LDPC 符号に対するタブー探索法に基づく勾配降下ビット反転復号法

高　橋　良　介（専修大学ネットワーク情報学部）
高　野　祐　一（筑波大学システム情報系）

Gradient-Descent Bit-Flipping Decoding Based on Tabu Search for LDPC Codes

Ryosuke TAKAHASHI（School of Network and Information, Senshu University）
Yuichi TAKANO（Faculty of Engineering, Information and Systems, University of Tsukuba）

This paper is concerned with bit-flipping algorithms for decoding low-density parity check codes. The gradient-descent bit flipping（GDBF）is known as a high-performance class of bit-flipping algorithms; however, once GDBF becomes trapped at local optimal solutions, it cannot escape from them. Several algorithms, including noisy gradient-descent bit flipping（NGDBF）, have been devised to overcome this drawback. For the same purpose, we develop a bit-flipping algorithm based on the tabu search, which is one of the most effective metaheuristics. The simulation results demonstrate that our algorithm achieved higher bit error rate performance than GDBF did. Moreover, our algorithm required a smaller number of iterations to find a good-quality solution than NGDBF did.

キーワード：誤り訂正，LDPC 符号，タブー探索法，ビット反転復号法，勾配降下ビット反転法

Key words：Error Correction, LDPC Codes, Tabu Search, Bit-Flipping Decoding, Gradient-Descent Bit Flipping

1. はじめに

データを通信・記録する際には，雑音（ノイズ）による誤りが生じる。CD 表面の傷や QR コードの汚れなどが雑音の原因となり，ディジタル機器にとっては僅かな誤りでさえも深刻な問題となる。このような状況下でも通信の高い信頼性を確保するためには，通信で生じた誤りを情報の受信側で訂正する，誤り訂正技術が重要となる（和田山［2010］）。

Gallager［1962］により提案された低密度パリティ検査（low-density parity check：LDPC）符号は，疎なパリティ検査行列により定義される 2 元線形符号であり，高い誤り訂正能力を持ち，高速な復号が可能である。LDPC 符号の復号法としては，メッセージ交換処理に基づく sum-product 復号法（Aji and McEliece［2000］, Kschischang et al.［2001］）や min-sum 復号法（Fossorier et al.［1999］, Wiberg［1995］）が一般的に使用される。また近年は，数理最適化手法を利用した復号法も盛んに研究されている（Feldman et al.［2005］, Helmling et al.［2012］, Tanatmis et al.［2010］，柿沼・高野［2018］）。これらの復号法は高い誤り訂正能力を有するが，一方でアルゴリズムが複雑であり実装が難しいという難点がある。

受付：2018 年 11 月 21 日
受理：2018 年 11 月 21 日

本論文では，受信語に対してパリティ検査式が成立するようにビットの反転を繰り返す，ビット反転復号法に着目する。この方法の誤り訂正能力は sum-product 復号法や min-sum 復号法に劣るが，アルゴリズムが単純であるため実装が容易であるという利点がある。それゆえビット反転復号法の性能を改善し，高性能の復号法との誤り訂正能力の差を縮めることは重要な技術的課題とされている（Wadayama et al.［2010］）。

これまでに様々なビット反転復号法のアルゴリズムが提案されているが（詳しくは Sundararajan et al.［2014］, Chang and Su［2015］の参考文献を参照されたい），その中でも Wadayama et al.［2010］により提案された勾配降下ビット反転法（gradient-descent bit flipping : GDBF）が有効とされている。一方で，素朴なビット反転復号法では局所的最適解に到達するとそこから抜け出すことができず，誤り訂正能力の改善を妨げる原因となっている。局所的最適解から抜け出すことを目的として様々な方法が提案されているが（Asatani et al.［2015］, Li et al.［2017］, Wadayama et al.［2010］），反転するビットの選択基準に雑音を加えた GDBF（noisy gradient-descent bit flipping : NGDBF）が Sundararajan et al.［2014］により提案され，その有効性が報告されている。

