

次数分布を考慮した非正則 LDPC 符号に対する Replica Shuffled BP 復号法

Replica Shuffled BP Decoding Algorithms for Irregular LDPC Codes Based on Degree Distributions

佐藤 芳行* 細谷 剛* 八木 秀樹† 平澤 茂一*

Yoshiyuki SATO Gou HOSOYA Hideki YAGI Shigeichi HIRASAWA

Abstract— The combination of low-density parity-check (LDPC) codes and iterative decoding algorithms has shown to have high performance. A well-known iterative decoding algorithm, the belief-propagation (BP) decoding algorithm, decodes all bits in parallel at each iteration. An improved BP decoding algorithm, the shuffled BP (SBP) decoding algorithm, decodes each bits in serial at each iteration. Moreover an improved SBP decoding algorithm, the Replica SBP (RSBP) decoding algorithm, uses two sub decoders. Since the SBP decoding performs in the decreasing order of bit node degree for irregular LDPC codes, this decoding shows high decoding performance. However, as for the RSBP decoding such strategy wouldn't work due to the use of two sub decoders. Therefore in this paper, we propose an efficient strategy of the RSBP decoding based on degree distribution for irregular LDPC codes. We show simulation results which indicate that the performance of the proposed decoding is better. In addition, we also show fast convergence of the proposed decoding by density evolution.

Keywords— low-density parity-check code, irregular LDPC code, belief propagation decoding, replica shuffled BP decoding, density evolution

1 はじめに

低密度パリティ検査 (LDPC) 符号 [1], [2] は, 1962 年に R.G. Gallager によって提案された後, 長い間その存在が忘れられていたが, 最近その復号性能が見直されて盛んに研究されている. LDPC 符号に対する代表的な復号法である belief-propagation (BP) 復号法は, sum-product 復号法とも呼ばれ, LDPC 符号に対し確率的な復号を繰り返し行うことにより, 復号誤り確率を低減できる特徴を持つ. 特に, 長い符号長の LDPC 符号に対して BP 復号法を用いることにより, Shannon 限界に漸近する復号性能を達成できる [4], [5]. また, その復号計算量は符号長 N に対して $O(N)$ という高速性を併せ持つ.

* 〒 169-8555 東京都新宿区大久保 3-4-1 早稲田大学創造理工学研究科経営システム工学専攻, Major in Industrial and Management Systems Engineering, Graduate School of Creative Science and Engineering, Waseda University, 3-4-1 Okubo, Shinjuku-ku, Tokyo, 169-8555 Japan. E-mail: yosiyuki@hirasa.mgmt.waseda.ac.jp

† 〒 182-8585 東京都調布市調布ヶ丘 1-5-1 電気通信大学先端領域教育センター, Education and Research Center for Frontier Science, University of Electro-Communications, 1-5-1 Chofugaoka, Chofu, Tokyo, 182-8585 Japan.

BP 復号法は各受信ビットごとに事後確率の計算を行う. BP 復号法はタナグラフ上で確率伝播アルゴリズムとして表すことができ, 各ビットのメッセージ交換を並列に行う. 一方, Shuffled BP (SBP) 復号法 [7], [8], [11] では収束を早めるため受信ビットを一つずつ逐次的 (直列) にメッセージの交換を行う. これにより, 更新されたビットからのメッセージを直ちに利用できるため, より正確に事後確率を計算できると期待できる. したがって, SBP 復号法は少ない繰り返し数で高い復号性能を得ることができ, その結果収束を早めることができる. また, より収束を早くするために複数のサブ復号器を用いて復号を行う Replica Shuffled BP (RSBP) 復号法が提案されている [8], [11]. RSBP 復号法では各サブ復号器で異なる順序で復号を行い, 各ビットごとにエラー数の少ない方の復号器のメッセージを使って復号を行う.

BP 復号法の反復特性を解析する手法として密度発展法 (density evolution) [4], [5] が知られている. LDPC 符号には検査行列の列重みと行重みが一定である正則 LDPC 符号と重みが一定ではない非正則 LDPC 符号があり, 密度発展法を用いた解析により, 正則 LDPC 符号よりも性能が優れた非正則 LDPC 符号の次数分布が得られる [3], [5]. さらに, BP 復号法を改良した RSBP 復号法 [11] では密度発展法により収束が早くなる理論的な解析が行われている. また, 文献 [10] では理論的な解析はされていないが, 非正則 LDPC 符号に対して次数分布を考慮した SBP 復号法を提案している. しかし, RSBP 復号法に対して非正則 LDPC 符号の次数分布を考慮した復号法の開発及び, その反復性能の解析は行われていない.

