Multiple de Bruijn Sequences and the Cross-Join Method

Abbas Alhakim* and Janusz Szmidt**

*American University of Beirut, Lebanon. aa145@aub.edu.lb
**Military Communication Institute, Poland. j.szmidt@wil.waw.pl

Abstract

We show how to construct binary multi de Bruijn sequences using the cross-join method. We experimentally confirm that some multi de Bruijn sequences can be generated by Galois Nonlinear Feedback Shift Registers.

1 Introduction

De Bruijn sequences have been investigated for decades [6, 2, 7]. It is known that binary de Bruijn sequences can be generated by Nonlinear Feedback Shift Registers (*NLFSR*) [8]. Knowing a de Bruijn sequence one can apply the cross-join method to construct new de Bruijn sequences [8, 7, 9, 3, 12]. In papers [11, 1] we proved that the cross-join method generates all de Bruijn sequences of given order. In [1] an algorithm was explicitly given that begins with a de Bruijn sequence from a finite alphabet and outputs a Hamiltonian path in the corresponding cross-join graph. The paper [13] generalizes the notion of de Bruijn sequences to *multi de Bruijn sequences*, where patterns of fixed length appear m times ($m = 1$ for ordinary de Bruijn sequences). Multi de Bruijn sequences over some alphabets appear in biological investigations [10].

We prove that all binary multi de Bruijn sequences can be generated starting from one such sequence by using the cross-join method. The proof is non-constructive and there are needed methods to construct multi de Bruijn sequences. We implemented this method for the case of multi de Bruijn sequences of the type $C(2, 2, 3)$ (binary multi de Bruijn sequences of order 3 with multiplicity 2 and patterns of length 3). Galois *NLFSRs* were considered in papers [4, 5]. We confirmed experimentally that some of sequences of type $C(2, 2, 3)$ can be generated by Galois *NLFSRs* listed in [5]. In fact, they are modified sequences where one of the patterns has a lower multiplicity. It is an open problem whether all binary multi de Bruijn sequences can be generated by suitable Galois *NLFSRs*.

2 Multi de Bruijn sequences

We introduce multi de Bruijn sequences following Tesler's paper [13]. Let Ω be a totally ordered alphabet of size $q \geq 1$. A linear sequence is an ordinary sequence of elements of Ω denoted $a_1a_2 \dots a_n$. Define the *cyclic shift* of a linear sequence by $\rho(a_1a_2 \dots a_n) = a_na_1 \dots a_{n-1}$. In a cyclic sequence, we treat all rotations of a given linear sequence as equivalent. A *k-mer* is a sequence of length k over Ω. The set of all *k-mers* over Ω is Ω^k. A *cyclic de Bruijn sequence* is a cyclic sequence over alphabet Ω in which all *k-mers* occur exactly once. The length of such a sequence is $N = q^k$.

Definition 2.1 *A cyclic multi de Bruijn sequence is a cyclic sequence over alphabet Ω of size k in which each k-mer occurs exactly m-times with $m, q, k \geq 1$. k is the order of the sequence.*

Let $C(m, q, k)$ denote the set of all such sequences. The length of such a sequence is $N = mq^k$, since each of the q^k *k-mers* accounts for m-starting positions. Tesler [13] derived the formula for the cardinality of $C(m, q, k)$. In the following we consider multi de Bruijn sequences over binary alphabet $\Omega = \{0, 1\}$.

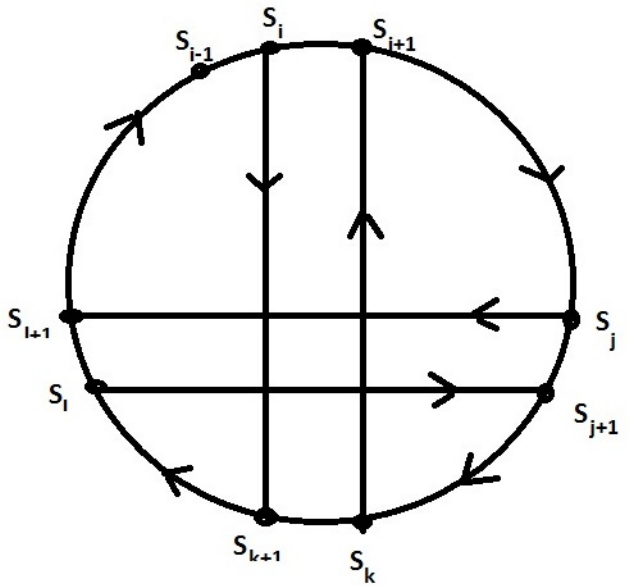

Figure 1: The geometric depict of the cross-join method.

