A Cycle Joining Construction of the Prefer-Max De Bruijn Sequence

Gal Amram, Amir Rubin, Yotam Svoray, Gera Weiss

Department of Computer Science, Ben-Gurion University of the Negev

Abstract

This article introduces a novel approach for constructing the well-known Prefer-Max De Bruijn sequence using the cycle joining technique. This proposed construction provides an alternative confirmation of established results from previous studies. Firstly, the new construction offers an alternative proof for the seminal FKM-theorem. Secondly, it confirms the accuracy of recently discovered shift rules for Prefer-Max, Prefer-Min, and their reversals. Thirdly, it establishes the validity of the Onion theorem, which states that the reverse of the Prefer-Max De Bruijn sequence can be extended into an infinite De Bruijn sequence.

1. Introduction

For $n > 0$ and an alphabet $[k] = \{0, \ldots, k-1\}$ where $k > 0$, an (n, k)-De Bruijn sequence [12, 17] (abbreviated to (n, k)-DB sequence) is a total ordering of all words of length n over $[k]$ such that the successor of each word σw is a word of the form $w\tau$, where $w \in [k]^{n-1}$ and $\sigma, \tau \in [k]$. This applies also for the last and first words in the ordering, i.e., the first word must be a successor of the last one. DB sequences are used in various fields including cryptography [3, 4, 13, 39, 43], electrical engineering (mainly since they correspond to feedback-shift-registers) [7, 10, 11, 30, 31, 32, 36], molecular biology [33], and neuroscience [1].

The number of (n, k)-DB sequences (up to rotations) is $\frac{(k!^{k^{n-1}})}{k^n}$ [44]. Beyond this enumeration, many DB sequences were discovered for $k = 2$ (e.g. [2, 15, 16, 23, 28, 40]). However, for the non-binary case, the number of known constructions is smaller [5, 6, 14, 22, 41].

There are several standard construction techniques for DB sequences. First, a common technique uses De Bruijn graph (DB graph) [12, 26]: the directed

Email addresses: `galamra@cs.bgu.ac.il` (Gal Amram), `amirrub@cs.bgu.ac.il` (Amir Rubin), `svoray@math.utah.edu` (Yotam Svoray), `geraw@cs.bgu.ac.il` (Gera Weiss)

Preprint submitted to Elsevier *August 13, 2023*

graph over all n-length words with edges of the form $((\sigma w),(w\tau))$. DB sequences are the Hamiltonian cycles in the DB graph. Hence, generating Hamiltonian cycles in a DB graph constitutes a construction technique for DB sequences [4, 8, 33, 36, 43]. Second, some DB sequences were shown to be equal to a concatenation of certain words [14, 22]. Third, some DB sequences are constructed by a *shift rule*; a formula to produce the successor of a given word [5, 6, 14, 24, 25, 40, 41].

This paper employs a fourth common construction technique: a *cycle joining* (a.k.a. *cross-join*) construction [3, 4, 7, 10, 20, 25, 27, 29, 39, 46]. Roughly speaking, in this technique, the n-length words are divided into the equivalence classes of the 'rotation of' equivalence relation. Then, each equivalence class is ordered in a way that satisfies the successor property (hence, the classes are named cycles), and the cycles are "inserted" one into the other, so that the successor property holds for the entire obtained sequence.

One of the most well-known and studied DB sequences are the prefer-max sequences [18, 35] and its complement, the prefer-min sequences. These are the lexicographically maximal and minimal DB sequences. The prefer-max can be generated by the "granddaddy" greedy algorithm [35] that repeatedly chooses the lexicographic maximal legal successor that has not been added yet. An analogous process produces the prefer-min sequence.

Beyond this greedy approach, three constructions for the prefer-max sequence are known:

1. A concatenation of words of length n [22] (the FKM theorem).
2. A shift rule [5, 6, 25] (a shift rule for $k = 2$ is given in [19, 45]).
3. A cycle joining construction [25].

It is possible to prove (1) and conclude (2) and (3) from it: In [5, 25], the prefer-max shift rule was proved based on the correctness of the FKM theorem. This establishes (2) from (1). Furthermore, to get also (3), a shift-rule can be translated to a cycle joining construction in a straightforward manner. Specifically, to extract a cycle joining construction from a shift rule, one needs to identify the words whose successors/predecessors are not just rotations of themselves. This is how Gabric et al. [25] prove that a cycle joining construction generates the prefer-max sequence.

It is also possible to prove (2) and conclude (1) and (3) from it. In [6], a self-contained correctness proof for the prefer-max shift rule is given and it is shown that the FKM theorem follows from it. As explained above, from [6] we can also conclude a cycle joining construction.

This paper completes the "missing edge" of the triangle, i.e., we prove (3) and conclude (1) and (2) from it. We provide a self-contained correctness proof for a cycle joining construction for prefer-max in Section 3. Then, we show that the prefer-max shift rule, and thus the FKM theorem, immediately follow from the correctness of our construction. Hence, we also provide an alternative proof for those two constructions of prefer-max, see Section 4.

To be precise, we provide a cycle joining construction for the reverse of prefer-max. However, as we elaborate in the preliminaries, with reversing and complementing, the construction can be modified into constructions of prefer-max, prefer-min, and its reversal, see Section 2.

Our construction follows a standard pattern. We order all cycles, and repeatedly insert the cycles into the sequence constructed so far. This approach differs from [25], in which the set of all "cycle-crossing" edges is identified. Note that this set can be extracted from the prefer-max shift rule. The pattern we follow allows us to easily conclude the *Onion theorem* [42] (see also note in [9]). That is, the prefer-max over $[k+1]$ is a suffix of the prefer max over $[k]$ and hence, effectively, the reverse of prefer-max is an infinite DB sequence over the alphabet $\mathbb{N}$, see Section 4.

2. Preliminaries

We focus on alphabets of the form $\Sigma = [k]$ for $k > 0$ (recall that $[k] = \{0, \ldots, k-1\}$), or $\Sigma = \mathbb{N}$. Naturally, the symbols in Σ are totally ordered by $0 < 1 < \cdots$. A word over Σ is a sequence of symbols. Throughout the paper, non-capital letters from the Latin alphabet (e.g. u, w, x, w' etc.) denote words over Σ, and non-capital letters from the Greek alphabet denote (e.g. σ, τ, σ' etc.) denote symbols in Σ. $|w|$ denotes the length of a word w, and we say that w is an n-word if $|w| = n$. ε is the unique 0-word. For a word w, $w^0 = \varepsilon$, and $w^{t+1} = w \cdot w^t$. Σ^n is the set of all n-words over Σ. R is the reverse operator over words, i.e. $R(\sigma_0\sigma_1 \cdots \sigma_{n-2}\sigma_{n-1}) = (\sigma_{n-1}\sigma_{n-2} \cdots \sigma_1\sigma_0)$.

