

Shuffled BP 復号法に対する探索的なシンボルノードのグループ分割法

A Heuristic Search Method of Symbol Node Grouping for Shuffled BP Decoding Algorithm

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 the low-density parity-check (LDPC) codes and the iterative decoding algorithm has shown high performance. A well-known iterative decoding algorithm, the belief-propagation (BP) decoding algorithm, decodes all symbols in parallel at each iteration. An improved BP decoding algorithm, the shuffled BP decoding algorithm, decodes each symbol in serial at each iteration. To reduce the decoding delay of the shuffled BP decoding algorithm, the group shuffled BP decoding algorithm divides all symbols into some number of groups. It processes each group in serial, and all symbols in a same group in parallel. While it generates the groups in symbol positions order, in this paper we propose a new symbol grouping algorithm which generates group with respect to the structure of a graph in Tanner graph of the codes. We show simulation results which indicate that the decoding performance of the proposed method is better than those of the shuffled BP decoding algorithm and the group shuffled BP decoding algorithm.

Keywords— low-density parity-check code, belief propagation decoding, shuffled BP decoding, Tanner graph, symbol grouping method

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 [3],[4]. 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.

BP 復号法は各受信シンボルごとに事後確率の計算を行う. 通常の BP 復号法はタナー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, 各シンボルのメッセ

ジ交換を並列に行う. 一方, 収束を速くする Shuffled BP 復号法 [5],[6],[7] では受信シンボル一つずつ逐次的 (直列) にメッセージの交換を行う. これにより, 更新されたシンボルからのメッセージをすぐに利用できるため, 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, Shuffled BP 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るので, 収束を早くすることができる. しかし, 受信シンボルを逐次的に復号すると, 復号遅延が発生してしまう. 復号遅延を減らすため, 受信シンボルを順番にグループ分けし, グループ内では並列に, グループ間では逐次的に復号を行う Group Shuffled BP 復号法 [5],[6],[7] が提案されている. Group Shuffled BP 復号法を用いることで, Shuffled BP 復号法に近い復号性能で, さらに復号遅延を減らすことができる. Shuffled BP 復号法に関しては [6],[8],[9],[7] などで, さらに収束を早めるための研究がなされている. 本研究では, Group Shuffled BP 復号法に注目し, 復号遅延を減らした上で収束を早くするグループ分割法を考える.

Group Shuffled BP 復号法 [5],[6],[7] ではシンボルの位置順にグループ分けを行うのに対し, 本研究では, タナーグラフの構造を考慮に入れ, より効果的にメッセージの交換を行えるシンボルのグループ分割法を提案する. 提案したグループ分割法は従来のグループ分割法と比較して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.

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. 本研究では符号長 N , 情報記号数 K の 2 元 LDPC 符号を対象とする. また, 検査行列 H の各行の 1 の数を行重み w_r , 各列の 1 の数を列重み w_c とする.

LDPC 符号の符号語 $x = (x_1, x_2, \dots, x_N) \in \{0, 1\}^N$, $xH^T = \mathbf{0}$ の各符号シンボルは $1 \rightarrow 1, 0 \rightarrow -1$ 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 $e = (e_1, e_2, \dots, e_N)$ が加わった受信語を $y = (y_1, y_2, \dots, y_N)$ と表す. 受信側では受信語 y から推定系列 \hat{x} に復号する.

* 〒 169-8555 東京都新宿区大久保 3-4-1 早稲田大学理工学部経営システム工学科, School of Science and Engineering, Waseda University, Okubo 3-4-1, Shinjuku-ku, Tokyo, 169-8555 Japan. E-mail: yosiyuki@hirasa.mgmt.waseda.ac.jp

† 〒 169-8050 東京都新宿区西早稲田 1-6-1 早稲田大学メディアネットワークセンター, Media Network Center, Waseda University, Totsuka-Machi 1-104, Shinjuku-ku, Tokyo, 169-8050 Japan.

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 復号法は各シンボル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

$$\lambda_n = \log \frac{\Pr(y_n | x_n = 0)}{\Pr(y_n | x_n = 1)} \quad (1)$$

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.