Definition 2.2 *Let a sequence* $(x_i) \in \mathcal{C}(m, 2, k)$ *be represented as a sequence of its states* (S_i)*, where each state is a k-mer* $S_i = (x_i, x_{i+1}, \ldots, x_{i+k-1})$*. It is conjugate to a state* $S_j = (x_j, x_{j+1}, \ldots, x_{j+k-1})$ *if* $x_i = x_j + 1$*. We denote this* $S_i = \hat{S}_j$*. The state* S_i *is a companion of the state* S_j *if* $x_{i+k-1} = x_{j+k-1} + 1$*.*

Definition 2.3 *Let a multi de Bruijn sequence* (x_i) *be considered as a cyclic sequence and represented as a sequence of states* (S_i)*. Then four succeeding states* $(S_i, S_j, \hat{S}_i, \hat{S}_j)$ *are called the* cross-join pair *for the sequence* (x_i)*.*

Definition 2.4 *Let* $(x_i) \in \mathcal{C}(m, 2, k)$ *and* $(S_i, S_j, \hat{S}_i, \hat{S}_j)$ *be its cross-join pair. We construct a new multi de Bruijn sequence* (y_i) *by swapping the successors of* S_i *and* $\hat{S}_i$ *and the successors of* S_j *and* $\hat{S}_j$*. That is, by going from* S_i *to the successor of* $\hat{S}_i$*, then from* $\hat{S}_j$ *to the successor of* S_j *and so on until closing the cycle. This construction is called the cross-join method.*

To be more precise, let us denote $\hat{S}_i = S_k$ and $\hat{S}_j = S_l$. Then the original sequence has states that proceed as:

$$S_i, S_{i+1}, \ldots, S_j, S_{j+1}, \ldots, S_k, S_{k+1}, \ldots, S_l, S_{l+1}, \ldots; S_{i-1}.$$

After the cross-join operation the modified sequence has states that proceed as:

$$S_i, S_{k+1}, \ldots, S_l, S_{j+1}, \ldots, S_k, S_{i+1}, \ldots, S_j, S_{l+1}, \ldots, S_{i-1}.$$

The conjugate pair of states $S_i, \hat{S}_i$ splits the full cycle into two shorter cycles after interchanging their successors. Then the states $S_j, \hat{S}_j$ are on different cycles and after interchanging their successors we obtain a new de Bruijn cycle (see Figure 1).

Definition 2.5 *Let* $(x_i), (y_i)$ *be two sequences from* $\mathcal{C}(m, 2, k)$*. The length of the sequences is* $N = m2^k$*. We take the least lexicographical representatives of both sequences and consider the length* L *of the longest common initial path of these sequences*

$$(x_1, x_2, \cdots, x_L, \cdots, x_N), \quad (x_1, x_2, \cdots, x_L, y_{L+1}, \cdots, y_N).$$

We define the function (pseudo-distance) of the sequences as $d(x,y) = N - L$.

Lemma 2.6 *The function* $d(x,y)$ *has the properties:*

- $d(x,x) = 0$ *for all* $x \in \mathcal{C}(m,2,k)$.
- $d(x,y) = d(y,x)$ *for all* $x, y \in \mathcal{C}(m,2,k)$.

There are examples of three multi de Bruijn sequences which are concatenation of de Bruijn sequences of lower order for which the triangle inequality is not satisfied. It seems that when we exclude such cases then the triangle inequality is satisfied on the set of remaining multi de Bruijn sequences of a given order.

Definition 2.7 *Let* x *and* y *be two distinct multi de Bruijn sequences. We say that* y *is a neighbour of* x *if* y *can be obtained from* x *by applying a sequence of cross-join operations.*

Lemma 2.8 *Let* $x = (x_i)$ *and* $y = (y_i)$ *be two distinct multi de Bruijn sequences from the space* $\mathcal{C}(m,2,k)$. *Then there exists a multi de Bruijn sequence* $u \in \mathcal{C}(m,2,k)$, *which is a neighbour of* x *in* $\mathcal{C}(m,2,k)$ *such that* $d(u,y) < d(x,y)$.

