低密度パリティ検査符号復号問題を制約なし二次形式二値変数最適化問題に変換した解法

多和田 雅師[†1,a)]　田中 宗[†2,†3]　戸川 望[†1,b)]

概要：制約なし二次形式二値変数最適化 (quadratic unconstrained binary optimization; QUBO) 問題を解くハードウェアアクセラレータの開発が進められている．QUBO 問題の計算複雑度は一般に NP 困難であり, 古典的コンピュータで効率的に解く手法は発見されていない．古典的コンピュータと異なる原理で動作するイジングマシンにより QUBO 問題を解くことが期待されている．特に実用的な問題の多くは NP 困難な組合せ最適化問題であり, QUBO 問題を介してイジングマシンで解く研究がされている．ここで低密度パリティ検査 (low density parity check; LDPC) 符号復号問題に注目する．LDPC 符号は通信路の誤り訂正を可能とする符号であり, 復号処理を組合せ最適化問題としてとらえて解法するアプローチが研究されているが QUBO 問題へ厳密に変換して解く手法は存在しない．本稿では LDPC 符号復号を組合せ最適化問題として定義し, QUBO 問題へ変換する手法を提案する．提案手法により変換された QUBO 問題をイジングマシンを用いて求解し, 元の LDPC 符号復号問題の解が得られることを示す．

キーワード：イジングマシン, QUBO, イジングモデル, LDPC 符号

1. はじめに

近年, イジングマシンと呼ばれる制約なし二次形式二値変数最適化 (quadratic unconstrained binary optimization; QUBO) 問題を解くハードウェアアクセラレータの開発が進められている．QUBO 問題の計算複雑度は NP 困難であり, 古典的コンピュータで効率的に解く手法は発見されていない．古典的コンピュータと異なる原理で動作するイジングマシンにより QUBO 問題を解くことが期待されている．特に実用的な問題の多くは NP あるいは NP 困難な組合せ最適化問題であり, QUBO 問題を介してイジングマシンで解く研究がされている．組合せ最適化問題をイジングマシンで求解可能な QUBO 問題に変換する手法が強く求められている．

通信路の誤り訂正符号として低密度パリティ検査 (low density parity check; LDPC) 符号 [1] が使用されている．LDPC 符号の復号問題は実用的な課題である．本稿では LDPC 符号復号を組合せ最適化問題として定義し, QUBO 問題へ変換する手法を提案する．提案手法により生成される QUBO 問題の解が厳密にもとの LDPC 符号復号問題の解となることを証明する．変数の個数が異なる 2 種類の符号方式を実装する．2 種類の符号方式それぞれを用いて LDPC 符号復号問題を QUBO 問題に変換し, 正しく解が得られる確率を実験により測定する．

本稿の構成は以下の通りである．2 章では関連研究を紹介し本研究の立ち位置を示す．3 章では LDPC 符号およびその復号問題を定義する．4 章では QUBO 問題を定義する．5 章では LDPC 符号復号問題をイジングマシンで解くための QUBO 問題への変換手法を提案する．6 章ではイジングマシンを用いて提案手法が実際に有効であることを示し, 7 章で本稿をまとめる．

2. 関連研究

LDPC 符号復号問題に対する既存手法にはベイズ推定を用いたアプローチとして sum-product 復号法が存在する [2]．反復計算により解を推定する sum-product 復号法は最大事後確率推定であるが, 一般的な LDPC 符号に適用する場合には近似推定であり正確な復号結果が得られるとは限らない．LDPC 符号の復号処理を組合せ最適化問題としてとらえるアプローチが存在する [3], [4]．線形計画法や内点法に基づく復号であり, 近似解を得る復号法である．イジングマシンを用いて緩和的な LDPC 符号を復号する手法が存在する [5]．LDPC 符号は軟判定復号可能な符号であるが, この手法は受信信号を硬判定した 2 値ベクトルを入力として問題を解いており, 近似的な復号法である．

LDPC 符号復号問題の計算複雑度は NP 困難であるため既存手法は厳密な LDPC 符号復号問題ではなく近似問題を対象とし, あるいは近似解を得ている．本稿の手法は, ヒューリスティックな動作をするイジングマシンを対象として LDPC 符号復号問題を QUBO 問題に厳密に変換して解く．QUBO 問題のコスト関数を最小化する理想的なイジ

†1 現在，早稲田大学基幹理工学部情報通信学科
†2 現在，早稲田大学グリーン・コンピューティング・システム研究機構
†3 現在，科学技術振興機構さきがけ
a) tawada@togawa.cs.waseda.ac.jp
b) togawa@togawa.cs.waseda.ac.jp

ングマシンがあればLDPC符号復号問題の厳密な解を得ることが保証される.

