Linear-Time Encoders for Two-Dimensional Bounded-Weight Constrained Codes

Tuan Thanh Nguyen, Kui Cai, Kees A. Schouhamer Immink, and Yeow Meng Chee

***Abstract*—In this work, given $n, p > 0$, efficient encoding/decoding algorithms are presented for mapping arbitrary data to and from $n \times n$ binary arrays in which the weight of every row and every column is at most pn. Such constraint, referred to as p-bounded-weight-constraint, is crucial for reducing the parasitic currents in the crossbar resistive memory arrays, and has also been proposed for certain applications of the holographic data storage.**

I. Introduction

In resistive memories, the memory cell is a passive two-terminal device that can be both read and written over a simple crossbar structure [1], [2], which facilitates a huge density advantage. However, a fundamental and challenging problem of the crossbar memory arrays is the *sneak path* problem. When a cell in a crossbar array is read, a voltage is applied upon it, and current measurement determines whether it is in a low-resistance state (LRS, corresponding to a '1') or a high-resistance state (HRS, corresponding to a '0'). The sneak path is due to an effect by which in parallel to the desired measurement path, alternative current paths through other array cells distort the measurement, which may result in reading an erroneous state. The sneak path problem was addressed by numerous works with different approaches and at various system layers [3], [4]. An effective method to reduce the sneak path effect is by enforcing fewer memory cells with the LRSs. This can be achieved by applying constrained coding techniques to convert the user data into a 2D-constrained array that limits number of 1's in every row and every column. For example, Ordentlich and Roth [3] required the weight in every row and every column to be at most half, and presented efficient encoders with redundancy at most $(2n-1)$ for $n \times n$ array. In [5], the authors studied the bounds of codes that required the weight in every row and every column is precisely pn and provided a coding scheme based on the *enumeration coding technique*. The redundancy of the proposed encoder was at most $2n\mu(n,p) + O(n + \log n)$, where $\mu(n,p)$ is the least redundancy required to encode one-dimensional binary codewords of length n such that the weight is pn. Besides resistive memories, this constraint has been proposed for applications in holographic storage systems as well [5], [6].

In this work, we provide efficient designs for $n \times n$ binary arrays that satisfy the *p-bouned-weight-constraint*, in which the weight in every row and every column is at most pn for arbitrary $p \in (0,1)$. Due to space constraints, we only summarise the results and describe the main idea of the algorithms. Details can be found in [7] and the results have been presented in ISIT 2021.

Tuan Thanh Nguyen and Kui Cai are with the Singapore University of Technology and Design (email: {tuanthanh_nguyen, cai_kui}@sutd.edu.sg). The work of Kui Cai and Tuan Thanh Nguyen is supported by Singapore Ministry of Education Academic Research Fund MOE2019-T2-2-123.

Kees A. Schouhamer Immink is with the Turing Machines Inc, Willemskade 15d, 3016 DK Rotterdam, The Netherlands (email: schouhamerimmink@gmail.com).

Yeow Meng Chee is with the Department of Industrial Systems Engineering and Management, National University of Singapore (email: ymchee@nus.edu.sg). The research of Yeow Meng Chee is supported by the Singapore Ministry of Education under grant MOE2017- T3-1-007.

This paper was presented at 2021 IEEE International Symposium on Information Theory [7].

A. Notation

For a binary sequence $\boldsymbol{x}$, we use $\mathrm{wt}(\boldsymbol{x})$ to denote the weight of $\boldsymbol{x}$, i.e the number of ones in $\boldsymbol{x}$. We use $\overline{\boldsymbol{x}}$ to denote the complement of $\boldsymbol{x}$. For example, if $\boldsymbol{x} = 00111$ then $\mathrm{wt}(\boldsymbol{x}) = 3$ and $\overline{\boldsymbol{x}} = 11000$. Given two binary sequences $\boldsymbol{x} = x_1 \ldots x_m$ and $\boldsymbol{y} = y_1 \ldots y_n$, the *concatenation* of the two sequences is defined by $\boldsymbol{xy} \triangleq x_1 \ldots x_m y_1 \ldots y_n$.

