2009 International Symposium on Nonlinear Theory and its Applications
NOLTA'09, Sapporo, Japan, October 18-21, 2009

An Algorithm For Generating All Full-Length Sequences Which Are Based On Discretized Markov Transformations

Hiroshi Fujisaki

Graduate School of Natural Science and Technology
Kanazawa University
Kakuma-machi, Kanazawa, Ishikawa, 920-1192 Japan
Email: fujisaki@t.kanazawa-u.ac.jp

Abstract—We consider discretized Markov transformations and give an algorithm, called the bounded monotone truth-table algorithm, for generating *all* full-length sequences which are based on the discretized transformations. The algorithm is efficient in the sense that it guarantees to generate all full-length sequences without computing their total number.

1. Introduction

Full-length sequences or maximal-period sequences have found major applications in a wide variety of systems such as cryptography and digital communication systems. To generate full-length sequences, a LFSR (linear feedback shift register) is commonly used. On the other hand, in view of randomness in chaotic dynamics of one-dimensional ergodic transformations, sequences based on discretized Bernoulli transformations were proposed in [1] and [2]. The latter sequences have a great advantage in terms of their family size. For instance, for binary sequences of length 2^n, while the total number of the former sequences is much less than $2^n/n$, the total number of the celebrated de Bruijn sequences is known to be $2^{2^{n-1}-n}$.

In previous research [3], we generally defined discretized Markov transformations and found an algorithm to give the total number of full-length sequences based on discretized Markov transformations. The discretized Markov transformations, which can be regarded as examples of *ultradiscrete* dynamical systems [4], are permutations of subintervals in Markov partitions determined from the transformations. From this viewpoint, de Bruijn sequences are merely special examples of full-length sequences in the discretized Markov transformations. In fact, they are full-length sequences based on a subclass of the discretized dyadic transformations.

Given a Markov transformation, we obtain countably many discretized Markov transformations as defined in [3]. If we fix a discretized Markov transformation, we can compute the total number of full-length sequences based on the discretized transformations from the algorithm found in [3].

The instant we know its total number, it is natural to consider a problem to give an algorithm for generating *all* the full-length sequences. This question is not only interesting from the mathematical viewpoint but also important in a practical sense. As stated above, full-length sequences principally admit a broad variety of applications. However, it is quite a difficult but challenging problem. Even for de Bruijn sequences, only a few algorithms are known for generating all de Bruijn sequences, while a number of results have contributed to generations of a single sequence or a small fraction of the sequences [5]–[6].

Under the assumption that full-length sequences were already given, by transpositions of the sequences, an algorithm was proposed in [7] for generating relatively many sequences based on the discretized r-adic transformations which included de Bruijn sequences for $r = 2$. It is an interesting approach which generates comparably many, but not all, full-length sequences.

In this report we tackle the problem to give an algorithm for generating all the full-length sequences. To this end, we consider general discretized Markov transformations and give a novel algorithm, called the bounded monotone truth-table algorithm, for generating *all* full-length sequences which are based on the discretized transformations. The algorithm is efficient in the sense that it guarantees to generate all full-length sequences without computing their total number. The algorithm proposed here is of course applicable to generation of all de Bruijn sequences.

2. Previous Results On Generating De Bruijn Sequences

To avoid redundancies, we freely use the technical terms defined in [3] throughout this report. If the reader encounters a jargon, please consult [3].

In [3] we defined the discretized Markov transformations and found an algorithm to give the total number of full-length sequences which are based on the discretized Markov transformations as follows.

For an irreducible aperiodic Markov transformation T, given a Markov partition $\mathcal{P}$ with respect to T, corresponding each subinterval $I \in \mathcal{P}$ to one arc $a(I)$, we obtain the set $\mathcal{A}$ of arcs. For each ordered pair (I, J) of elements of $\mathcal{P}$, one vertex $v(I, J)$ adjacent from $a(I)$ and to $a(J)$ is allowed exactly when $J \subset T|_I(I)$. Thus we obtain the directed graph $G = (\mathcal{V}, \mathcal{A})$ representing the Markov transformation. Generally, this is not Eulerian.