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

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

[BP 復号法]

初期条件) $H_{mn} = 1$ を満足する (m, n) に対して $z_{mn}^{(0)} := \lambda_n$ 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.

$$\tau_{mn}^{(i)} := \prod_{n' \in \mathcal{N}(m) \setminus n} \tanh(z_{mn'}^{(i-1)}/2), \quad (4)$$

$$\varepsilon_{mn}^{(i)} := \log \frac{1 + \tau_{mn}^{(i)}}{1 - \tau_{mn}^{(i)}}. \quad (5)$$

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.

$$z_{mn}^{(i)} := \lambda_n + \sum_{m' \in \mathcal{M}(n) \setminus m} \varepsilon_{m'n}^{(i)}. \quad (6)$$

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 $\hat{\mathbf{x}}^{(i)} = (\hat{x}_1^{(i)}, \hat{x}_2^{(i)}, \dots, \hat{x}_N^{(i)})$ を求める.

$$\hat{x}_n^{(i)} := \begin{cases} 0, & \text{if } z_n^{(i)} \leq 0; \\ 1, & \text{otherwise.} \end{cases} \quad (7)$$

ここで, $z_n^{(i)}$ は

$$z_n^{(i)} := \lambda_n + \sum_{m \in \mathcal{M}(n)} \varepsilon_{mn}^{(i)}, \quad (8)$$

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 へ行く. \square

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]. タナーグラフは検査行列 H の列に対応する N 個のノードをシンボルノード s_1, s_2, \dots, s_N , 行に対応する M 個のノードをチェ

クノード c_1, c_2, \dots, c_M とし, $H_{mn} = 1$ を満足するシンボルノード s_n とチェックノード c_m を枝で結んだ2部グラフである. BP 復号法は全てのシンボルに対し, 並列にメッセージ交換を行う.

3 従来手法

3.1 Shuffled BP 復号法

2.2 節で述べた通常の BP 復号法では各シンボルのメッセージの交換を並列に行うのに対し, Shuffled BP 復号法 [5],[6],[7] では各シンボル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), (5) で $\varepsilon_{mn}^{(i)}$ を計算するとき, $i - 1$ 回目の繰り返しで計算された $z_{mn'}^{(i-1)}$ 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 $z_{mn'}^{(i)}$ 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 Shuffled BP 復号法では, 各シンボルのメッセージの交換を逐次的に行うことで, できる限り繰り返し i 回目で計算された $z_{mn'}^{(i)}$ を用いて $\varepsilon_{mn}^{(i)}$ を計算する.

[Shuffled BP 復号法]

Shuffled BP 復号法の初期条件と step2 の推定系列の計算, 復号終了判定条件は BP 復号法と同じである. Shuffled BP 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.

$$\begin{aligned} \tau_{mn}^{(i)} := & \prod_{n' \in \mathcal{N}(m) \setminus n, n' < n} \tanh(z_{mn'}^{(i)}/2) \\ & \times \prod_{n' \in \mathcal{N}(m) \setminus n, n' > n} \tanh(z_{mn'}^{(i-1)}/2). \end{aligned} \quad (9)$$

$$\varepsilon_{mn}^{(i)} := \log \frac{1 + \tau_{mn}^{(i)}}{1 - \tau_{mn}^{(i)}}. \quad (10)$$

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

$$z_{mn}^{(i)} := \lambda_n + \sum_{m' \in \mathcal{M}(n) \setminus m} \varepsilon_{m'n}^{(i)}. \quad (11)$$

\square

3.2 Group Shuffled BP 復号法

Shuffled BP 復号法は各シンボルのメッセージ交換を逐次的に行うため, 復号に大幅な遅延が生じる. 一方, 長い符号長でハードウェアの実装を行う際, VLSI において各シンボルを並列処理することが難しい. これらの問題を回避するため, Group Shuffled BP 復号法 [5],[6],[7] では符号のシンボルをいくつかのグループに分け, それぞれのグループ内におけるシンボルのメッセージ交換は並列に行い, グループ間のメッセージ交換では逐次的に

行い、復号の遅延の低減を図る。また、符号長が長い符号の場合、グループに分けることにより、ハードウェアの実装を行う際に並列化が容易となる。

ここでは N 個のシンボルを受信シンボルの位置順に G 個のグループに分ける。したがって、各々のグループのシンボル数を N_G と表すとき、 $N_G = \frac{N}{G}$ が成り立つ。ただし、 $G|N$ とする。

[Group Shuffled BP 復号法]