Let $\boldsymbol{A}_n$ denote the set of all $n \times n$ binary arrays. The ith row of an array $A \in \boldsymbol{A}_n$ is denoted by A_i and the ith column is denoted by A^i. We use $A_{i;\langle j \rangle}$ to denote the sequence obtained by taking the first j entries of the row A_i and use $A^{i;\langle j \rangle}$ to denote the sequence obtained by taking the first j entries of the column A^i. Given $n, p > 0$, we set

$$\boldsymbol{B}(n,p) = \Big\{ A \in \boldsymbol{A}_n : \mathrm{wt}(A_i) \leqslant pn \text{ and } \mathrm{wt}(A^i) \leqslant pn \text{ for } 1 \leqslant i \leqslant n \Big\}.$$

In this work, we are interested in the problem of designing efficient coding methods that encode (decode) binary data to (from) $\boldsymbol{B}(n,p)$. Particularly, we provide two linear-time encoders.

II. Encoder I for $\boldsymbol{B}(n,p)$ when $p > 1/2$

Encoder I adapts the sequence replacement technique and the result in [8] (see Theorem 1), with the antipodal matching constructed in [3] (see Definition 1) to encode arbitrary data to $\boldsymbol{B}(n,p)$ when $p > 1/2$ with at most $n+3$ redundant bits.

Theorem 1 (Nguyen *et al.* [8])**.** *Given p_1, p_2 where $0 \leqslant p_1 < 1/2 < p_2 \leqslant 1$, let $c = \min\{1/2 - p_1, p_2 - 1/2\}$. For $(1/c^2)\log_e m \leqslant \ell \leqslant m$, there exists linear-time algorithm $\mathrm{ENC_{seq}} : \{0,1\}^{m-1} \to \{0,1\}^m$ such that for all $\boldsymbol{x} \in \{0,1\}^{m-1}$ if $\boldsymbol{y} = \mathrm{ENC_{seq}}(\boldsymbol{x})$ then $\mathrm{wt}(\boldsymbol{y}) \in [p_1 m, p_2 m]$) and for every window $\boldsymbol{w}$ of size ℓ of $\boldsymbol{y}$, $\mathrm{wt}(\boldsymbol{w}) \in [p_1\ell, p_2\ell]$.*

Definition 1 (Ordentlich and Roth [3])**.** An antipodal matching ϕ is a mapping from $\{0,1\}^n$ to itself with the following properties holding for every $\boldsymbol{x} \in \{0,1\}^n$:

1) $\mathrm{wt}(\phi(\boldsymbol{x})) = n - \mathrm{w}(\boldsymbol{x})$.
2) If $\mathrm{wt}(\boldsymbol{x}) > n/2$ then $\phi(\boldsymbol{x})$ has all its 1's in positions where $\boldsymbol{x}$ has 1's.
3) $\phi(\phi(\boldsymbol{x})) = \boldsymbol{x}$.

Set $m = n^2 - (n+3)$, $\ell = n$, $p_1 = 0$, $p_2 = p$ and $c = p - 1/2$. According to Theorem 1, for sufficient n that $(1/c^2)\log_e(n^2 - n - 3) \leqslant n \leqslant n^2 - n - 3$, there exists linear-time encoder, $\mathrm{ENC_{seq}} : \{0,1\}^m \to \{0,1\}^{m+1}$ such that for all $\boldsymbol{x} \in \{0,1\}^m$ and $\boldsymbol{y} = \mathrm{ENC_{seq}}(\boldsymbol{x})$ we have $\mathrm{wt}(\boldsymbol{w}) \in [0, pn]$ for every window $\boldsymbol{w}$ of size n of $\boldsymbol{y}$. In addition, we follow [3] to construct the antipodal matchings ϕ for sequences of length $n-1$.

p-bounded Encoder I, $\text{ENC}^{\text{I}}_{\boldsymbol{B}(n,p)}$.

INPUT: $\boldsymbol{x} \in \{0,1\}^m$
OUTPUT: $A \triangleq \text{ENC}^{\text{I}}_{\boldsymbol{B}(n,p)}(\boldsymbol{x}) \in \text{B}(n,p)$ with $p > 1/2$

(I) Set $\boldsymbol{y} = \text{ENC}_{\text{seq}}(\boldsymbol{x}) \in \{0,1\}^{m+1}$. Suppose that $\boldsymbol{y} = y_1y_2\dots y_{n^2-n-2}$.
(II) Fill n^2-n-1 bits of $\boldsymbol{y}$ to A row by row as follows.
- Set $A_i \triangleq y_{n(i-1)+1}\dots y_{ni}$ for $1 \leqslant i \leqslant n-2$.
- Set $A_{n-1} \triangleq y_{n(n-2)+1}\dots y_{n^2-n-2} *_1 *_2$
- Suppose that $A_n = z_1z_2\dots z_n$
- If $\text{wt}\Big(A_{n-1;\langle n-2\rangle}\Big) > p(n-2)$, flip all bits in $A_{n-1;\langle n-2\rangle}$ and set $*_1 = 1$, otherwise set $*_1 = 0$.

