$, $\lambda_2 \leq \lambda_2^{\text{sc}}$ and $\sum_{k=2}^{v_{\max}} \lambda_k = 1$.
   Let $\lambda^{(l+1)}(x)$ be the output of linear programming.

□

3.2 Design Method with EDE

In previous subsection, we showed the design method by using the IDA. We can obtain the suboptimal degree sequence by the use of this method. In this subsection, furthermore, we propose the design method to derive LDPC codes with the higher rate by the effective use of the EDE. We then don't use the Gaussian approximation to generate an EXIT chart at all.

Starting from $\hat{\lambda}(x)$ obtained by design method with the IDA, the EDE is executed. Then using true input-output PDFs of the EDE, we can generate the EXIT charts for different variable degrees.

The design method using the EDE is as follows.

[Design Method with EDE]

1) (Global Initialization) Assume that ρ is given. Let $l := 0$, and let $\lambda^{(0)}(x) := \hat{\lambda}(x)$ be a variable degree sequence obtained by the design method with the IDA.

2) (EDE Initialization) Let f_A and f_B be the initial PDFs corresponding to the forward and backward state

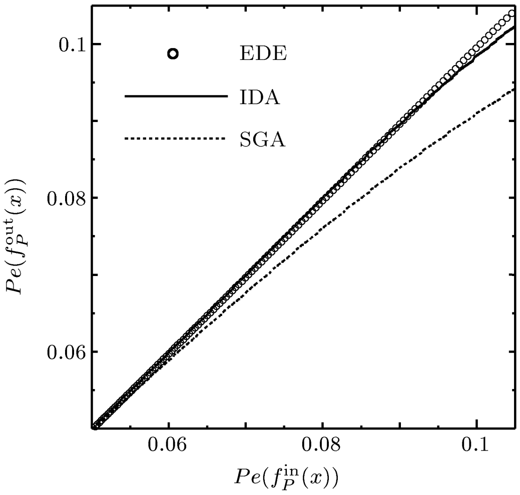

Figure 8: Actual convergence behavior of the EDE compared with the predicted trajectory using the SGA and the IDA.

probabilities, respectively. Furthermore, let f_C be an initial PDF for channel outputs and let $f_P := f_C$ and $j := 1$.

3) (EXIT chart calculation by EDE)
   (a) $\nu_{P,j} := P_e(f_P)$.
   (b) $f_Q := F_Q(f_P, \rho), f_D := F_D(f_Q, \lambda^{(l)})$.
   (c) $f_C := F_C(f_A, f_B)$.
   (d) $f_A := F_A(f_A, f_D), f_B := F_B(f_B, f_D)$.
   (e) For each $k = 2, 3, \ldots, v_{\max}$, obtain $\psi_k(\nu_{P,j}) := P_e(F_P(f_C, f_Q, x^{k-1}))$.
   (d) $f_P := F_P(f_C, f_Q, \lambda^{(l)})$.
   (e) If $j \geq j_{\max}$ or $P_e(f_P) < \epsilon'$ for a small ϵ', then go to step (4), otherwise let $j := j+1$, and go to step (a).

4) (Optimization) Solve the following optimization problem by linear programming:
   Maximize $\sum_{k=2}^{v_{\max}} \lambda_k/k$
   Subject to $\sum_{k=2}^{v_{\max}} \lambda_k \psi_k(\nu_{P,i}) < \nu_{P,i}$ $(\forall i \in [1, j])$, $\lambda_k \geq 0$ $(k \in [2, v_{\max}])$, $\lambda_2 \leq \lambda_2^{\rm sc}$ and $\sum_{k=2}^{v_{\max}} \lambda_k = 1$.

   Let $\lambda^{(l+1)}(x)$ be the output of linear programming.