Group Shuffled BP 復号法でも BP 復号法の初期条件と step2 の推定系列の計算、復号終了判定条件は同じである。Group Shuffled BP 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') $g = 1, 2, \dots, G$ において、以下の処理を行う。(行処理) $n = (g-1)N_G + 1, (g-1)N_G + 2, \dots, gN_G$ における $m \in \mathcal{M}(n)$ に対して次式を計算する。

$$\begin{aligned} \tau_{mn}^{(i)} := & \prod_{n' \in \mathcal{N}(m) \setminus n, n' < (g-1)N_G + 1} \tanh(z_{mn'}^{(i)}/2) \\ & \times \prod_{n' \in \mathcal{N}(m) \setminus n, n' \geq (g-1)N_G + 1} \tanh(z_{mn'}^{(i-1)}/2). \end{aligned} \quad (12)$$

$$\varepsilon_{mn}^{(i)} := \log \frac{1 + \tau_{mn}^{(i)}}{1 - \tau_{mn}^{(i)}}. \quad (13)$$

(列処理) $n = (g-1)N_G + 1, (g-1)N_G + 2, \dots, gN_G$ における $m \in \mathcal{M}(n)$ に対して次式を計算する。

$$z_{mn}^{(i)} := \lambda_n + \sum_{m' \in \mathcal{M}(n) \setminus m} \varepsilon_{m'n}^{(i)}. \quad (14)$$

□

Group Shuffled BP 復号法は $G = 1$ のとき BP 復号法と等価であり、 $G = N$ のとき Shuffled BP 復号法と等価となる。また、十分に復号法を繰り返すことで G の値によらず高い復号性能を示すことが知られている [5],[6]。

Shuffled BP 復号法は各シンボルのメッセージ交換を逐次的に行う。Group Shuffled BP 復号法は同一グループ内のシンボルのメッセージ交換を並列に、グループ間のメッセージ交換を逐次的に交換を行う。

4 提案手法

4.1 提案の概要

Shuffled BP 復号法は各シンボルのメッセージ交換を逐次的に行うため復号に遅延が生じる。その遅延を低減するために Group Shuffled BP 復号法が提案されたが、Group Shuffled BP 復号法では受信シンボルを位置順にグループ分けするため、Shuffled BP 復号法よりも性能が劣化する。そこで本論文では Group Shuffled BP 復号法において、シンボルをグループ分けするときに、タ

ナグラフの構造を利用し、できる限り性能を劣化させないグループ分割法を提案する。

3.1 節で述べたように、繰り返し $i-1$ 回目の $z_{mn'}^{(i-1)}$ を用いて $\varepsilon_{mn}^{(i)}$ を計算するよりも、できる限り繰り返し i 回目の $z_{mn'}^{(i)}$ を用いたほうが正確に推定系列 $\hat{x}^{(i)}$ を求められる傾向がある [5],[6]。そこで提案手法では、繰り返し i 回目にできるだけ $z_{mn'}^{(i)}$ を用いて $\varepsilon_{mn}^{(i)}$ を計算できるようにシンボルをグループ分けする。提案手法によりグループ分けを行った後は、従来の Group Shuffled BP 復号法と同様に復号する。

提案手法によるグループ分割法の概要をタナグラフを用いて説明する。シンボルをグループ分けするとき、図 1 (a) のように一つのチェックノード c_m に対してシンボル $s_n, s_{n'}$ が接続している場合には、 $s_n, s_{n'}$ を同じグループに含めない。一方、図 1 (b) のように各シンボルノードが異なるチェックノード $c_m, c_{m'}, c_{m''}$ に接続している場合、 $s_n, s_{n'}$ を同じグループに含める。Group Shuffled BP 復号法では、同一グループ内で並列にメッセージを交換するため、図 1 (a) のような場合、式 (12) の $s_{n'}$ に対する行処理において繰り返し i 回目の $z_{mn}^{(i)}$ を用いることができないためである。

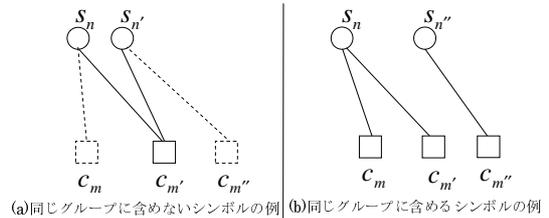


図 1: 提案グループ分割法の方針

4.2 提案するグループ分割アルゴリズム

ここで、先に説明した方針を実現するアルゴリズムを提案する。

今、シンボルの数を N 、グループの数を G とする。ただし、 $G|N$ とする。シンボル位置の集合を $\Phi = \{1, 2, \dots, N\}$ とし、 $\mathcal{A}_g (g = 1, 2, \dots, G)$ をグループ g に振り分けたシンボル位置集合、どのグループにも振り分けていないシンボル位置集合を $\bar{\Delta} = \Phi \setminus \bigcup_{g=1}^G \mathcal{A}_g$ とする。