Definition 1. *An* (n, k)***-DB sequence*** *is a total ordering*[1] *of* $[k]^n$ *satisfying:*

1. *A word of the form* τw *is followed by a word of the form* $w\sigma$*.*
2. *If the last word is* τx*, then the first word is of the form* $x\sigma$*.*

We also consider infinite De Bruijn sequences: An n**-DB sequence** is a total ordering of $\mathbb{N}^n$ that satisfies condition 1 in Definition 1.

Definition 2. *For* $n, k > 0$*,* ***the prefer-max (resp. prefer-min)*** (n, k)***-DB sequence****, denoted* $\mathsf{Pmax}(n, k)$ *(resp.* $\mathsf{Pmin}(n, k)$*), is the sequence constructed by the following greedy algorithm:*

[1] An (n, k)-DB sequence is also commonly defined as a (cyclic) sequence of k^n symbols $\sigma_0\sigma_1 \cdots$, for which every n-word appears in it as a subword. This presentation is essentially identical to ours by writing $w_i = (\sigma_{i-(n-1) \mod k^n}) \cdots (\sigma_{i-1 \mod k^n})\sigma_i$.

- $w_0 = 0^{n-1}(k-1)$ *(resp.* $(k-1)^{n-1}0$*),*
- *If* $w_i = \sigma x$ *then* $w_{i+1} = x\tau$ *where* τ *is the maximal (resp. minimal) symbol such that* $w_j \neq x\tau$ *for all* $j \leq i$*.*

We note that the prefer max DB-sequence is indeed a DB-sequence (i.e., satisfies the conditions of Definition 1). For more details, see [35].

Example 3. $\mathsf{Pmax}(3,3)$ *is the sequence:*

$$002, 022, 222, 221, 212, 122, 220, 202, 021, 211, 112, 121, 210, 102, 020, 201, 012, 120, 200, 001, 011, 111, 110, 101, 010, 100, 000.$$

$\mathsf{Pmin}(3,3)$ *is the sequence:*

$$220, 200, 000, 001, 010, 100, 002, 020, 201, 011, 110, 101, 012, 120, 202, 021, 210, 102, 022, 221, 211, 111, 112, 121, 212, 122, 222.$$

Let $\mathsf{rev\_Pmax}(n,k)$ (resp. $\mathsf{rev\_Pmin}(n,k)$) be the reverse of $\mathsf{Pmax}(n,k)$ (resp. $\mathsf{Pmin}(n,k)$). That is, the sequence obtained by taking the ith word u_i to be $R(w_{k^n-1-i})$, i.e. the reverse of the k^n-1-i word of $\mathsf{Pmax}(n,k)$ (resp. $\mathsf{Pmin}(n,k)$).

Example 4. $\mathsf{rev\_Pmax}(3,3)$ *is the sequence:*

$$000, 001, 010, 101, 011, 111, 110, 100, 002, 021, 210, 102, 020, 201, 012, 121, 211, 112, 120, 202, 022, 221, 212, 122, 222, 220, 200.$$

$\mathsf{rev\_Pmin}(3,3)$ *is the sequence:*

$$222, 221, 212, 121, 211, 111, 112, 122, 220, 201, 012, 120, 202, 021, 210, 101, 011, 110, 102, 020, 200, 001, 010, 100, 000, 002, 022.$$

Observe that $\mathsf{Pmax}(n,k)$ and $\mathsf{Pmin}(n,k)$ (and likewise $\mathsf{rev\_Pmax}(n,k)$ and $\mathsf{rev\_Pmin}(n,k)$) are derived one from the other by replacing each symbol σ with $k-1-\sigma$. Hence, by reversing the sequence and subtracting the indices from $k-1$, properties and constructions for one of those sequences translate to corresponding properties and constructions for all other three.

The fundamental FKM-theorem [21, 22] (named after Fredricksen, Kessler, and Maiorana) links between the Pmin sequence and Lyndon words [34], as we elaborate below.

Definition 5.
1. *A word* w' *is a* ***rotation*** *of* w*, and we write* $w \sim w'$*, if* $w = xy$ *and* $w' = yx$ *for some words* x *and* y*.*
2. *A word* w *is* ***periodic*** *if* $w = x^t$ *where* $t > 1$*.*
3. *A word* $w \neq \varepsilon$ *is a* ***Lyndon-word*** *if it is non-periodic, and lexicographically-minimal among its rotations.*

Let $\leq_{lex}$ denote the lexicographic ordering of words.

Theorem 6 (FKM-Theorem)**.** *Let $L_0 <_{lex} L_1 <_{lex} \cdots$ be all Lyndon-words over $[k]$ whose length divides n, ordered lexicographically. Then,* $\mathsf{Pmin}(n,k) = L_0 L_1 \cdots$.

As we present a construction for $\mathsf{rev\_Pmax}(n,k)$, the co-lexicographic ordering is of importance to us. That is, we write $w_1 \leq_{co\text{-}lex} w_2$ if w_1 is a suffix of w_2, or we can write

$$w_1 = y_1 \sigma x, \ w_2 = y_2 \tau x, \text{ such that } \sigma < \tau.$$

Equivalently, *co-lex* can be defined by: $w_1 \leq_{co\text{-}lex} w_2$ if $R(w_1) \leq_{lex} R(w_2)$.

3. Cycle Joining Construction

In this section we define the cycle joining construction for $\mathsf{rev\_Pmax}(n,k)$, and prove its correctness. We further conclude that $\mathsf{rev\_Pmax}$ is an infinite DB sequence. We obtain these results as follows: For every $n, k > 0$ we construct an (n,k)-DB sequence, $D(n,k)$. We then show that our construction yields an n-DB sequence, $D(n)$, as $D(n,k)$ is a prefix of $D(n,k{+}1)$. Finally, we prove that $D(n,k) = \mathsf{rev\_Pmax}(n,k)$.

3.1. The Construction

A key-word is an n-word that is *co-lex* maximal among its rotations.[2] Let $key_0, key_1, key_2, \ldots$ be an enumeration of all key-words in *co-lex* order. That is, $key_0 <_{co\text{-}lex} key_1 <_{co\text{-}lex} \cdots$. Let $c(n,k)$ be the number of all key-words of length n over $[k]$. The cycle of key_m is a sequence C_m whose elements are all key_m-rotations (i.e., C_m is the equivalence class of key_m with respect to $\sim$, as defined in Definition 5).

We define an order on the elements of C_m as follows: First, $C_0 = (0^n)$, so it is given the trivial order. Now, for $m > 0$,we can write $key_m = 0^l(\sigma{+}1)w$ for some $\sigma \in \Sigma$ and $l \geq 0$. Then we set $w0^l(\sigma{+}1)$ to be the first word in C_m, and each word $\sigma w'$ in C_m is followed by $w'\sigma$. Hence, the last word in C_m is $(\sigma{+}1)w0^l$.

We thus define the corresponding functions below:

Definition 7. *For $m > 0$ let:*

- $key(C_m) = key_m = 0^l(\sigma{+}1)w$*;*
- $first(C_m) = w0^l(\sigma{+}1)$*;*
- $last(C_m) = (\sigma{+}1)w0^l$*.*

Additionally, $key(C_0) = first(C_0) = last(C_0) = 0^n$*.*

[2] This is inspired by the notion of Lyndon words - non-periodic words that are lexicographically minimal among their rotations. In our definition of a key-word we use *co-lex* ordering, take the maximal element, and do not require periodicity.