本論文では，タブー探索法（Glover［1989, 1990］）に基づくビット反転復号法を提案する。タブー探索法は，局所探索法を一般化した汎用的なアルゴリズムの枠組みであるメタ戦略の一種として知られている（柳浦・茨木［2001］）。タブー探索法では解の履歴をタブーリストに記録し，リストに含まれる解への移動を禁止することで，局所的最適解から抜け出して広範囲の解を探索することができる。Asatani et al.［2015］や Li et al.［2017］によって提案されたビット反転復号法は，タブー探索法の特殊例とみなすことができる。シミュレーション実験の結果，提案手法は GDBF よりもビット誤り率が小さく，NGDBF と比較して少ない反復回数で良質な解を得られることが分かった。

本論文の構成は以下のようになる。2 節では符号化通信システムの基礎知識を解説する。3 節ではビット反転復号法の既存手法と提案手法のアルゴリズムを説明する。4 節ではシミュレーション実験の結果を報告し，提案手法の有効性を検証する。5 節では本論文のまとめと今後の課題を述べる。

2. 符号化通信システム

本節では，本論文で対象とする LDPC 符号と通信路モデルについて説明し，符号化通信の例を示す。

2.1 LDPC 符号

$\mathbb{F}_2^n$ を 2 元有限体上の長さ n のすべてのベクトルの集合とする。ここでは単純に，$\mathbb{F}_2^n$ は長さ n の 2 進数のベクトルの集合と考えてもよい。添え字集合 $M := \{1, 2, \cdots, m\}$，$N := \{1, 2, \cdots, n\}$ を定義し，2 進数の行列 $\boldsymbol{H} := (h_{ij})_{(i,j)\in M\times N} \in \{0,1\}^{m\times n}$ をパリティ検査行列とする。このとき 2 元線形符号 C は，パリティ検査式 $\boldsymbol{Hc} = \boldsymbol{0}$ を満たす符号語 $\boldsymbol{c} := (c_j)_{j\in N} \in \mathbb{F}_2^n$ の集合として，以下のように定義される：

$$C := \{\boldsymbol{c} \in \mathbb{F}_2^n \mid \boldsymbol{Hc} = \boldsymbol{0}\} = \left\{\boldsymbol{c} \in \mathbb{F}_2^n \,\middle|\, \sum_{j\in N} h_{ij} c_j = 0 \quad (i \in M)\right\}.$$

例として $(m, n) = (3, 5)$ とし，以下の検査行列を考える：

$$\boldsymbol{H} = \begin{pmatrix} 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 \end{pmatrix}. \tag{1}$$

このとき 2 進数の演算であることに注意すると，2 元線形符号は以下のようになる：

$$C=\left\{(0,0,0,0,0)^{\top},\ (0,0,1,1,1)^{\top},\ (1,1,0,1,1)^{\top},\ (1,1,1,0,0)^{\top}\right\}.$$

LDPC 符号では，パリティ検査行列 $\boldsymbol{H}$ として疎行列を用いる。検査行列 $\boldsymbol{H}$ に対して，第 i 行の非ゼロ成分の列番号の集合を $N(i):=\left\{j\in N \mid h_{ij}=1\right\}$ とし，第 j 列の非ゼロ成分の行番号の集合を $M(j):=\left\{i\in M \mid h_{ij}=1\right\}$ とする。上記の例では $N(2)=\{2,3,4\}$ となり，$M(4)=\{2,3\}$ となる。LDPC 符号では $N(i)$ や $M(j)$ の要素数が少ないため，計算量が削減され高速な復号が可能となる。

2.2 AWGN 通信路

本論文では，通信路モデルとして加法的白色ガウス雑音（additive white Gaussian noise：AWGN）通信路を考える。AWGN 通信路では，符号語 $\boldsymbol{c}\in C$ に対して以下のようにバイナリ-バイポーラ変換を施し，送信語 $\tilde{\boldsymbol{c}}:=(\tilde{c}_j)_{j\in N}\in\{-1,+1\}^n$ に変換する：

$$\tilde{c}_j=\begin{cases}+1 & c_j=0 \text{ の場合}\\ -1 & c_j=1 \text{ の場合}\end{cases}\quad (j\in N).$$

