Investigating the discrepancy property of de Bruijn sequences

Daniel Gabric Joe Sawada*

May 5, 2020

Abstract

The discrepancy of a binary string refers to the maximum (absolute) difference between the number of ones and the number of zeroes over all possible substrings of the given binary string. We provide an investigation of the discrepancy of known simple constructions of de Bruijn sequences. Furthermore, we demonstrate constructions that attain the lower bound of $\Theta(n)$ and a new construction that attains the previously known upper bound of $\Theta(\frac{2^n}{\sqrt{n}})$. This extends the work of Cooper and Heitsch [*Discrete Mathematics*, 310 (2010)].

1 Introduction

Let $\mathbf{B}(n)$ denote the set of binary strings of length n. A *de Bruijn sequence* is a circular string of length 2^n that contains every string in $\mathbf{B}(n)$ as a substring. By this definition, each such substring must occur exactly once. As an example,

$$\underline{1111110111100111010111}000110110100110010110000101010001001000000$$

is a de Bruijn sequence of order $n = 6$; it contains each length 6 binary string as a substring when viewed circularly. There is an extensive literature on de Bruijn sequences motivated in part by their random-like properties. As articulated by Golomb [18], de Bruijn sequences:

- are *balanced*: they contain the same number of 0s and 1s;
- satisfy a *run property*: there are an equal number of contiguous runs of 0s and 1s of the same length in the sequence,
- satisfy a *span-n property*: they contain every distinct length n binary string as a substring.

From our example above for $n = 6$, note that there are exactly 2^{n-1} 0s and 1s respectively; there are 2^{n-2} contiguous runs of 0s and 1s respectively; and by definition, it contains every distinct length n binary string as a substring.

Despite these properties, many de Bruijn sequences display other properties that are far from random. For instance, consider the greedy prefer-1 construction [22]. After starting with an initial seed, successive bits are appended by always trying a 1 first. Only if adding a 1 results in repeating a length n substring will a 0 be appended instead. As one would expect, the resulting de Bruijn sequence (illustrated above for $n = 6$) has a much higher ratio of 1s to 0s at the start of the sequence. One measure that accounts for this is the *discrepancy*,

*Research supported by the *Natural Sciences and Engineering Research Council of Canada* (NSERC) grant RGPIN-2018-04211.

which is defined to be the maximum absolute difference between the number of 0s and 1s in any substring of a given sequence. The discrepancy in our example sequence for $n = 6$ is $|17 - 5| = 12$ as witnessed by the underlined substring. The sequences generated by this prefer-1 approach are known to have discrepancy $\Theta(\frac{2^n \log n}{n})$ [5] with an exact formulation based on the Fibonacci and Lucas numbers [6]. In contrast, the expected discrepancy of a random sequence of length 2^n is $\Theta(2^{n/2}\sqrt{\log n})$ [5]. Some applications in pseudo-random bit generation require de Bruijn sequences that do not have large discrepancy. For example, when used as a carrier signal, a de Bruijn sequence with a large discrepancy causes spectral peaks that could interfere with devices operating at these frequencies [23]. Similar measures described as "balance" and "uniformity" are discussed in [19]. However, they focus only on $n = 2$ and instead vary the size of the alphabet. They explain that de Bruijn sequences with good balance and uniformity are useful in the planning of reaction time experiments [10,28]. De Bruijn sequences with high discrepancy necessarily have bad balance and uniformity.

In this paper, we extend the work initiated by Cooper and Heitsch [5] providing a more complete analysis of discrepancy for a wide variety of de Bruijn sequence constructions. In particular, we:

1. evaluate the discrepancies of an additional 12 efficient/interesting de Bruijn sequence constructions up to $n = 30$,
2. demonstrate de Bruijn sequences constructions that attain the minimum possible discrepancy of $\Theta(n)$, and
3. present a new de Bruijn sequence construction which has discrepancy meeting the asymptotic upper bound of $\Theta(\frac{2^n}{\sqrt{n}})$.

The second result formalizes preliminary work presented in [15]. The asymptotic upper bound achieved in the third result was previously known [4, 11], however no specific construction was known to attain this bound.

The remainder of this paper is presented as follows. We begin with an overview of our experimental results for 13 de Bruijn sequence constructions, including the prefer-1. They are partitioned into four groups which are further analyzed in Sections 2, 3, 4, and 5. We conclude in Section 6 with open problems and future avenues of research.

1.1 The discrepancy of de Bruijn sequence constructions up to $n = 25$

In Table 1 we present exact discrepancies for 13 de Bruijn sequence constructions for values of n between 10 and 25. The results are partitioned into the following four groups based on increasing discrepancy. A larger table up to $n = 30$ is provided in the appendix.

Group 1: Constructions based on the Complementing Cycling Register (CCR) which has feedback function $f(a_1a_2\cdots a_n) = a_1 + 1 \pmod 2$.

Group 2: The greedy prefer-same and prefer-opposite sequences along with a lexicographic composition construction.

Group 3: Constructions based on the Pure Cycling Register (PCR) which has feedback function $f(a_1a_2\cdots a_n) = a_1$. Table 1 also shows a random entry based on taking the average discrepancy of 10000 randomly generated[1] sequences of length 2^n.

Group 4: Two constructions based on joining smaller weight-range cycles.

[1]The sequences were generated in C using the `srand` and `rand` functions.

Details about the constructions from each group are presented in their respective upcoming sections. Implementations for each of these constructions can be found at `http://debruijnsequence.org`. Each construction can generate each symbol in $O(n)$ time bit (or better) using only $O(n)$ space except for the **Pref-same** and **Pref-opposite** algorithms which require $O(2^n)$ space using their greedy construction.