5) (Termination condition) If $l \geq l_{\max}$ or $\sum_{k=2}^{v_{\max}} (\lambda_k^{(l+1)} - \lambda_k^{(l)})^2 < \epsilon$ for a small ϵ, then output $\lambda^{(l+1)}(x)$ and stop, otherwise let $l := l+1$, and go to step 2). □

In this method, steps 2) and 3) are almost the same as the EDE algorithm mentioned in Section 2. If this method is successful, a variable degree sequence obtained by this method is the local optimum, since exact values of the EDE are used to EXIT charts for different variable degrees.

4 Results

In this section, we first discuss the effectiveness of the design method with IDA. The variable degree sequence $\lambda(x) = 0.25x + 0.424687x^2 + 0.208265x^9 + 0.117048x^{10}$ is obtained by this method when we set $(g, b, \eta_G, \eta_B) = (0.01, 0.01, 0.01, 0.22)$ as parameters of GE channel and $\rho(x) = x^6$. Figure 8 shows the EXIT charts obtained by using the EDE, the SGA and the IDA in the above parameter settings, where x and y axes imply $Pe(f_P^{\rm in}(x))$ and $Pe(f_P^{\rm out}(x))$, respectively[3]. In this figure, we see that the EDE is successful since all points $(Pe(f_P^{\rm in}(x)), Pe(f_P^{\rm out}(x))$ of the EDE are below the function $y = x$. From this figure, we see that the predicted trajectory using the IDA is more accurate than that of the SGA. The SGA tends to be less accurate in the high error probability region of $f_P(x)$, meanwhile the IDA is a good approximation of the EDE.

Next, we show the variable degree sequences obtained by each design method in various parameter settings. Hereafter, we use the parameters (g, b, η_G) = $(0.01, 0.01, 0.01)$, and η_B are set to 0.15, 0.22 and 0.4 in Tables 1, 2 and 3, respectively. $\lambda_2^{\rm sc}$ are set to the same constant for each $c_{\max}$, where $c_{\max}$ is a maximum degree of check node. For example, we set $\lambda_2^{\rm sc} = 0.26$ for $c_{\max} = 7$ in Table 1.

In the conventional method, the reason of $\rho_{c_{\max}} \neq 1$ is that the EDE fails for the designed degree sequence as mentioned at the beginning of section 3.

Let $R_{\rm conv}$, $R_{\rm IDA}$ and $R_{\rm EDE}$ be the code rates corresponding to the degree sequences obtained by the methods using the conventional, the IDA and the EDE, respectively. When for each method the same value of $c_{\max}$ is used, from these tables we see that $R_{\rm conv} < R_{\rm IDA} < R_{\rm EDE}$ except for the case of $\eta_B = 0.15$ and $c_{\max} = 7$. Therefore, the proposed methods are effective in most cases.

As $c_{\max}$ gets larger, furthermore, the code rates of the degree sequences obtained by the proposed methods tend to be higher. It is natural things in the design of irregular LDPC codes. Meanwhile, conventional method seems to have highest code rate when $c_{\max}$ is relatively small [7]. We conclude that these relations are based on the accuracy of design methods.

At last, we discuss the reason that we cannot use the only design method with the EDE. The two conditions to generate the EXIT chart by using the EDE are as follows:

(i) The EDE must be successful,
(II) Enough points of the EXIT chart are obtained by the EDE.

In order to utilize this method, therefore, the variable degree sequence which satisfies these conditions is needed. Using the design method with IDA to obtain such variable degree sequence is very useful and effective.

5 Conclusion

In this paper, we proposed a design method of irregular LDPC codes by using the IDA and the EDE. The IDA which approximates the input-output densities by

[3] The EXIT chart of this type has been used in [5].

Table 1: Degree Sequences with Parameters $(g, b, \eta_G, \eta_B) = (0.01, 0.01, 0.01, 0.15)$.