このとき，

$$\sum_{j\in N} h_{ij}c_j=0 \ \Leftrightarrow\ \sum_{j\in N(i)} c_j=0 \ \Leftrightarrow\ \prod_{j\in N(i)}\tilde{c}_j=1$$

となるので，送信語 $\tilde{\boldsymbol{c}}$ に関するパリティ検査式は以下のように記述できる：

$$\mathrm{synd}_i(\tilde{\boldsymbol{c}}):=\prod_{j\in N(i)}\tilde{c}_j=1 \quad (i\in M).$$

ここで $\mathrm{synd}_i(\tilde{\boldsymbol{c}})$ はバイポーラシンドロームと呼ばれる。

通信路の雑音を表すベクトルを $\boldsymbol{z}:=(z_j)_{j\in N}\in\mathbb{R}^n$ とし，各成分 z_j は平均 0，分散 σ^2 の独立同分布の正規乱数とする。AWGN 通信路では送信語 $\tilde{\boldsymbol{c}}$ に雑音 $\boldsymbol{z}$ が加わり，受信者が受け取る受信語は $\boldsymbol{y}=\tilde{\boldsymbol{c}}+\boldsymbol{z}$ となる。

2.3 符号化通信の例

検査行列（1）に対してパリティ検査式を満たすように，送信者はメッセージを符号語 $\boldsymbol{c}\in C$ に符号化し，さらにバイナリ-バイポーラ変換を施して通信路に送る。ここでは，符号語 $\boldsymbol{c}=(1,1,1,0,0)^{\top}$ を $\tilde{\boldsymbol{c}}=(-1,-1,-1,+1,+1)^{\top}$ に変換して送信し，通信路の雑音 $\boldsymbol{z}=(0.2,0.3,-0.1,0.4,-0.3)^{\top}$ が加わって，

$$\boldsymbol{y}=\tilde{\boldsymbol{c}}+\boldsymbol{z}=(-0.8,-0.7,-1.1,1.4,0.7)^{\top}$$

が受信されたとする。

受信側の推定語を $\boldsymbol{x}:=(x_j)_{j\in N}\in\{-1,+1\}^n$ とするとき，単純な方法としては受信語の各ビットの符号に基づいて，

$$x_j=\mathrm{sign}(y_j):=\begin{cases}+1 & y_j\ge 0 \text{ の場合}\\ -1 & y_j<0 \text{ の場合}\end{cases}\quad (j\in N)$$

と推定することが考えられる。この例では $\boldsymbol{x}=\tilde{\boldsymbol{c}}$ となり，送信語 $\tilde{\boldsymbol{c}}$ が復元されて以下のようにパリティ検査式が成立する：

$$\mathrm{synd}_i(\boldsymbol{x}) = \prod_{j \in N(i)} x_j = 1 \quad (i \in M). \tag{2}$$

一方でパリティ検査式が成立しない場合には，パリティ検査式が成立するように推定語 $\boldsymbol{x}$ を修整することで，より高精度の復号が可能になる。

3. ビット反転復号法

本節では，既存手法である勾配降下ビット反転法（GDBF）を紹介し，タブー探索法に基づくビット反転復号法を提案する。

3.1 勾配降下ビット反転法

Wadayama et al.［2010］により提案された GDBF では，以下の目的関数を考える：

$$f(\boldsymbol{x}) := \underbrace{\sum_{j \in N} x_j y_j}_{(3.\mathrm{a})} + \underbrace{\sum_{i \in M} \mathrm{synd}_i(\boldsymbol{x})}_{(3.\mathrm{b})}. \tag{3}$$

