

ピボットの置換による符号化率が可変なパンクチャド低密度パリティ検査符号の構成法

1G06H027-7 長田佳史
指導教員 後藤正幸

1 研究目的

近年、情報化社会における情報通信技術の発達と共に、情報伝送に対する信頼性の確保が不可欠となっており、誤り訂正符号・復号技術の果たす役割が重要になっている。

誤り訂正符号の中でも低密度パリティ検査 (LDPC) 符号は、繰り返し型確率伝播 (BP) 復号を行うことで優れた復号性能を示すことが知られている。誤り訂正符号は符号化率と訂正能力がトレードオフの関係にあるため、通信路の状態により適切な符号化率をもつ符号を用いる必要があるが、符号化率ごとに符号を用意することは効率的ではない。そのため、送信する符号語ビットを削って (パンクチャ) 送信しないことで、1つの符号を用意だけで符号化率を可変にする方法が提案されている [1]。

LDPC 符号と BP 復号法の組み合わせに対して、パンクチャするビット位置とパンクチャするビットの順番を決定する符号化率が可変なパンクチャド LDPC (RCP-LDPC) 符号の構成法に関して様々な研究が行われている。パンクチャするビットの選択法として符号語ビットが復元できるまでの繰り返し回数 (復元レベル) に基づくパンクチャ法 [1] が良い復号性能をもつが、多くのビットをパンクチャできないため、高い符号化率を構成できないことが問題である。

一方、パンクチャできるビット数を大きくするために、ピボットと呼ばれる符号語ビットと他の符号語ビットを入れ替える (置換する) 手法 [2] があるが、元々はパンクチャを対象として考案された手法ではないため、パンクチャド符号の性能に影響する復元レベルを考慮していない。

そこで本研究では、復元レベルを考慮しつつピボットを探索してパンクチャできるビット数を大きくすることで、符号化率の可変な範囲を広げた RCP-LDPC 符号の構成法を提案し、計算機シミュレーションによる結果から提案手法の有効性を示す。

2 LDPC 符号と通信路モデル

本研究では、符号長を N 、検査ビット数を $M (< N)$ とした 2 元の LDPC 符号を対象とする。LDPC 符号のパリティ検査行列は各列を変数ノード、各行をチェックノードとした 2 部グラフ (タナーグラフ) で表すことができ、 $n = 1, 2, \dots, N$, $m = 1, 2, \dots, M$, $N > M$ 、において変数ノード n とチェックノード m は対応する検査行列の m 行 n 列目の要素が 1 であるとき枝で接続される。このときの符号化率 R_0 は $R_0 = (N - M)/N = 1 - M/N$ と表される。

LDPC 符号の符号語 $c = (c_1, c_2, \dots, c_N)$, $c_n \in \{0, 1\}$ 、を加法的白色ガウス雑音 (AWGN) 通信路を介して送信し、雑音を加わった受信語 $y = (y_1, y_2, \dots, y_N) \in \mathcal{R}^N$ を得る。ここで \mathcal{R} は実数を表し、変数ノードと符号語ビットの位置は 1 対 1 に対応する。また受信側では受信語 y から推定系列に復号する。

3 従来研究

3.1 RCP-LDPC 符号 [1]

符号化率の高いパンクチャド符号は、符号化率の低い符号の符号語ビットをパンクチャすることで構成できる。 $P (< M)$ ビットをパンクチャしたときに得られる符号の符号化率 R は $R = (N - M)/(N - P)$ と表される。

各ビット位置 $n = 1, 2, \dots, N$ に対して対数尤度比 (LLR) の初期値を $U_{ch;n} = \log\{\Pr(y_n|c_n = 0)/\Pr(y_n|c_n = 1)\}$ と表す。パンクチャしたビット位置の LLR は初期値が 0 であり、送信したビットが 0 か 1 か判定できないため、BP 復号法の実行過程においてパンクチャしたビットの LLR を非零に更新する (復元する) 必要がある。パンクチャしたビット位置に対応する変数ノード (パンクチャド変数ノード) を v とすると、タナーグラフ上において、 v と接続するチェックノードのうち 1 つでも v 以外のパンクチャド変数ノードと接続しないとき v は復元される。全てのパンクチャド変数ノードは以下で定義する復元レベルによって分類できる。定義 1: BP 復号法の繰り返し $i (\geq 0)$ 回目で復元されるパンクチャド変数ノードを i -SR ノードと呼ぶ。ただし 0-SR ノードはパンクチャされていない変数ノードを表す。□パンクチャド LDPC 符号は、できるだけ少ない繰り返し回数で復元した方が復号性能が良いことが知られている [1]。