	(Group 1)				(Group 2)		
n	**Huang**	**CCR2**	**CCR3**	**CCR1**	**Pref-same**	**Lex-comp**	**Pref-opposite**
10	12	13	13	16	24	24	27
11	13	14	15	18	29	29	34
12	15	16	16	22	35	35	43
13	16	17	18	23	43	43	52
14	18	19	20	30	48	48	63
15	19	21	21	29	59	59	74
16	21	22	23	36	68	68	87
17	22	24	25	37	79	79	100
18	24	26	26	43	88	88	115
19	25	27	28	43	103	103	130
20	27	29	30	52	114	114	147
21	28	31	31	50	127	127	164
22	30	32	33	59	142	142	183
23	31	34	35	59	155	155	202
24	33	36	36	67	172	172	223
25	35	37	38	66	187	187	244

	(Group 3)					(Group 4)	
n	**PCR4**	**Random**	**PCR3**	**PCR2**	**PCR1**	**Cool-lex**	**Weight-range**
10	29	50	75	101	120	131	131
11	41	71	141	180	222	257	257
12	51	101	248	321	416	468	468
13	70	143	468	587	784	801	930
14	85	203	850	1065	1488	1723	1723
15	110	288	1604	1974	2824	3439	3439
16	175	407	2965	3632	5376	6443	6443
17	246	575	5594	6785	10229	11452	12878
18	326	815	10461	12635	19484	24319	24319
19	462	1157	19765	23746	37107	48629	48629
20	730	1634	37243	44585	71250	92388	92388
21	954	2311	70575	84270	138332	167975	184766
22	1327	3264	133737	159281	268582	352727	352727
23	1820	4565	254322	302449	521553	705443	705443
24	2684	6252	484172	574819	1012795	1352090	1352090
25	3183	9192	924071	1096009	1966813	2496163	2704168

Table 1: Discrepancies of de Bruijn sequence constructions of order n ordered by increasing discrepancy and partitioned into four groups.

1.2 Computing the discrepancy of a de Bruijn sequence

Since de Bruijn sequences have the same number of 0s as 1s, the discrepancy for each of the 2^n linear versions of a given (circular) de Bruijn sequence will be the same. Given a linear version $\mathcal{D}$ of a de Bruijn sequence, the discrepancy of $\mathcal{D}$ can be computed in linear time by keeping track of two values while scanning $\mathcal{D}$ one bit a time from left to right:

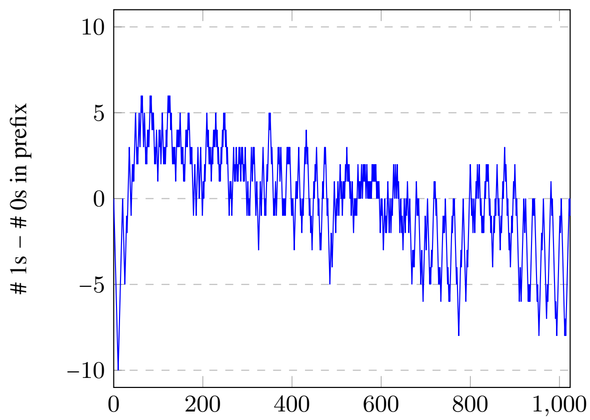
- the maximum value d_1 of the number of 1s minus the number of 0s in any prefix of $\mathcal{D}$, and
- the maximum value d_2 of the number of 0s minus the number of 1s in any prefix of $\mathcal{D}$.

The discrepancy of $\mathcal{D}$ is $d_1 + d_2$.

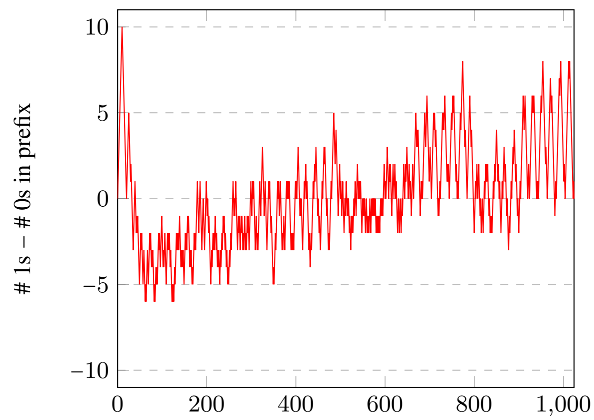
2 Group 1: CCR-based constructions

In this section we consider the four de Bruijn sequence constructions in Group 1 based on the CCR. The sequences generated by the constructions **CCR1**, **CCR2**, and **CCR3** are based on shift-rules presented in [17]. The sequences generated by the **CCR2** and **CCR3** constructions can also be constructed by concatenation approaches [16] described later in this section; the equivalence of the shift-rules to their respective concatenation constructions has been confirmed up to $n = 30$, though no formal proof has been given. The **Huang** construction is a shift-rule based construction in [20]. Since every de Bruijn sequence of order n contains the substring 0^n, a lower bound on discrepancy is clearly n. In this section we prove that two aforementioned concatenation based constructions have discrepancy at most $2n$, and thus attain the smallest possible asymptotic discrepancy of $\Theta(n)$.

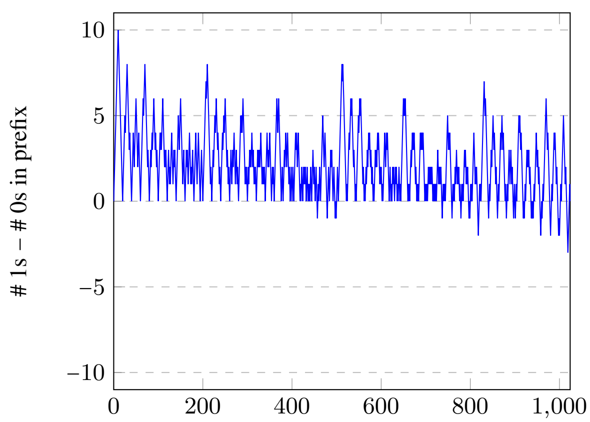
To get a better feel for these four de Bruijn sequence constructions, the following graphs illustrate the running difference between the number of 1s and the number of 0s in each prefix of the given de Bruijn sequence. The examples are for $n = 10$, so the de Bruijn sequences have length $2^{10} = 1024$.



