MORE DE BRUIJN SEQUENCES AS CONCATENATION OF LYNDON WORDS

ABBAS ALHAKIM
DEPARTMENT OF MATHEMATICS
AMERICAN UNIVERSITY OF BEIRUT
BEIRUT, LEBANON

Abstract. We consider a de Bruijn sequence dB over a finite alphabet that is constructed via a preference function P. We use P to introduce a total order on the set of all sequences and show that it lists the de Bruijn sequences of a given order so that dB is the minimal sequence. We also show that an appropriate bijective image of the binary, prefer-opposite de Bruijn sequence is uniquely factored as a concatenation of Lyndon words. This presents a second example to the well known prefer-one de Bruijn sequence, both in terms of minimality, and in terms of concatenation of Lyndon words. We also present other examples that suggest that the concatenation property is universal for all de Bruijn sequences.

1. Introduction

The lexicographically smallest de Bruijn sequence is by far the most studied of all de Bruijn sequences. One reason may be that it is generated by the well known prefer one greedy algorithm, first discovered by Martin [10], and rediscovered several times later, see Fredrisksen [7] (note that we consider that 1 is *less* than 0). Another method of generating this smallest sequence (say, of order n) is via concatenating all Lyndon words, of lengths that divide n, in increasing lexicographical order. Donald Knuth [9] calls this construction "almost magical". It is due to this construction that many authors claim that one of the many applications of Lyndon words is to construct de Bruijn sequences. In this paper we show that the relationship between Lyndon words and de Bruijn sequences extends much further than the prefer one sequence. The main tool to do this is a transform that encodes a sequence by a sequence with the same alphabet and that is defined via a preference function. Firstly, we establish a fundamental result that every de Bruijn sequence is minimal with respect to a lexicographical order defined by the preference function that creates this de Bruijn sequence. More specifically, given a preference function that produces a de Bruijn sequence, we encode every de Bruijn sequence by keeping track of the levels of preference taken all along the sequence. We then compare these *trail sequences* via lexicographical order. It is then the de Bruijn sequence generated by this preference function that receives the lexicographically smallest encoding. The 'minimality' of the prefer one sequence is thus revealed as a special case of this general result.

Furthermore, we study two relatives of the prefer one sequence, the prefer same and the prefer opposite sequences. These two sequences have, respectively, the lexicographically smallest and largest run length encoding, see [3] for definition and proof. These optimality results follow easily from our main result. More importantly, we show that their preference trails essentially consist of a concatenation of Lyndon words, when they are encoded with respect to their own preference functions. We conclude with a conjecture that every trail sequence of a de Bruijn sequence is *essentially* a concatenation of Lyndon words, laid in some order that depends on the underlying preference function.

The rest of the paper is organized as follows. In Section 2 we give basic definitions and background about preference functions and Lyndon words with preliminary lemmas that will be essential for the rest of the paper.

---

[1]This research was partially supported by the University Research Board (URB) of the American University of Beirut. Project Number 104107

Key words and phrases. De Bruijn sequence, Lyndon words, preference function, prefer-one sequence, prefer-opposite sequence, prefer-same sequence.

2. PRELIMINARIES

For an integer $n \geq 1$, $\mathcal{A}^n$ refers to the set of all strings of n bits, taken from an ordered alphabet $\mathcal{A}$ with q symbols. We will denote these symbols as $\{0, 1, \ldots, q-1\}$. These strings will be referred to as *n-words*, and denoted as $a_1 \cdots a_n$ and often as $(a_1, \ldots, a_n)$ for notational clarity. α^n denotes the word obtained by concatenating the word α n times.

For an integer $n \geq 1$, a de Bruijn sequence of order n, over the alphabet $\mathcal{A}$, is defined such that every string of n consecutive bits occurs exactly one time as a substring. For example, 0001011100 is a binary De Bruijn sequence of order 3, observing that all 3-strings occur in the order 000, 001, 010, 101, 011, 111, 110, 100. It is customary to consider cyclic rotations of a de Bruijn sequence as equivalent. With this equivalence class interpretation, we usually remove the last $n-1$ bits and wrap the remaining bits on a circle. The above sequence is represented as [00010111]. The only other binary sequence of order 3 is represented as [00011101]. While there are 16 sequences of order 4. In fact, for a general n, the number of non-rotationally equivalent de Bruijn sequences is $2^{2^{n-1}-n}$. The formula for nonbinary has an even higher rate of growth, it can be found in [7], together with a historical reference of the early development and applications of de Bruijn sequences. The first part in the next definition follows Golomb [8]. The *span* was defined in Alhakim [2].