3. LDPC符号復号問題

雑音のある通信路を通じて有限長の信号を送受信することを考える. 通信路を通る信号に雑音が加わると誤った情報が伝わるため, 信号を誤り訂正符号化する必要がある. 誤り訂正符号にLDPC符号を用いるとき, 受信信号を復号して正しい情報を生成する. この復号処理の計算複雑度はNP困難となる.

3.1 通信路の定義

本稿では加法的白色ガウス雑音 (additive white Gaussian noise; AWGN) 通信路を仮定する. 通信路上を流れる信号は1シンボルで1ビットの情報を表す二位相偏移変調 (binary phase-shift keying; BPSK) とし, シンボルを$+1/-1$で表現するbipolar形式を用いる. 送信された信号は$+1/-1$の2値であるが, AWGN通信路を通じて受信された信号は正規分布に応じた雑音が加算され, 対数尤度比 (log likelihood ratio; LLR) として実数で与えられる. このときLLRが正であれば$+1$が送信された可能性が高く, 負であれば-1が送信された可能性が高いという情報を表す. LLRの絶対値はその情報の尤もらしさを表す. 送受信する情報と符号語は0/1のunipolar形式 (binary値) で表すため, 符号語を送信するとき及び復号するときにunipolar形式とbipolar形式を変換する必要がある.

送信するビット列をb, 送信する信号ベクトルをsとする. 送信する信号のシンボル数 (符号長) をnとする. 式 (2) に示すようにsのi番目の要素s_iはbのi番目の要素b_iより計算される.

$$s \in \{-1, +1\}^n \tag{1}$$

$$s_i = \begin{cases} +1 & (b_i = 0) \\ -1 & (b_i = 1) \end{cases} \tag{2}$$

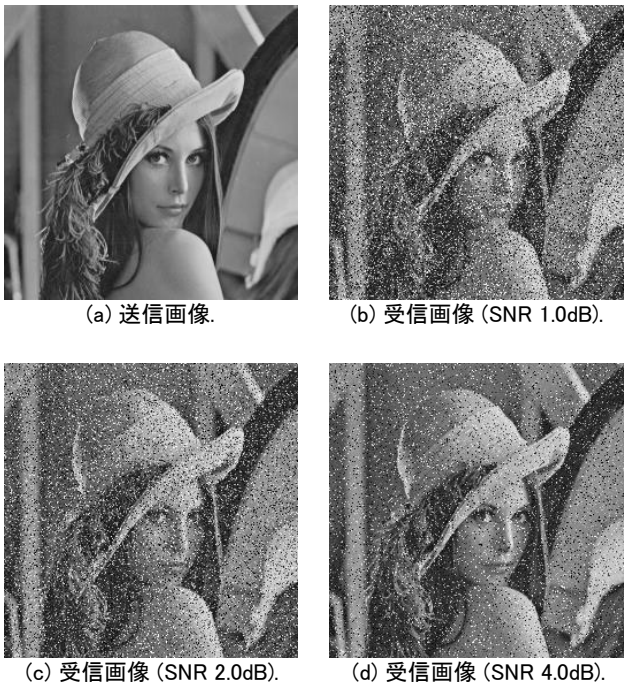
AWGN通信路による雑音をeとする. 雑音eは正規分布に従う独立な乱数e_iからなるベクトルである. 受信した信号をrとする. rのi番目の要素r_iはs_iとeのi番目の要素e_iより計算される.

$$e \in \mathbb{R}^n \tag{3}$$

$$r \in \mathbb{R}^n \tag{4}$$

$$r_i = s_i + e_i \tag{5}$$

図1にAWGN通信路で画像を送受信したときの信号雑音比 (signal noise ratio; SNR) と画像の変化を示す. 256ピクセル四方8ビットの画像を524,288シンボルの信号sとしてAWGN通信路で送受信した想定でSNRが1.0dB, 2.0dB, 4.0dBの雑音eを加算し, 得られた受信信号rの各LLRを硬判定しビットに変換し受信画像とした. 送信画像 (図1(a)) に対し受信画像 (図1(b), (c), (d)) では画像が変化していることがわかる. SNRが大きいほど雑音の分散に対する信号の分散が大きく, 送受信の画像間で変化が少なくなる.