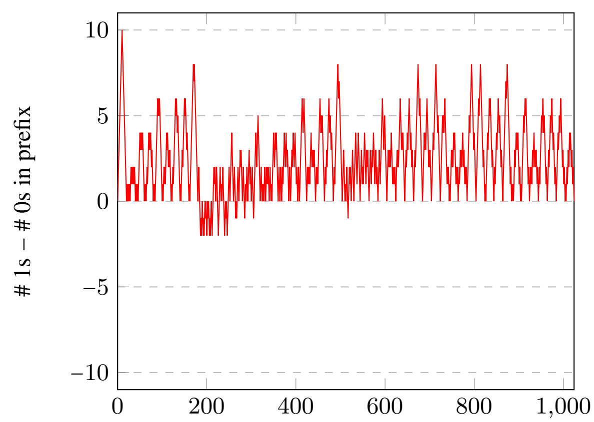
CCR1 sequence for $n = 10$



CCR2 sequence for $n = 10$



CCR3 sequence for $n = 10$



Huang sequence for $n = 10$

Recall that the CCR is a feedback shift register with feedback function $f(a_1a_2\cdots a_n) = a_1 + 1 (\bmod\ 2)$. The CCR partitions $\mathbf{B}(n)$ into equivalence classes of strings called *co-necklaces*. For example, the following four columns are the co-necklace equivalence classes for $n = 5$:

00000	**00010**	**00100**	**01010**
00001	00101	01001	$\underline{10101}$
00011	01011	$\underline{10011}$	
00111	$\underline{10111}$	00110	
01111	01110	01101	
$\underline{11111}$	11101	11011	
11110	11010	10110	
11100	10100	01100	
11000	01000	11001	
10000	10001	10010	

The *periodic reduction* of string α, denoted $pr(\alpha)$ is the smallest prefix β of α such that $\alpha = \beta^t$ for some $t \geq 1$. In [16], the following two de Bruijn sequence constructions **CCR2** and **CCR3** concatenate the periodic reductions of $\alpha\overline{\alpha}$ for given representatives α of each co-necklace equivalence class.

Algo CCR2

1. Let the representative for each co-necklace equivalence class of order n be its lexicographically smallest string.
2. Let $\alpha_1, \alpha_2, \ldots, \alpha_m$ denote these representatives in colex order.
3. **Output**: $pr(\alpha_1\overline{\alpha_1}) \cdot pr(\alpha_2\overline{\alpha_2}) \cdots pr(\alpha_m\overline{\alpha_m})$.

For $n = 5$, the representatives for this algorithm are the bolded strings in the equivalence classes above and **Algo CCR2** produces:

$$0000011111 \cdot 0010011011 \cdot 0001011101 \cdot 01.$$

Algo CCR3

1. Let the representative for each co-necklace equivalence class of order n be the string obtained by taking the lexicographically smallest string, removing its largest prefix of the form 0^j, and then appending 1^j to the end.
2. Let $\alpha_1, \alpha_2, \ldots, \alpha_m$ denote these representatives in lexicographic order.
3. **Output**: $pr(\alpha_1\overline{\alpha_1}) \cdot pr(\alpha_2\overline{\alpha_2}) \cdots pr(\alpha_m\overline{\alpha_m})$.

For $n = 5$, the representatives for this algorithm are the underlined strings in the equivalence classes above and **Algo CCR3** produces:

$$1001101100 \cdot 10 \cdot 1011101000 \cdot 1111100000.$$

We now prove that the discrepancy resulting from these two de Bruijn sequence constructions is at most $2n$.

Lemma 2.1 *Consider a sequence of binary strings $\alpha_1, \alpha_2, \cdots, \alpha_m$ where each α_i has the same number of 0s as 1s and has discrepancy at most n. Then $\mathcal{S} = \alpha_1\alpha_2\cdots\alpha_m$ has discrepancy at most $2n$.*

Proof. Consider a shortest substring of $\mathcal{S}$ of the form $\alpha_i\alpha_{i+1}\cdots\alpha_j$ that has the same discrepancy as $\mathcal{S}$. Its discrepancy will be the same as that of $\alpha_i\alpha_j$ which gives an upper bound of $2n$. □

Theorem 2.2 *The de Bruijn sequences constructed by* **Algo CCR2** *and* **Algo CCR3** *have discrepancy at most $2n$.*

Proof. Given a length n binary string α, $\alpha\overline{\alpha}$ has the same number of 0s and 1s and has discrepancy at most n. These properties also hold for $pr(\alpha\overline{\alpha})$ by definition of the periodic reduction. Thus, by Lemma 2.1, the sequences constructed by **Algo CCR2** and **Algo CCR3** have discrepancy at most $2n$. □

Interestingly, from Table 1, these two concatenation-based constructions do not demonstrate the smallest discrepancy for $n \leq 30$. The construction by Huang [20], which is based on a cycle-joining approach, demonstrates slightly smaller discrepancy. In particular the author states:

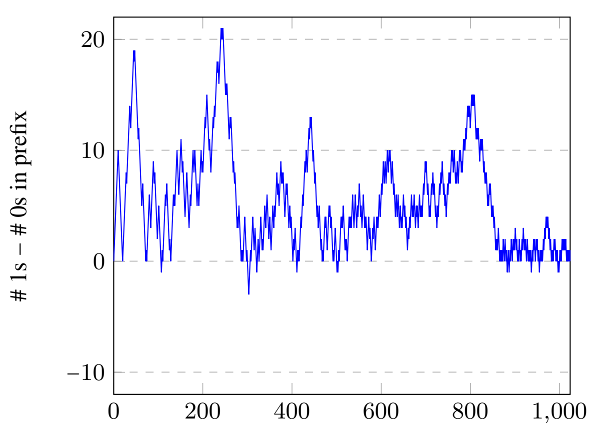
> *"It seems clear that the sequences produced by our algorithm have a relatively good characteristic of local 0-1 balance in comparison with the ones produced by the 'prefer one' algorithm."*

So the author indicates that their construction may have small discrepancy, however no analysis is provided.