Lemma 2.8 is crucial in the proof of following

Theorem 2.9 *Any two distinct multi de Bruijn sequences in* $\mathcal{C}(m,2,k)$ *can be connected by applying a sequence of the cross-join operations.*

Proof: Let x and y be two distinct sequences in $\mathcal{C}(m,2,k)$. By Lemma 2.8 x has a neighbour u_1 such that $d(u_1,y) < d(x,y)$. If $u_1 = y$ then we are done, otherwise the same argument can be iterated to get a sequence u_2, which is a neighbour of u_1, with $d(u_2,y) < d(u_1,y)$. Due to the strict inequality, and since the number of sequences in $\mathcal{C}(m,2,k)$ is finite, it is evident that this iterative process must end at y after a finite number of steps l, leading to the desired path $u_0 = x, u_1, \ldots, u_l = y$. □

Proof of Lemma 2.8. We take the least lexicographical representatives of the sequences $x = (X_i)$ and $y = (Y_i)$, where X_i and Y_i are successive states of the multi de Bruijn sequences. Let M_0 be the maximal common initial sequence of x and y. Suppose that the sequence

$$M_0 : \mathbf{0} = X_1 \to X_2 \to \cdots \to X_{L_0}$$

is common to x and y and L_0 is maximal. Since $x \neq y$, $L_0 < N$ and for the successors of X_{L_0} in x and y at least one is distinct from the state $\mathbf{0}$. Let us refer to these successors as $X^{(1)}$ and X_{L_0+1}. Since x is a multi de Bruijn sequence it contains every state, so it must contain X_{L_0+1}. The later is at least one of the states in $\widetilde{M_0}$ the complement of M_0 in x; that is, the sub-sequence of x that starts with $X^{(1)}$ and ends just before the state 0. Let ${}^*X_0$ be the the predecessor of X_{L_0+1} in x. Since X_{L_0+1} belongs to $\widetilde{M_0}$, the state ${}^*X_0$ is either in $\widetilde{M_0}$ or it is X_{L_0} itself. But the later is not possible because otherwise the common initial sub-sequence of x and y would extend to X_{L_0+1} defying the maximality of M_0. Now X_{L_0} and ${}^*X_0$ are predecessors of the same state and hence they are conjugate. Swapping their successors we split x into two cycles, a cycle C_1 that includes the state $\mathbf{0}$ and another cycle $\widetilde{C_1}$ that includes the states ${}^*X_0 \to X^{(1)}$.

The cycle C_1 aligned to start at the state $\mathbf{0}$ and the multi de Bruijn cycle y have a maximal common initial sequence of states

$$M_1 : \mathbf{0} = X_1 \to X_2 \to \cdots \to X_{L_0} \to \cdots \to X_{L_1}$$

where $L_1 \geq X_{L_0+1}$. The rest of the proof depends on establishing the following

Claim 1: It is possible to join C_1 and $\widetilde{C_1}$ by using the state in $\widetilde{M_1}$ - the complement of M_1 in C_1 - and a conjugate state in $\widetilde{C_1}$, i.e., there is a state in $\widetilde{M_1}$ that has a conjugate in $\widetilde{C_1}$. The proof of the claim goes by contradiction and it will be presented in the entire paper.

Corollary 2.10 *Starting with a multi de Bruijn sequence in $\mathcal{C}(m,2,k)$ and applying the cross-join method generates all sequences in $\mathcal{C}(m,2,k)$.*

We have implemented the cross-join method for the multi de Bruijn sequences of the type $\mathcal{C}(2,2,3)$. We have started from the first sequences in the list in Example 2.11 and generated all sequences. The implementation has been done in SAGE [14]. For each sequence the succeeded states are represented as decimals and the sequence representative is a least lexicographical one.