Let $H = (\mathcal{W}, \mathcal{B})$ be the Eulerian subgraph spanning G with maximal number of arcs.

Under the above-mentioned one-to-one correspondence between $\mathcal{P}$ and $\mathcal{A}$, we obtain the partition $\mathcal{Q}$ which corresponds to $\mathcal{B}$. Then the discretized Markov transformation $\widehat{T}$ is defined by a permutation $\widehat{T} : \mathcal{Q} \to \mathcal{Q}$ with $\widehat{T}(I) \subset T|_I(I)$ for all $I \in \mathcal{Q}$. Eventually, the number of full-length sequences in the discretized Markov transformation is given by the cofactor of C_{11} in the matrix of admittance C of H.

In view of this definition, the celebrated de Bruijn sequences are merely special examples of full-length sequences from the discretized Markov transformations as stated in the Introduction.

Now we review the algorithms for generating all de Bruijn sequences.

For each positive integer n, there are exactly $2^{2^{n-1}-n}$ de Bruijn sequences of length 2^n [8]-[9]. The most important part of de Bruijn's proof for this fact lies in the recognition of a relation between the graphs G_n and G_{n+1}:

$$G_{n+1} = G_n^*, \tag{1}$$

where $G_n = (\mathcal{V}_n, \mathcal{A}_n)$ $(n > 1)$ is the de Bruijn graph with $\mathcal{V}_n = \{0, 1\}^{n-1}$ and $\mathcal{A}_n = \{0, 1\}^n$. An arc $a_1 a_2 \cdots a_n \in \mathcal{A}$ goes from $a_1 a_2 \cdots a_{n-1}$ to $a_2 a_3 \cdots a_n$. G_n^* is the arc digraph of G_n.

Using the relation (1), an algorithm for generating all de Bruijn sequences was stated in [6]. To generate all de Bruijn sequences of length 2^n, it always requires $2^{2^{n-2}}$-bit *initial* memory, which is rather costly in terms of the amount of storage required.

Unfortunately, however, this algorithm cannot apply to generating all full-length sequences from general discretized Markov transformations since relations like (1) do not hold in general. In fact, for discretized golden mean transformations, (1) is violated as shown in [3].

To state the known algorithm not using (1) for generating all de Bruijn sequences, following [5], we introduce

Definition 1 *A preference function p is a $k(\geq 2)$-dimensional vector-valued function of $n-1$ $(n \geq 2)$ variables such that, for each choice of $a_1, a_2, \cdots, a_{n-1}$ from the k-alphabet $\{0, 1, \cdots, k-1\}$, $(p_1(a_1, \cdots, a_{n-1}), \cdots, p_t(a_1, \cdots, a_{n-1}))$ is a rearrangement of $0, \cdots, k-1$.*

In particular, for the binary alphabet, if $p_1(a_1, \cdots, a_{n-1}) \equiv 1$, p is called a prefer-one function [6].

Using the prefer-one function in conjunction with backtracking, an algorithm for generating all de Bruijn sequences was introduced in [6]. Accordingly, to generate a single de Bruijn sequence of length 2^n, it always requires a linear order of $n2^n$-bit memory and requires operations of order $O(2^n)$ at least, where O denotes Landau's symbol. Thus it totally needs operations of order $O(2^{2^{n-1}})$ for generating all de Bruijn sequences. Actually, it was pointed out in [6] that the algorithms using the prefer-one function were expensive in terms of the amount of storage required.

Again this algorithm cannot apply to generating all full-length sequences from general discretized Markov transformations since the preference functions are not always well defined. Actually, for discretized golden mean transformations, we always have $(p_1(a_1, \cdots, a_{n-2}, 1), p_2(a_1, \cdots, a_{n-2}, 1)) = (0, 0)$, which is not a rearrangement of $0, 1$.