(III) For $1 \leqslant i \leqslant (n-1)$, we check the ith column:
- if $\text{wt}\Big(A^{i;\langle n-1\rangle}\Big) > pn$, set $z_i = 1$ and replace $A^{i;\langle n-1\rangle}$ with $\phi\Big(A^{i;\langle n-1\rangle}\Big)$
- Otherwise, set $z_i = 0$.

(IV) Check the nth row:
- If $\text{wt}\Big(A_{n;\langle n-1\rangle}\Big) > pn$, set $*_2 = 1$ and replace $A_{n;\langle n-1\rangle}$ with $\phi\Big(A_{n;\langle n-1\rangle}\Big)$
- Otherwise, set $*_2 = 0$.

(V) Check the nth column:
- If $\text{wt}\Big(A^{n;\langle n-1\rangle}\Big) > pn$, set $z_n = 1$ and replace $A^{n;\langle n-1\rangle}$ with $\phi\Big(A^{n,\langle n-1\rangle}\Big)$.
- Otherwise, set $z_n = 0$.

(VI) Output A.

Theorem 2. *The Encoder $\text{ENC}^{\text{I}}_{\boldsymbol{B}(n,p)}$ is correct. In other words, $\text{ENC}^{\text{I}}_{\boldsymbol{B}(n,p)}(\boldsymbol{x}) \in \text{B}(n,p)$ with $p > 1/2$ for all $\boldsymbol{x} \in \{0,1\}^m$. The redundancy is $n+3$ (bits).*

III. ENCODER II FOR $\boldsymbol{B}(n,p)$ WHEN $p < 1/2$

A key ingredient of our method is the swapping function.

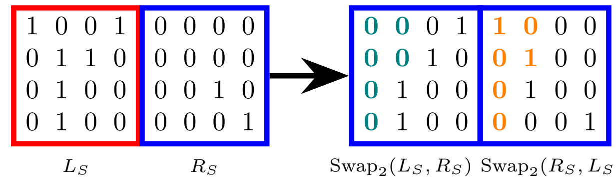
Definition 2. Given $\boldsymbol{y} = y_1y_2\dots y_m, \boldsymbol{z} = z_1z_2\dots z_m$. For $1 \leqslant t \leqslant m$, we use $\text{Swap}_t(\boldsymbol{y},\boldsymbol{z})$, $\text{Swap}_t(\boldsymbol{z},\boldsymbol{y})$ to denote the sequences obtained by swapping the first t bits of $\boldsymbol{y}$ and $\boldsymbol{z}$, i.e. $\text{Swap}_t(\boldsymbol{y},\boldsymbol{z}) = z_1z_2\dots z_ty_{t+1}y_{t+2}\dots y_m$ and $\text{Swap}_t(\boldsymbol{z},\boldsymbol{y}) = y_1y_2\dots y_tz_{t+1}z_{t+2}\dots z_m$.

Lemma 1 (Swapping Lemma)**.** *Given $\boldsymbol{x} = \boldsymbol{yz} \in \{0,1\}^{2m}$, $\boldsymbol{y} = y_1y_2\dots y_m, \boldsymbol{z} = z_1z_2\dots z_m$. If $\boldsymbol{x}$ is p-bounded then there exists an index t, referred to as a swapping index of $\boldsymbol{x}$ and $\boldsymbol{y}$, such that both $\text{Swap}_t(\boldsymbol{y},\boldsymbol{z})$ and $\text{Swap}_t(\boldsymbol{z},\boldsymbol{y})$ are p-bounded.*