Definition 2.1. For an integer $n \geq 1$, a *preference function* is a function P from $\mathcal{A}^n$ to S, where S is the set of all permutations of the elements of $\mathcal{A}$. We write $P(\mathbf{x}) = (P_0, \ldots, P_{q-1})$ for every n-word $\mathbf{x} = (x_1, \ldots, x_n)$; where the right hand side is an arrangement of $0, \ldots, q-1$. Furthermore, the *span* of P is the smallest integer s, $0 \leq s \leq n$, such that $P(x_1, \ldots, x_n)$ is fully determined by $(x_{n-s+1}, \ldots, x_n)$, for all n-words $(x_1, \ldots, x_n)$.

The following recursive construction produces a unique finite binary sequence $\{a_i\}$, provided that a preference function of span s and an arbitrary initial n-word $(I_1, \cdots, I_n)$ with $n > s$ are given. We denote the unique resulting sequence by (P, I).

1. For $i = 1, \cdots, n$ let $a_i = I_i$.
2. Suppose that $a_1, \cdots, a_k$ for some integer $k \geq n$ have been defined. Let $a_{k+1} = P_i(a_{k-s+1}, \ldots, a_k)$ where i, $0 \leq i \leq q-1$ is the smallest integer such that $(a_{k-n+2}, \ldots, a_{k+1})$ has not appeared in the sequence as a substring, if such an i exists.
3. If no such i exists, halt the program (the construction is complete.)

The following lemma is a slight generalization of Lemma 2 of Chapter 3 in Golomb [8]. The proof is essentially the same.

Lemma 2.2. *Consider an arbitrary preference function P of span $s \geq 0$ and initial word $I = (I_1, \cdots, I_n)$; $n > s$. Then every n-word occurs at most once in (P, I). Furthermore, (P, I) ends with the pattern $(I_1, \cdots, I_{n-1})$.*

It follows that the sequence (P, I) can be identified with a cyclic string, by removing the last pattern $(I_1, \ldots, I_{n-1})$ and wrapping the rest around a circle. A preference function P is said to be *complete* if there exists an initial word I such that (P, I) is a de Bruijn sequence.

Definition 2.3. For an integer i such that $0 \leq i < q$, the i^{th} column function induced by P is a function from $\mathbf{A}^s$ to $\mathbf{A}^s$ defined as

$$g_i(x_1, \ldots, x_s) = (x_2, \ldots, x_s, P_i(x_1, \ldots, x_s)).$$

Clearly, g_i defines at least one cycle of length $k \geq 1$. That is, a sequence of k s-words $v_1, \ldots, v_k$ in $\mathbf{A}^s$ such that $g_i(v_j) = v_{j+1}$ for $j = 1, \ldots, k-1$ and $g_i(v_k) = v_1$.

Theorem 3.1 and Corollary 3.2 in Alhakim [4] provide a characterization of complete preference functions, along with legitimate initial words I. We will refer to these initial words as de Bruijn seeds. Briefly, in a complete preference function, the column function g_{q-1} must have exactly one cycle. Also a de Bruijn seed $(I_1, \ldots, I_n)$ must be such that $(I_1, \ldots, I_{n-1})$ is a path on g_{q-1}. For example, the corresponding cycles of the complete preference functions of Table 1 are respectively $2 \to 2$, $0 \to 1 \to 0$, $0 \to 1 \to 2 \to 0$ and $00 \to 01 \to 10 \to 00$. The first cycle means that $2 \cdots 20 = 2^{n-1}0$, $2 \cdots 21 = 2^{n-1}1$ and 2^n are all