[提案アルゴリズム]

1. $g = 1, 2, \dots, G$ において $\mathcal{A}_g := \emptyset$ とする。
2. $g = 1, 2, \dots, G$ における \mathcal{A}_g に対して、ランダムに $n \in \bar{\Delta}$ となる n を 1 つ振り分ける。 $g := 1$ とする
3. 次式を満足する k を 1 つ \mathcal{A}_g に振り分ける。

$$k = \arg \min_{n \in \bar{\Delta}} \left| \bigcup_{j \in \mathcal{A}_g} \mathcal{M}(j) \cap \mathcal{M}(n) \right|. \quad (15)$$

もし、式 (15) を満足する k が複数ある場合、その中から一つをランダムに選択し、 \mathcal{A}_g に振り分ける。

4. ここで, $g < G$ ならば $g := g+1$ として 3.へ. $g = G$ ならば 5.へ.

5. $|A_g| < \frac{N}{G}$ ならば, $g := 1$ として 3.へ. $|A_g| = \frac{N}{G}$ ならばアルゴリズムを終了する. □

5 シミュレーションによる評価と考察

5.1 シミュレーション条件

提案するグループ分割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], Shuffled BP 復号法, 従来の Group Shuffled BP 復号法 ($G = 2, 4, 10$) [5], 提案手法でグループ分割した Group Shuffled BP 復号法 ($G = 2, 4, 10$) の性能を比較する. ここでは MacKay による構成法 [2] で得られた符号長 $N = 4000$, 行重み $w_r = 6$, 列重み $w_c = 3$ の正則 LDPC 符号と符号長 $N = 500$, 行重み $w_r = 6$, 列重み $w_c = 3$ の正則 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.

シミュレーションでは, 従来手法と提案手法の復号性能について比較し, 提案したグループ分割法において (a) G の変化による復号性能の変化を確かめる. また, (b) それぞれの復号アルゴリズムの平均繰返し回数を比較することで, 復号の総演算量について示す. 最後に (c) グループの分割について考察する.

5.2 シミュレーション結果

まず, 復号性能を比較するため, 図 2~3 において $N = 4000$, 最大繰返し回数 I_{\max} を 10, 60 としたときの従来手法と提案手法の BER の結果を示す. 図 4 では $N = 500$, $I_{\max} = 5$ としたときの各手法の BER の結果を示す. また, 図 5 では提案したグループ分割法において G の変化させたときの BER の結果を示す.

次に計算量を比較するため, 図 6 において $N = 4000$, $I_{\max} = 10$ としたときの各手法の平均繰返し回数を示す. また, 図 7 では $N = 4000$, $I_{\max} = 10$ とし, 提案したグループ分割法を用いてグループ分割を行った後, 各シンボルを逐次的に復号した結果を示す.

図 2~5, 7 では横軸に信号対雑音比 (SN 比) [dB], 縦軸に復号のビット誤り率 (BER) をとる. 図 6 では横軸に SN 比, 縦軸に平均繰返し回数をとる. それぞれの図において, “提案” は $G = 2, 4, 10$ において提案手法でグループ分割した Group Shuffled BP 復号法を, “従来” は $G = 2, 4, 10$ においてシンボル位置順にグループ分割した Group Shuffled BP 復号法を表す.