Method	Degree sequence	Rate R	R/C
Conv. [7]	$\lambda_2 = 0.26$, $\lambda_3 = 0.3984$, $\lambda_4 = 0.2245$, $\lambda_5 = 0.1171$; $\rho_6 = 0.08$, $\rho_7 = 0.92$	0.5771	0.919
	$\lambda_2 = 0.25$, $\lambda_3 = 0.3683$, $\lambda_6 = 0.3718$, $\lambda_7 = 0.0099$; $\rho_7 = 0.15$, $\rho_8 = 0.85$	0.5896	0.939
IDA	$\lambda_2 = 0.26$, $\lambda_3 = 0.5345$, $\lambda_7 = 0.2055$; $\rho_7 = 1$	0.5767	0.919
	$\lambda_2 = 0.25$, $\lambda_3 = 0.4222$, $\lambda_8 = 0.2384$, $\lambda_9 = 0.0894$; $\rho_8 = 1$	0.5908	0.941
	$\lambda_2 = 0.2$, $\lambda_3 = 0.4220$, $\lambda_{10} = 0.2515$, $\lambda_{11} = 0.1265$; $\rho_9 = 1$	0.5993	0.955
	$\lambda_2 = 0.1668$, $\lambda_3 = 0.4052$, $\lambda_{12} = 0.3291$, $\lambda_{13} = 0.0989$; $\rho_{10} = 1$	0.6055	0.965
EDE	$\lambda_2 = 0.26$, $\lambda_3 = 0.5219$, $\lambda_6 = 0.2181$; $\rho_7 = 1$	0.5802	0.924
	$\lambda_2 = 0.25$, $\lambda_3 = 0.4196$, $\lambda_7 = 0.139$, $\lambda_8 = 0.1914$; $\rho_8 = 1$	0.595	0.948
	$\lambda_2 = 0.2$, $\lambda_3 = 0.4206$, $\lambda_9 = 0.2064$, $\lambda_{10} = 0.1730$; $\rho_9 = 1$	0.6038	0.962
	$\lambda_2 = 0.1668$, $\lambda_3 = 0.4058$, $\lambda_{11} = 0.3289$, $\lambda_{16} = 0.0957$, $\lambda_{17} = 0.0028$; $\rho_{10} = 1$	0.6074	0.968

Table 2: Degree Sequences with Parameters $(g, b, \eta_G, \eta_B) = (0.01, 0.01, 0.01, 0.22)$.

Method	Degree sequence	Rate R	R/C
Conv. [7]	$\lambda_2 = 0.25$, $\lambda_3 = 0.3829$, $\lambda_7 = 0.1256$, $\lambda_8 = 0.2415$; $\rho_6 = 0.13$, $\rho_7 = 0.87$	0.5147	0.945
	$\lambda_2 = 0.2$, $\lambda_3 = 0.282$, $\lambda_8 = 0.1948$, $\lambda_9 = 0.1167$, $\lambda_{17} = 0.0253$, $\lambda_{18} = 0.1812$; $\rho_8 = 0.4$, $\rho_9 = 0.6$	0.5196	0.954
	$\lambda_2 = 0.1668$, $\lambda_3 = 0.263$, $\lambda_9 = 0.2455$, $\lambda_{10} = 0.0001$, $\lambda_{14} = 0.1444$, $\lambda_{30} = 0.1114$, $\lambda_{33} = 0.0688$; $\rho_9 = 0.76$, $\rho_{10} = 0.24$	0.4944	0.908
IDA	$\lambda_2 = 0.25$, $\lambda_3 = 0.4210$, $\lambda_9 = 0.0274$,$\lambda_{10} = 0.3016$; $\rho_7 = 1$	0.5215	0.958
	$\lambda_2 = 0.2$, $\lambda_3 = 0.3029$, $\lambda_8 = 0.0273$, $\lambda_9 = 0.2294$, $\lambda_{33} = 0.0857$, $\lambda_{34} = 0.1547$; $\rho_9 = 1$	0.5312	0.976
	$\lambda_2 = 0.1668$, $\lambda_3 = 0.2946$, $\lambda_9 = 0.1031$, $\lambda_{10} = 0.1391$, $\lambda_{29} = 0.0671$, $\lambda_{49} = 0.0807$, $\lambda_{50} = 0.1486$; $\rho_{10} = 1$	0.5325	0.978
EDE	$\lambda_2 = 0.25$, $\lambda_3 = 0.4259$, $\lambda_9 = 0.0344$,$\lambda_{10} = 0.2897$; $\rho_7 = 1$	0.5234	0.961
	$\lambda_2 = 0.2, \lambda_3 = 0.3052, \lambda_8 = 0.0313, \lambda_9 = 0.2287, \lambda_{33} = 0.1175, \lambda_{34} = 0.1173; \rho_9 = 1$	0.5333	0.979
	$\lambda_2 = 0.1668$, $\lambda_3 = 0.2968$, $\lambda_9 = 0.09$, $\lambda_{10} = 0.1659$, $\lambda_{40} = 0.1786$, $\lambda_{41} = 0.0425$, $\lambda_{68} = 0.0594$; $\rho_{10} = 1$	0.5355	0.983