(a) 送信画像. (b) 受信画像 (SNR 1.0dB). (c) 受信画像 (SNR 2.0dB). (d) 受信画像 (SNR 4.0dB).

図 1 AWGN通信路で画像を送受信したときの信号雑音比 (SNR) と雑音により変化した画像.

3.2 LDPC符号

LDPC符号は誤り訂正符号であり, 符号長が十分に大きいときに誤り訂正能力がシャノン限界に迫る効率的な符号である [6]. LDPC符号は符号語の特定のビットをXOR演算したときに定数0になるように構成される. この性質を表すためにパリティ検査行列と呼ばれる疎行列が用いられる.

3.2.1 LDPC符号の定義

LDPC符号は生成行列Gとパリティ検査行列Hで定義される. 符号化対象となる情報をkビット, 符号長をnビット, パリティ検査式の数をmとする. 生成行列Gはk行n列行列, 検査行列Hはm行n列行列である. Gの転置行列をG^Tとして, HとG^Tのガロア体GF(2)上の行列積はm行k列の零行列$\hat{O}$となる.

$$HG^T = \hat{O} \tag{6}$$

3.2.2 LDPC符号の符号化手順

送信する情報をv, 情報のビット長をkとしてvはk次2元列ベクトルとする.

$$v \in \{0, 1\}^k \tag{7}$$

情報から生成する符号語をw, 符号語のビット長をnとしてwはn次2元列ベクトルとする.

$$w \in \{0, 1\}^n \tag{8}$$

n行k列の生成行列Gによって符号語wはガロア体GF(2)上の行列積を使い式 (9) で計算される.

$$w = G^T v \tag{9}$$

3.2.3 LDPC 符号の復号手順

LDPC 符号は検査行列 H と符号語 w とのガロア体 GF(2) 上の行列積が m 次零ベクトル O になるという性質がある.

$$Hw = O \tag{10}$$

符号語はすべてこの性質を満たすため, LDPC 符号の誤り訂正はこの性質をもとに復号処理される. LDPC 符号の復号では受信された信号 r に対し式 (10) を満たす尤度の高い復号符号語 $\hat{w}$ を 1 つ発見する. 復号符号語 $\hat{w}$ と復号情報 $\hat{v}$, 生成行列 G は式 (11) を満たすため, ガウスの消去法により復号情報 $\hat{v}$ を求めることができる.

$$\hat{w} = G^T \hat{v} \tag{11}$$

3.3 LDPC 符号復号問題の定義

前節の準備のもと LDPC 符号復号問題を定義する. ただし $\hat{w}_i$ は $\hat{w}$ の i 番目の要素とする.

LDPC 符号復号問題

入力

- LDPC 符号 G, H
- 受信信号 r

出力

- 復号符号語 $\hat{w}$

コスト関数

- $f = \sum_{i=0}^{n-1}(r_i - (1 - 2\hat{w}_i))^2$

制約条件

- $H\hat{w} = O$

コスト関数 f を最小化する出力 $\hat{w}$ を得るのが LDPC 符号復号問題である.

4. QUBO 問題

4.1 QUBO 問題の式表現

QUBO 問題は式 (12) で表現されるコスト関数 $\mathcal{H}$ を最小化する 0/1 変数 x_i の組合せ x を探索する問題である. ただし $A_{i,j}$ を 2 次項の係数, B_i を 1 次項の係数, C を定数項とする. 変数ベクトル x の要素数を q とする.

$$\mathcal{H} = \sum_{i=0}^{q-2}\sum_{j=i+1}^{q-1} A_{i,j}x_i x_j + \sum_{i=0}^{q-1} B_i x_i + C \tag{12}$$

コスト関数 $\mathcal{H}$ を式 (13) のように変形することで, QUBO 問題は QUBO 係数行列と定数項の組合せとして表現できる. i 行 j 列の要素を $Q_{i,j}$ とする行列 Q を QUBO 係数行列とする. C を定数項とする. 本稿では Q を上三角行列とし, $i > j$ のとき $Q_{i,j} = 0$ である.

$$\begin{aligned}\mathcal{H} &= \sum_{i=0}^{q-1}\sum_{j=0}^{q-1} Q_{i,j}x_i x_j + C \\ &= x^T Q x + C\end{aligned} \tag{13}$$

$x_i \in \{0, 1\}$ より $x_i^2 = x_i$ であるため, 式 (13) は式 (12) と等価である. QUBO 問題の解 x は定数項 C に依存しない.