Note that the sequence $C_0, \dots, C_{c(n,k)-1}$ defines a partition of the set $[k]^n$.

Example 8. *For $n = 6$ and $k = 3$, take m such that $key_m = 002012$. Hence, $C_m = (first(C_m) = 012002, 120020, 200201, key(C_m) = 002012, 020120, last(C_m) = 201200)$.*

We are ready to present the cycle joining construction. We inductively define an ordering of all words, where at step m we extend the ordering of the elements in $\bigcup_{i=0}^{m-1} C_i$ to include the elements of C_m, while respecting the ordering within C_m defined above.

Construction 9. *For each $m < c(n,k)$, we inductively define a sequence D_m, over the words $\bigcup_{i=0}^{m} C_i$, as follows:*

- $D_0 = (0^n)$.
- Write $key_{m+1} = 0^l(\sigma{+}1)w$. D_{m+1} is obtained by inserting the sequence C_{m+1} immediately after the word $\sigma w 0^l \in D_m$.

Let $D(n,k) = D_{c(n,k)-1}$.

Remark 10. Note that if $key_{m+1} = 0^l(\sigma{+}1)w$, then $\sigma w 0^l \in D_m$ as the key-word that is a rotation of $\sigma w 0^l$ is *co-lex* smaller than $0^l(\sigma{+}1)w$.

Example 11. For $n = k = 3$, the key-words are:

$$key_0 = 000, key_1 = 001, key_2 = 011, key_3 = 111, key_4 = 002, key_5 = 102,$$
$$key_6 = 012, key_7 = 112, key_8 = 022, key_9 = 122, key_{10} = 222.$$

To demonstrate the cycle joining construction, we show $D(3,3)$ below, along with cycle-parenthesis. Parenthesis of cycle C_i are denoted $({}_i, \dots, {}_i)$. We also provide a construction illustration in Figure 1.

$$D(3,3) = ({}_0 000, {}_0)({}_1 001, 010, ({}_2 101, 011, ({}_3 111, {}_3) 110, {}_2) 100, {}_1)({}_4 002, ({}_5 021, 210, 102, {}_5) 020, ({}_6 201, 012, ({}_7 121, 211, 112, {}_7) 120, {}_6)({}_8 202, 022, ({}_9 221, 212, 122, {}_9)({}_{10} 222, {}_{10}) 220, {}_8) 200 {}_4).$$

Observe that the following two properties hold:

1. $D(3,2)$ *(colored in red) is a prefix of $D(3,3)$.*
2. $D(3,3) = \mathsf{rev\_Pmax}(3,3)$ *(see Example 3).*

Later, we will prove that these are general features of our cycle joining construction.

We show now that we indeed construct a DB sequence.

Theorem 12. *$D(n,k)$ is an (n,k)-De Bruijn sequence that starts with 0^n and ends with $(k{-}1)0^{n-1}$.*

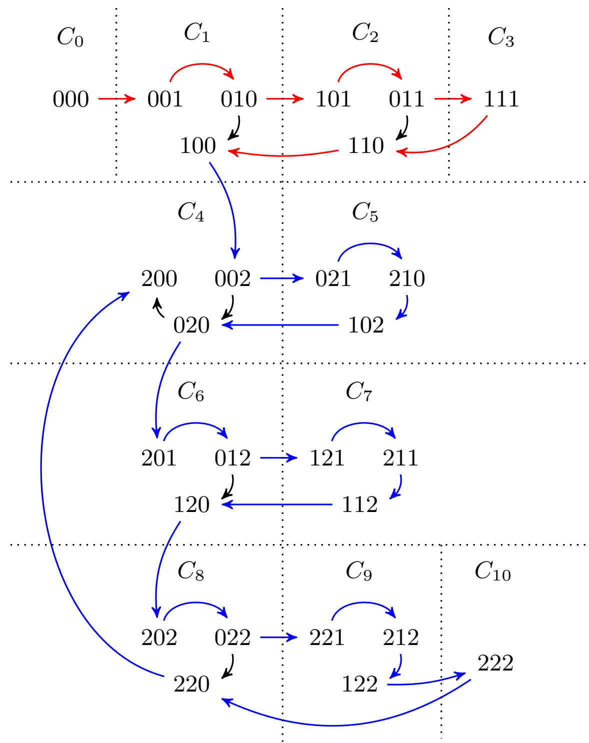

Figure 1: Our cycle joining construction. Red arrows are $D(3,2)$ successors. Blue arrows extend $D(3,2)$ into $D(3,3)$. Black arrows are cycle-successors that are not $D(3,3)$ successors.

Proof. Clearly, each $w \in [k]^n$ appears exactly once in $D(n,k)$. Hence, to show that $D(n,k)$ is a DB-sequence, we prove by induction that the successor property is an invariant of the construction: it holds for each sequence D_m, $m < c(n,k){-}1$. It vacuously holds for $D_0 = (0^n)$. In addition, by the construction rule, if we insert a cycle C_m between τw and $w\sigma$, or after the final word τw, then $\mathit{first}(C_m)$ is of the form $w\xi$, and thus $\mathit{last}(C_m) = \xi w$. Therefore, $D(n,k) = D_{c(n,k)-1}$ is a DB sequence.

Now, since at each step we insert cycles after some word, the first word in $D(n,k)$ is 0^n. It remains to show that the last word of $D(n,k)$ is $(k{-}1)0^{n-1}$. By the construction rule, for each $\tau < k{-}1$, the cycle of $(\tau{+}1)0^{n-1}$ is inserted after $\tau 0^{n-1}$. Hence, $0^n, 10^{n-1}, 20^{n-1}, \ldots, (k{-}1)0^{n-1}$ is a subsequence of $D(n,k)$. Assume towards a contradiction that $(k{-}1)0^{n-1}$ is not the final word of $D(n,k)$, and thus its successor is $w = 0^{n-1}\sigma \in C_m$, for some m and $\sigma < k{-}1$. Hence, by the ordering of C_m, $(k-1)0^{n-1}$ also precedes $\sigma 0^{n-1}$, in contradiction to the subsequence established above. ☐

Following Gabric et al. [25], we define the weight of a word to be the sum of its symbols (viewed as natural numbers). Note that the next hold:

- All words in a cycle has the same weight.
- For $w_i \in C_i$, $w_j \in C_j$ where $i < j$, the weight of w_i is smaller than, or equal to the weight of w_j.

Hence, by the inductive argument in the proof of Theorem 12:

Corollary 13. *For $t \leq (k-1)n$, when applied over the cycles whose words are of weight at most t, our construction rule constructs a DB sequence of all words of weight at most t.*

From this point onward, given two n-words $w, w' \in [k]^n$ we write $w < w'$ if w appears before w' in $D(n,k)$. Note that for each $k > 0$, the linear ordering $D(n, k{+}1)$ extends $D(n,k)$. Hence, $<$ is well defined as it does not depend on k. We further remark that the notation $<$ is reasonable since for the case $n = 1$, $<$ coincides with the natural ordering of $[k]$.