So far, to the best of the author's knowledge, only a few algorithms for generating all de Bruijn sequences are known, and they strictly depend on the properties of de Bruijn sequences. Therefore we cannot directly apply these algorithms to generating all full-length sequences from general discretized Markov transformations. Thus the algorithm proposed in this research is novel as well as not trivial.

3. The Bounded Monotone Truth-Table Algorithm

To simplify discussions, we restrict the $k(\geq 2)$-alphabet to the binary alphabet $\{0, 1\}$. This simplification does not essentially change any mathematical treatments for the case $k > 2$ except $k!$ cases to be counted.

To generate all full-length sequences based on a given discretized Markov transformation, in view of results in [3] which are briefly reviewed in the beginning of Section 2, we can begin with the Eulerian subgraph $H = (\mathcal{W}, \mathcal{B})$ spanning G with maximal number of arcs.

Since H is Eulerian, it is connected and every vertex has an even degree. Thus, with the binary alphabet, the degree of each vertex is 2 or 4.

To proceed to the main steps, we contract H into a regular graph in advance as follows.

Step 0a) If a vertex $v \in \mathcal{W}$ has the degree of 2, it has a unique pair of vertices $u, w \in \mathcal{W}$ such that u is adjacent to v and w is adjacent from v. Since we are interested in only full-length sequences, we can regard the pair of arcs (uv, vw) as a single arc. In other words, we can shorten the vertex v from H without changing any Eulerian paths in H.

If all the vertices have the degree of 4, go to 0b). Otherwise, go to 0a).

We refer the resulting graph $H' = (\mathcal{W}', \mathcal{B}')$ to the *contracted regular graph* from H.

Let $\mathcal{W}' = \{w'_1, w'_2, \cdots, w'_{\sharp\mathcal{W}'}\}$. For a set A, we use $\sharp A$ to denote the cardinality of A. Each vertex can be represented by a binary word as $w'_i = a_1^{(i)} a_2^{(i)} \cdots a_{|w'_i|}^{(i)} \in \{0,1\}^{|w'_i|}$, $1 \le i \le \sharp\mathcal{W}'$. For a word w, we use $|w|$ to denote the length of w.

Now in H', two arcs $0w'_i$ and $1w'_i$ terminate at w'_i, while w'_i0 and w'_i1 start at w'_i. If a path admits $0w'_i0$, which is equivalent to that it admits $1w'_i1$ (i.e. the path admits $0w'_i0 \cdots 1w'_i1$) or that other path admits $1w'_i1$ (and the path does not admit $1w'_i1$), we set $t_i = 0$. On the other hand, if a path admits $0w'_i1$, which is equivalent to that it admits $1w'_i0$ (i.e. the path admits $0w'_i1 \cdots 1w'_i0$) or that other path admits $1w'_i0$ (and the path does not admit $1w'_i0$), we set $t_i = 1$. Thus we obtain a $\sharp\mathcal{W}'$-dimensional vector $t = (t_1, t_2, \cdots, t_{\sharp\mathcal{W}'}) \in \{0,1\}^{\sharp\mathcal{W}'}$. We call it the *truth table* of H'.

Example 1 *The truth table of de Bruijn graph* $G_n = (\{0,1\}^{n-1}, \{0,1\}^n)$ $(n > 1)$ *is given by* $(a_1, a_2, \cdots, a_{n-1}) \in \{0,1\}^{n-1}$.

The truth table t of H' defines a permutation $\sigma_t : \mathcal{B}' \to \mathcal{B}'$ by

$$\sigma_t = \begin{pmatrix} 0w'_1 & \cdots & 0w'_{\sharp\mathcal{W}'} & 1w'_1 & \cdots & 1w'_{\sharp\mathcal{W}'} \\ w'_it_1 & \cdots & w'_it_{\sharp\mathcal{W}'} & w'_i\overline{t_1} & \cdots & w'_i\overline{t_{\sharp\mathcal{W}'}} \end{pmatrix},$$

which represents a discretized Markov transformation. For $a \in \{0,1\}$, we use $\overline{a}$ to denote the binary complement of a, i.e. $\overline{0} = 1$ and $\overline{1} = 0$.