4.2 組合せ最適化問題の QUBO 問題定式化

組合せ最適化問題をイジングマシンで解くためには, QUBO 問題へ定式化する必要がある. 組合せ最適化問題は一般にコスト関数と制約条件が存在する. QUBO 問題には制約条件を指定できないため, 組合せ最適化問題のコスト関数と制約条件を QUBO 問題のコスト関数 $\mathcal{H}$ にスカラ関数として埋め込む必要がある. QUBO 問題のコスト関数 $\mathcal{H}$ は変数 x_i に対して高々 2 次の多項式であるため, 高次の次数が存在する場合には次数削減を考える必要がある.

ここで理想的なイジングマシンを考える. 理想的なイジングマシンは QUBO 問題のコスト関数 $\mathcal{H}$ を最小化 (最大化) する変数の組合せ x のうち, 1 通りの組合せを出力する. 組合せ最適化問題を QUBO 問題に変換して解くためには, QUBO 問題の解がもとの組合せ最適化問題の解とならなければならない.

5. 提案手法

本章では LDPC 符号復号問題を QUBO 問題に符号化する手法を提案する.

5.1 パリティ検査制約

LDPC 符号の符号語の制約条件を考える. LDPC 符号の符号語 w はパリティ検査行列 H との行列積が零ベクトルとなる. 復号符号語 $\hat{w}$ とパリティ検査行列 H との行列積が零ベクトルとなる制約を定式化する.

パリティ検査行列 H の j 行目 H_j に注目する. H_j は n 次ベクトルである. H_j と $\hat{w}$ とのガロア体 GF(2) 上の行列積はスカラ値 0 であるため, 各行 j に対して以下の式で制約する. $H_{j,i}$ を H の j 行 i 列目の要素とし, l_j を任意の整数とする.

$$H_j\hat{w} = 0 \tag{14}$$

$$\iff \sum_{i=0}^{n-1} H_{j,i}\hat{w}_i = 2l_j \tag{15}$$

式 (14) ではガロア体 GF(2) 上の行列積であり, $\hat{w}$ の特定のビットの XOR 演算の結果が 0 であることを意味する. QUBO 問題のコスト関数では XOR 演算などのガロア体 GF(2) 上の演算を直接表現できないため, XOR 演算を算術和が偶数であると解釈する. 式 (15) は任意の整数 l_j を用いて行列積の結果が偶数であるという制約を表現した. 本節の制約項を表す部分コスト関数 $\mathcal{H}_S$ を式 (16) で定義する.

$$\mathcal{H}_S = \sum_{j=0}^{m-1}(\sum_{i=0}^{n-1} H_{j,i}\hat{w}_i - 2l_j)^2 \tag{16}$$

任意の整数 l_j を 0/1 変数 $y_{j,k}$ で符号化することを考える. H_j 中の 1 の数を $|H_j|$ とするとき, l_j の値域は $0 \le l_j \le \lfloor \frac{|H_j|}{2} \rfloor$ である. このとき Unary 符号と Binary 符号いずれかを用いて l_j を符号化する [7].

- Unary 符号

  $0 \le k \le \lfloor \frac{|H_j|}{2} \rfloor$ とし, $l_j = \sum_{k=0}^{\lfloor \frac{|H_j|}{2} \rfloor} y_{j,k}$
- Binary 符号

$0 \leq k \leq \log\lfloor \frac{|H_j|}{2} \rfloor$ とし, $l_j = \sum_{k=0}^{\log\lfloor \frac{|H_j|}{2} \rfloor} 2^k y_{j,k}$

5.2 最尤推定項

LDPC 符号復号問題の最適化項について考える. 本節の項を表す部分コスト関数 $\mathcal{H}_\mathcal{C}$ に最適化項 $f = \sum_{i=0}^{n-1}(r_i - (1-2\hat{w}_i))^2$ を反映させる.

$$\begin{aligned}\mathcal{H}_\mathcal{C} &= \sum_{i=0}^{n-1}(r_i - (1-2\hat{w}_i))^2 \\ &= \sum_{i=0}^{n-1}({r_i}^2 - 2r_i(1-2\hat{w}_i) + (1-2\hat{w}_i)^2) \\ &= \sum_{i=0}^{n-1}({r_i}^2 - 2r_i + 4r_i\hat{w}_i) + n \end{aligned} \tag{17}$$

5.3 QUBO 変数と制約係数

5.1 節と 5.2 節で生成した $\mathcal{H}_\mathcal{C}$ と $\mathcal{H}_\mathcal{S}$ より, コスト関数 $\mathcal{H}$ を生成する. α を正の実数をとる制約項の係数とする.