本研究では, RSBP 復号法に対して非正則 LDPC 符号の次数分布を考慮した復号法を提案し, 従来 of RSBP 復号法より少ない繰り返し回数で優れた誤り訂正能力を持つことを計算機シミュレーションによって示す. さらに提案する RSBP 復号法を密度発展法によって解析を行い, 収束が早くなることを示す.

2 準備

2.1 LDPC 符号と通信路モデル

LDPC 符号 [1],[2] は, 非零要素が非常に少ない M 行 N 列の検査行列 H により定義される符号である. LDPC

符号は検査行列 H より構成されるタナーグラフによって表すことができる。タナーグラフ上において、ビットノードの中で次数 l となる辺の割合を λ_l 、チェックノードの中で次数 l となる辺の割合を ρ_l とする。本研究では符号長 N 、次数分布として $\lambda(x) = \sum_{l=1}^{d_v} \lambda_l x^{l-1}$ 、 $\rho(x) = \sum_{l=1}^{d_c} \rho_l x^{l-1}$ を持つ 2元非正則 LDPC 符号を対象とする。なお、 d_v と d_c はそれぞれビットノードとチェックノードの最大次数とする。

2元LDPC符号の符号語 $\mathbf{x} = (x_1, x_2, \dots, x_N) \in \{0, 1\}^N$ 、 $\mathbf{x}H^T = \mathbf{0}$ 、の各符号ビットは $1 \rightarrow 1, 0 \rightarrow -1$ と変調して送信される。ここで T は行列の転置を表す。本論文では加法的白色ガウス通信路を仮定し、雑音 $\mathbf{e} = (e_1, e_2, \dots, e_N)$ が加わった受信語を $\mathbf{y} = (y_1, y_2, \dots, y_N)$ と表す。受信側では受信語 \mathbf{y} から推定系列 $\hat{\mathbf{x}}$ に復号する。

2.2 BP 復号法

BP 復号法 [1],[2] は LDPC 符号の代表的な復号法であり、sum-product 復号法とも呼ばれる。BP 復号法は各ビットに対し事後確率復号を行う。

受信語 $\mathbf{y} = (y_1, y_2, \dots, y_N)$ の各時点 $n = 1, 2, \dots, N$ に対して対数尤度比を

$$U_{ch,n} = \log \frac{\Pr(y_n | x_n = 0)}{\Pr(y_n | x_n = 1)} \quad (1)$$

と表す。また、パリティ検査行列 H の第 m 行第 n 列の要素を H_{mn} とし、検査行列 H に対して次の 2 つの集合を定義する。

$$\mathcal{N}(m) = \{n : H_{mn} = 1\}, \quad (2)$$

$$\mathcal{M}(n) = \{m : H_{mn} = 1\}. \quad (3)$$

[BP 復号法]

初期条件) $H_{mn} = 1$ を満足する (m, n) に対して $z_{mn}^{(0)} := U_{ch,n}$ とする。 $i := 1$ とし、最大繰り返し数を I_{\max} とする。

step1-a) $n = 1, 2, \dots, N$ において、以下の行処理を行う。

(行処理) $m \in \mathcal{M}(n)$ において次式を計算する。

$$U_{mn}^{(i)} := 2 \tanh^{-1} \left(\prod_{n' \in \mathcal{N}(m) \setminus n} \tanh \left(\frac{V_{mn'}^{(i-1)}}{2} \right) \right). \quad (4)$$

step1-b) $n = 1, 2, \dots, N$ において、以下の列処理を行う。

(列処理) $m \in \mathcal{M}(n)$ において次式を計算する。

$$V_{mn}^{(i)} := U_{ch,n} + \sum_{m' \in \mathcal{M}(n) \setminus m} U_{m'n}^{(i)}. \quad (5)$$

step2) 以下の推定系列の計算と復号終了判定を行う。

(推定系列の計算) $n = 1, 2, \dots, N$ において次式のように繰り返し回数が i 回のときの推定系列 $\hat{x}_1^{(i)} = (\hat{x}_1^{(i)},$

$\hat{x}_2^{(i)}, \dots, \hat{x}_N^{(i)})$ を求める。

$$\hat{x}_n^{(i)} := \begin{cases} 0, & \text{if } V_n^{(i)} \leq 0; \\ 1, & \text{otherwise.} \end{cases} \quad (6)$$

ここで、 $V_n^{(i)}$ は

$$V_n^{(i)} := U_{ch,n} + \sum_{m \in \mathcal{M}(n)} U_{mn}^{(i)}, \quad (7)$$

とする。

(復号終了判定) $\hat{\mathbf{x}}^{(i)} H^T = \mathbf{0}$ または $i = I_{\max}$ のとき $\hat{\mathbf{x}}^{(i)}$ を推定系列 $\hat{\mathbf{x}}$ とし、復号を終了する。それ以外のときは $i := i + 1$ とし step1-a へ行く。 □