$0 \to 0, 1, 2$	$0 \to 2, 0, 1$	$0 \to 0, 2, 1$
$1 \to 0, 1, 2$	$1 \to 1, 2, 0$	$1 \to 0, 1, 2$
$2 \to 0, 1, 2$	$2 \to 0, 2, 1$	$2 \to 2, 1, 0$
$00 \to 0, 2, 1$	$10 \to 1, 2, 0$	$20 \to 1, 2, 0$
$01 \to 1, 2, 0$	$11 \to 0, 1, 2$	$21 \to 0, 2, 1$
$02 \to 2, 1, 0$	$12 \to 1, 2, 0$	$22 \to 0, 2, 1$

TABLE 1. Top: complete preference diagrams of span 0 (left), and span 1 (middle and right). Bottom: *one* complete preference diagram of span 2.

de Bruijn seeds of length $n > 1$. Likewise, 01,010, 0101 and 01010 are de Bruijn seeds of various lengths for the second preference function, while 01201201 and $001100110\cdots$ are examples of seeds for the last two preference functions. Proofs and more details are given in [4].

3. The minimality of a de Bruijn sequence

We begin this section with the following definition.

Definition 3.1. Let P be a preference function of span s and $n \geq s$. Then

(a) P defines an operator $T_P^{(n)}$ that acts on arbitrary sequence $S = d_1 \ldots d_l$ of length $l > s$ as $T_P^{(n)}(S) = d_1 \ldots d_n | c_1 \ldots c_{l-n}$, where for $i = 1, \ldots, l-n$ $d_{n+i} = P_{c_i}(d_{n+i-s} \ldots d_{n+i-1})$.

We refer to the the first n digits as the leading digits, and to the digits c_i as the preference trail digits, or simply the trail digits of S.

(b) We define the P-lex order, denoted $\prec_P$, as the total order on the set of sequences: for two sequences $S_1 = d_1 \ldots d_l$ and $S_2 = d'_1 \ldots d'_m$, $S_1 \prec_P S_2$ if and only if $c_1 \ldots c_{l-s}$ is lexicographically smaller than $c'_1 \ldots c'_{m-s}$, where $c_1 \ldots c_{l-s}$ and $c'_1 \ldots c'_{m-s}$ are resp. the trail sequences of S_1 and S_2 without the leading digits.

As an example, using the matrices in Table 1, the base 3 sequence 012210 is encoded using $n = s$ as $|012210$, $0|21122$, $0|22010$ and $01|1120$ respectively. Observe that the first preference function has no leading digits and it outputs the same input sequence. It is also evident that, for all preference functions, the initial sequence can be recovered uniquely by tracing the corresponding matrix, thanks to the leading digits. Another obvious but important observation is that the same sequence can have various lexicographical orders depending on the underlying function P.

In order to compare two de Bruijn sequences using the P-lex order, we will exclude the trail digits within the initial words and compare the trail of the $d_{n+1} \ldots d_{q^n}$.

Theorem 3.2. *Let P be a complete preference function of span s and $I = d_1 \ldots d_n$, $n > s$ be a de Bruijn seed such that $T_P^{(s)}(I) = d_1 \ldots d_s |(q-1)^{n-s}$. For an arbitrary de Bruijn sequence dB'_n that is not rotationally equivalent to dB_n we have $dB_n \prec_P dB'_n$ where $dB_n = (P, I)$.*

For convenience of notation, we will denote $T_P^{(n)}(dB'_n) = d'_1 \ldots d'_n | c'_1 \ldots c'_{q^n}$ for any de Bruijn sequence. That is, we apply $T_P^{(n)}$ to a version of dB'_n that begins and ends with $d'_1 \ldots d'_n$. In the case of dB_n, this amounts to appending the trail digits $(q-1)^s$ at the end of the sequence, which obviously has no effect on the P-lex order of dB_n.

Proof. Denote dB_n and dB'_n respectively by $d_1 \ldots d_{q^n}$ and $d'_1 \ldots d'_{q^n}$ and let

$T_P^{(n)}(dB_n) = d_1 \ldots d_n | c_1 \ldots c_{q^n}$ and $T_P^{(n)}(dB'_n) = d'_1 \ldots d'_n | c'_1 \ldots c'_{q^n}$.