Next, we introduce a metric d in the set of discretized Markov transformations by $d(\sigma_t, \sigma_{t'}) = d_H(t, t')$ for $t, t' \in \{0,1\}^{\sharp\mathcal{W}'}$, where d_H is the Hamming distance of t and t', i.e. the number of components where t and t' differ.

The following lemma navigates the next steps through the proposed algorithm:

Lemma 1 *Let σ_t be a discretized Markov transformation generating a full-length sequence, which is equivalent to σ_t itself being a full cycle. Then any discretized Markov transformation $\sigma_{t'}$ with $d(\sigma_t, \sigma_{t'}) = 1$ cannot be a full cycle.*

Before taking the main steps, we need another prerequisite:

Step 0b) For $i = 1, 2, \cdots, \sharp\mathcal{W}'$, if a vertex has a form of $w'_i = 0a_2^{(i)} \cdots a_{|w'_i|-1}^{(i)}0$, set $t_i = 1$, or if a vertex has a form of $w'_i = 1a_2^{(i)} \cdots a_{|w'_i|-1}^{(i)}1$, set $t_i = 0$, which preliminarily prevents cycles of single arcs. Since the original transformation is irreducible and aperiodic, either takes place at least once, but both occur at most once. Hereafter we fix such t_i. After we exclude all such fixed t_i from t and renumber the coordinates of the rest components in t, we obtain a $\sharp\mathcal{W}' - 1$- or $\sharp\mathcal{W}' - 2$-dimensional vector and denote it by $\tilde{t} = (\tilde{t}_1, \cdots, \tilde{t}_W)$. We refer to $\tilde{t}$ as the contracted truth table. By definition, we obtain a one-to-one correspondence $\gamma : t \mapsto \tilde{t}$ and $\gamma^{-1} : \tilde{t} \mapsto t$.

The following subroutine plays an important role in the proposed algorithm.

Step B1) Let m $(1 < m \le 2\sharp\mathcal{W}')$ be the least period with

$$\sigma_t^m(0w'_1) = 0w'_1.$$

For each i $(1 \le i \le \sharp\mathcal{W}')$, if there exists n $(1 \le n \le m \le 2\sharp\mathcal{W}')$ such that $\sigma_t^n(0w'_1) = 0w'_i$, then set $r_i = 1$. Otherwise set $r_i = 0$. Thus we obtain $r = (r_1, \cdots, r_{\sharp\mathcal{W}'})$. If $r_i = 1$ for all i $(1 \le i \le \sharp\mathcal{W}')$, then we obtain a full cycle σ_t and return. Otherwise go to B2).

Step B2) Set $\tilde{r} = \gamma(r)$ and $\tilde{r} = (\tilde{r}_1, \cdots, \tilde{r}_W)$. Let $R_0 = \max\{i : \tilde{r}_i = 0,\ 1 \le i \le W\}$ and $R_1 = \max\{i : \tilde{r}_i = 1,\ 1 \le i \le W\}$. If there exists n $(1 \le n \le m \le 2\sharp\mathcal{W}')$ such that $\sigma_t^n(0w'_1) = 1w'_{\sharp\mathcal{W}'}$, then set $t = \gamma^{-1}(\tilde{t}_1, \cdots, \tilde{t}_{R_0-1}, \overline{\tilde{t}_{R_0}}, \tilde{t}_{R_0+1}, \cdots, \tilde{t}_W)$ and go to B1). Otherwise set $t = \gamma^{-1}(\tilde{t}_1, \cdots, \tilde{t}_{R_1-1}, \overline{\tilde{t}_{R_1}}, \tilde{t}_{R_1+1}, \cdots, \tilde{t}_W)$ and go to B1).