BP 復号法はタナーグラフ上における確率伝播アルゴリズムとして表すことができる [2]。

3 従来手法

3.1 Shuffled BP 復号法

2.2 節で述べた通常の BP 復号法では各ビットのメッセージの交換を並列に行うのに対し、SBP 復号法 [7], [8], [11] では各ビットのメッセージの交換を一つずつ逐次的に行う。

通常の BP 復号法では式 (4) で $U_{mn}^{(i)}$ を計算するとき、 $i-1$ 回目の繰り返しで計算された $V_{mn'}^{(i-1)}$ 、 $n' \in \mathcal{N}(m) \setminus n$ を用いる。一方、これらの式においてできる限り繰り返し i 回目で計算された $V_{mn'}^{(i)}$ を用いると、より正確に推定系列 $\hat{\mathbf{x}}^{(i)}$ を求められると期待できる。そこで SBP 復号法では、各ビットごとに逐次的にメッセージの交換を行い、できる限り繰り返し i 回目で計算された $V_{mn'}^{(i)}$ を用いて $U_{mn}^{(i)}$ を計算する。

[Shuffled BP 復号法]

SBP 復号法の初期条件と step2 の推定系列の計算、復号終了判定は BP 復号法と同じである。SBP 復号法では通常の BP 復号法の step1-a, step1-b を以下のように変更する。

step1') $n = 1, 2, \dots, N$ において、以下の行処理と列処理を行う。

(行処理) $m \in \mathcal{M}(n)$ において次式を計算する。

$$U_{mn}^{(i)} := 2 \tanh^{-1} \left(\prod_{\substack{n' \in \mathcal{N}(m) \setminus n \\ n' < n}} \tanh \left(\frac{V_{mn'}^{(i)}}{2} \right) \times \prod_{\substack{n' \in \mathcal{N}(m) \setminus n \\ n' > n}} \tanh \left(\frac{V_{mn'}^{(i-1)}}{2} \right) \right). \quad (8)$$

(列処理) $m \in \mathcal{M}(n)$ において次式を計算する。

$$V_{mn}^{(i)} := U_{ch,n} + \sum_{m' \in \mathcal{M}(n) \setminus m} U_{m'n}^{(i)}. \quad (9)$$

□

3.2 Replica Shuffled BP 復号法

SBP 復号法は各ビットのメッセージを $n = 1, 2, \dots, N$ の順に逐次的に更新していくと, n が大きくなるにつれてビットごとの誤り数が徐々に減少する. 一方, $n = N, N-1, \dots, 1$ の順に更新していくと, n が小さくなるにつれてビットごとの誤り数が徐々に減少する. RSBP 復号法は同時に 2 つのサブ復号器を用いる. ここで, 一つのサブ復号器では $n = 1, 2, \dots, N$ の順に逐次的にメッセージを更新し, もう一方のサブ復号器では $n = N, N-1, \dots, 1$ の順に逐次的にメッセージの更新を行う. その際, 各ビット毎にエラー数の少ない復号器のメッセージを選択し復号を行う.

ここで 2 つのサブ復号器を \vec{D} , \overleftarrow{D} とし, 行処理と列処理で得られるメッセージの値をそれぞれ $\vec{U}_{mn}^{(i)}$, $\overleftarrow{U}_{mn}^{(i)}$ と $\vec{V}_{mn}^{(i)}$, $\overleftarrow{V}_{mn}^{(i)}$ とする. ただし, \vec{D} では $n = 1, 2, \dots, N$, \overleftarrow{D} では $n = N, N-1, \dots, 1$ の順にメッセージの更新を行う. ここで $1 \leq n \leq N$ において $\mathcal{R}(n)$ と $\overline{\mathcal{R}}(n)$ を定義する.

$$\mathcal{R}(n) = \{n' \mid n \leq n' \leq N - n + 1\}, \quad (10)$$

$$\overline{\mathcal{R}}(n) = \{n' \mid 1 \leq n' \leq N, n' \notin \mathcal{R}(n)\}. \quad (11)$$

ただし, $N/2 + 1 < n$ ときは $\mathcal{R}(n) = \emptyset$ となる.

[RSBP 復号法]

RSBP 復号法の初期条件と step2 の推定系列の計算, 復号終了判定は BP 復号法と同じである. RSBP 復号法では SBP 復号法の step1' を以下のように変更する.

step1'”) $n = 1, 2, \dots, N$ における $m \in \mathcal{M}(n)$, $p = N - n + 1$ における $q \in \mathcal{M}(p)$ に対し, 以下の行処理と列処理を行う.

