日本オペレーションズ・リサーチ学会
2021年 春季研究発表会

数理計画技法に基づいたLDPC符号の復号
— 厳密解 (最尤復号) と近似解 —

05001291　防衛大学校 理工学研究科　*渡辺 怜　WATANABE Ryo
01107880　防衛大学校 情報工学科　片岡 靖詞　KATAOKA Seiji

1　意義と目的

数理計画問題としての線形符号の最尤復号は下記の式

$$\begin{aligned} \min \quad & \boldsymbol{\lambda}^{\mathsf{T}} \boldsymbol{x} \\ \text{s.t.} \quad & H\boldsymbol{x} \equiv \boldsymbol{0} \pmod 2 \\ & x \in \{0,1\} \end{aligned}$$

で与えられる．本研究の主題はLDPC符号の復号において数理計画技法に基づいた解法を開発することであり，次の2つを目的として掲げる．

1. 最尤復号を実現するための新たな**定式化の提案**
2. 数理計画ソルバを用いたmatheuristicと呼ばれる手法を適用し，従来法より高精度の解を実用時間で求める**近似解法の開発**

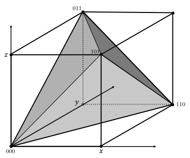
Feldmanらによる線形計画 (LP) 復号 [1] が発表されて以来，数理計画技法を用いた誤り訂正符号の復号研究が進んでいる [2]．本研究ではFeldman型の定式化を基にした新たな定式化を提案するとともに，matheuristicと呼ばれる部分的に整数条件を加えソルバによる近似解法アルゴリズムを開発する．

2　厳密解法

2.1　提案定式化 (Tetra-pile Formulation:TPF)

Feldmanらによる定式化は排他的論理和を包括的に表現できるが，超多面体構造の符号多面体を構成しており，複雑な式となっている．本研究ではより少ない変数だけで構成される簡素な多面体構造に注目し，それを組み立てることにより全体的な排他的論理和を表すことを考える．補助変数の追加により変数の数は増えるが，Feldman型の定式化よりも大幅に制約式の数を抑えることができる．

3変数による排他的論理和の表現に着目する．このとき $z = x \oplus y$ の関係は図1の四面体の頂点に相当する．



$$\begin{cases} x + y + z & \le 2 \\ -x + y + z & \le 0 \\ x - y + z & \le 0 \\ x + y - z & \le 0 \end{cases}$$

図1　$z = x \oplus y$ の関係を示す正四面体とその式表現

パリティ検査行列 H の行の非ゼロ要素に対応する変数に対して，3変数だけの表現を元に組み立てて，多変数の排他的論理和を考える．例えば6変数 (x_1〜x_6) の場合は，図2のように3つの補助変数 (z_1, z_2, z_3) を追加して構成する．図2の△一つあたりに4本の不等式で正四面体を構築している．この提案定式化は正四面体を積み上げるような構造となっているため，本研究では **Tetra-pile Formulation(TPF)** と呼ぶことにする．

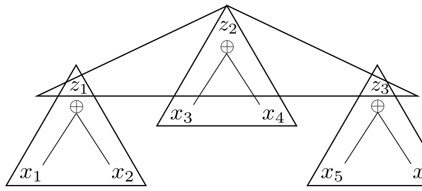


図2　$(x_1 \oplus x_2) \oplus (x_3 \oplus x_4) \oplus (x_5 \oplus x_6) = 0$ の組み立て

2.2　計算機実験 (最尤復号)

実験環境は計算機はCPU：Corei7-7820X，RAM：128G，ソルバ：Gurobi9.0を使用し，使用した行列 H はMacKay[3] のライブラリにあるもので (符号長. パリティ長) が (96.48)〜(816.408) のものを使用した．実験回数は10000回基準とするもノイズやサイズ次第で1000回として整数計画法 (Integer Programming：IP) により行った．結果は全ての定式化で最尤復号となるために誤り率は同一であるので，CPU時間 (CPU：秒) と標準偏差 (SD：秒) のみを示す．