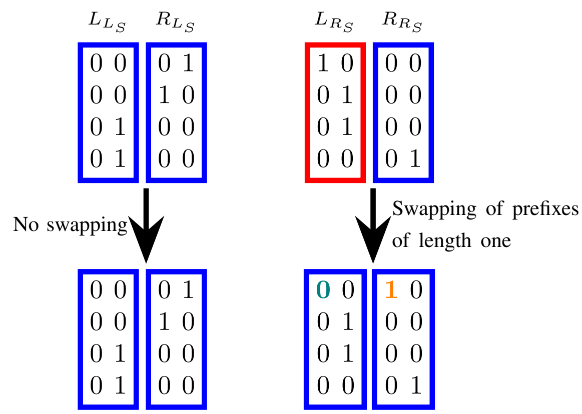
We now describe briefly Encoder II.
- In phase 1, the encoder encodes the information of length N into array S of size $m\times n$ where every row is p-bounded.
- In phase 2, the encoder follows the swapping procedure (illustrated through example in Figure 1) to ensure that every column of S is p-bounded.
- In phase 3, it encodes all *swapping indices* into an array S' of size $(n-m)\times n$ such that its every row and every column is p-bounded. The encoder outputs the concatenation of S and S', which is an array of size $n \times n$.

See [7] for more details, we show that $m = n - \Theta(\log n)$ and the redundancy of our encoder is at most $n\mu(n,p) + O(n\log n)$ redundant bits.

(a) We divide S into two subarrays L_S and R_S. In this step, only L_S is not p-bounded. To obtain two p-bounded subarrays, we swap their prefixes of length six to obtain $\text{Swap}_2(L_S,R_S)$ and $\text{Swap}_2(R_S,L_S)$, respectively.



(b) We divide L_S into two subarrays L_{L_S} and R_{L_S}, and divide R_S into two subarrays L_{R_S} and R_{R_S}, we observe that only L_{R_S} is not p-bounded. We proceed as before and swap their prefixes of length one.



(c) Finally, we divide each of the four subarrays into two, resulting in the $n = 8$ columns. The final output is then:

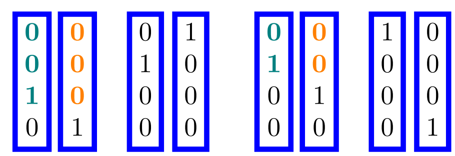


Fig. 1: Example for $n = 8, p = 1/4$. The current subarray S is of size 8×4. The subarrays, highlighted in red, are not p-bounded while those, highlighted in blue, are p-bounded.

REFERENCES

[1] A. Chen, “Accessibility of nano-crossbar arrays of resistive switching devices,” *Proc. 11th IEEE Int. Conf. Nanotechnol.*, pp. 1767-1771, 2011.
[2] R. Ben Hur and S. Kvatinsky, “Memory processing unit for in-memory processing,” in *Proc. IEEE ACM Int. Symp. Nanosc. Archit.*, Jul. 2016.
[3] E. Ordentlich and R. M. Roth, “Low complexity two-dimensional weight-constrained codes,” *2011 IEEE International Symposium on Information Theory Proceedings*, St. Petersburg, 2011, pp. 149-153.
[4] Y. Cassuto, S. Kvatinsky and E. Yaakobi, “Information-Theoretic Sneak-Path Mitigation in Memristor Crossbar Arrays,” in *IEEE Transactions on Information Theory*, vol. 62, no. 9, pp. 4801-4813, Sep. 2016.
[5] E. Ordentlich and R. M. Roth, “Two-dimensional weight-constrained codes through enumeration bounds,” in *IEEE Transactions on Information Theory*, vol. 46, no. 4, pp. 1292-1301, Jul. 2000, doi: 10.1109/18.850669.
[6] R. Talyansky, T. Etzion, and R. M. Roth, “Efficient Code Constructions for Certain Two-Dimensional Constraints”, *IEEE Transactions on Information Theory*, vol. 45, no. 2, pp. 794-799, Mar. 1999.
[7] T. T. Nguyen, K. Cai, K. A. S. Immink and Y. M. Chee, “Efficient Design of Capacity-Approaching Two-Dimensional Weight-Constrained Codes,” *2021 IEEE International Symposium on Information Theory (ISIT)*, 2021.
[8] T. T. Nguyen, K. Cai and K. A. Schouhamer Immink, “Efficient Design of Subblock Energy-Constrained Codes and Sliding Window-Constrained Codes,” in *IEEE Transactions on Information Theory*, **67** (12), Dec. 2021.