(行処理)

$$U_{mn}^{(i)} = 2 \tanh^{-1} \left(\prod_{\substack{n' \in \mathcal{N}(m) \setminus n \\ n' \in \mathcal{R}(n)}} \tanh \left(\frac{V_{mn'}^{(i)}}{2} \right) \times \prod_{\substack{n' \in \mathcal{N}(m) \setminus n \\ n' \in \overline{\mathcal{R}}(n)}} \tanh \left(\frac{V_{mn'}^{(i-1)}}{2} \right) \right), \quad (12)$$

$$U_{qp}^{(i)} = 2 \tanh^{-1} \left(\prod_{\substack{p' \in \mathcal{N}(q) \setminus p \\ p' \in \mathcal{R}(p)}} \tanh \left(\frac{V_{qp'}^{(i)}}{2} \right) \times \prod_{\substack{p' \in \mathcal{N}(q) \setminus N-p \\ p' \in \overline{\mathcal{R}}(N-p)}} \tanh \left(\frac{V_{qp'}^{(i-1)}}{2} \right) \right), \quad (13)$$

(列処理)

$$V_{mn}^{(i)} := U_{ch,n} + \sum_{m' \in \mathcal{M}(n) \setminus m} U_{m'n}^{(i)}. \quad (14)$$

$$V_{qp}^{(i)} := U_{ch,p} + \sum_{q' \in \mathcal{M}(p) \setminus q} U_{q'p}^{(i)}. \quad (15)$$

□

サブ復号器が 2 個での RSBP 復号法は一回の繰り返しの中で, 2 つのサブ復号器の誤りが少ないほうのメッセージを利用しながら復号を行う. また, サブ復号器の数が 2 個以上のときでも同様に考えることができる. 文献 [8] の RSBP 復号法では, non-synchronous RSBP 復号法と synchronous RSBP 復号法が提案されているが, 本研究では synchronous RSBP 復号法を対象とする. RSBP 復号法における各サブ復号器の動作を図 1 に示す.

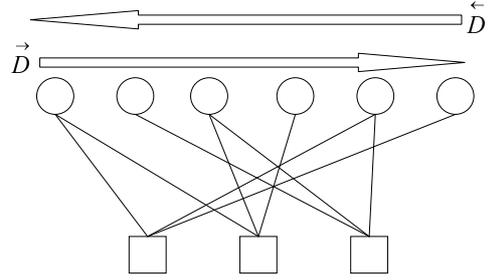


図 1: 従来の RSBP 復号法におけるサブ復号器の復号順 ($N = 6$)

4 提案手法

4.1 提案の概要

非正則 LDPC 符号に対しては次数が均一の正則 LDPC 符号よりも性能が優れた次数分布が存在することが知られている [5]. また, 非正則 LDPC 符号に対し, SBP 復号法で復号するとき, 次数の大きいビットから復号することで復号性能が向上することが知られている [10]. 従来の RSBP 復号法で 2 つのサブ復号器 \vec{D} , \overleftarrow{D} を用いたとき, それぞれのサブ復号器のメッセージ更新順序は反対向きとなるため, 1 つのサブ復号器に対して次数の大きい順にビット位置を並べると, もう一方のサブ復号器では次数の小さいものから更新してしまうという問題がある. そこで本研究では, 非正則 LDPC 符号に対する RSBP 復号法において, 次数の大きさを考慮した RSBP 復号法の更新順序の決定方法を提案する.

提案する復号法では 2 つサブ復号器を D_1 , D_2 とし, 同じ方向に復号させることを考える. まず, 次数の大きいビット位置を $1, N/2 + 1, 2, N/2 + 2, \dots, N/2 - 1, N - 1, N/2, N$ の順に並べる (簡単のため, N は偶数とする). サブ復号器 D_1 では従来と同様に $n = 1, 2, \dots, N$ の順に復号を行う. 一方, サブ復号器 D_2 ではビット位置 $N/2 + 1$ から復号を開始し, $p = N/2 + 1, N/2 + 2, \dots, N$ の順に復号し, N まで達したとき $p = 1$ に戻り $1, 2, \dots, N/2$ の順に復号する. 2 つのサブ復号器が同じ方向に復号し