3.2. The Infinite Nature of the Construction

We turn to prove that our cycle joining construction effectively provides us with an infinite De Bruijn sequence:

Theorem 14. $D(n) = \bigcup_{k=0}^{\infty} D(n,k)$ *is an n-De Bruijn sequence.*

In order to prove Theorem 14, we show that each $D(n,k)$ is a prefix of $D(n,k{+}1)$. (Recall that $D(n,k{+}1)$ is obtained by inserting the cycles $C_0, \ldots, C_{c(n,k+1)-1}$, one by one, into the sequence constructed so far). Towards proving our claim, we prove some progression property of the construction: we show that if $m < r$, then the cycle C_r is not inserted before the cycle C_m. Later, we will use this to prove that $D(n,k)$ is a prefix of $D(n,k{+}1)$ as follows. We show that after we insert the cycles $C_0, \ldots, C_{c(n,k)-1}$ (and thus construct $D(n,k)$), the cycle $C_{c(n,k)}$ is appended to $D(n,k)$. As a result, by the progression property, we conclude the required. The next proposition formalizes the discussed progression property:

Proposition 15 (Progression Property)**.** *If $m < r$, then $\mathit{first}(C_m) < \mathit{first}(C_r)$.*

The remainder of this subsection is devoted to proving Proposition 15. We start with a few technical lemmata.

Lemma 16. *If $0^l(\sigma{+}1)w$ is a key-word, then $0^l\sigma w$ is also a key-word.*

Proof. This lemma is essentially identical to [5, Lemma 7], which deals with the analogous case for *lex*-ordering, and minimal rotation. Essentially, the proof goes by arguing that if a rotation of $0^l\sigma w$ is *co-lex*-smaller than it, then the corresponding rotation of $0^l(\sigma+1)w$ is also *co-lex*-smaller than it. □

Lemma 17. *If $key_m = 0^l w$ where $0^l w \neq 0^n$, and $key_r = zw$ where $z \neq 0^l$, then all elements of C_r precede $w0^l$.*

Proof. We prove by induction on z ordered by lex. That is, we assume that the lemma holds for each $z' <_{lex} z$, and prove for z. Note that $m < r$, and consider the sequence D_r, obtained by inserting C_r into D_{r-1} (which already includes the elements of C_m). As $z \neq 0^l$, we may write $key_r = zw = 0^t(\tau{+}1)yw$. By the construction rule, C_r is inserted immediately after $\tau yw0^t$. It is sufficient to prove that $\tau yw0^t$ precede $w0^l$.

Note that, by Lemma 16, $0^t\tau yw$ is a key-word. Hence, if $0^t\tau y \neq 0^l$, we are done by the induction hypothesis. Otherwise, $\tau yw0^t = 0^{|\tau y|}w0^t \in C_m$. In addition, note that $zw = 0^t(\tau{+}1)yw$ implies that $t < l$. Consider the prefix of C_m that starts in $key_m = 0^l w$. In this prefix, the leading zeros of $0^l w$ are shifted to the right one by one. Hence, since $t < l$, $\tau yw0^t = 0^{|\tau y|}w0^t < w0^l$, as required. □

We can now prove Proposition 15.

Proof of Proposition 15. Note that it suffices to prove the result only for $r = m{+}1$. Write $key_{m+1} = 0^l(\sigma{+}1)w$. C_{m+1} is inserted after $\sigma w0^l$. By Lemma 16, $0^l\sigma w$ is a key-word. If $0^l\sigma w = key_m$, we are done since we then have $\mathit{first}(C_m) \leq \sigma w0^l < \mathit{first}(C_{m+1})$. Otherwise, we have that

$$0^l\sigma w <_{co\text{-}lex} key_m <_{co\text{-}lex} key_{m+1} = 0^l(\sigma{+}1)w.$$

Therefore, $key_m = z\sigma w$ where $z \neq 0^l$. Hence, by Lemma 17, all elements of C_m precede $\sigma w0^l$, and thus, in particular, $\mathit{first}(C_m) < \mathit{first}(C_{m+1})$. □

Finally, Theorem 14 follows.

Proof of Theorem 14. It is sufficient to show that $D(n,k)$ is a prefix of $D(n,k{+}1)$. By Theorem 12, $D(n,k) = D_{c(n,k)-1}$ ends in $(k{-}1)0^{n-1}$. $key_{c(n,k)}$ is the *co-lex* minimal key-word in $[k{+}1]^n \setminus [k]^n$, hence $key_{c(n,k)} = 0^{n-1}k$. By the construction rule, $C_{c(n,k)}$ is inserted after $(k{-}1)0^{n-1}$. By Proposition 15, every cycle C_m where $m > c(n,k)$ is inserted after $\mathit{first}(C_{c(n,k)})$ (and thus after $(k{-}1)0^{n-1}$). Consequently, $D(n,k) = D_{c(n,k)-1}$ is a prefix of $D(n,k{+}1) = D_{c(n,k+1)-1}$. □

3.3. A Nesting Structure

We next describe an interesting nesting structure that exists in our cycle joining construction. This structure will be used later to prove that our sequence is the reverse of the prefer-max sequence.

We start with an immediate observation on the structure of $D(n,k)$, implied by Proposition 15, which we call the *parenthesis property.* If $m < r$, then, by Proposition 15, C_r was not inserted before C_m. Therefore, either (1) the cycle C_r entirely follows C_m, or (2) it is embedded into C_m. We term this property *the parenthesis property* as we consider (virtual) parenthesis that wrap each cycle, as in Example 11. We turn to formalize the parenthesis property, and show when either of the cases holds.

Corollary 18 (The Parenthesis Property)**.** *If C_m and C_r are two cycles such that $m < r$, then only one of the following holds:*

1. *$last(C_m) < first(C_r)$, or*
2. *$first(C_m) < first(C_r) \leq last(C_r) < last(C_m)$.*

Example 19. Recall Example 11. Cycle C_5 entirely precedes C_6, demonstrating the first case. Cycle C_{10} is embedded in C_4, demonstrating the second case.

Definition 20. 1. *We say that C_r **is embedded in** C_m, if*

$$first(C_m) < first(C_r) \leq last(C_r) < last(C_m).$$

2. *C_r is said to be **immediately embedded in** C_m if there is no cycle C_l such that C_r is embedded in C_l and C_l is embedded in C_m.*
3. *We inductively define the statement: "C_r **is t-embedded in** C_m":*
   - *C_r is 1-embedded in C_m if it is immediately embedded in C_m.*
   - *C_r is $(t+1)$-embedded in C_m if there exists a cycle C_l such that C_r is t-embedded in C_l and C_l is 1-embedded in C_m.*

We now investigate the possible relations between key-words of cycles C_m and C_r, considering the next two cases. When C_r immediately follows C_m, and when it is immediately-embedded in C_m. First, we show that if we insert a cycle C_r after $last(C_m)$, then key_r is obtained by increasing the first non-zero symbol in key_m by one.