In what follows, we regard $\tilde{t}$ as the positional notation using base 2 for a nonnegative integer. The following lemma is the essential ingredient in the proposed algorithm:

Lemma 2 *Let $\widetilde{\mathcal{T}} = \{\tilde{t} \in \{0,1\}^W : \sigma_t = \sigma_{\gamma^{-1}(\tilde{t})}$ is a full cycle$\}$. If we input t with $\tilde{t} = \underbrace{(0, \cdots, 0)}_{W}$ into B1), then we obtain the full cycle σ_t whose contracted truth table $\tilde{t}$ is the lower bound of $\widetilde{\mathcal{T}}$. On the other hand, if we input t with $\tilde{t} = \underbrace{(1, \cdots, 1)}_{W}$ into B1), then we obtain the full cycle σ_t whose contracted truth table $\tilde{t}$ is the upper bound of $\widetilde{\mathcal{T}}$.*

As stated in Section 2, we know for each positive integer n, there exists $2^{2^{n-1}-n}$ de Bruijn sequences of length 2^n. Thus, for generating all de Bruijn sequences, by checking the number of distinct generated sequences, one is immediately given a condition when the algorithm stops. On the contrary, for generating all full-length sequences which are based on the discretized Markov transformations, it is not so simple. If we want to know the total number of the full-length sequences, we need to compute the cofactor of C_{11} in the matrix of admittance C of H'.

For a $N \times N$ matrix, it is known that the asymptotic complexity is $O((N-1)^{2.376})$ to compute the cofactor [10]. Note that $N = \sharp\mathcal{B}'$ has an exponential order with base 2 for discretized Markov transformations.

Lemma 2 is of great use since it guarantees that the proposed algorithm stops without calculating the total number of the full-length sequences.

Now we are in the right position to introduce the main steps.

Step 1)(Computation of the upper bound) Set t with $\tilde{t} = (\underbrace{1, \cdots, 1}_{W})$ and go to B1). After returning from B1), output σ_t and set $v = (v_1, \cdots, v_W) = \tilde{t}$. Go to 2).

Step 2) Set t with $\tilde{t} = (\underbrace{0, \cdots, 0}_{W})$ and go to B1). After returning from B1), output σ_t and set $\tilde{t}^{(1)} = (\tilde{t}_1^{(1)}, \cdots, \tilde{t}_W^{(1)}) = \tilde{t}$. Set $i = 1$. and go to 3).

Step 3) If $\tilde{t}_W^{(i)} = 0$, then set $\tilde{t} = \tilde{t}^{(i)} + 11$. If $\tilde{t}_W^{(i)} = 1$, then set $\tilde{t} = \tilde{t}^{(i)} + 1$. The addition is performed in the binary number system regarding $\tilde{t}$ and $\tilde{t}^{(i)}$ as the positional notations using base 2 for nonnegative integers. Go to B1). After returning from B1) go to 4).

Step 4) If $\tilde{t} < v$, then output σ_t. Set $\tilde{t}^{(i+1)} = (\tilde{t}_1^{(i+1)}, \cdots, \tilde{t}_W^{(i+1)}) = \tilde{t}$. Set $i = i + 1$ and go to 3). If $\tilde{t} = v$, then stop.

Because of the following remark, we refer to the proposed algorithm as the bounded monotone truth-table algorithm.

Remark 1 *The resulting sequence of contracted truth tables is bounded and monotone increasing:*

$$\tilde{t}^{(1)} < \tilde{t}^{(2)} < \cdots \leq v.$$

4. Application To Generating All De Bruijn Sequences

To see that the proposed algorithm works well, we apply it to generations of all $2^{2^{n-1}-n}$ de Bruijn sequences of length 2^n.