$$\mathcal{H} = \mathcal{H}_\mathcal{C} + \alpha\mathcal{H}_\mathcal{S} \tag{18}$$

$\mathcal{H}$ は 2 値変数 x_i に対し高々 2 次の関数であり, QUBO 問題となる. 変換前の LDPC 符号復号問題に対して, 制約条件を満足する中で, 最適化項が最小となるものが解となる. コスト関数 $\mathcal{H}$ は制約条件がスカラ値として埋め込まれているので, コスト関数 $\mathcal{H}$ の最小値は, 制約項の係数 α が大きいときには制約条件を満たすが, 制約項の係数 α が小さいとき最適化項 $\mathcal{H}_\mathcal{C}$ が支配的となり制約条件を満たさない可能性がある. コスト関数 $\mathcal{H}$ が最小値をとるような変数の組合せ x のことを基底状態と呼ぶ. 基底状態が制約条件を満たすような制約項の係数 α を式 (19) で与える.

$$\alpha > \sum_{i=0}^{n-1} 4|r_i| \tag{19}$$

定理 1. 提案手法により作成されたコスト関数 $\mathcal{H}$ において, 制約係数 α が式 *(19)* を満たすとき, $\mathcal{H}$ の基底状態は *LDPC* 符号復号問題の解となる.

証明. コスト関数 $\mathcal{H}$, 部分コスト関数 $\mathcal{H}_\mathcal{C}$, $\mathcal{H}_\mathcal{S}$ を x の関数とみなし, それぞれ $\mathcal{H}(x)$, $\mathcal{H}_\mathcal{C}(x)$, $\mathcal{H}_\mathcal{S}(x)$ とする. $\mathcal{H}(x)$ の基底状態を x_{true} とする. *LDPC* 符号復号問題の制約条件を満足する任意の状態を x_{sat} とする. *LDPC* 符号復号問題の制約条件を満足しない任意の状態を x_{vio} とする. 最尤推定項 $\mathcal{H}_\mathcal{C}$ の最大値と最小値をそれぞれ ${\mathcal{H}_\mathcal{C}}^{max}$, ${\mathcal{H}_\mathcal{C}}^{min}$ とする.

式 *(17)* より式 *(20)*, 式 *(21)* が成立する.

$${\mathcal{H}_\mathcal{C}}^{max} = \sum_{i=0}^{n-1}(r_i^2 + 2|r_i|) + n \tag{20}$$

$${\mathcal{H}_\mathcal{C}}^{min} = \sum_{i=0}^{n-1}(r_i^2 - 2|r_i|) + n \tag{21}$$

よって式 *(22)* が成立する.

$${\mathcal{H}_\mathcal{C}}^{max} - {\mathcal{H}_\mathcal{C}}^{min} = \sum_{i=0}^{n-1} 4|r_i| \tag{22}$$

したがって式 *(19)*, 式 *(22)* より式 *(23)* が成立する.

$$\alpha > {\mathcal{H}_\mathcal{C}}^{max} - {\mathcal{H}_\mathcal{C}}^{min} \tag{23}$$

x_{sat} は制約条件を満足するので式 *(24)* が成立する. x_{vio} は制約条件を満足せず, $\mathcal{H}_\mathcal{S}$ は整数値となるので式 *(25)* が成立する.

$$\mathcal{H}_\mathcal{S}(x_{sat}) = 0 \tag{24}$$

$$\mathcal{H}_\mathcal{S}(x_{vio}) \geq 1 \tag{25}$$

任意の状態 x に対して ${\mathcal{H}_\mathcal{C}}^{min} \leq \mathcal{H}_\mathcal{C}(x) \leq {\mathcal{H}_\mathcal{C}}^{max}$ であるので, 式 *(26)*, 式 *(27)* が成立する.

$${\mathcal{H}_\mathcal{C}}^{min} \leq \mathcal{H}_\mathcal{C}(x_{sat}) \leq {\mathcal{H}_\mathcal{C}}^{max} \tag{26}$$

$${\mathcal{H}_\mathcal{C}}^{min} \leq \mathcal{H}_\mathcal{C}(x_{vio}) \leq {\mathcal{H}_\mathcal{C}}^{max} \tag{27}$$

式 *(18)*, 式 *(24)*, 式 *(26)* より, 式 *(28)* が成立する. 式 *(18)*, 式 *(25)*, 式 *(27)* より, 式 *(29)* が成立する.