Lemma 21. *If C_r immediately follows C_m, i.e., $first(C_r)$ is the successor of $last(C_m)$, and $key_m = 0^l(\sigma{+}1)w$, then $key_r = 0^l(\sigma{+}2)w$.*

Proof. Consider the sequence D_r, obtained by inserting C_r into D_{r-1}. Write $key_r = 0^{l'}(\tau{+}1)w'$. By the construction rule, C_r was inserted after $\tau w' 0^{l'}$. By Proposition 15, D_{r-1} already includes the elements of C_m thus $\tau w' 0^{l'} = last(C_m) = (\sigma{+}1)w0^l$. As $0^l(\sigma{+}1)w$ and $0^{l'}(\tau{+}1)w'$ are key-words, both w and w' end in a non-zero symbol. Hence, equality $\tau w' 0^{l'} = (\sigma{+}1)w0^l$ proves that $l = l'$, $\sigma{+}1 = \tau$, and $w = w'$. As a result, $key_r = 0^{l'}(\tau + 1)w' = 0^l(\sigma{+}2)w$, as required. □

Now, we show that if we choose to embed C_r in C_m, then key_m is obtained by zeroing the first non-zero symbol in key_r.

Lemma 22. *If C_r is immediately embedded in C_m and $key(C_r) = 0^i(\sigma{+}1)0^j w$ where w does not start with 0, then:*

- *$key(C_m) = 0^{i+1+j}w$.*
- *If $u \in C_m$ and $last(C_r) < u$, then $u = 0^{j_2}w0^{i+1+j_1}$ where $j_1{+}j_2 = j$.*

Proof. By Lemma 16, $0^{i+1+j}w$ is a key-word. Hence, to prove the first item, we need to show that $0^{i+1+j}w \in C_m$. Now, since C_r is immediately embedded in C_m, the predecessor of $\mathit{first}(C_r)$ is $v \in C_m$, or $\mathit{last}(C)$, for some cycle C that is also immediately embedded in C_m. By repeatedly applying this reasoning, we construct a sequence of cycles $C_{i_0}, C_{i_1}, \ldots, C_{i_l}$ such that

- $C_{i_l} = C_r$.
- For each $t \in \{1, \ldots, l\}$, the predecessor of $\mathit{first}(C_{i_t})$ is $\mathit{last}(C_{i_{t-1}})$.
- The predecessor of $\mathit{first}(C_{i_0})$ is a word $v \in C_m$.

Now, by applying Lemma 21 l-times, $\mathit{key}_{i_0} = 0^i(\tau{+}1)0^jw$ where $\tau{+}1 = \sigma{+}1{-}l$. Therefore, $\mathit{first}(C_{i_0}) = 0^jw0^i(\tau{+}1)$, and, by the construction rule,

$$v = \tau 0^j w 0^i.$$

To prove that indeed $0^{i+1+j}w \in C_m$, we need to show that $\tau = 0$ as $\tau = 0$ implies that $0^{i+1+j}w$ is a rotation of $v \in C_m$. As $\mathit{key}_{i_0} = 0^i(\tau{+}1)0^jw$ is a key-word, by Lemma 16, $0^i\tau 0^jw$ is also a key-word. As it is a rotation of $v \in C_m$ it is key_m. If $\tau > 0$, then, by the definition of last, $v = \tau 0^jw0^i = \mathit{last}(C_m)$, in contradiction to the fact that C_r is embedded in C_m. Hence, $\tau = 0$ and the first item holds. Moreover, the second item easily follows as $v, u, \mathit{last}(C_m) \in C_m$ and

$$v = 0^{j+1}w0^i < u \leq \mathit{last}(C_m) = w0^{i+1+j}. \quad \square$$

By applying the previous lemma several times, we conclude the next corollary.

Corollary 23. *Assume that C_r is t-embedded in C_m. Then we have that:*

1. *$\mathit{key}(C_r)$ includes at least t non-zero symbols.*
2. *If we write $\mathit{key}(C_r) = uv$ where u is the minimal prefix of $\mathit{key}(C_r)$ that includes t non-zero symbols, then $\mathit{key}(C_m) = 0^{|u|}v$.*

3.4. Equivalence to the Reverse of the Prefer-Max

We are ready to show that we indeed construct the reverse of prefer-max.

Theorem 24. *$D(n, k) = \mathsf{rev\_Pmax}(n, k)$.*

Recall that $\mathsf{Pmax}(n, k)$ is the only De Bruijn sequence that (1) starts with $0^{n-1}(k{-}1)$, and (2) $w(\tau{+}1)$ appears in it before $w\tau$ for every $w \in [k]^{n-1}$ and $\tau \in [k]$. Hence, to prove Theorem 24, we shall prove the symmetric property: (1) $D(n, k)$ ends in $(k{-}1)0^{n-1}$, and (2) $\tau w < (\tau{+}1)w$. The former was already obtained in Theorem 12, and we focus on proving the later.

Proposition 25. *For any n-word τw, $\tau w < (\tau{+}1)w$.*

Proof. Let C_r be the cycle of $(\tau{+}1)w$. We start by proving the claim for the restricted case $(\tau{+}1)w = \mathit{last}(C_r)$. Write $\mathit{key}(C_r) = 0^l(\sigma{+}1)w'$ and hence, $\mathit{last}(C_r) = (\sigma{+}1)w'0^l = (\tau{+}1)w$, and $\mathit{first}(C_r) = w'0^l(\sigma{+}1)$. By the construction rule, C_r is inserted after $\sigma w'0^l = \tau w$, and the required follows.

We turn to deal with the general case in which $(\tau{+}1)w \neq \mathit{last}(C_r)$. In this case we may write $\mathit{key}(C_r) = 0^l(\sigma{+}1)w_1(\tau{+}1)w_2$, where

$$(\tau{+}1)w = (\tau{+}1)w_2 0^l(\sigma{+}1)w_1. \tag{1}$$

Let C_m be the cycle of τw. Clearly, the maximal rotation of τw is *co-lex* smaller than the maximal rotation of $(\tau{+}1)w$. Consequently, $\mathit{key}_m <_{\mathit{co\text{-}lex}} \mathit{key}_r$ and thus $m < r$. Hence, by Proposition 15, $\mathit{first}(C_m) < \mathit{first}(C_r)$.