式（3.a）は受信語 $\boldsymbol{y}$ と推定語 $\boldsymbol{x}$ の類似度を表し，推定語を $\boldsymbol{x} = (\mathrm{sign}(y_j))_{j \in N}$ とした場合に最大値 $\sum_{j \in N} |y_j|$ となる。また式（3.b）は，推定語 $\boldsymbol{x}$ によって成立するパリティ検査式の数を表しており，最大値は m となる。したがって目的関数（3）を最大化することで，受信語 $\boldsymbol{y}$ となるべく近く，かつパリティ検査式が数多く成立するような推定語 $\boldsymbol{x}$ を求めることができる。

推定語 $\boldsymbol{x}$ の第 k 成分の符号を反転させる関数を以下のように定義する：

$$\mathrm{flip}_k(\boldsymbol{x}) := (x_1, \cdots, x_{k-1}, -x_k, x_{k+1}, \cdots, x_n)^\top \in \{-1, +1\}^n.$$

ビット反転は $-x_k = x_k - 2x_k$ と表せることに注意すると，ビット反転後の目的関数値はテイラー展開により以下のように近似できる：

$$f(\mathrm{flip}_k(\boldsymbol{x})) \approx f(\boldsymbol{x}) - 2 \underbrace{x_k \frac{\partial}{\partial x_k} f(\boldsymbol{x})}_{\Delta_k^{(\mathrm{GD})}(\boldsymbol{x})}.$$

ここで $\Delta_k^{(\mathrm{GD})}(\boldsymbol{x})$ は反転関数と呼ばれ，その値が負の場合には，x_k のビット反転によって目的関数値の増加が期待できる。また反転関数は以下のように計算できる：

$$\begin{aligned} \Delta_k^{(\mathrm{GD})}(\boldsymbol{x}) &:= x_k \frac{\partial}{\partial x_k} f(\boldsymbol{x}) = x_k \left(y_k + \sum_{i \in M(k)} \prod_{j \in N(i) \setminus \{k\}} x_j \right) \\ &= x_k y_k + \sum_{i \in M(k)} \mathrm{synd}_i(\boldsymbol{x}). \end{aligned} \tag{4}$$

GDBF は反転関数（4）が最小になる x_k のビット反転を繰り返し，パリティ検査式が成立する推定語 $\boldsymbol{x}$ を見つけたら終了する。この方法は目的関数（3）を最大化しているので，正確には勾配上昇法と呼ぶべきであるが，慣例的に勾配降下法と呼ばれている。

GDBF のアルゴリズムは以下のようになる：

GDBF のアルゴリズム

Step 0（初期設定） 反復回数の上限 $T_{\max}$ を設定し，反復回数を $t \leftarrow 1$ とする。初期解を $\boldsymbol{x} \leftarrow (\mathrm{sign}(y_j))_{j \in N}$ とする。

Step 1（パリティ検査） パリティ検査式（2）が成立する場合には，解 $\boldsymbol{x}$ を出力して終了する。

Step 2（ビット反転） 添え字 $\hat{k} \in \arg\min_{k \in N} \Delta_k^{(\mathrm{GD})}(\boldsymbol{x})$ を求めて，$\boldsymbol{x} \leftarrow \mathrm{flip}_{\hat{k}}(\boldsymbol{x})$ とする。

Step 3（反復回数上限） $t = T_{\max}$ の場合には，解 $\boldsymbol{x}$ を出力して終了する。そうでなければ $t \leftarrow t+1$ として Step 1 に戻る。

GDBF では局所的最適解に到達すると，そこから目的関数値を改善することができずに終了してしまう。そこで Sundararajan et al.［2014］により提案された NGDBF では，局所的最適解から抜け出すために雑音を付加した以下の反転関数を利用する：

$$\Delta_k^{(\mathrm{NGD})}(\boldsymbol{x}) := x_k y_k + w \sum_{i \in M(k)} \mathrm{synd}_i(\boldsymbol{x}) + q_k. \tag{5}$$