For the case $n = 4$, we have all 16 de Bruijn sequences of length 2^4. To avoid cycles of single arc preliminarily, we define its truth table by $(1, t_2, t_3, \cdots, t_7, 0) \in \{0, 1\}^8$. Thus we write $\tilde{t} = (\tilde{t}_1, \cdots, \tilde{t}_6)$. We regard a binary word $\tilde{t}_1 \cdots \tilde{t}_6$ as the binary vector $(\tilde{t}_1, \cdots, \tilde{t}_6)$ in the following.

Using the subroutine B1)–B2) with the vector $\underbrace{1 \cdots 1}_{6}$, we obtain the upper bound $v = 111110$. Similarly, with the initial vector $\underbrace{0 \cdots 0}_{6}$, we obtain the lower bound $\tilde{t}^{(1)} = 000111$. Then the proposed algorithm generates a bounded and monotone increasing sequence of contracted truth tables as follows:

$$\begin{aligned} \tilde{t}^{(1)} &= 000111 < 001110 < 010011 < 010110 \\ &< 010101 < 011010 < 011100 < 011111 \\ &< 100011 < 101010 < 110001 < 110010 \\ &< 110111 < 111000 < 111011 < 111110 = v, \end{aligned}$$

which demonstrates all distinct 16 de Bruijn sequences.

5. Summary

We considered discretized Markov transformations and gave a novel algorithm, called the bounded monotone truth-table algorithm, for generating *all* full-length sequences which were based on the discretized transformations. The algorithm was efficient in the sense that it guaranteed the generation of all full-length sequences without computing their total number. It was not expensive as well in computing time and memory in the sense that it did not use the prefer-one function. We have also shown that the algorithm proposed here was applicable to generation of all de Bruijn sequences.

Acknowledgments

This study was mainly supported by the Grant-in-Aid for Scientific Research (C) under Grant No. 21560392 from the Japan Society for the Promotion of Science and in part by the Telecommunications Advancement Foundation. The author thanks Dr. Roger R. Anderson, retired Research Scientist in the Department of Physics and Astronomy, the University of Iowa, USA for his helping the author's English writings.

References

[1] N. Masuda and K. Aihara, "Chaotic cipher by finite-state baker's map", *Trans. of IEICE*, vol. 82-A, pp.1038–1046, 1999 (in Japanese).

[2] A. Tsuneda, Y. Kuga, and T. Inoue, "New Maximal-Period Sequences Using Extended Nonlinear Feedback Shift Registers Based on Chaotic Maps", *IEICE Trans. on Fundamentals*, vol. E85-A, pp.1327–1332, 2002.

[3] H. Fujisaki, "Discretized Markov Transformations – An Example of Ultradiscrete Dynamical Systems –," *IEICE Trans. Fundamentals*, vol.E88-A, pp.2684–2691, 2005.

[4] R. Hirota and D. Takahashi, *Discrete and Ultradiscrete Systems*, Kyoritsu Shuppan, 2003.

[5] S. W. Golomb, *Shift Register Sequences*, Revised Edition, Aegean Park Press, 1982.

[6] H. Fredricksen, "A Survey of Full Length Nonlinear Shift Register Cycle Algorithm," *SIAM Review*, vol.24, pp. 195–221, 1982.

[7] D. Yoshioka and A. Tsuneda, "On generation of pseudochaotic sequences obtained by discretized chaos maps," *Proc. of NOLTA 2007*, pp. 136–139, 2007.

[8] N. G. de Bruijn, "A Combinatorial Problem", *Nederl. Akad. Wetensch. Proc.*, vol. 49, pp.758–764, 1946.

[9] C. Flye Sainte-Marie, "Solution to problem number 58", *L'Intermediare des Mathematiciens,*, vol. 1, pp. 107–110, 1894.

[10] J. R. Bunch and J. Hopcroft, "Triangular factorization and inversion by fast matrix multiplication", *Mathematics of Computation*, vol. 28, pp. 231–236, 1974.