Now, if $\mathit{last}(C_m) < \mathit{first}(C_r)$, then every element of C_m precedes every element of C_r and we are done. Otherwise, by the parenthesis property, C_r is embedded in C_m. For $\sigma \in [k]$, let $|w|_\sigma$ denote the number of occurrences of σ in w, and note that $|\tau w|_0 - |(\tau+1)w|_0 \in \{0, 1\}$. Use Corollary 23 to conclude that $|\tau w|_0 - |(\tau{+}1)w|_0 = 1$ and that C_r is immediately embedded in C_m. Moreover, as $|\tau w|_0 - |(\tau{+}1)w|_0 = 1$, we have $\tau = 0$. Hence, by Equation 1,

$$\tau w = 0w_2 0^l(\sigma{+}1)w_1. \tag{2}$$

Furthermore, we get that the key of the cycle that includes $(\tau{+}1)w$, $\mathit{key}(C_r) = 0^l(\sigma{+}1)w_1 1 w_2$. Write $w_1 = 0^j w_1' 0^i$ and $w_2 = 0^p w_2'$ where w_1' and w_2' do not start or end with zero. Therefore, we have $\mathit{key}(C_r) = 0^l(\sigma{+}1)0^j w_1' 0^i 1 0^p w_2'$.

Assume towards a contradiction that $1w = (\tau{+}1)w < \tau w = 0w$. Recall that $\tau w = 0w \in C_m$ and C_r is embedded in C_m, and conclude (based on our assumption that $1w = (\tau{+}1)w < \tau w = 0w$) that the last element of C_r must also appear before $0w$: $1w \leq \mathit{last}(C_r) < \tau w = 0w \leq \mathit{last}(C_m)$. Therefore, by Lemma 22,

$$\tau w = 0w = 0^{j_2} w_1' 0^i 1 0^p w_2' 0^{l+1+j_1}, \text{ where } j_1{+}j_2 = j. \tag{3}$$

By Equations 2 and 3, since $w_1 = 0^j w_1' 0^i$ and $w_2 = 0^p w_2'$,we have:

$$0^{j_2} w_1' 0^i 1 0^p w_2' 0^{l+1+j_1} = 0^{p+1} w_2' 0^l (\sigma{+}1) 0^j w_1' 0^i. \tag{4}$$

Therefore, $|0^{j_2} w_1' 0^i 1 0^r w_2' 0^{l+1+j_1+1}|_1 = |0^{p+1} w_2' 0^l(\sigma{+}1)0^j w_1' 0^i|_1$ and thus $\sigma{+}1 = 1$. Hence,

$$\mathit{key}(C_r) = 0^l 1 0^j w_1' 0^i 1 0^p w_2' \tag{5}$$

and Equation 4 can be rewritten as follows:

$$0^{j_2} w_1' 0^i 1 0^p w_2' 0^{l+1+j_1} = 0^{p+1} w_2' 0^l 1 0^j w_1' 0^i. \tag{6}$$

For the remainder of the proof we assume that $w'_1 \neq \varepsilon$ and $w'_2 \neq \varepsilon$. The other cases are dealt similarly.

By deleting the initial and final segments of zeros, we get from Equation 6,

$$j_2 = p{+}1, \quad w'_1 0^i 1 0^p w'_2 = w'_2 0^l 1 0^j w'_1. \tag{7}$$

Now, by Equation 5,

$$0^i 1 0^p w'_2 0^l 1 0^j w'_1 \leq_{co\text{-}lex} 0^l 1 0^j w'_1 0^i 1 0^p w'_2. \tag{8}$$

By Equation 7, these words have the same suffix thus $0^i 1 0^p \leq_{co\text{-}lex} 0^l 1 0^j$. Hence, $j \leq p$. Therefore, by Equation 3, $j_2 \leq p$, in contradiction to Equation 7. □

Finally, Theorem 24 follows.

Proof of Theorem 24. $\mathsf{rev\_Pmax}(n,k)$ is the only sequence that includes all n-words (and no other elements), ends with $(k{-}1)0^{n-1}$, and satisfies $\tau w < (\tau{+}1)w$ for each $\tau \leq k{-}2$ and $w \in [k]^{n-1}$. Hence, the theorem is implied by Theorem 12 and Proposition 25. □

4. Properties of Prefer-Max Implied by Our Construction

We present applications induced by our construction. Specifically, first, we prove that $\mathsf{rev\_Pmax}$ is in fact an infinite De Bruijn sequence. Second, we extract from the construction the shift rule for $\mathsf{rev\_Pmax}$, proposed in [6]. Finally, as noted in [6], this shift rule provides an alternative proof for the FKM-theorem.

4.1. A Proof of the Onion Theorem

Inside the proof of Theorem 14, we provided an alternative proof of the Onion theorem [42] (see also a note in [9]):

Theorem 26 (Onion Theorem [42])**.** *For all n, k, $\mathsf{rev\_Pmax}(n,k)$ is a prefix of $\mathsf{rev\_Pmax}(n,k+1)$.*

Proof. From the proof of Theorem 14, $D(n,k)$ is a prefix of $D(n,k{+}1)$. Moreover, from Theorem 24 we have that $D(n,k)$ equals to $\mathsf{rev\_Pmax}(n,k)$, and the result follows. □

Note that it follows that $\mathsf{rev\_Pmax}(n) = \bigcup_{k=1}^{\infty} \mathsf{rev\_Pmax}(n,k)$ is an infinite De Bruijn sequence.

4.2. An Efficiently Computable Shift Rule

By the correctness of our cycle joining construction, we conclude the correctness of the efficient shift rule given in [6]. For a word w, we write $\mathit{last}(w)$ if $w = \mathit{last}(C)$ for a cycle C. The successor function of $\mathsf{rev\_Pmax}(n,k)$ (resp. $\mathsf{rev\_Pmax}(n)$) is $\mathit{succ} : [k]^n \setminus \{(k-1)0^{n-1}\} \to [k]^n$ (resp. $\mathit{succ} : \mathbb{N}^n \to \mathbb{N}^n$), defined by

$$\mathit{succ}(\sigma w) = \begin{cases} w(\sigma{+}1) & \text{if } \mathit{last}((\sigma{+}1)w) \\ w0 & \text{if } \neg\mathit{last}((\sigma{+}1)w) \text{ and } \mathit{last}(\sigma w) \\ w\sigma & \text{otherwise} \end{cases} .$$

Theorem 27 (Amram et al. [6])**.** *succ is a shift rule for* $\mathsf{rev\_Pmax}(n,k)$.

Proof. First, assume that $\mathit{last}((\sigma{+}1)w)$. Let C be the cycle of $(\sigma{+}1)w$, and note that $\mathit{first}(C) = w(\sigma{+}1)$. Hence, by Construction 9, C was inserted after σw. Furthermore, by the construction rule, no other cycle C' was inserted after σw afterwards and thus the successor of σw is $w(\sigma{+}1)$.