ていくことで、1つの方向のみ次数の大きさを考慮してビット位置を並び換えることができる。

4.2 提案手法のアルゴリズム

[非正則 LDPC 符号に対する提案 RSBP 復号法]

s1. ビット位置 $1, N/2+1, 2, N/2+2, \dots, N/2-1, N-1, N/2, N$ の順に次数の大きいビットを並べる。

s2. サブ復号器 D_1 と D_2 はそれぞれ $n = 1, 2, \dots, N$ と $p = N/2+1, N/2+2, \dots, N, 1, 2, \dots, N/2$ の順に復号する。

s3. RSBP 復号法を行う。その際、式 (10), (11) を $\mathcal{R}'(n)$, $\overline{\mathcal{R}}'(n)$ として以下のように定義する。

$$\mathcal{R}'(n) = \{n' \mid n \leq n' \leq N/2, n + N/2 \leq n' \leq N\}, \quad (16)$$

$$\overline{\mathcal{R}}'(n) = \{n' \mid 1 \leq n' \leq N, n' \notin \mathcal{R}'(n)\}. \quad (17)$$

□

提案 RSBP 復号法における各サブ復号器の動作を図 2 に示す。

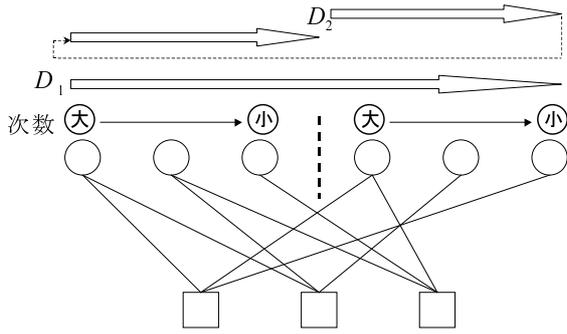


図 2: 提案 RSBP 復号法におけるサブ復号器の復号順 ($N = 6$)

5 密度発展法による分析

密度発展法 [4] は LDPC 符号アンサンブル上での BP 復号法の平均誤り率特性を計算する反復計算法である。反復計算において、BP 復号法の計算過程で交換されるメッセージの確率密度関数を BP 復号法と同様の手順で繰り返し計算する。密度発展法では、通信路と復号法の対称性より全ゼロの符号語を送り、符号長 N は $N \rightarrow$ とし、LDPC 符号のグラフはループが存在しないものと仮定する。

5.1 Shuffled BP と Replica Shuffled BP 復号法に対する密度発展法

文献 [11] より、SBP 復号法と RSBP 復号法の密度発展法による解析が行われている。SBP 復号法と RSBP 復号法では、ビットノードから出るメッセージとビットノードへ入るメッセージの確率密度関数はビット位置に

依存する。そこで、LDPC 符号の符号語のビットを G 個のグループに分け、あるチェックノードがある一つのグループ内のビットノードと接続する数は高々 1 つであると仮定する。通信路から得られるメッセージの確率密度関数を $f_{U_{ch}}$ とし、反復回数が i 回目、グループが g 番目のビットノードへ入る確率密度関数を $f_{U_g}^{(i)}$ 、ビットノードから出る確率密度関数を $f_{V_g}^{(i)}$ とする。

SBP 復号法のビットノード側の処理は通常の BP 復号法と同様に考え、 $g = 1, 2, \dots, G$ において

$$f_{V_g}^{(i)} = \sum_{l=1}^{d_v} \lambda_l \mathcal{F}^{-1} \left(\mathcal{F}(f_{U_{ch}}) \cdot \left(\mathcal{F}(f_{U_g}^{(i)}) \right)^{l-1} \right), \quad (18)$$

となる。ただし、 \mathcal{F} はフーリエ変換の処理を示す。

式 (8) から密度発展法の $f_{U_g}^{(i)}$ の計算時、 $g' < g$ ならば $f_{V_{g'}}^{(i)}$ 、 $g' > g$ ならば $f_{V_{g'}}^{(i-1)}$ を用いる。そこで、チェックノード側での処理で利用する確率密度関数は、反復 i 回目と $i-1$ 回目でそれぞれ平均をとったものを利用する。

$$f_{V_{g' < g}}^{(i)} = \frac{1}{g-1} \sum_{g'=1}^{g-1} f_{V_{g'}}^{(i)}, \quad (19)$$

$$f_{V_{g' > g}}^{(i-1)} = \frac{1}{G-g} \sum_{g'=g+1}^G f_{V_{g'}}^{(i-1)}. \quad (20)$$

チェックノードの処理は再帰的な方法によって計算することができる [6]。まず、 ψ を以下のように定義する。

$$\psi(V_1, V_2) = 2 \tanh^{-1} \left(\tanh \left(\frac{V_1}{2} \right) \tanh \left(\frac{V_2}{2} \right) \right). \quad (21)$$

通常の BP 復号法の密度発展法においてチェックノードの処理は以下の式 (22) として表す。ただし、 Ψ は式 (21) における確率密度関数の処理を表すものである。

$$f_U = \Psi(\dots \Psi(\Psi(f_V, f_V), f_V), \dots, f_V) = \Psi^{d_c-1} f_V \quad (22)$$