$${\mathcal{H}_\mathcal{C}}^{min} \leq \mathcal{H}(x_{sat}) \leq {\mathcal{H}_\mathcal{C}}^{max} \tag{28}$$

$${\mathcal{H}_\mathcal{C}}^{min} + \alpha \leq \mathcal{H}(x_{vio}) \tag{29}$$

式 *(23)*, 式 *(28)*, 式 *(29)* より, 式 *(30)* が成立する.

$$\begin{aligned}&\mathcal{H}(x_{sat}) \leq {\mathcal{H}_\mathcal{C}}^{max} \\ &= {\mathcal{H}_\mathcal{C}}^{min} + ({\mathcal{H}_\mathcal{C}}^{max} - {\mathcal{H}_\mathcal{C}}^{min}) \\ &< {\mathcal{H}_\mathcal{C}}^{min} + \alpha \leq \mathcal{H}(x_{vio})\end{aligned} \tag{30}$$

ここで基底状態 x_{true} はすべての状態 x に対して $\mathcal{H}$ の最小値をとる. よって式 *(30)* より基底状態 x_{true} は制約条件を満足する. 基底状態 x_{true} と状態 x_{sat} を比較すると, ともに制約条件を満足するため, 式 *(31)* が成立する.

$$\mathcal{H}_\mathcal{C}(x_{true}) \leq \mathcal{H}_\mathcal{C}(x_{sat}) \tag{31}$$

よって最適化項が最小化されており, *QUBO* 問題の基底状態 x_{true} は *LDPC* 符号復号問題の解である. □

5.4 LDPC 符号復号問題を QUBO 問題に変換する例

前節の提案手法により LDPC 符号復号問題を QUBO 問題に変換する例を示す. 下記 LDPC 符号復号問題を考える.

- 生成行列 $G = \begin{pmatrix}1 & 1 & 1 & 0 & 0\\ 0 & 0 & 1 & 1 & 1\end{pmatrix}$
- 検査行列 $H = \begin{pmatrix}1 & 1 & 0 & 0 & 0\\ 0 & 1 & 1 & 1 & 0\\ 0 & 0 & 0 & 1 & 1\end{pmatrix}$
- 受信信号 $r = \begin{pmatrix}-1.5 & -1.2 & 1.2 & 0.2 & -0.8\end{pmatrix}$

検査行列 H よりパリティ検査制約項 $\mathcal{H}_\mathcal{S}$ を生成する. 本例ではパリティ制約を表す任意の偶数の符号方式に Unary 符号を用いる.

$$\begin{aligned}\mathcal{H}_\mathcal{S} &= \sum_{j=0}^{2}(\sum_{i=0}^{4} H_{j,i}\hat{w}_i - 2l_j)^2 \\ &= (\hat{w}_0 + \hat{w}_1 - 2y_{0,0})^2 \\ &\quad +(\hat{w}_1 + \hat{w}_2 + \hat{w}_3 - 2y_{1,0})^2 \\ &\quad +(\hat{w}_3 + \hat{w}_4 - 2y_{2,0})^2\end{aligned} \tag{32}$$

最適化項 $\mathcal{H}_C$ を生成する.

$$\begin{aligned}\mathcal{H}_C &= \sum_{i=0}^{4}(r_i^2 - 2r_i + 4r_i\hat{w}_i) + 5 \\ &= ((-1.5)^2 - 2(-1.5) + 4(-1.5)\hat{w}_0) \\ &\quad +((-1.2)^2 - 2(-1.2) + 4(-1.2)\hat{w}_1) \\ &\quad +((1.2)^2 - 2(1.2) + 4(1.2)\hat{w}_2) \\ &\quad +((0.2)^2 - 2(0.2) + 4(0.2)\hat{w}_3) \\ &\quad +((-0.8)^2 - 2(-0.8) + 4(-0.8)\hat{w}_4) + 5 \end{aligned} \tag{33}$$

定理 1 より $\alpha > \sum_{i=0}^{n-1} 4|r_i|$ ならば QUBO 問題の解が LDPC 符号復号問題の解となることを保証できる. $\sum_{i=0}^{n-1} 4|r_i|$ は 19.6 なので α を 20 とする.