ただし，w はパリティ検査の重みを表す非負パラメータとし，q_k は平均 0，分散 γ^2 の独立同分布の正規乱数とする。

3.2 タブー探索法

本論文では，タブー探索法（Glover［1989, 1990］）に基づくビット反転復号法を提案する。

タブー探索法では解の履歴をタブーリスト L に記録し，リストに含まれる解への移動を禁止する。具体的にはタブーリスト L の解を生成するようなビット反転を除外し，その上で反転関数（4）が最小となる x_k のビット反転を行なう。この操作により局所的最適解から抜け出して広範囲の解を探索することができるが，一方で目的関数値が減少してしまう可能性があるため，解の履歴の中で目的関数値が最大となる解を暫定解 $\hat{\boldsymbol{x}}$ として保持する。またタブーリストに追加された解は，タブー期間と呼ばれる一定の反復回数 T_{tabu} を過ぎると，タブーリストから削除する。

タブー探索法のアルゴリズムは以下のようになる：

タブー探索法のアルゴリズム

Step 0（初期設定） 反復関数の上限 $T_{\max}$ を設定し，反復回数を $t \leftarrow 1$ とする。タブーリストを空集合 $L \leftarrow \emptyset$ とし，タブー期間 T_{tabu} を設定する。初期解を $\boldsymbol{x} \leftarrow (\mathrm{sign}(y_j))_{j \in N}$ とし，暫定解を $\hat{\boldsymbol{x}} \leftarrow \boldsymbol{x}$ とする。

Step 1（ビット反転） 添え字 $\hat{k} \in \arg\min_{k \in N} \left\{ \Delta_k^{(\mathrm{GD})}(\boldsymbol{x}) \mid \mathrm{flip}_k(\boldsymbol{x}) \notin L \right\}$ を求めて，$\boldsymbol{x} \leftarrow \mathrm{flip}_{\hat{k}}(\boldsymbol{x})$ とする。

Step 2（暫定解更新） $f(\boldsymbol{x}) > f(\hat{\boldsymbol{x}})$ の場合には，$\hat{\boldsymbol{x}} \leftarrow \boldsymbol{x}$ とする。

Step 3（反復回数上限） $t = T_{\max}$ の場合には，暫定解 $\hat{\boldsymbol{x}}$ を出力して終了する。

Step 4（タブーリスト更新） タブーリスト L に解 $\boldsymbol{x}$ を追加し，$|L| > T_{\mathrm{tabu}}$ の場合には最も古い解をリスト L から削除する。$t \leftarrow t+1$ として Step 1 に戻る。

乱数に依存して解を探索する NGDBF と比較して，タブー探索法では良質な解を短時間で見つけることが期待できる。

4. シミュレーション実験

本節ではシミュレーション実験を通して，提案手法の有効性を検証する。

4.1 設定

Encyclopedia of Sparse Graph Codes（http://www.inference.phy.cam.ac.uk/mackay/codes/data.html）から，LDPC 符号 PEGReg504x1008 のパリティ検査行列のデータを取得した。このデータはビット反転復号法の性能を検証するために，多くの論文で共通して利用されている（Sundararajan et al.［2014］, Wadayama et al.［2010］）。検査行列の行数（パリティ検査式の数）は $m=504$，列数（符号語の長さ）は $n=1008$ であり，符号化率は $R=0.5$ である。

以降では，以下の手法の性能を比較する：

GDBF 反転関数（4）を用いた勾配降下ビット反転法（Wadayama et al.［2010］）
NGDBF 雑音を付加した反転関数（5）を用いた勾配降下ビット反転法（Sundararajan et al.［2014］）
TS タブー探索法（提案手法）

なお Sundararajan et al.［2014］と同様に，NGDBF のパリティ検査の重みパラメータは $w=0.75$ と設定し，反転関数に付加する雑音の分散は通信路の分散と等しい（$\gamma^2=\sigma^2$）とした。これらの復号法はすべて Python 言語を用いて実装した。