そして、実際にチェックノード側の処理において g 番目のグループ計算を行うとき、チェックノードへ入ってくるメッセージの組み合わせは、

$$\binom{G-1}{dc-1}, \quad (23)$$

となり、 i 回目のメッセージを j 回用いる組み合わせは

$$\binom{g-1}{j} \cdot \binom{G-g}{dc-1-j}, \quad (24)$$

となる。これらを考慮して、チェックノード側の処理は以下ようになる。

$$f_{U_g}^{(i)} = \sum_{l=1}^{d_c} \rho_l \sum_{j=0}^{l-1} \frac{\binom{g-1}{j} \cdot \binom{G-g}{l-1-j}}{\binom{G-1}{l-1}} \cdot \Psi \left(\Psi^j f_{V_{g' < g}}^{(i)}, \Psi^{l-1-j} f_{V_{g' < g}}^{(i-1)} \right). \quad (25)$$

RSBP 復号法の密度発展法は基本的に SBP 復号法の密度発展法の更新ルールを拡張させたものとなる。RSBP 復号法の密度発展法では SBP 復号法と同様に計算を行

うが、確率密度関数の計算を行うときのメッセージを以下のように修正したものを用いる．

$$f_{V_{G+1-g}}^{(i-1)} \leftarrow f_{V_g}^{(i)} \quad 1 \leq g \leq G/2, \quad (26)$$

$$f_{V_{G+1-g}}^{(i)} \leftarrow f_{V_g}^{(i)} \quad G/2 + 1 \leq g \leq G. \quad (27)$$

5.2 次数の大きいビット順による Shuffled BP 復号法に対する密度発展法

文献 [10] より、非正則 LDPC 符号において次数の大きいビットから順に復号すると復号性能は良くなることが知られているが、密度発展法による解析は行われていない．そこで、ここでは次数の大きいビット順に SBP 復号法を行ったときの密度発展法を考え、確率密度関数の更新式を示す．

従来の SBP 復号法の密度発展法と同様に G 個のグループを考え、あるチェックノードがある一つのグループ内のビットノードと接続する数は高々1つであると仮定する．また、従来と同様に $g = 1, 2, \dots, G$ の順に計算する．しかし、次数の大きい順に計算することを考えるので、次数 l が大きい順に $G \cdot \lambda_l$ 個のグループに対して以下のビットノード側の処理を行う．

$$f_{V_g}^{(i)} = \mathcal{F}^{-1} \left(\mathcal{F}(f_{U_{ch}}) \cdot \left(\mathcal{F}(f_{U_g}^{(i)}) \right)^{l-1} \right), \quad (28)$$

ここで、 $f_{V_{g' < g}}^{(i)}$ と $f_{V_{g' > g}}^{(i)}$ の計算、チェックノード側の処理となる式 (19)、式 (20)、式 (25) は従来と同様に計算する．

5.3 提案 Replica Shuffled BP 復号法に対する密度発展法

提案する RSBP 復号法では、次数の大きいビットを $1, N/2 + 1, 2, N/2 + 2, \dots, N/2 - 1, N - 1, N/2, N$ の順に並び換え復号を行う．したがって、密度発展法では $g = 1, 2, \dots, G$ の順に計算するが、 $1 \leq g \leq G/2$ で次数 l が大きい順に $G/2 \cdot \lambda_l$ 個のグループに対して式 (28) の処理を行い、 $G/2 + 1 \leq g \leq G$ ではまた次数 l が大きい順に $G/2 \cdot \lambda_l$ 個のグループに対して式 (28) の処理を行う、

また、従来の式 (26)、式 (27) は以下のように修正する．

$$f_{V_{g+G/2}}^{(i-1)} \leftarrow f_{V_g}^{(i)} \quad 1 \leq g \leq G/2, \quad (29)$$

$$f_{V_{g-G/2}}^{(i)} \leftarrow f_{V_g}^{(i)} \quad G/2 + 1 \leq g \leq G. \quad (30)$$

6 シミュレーションによる評価および数値計算結果

6.1 シミュレーション条件

提案する RSBP 復号法の有効性を示すため、計算機シミュレーションにより評価を行った．シミュレーションでは、BP 復号法 [1],[2]、SBP 復号法、従来の RSBP 復号法 [7]、提案 RSBP 復号法の性能を比較する．こ

では符号長 $N = 4000$ 、次数分布が $\lambda(x) = 0.0769x + 0.6923x^2 + 0.2308x^6$ 、 $\rho(x) = 0.46135x^5 + 0.53865x^6$ の非正則 LDPC 符号を用いる．通信路には加法的白色ガウス雑音 (AWGN) 通信路を仮定し、各復号法を 10^6 回実行するか、もしくは復号が 100 回失敗するまでシミュレーションを行う．