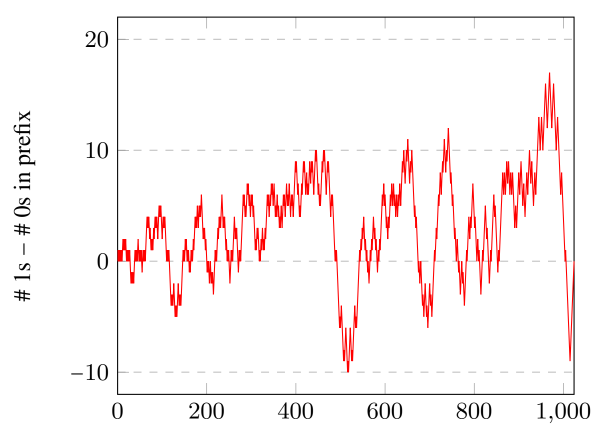
3 Group 2: Prefer-same, prefer-opposite, and lexicographic compositions

In this section we consider the three de Bruijn sequence constructions in Group 2. The **Pref-same** [3, 9, 12] and the **Pref-opposite** [2] are greedy constructions based on the last bit of the sequence as it is constructed. They have the downside of requiring an exponential amount of memory. The **Lex-comp** construction [13] is obtained by concatenating lexicographic compositions. Its construction was an attempt to efficiently generate the sequence generated by the **Pref-same** approach; it was conjectured to be the same for a very long prefix. Observe that it attains the same discrepancy as the **Pref-same** for all values of n tested.

To get a better feel for the two greedy de Bruijn sequence constructions, the following graphs illustrate the running difference between the number of 1s and the number of 0s in each prefix of the given de Bruijn sequence. The examples are for $n = 10$, so the de Bruijn sequences have length $2^{10} = 1024$.



Pref-same sequence for $n = 10$



Pref-opp sequence for $n = 10$

In the following table we study some experimental results for the **Pref-same** construction. In particular, for $10 \le n \le 25$ we compute the maximum difference between the number of 1s and the number of 0s along with the maximum difference between the number 0s and the number of 1s, over all prefixes of each **Pref-same** de Bruijn sequence of order n. Adding these two values together, we get the discrepancies shown in Table 1.

n	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
max(#1s − #0s)	21	26	31	36	43	50	57	64	73	82	91	100	111	122	133	144
max(#0s − #1s)	3	3	4	7	5	9	11	15	15	21	23	27	31	33	39	43
discrepancy	24	29	35	43	48	59	68	79	88	103	114	127	142	155	172	187

Interestingly, the values in the row max(#1s − #0s) are equivalent to the known sequence A008811 in the Online Encyclopedia of Integer Sequences (OEIS) [1] offset by four positions. The sequence enumerates the "Expansion of $x(1+x^4)/((1-x)^2(1-x^4))$" and the provided formula demonstrates that each value is $\Theta(n^2)$. More specifically the values match the sequence for $6 \le n \le 30$. This leads to the following conjecture.

Conjecture 3.1 *The de Bruijn sequences constructed by the* **Pref-same** *and* **Lex-comp** *algorithms have discrepancy* $\Theta(n^2)$.

A similar analysis was performed for sequences generated by the **Pref-opposite** construction.

n	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
max(#1s − #0s)	10	13	17	21	26	31	37	43	50	57	65	73	82	91	101	111
max(#0s − #1s)	17	21	26	31	37	43	50	57	65	73	82	91	101	111	122	133
discrepancy	27	34	43	52	63	74	87	100	115	130	147	164	183	202	223	244

Remarkably, observe that the two middle rows are a shift from each other by two positions. Just as interesting, the sequences also correspond to a known sequence in OEIS [1], namely A033638. Specifically, the row max(#1s− #0s) corresponds to this sequence shifted by four positions. The sequence does not match for $n < 10$, but we have verified it matches for $10 \le n \le 30$. The sequence corresponds to "quarter squares plus 1", and by applying the appropriate shifts, the discrepancy for the **Prefer-opposite** sequence of order n, for $10 \le n \le 30$ is given by:

$$\lfloor \frac{(n-4)^2}{4} \rfloor + \lfloor \frac{(n-2)^2}{4} \rfloor + 2.$$

This leads to the following conjecture.

Conjecture 3.2 *The de Bruijn sequence constructed by the* **Pref-opposite** *algorithm has discrepancy* $\Theta(n^2)$.

We conclude this section with an observation regarding the **Pref-opposite** de Bruijn sequence: For $2 \le n \le 25$, each sequence has the following suffix where $j = \lceil n/3 \rceil$:

$$0^j 1^{n-j} \cdot 0^{j-1} 1^{n-j+1} \cdots 01^{n-1} \cdot 10^{n-1}.$$

For example, when $n = 10$, the **Pref-opposite** de Bruijn sequence has suffix

$$0000001111 \cdot 00000\underline{11111 \cdot 0000111111 \cdot 0001111111 \cdot 0011111111 \cdot 0111111111 \cdot 1}000000000,$$

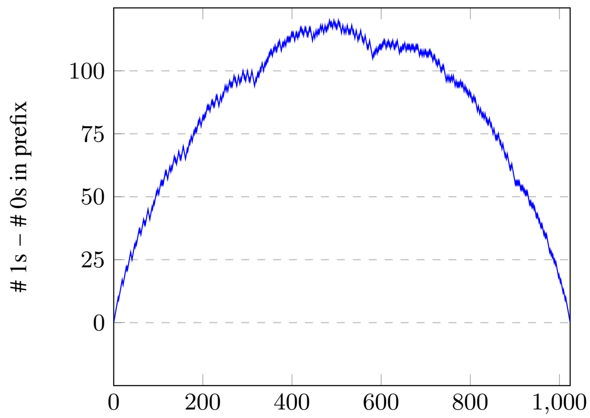
and the underline section has $5+6+7+8+10$ ones and $4+3+2+1$ zeros. A slight rearrangement gives a lower bound of $(5-1)+(6-2)+(7-3)+(8-4)+10 = 4 \cdot 4 + 10 = 26$ for the discrepancy of the sequence. The actual discrepancy is 27. More generally, if this suffix is indeed a suffix for each **Pref-opposite** de Bruijn sequence, then a lower bound on its discrepancy will be