5.3 考察

(a) 復号結果について

図 2 から G の値が増えるにしたがい, BER が低減することが分かる. 従来手法と提案手法において G の値が

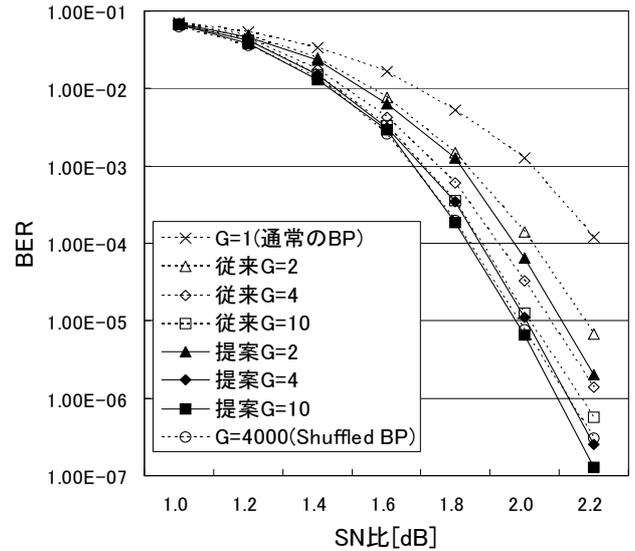


図 2: $N = 4000$, $I_{\max} = 10$ のときの BER

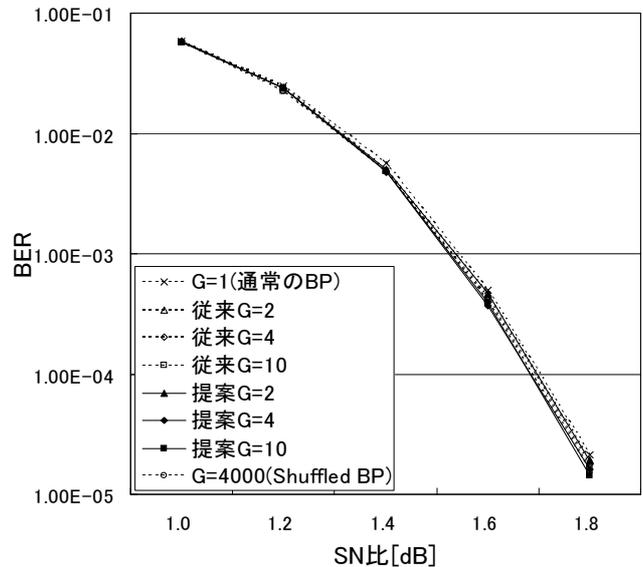


図 3: $N = 4000$, $I_{\max} = 60$ のときの BER

同一の場合を比べると提案手法の方が高い復号性能を示した. これは提案のグループ分割法が従来手法に比べ, より多くのシンボルで繰返し i 回目に更新されたメッセージ $z_{mn}^{(i)}$ を用いて式 (12) の計算ができたためと考えられる. また, 従来の Shuffled BP 復号法 ($G = 4000$) と比べても, 提案したグループ分割法の $G = 4, 10$ の場合の方が良い BER を達成した. G の値が小さい方が復号遅延は少ないため, 少ないグループ数にもかかわらず高い復号性能が得られたことは, 復号性能を向上させつつ従来の Group Shuffled BP 復号法より復号遅延を減らすことができたといえる.

図 3 より最大繰返し回数 I_{\max} を変化させたときの BER について比較する. $I_{\max} = 60$ のときでも $I_{\max} =$

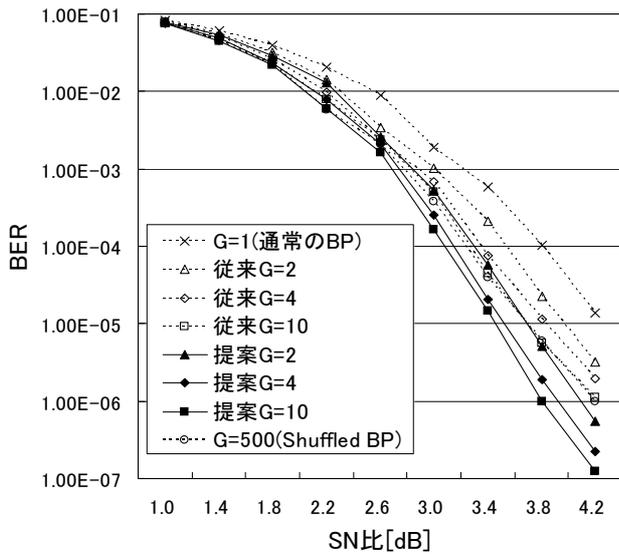


図 4: $N = 500$, $I_{\max} = 5$ のときの BER

10 のときと同様に提案手法の方が良い BER を達成しているが、 I_{\max} の値が十分に大きいため従来手法と提案手法の BER の差がほぼなくなっている。 I_{\max} を大きい値とし、十分に復号法を繰り返すことで G の値によらず同等の性能を示すことが知られているためである [5],[6]。次に符号長 N が短い場合を考える。図 4 より $N = 500$ で $I_{\max} = 5$ のときは提案したグループ分割法の $G = 2, 4, 10$ で従来の Shuffled BP 復号法 ($G = 500$) より高い復号性能が得られた。したがって、 N が短いときに高い復号性能を示すことがわかる。