6.2 シミュレーション結果

まず、復号性能を比較するため、図 3 において $N = 4000$ 、最大繰り返し回数 $I_{\max} = 5$ としたときの通常の BP 復号法と SBP 復号法、最大繰り返し回数 $I_{\max} = 3, 5$ としたときの従来の RSBP 復号法と提案する RSBP 復号法の BER の結果を示す．図 3 は横軸に SN 比 [dB]、縦軸に復号誤り確率の BER をとる．

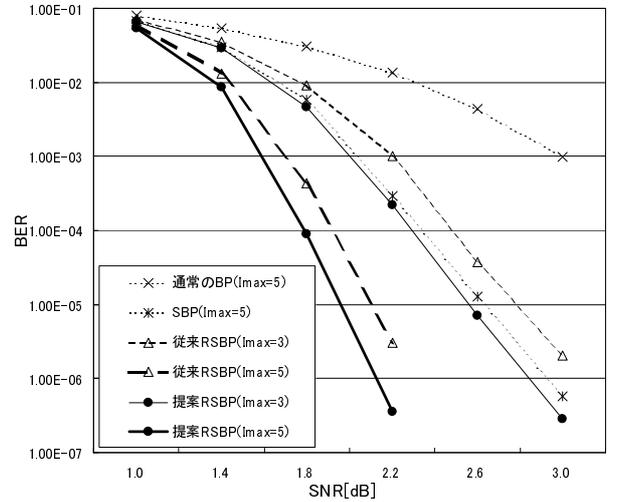


図 3: $N = 4000$ 、 $I_{\max} = 3, 5$ のときの BER

図 3 より、提案する RSBP 復号法が従来の RSBP 復号法より $I_{\max} = 3, 5$ において高い復号性能を持つことがわかる．このように次数が一定ではない非正則 LDPC 符号に対して、次数を考慮したビット位置に並び換え、サブ復号器の復号する順番も変えることで復号性能を向上させることができた．さらに、提案する RSBP 復号法では $I_{\max} = 3$ のとき、 $I_{\max} = 5$ の SBP 復号法よりも高い復号性能を示し、少ない繰り返し数で繰り返し回数の多い SBP 復号法より収束が早くなることがわかる．

6.3 密度発展法による数値計算結果

以下では、SBP 復号法、次数の大きい順に復号した SBP 復号法、RSBP 復号法、提案 RSBP 復号法の復号のビット誤り率 (BER) を図 4、図 5 に示す．図 4、図 5 は縦軸が BER で、横軸が反復繰り返し数である．ここでは、信号対雑音比 (SN 比)[dB] が SN 比=0.45172[dB] とし、グループの数は $G = 1000$ とし、次数分布は $\lambda(x) = 0.30013x + 0.28395x^2 + 0.41592x^7$ 、 $\rho(x) = 0.22919x^5 + 0.77081x^6$ となる非正則 LDPC 符号を用いる．図 4 よ

り、従来のSBP復号法よりも次数の大きい順に復号したSBP復号法の方が少ない反復で収束していることが分かる。また同様に図5より、提案するRSBP復号法も従来のRSBP復号法より早く収束していることが確認できる。