$$(\lceil n/2 \rceil - 1)(\lfloor n/2 \rfloor - 1) + n \;=\; \Omega(n^2).$$

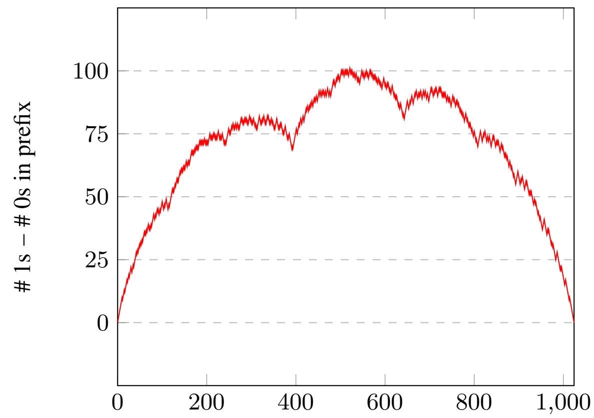
4 PCR-based constructions

In this section we consider the four de Bruijn sequence constructions in Group 3 based on the PCR. The constructions **PCR1**, **PCR2**, **PCR3**, and **PCR4** are based on shift-rules presented in [17]. The sequences generated by **PCR1** are the same as the ones generated by the prefer-0 greedy construction (the complement of the prefer-1) and a very efficient necklace concatenation construction based on lexicographic order [14]. The sequences generated by **PCR2** are the same as the ones generated by a more efficient necklace concatenation construction based on colex order [7, 8]. The **PCR3** is based on a general approach in [21] and revisited in [27].

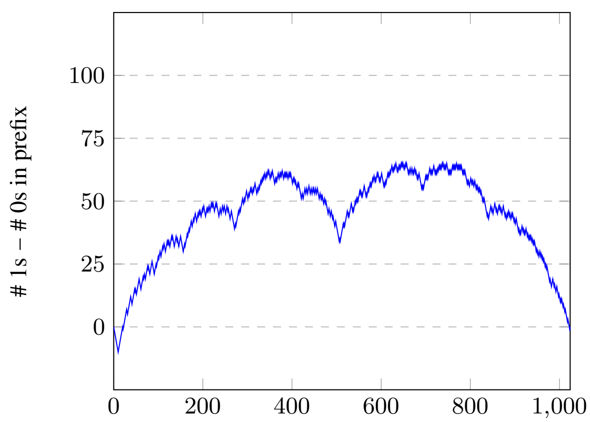
To get a better feel for these four de Bruijn sequence constructions, the following graphs illustrate the running difference between the number of 1s and the number of 0s in each prefix of the given de Bruijn sequence. The examples are for $n = 10$, so the de Bruijn sequences have length $2^{10} = 1024$.



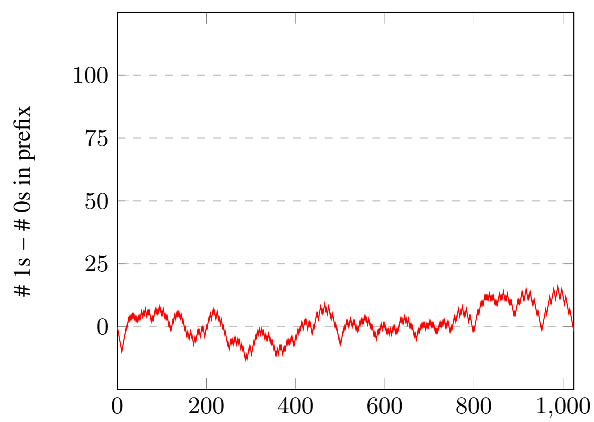
PCR1 sequence for $n = 10$



PCR2 sequence for $n = 10$



PCR3 sequence for $n = 10$



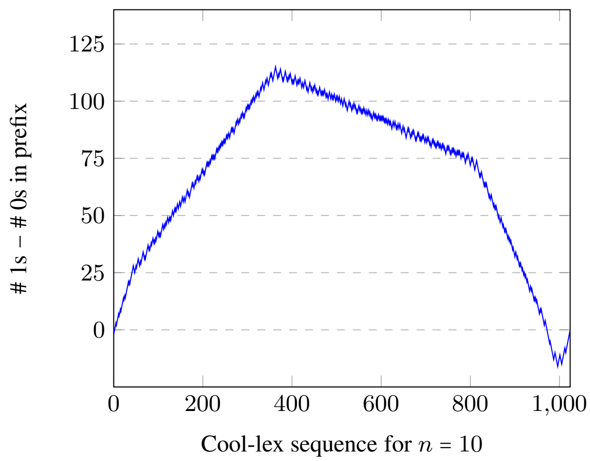
PCR4 sequence for $n = 10$

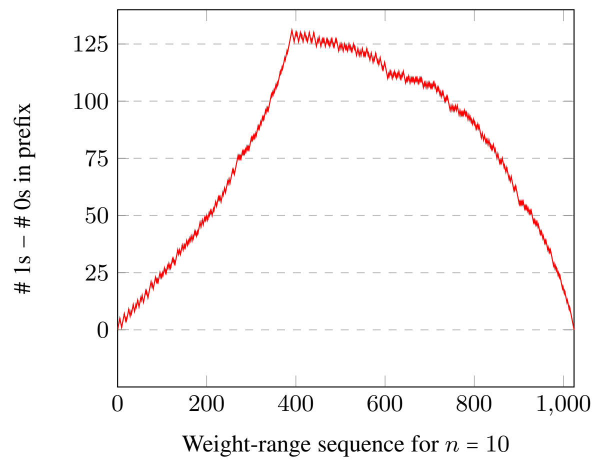
The discrepancy for the sequence generated by the **PCR1** construction has already been studied in [5] where they show that the discrepancy is $\Theta(\frac{2^n \log n}{n})$. The sequences generated by the **PCR2** and **PCR3** constructions appear to have a similar growth trajectories. More interesting are the sequences generated by the **PCR4** construction that, from Table 1, appear to have discrepancy that is closest to that of a random string. It would be interesting to do a more detailed investigation of this construction, which is based on a very simple successor rule.

5 Weight range constructions

In this section we consider two de Bruijn sequence constructions which are based on joining smaller cycles based on weight (number of 1s). The **Cool-lex** construction [24], is a concatenation approach which is based on creating underlying cycles which contain all strings with weights d and $d+1$ given $0 \leq d < n$. Then, appropriate such cycles can be joined together to obtain a de Bruijn sequence [25]. By the nature of how the cycles are joined, the first half of the resulting de Bruijn sequence contains mostly length n substrings of weight less than or equal to $n/2$. Similarly, the latter half mostly contains length n substrings with weight greater than or equal to $n/2$. Thus, as one would expect, the resulting de Bruijn sequence has a very large discrepancy. The **Weight-range** construction is a new construction presented in this section which we prove attains the maximal possible asymptotic discrepancy of $\Theta(2^n/\sqrt{n})$ [4, 11].