通信の信頼性（雑音の小ささ）を表す SN 比（signal-to-noise ratio：SNR）は $1/(2\sigma^2 R)$ と定義され，デシベル（dB）を単位として表示する。すなわち SN 比が A［dB］の場合は，雑音の分散は $\sigma^2=10^{-A/10}/(2R)$ となる。送信する符号語は $\boldsymbol{c}=\boldsymbol{0}$ とし，モンテカルロ法によりビット誤り率（bit error rate：BER）を推定し，復号法の誤り訂正能力を評価する。

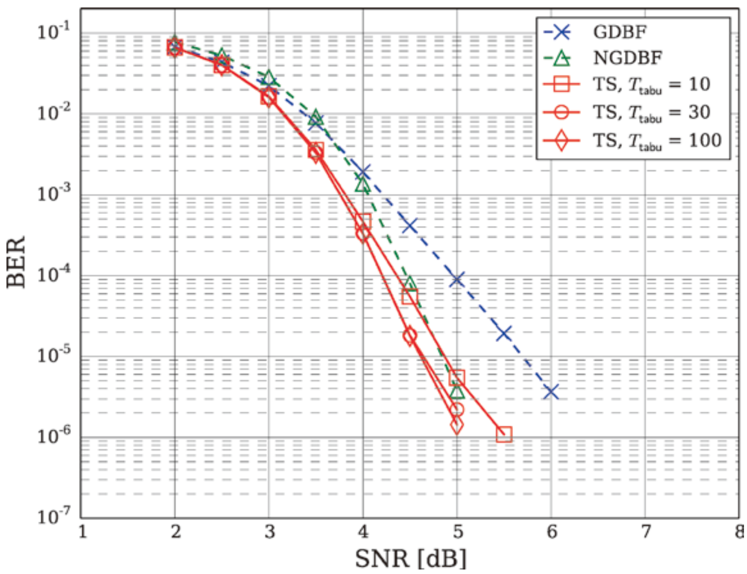

図 1 SN 比に対するビット誤り率（$T_{max}=100$）

4.2　実験結果

SN 比に対する各手法のビット誤り率の結果を図 1 に示す。ここでは反復回数上限を $T_{\mathrm{max}}=100$ と設定した。タブー探索法はすべての SN 比に対して GDBF よりもビット誤り率が小さく，SN 比が大きくなるほど二つの手法のビット誤り率の差は広がる。一方で NGDBF は，SN 比が小さい場合に誤り訂正能力が低く，GDBF よりもビット誤り率が大きくなってしまう場合がある。またタブー探索法は，タブー期間を長くするほど誤り訂正能力が向上し，タブー期間を反復回数上限の 3 割以上 ($T_{\mathrm{tabu}}=30,100$) とした場合には，すべての SN 比に対して NGDBF よりもビット誤り率が小さくなっている。