Table 3: Degree Sequences with Parameters $(g, b, \eta_G, \eta_B) = (0.01, 0.01, 0.01, 0.4)$.

Method	Degree sequence	Rate R	R/C
Conv. [7]	$\lambda_2 = 0.2323$, $\lambda_3 = 0.2537$, $\lambda_{10} = 0.1695$, $\lambda_{12} = 0.0935$, $\lambda_{13} = 0.0099$, $\lambda_{41} = 0.0077$, $\lambda_{46} = 0.2334$; $\rho_6 = 0.35$, $\rho_7 = 0.65$	0.3469	0.815
IDA	$\lambda_2 = 0.2323$, $\lambda_3 = 0.2794$, $\lambda_8 = 0.138$, $\lambda_9 = 0.0673$, $\lambda_{100} = 0.283$; $\rho_7 = 1$	0.3968	0.932
EDE	$\lambda_2 = 0.2323$, $\lambda_3 = 0.3069$, $\lambda_9 = 0.1453$, $\lambda_{10} = 0.0403$, $\lambda_{100} = 0.2752$; $\rho_7 = 1$	0.4082	0.959

executing the DE twice is more accurate than that of the SGA. Consequently, we showed that code rate obtained by the design method with the IDA was higher than that of the conventional method. Furthermore, we proposed the design method by using the EDE. Consequently, we can design good LDPC codes with the higher rate by the effective use of the EDE.

References

[1] R. G. Gallager, Low Density Parity Check Codes, Cambridge, MA: MIT Press, 1963.

[2] T. J. Richardson, R. L. Urbanke, "The Capacity of Low-Density Parity-Check Codes Under Message-Passing Decoding," , IEEE Transactions on Information Theory, vol. 47, no.2, pp.599 - 618, Feb. 2001.

[3] T. J. Richardson, M. A. Shokrollahi and R. L. Urbanke, "Design of Capacity-Approaching Irregular Low-Density Parity-Check Codes," IEEE Transactions on Information Theory, vol. 47, no.2, pp.619 - 637, Feb. 2001.

[4] S. Y. Chung, T. J. Richardson and R. L. Urbanke, "Analysis of Sum-Product Decoding of Low-Density Parity-Check Codes Using a Gaussian Approximation," IEEE Transactions on Information Theory, vol. 47, no.2, pp.657 - 670, Feb. 2001.

[5] M. Ardakani and F. R. Kschischang, "A More Accurate One-Dimensional Analysis and Design of Irregular LDPC Codes," IEEE Trans. on Commun., vol. 52, no. 12, pp.2106-2114, Dec. 2004.

[6] A. W. Eckford, F. R. Kschischang and S. Pasupathy, "Analysis of Low-Density Parity-Check Codes for the Gilbert-Elliott Channel, " IEEE Trans. on Information Theory, vol. 51, no.11, pp.3872-3889, Nov. 2005.

[7] A. W. Eckford, F. R. Kschischang and S. Pasupathy, "On Designing Good LDPC Codes for Markov Channels," IEEE Trans. on Information Theory, vol.53, no.1, pp5-21, Jan. 2007.