To get a better feel for these two de Bruijn sequence constructions, the following graphs illustrate the running difference between the number of 1s and the number of 0s in each prefix of the given de Bruijn sequence. The examples are for $n = 10$, so the de Bruijn sequences have length $2^{10} = 1024$.





Notice that if we had shifted the starting position of the **Cool-lex** sequence the profile of the graph would be very similar to that the **Weight-range** sequence. In fact, the discrepancies of the two sequences are the same except when $n \bmod 4 \equiv 1$ (see Table 1). This will be discussed more after we present the **Weight-range** construction.

A *minimum weight de Bruijn sequence* is a cyclic sequence that contains each binary string of length n with weight at least w exactly once. A *maximum weight de Bruijn sequence* is defined similarly where the weight of each string is at most w. A construction for the former sequence is given in [26]; it is constructed by concatenating the periodic reduction of each necklace of weight $\geq w$ when the necklaces are listed in lexicographic order. Let the resulting sequence be denoted by $\mathcal{D}_w(n)$.

Remark 5.1 *For any $w < n$, $\mathcal{D}_w(n)$ begins with $0^{n-w}1^w$ and ends with 1^n.*

By complementing the bits in $\mathcal{D}_w(n)$, we obtain a maximum weight de Bruijn sequence with weight at most $n - w$. Denote this sequence by $\overline{\mathcal{D}}_w(n)$. From the previous remark, it begins with $1^{n-w}0^w$ and ends with 0^n.

Example 1 The necklaces of length 6 with weight $w \geq 3$ in lexicographic order are:

$$000111, 001011, 001101, 001111, 010101, 010111, 011011, 011111, 111111.$$

Concatenating together their periodic reductions we obtain the minimum weight de Bruijn sequence $\mathcal{D}_3(6)$.

$$000111 \cdot 001011 \cdot 001101 \cdot 001111 \cdot 01 \cdot 010111 \cdot 011 \cdot 011111 \cdot 1$$

As further examples,

$$\mathcal{D}_4(6) = 001111 \cdot 010111 \cdot 011 \cdot 011111 \cdot 1$$

and

$$\overline{\mathcal{D}}_4(6) = 110000 \cdot 101000 \cdot 100 \cdot 100000 \cdot 0.$$

From the above example observe that:

- $\mathcal{D}_3(6)$ contains all binary strings of length 6 with weight greater than or equal to 3,
- $\overline{\mathcal{D}}_4(6)$ contains all binary strings of length 6 with weight less than or equal to 2,
- The length $n-1$ prefix of $\overline{\mathcal{D}}_4(6)$, namely 11000, appears in the wraparound of $\mathcal{D}_3(6)$.

Let $\mathcal{D}^r_w(n)$ denote the sequence $\mathcal{D}_w(n)$ with the suffix 1^{w-1} rotated to the front. Then by applying the Gluing Lemma [25], the following is a de Bruijn sequence of order 6:

$$\underbrace{\underline{11000}010100010010000000}_{\overline{\mathcal{D}}_4(6)} \cdot \underbrace{\underline{11000}1110010110011010011110101011101101111}_{\mathcal{D}^r_3(6)}\ .$$

Applying this strategy more generally, let $\mathcal{DB}_{max}(n)$ denote the de Bruijn sequence obtained by joining two such smaller cycles.

Weight-range construction

$$\mathcal{DB}_{max}(n) = \overline{\mathcal{D}}_w(n) \cdot \mathcal{D}^r_{w'}(n),$$

where $w = \lfloor n/2 \rfloor + 1$ and $w' = \lceil n/2 \rceil$.

A complete C implementation to construct $\mathcal{DB}_{max}(n)$ is given in the Appendix[2].

The following technical lemma leads to a lower bound for the discrepancy of $\mathcal{DB}_{max}(n)$.

Lemma 5.2 *A maximum weight de Bruijn sequence of order n and maximum weight w has $\binom{n-1}{w}$ more 0s than 1s.*

Proof. By definition, a maximum weight de Bruijn sequence of order n and maximum weight w contains every binary string of length n with weight at most w as a substring exactly once. Since each bit in this

[2]It is also available at `http:\debruijnsequence.org`.

sequence belongs to n different strings the total number of 1s in the sequence is

$$\begin{aligned} ones &= \frac{1}{n}\sum_{d=0}^{w} d\binom{n}{d} \\ &= \frac{0}{n}\binom{n}{0} + \frac{1}{n}\binom{n}{1} + \frac{2}{n}\binom{n}{2} + \cdots + \frac{w}{n}\binom{n}{w} \\ &= 0 + \binom{n-1}{0} + \binom{n-1}{1} + \cdots + \binom{n-1}{w-1}, \end{aligned}$$

and the total number of 0s is

$$\begin{aligned} zeros &= \frac{1}{n}\sum_{d=0}^{w} (n-d)\binom{n}{d} \\ &= \frac{n}{n}\binom{n}{0} + \frac{n-1}{n}\binom{n}{1} + \frac{n-2}{n}\binom{n}{2} + \cdots + \frac{n-w}{n}\binom{n}{w} \\ &= \binom{n-1}{0} + \binom{n-1}{1} + \binom{n-1}{2} + \cdots + \binom{n-1}{w}. \end{aligned}$$

Thus $zeros - ones = \binom{n-1}{w}$. □

Theorem 5.3 *The de Bruijn sequence $\mathcal{DB}_{max}(n)$ has discrepancy at least $\binom{n-1}{\lfloor n/2 \rfloor} + \lfloor \frac{n}{2} \rfloor$.*