We begin by establishing the inequality when $d'_1 \ldots d'_n = d_1 \ldots d_n$. Suppose, for a contradiction, that $dB'_n \prec_P dB_n$. Then there exists a minimal $i \geq 1$ such that $c'_i < c_i$. Since dB_n follows the preference strategy of P, the pattern $c_{i-n+s+1} \ldots c_{i-1} c'_i$ must have appeared earlier, preceded by the same s leading digits $d_{i-n+1} \ldots d_{i-n+s}$. By the minimality of i, all trail digits of dB_n and dB'_n are identical up to $i-1$. Thus by the assumption that $d'_1 \ldots d'_n = d_1 \ldots d_n = I$, dB'_n includes a repeated n-word, contradicting the fact that it is a de Bruijn sequence.

We will now tackle the case when $d'_1 \ldots d'_n \neq I$, that is, when dB' is rotated to start at any word other than I. We do this in two steps. First, consider the sequence $S = (P, d'_1 \ldots d'_n)$ whose trail sequence is

$d'_1 \dots d'_n | b_1 \dots b_l$, and which may or may not be a de Bruijn sequence. Since both S and dB'_n begin with the same initial word, the same argument as above establishes that $S \prec_P dB'_n$. Furthermore, since S follows the preference strategy of P, any digit placed after the terminal trail digit b_l leads to a repetition. So the first disagreement ($b_i < c'_i$) occurs at some $i < l$ or else dB'_n cannot be continued into a de Bruijn sequence.

Next we compare S and dB_n. Since $d'_1 \dots d'_n \neq I$, it is not of the form $d_1 \dots d_s | (q-1)^{n-s}$. Thus, one or more of the *wrap-around* words $d'_1 \dots d'_n, d'_2 \dots d'_{n+1}, \dots, d'_n \dots d'_{2n-1}$ is not a wrap-around word of dB_n, and therefore an internal word of the latter. Let j be the smallest index where a wrap-around word w of S is encountered for a second time upon proposing a trail digit $b_j = c$, it is avoided in S by either using higher preference $b_j > c$ if possible, or else S is stopped at b_{j-1} (i.e., $l = j-1$). In dB_n however, w is not a wrap-around word so that $c_j = c$.

Clearly, if $j < l$ then $dB_n \prec_P S$, and by the earlier proof above $S \prec_P dB'_n$, which shows that $dB_n \prec_P dB'_n$. If $j - 1 = l$ then $S \prec_P dB_n$, as $b_1 \dots b_l = c_1 \dots c_l$ and $l < q^n$. However, we proved above that the first disagreement between S and dB'_n occurs at $i < l$, implying that $c_i = b_i < c'_i$. This completes the proof. $\square$

All complete binary preference functions of span 1 are represented by the matrices

$$F = \begin{bmatrix} 1 & 0 \\ 1 & 0 \end{bmatrix}; O = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}; S = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}; \quad \text{and } Z = \begin{bmatrix} 0 & 1 \\ 0 & 1 \end{bmatrix}.$$

Observe that The Ford sequence, or prefer-one sequence is $(F, 0^n)$, first attributed to Martin [10], and $(Z, 1^n)$ is clearly its bitwise complement. $(O, 0^n)$ is the prefer-opposite sequence, see Alhakim [1], while $(S, 010\cdots)$ is the prefer-same sequence, where $010\cdots$ is the alternating string of length n. Alhakim *et. al.* [3] shows that the last two sequences respectively have the lexicographically smallest and largest representation in *run length encoding*. These results follow almost immediately from Theorem 3.2, the proofs are omitted for brevity.

4. Factoring Into Lyndon words

Recall that a Lyndon word is a finite word that is smaller than all of its rotations. For example, 0012 and 0021 are Lyndon words of size 4 but 0101 is not because it is equal to one of its rotations. Note that single symbols are Lyndon words. It is well known that the lexicographically least de Bruijn sequence is a concatenation of Lyndon words of lengths diving n and arranged in increasing lexicographical order. In this section we present a Lyndon decomposition of the trail sequence of the prefer-opposite sequence $\mathbf{o}_n = (O, I = 0^n)$, where the preference function O is given at the end of the previous section.

For a Lyndon word η, we define the *weight* $w(\eta)$ to be the number of *zeros* in η. The following theorem states that the preference trail of the prefer-opposite sequence is a concatenation of words that are essentially Lyndon words except for few exceptions, depending on the size n, that are well-defined rotations of Lyndon words. Let $\bar{n} = n - 1$ and $L(\bar{n})$ be the set of all Lyndon words with a length that divides $\bar{n}$. Recall that η^2 is a concatenation of two copies of the word η.