QUBO 問題の変数 x を式 (34) とする.

$$\begin{aligned}&(x_0, x_1, x_2, x_3, x_4, x_5, x_6, x_7) \\ &= (\hat{w}_0, \hat{w}_1, \hat{w}_2, \hat{w}_3, \hat{w}_4, y_{0,0}, y_{1,0}, y_{2,0})\end{aligned} \tag{34}$$

以上より QUBO 問題のコスト関数 $\mathcal{H}$ が定まる.

$$\mathcal{H} = x^T Q x + C \tag{35}$$

ただし, Q,C は式 (36), 式 (37) で定める.

$$Q = \begin{pmatrix} 14 & 40 & 0 & 0 & 0 & -80 & 0 & 0 \\ 0 & 35.2 & 40 & 40 & 0 & -80 & -80 & 0 \\ 0 & 0 & 24.8 & 40 & 0 & 0 & -80 & 0 \\ 0 & 0 & 0 & 40.8 & 40 & 0 & -80 & -80 \\ 0 & 0 & 0 & 0 & 16.8 & 0 & 0 & -80 \\ 0 & 0 & 0 & 0 & 0 & 80 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 80 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 80 \end{pmatrix} \tag{36}$$

$$C = 15.01 \tag{37}$$

6. 実験評価

6.1 実験手順

提案手法により LDPC 符号復号問題を QUBO 問題に変換し, イジングマシンで解き正答率を評価する.

Step 1 LDPC 符号復号問題を作成

Step 2 提案手法を用いて LDPC 符号復号問題を QUBO 問題に変換

Step 3 イジングマシンを用いて QUBO 問題を求解

Step 4 得られた QUBO 問題の解を LDPC 符号復号問題の解に変換

提案手法への入力とする LDPC 符号復号問題の生成パラメータを表 1 に示す. 4 種類のデータセットに対して各データセットで 100 通りずつ, 合計 900 通りの LDPC 符号復号問題を生成した. 生成した LDPC 符号は pyldpc[8] で作成し, 検査行列は正則行列である. 作成した LDPC 符号復号問題を提案手法を用いて QUBO 問題に変換し, 表 2 の生成パラメータに応じて 900 通りずつ, 合計 3600 通りの QUBO 問題を作成した. 各 QUBO 問題に対しイジングマシンを用いて解を得た.

6.2 実験環境

提案手法に入力する LDPC 符号復号問題の生成には python 3.7.0 と pyldpc を用いた. 提案手法の実行には OS は CentOS 7.6, プロセッサは Intel Xeon Platinum 2.5GHz, メモリは 1.0TB を用いた.

表 1 LDPC 符号復号問題の生成パラメータ.

符号長 n [ビット]	SNR [dB]	パリティ検査行列の密度 [%]
32	1.0	12.5
32	2.0	12.5
32	4.0	12.5
128	1.0	3.1
128	2.0	3.1
128	4.0	3.1
256	1.0	1.6
256	2.0	1.6
256	4.0	1.6

表 2 QUBO 問題の生成パラメータ.

制約の符号化方式	制約項の係数 α
Unary 符号	20
Unary 符号	2,000
Binary 符号	20
Binary 符号	2,000

QUBO 問題を解くイジングマシンと実行パラメータを下記に示す. 使用するイジングマシンは係数が整数値のみ設定できるため, $\mathcal{H}$ を 50 倍にし, 整数値に丸め, QUBO 問題のコスト関数として設定した.

- QUBO ソルバ: 富士通デジタルアニーラ [9]
- 実行アルゴリズム: レプリカ交換方式 [10]
- レプリカ数: 128 個
- イタレーション回数: 100,000,000 回

6.3 実験結果・考察

提案手法を用いて LDPC 符号復号問題を QUBO 問題に変換し, 得られた QUBO 問題のパラメータを表 3 に示す. QUBO 問題を入力としイジングマシンで解いた結果を表 4, 表 5 に示す.

表 4, 表 5 より LDPC 符号復号問題のパリティ検査制約を QUBO 問題のコスト関数に符号化する場合には Binary 符号より Unary 符号の方が正答率が高いことがわかる. これはパリティ検査制約の制約充足状態の数が Binary 符号より Unary 符号の方が多く, 解空間に対する基底解の占める割合が高いためだと考えられる.

7. おわりに

本稿では LDPC 符号復号問題を定義し, LDPC 符号復号問題を QUBO 問題に変換する手法を提案した. QUBO 問題の解が LDPC 符号復号問題となるハイパパラメータの値の範囲を提案し証明した. イジングマシンを用いた実験により提案手法の有効性を示した. 今後の課題はスピン数を増加させた場合の本手法を用いた LDPC 符号復号問題の正答率を評価することである.

謝辞

本研究の成果は, 国立研究開発法人新エネルギー・産業技術総合開発機構 (NEDO) 委託業務による.