Now, we handle the second case: $\neg\mathit{last}((\sigma{+}1)w)$ and $\mathit{last}(\sigma w)$. Write $\sigma w \in C_m$, and hence, $\mathit{first}(C_m) = w\sigma$. Let $w\tau$ be the successor of σw. We need to show that $\tau = 0$. First, we argue that $\neg\mathit{first}(w\tau)$. Assume towards a contradiction that $\mathit{first}(w\tau)$. Hence, by the construction rule, the cycle of $w\tau$ was inserted after $(\tau{-}1)w$. Use Proposition 15 to conclude that no cycle was inserted between $(\tau{-}1)w$ and $w\tau$. Therefore, $\tau{-}1 = \sigma$, which implies that $w\tau = w(\sigma+1)$, and $\mathit{last}((\sigma{+}1)w)$ follows, in contradiction to the assumption. Therefore, $\neg\mathit{first}(w\tau)$ and hence, in particular, $\tau \neq \sigma$ (since $\mathit{last}(\sigma w)$ implies $\mathit{first}(w\sigma)$). As a result, σw and $w\tau$ are not in the same cycle. Furthermore, $\neg\mathit{first}(w\tau)$ implies that the cycle of σw is immediately embedded in the cycle of $w\tau$. By the second bullet of Lemma 22, $\tau = 0$ as required.

Lastly, we deal with the third case. Hence, $\neg\mathit{last}(\sigma w)$ and thus the successor of σw in its cycle is $w\sigma$. Therefore, we should verify that no cycle C_r was inserted between σw and $w\sigma$. Assume otherwise, and conclude that $\mathit{first}(C_r) = w(\sigma{+}1)$. Hence, $\mathit{last}(C_r) = (\sigma{+}1)w$, in contradiction to the case we are dealing with. □

By Theorems 26 and 27, we conclude,

Theorem 28. *succ is a shift rule for* $\mathsf{rev\_Pmax}(n)$*.*

4.3. A Proof of the FKM theorem

Following an observation from [6], our results form an alternative proof for the seminal FKM-theorem (Theorem 6). This alternative proof can be summarized as follows: For $n, k > 0$, let $\mathit{next} : [k]^n \setminus \{0(k-1)^n\} \to [k]^n$ be the function constructed from succ by the next rule: if $\mathit{succ}(\sigma_1 \cdots \sigma_n) = \sigma_2 \cdots \sigma_{n+1}$, then

$$\mathit{next}((k{-}1){-}\sigma_1 \cdots, (k{-}1){-}\sigma_n) = ((k{-}1){-}\sigma_2 \cdots, (k{-}1){-}\sigma_{n+1}).$$

Hence, *next* is a shift rule for $\mathsf{rev\_Pmin}(n,k)$. Now, let $next^{-1}$ be the function constructed from *next* by the next rule: if $next(\sigma_1 \cdots \sigma_n) = \sigma_2 \cdots \sigma_{n+1}$, then

$$next^{-1}(\sigma_{n+1} \cdots \sigma_2) = \sigma_n \cdots \sigma_1.$$

Hence, $next^{-1}$ is a shift rule for $\mathsf{Pmin}(n,k)$. We leave for the reader to verify that $next^{-1}$ is the shift rule proposed in [5] (details can also be found in [6]).

Now, let $L_0, L_1, \ldots$ be an enumeration of all Lyndon words (recall Definition 5) over $[k]$ whose length divides n, ordered lexicographically. Therefore, according to the proof of Theorem 4 in [5], $next^{-1}$ constructs the sequence $L_0 L_1 \cdots$, which implies that $\mathsf{Pmin}(n,k) = L_0 L_1 \cdots$.

5. Conclusion

For all $n, k > 0$, we presented a cycle joining construction for the reverse of prefer-max sequence, $\mathsf{rev\_Pmax}(n,k)$. Since the sequences $\mathsf{Pmax}(n,k)$, $\mathsf{Pmin}(n,k)$, and $\mathsf{rev\_Pmin}(n,k)$ can be derived from $\mathsf{rev\_Pmax}(n,k)$, our construction can be modified into a cycle joining construction of any of those sequences.

We showed that our construction implies the correctness of the *Onion-theorem*. That is, for all $n, k > 0$, $\mathsf{rev\_Pmax}(n,k)$ is a prefix of $\mathsf{rev\_Pmax}(n,k{+}1)$, and thus $\mathsf{rev\_Pmax}(n)$ is an infinite DB sequence. Moreover, we showed that our construction also implies the correctness of the shift rules given in [6]. These shift rules are efficiently computable [5, 6].

As a result, our construction also implies the seminal FKM-theorem (Theorem 6). This theorem was presented in [22] with only a partial proof: the described concatenation of Lyndon words constructs a De Bruijn sequence. A quarter of a century later, Moreno gave an alternative proof to that fact [37], and only a decade later, extended the proof, together with Perrin, into a complete proof for the FKM theorem [38]. Amram et al. [6] proved that the shift rule given in Section 4, combined with statements proved in [5, Theorem 4] provide an alternative proof for Theorem 6. Hence, our cycle joining construction also constitutes an alternative proof for the FKM-theorem.

References

[1] Geoffrey Karl Aguirre, Marcelo Gomes Mattar, and Lucía Magis-Weinberg. de Bruijn cycles for neural decoding. *NeuroImage*, 56(3):1293–1300, 2011.

[2] Abbas Alhakim. A simple combinatorial algorithm for de Bruijn sequences. *Am. Math. Mon.*, 117(8):728–732, 2010.

[3] Abbas Alhakim and Mufutau Akinwande. A recursive construction of nonbinary de Bruijn sequences. *Des. Codes Cryptogr.*, 60(2):155–169, 2011.

[4] Abbas Alhakim and Maher Nouiehed. Stretching de Bruijn sequences. *Des. Codes Cryptogr.*, 85(2):381–394, 2017.

[5] Gal Amram, Yair Ashlagi, Amir Rubin, Yotam Svoray, Moshe Schwartz, and Gera Weiss. An efficient shift rule for the prefer-max De Bruijn sequence. *Discret. Math.*, 342(1):226–232, 2019.

[6] Gal Amram, Amir Rubin, Yotam Svoray, and Gera Weiss. De Bruijn Sequences: From Games to Shift-Rules to a Proof of the Fredricksen-Kessler-Maiorana Theorem. *arXiv e-prints*, page arXiv:1805.02405v3, 2020.

[7] Fred S. Annexstein. Generating De Bruijn sequences: An efficient implementation. *IEEE Trans. Computers*, 46(2):198–200, 1997.

[8] Verónica Becher and Pablo Ariel Heiber. On extending de bruijn sequences. *Inf. Process. Lett.*, 111(18):930–932, 2011.

[9] Verónica Becher and Lucas Cortés. Extending de bruijn sequences to larger alphabets. *Information Processing Letters*, 168:106085, 2021.

[10] Taejoo Chang, Iickho Song, Hyung Myung Kim, and Sung Ho Cho. Cross-joins in de Bruijn sequences and maximum length linear sequences. *IEICE TRANSACTIONS on Fundamentals of Electronics, Communications and Computer Sciences*, 76(9):1494–1501, 1993.

[11] Zuling Chang, Martianus Frederic Ezerman, San Ling, and Huaxiong Wang. Construction of de Bruijn sequences from product of two irreducible polynomials. *Cryptogr. Commun.*, 10(2):251–275, 2018.

[12] N. G. De Bruijn. A combinatorial problem. *Proceedings of the Koninklijke Nederlandse Akademie van Wetenschappen. Series A*, 49(7):758, 1946.