Proof. Let $w = \lfloor n/2 \rfloor + 1$ and $w' = \lceil n/2 \rceil$. Recall that $\overline{\mathcal{D}}_w(n)$ is a maximum weight de Bruijn sequence with maximum weight $n - w$. Thus, by Lemma 5.2, it has $\binom{n-1}{n-w} = \binom{n-1}{n-(\lfloor n/2 \rfloor+1)} = \binom{n-1}{\lfloor n/2 \rfloor}$ more 0s than 1s. Consider $\overline{\mathcal{D}}_w(n)$ with its prefix of 1^{n-w} removed. The resulting string, which is a substring of $\mathcal{DB}_{max}(n)$, has $\binom{n-1}{\lfloor n/2 \rfloor} + (n-w)$ more 0s than 1s. When n is odd we have $n-w = n - \lfloor n/2 \rfloor - 1 = \lfloor \frac{n}{2} \rfloor$ and thus $\mathcal{DB}_{max}(n)$ has discrepancy at least $\binom{n-1}{\lfloor n/2 \rfloor} + \lfloor \frac{n}{2} \rfloor$. When n is even, we additionally add the length $n-1$ prefix of $\mathcal{D}^r_{w'}(n)$ which has more 0s than 1s (exactly one more). Since $n - w + 1 = n - (\lfloor n/2 \rfloor - 1) + 1 = \lfloor \frac{n}{2} \rfloor$ (when n is even) this again means that $\mathcal{DB}_{max}(n)$ has discrepancy at least $\binom{n-1}{\lfloor n/2 \rfloor} + \lfloor \frac{n}{2} \rfloor$. □

By applying Stirling's approximation to $\binom{n-1}{\lfloor n/2 \rfloor}$ we obtain the following corollary.

Corollary 5.4 *The discrepancy of the de Bruijn sequence $\mathcal{DB}_{max}(n)$ attains the asymptotic upper bound of $\Theta(\frac{2^n}{\sqrt{n}})$.*

Observe from Table 1 that the discrepancy of $\mathcal{DB}_{max}(n)$ is exactly $\binom{n-1}{\lfloor n/2 \rfloor} + \lfloor \frac{n}{2} \rfloor$ for $10 \le n \le 25$. This leads to the following conjecture.

Conjecture 5.5 *The de Bruijn sequence $\mathcal{DB}_{max}(n)$ has discrepancy equal to $\binom{n-1}{\lfloor n/2 \rfloor} + \lfloor \frac{n}{2} \rfloor$, and moreover, it is the maximal possible discrepancy over all de Bruijn sequences of order n.*

As noted earlier, the discrepancy of the **cool-lex** construction matches the discrepancy for the **weight-range** construction for $10 \le n \le 25$, except for when $n \bmod 4 \equiv 1$ (see Table 1). As illustration, the **cool-lex** construction first constructs cycles of the following weights for $n = 6, 7, 8, 9$:

- $n = 6$: (0,1,2), (3,4), (5,6)
- $n = 7$: (0,1), (2,3), (4,5), (6,7)

- $n = 8$: (0,1,2), (3,4), (5,6), (7,8)
- $n = 9$: (0,1), (2,3), (4,5), (6,7), (8,9)

before joining them together one at a time. Note when $n = 9$, strings with weights 4 and 5 are grouped together before the smaller cycles are joined together. This causes a reduction in the discrepancy compared to the **weight-range** construction. It is possible, however, to tweak the **cool-lex** implementation so the discrepancies are equivalent. For instance for $n = 9$, the smaller cycles with weights $(0,1,2),(3,4),(5,6),(7,8,9)$ could be joined together instead.

6 Future directions and open problems

In this paper, we investigated the discrepancies of 13 de Bruijn sequence constructions. We proved that two constructions attain the lower bound of $\Theta(n)$ and presented one new construction that attains the upper bound of $\Theta(\frac{2^n}{\sqrt{n}})$. It remains an interesting problem to demonstrate a construction with discrepancy that is close to that of a random stream of bits of the same length. Some avenues of future research include the following.

1. Simplify the description of the **Huang** construction [20]. Does it have the smallest discrepancy over all de Bruijn sequences?
2. Answer the conjectures regarding the discrepancies for the greedy **Pref-same** and **Pref-opposite** constructions (Conjecture 3.1 and Conjecture 3.2).
3. Analyze the discrepancy of **PCR4** which had discrepancy closest to one we might expect from a random stream of bits.
4. Determine whether or not the maximal discrepancy of any de Bruijn sequence is $\binom{n-1}{\lfloor n/2 \rfloor} + \lfloor \frac{n}{2} \rfloor$ (Conjecture 5.5).
5. Generalize the investigation of disrepancy to de Bruijn sequences over an arbitrary alphabet size k.
6. Study the distribution of discrepancy over all possible de Bruijn sequences.

References

[1] OEIS Foundation Inc. (2020), The On-Line Encyclopedia of Integer Sequences, http://oeis.org.

[2] A. Alhakim. A simple combinatorial algorithm for de Bruijn sequences. *The American Mathematical Monthly*, 117(8):728–732, 2010.

[3] A. Alhakim, E. Sala, and J. Sawada. Revisiting the prefer-same and prefer-opposite de Bruijn sequence constructions. *Theoretical Computer Science (to appear)*, 2020.

[4] S. R. Blackburn and I. E. Shparlinski. Character sums and nonlinear recurrence sequences. *Discrete Math.*, 306(12):1126–1131, June 2006.

[5] J. Cooper and C. Heitsch. The discrepancy of the lex-least de Bruijn sequence. *Discrete Mathematics*, 310:1152–1159, 2010.

[6] J. Cooper and C. E. Heitsch. Generalized Fibonacci recurrences and the lex-least de Bruijn sequence. *Advances in Applied Mathematics*, 50:465–473, 2010.

[7] P. B. Dragon, O. I. Hernandez, J. Sawada, A. Williams, and D. Wong. Constructing de Bruijn sequences with co-lexicographic order: The k-ary Grandmama sequence. *submitted manuscript*, 2017.

[8] P. B. Dragon, O. I. Hernandez, and A. Williams. The grandmama de Bruijn sequence for binary strings. In *Proceedings of LATIN 2016: Theoretical Informatics: 12th Latin American Symposium, Ensenada, Mexico*, pages 347–361. Springer Berlin Heidelberg, 2016.