表 3 生成された QUBO 問題のパラメータ．各行 600 問の問題．

符号長 [ビット]	SNR [dB]	符号化方式	制約係数 α	変数の平均個数 [個]	係数行列 Q の 非零要素平均密度 [%]
32	1.0〜4.0	Unary 符号	20〜2,000	64.0	7.3
32	1.0〜4.0	Binary 符号	20〜2,000	64.0	7.3
128	1.0〜4.0	Unary 符号	20〜2,000	256.0	1.8
128	1.0〜4.0	Binary 符号	20〜2,000	256.0	1.8
256	1.0〜4.0	Unary 符号	20〜2,000	512.0	0.9
256	1.0〜4.0	Binary 符号	20〜2,000	512.0	0.9

表 4 QUBO 問題の正答率 (Unary 符号)．各正答率・平均実行時間は 100 問の問題と 12,800 個の解答より算出．

符号長	SNR	$\alpha = 20$		$\alpha = 2,000$	
[ビット]	[dB]	正答率 [%]	平均実行時間 [ms]	正答率 [%]	平均実行時間 [ms]
32	1.0	49.7	212.0	58.7	210.7
32	2.0	73.1	210.7	62.6	212.4
32	4.0	91.7	210.2	89.3	210.9
128	1.0	31.0	214.2	33.0	214.1
128	2.0	69.0	213.9	61.0	214.2
128	4.0	93.9	214.0	91.6	215.2
256	1.0	23.0	219.4	14.0	218.7
256	2.0	58.4	219.7	53.5	218.7
256	4.0	93.9	221.5	88.0	219.1

表 5 QUBO 問題の正答率 (Binary 符号)．各正答率・平均実行時間は 100 問の問題と 12,800 個の解答より算出．

符号長	SNR	$\alpha = 20$		$\alpha = 2,000$	
[ビット]	[dB]	正答率 [%]	平均実行時間 [ms]	正答率 [%]	平均実行時間 [ms]
32	1.0	0.64	212.4	0.65	210.4
32	2.0	0.68	212.5	0.67	212.2
32	4.0	0.87	210.8	0.84	211.0
128	1.0	0.06	214.8	0.00	213.6
128	2.0	0.07	215.0	0.08	213.7
128	4.0	0.41	214.6	0.02	214.7
256	1.0	0.00	219.6	0.00	217.6
256	2.0	0.00	219.0	0.00	218.6
256	4.0	0.00	219.0	0.00	218.6

参考文献

[1] R. Gallager, “Low-density parity-check codes,” *IRE Transactions on information theory*, vol.8, no.1, pp.21–28, 1962.

[2] F.R. Kschischang, B.J. Frey, and H.-A. Loeliger, “Factor graphs and the sum-product algorithm,” *IEEE Transactions on information theory*, vol.47, no.2, pp.498–519, 2001.

[3] J. Feldman, M.J. Wainwright, and D.R. Karger, “Using linear programming to decode binary linear codes,” *IEEE Transactions on Information Theory*, vol.51, no.3, pp.954–972, 2005.

[4] T. Wadayama, “Interior point decoding for linear vector channels based on convex optimization,” *IEEE Transactions on Information Theory*, vol.56, no.10, pp.4905–4921, 2010.

[5] Z. Bian, F. Chudak, R. Israel, B. Lackey, W.G. Macready, and A. Roy, “Discrete optimization using quantum annealing on sparse Ising models,” *Frontiers in Physics*, vol.2, p.56, 2014.

[6] C.E. Shannon, “A mathematical theory of communication,” *Bell system technical journal*, vol.27, no.3, pp.379–423, 1948.

[7] G. Rosenberg, P. Haghnegahdar, P. Goddard, P. Carr, K. Wu, and M.L. de Prado, “Solving the optimal trading trajectory problem using a quantum annealer,” *IEEE Journal of Selected Topics in Signal Processing*, vol.10, no.6, pp.1053–1060, Sep. 2016.

[8] “hichamjanati/pyldpc: Creation of LDPC codes amp; simulation of coding and decoding binary data. Applications to sound and image files.,” `https://github.com/hichamjanati/pyldpc`.

[9] S. Tsukamoto, M. Takatsu, S. Matsubara, and H. Tamura, “An accelerator architecture for combinatorial optimization problems,” *Fujitsu Sci. Tech. J*, vol.53, no.5, pp.8–13, 2017.

[10] K. Hukushima and K. Nemoto, “Exchange monte carlo method and application to spin glass simulations,” *Journal of the Physical Society of Japan*, vol.65, no.6, pp.1604–1608, 1996.