Theorem 4.1. *The trail sequence part of $T_O^{(n)}(\mathbf{o}_n)$ is a concatenation of all Lyndon words in $L(\bar{n})$, such that each word appears twice, starting with two consecutive 0, with the other words appended inductively as follows. Suppose η_0 has just been appended and let $\eta = \tau 01^j$ be the lexicographically next word in $L(\bar{n})$, where $\tau = c_1 \dots c_{\lambda-j-1}$ and λ is the length of η. Let $w = w(\eta) \mod 2$. Then*

(1) If either $w = j = 1$ or $w = 1$, $j = 2$ and $\tau = 0^{\lambda-j-1}$ then append η^2.

(2) If $w = 1$, $j > 2$ and $\tau = 0^{\lambda-j-1}$ then append $\eta \cdot \tilde{\eta}_1 \dots \tilde{\eta}_{j-2} \cdot \eta$ where $\tilde{\eta}_1 = 0^{\lambda-j}101^{j-2}$, $\tilde{\eta}_2 = 0^{\lambda-j}1101^{j-3}, \dots, \tilde{\eta}_{j-2} = 0^{\lambda-j}1^{j-2}01$.

(3) If $w = 1$, $j > 1$ and $\tau \neq 0^{\lambda-j-1}$ then if $\tilde{\eta}_1$ is a Lyndon word append $\eta \cdot \tilde{\eta}_1 \dots \tilde{\eta}_{j-1} \cdot \eta$ otherwise (if $\tilde{\eta}_1$ is not a Lyndon word) append η^2, where $\tilde{\eta}_1 = \tau 001^{j-1}$, $\tilde{\eta}_2 = \tau 0101^{j-2}, \dots, \tilde{\eta}_{j-1} = \tau 01^{j-2}01$.

(4) If $w = 0$ and $\tau \neq 0^{\lambda-j-1}$ then append η.

(5) If $w = 0$ and $\tau = 0^{\lambda-j-1}$ then append $\eta_1 \star \eta_2 \star \dots \star \eta_{j+1}$, where $\eta_1 = \eta = 0^{\lambda-j}1^j$, $\eta_2 = 0^{\lambda-j-1}101^{j-1}$, $\eta_3 = 0^{\lambda-j-1}1101^{j-2}, \dots, \eta_j = 0^{\lambda-j-1}1^{j-2}01$, $\eta_{j+1} = 0^{\lambda-j-1}1^j0$. The stars ($\star$) indicate segments of the sequence that contain the possible Lyndon words which are lexicographically ordered between η_i and η_{i+1}, arranged according to (1)-(4).

We will give a proof of this in the extended paper, due to the lack of space. We also omit a similar factorization theorem for the prefer-same sequence and only present some examples. We first list the factorization of the prefer-opposite sequence for orders 4 to 7. the Lyndon words are separated by one dot, blocks of types (1)-(4) are separated by an asterik ($*$), while words of type (5) are in bold. Note that there is a missing 1 relating to the missing word 1^n in $(O, 0^n)$.

$n = 4:\ 0000|0 \cdot 0 * \mathbf{001} \cdot \mathbf{010} * 011 \cdot 011 * 1$

$n = 5:\ 00000|0 \cdot 0 * 0001 \cdot 0001 * \mathbf{0011} \cdot 01 \cdot 01 * \mathbf{0110} * 0111 \cdot 0111 * 1$

$n = 6:\ 000000|0 \cdot 0 * \mathbf{00001} \cdot \mathbf{00010} * 00011 \cdot 00011 * 00101 \cdot 00101 * \mathbf{00111} * \mathbf{01011} * \mathbf{01101} * \mathbf{01110} * 01111 \cdot 01111 * 1$

$n = 7:\ 0000000|0 \cdot 0 * 000001 \cdot 000001 * \mathbf{000011} * \mathbf{000101} * \mathbf{000110} * 000111 \cdot 000101 \cdot 000111 * 001*$
$*001011 \cdot 001 \cdot 001011 * 001101 \cdot 001101 * \mathbf{001111} * 01 \cdot 01 * \mathbf{010111} * 011 \cdot 011 * \mathbf{011101} * \mathbf{011110} * 011111 \cdot 011111 * 1$

The following is a factorization of trail sequences of the prefer-same sequence with $n = 4$ and 5.