Example 2.11 *The sequences $C(2,2,3)$, $|C(2,2,3)| = 82$. (Tesler [?])*

(04252104635677 31)	(04256352104677 31)	(04256310426577 31)	(04210426356773 1)	(00425214635673 1)	(00425635214677 31)
(00425631425677 31)	(00425635677314 21)	(00463142525677 31)	(00463142525677 31)	(00463525214677 31)	(00463521425677 31)
(00463567735214 21)	(00467731425635 21)	(00467735635214 21)	(00467352142567 31)	(00467735214256 31)	(00425256314677 31)
(00425677356314 21)	(00425673142567 31)	(00425677314256 31)	(00421463525677 31)	(00421463567735 21)	(00421467735635 21)
(00421467352567 31)	(00421467735256 31)	(00421425673567 31)	(00421425677356 31)	(00463525677314 21)	(00467731463525 21)
(00467352567314 21)	(00467735256314 21)	(00467731425673 52)	(00425677314635 21)	(00425673567314 21)	(00425256773146 31)
(00425673146735 21)	(00425631467735 21)	(00421046352567 31)	(00421046356773 52)	(00421046773563 52)	(00421046735256 31)
(04673104677352 52)	(04631046773525 21)	(04256773104635 21)	(04256731046735 21)	(04256773521046 31)	(04256773521046 31)
(04252567310467 31)	(04252567731046 31)	(04252104673563 31)	(04210467735631 31)	(04210467356731 31)	(04256731042567 31)
(04210467735631)	(04256731046735 21)	(04252567314635 21)	(04210425673567 31)	(04210425673567 31)	(04256731042567 31)
(00467352521467 31)	(00467735252146 31)	(00467352567314 31)	(00425677352146 31)	(00425673521467 31)	(00425214673567 31)
(00425214677356 31)	(00463521467735 21)	(00463142567735 21)	(00467735631425 21)	(00467314252567 31)	(00467352146735 21)
(00463146773525 21)	(00467314256773 52)	(00467352104673 52)	(00467352146735 21)	(00467735214673 52)	(00467352146735 21)
(00421467356735 21)	(00421425635677 31)	(04673521046735 21)	(04210467356735 21)		

The green and the red sequences are the contatenation of de Bruijn sequences of lower order.

3 Galois NLFSRs

Following Dubrova [4, 5] we introduce Galois NLFSRs. Each bit i in the state of Galois NLFSR is updated to its next-state function which is a Boolean function of state variables. We considered Galois NLFSRs of order 4 given by Dubrova ([5], Table 1) which have period 15. We checked experimentally that some of them generate modified multi de Bruijn sequences of type $C(2,2,3)$ in which one of the 3-*mers* (here (000) and in the one case (111)) appears once.
Open problem: All binary multi de Bruijn sequences can be generated by some Galois NLFSRs.

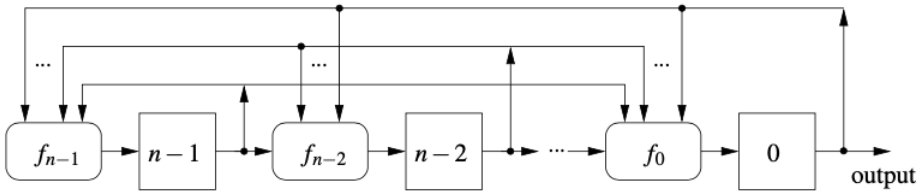

Fig. 1 (from [5]). A scheme of Galois NLFSRs of order n.

Example 3.1 *The list of Galois NLFSRs generating modified multi de Bruijn sequences of the type C(2, 2, 3).*

#	f_3	f_2	f_1	f_0
1	x_0	$1+x_0+x_1+x_3+x_0x_1$	$1+x_1+x_2+x_3$	$1+x_0+x_1+x_2+x_0x_2$
2	x_0	$x_3+x_0x_2$	$x_2+x_3+x_1$	$x_1+x_0+x_0x_2$
3	x_0	$x_3+x_0x_2$	$x_2+1+x_3+x_0x_3$	$x_1+1+x_2+x_0+x_2x_0$
4	x_0	$x_3+x_0x_2$	$x_2+1+x_0+x_1x_0$	$x_1+1+x_2+x_0x_2$
5	x_0	$x_3+1+x_1+x_2+x_1x_2$	$x_2+x_1+x_0x_1$	$x_1+1+x_0+x_0x_2$
6	x_0	$x_3+x_1x_2$	$x_2+x_1+x_1x_0$	$x_1+x_2x_0$
7	x_0	$x_3+x_1x_2$	x_2+1+x_0	$x_1+1+x_2+x_3+x_2x_3$
8	x_0	$x_3+x_1+x_0x_1$	$x_2+1+x_3+x_1$	$x_1+x_2+x_0x_2$
9	x_0	$x_3+x_2+x_0x_2$	$x_2+1+x_0+x_1+x_0x_1$	$x_1+1+x_0+x_0x_2$
10	x_0	$x_3+x_2+x_1x_2$	$x_2+x_3x_1$	$x_1+x_2+x_3+x_2x_3$
11	x_0	$x_3+x_2+x_1x_2$	$x_2+x_0x_1$	$x_1+x_2+x_0x_2$
12	x_0	$x_3+x_2+x_1x_2$	x_2+x_0	$x_1+x_2+x_0x_2$
13	x_0	$x_3+x_2+x_2x_0$	x_2	$x_1+x_2x_0$
14	x_0	$x_3+1+x_1+x_2+x_1x_2$	$x_2+1+x_0+x_1x_0$	$x_1+1+x_0+x_2+x_2x_0$
15	$x_0+x_1x_2$	$x_3+x_0x_2$	x_2	$x_1+1+x_0+x_2+x_2x_0$
16	$x_0+1+x_1+x_2+x_1x_2$	$x_3+1+x_1+x_2+x_1x_2$	x_2	x_1+x_2
17	$x_0+1+x_1+x_3+x_1x_3$	$x_3+x_0x_1$	x_2+1+x_0	$x_1+1+x_2+x_0$
18	$x_0+x_2x_3$	$x_3+x_1+x_2+x_1x_2$	$x_2+x_1+x_1x_3$	x_1+x_0