図3〜6は H が (204.102), (408.204) で (各列. 行の非ゼロ要素数) が (3.6)，SNRが0.5〜5.0dBのものにおけるCPU時間と標準偏差の結果である．

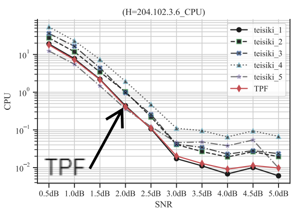


図3　204.102.3.6 の処理時間

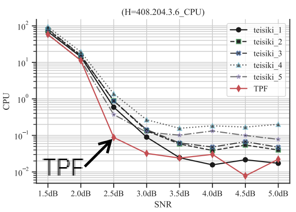


図4　408.204.3.6 の処理時間

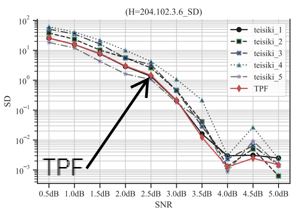


図5　204.102.3.6 の標準偏差

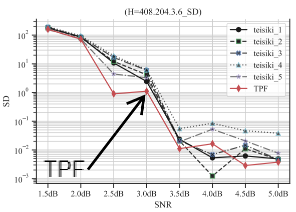


図6　408.204.3.6 の標準偏差

図3〜6において矢印で示した線がTPFの結果である．TPFは構造を簡素化したために，制約多面体の整数端点がむき出しになり易くなる．さらに制約式の数を大幅に抑えたことで，様々なサイズや密度のパリティチェック行列 H，各ノイズに対してもバランス良く実行速度が早くなり，標準偏差も安定していた．

TPFはLP復号として用いても，他の定式化よりも高速に求解できる．この特性に基づき，ソルバによる近似解法においても，TPFを用いることにする．

3 近似解法

3.1 sum-product 復号法からの情報

効果的な近似解法を開発するために，sum-product 復号法の実行過程で得られる情報について考える．sum-product 復号法は反復処理により精度を高める確率推論アルゴリズムであり，このアルゴリズムによる復号過程は特徴的な性質を持っている．図 7・8 は SNR が 0.5dB と 1.5dB での出力結果の推移である．

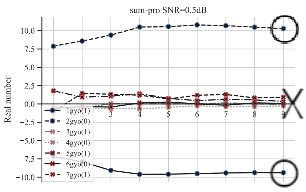


図 7　0.5dB での復号過程

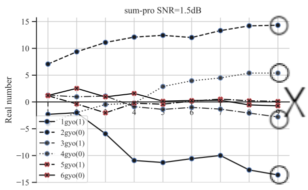


図 8　1.5dB での復号過程

○が正しく復号できたビット，× が誤って復号されたビットである．正しく復号される可能性が高い場合は $+(-)$ 値に大きく振れていき，逆に復号が誤る場合は 0 の周辺で値が推移する傾向にある．sum-product 法からの出力値を sumpinf として近似解法に活用する．

3.2 matheuristic 復号法

matheuristic とは探索範囲を広くとる代わりに，探索範囲での最適解をソルバで求める手法である．復号法で用いる場合は正しく復号される可能性の高い変数 (絶対値が大きい値) を 0(1) に固定し，誤る可能性が高い変数 (絶対値が 0 に近い値) を 0-1 変数として設定する．0-1 変数が混在しているが，数を少なくしてソルバにより最適に解く．

復号法の流れは (図 9) まず TPF により IP より高速な線形計画法として問題を解き，0-1 解が得られたときは最適性保証付きとして終了する．0-1 解が得られないときには matheuristic として一部の変数を固定・0-1 設定としてソルバで解く．

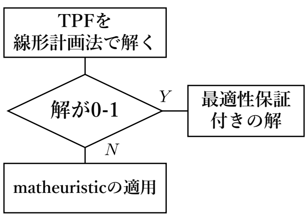


図 9　matheuristic 復号法

この方法を **matheuristic 復号法**と呼ぶことにする．

変数の固定・設定方法は検査行列 H 全体で $h_{ij} = 1$ となる箇所に $\mathrm{sumpinf}_j (j \in \{1, \cdots, n\})$ を代入する．そして複数行をブロックとして $B(B \subseteq \{1, \cdots, m\})$ を定める．B の範囲で代入された $\mathrm{sumpinf}_j$ の絶対値の大小を調べる．この結果により次のように固定・設定を行う．