符号化率 R_0 の符号からパンクチャによって目的の符号化率 R の符号を生成するために必要なパンクチャド変数ノードの数 T は $T = \lfloor N(R - R_0)/R \rfloor$ で得られる。ここで実数 A に対して $\lfloor A \rfloor$ は A 以下の最大の整数を表す記号である。

RCP-LDPC 符号のパンクチャを行うにあたっては、最大でパンクチャするビット数とパンクチャする位置及び順番を決めておけば様々な符号化率に対応して符号が構成できる。

3.2 ピボットの探索法 [2]

長さ L の連続したビット位置をパンクチャする位置の候補 (パンクチャ候補) と仮定する。このパンクチャ候補が復元可能で長さ $L + 1$ に伸ばしたときに復元不可能である場合、ピボットを以下のように定義し、その探索法を述べる。定義 2: 長さ $L + 1$ のパンクチャ候補のうち、ある 1 つの変数ノードをパンクチャしていない場合に長さ $L + 1$ のパンクチャ候補が復元可能であるとき、このような変数ノードをピボットと呼ぶ。□

[ピボット探索法]

初期設定) 長さ L のパンクチャ候補の両端の変数ノードをピボット集合として初期化する。

step1) ピボット集合のそれぞれのピボットに対し、隣接するチェックノードと長さ L のパンクチャ候補内の変数ノードとの接続数が 1 のとき、それらの変数ノードをピボットとし、ピボット集合に追加する。

step2) step1) で新しいピボットが発見されなかった場合、ピボット集合を出力してアルゴリズム終了する。それ以外の場合、step1) へ戻る。□

4 提案手法

RCP-LDPC 符号の構成法として、復元レベルに基づくパンクチャ法 [1] が提案されているが、高い符号化率のパンクチャ符号を構成できないという問題がある。そこで本研究では、3.2 節で説明したピボット探索法を利用して符号化率が高く設定でき、復元レベルが小さい RCP-LDPC 符号の構成法を提案する。まず提案する構成法では、与えられたタナグラフに対して復元レベルが小さくて復元可能な長さ L が大きい (長さ $L + 1$ では復元できない) パンクチャ候補を予め調べ、ピボット探索法を実行する。次に、探索されたピボットである変数ノードとパンクチャ候補の長さを大きくする変数ノードの置換を繰り返していくことで、最大でパンクチャする数を大きくしていく。

[探索ピボットの置換法]

初期設定) $l := L + 1$ とし、試行回数 F_{\max} を設定する。

step1) 長さ l のパンクチャ候補が復元不可能な場合、ピボット探索法を実行し、ピボットを全て探索する。もしこの長さで復元可能な場合は、step5) へ行く。

step2) step1) で得られたピボット集合の中から変数ノードを 1 つランダムに選択する。

step3) 長さ l のパンクチャ候補を除く変数ノードの中から step2) で得られたピボットと置換したとき、パンクチャ候補が復元可能で、なおかつ現在の最大復元レベルが大きくなり変数ノードを探索するまで最大 F_{\max} 回繰り返し、発見されればそのノードを選択する。発見されない場合は、復元レベルを考慮せず、パンクチャ候補を復元可能にする変数ノードから 1 つをランダムに選択する。そのようなノードもない場合は、現在までに得られたパンクチャ候補とタナグラフを出力してアルゴリズムを終了する。それ以外の場合、step4) へ行く。

step4) step2) と step3) で選択したピボットと変数ノードを置換する。

step5) $l := l + 1$ とし、右端が左端に 1 伸ばした長さ l のパンクチャ候補を新たな候補として step1) へ戻る。□

上記のアルゴリズムを実行して得られた復元可能なパンクチャ候補に含まれる変数ノードを目的の符号化率 R に応じてパンクチャする。ただし、符号化率を可変とするためにパンクチャ候補の中からパンクチャする変数ノードの順番は予め決めておく [1]。このようにすることで目的符号化率 R に応じて最大復元レベルが小さく、少ない繰り返し回数で多くのパンクチャ変数ノードを復元できる RCP-LDPC 符号を構成することができる。