ここまで、 G の値を 2, 4, 10 と変化させたときの復号性能を示した。次に図 5 より提案したグループ分割法における有効なグループの数について考察する。 $G = 20, 50$ の場合の BER を比べると、ほぼ同じであることが分かる。また、提案したグループ分割法における $G = 10, 20$ の場合、復号性能の差は少ない。そのため、復号の遅延を考えると、 $G = 10$ で十分な復号性能が得られると考えられる。

(b) 復号の演算量について

全ての復号法において繰り返し 1 回あたりの演算量は同じであるため、復号法の平均繰り返し回数が小さいほど、復号にかかる総演算量が小さくなる。図 6 では平均の繰り返し回数について示し、それぞれの復号法の総演算量を比較する。図 6 から、平均繰り返し回数は G の値が大きくなるにつれて小さくなる。また、 G の値が同一の場合、従来手法より提案手法の方が、平均繰り返し回数は少ないことが分かる。特に提案手法で $G = 10$ の場合、最も平均繰り返し回数が少ない Shuffled BP 復号法とほぼ同等である。したがって、提案したグループ分割法を用いると、従来のグループ分割法より平

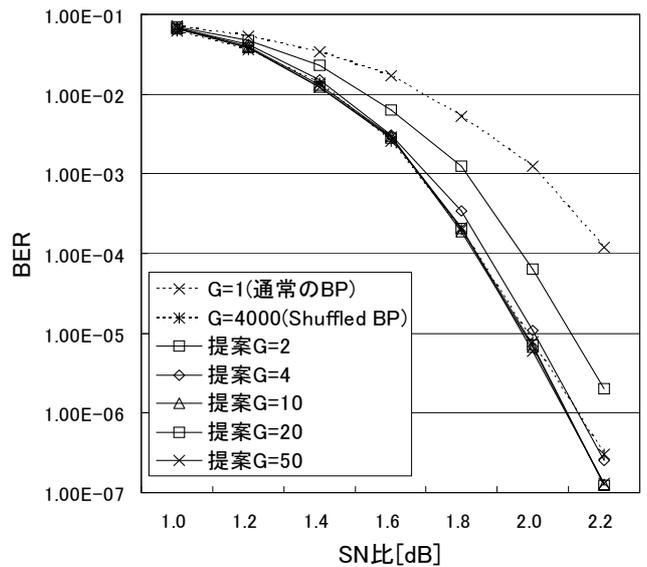


図 5: $N = 4000$, $I_{\max} = 10$ でグループ数 G を変化させたときの BER

均繰り返し回数が少なくなり、総演算回数を低減できるといえる。

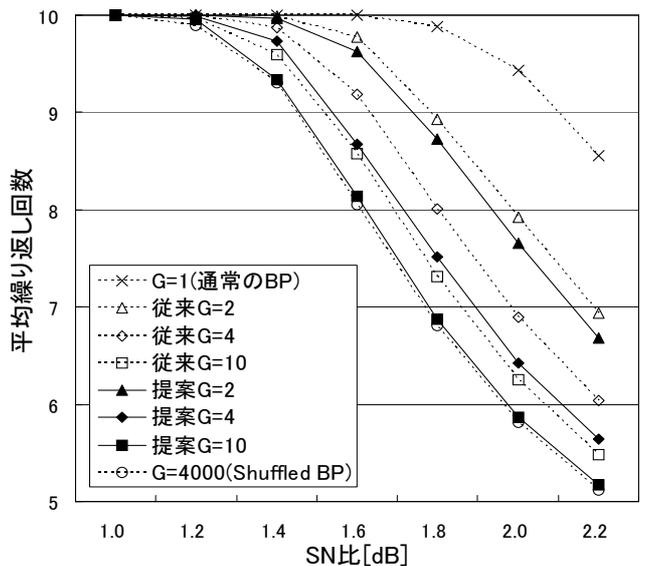


図 6: $N = 4000$, $I_{\max} = 10$ のときの平均繰り返し回数

(c) グループ分割法の考察

まず、各グループにおいてそれぞれのチェックノードが複数のシンボルノードに接続された場合、グループ内の重複したシンボルノードの総数を比較する。重複を許したシンボル数が多いほど Group Shuffled BP 復号法において繰り返し i 回目のメッセージ $z_{mn}^{(i)}$ を用いることができないといえる。 $G = 2$ のとき、従来のグループ分割法では各グループ内で重複したシンボルの数は 4030 ~ 4040 程度となり、提案したグループ分割法では 3996 ~ 3998 程度となる。また、 $G = 4$ のとき、従来のグルー