$n = 4:\ 0101|0 * 001 \cdot 0 \cdot 001 * 011 * 1 \cdot 011 * 1$

$n = 5:\ 01010|0 * 0001 \cdot 0 \cdot 0001 * 0011 * 01 \cdot 01 * 0111\dot{0}011 \cdot 0111 * 1$

Finally, letting P be the preference function of span 2 given in Table 1, which was arbitrarily chosen, we give a factorization of $(P, 0010)$. Note that Lyndon words of sizes 1 and 2, that divide $n - 2$ are each repeated $3^2 = 9$ times. Also note that there is only one *rotated* Lyndon word (underlined).

$n = 4, q = 3:\ 0010|0 \cdot 0 \cdot 0 \cdot 01 \cdot 0 \cdot 0 \cdot 02 \cdot 1 \cdot 0 \cdot 0 \cdot 0 \cdot 01 \cdot 01 \cdot \underline{10} \cdot 12 \cdot 01 \cdot 1 \cdot 01 \cdot 01 \cdot 02 \cdot 0 \cdot 01 \cdot 02 \cdot 02 \cdot 02 \cdot 02 \cdot 01 \cdot 02 \cdot 02 \cdot 1 \cdot 1 \cdot 1 \cdot 1 \cdot 1 \cdot 12 \cdot 12 \cdot 02 \cdot 2 \cdot 2 \cdot 12 \cdot 1 \cdot 12 \cdot 1 \cdot 12 \cdot 12 \cdot 2 \cdot 2 \cdot 2 \cdot 2 \cdot 12 \cdot 12 \cdot 2 \cdot 2 \cdot 2$

5. Discussion and Conclusion

We introduced a transform that maps a de Bruijn sequence (or any sequence) to a trail sequence using an arbitrary but fixed preference function. Observe that the prefer-zero sequence $(Z, 1^n)$ is identical to its trail sequence when the leading digits 1^n are not considered. It is a concatenation of Lyndon words that appear one time each. In this paper we have presented binary de Bruijn sequences of span 1 whose trail sequence is a concatenation of Lyndon words, appearing twice each. More numerical experimentation strongly suggest that the trail sequence of any q-ary de Bruijn sequence generated by a preference function of span s is equally a concatenation of Lyndon words that divide $n - s$, and each appearing q^s times, in a way that if η_1 is less than η_2 then the first appearance of η_1 occurs before the first appearance of η_2. This is a subject of further research.

References

[1] A. Alhakim. A Simple Combinatorial Algorithm for de Bruijn Sequences. The American Mathematical Monthly, **117**, Number 8, (2010) 728-732.

[2] A. Alhakim. Spans of preference functions. Discrete Applied Mathematics, Vol. 160, 7-8, (2012) 992-998.

[3] A. Alhakim, E. Sala, J. Sawada. Revisiting the prefer-same and prefer-opposite de Bruijn Sequence Constructions. Theoretical Computer Science, (2020).

[4] A. Alhakim, Designing preference functions for de Bruijn sequences with forbidden words. Des. Codes Cryptogr. 90, 2319-2335, (2022).

[5] C. Eldert, H. J. Gray, H. M. Gurk, M. Rubinoff. Shifting Counters. AIEE Trans., 77, (1958), 70-74.

[6] L. R. Ford, A Cyclic Arrangement of m-tuples, Report P-1071, Rand Corp., 1957.

[7] H. Fredricksen, A Survey of Full Length Nonlinear Shift Register Cycle Algorithms, *SIAM Review*, **24** (1982) 195-221.

[8] S. Golomb, *Shift Register Sequences*, Holden-Day, San Francisco, 1967.

[9] D. Knuth, *The Art of Computer Programming*, vol. 4, Addison Wesley, 2011.

[10] M. H. Martin, A Problem in Arrangements, *Bulletin of the American Mathematical Society*, **40** (1934) 859-864.

[11] Evan Sala, E., Sawada, J., Alhakim, A. Efficient constructions of the Prefer-same and Prefer-opposite de Bruijn sequences. Submitted Manuscript.

E-mail address: `aa145@aub.edu.lb`