反復回数上限に対する各手法のビット誤り率の結果を図 2，3 に示す。タブー期間は反復回数上限

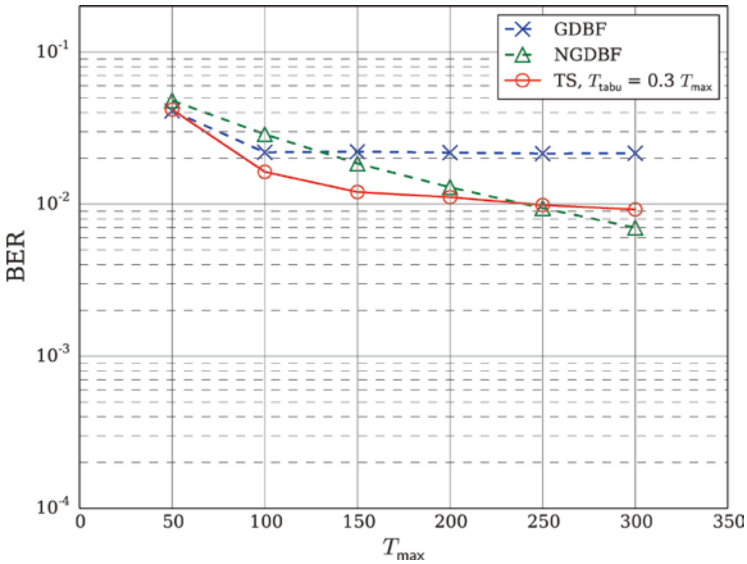

図 2　反復回数上限に対するビット誤り率（SNR ＝3 dB）

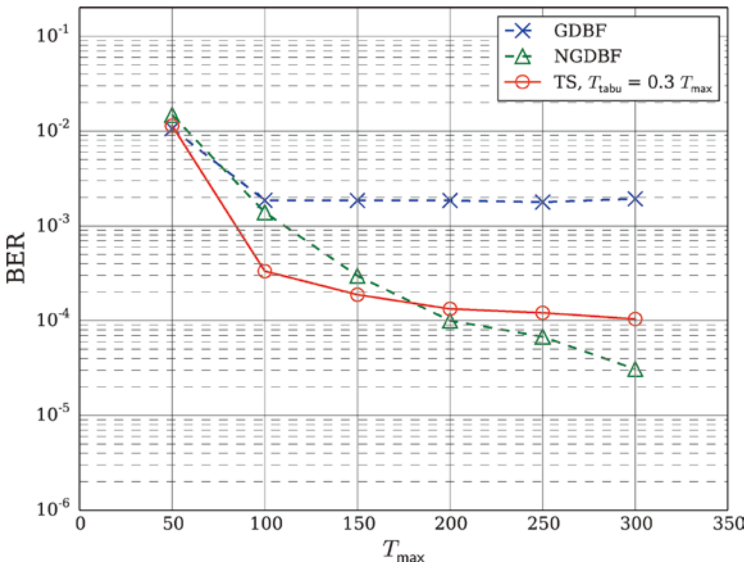

図 3　反復回数上限に対するビット誤り率（SNR＝4 dB）

の 3 割 $(T_{\mathrm{tabu}} = 0.3\,T_{\max})$ に設定し，SN 比は図 2 では 3 dB とし，図 3 では 4 dB とした。GDBF は局所的最適解に到達するとそこから抜け出すことができないため，反復回数を 100 以上に増やしてもビット誤り率はほとんど改善しない。一方で，タブー探索法と NGDBF は反復回数を増やすほど誤り訂正能力が向上し，特にタブー探索法は比較的少ない反復回数でビット誤り率を大きく減少させている。NGDBF は反復回数を十分に多くすればタブー探索法よりもビット誤り率が小さくなるが，反復回数が少ない場合に誤り訂正能力が低く，図 2 では $T_{\max} \leq 100$ の場合に GDBF よりもビット誤り率が大きくなってしまう。

5. おわりに

本論文では，タブー探索法に基づくビット反転復号法を提案した。提案手法は解の履歴をタブーリストに記録し，リストに含まれる解への移動を禁止することで，局所的最適解から抜け出して広範囲の解を探索することができる。シミュレーション実験の結果，提案手法は GDBF よりも誤り訂正能力が優れており，NGDBF よりも少ない反復回数でビット誤り率を大きく減少させられることが分かった。これらの研究成果は，誤り訂正におけるメタ戦略の有効性を示したという点でも，意義があると考えられる。

今後の課題として，以下のような方向性が挙げられる。まずタブー探索法では，解の履歴を保存して移動を禁止する以外にも様々な禁止規則があり，禁止規則を変更することでタブー探索法の性能が向上する可能性がある。また本研究では，ビットを一つずつ反転させる単数ビット反転法を対象としたが，複数のビットを同時に反転させる複数ビット反転法に対してタブー探索法を適用することも考えられる。また NGDBF は，反復回数を十分に多くすればタブー探索法よりも誤り訂正能力が優れていると考えられるため，タブー探索法と NGDBF を融合させることで，より性能の良いアルゴリズムを設計できる可能性がある。

参考文献

Aji, S.M. and McEliece, R.J., “The Generalized Distributive Law”, *IEEE Transactions on Information Theory*, Vol. 46, No. 2, 2000, pp. 325‒343.