[13] Junwu Dong and Dingyi Pei. Construction for de Bruijn sequences with large stage. *Des. Codes Cryptogr.*, 85(2):343–358, 2017.

[14] Patrick Baxter Dragon, Oscar I. Hernandez, Joe Sawada, Aaron Williams, and Dennis Wong. Constructing de bruijn sequences with co-lexicographic order: The k-ary grandmama sequence. *Eur. J. Comb.*, 72:1–11, 2018.

[15] Patrick Baxter Dragon, Oscar I Hernandez, and Aaron Williams. The grandmama de Bruijn sequence for binary strings. In *LATIN 2016: Theoretical Informatics*, pages 347–361. Springer, 2016.

[16] Tuvi Etzion. Self-dual sequences. *J. Comb. Theory, Ser. A*, 44(2):288–298, 1987.

[17] C Flye-Sainte Marie. Solution to problem number 58. *L'Intermédiaire des Mathématiciens*, 1:107–110, 1894.

[18] LR Ford. A cyclic arrangement of m-tuples, report no. *P-1071, Rand Corporation, Santa Monica, California*, 1957.

[19] Harold Fredricksen. Generation of the ford sequence of length 2n, n large. *Journal of Combinatorial Theory, Series A*, 12(1):153–154, 1972.

[20] Harold Fredricksen. A class of nonlinear de Bruijn cycles. *J. Comb. Theory, Ser. A*, 19(2):192–199, 1975.

[21] Harold Fredricksen and Irving Kessler. Lexicographic compositions and deBruijn sequences. *Journal of Combinatorial Theory, Series A*, 22(1):17–30, 1977.

[22] Harold Fredricksen and James Maiorana. Necklaces of beads in k colors and k-ary de Bruijn sequences. *Discrete Mathematics*, 23(3):207–210, 1978.

[23] Daniel Gabric and Joe Sawada. A de Bruijn sequence construction by concatenating cycles of the complemented cycling register. In *International Conference on Combinatorics on Words*, pages 49–58. Springer, 2017.

[24] Daniel Gabric, Joe Sawada, Aaron Williams, and Dennis Wong. A framework for constructing de Bruijn sequences via simple successor rules. *Discrete Mathematics*, 341(11):2977–2987, 2018.

[25] Daniel Gabric, Joe Sawada, Aaron Williams, and Dennis Wong. A successor rule framework for constructing k-ary de Bruijn sequences and universal cycles. *IEEE Trans. Inf. Theory*, 66(1):679–687, 2020.

[26] Irving John Good. Normal recurring decimals. *Journal of the London Mathematical Society*, 1(3):167–169, 1946.

[27] Erik R. Hauge and Tor Helleseth. De Bruijn sequences, irreducible codes and cyclotomy. *Discret. Math.*, 159(1-3):143–154, 1996.

[28] Yuejiang Huang. A new algorithm for the generation of binary de Bruijn sequences. *J. Algorithms*, 11(1):44–51, 1990.

[29] Cees J. A. Jansen, Wouter G. Franx, and Dick E. Boekee. An efficient algorithm for the generation of DeBruijn cycles. *IEEE Trans. Inf. Theory*, 37(5):1475–1478, 1991.

[30] Abraham Lempel. On a homomorphism of the de Bruijn graph and its applications to the design of feedback shift registers. *Computers, IEEE Transactions on*, 100(12):1204–1209, 1970.

[31] Chaoyun Li, Xiangyong Zeng, Chunlei Li, and Tor Helleseth. A class of de Bruijn sequences. *IEEE Trans. Inf. Theory*, 60(12):7955–7969, 2014.

[32] Chaoyun Li, Xiangyong Zeng, Chunlei Li, Tor Helleseth, and Ming Li. Construction of de bruijn sequences from lfsrs with reducible characteristic polynomials. *IEEE Trans. Inf. Theory*, 62(1):610–624, 2016.

[33] Yaw-Ling Lin, Charles B. Ward, Bharat Jain, and Steven Skiena. Constructing orthogonal de Bruijn sequences. In *Algorithms and Data Structures - 12th International Symposium, WADS*, pages 595–606. Springer, 2011.

[34] Roger C Lyndon. On Burnside's problem. *Transactions of the American Mathematical Society*, 77(2):202–215, 1954.

[35] Monroe H Martin. A problem in arrangements. *Bulletin of the American Mathematical Society*, 40(12):859–864, 1934.

[36] Maryna Miroschnyk, Tetyana Korytchinko, Olexandr Demihev, Viktoriia Krylova, Dmitrij Karaman, and Inna Filippenko. Practical methods for de Bruijn sequences generation using non-linear feedback shift registers. In *2018 14th International Conference on Advanced Trends in Radioelecrtronics, Telecommunications and Computer Engineering (TCSET)*, pages 1157–1161. IEEE, 2018.

[37] Eduardo Moreno. On the theorem of fredricksen and maiorana about de Bruijn sequences. *Advances in Applied Mathematics*, 33(2):413–415, 2004.

[38] Eduardo Moreno and Dominique Perrin. Corrigendum to "on the theorem of fredricksen and maiorana about de Bruijn sequences"[adv. in appl. math. 33 (2)(2004) 413–415]. *Advances in Applied Mathematics*, 62:184–187, 2015.

[39] Johannes Mykkeltveit and Janusz Szmidt. On cross joining de bruijn sequences. *Contemporary Mathematics*, 63:335–346, 2015.

[40] Joe Sawada, Aaron Williams, and Dennis Wong. A surprisingly simple de Bruijn sequence construction. *Discrete Mathematics*, 339(1):127–131, 2016.

[41] Joe Sawada, Aaron Williams, and Dennis Wong. A simple shift rule for k-ary de Bruijn sequences. *Discrete Mathematics*, 340(3):524 – 531, 2017.

[42] Moshe Schwartz, Yotam Svoray, and Gera Weiss. On Embedding De Bruijn Sequences by Increasing the Alphabet Size. *arXiv e-prints*, page arXiv:1906.06157, June 2019.

[43] Meltem Sönmez Turan. Evolutionary construction of de Bruijn sequences. In Yan Chen, Alvaro A. Cárdenas, Rachel Greenstadt, and Benjamin I. P. Rubinstein, editors, *Proceedings of the 4th ACM Workshop on Security and Artificial Intelligence, AISec 2011, Chicago, IL, USA, October 21, 2011*, pages 81–86. ACM, 2011.

[44] T. van Aardenne-Ehrenfest and N. G. De Bruijn. Circuits and trees in oriented linear graphs. *Simon Stevin: Wis-en Natuurkundig Tijdschrift*, 28:203, 1951.

[45] Gera Weiss. A combinatorial game approach to state nullification by hybrid feedback. In *Decision and Control, 2007 46th IEEE Conference on*, pages 4643–4647. IEEE, 2007.

[46] Weiguo Zhang, Shuanggen Liu, and Huawei Huang. An efficient implementation algorithm for generating de Bruijn sequences. *Comput. Stand. Interfaces*, 31(6):1190–1191, 2009.