[9] C. Eldert, H. Gray, H. Gurk, and M. Rubinoff. Shifting counters. *AIEE Trans.*, 77:70–74, 1958.

[10] P. L. Emerson and R. D. Tobias. Computer program for quasi-random stimulus sequences with equal transition frequencies. *Behavior Research Methods, Instruments, & Computers*, 27(1):88–98, Mar 1995.

[11] G. Everest, A. J. Van Der Poorten, I. E. Shparlinski, and T. Ward. *Recurrence sequences*, volume 104. AMS Mathematical Surveys and Monographs, 2003.

[12] H. Fredricksen. A survey of full length nonlinear shift register cycle algorithms. *Siam Review*, 24(2):195–221, 1982.

[13] H. Fredricksen and I. Kessler. Lexicographic compositions and de Bruijn sequences. *J. Combin. Theory Ser. A*, 22(1):17 – 30, 1977.

[14] H. Fredricksen and J. Maiorana. Necklaces of beads in k colors and k-ary de Bruijn sequences. *Discrete Math.*, 23:207–210, 1978.

[15] D. Gabric and J. Sawada. A de Bruijn sequence construction by concatenating cycles of the complemented cycling register. In *Combinatorics on Words - 11th International Conference, WORDS 2017, Montréal, QC, Canada, September 11-15, 2017, Proceedings*, pages 49–58, 2017.

[16] D. Gabric and J. Sawada. Constructing de Bruijn sequences by concatenating smaller universal cycles. *Theoretical Computer Science*, 743:12–22, 2018.

[17] D. Gabric, J. Sawada, A. Williams, and D. Wong. A framework for constructing de Bruijn sequences via simple successor rules. *Discrete Mathematics*, 241(11):2977–2987, 2018.

[18] S. Golomb. On the classification of balanced binary sequences of period $2^n - 1$(corresp.). *IEEE Transactions on Information Theory*, 26(6):730–732, November 1980.

[19] Y. Hsieh, H. Sohn, and D. Bricker. Generating (n,2) de Bruijn sequences with some balance and uniformity properties. *Ars Combinatoria*, 72:277–286, 07 2004.

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

[21] C. J. A. Jansen, W. G. Franx, and D. E. Boekee. An efficient algorithm for the generation of DeBruijn cycles. *IEEE Transactions on Information Theory*, 37(5):1475–1478, Sep 1991.

[22] M. H. Martin. A problem in arrangements. *Bull. Amer. Math. Soc.*, 40(12):859–864, 1934.

[23] A. A. Philippakis, A. M. Qureshi, M. F. Berger, and M. L. Bulyk. Design of compact, universal DNA microarrays for protein binding microarray experiments. In T. Speed and H. Huang, editors, *Research in Computational Molecular Biology*, pages 430–443, Berlin, Heidelberg, 2007. Springer Berlin Heidelberg.

[24] F. Ruskey, J. Sawada, and A. Williams. De Bruijn sequences for fixed-weight binary strings. *SIAM Journal on Discrete Mathematics*, 26(2):605–617, 2012.

[25] J. Sawada, A. Williams, and D. Wong. Universal cycles for weight-range binary strings. In *Combinatorial Algorithms - 24th International Workshop, IWOCA 2013, Rouen, France, July 10-12, 2013, Revised Selected Papers*, pages 388–401, 2013.

[26] J. Sawada, A. Williams, and D. Wong. The lexicographically smallest universal cycle for binary strings with minimum specified weight. *Journal of Discrete Algorithms*, 28:31–40, 2014.

[27] J. Sawada, A. Williams, and D. Wong. A surprisingly simple de Bruijn sequence construction. *Discrete Math.*, 339:127–131, 2016.

[28] H.-S. Sohn, D. L. Bricker, J. R. Simon, and Y. Hsieh. Optimal sequences of trials for balancing practice and repetition effects. *Behavior Research Methods, Instruments, & Computers*, 29(4):574–581, Dec 1997.

A Table of discrepancies

n	Huang	(Group 1) CCR2	CCR3	CCR1	Pref-same	(Group 2) Lex-comp	Pref-opposite
10	12	13	13	16	24	24	27
11	13	14	15	18	29	29	34
12	15	16	16	22	35	35	43
13	16	17	18	23	43	43	52
14	18	19	20	30	48	48	63
15	19	21	21	29	59	59	74
16	21	22	23	36	68	68	87
17	22	24	25	37	79	79	100
18	24	26	26	43	88	88	115
19	25	27	28	43	103	103	130
20	27	29	30	52	114	114	147
21	28	31	31	50	127	127	164
22	30	32	33	59	142	142	183
23	31	34	35	59	155	155	202
24	33	36	36	67	172	172	223
25	35	37	38	66	187	187	244
26	36	39	40	77	208	208	267
27	38	41	42	74	224	224	290
28	40	43	43	85	246	246	315
29	41	44	45	84	264	264	340
30	43	46	47	94	286	286	367

n	PCR4	Random	(Group 3) PCR3	PCR2	PCR1	(Group 4) Cool-lex	Weight-range
10	29	50	75	101	120	131	131
11	41	71	141	180	222	257	257
12	51	101	248	321	416	468	468
13	70	143	468	587	784	801	930
14	85	203	850	1065	1488	1723	1723
15	110	288	1604	1974	2824	3439	3439
16	175	407	2965	3632	5376	6443	6443
17	246	575	5594	6785	10229	11452	12878
18	326	815	10461	12635	19484	24319	24319
19	462	1157	19765	23746	37107	48629	48629
20	730	1634	37243	44585	71250	92388	92388
21	954	2311	70575	84270	138332	167975	184766
22	1327	3264	133737	159281	268582	352727	352727
23	1820	4565	254322	302449	521553	705443	705443
24	2684	6252	484172	574819	1012795	1352090	1352090
25	3183	9192	924071	1096009	1966813	2496163	2704168
26	4108	13074	1766284	2092284	3819605	5200313	5200313
27	5604	17933	3382851	4004050	7453523	10400613	10400613
28	7629	22672	6488970	7672443	14544826	20058314	20058314
29	10433	34591	12468181	14730243	28382864	37442182	40116614
30	13637	57357	23991972	28316271	55421919	77558775	77558775