ブ分割法では 1330 ~ 1400 程度となり，提案したグループ分割法では 1000 ~ 1010 程度となる． $G = 10$ のとき，従来のグループ分割法では，250 ~ 290 程度だが，提案したグループ分割法では 10 ~ 20 程度となり， G の値が大きいほど提案した分割法によってグループ分けができることがわかる．

次に，提案したグループ分割法について考える．(a)，(b) より提案したグループ分割法は，メッセージの伝播の順番が入れ替わったことにより，復号性能，総演算量の両面で効果があると考えられる．これを検証するため，図 7 に提案したグループ分割法を用いて復号するシンボル順番を決め，各シンボルを逐次的に復号した結果を示す．図 7 より， $G = 2, 4, 10$ で提案手法を用いてグループ分割を行い各シンボルを逐次的に復号すると，Shuffled BP 復号法よりも復号性能が向上することがわかった．これは，提案手法でグループ分割を行った際シンボル位置を入れ替えることがメッセージの伝播の順序を換えたことと等価となり，復号性能に影響したためと考えられる．

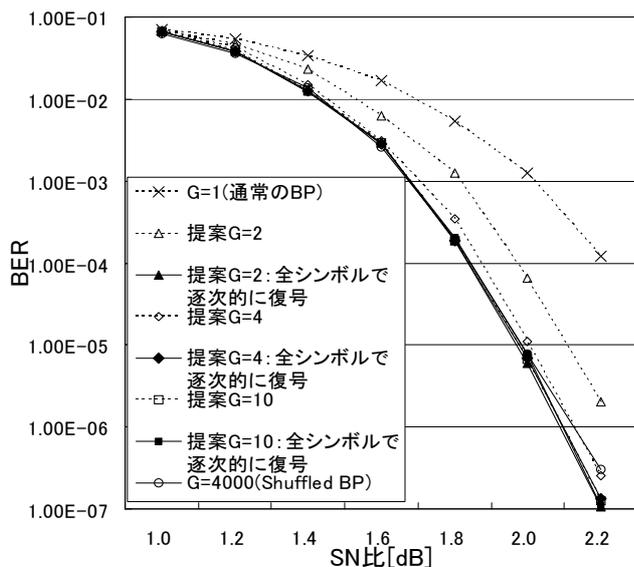


図 7: $N = 4000$, $I_{\max} = 10$ で提案によるグループ分けを行い，全シンボルを逐次的に復号したときの BER

6 まとめと今後

本研究では Group Shuffled BP 復号法における効果的なシンボルのグループ分割法を提案した．シミュレーションによる評価からグループ数 G が同じ場合，従来の Group Shuffled BP 復号法と提案の Group Shuffled BP 復号法を比較すると，復号性能が向上することを示した．また， $G = 4, 10$ の場合，提案したグループ分割法による Group Shuffled BP 復号法は Shuffled BP 復号法より復号性能が優れていることも示した． G の値が小さい方が復号の遅延も少ないため，提案したグループ

分割法は復号の遅延を減らし，さらに Shuffled BP 復号法より高い復号性能を達成することを示した．また，提案したグループ分割法は総演算量を低減できることも示した．

今後の課題として，復号性能を最適にするグループ分割法の検討や，グループの中のシンボル数を不均一に分割する方法の検討が挙げられる．本論文では効果的なグループ分けを行うことで，メッセージの伝播順序が変更されて復号性能が変化することがわかった．従来，良いメッセージの伝播順序が検討されている [8] や，従来の Shuffled BP 復号法よりも収束の速い [6],[7] などの手法と本研究で提案した手法を比較することが，今後の課題として挙げられる．

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] 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.
- [4] 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.
- [5] J. Zhang and M. P. C. Fossorier, "Shuffled iterative decoding," *IEEE Trans. Commun.*, vol. 53, No. 2, pp. 209–213, Feb. 2005.
- [6] 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.
- [7] 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.
- [8] 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, vol. 29, No. 2, pp. 823–826, Nov. 2006.
- [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.