The sequences generated by NLFSRs in the decimal and the binary form.

#	Decimal	Binary	#	Decimal	Binary
1	(046773525214631)	(000111101010011)	10	(042563567731421)	(000101101111001)
2	(042525631467731)	(000101011001111)	11	(042146356773521)	(000100110111101)
3	(046773563521421)	(000111101101001)	12	(042525677314631)	(000101011110011)
4	(042146773563521)	(000100111101101)	13	(042567735631421)	(000101111011001)
5	(046773146352521)	(0001111001101010	14	(046356773521421)	(000110111101001)
6	(046314677352521)	(000110011110101)	15	(004252567314631)	(000010101110011)
7	(042142567735631)	(000100101111011)	16	(042563142567731)	(000101100101111)
8	(042142563567731)	(0001001011011110	17	(042567731425631)	(000101111001011)
9	(046352521467731)	(000110101001111)	18	(046352146773521)	(000110100111101)

The blue sequence has one 3-mer in decimal (7) and in binary (111).

We thank anonymous referee for very helpful remarks and indicating the improvements.

References

[1] A. Alhakim. Hamiltonicity of the Cross-Join Graph of de Bruijn Sequences. arXiv:1805.12059v2 [math.CO], 22 Feb 2020.

[2] N. G. de Bruijn. A Combinatorial Problem, Koninklijke Nederlandse Akademie v. Wetenschappen 49, pp. 758764, (1946).

[3] Z. Chang, M. F. Ezerman, A. A. Fahreza, S. Ling, J. Szmidt, H. Wang. Binary de Bruijn Sequences via Zech's Logarithms. SN Computer Science, 2(4), pp. 1-18, (2021).

[4] E. Dubrova. A scalable method for constructing Galois NLFSRs with period $2^n - 1$ using cross-join pairs. IEEE Trans. on Inform. Theory, 59(1), pp. 703-709, (2013).

[5] E. Dubrova, M. Teslenko, H. Tenhunen. On Analysis and Synthesis of (n,k)-Non-Linear Feedback Shift Registers. DATE: 2008, pp. 1286-1291.

[6] C. Flye-Sainte Marie. Solution to problem number 58, l'Intermédiare des Mathématiciens, vol. 1, pp. 107-110, (1894).

[7] H. Fredricksen. A Survey of Full Length Nonlinear Shift Register Cycle Algorithms. SIAM Review, Vol. 24, No. 2, pp. 195-221, (1982).

[8] S. Golomb. Shift register sequences. San Fransisco, Holden-Day, (1967), revised edition, Laguna Hills, CA, Aegean Park Press, 1982.

[9] T. Helleseth, T. Klöve. The Number of Cross-join pairs in maximum length linear sequences. IEEE Trans. on Inform. theory, 31, pp. 1731-1733, (1991).

[10] D. Kandel, Y. Matias, R. Unger, and P. Winkler. Shuffling biological sequences. Discrete Appl. Math., 71(1-3):171–185, 1996.

[11] J. Mykkeltveit, J. Szmidt. On Cross Joining de Bruijn Sequences. Contemporary Mathematics, 63, pp.335-346, (2015).

[12] J.Szmidt. Nonlinear Feedback Shift Registers and Zech's Logarithms. Proceedings of International Conference on Military Communications and Information Systems (ICMCIS) (2019).

[13] G. Tesler. Multi de Bruijn Sequences. Journal of Combinatorics.8(3), pp. 439-474 (2017).

[14] SAGE. System for Algebra and Geometry Experimentation. Open-System Mathematical Software System.