5 計算機シミュレーションによる実験結果及び考察

提案手法の有効性を検討するために計算機シミュレーションにより評価を行った。

5.1 シミュレーション条件

実験にはランダムに構成した $N = 1000$ 、符号化率 $R_0 = 0.5$ の LDPC 符号を用いて、AWGN 通信路を介して 1×10^6 個の符号語を送信した。提案手法を実行して符号化率 $R = 0.80, 0.90, 0.95$ の符号を構成した。全ての変数ノードの中からランダムにパンクチャしたものをランダム、復元レベルを考慮せず復元可能な長さ L が最も大きくなるように置換した変数ノードの集合をパンクチャしたものを比

較手法として提案手法との比較に用いた。ここで、最大でパンクチャする数は置換法で出力されたパンクチャ候補の長さの値とし、パンクチャする順番はランダムに決定するものとする。

5.2 結果及び考察

図 1 に実験結果を示す。ここで縦軸は復号後のビット誤り率 (BER) を表し、横軸は AWGN 通信路の信号対雑音比 (SN 比) [dB] を表す。点が下にあるほど復号性能は良く、SN 比が上がっても BER が減少してない場合には、復号可能な符号が構成されていないことを意味する。

この結果より、

- (1) 提案手法では、従来研究の復元レベルに基づくパンクチャ法 [1] やランダムでは構成できない高い符号化率の符号を構成できていることがわかる。
- (2) 提案手法は比較手法と同等の高い符号化率をもち、復元レベルが小さくなるように構成したことで、ランダム、比較手法よりも復号性能の良い符号を構成できていることがわかる。
- (3) $R = 0.95$ における最大復元レベルを調べると提案手法は 17-SR、比較手法は 45-SR となっており、提案手法により復元レベルを小さくできていることがわかる。

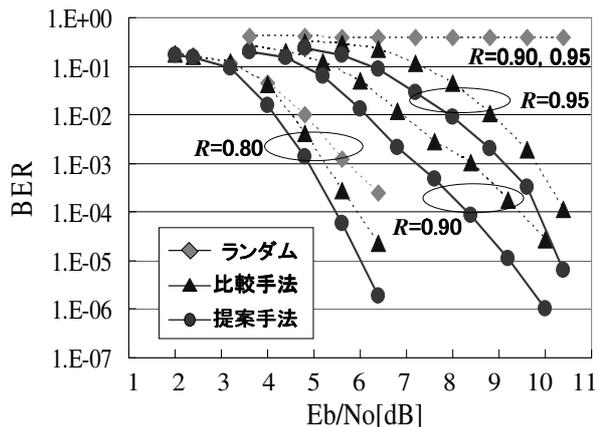


図 1. 復号結果

6 まとめと今後の課題

本論文では復元レベルを考慮しつつ、ピボットとパンクチャ候補を長くする変数ノードを置換することで高い符号化率をもつ RCP-LDPC 符号の構成法を提案した。実験結果より提案手法の性能は比較手法、ランダムより良いことを示した。

今後は、復元レベルが小さくなるように、提案手法を実行して得られた復元可能なパンクチャ候補に含まれる変数ノードのパンクチャする順番を決定することが課題である。

参考文献

- [1] J. Ha, J. Kim, D. Klinc, and S. W. McLaughlin, "Rate-compatible punctured low-density parity-check codes with short block lengths," *IEEE Trans. Inform. Theory*, Vol. 52, No. 2, pp. 728–738, Feb. 2006.
- [2] E. Paolini and M. Chiani, "Construction of near-optimum burst erasure low-density parity-check codes," *IEEE Trans. Commun.*, Vol. 57, No. 5, pp. 1320–1328, May 2009.