1. 絶対値が最大のとき：

$$p_B = \arg\max \left\{ |\mathrm{sumpinf}_j|, h_{ij} = 1, i \in B, j \in \{1, \cdots, n\} \right\}$$

として $\mathrm{sumpinf}_{p_B} > 0 (< 0)$ の時 $x_{p_B} = 0(1)$ に固定

2. 絶対値が最小のとき：

$$q_B = \arg\min \left\{ |\mathrm{sumpinf}_j|, h_{ij} = 1, i \in B, j \in \{1, \cdots, n\} \right\}$$

として x_{q_B} を 0-1 変数に設定

B の中で ℓ 番目までの最大・最小を選ぶことも考える．計算機実験により各パラメータを調整し $|B|$ は全体の 15% 程度, 固定変数 3 個, 設定変数 4 個 が速度と精度のバランスが良いことを確認した．

3.3 計算機実験 (matheuristic 復号法)

計算機実験は，2 章で行ったものと同じ環境と同じ MacKay[3] によるパリティ検査行列 H を用いた．実験回数は 10000 回を基準とするも一部大きなサイズでは 1000 回で実験を行った．結果は従来復号法である sum-product 復号法と TPF(IP) で平均ビット誤り率 (AveBitE：率)，CPU 時間 (CPU：秒) を比較した．図 10〜13 は H が (96.48), (204.102) での結果を示す．

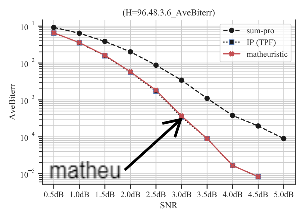


図 10　96.48.3.6 の誤り率

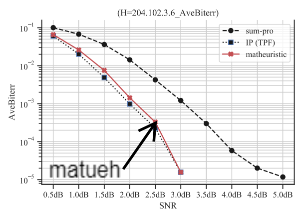


図 11　204.102.3.6 の誤り率

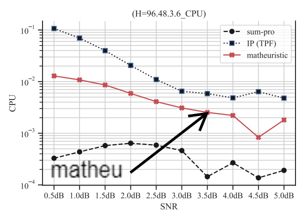


図 12　96.48.3.6 の処理時間

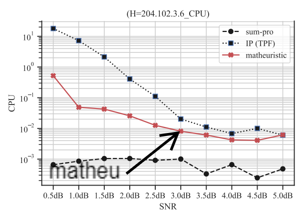


図 13　204.102.3.6 の処理時間

matherutistic 復号法は LDPC 復号問題において顕著に効果があったといえる（図 10〜13 で矢印で示した線）．IP による最尤復号には及ばないが sum-product 復号法よりも格段に精度が良く，IP を使用したときよりも実行時間を大きく改善したできた．これは IP より格段に早い線形計画法を用いているだけでなく，matheuristic により妥当性のある変数設定を行ったために，誤り率を低く抑えたまま高速化が可能となった．

4 結論

構造を簡素化した定式化 TPF を提案し，従来よりも安定した高速な最尤復号が可能となった．またソルバによる近似解法である matheuristic 復号法により，実行時間を大幅に抑えたまま従来法よりも高精度の近似解を得ることができた．本研究により，新しい考え方である matheuristic の有用性が示せたと考える．

参考文献

[1] J.Feldman et al.：Using LP to decode binary linear codes. *IEEE Trans. Info.Theory*, **51**(2005), 954-972.

[2] M.Helmling et al.：MP of binary linear codes. *IEEE Trans. Info. Theory*, **58**(2012), 4753-4769.

[3] D.J.C. MacKay: *Encyclopedia of Sparse Graph Codes*, `http://www.inference.phy.cam.ac.uk/mackay/` .