Asatani, J., Kawanishi, H., Tokushige, H. and Katayama, K., “Frequency Memory Based Gradient Descent Bit Flipping Algorithm”, *IEEJ Transactions on Electrical and Electronic Engineering*, Vol. 10, No. 5, 2015, pp. 585‒591.

Chang, T.C.Y. and Su, Y.T., “Dynamic Weighted Bit-Flipping Decoding Algorithms for LDPC Codes”, *IEEE Transactions on Communications*, Vol. 63, No. 11, 2015, pp. 3950‒3963.

Feldman, J., Wainwright, M.J. and Karger, D.R., “Using Linear Programming to Decode Binary Linear Codes”, *IEEE Transactions on Information Theory*, Vol. 51, No. 3, 2005, pp. 954‒972.

Fossorier, M.P., Mihaljevic, M. and Imai, H., “Reduced Complexity Iterative Decoding of Low-density Parity Check Codes Based on Belief Propagation”, *IEEE Transactions on Communications*, Vol. 47, No. 5, 1999, pp. 673‒680.

Gallager, R., “Low-Density Parity-Check Codes”, *IRE Transactions on Information Theory*, Vol. 8, No. 1, 1962, pp. 21‒28.

Glover, F., “Tabu Search—Part I”, *ORSA Journal on Computing*, Vol. 1, No. 3, 1989, pp. 190‒206.

Glover, F., “Tabu Search—Part II”, *ORSA Journal on Computing*, Vol. 2, No. 1, 1990, pp. 4‒32.

Helmling, M., Ruzika, S. and Tanatmis, A., “Mathematical Programming Decoding of Binary Linear Codes : Theory and Algorithms”, *IEEE Transactions on Information Theory*, Vol. 58, No. 7, 2012, pp. 4753‒4769.

Kschischang, F.R., Frey, B.J. and Loeliger, H.A., “Factor Graphs and the Sum-Product Algorithm”, *IEEE Transactions on Information Theory*, Vol. 47, No. 2, 2001, pp. 498‒519.

Li, H., Ding, H. and Zheng, L., “Modified Gradient Descent Bit-Flipping Decoding for Low-Density Parity-Check Codes”, *Wireless Personal Communications*, Vol. 96, No. 4, 2017, pp. 6459‒6472.

Sundararajan, G., Winstead, C. and Boutillon, E., “Noisy Gradient Descent Bit-Flip Decoding for LDPC Codes”, *IEEE Transactions on Communications*, Vol. 62, No. 10, 2014, pp. 3385-3400.

Tanatmis, A., Ruzika, S., Hamacher, H.W., Punekar, M., Kienle, F. and Wehn, N., “A Separation Algorithm for Improved LP-Decoding of Linear Block Codes”, *IEEE Transactions on Information Theory*, Vol. 56, No. 7, 2010, pp. 3277-3289.

Wadayama, T., Nakamura, K., Yagita, M., Funahashi, Y., Usami, S. and Takumi, I., “Gradient Descent Bit Flipping Algorithms for Decoding LDPC Codes”, *IEEE Transactions on Communications*, Vol. 58, No. 6, 2010, pp. 1610-1614.

Wiberg, N., “Codes and Decoding on General Graphs”, *European Transactions on Telecommunications*, Vol. 6, No. 5, 1995, pp. 513-526.

柿沼美希，高野祐一，「整数最適化復号法の性能評価」，『情報科学研究』，第 38 巻，2018 年 3 月，17-24 ページ.

柳浦睦憲，茨木俊秀，『組合せ最適化—メタ戦略を中心として—』，朝倉書店，2001 年.

和田山正，『誤り訂正技術の基礎』，森北出版，2010 年.