

Rank Modulation 方式の符号の新しい評価指標

New Performance Index of Code for Rank Modulation Scheme

戸部雅人
Masato Tobe鎌部浩
Hiroshi Kamabe岐阜大学
Gifu University

1 はじめに

フラッシュメモリでデータを読み書きするために提案されているデータ表現方法の1つに、Rank Modulation 方式 (RM 方式) がある [1]。この方式は、ブロック内でのセルの相対的な大きさに着目している。Andrew らはこの方式で書き込むとき、平均符号語長を最適化する Prefix-Free 符号の構成法を与えた [1]。

その構成方法で最適化された評価指標では、ある順列に遷移するためのコストとして、遷移に必要な基本操作の回数の最悪値を採用している。しかし、別の順列への遷移に必要な基本総裁の回数は最悪値より少ない場合がある。本稿では、このような場合を考慮した、より厳密な評価基準を提案する。

2 Rank Modulation 方式

RM 方式ではブロック内にあるセルの電荷量の相対関係によってデータを表す。一つのブロックには n 個のセルがあり、各セルは異なる電荷量を持っていると仮定する。電荷量の相対関係を、順列 $[a_1, a_2, \dots, a_n]$ で表す。この時 a_1 を最も電荷が高いセル、 a_n を最も電荷が低いセルとする。例えば $n = 4$ で、 $[2, 3, 4, 1]$ の時、最も電荷量が高いセルは 2 番目のセルであり、また最も電荷量が低いセルは 1 番目のセルである。

RM 方式において、電荷量の相対的な大きさの変更は、「push to the top」操作という基本的な操作のみによって実行される。push to the top 操作とは n 個のセル $[a_1, a_2, \dots, a_n]$ から、セル a_i を選び、ブロック内で最も電荷が高くなるように、電荷量を増加させて、 $[a_i, a_1, a_2, \dots, a_{i-1}, a_{i+1}, \dots, a_n]$ とする操作のことである。

ブロック内の相対関係の集合を W_n 、格納される情報シンボルの集合を $\{1, \dots, l\}$ とする。RM 方式における電荷の書き込みは、以下の2つの関数を定義することで実現される。順列 $s \in W_n$ から情報ビット $i \in \{1, \dots, l\}$ へ写像する復号関数 $\phi(s) = i$ と、順列 s と更新シンボル $i' \in \{1, \dots, l\}$ を引数として、 $\phi(s') = i'$ を満たすような $s' \in W_n$ を返す更新関数 $\mu(s, i') = s'$ である。

$s, s' \in W_n$ が与えられたとき s から s' に変換するために必要な push to the top 操作の回数の最小値をコスト $cost(s \rightarrow s')$ とする。

3 Prefix-Free Code

集合 $\{1, 2, \dots, n\}$ から、 $m \leq n$ 個の異なるシンボルを選んで作られる長さ m の系列 a を考える。長さが n の順列のうち、系列 a から始まる集合を $P_n(a)$ とする。

情報記号の数 l が与えられた時、Prefix-Free 符号は l 個の排反な集合 $P_n(a_1), \dots, P_n(a_l)$ を決めることで構成される。 $i, j \in \{1, \dots, n\}$ に対して、 $s_i \in P_n(a_i), s_j \in P_n(a_j)$ の時、復号関数 ϕ は $\phi(s_i) = i$ と与えられ、更新関数 μ は $\mu(s_i, j) = s_j$ となる。

系列 a_i の長さを $|a_i|$ とすると任意の順列から、 $P_n(a_i)$ に含まれる順列へは、最悪でも $|a_i|$ 回のコストで遷移できる。情報記号 i が確率 p_i で発生すると仮定した時、Prefix-Free 符号の指標となる平均符号語長は $\sum_{i=1}^l p_i |a_i|$ と定義される。Andrew らは n と l , $p_1 \leq p_2, \dots, \leq p_l$ が与えられた時に再帰処理を繰り返すことによって、平均符号長 $\sum_{i=1}^l p_i |a_i|$ を最適化する Prefix-Free 符号の構成法を提案した [1]。

4 性能指標の提案

Prefix-Free 符号において、 $P(a_j)$ 内の順列から $P(a_i)$ 内の順列へ遷移するときのコスト $cost(P(a_j) \rightarrow P(a_i))$ を $\max_{s_j \in P(a_j)} \{ \min_{s_i \in P(a_i)} cost(s_j \rightarrow s_i) \}$ とする。さらに $P(a_i)$ の平均コスト $\overline{cost}(a_i)$ を $\sum_{j=1, j \neq i}^l p_j cost(P(a_j) \rightarrow P(a_i))$ とする。Prefix-Free 符号の新しい評価基準である平均コスト長を以下に定義する。

$$\sum_{i=1}^l p_i \overline{cost}(a_i) \quad (1)$$

これは、[1] で提案された評価よりも、より厳密な平均符号長を計算していることになる。Andrew らが提案した符号構成アルゴリズム [1] は、系列の長さが同じ prefix 集合に対するコストを、最悪値で評価する構成方法である。しかし、平均コスト長を考えると、prefix 集合にも順序を設け、prefix 集合に対する記号の割り当て方法を工夫することによって、平均符号長の短い符号が構成できる。文献 [1] の Example 22 で構成された符号にたいして、本稿の評価関数を計算すると、1.2921 となる。しかし、この例において符号木の記号の割り当てを変更することで、評価関数の値が 1.2671 となる符号を構成できる。

5 まとめ

本稿では RM 方式の符号の一つである Prefix-Free 符号の新しい指標を提案した。

参考文献

- [1] A. Jiang, V. Bohossian, and J. Bruck, "Rank Modulation for Flash Memories", *IEEE Trans. on Inform. Theory*, vol. 55, no. 6, June, pp.2659-2673, 2009.