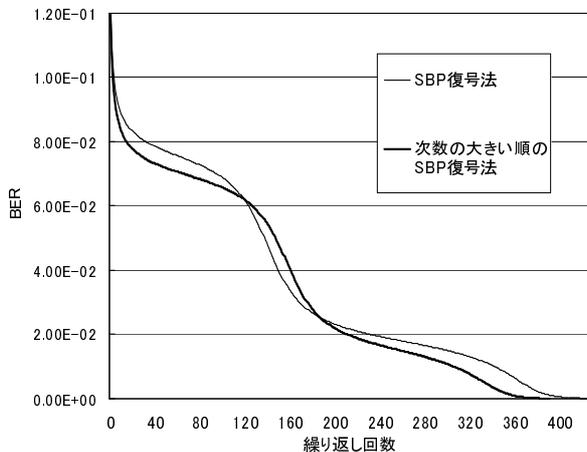


図4: SNR=0.45172[dB]でのSBP復号法, 次数の大きい順に復号したSBP復号法の各反復繰り返し数におけるBER

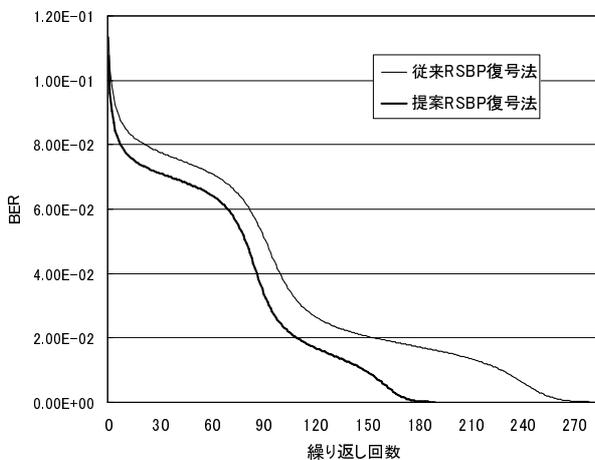


図5: SNR=0.45172[dB]でのRSBP復号法, 提案RSBP復号法の各反復繰り返し数におけるBER

7 まとめと今後

本研究では非正則LDPC符号に対して効果的にRSBP復号法を実行できるビット並び順を提案し、さらに密度発展法を用いて分析を行った。シミュレーションによる評価から従来のRSBP復号法に対して、提案RSBP復号法が収束を早めることを示した。この提案RSBP復号法では、少ない繰り返し回数 $I_{\max} = 3, 5$ のときに高い効果が得られる傾向がある。また、非正則LDPC符号の次数を考慮したSBP復号法とRSBP復号法の密度発

展法について確率密度関数の更新式を導出した。その結果、少ない繰り返し回数で誤り確率を小さくできることを示した。

今後の課題として、本研究ではRSBP復号法においてサブ復号器は2つとしたが、サブ復号器の数を増やしたときの性能の違いなどを調べるのが考えられる。さらに[12]では、密度発展法を用いたSBP復号法の解析を行っており、[11]と異なるSBP復号法の密度発展法が導かれている。本論文で導いた密度発展法とそれらの手法との比較などが今後の課題として挙げられる。

謝辞 著者の一人、佐藤は提案手法の実験結果に多くの示唆を与えて戴いた2007年度卒今田高史氏(現日本電気株式会社)に感謝いたします。

参考文献

- [1] R. G. Gallager, *Low density parity check codes*, MIT Press, 1963.
- [2] D. J. C. MacKay, "Good error-correcting codes based on very sparse matrices," *IEEE Trans. Inform. Theory*, vol. 45, no. 2, pp. 399–431, Mar. 1999.
- [3] M. G. Luby, M. Mitzenmacher, M. A. Shokrollahi and D. A. Spielman, "Improved Low-Density Parity-Check Codes Using Irregular Graphs," *IEEE Trans. Inform. Theory*, vol. 47, no. 2, pp. 585–598, Feb. 2001.
- [4] T. Richardson and R. Urbanke, "The capacity of low-density parity-check codes under message-passing decoding," *IEEE Trans. Inform. Theory*, vol. 47, no. 2, pp. 599–618, Feb. 2001.
- [5] T. Richardson and R. Urbanke, "Design of capacity-approaching irregular low-density parity-check codes," *IEEE Trans. Inform. Theory*, vol. 47, no. 2, pp. 619–637, Feb. 2001.
- [6] S. Chung, G. D. Forney, T. Richardson and R. Urbanke, "On the design of low-density parity-check codes within 0.0045 dB of the Shannon limit," *IEEE Commun. Lett.*, vol. 5, no. 2, pp. 58–60, Feb. 2001.
- [7] J. Zhang and M. P. C. Fossorier, "Shuffled iterative decoding," *IEEE Trans. Commun.*, vol. 53, No. 2, pp. 209–213, Feb. 2005.
- [8] J. Zhang, Y. Wang, and M. P. C. Fossorier, "Replica shuffled iterative decoding" *Proc. of 2005 IEEE Int. Symp. on Information Theory (ISIT2005)*, pp. 454–458, Adelaide, Australia, Sept. 2005.
- [9] Z. Chuan-Gang, Y. Jin-Sheng, L. Xue-Hong, L. Jia-Ru, "Improvement of shuffled iterative decoding" *Proc. of 2006 IEEE Information Theory Workshop*, pp. 114–116, Chengdu, China, Oct. 2006.
- [10] 内川 浩典, 原田 康祐, 佐藤 一美, "更新順序を決定する Shuffled BP 復号," 第29回情報理論とその応用シンポジウム予稿集, SITA2006, Hakodate, Hokkaido, vol. 29, No. 2, pp. 823–826, Nov. 2006.
- [11] J. Zhang, Y. Wang, and M. P. C. Fossorier, J. S. Yedidia, "Iterative decoding with replica" *IEEE Trans. Inform. Theory*, vol. 53, no. 5, pp. 1644–1663, May. 2007.
- [12] E. Sharon, S. Litsyn, and J. Goldberger, "Efficient serial message-passing schedules for LDPC decoding," *IEEE Trans. Inform. Theory*, vol. 53, no. 11, pp. 4076–4091, Nov. 2007.