# Probabilistic Generation of Good Span $n$ Sequences from Nonlinear Feedback Shift Registers

Kalikinkar Mandal and Guang Gong
Department of Electrical and Computer Engineering
University of Waterloo
Waterloo, Ontario, N2L 3G1, CANADA
Email: {kmandal, ggong}@ uwaterloo.ca

**Abstract**

A binary span $n$ sequence generated by an $n$-stage nonlinear feedback shift register (NLFSR) is a sequence with the randomness properties: period $2^n - 1$, balanced, and ideal $n$-tuple distribution. It is possible that it also has high linear span. For providing security in constrained environments such as RFID tags and sensor networks, a span $n$ sequence generated by a nonlinear feedback shift register can be used as a pseudorandom sequence generator or a building block in stream ciphers for generating random numbers, keystreams, etc. In this paper, we propose a random NLFSR and show the expected period of the random NLFSR sequences. Then we show a technique for producing span $n$ sequences using a nonlinear feedback shift register with the feedback function as the Welch-Gong (WG) transformation. By this approach, a number of span $n$ sequences with a moderate $n$ can be generated and the generated span $n$ sequences are having linear span either $2^n - 2$ (optimal) or lower bounded by $2^n - 2 - 4n$ (suboptimal). Furthermore, we consider a generalized method, in which the WG transformation is replaced by other functions such as three-term, five-term and Kasami functions and report the number of span $n$ sequences for each function.

**Keywords:** Nonlinear feedback shift register (NLFSR) generators, pseudorandom sequence generator (PRSG), stream ciphers, span $n$ sequences.

## 1 Introduction

In recent years, nonlinear feedback shift registers have received much attention in designing many cryptographic primitives such as pseudorandom sequence generators (PRSGs), stream ciphers, and lightweight block ciphers to provide security in communication systems. Ciphers based on NLFSRs are of great practical importance in many constrained environments, for instance, RFID tags and sensor networks due to their need for efficient hardware implementation and high throughput. Many cryptographic primitives have been designed using NLFSRs. For example, the design of the family of lightweight block ciphers KATAN & KTANTAN consists of NLFSRs, which are used for mixing the plaintext and the key properly for producing the ciphertext [6]. In a stream cipher, the encryption is performed by XORing the plaintext with the keystream bit by bit to produce the ciphertext, where the keystream, is a random-looking bit stream, which can be generated using NLFSRs and should satisfy as many of the following randomness properties as possible: long period, balance, equal distribution of runs and $k$-tuples, 2-level autocorrelation, low crosscorrelation and high linear span [7, 11, 13, 28]. Many proposals for stream ciphers constructed using nonlinear feedback shift registers can be seen in the Stream Cipher Project, ECRYPT, for example Grain and Trivium [10]. In general, NLFSRs cannot be directly used for generating keystreams in stream ciphers because the randomness properties including the period of sequences generated by the NLFSR with any feedback function are unknown and hard to determine.

However, *de Bruijn sequences* with period $2^n$ have known randomness properties, namely, the balance, ideal $n$-tuple distribution and large linear span [12, 21]. A *modified de Bruijn sequence* with period $2^n - 1$ is a pseudorandom sequence where each nonzero $n$-tuple occurs exactly once in one period of the sequence. This property is referred to as *the span $n$ property* of the pseudorandom sequences with period $2^n - 1$. Thus, a modified de Bruijn sequence is also called a span $n$ sequence. Often, de Bruijn sequences as well as span $n$ sequences are generated recursively by an $n$-stage nonlinear feedback shift register. Only, $m$-sequences, a class of span $n$ sequences, are generated by linear feedback shift registers.

A span $n$ sequence can be constructed from a de Bruijn sequence by removing any one zero from the run of zeros of length $n$ and similarly, a de Bruijn sequence can be formed from a span $n$ sequence by adding one zero to the run of zeros of length $n - 1$. Note that by adding an extra zero to the run of zeros of length $n - 1$ to an $m$-sequence, the linear span of the resultant de Bruijn sequence varies between $2^{n-1} + n + 1$ and $2^n - 1$ [4], but by removing any one zero from the run of zeros of length $n$ from the resultant de Bruijn sequence, it becomes an $m$-sequence or a span $n$ sequence with linear span $n$. So the lower bound of the linear span of the span $n$ sequence drops to $n$ [15] - this phenomenon suggests to study the randomness properties, in particular, the linear span property of span $n$ sequences instead of de Bruijn sequences for cryptographic usages. A span $n$ sequence with high linear span generated by an NLFSR can be used as a PRSG or a building block to design PRSGs and stream ciphers like a combinatorial generator. Until recently, there is no known construction of nonlinear feedback functions which generate span $n$ sequences.

Most of the research efforts devoted on span $n$ sequences have been concerned with the number of span $n$ sequences for different $n$ and the properties of all feedback functions [12, 21, 24], where the properties of feedback functions include the number of terms in the feedback functions [23, 24] and the weight of truth tables of the feedback functions [22, 24]. Mayhew and Golomb reported the number of span $n$ sequences for different values of the linear span of span $n$ sequences and for different values of the number of terms in the feedback functions ($4 \leq n \leq 6$) [21, 23] and Mayhew reported the number of span $n$ sequences for different weight classes of the truth tables of the feedback functions for $n = 6$ [24]. However, the number of span $n$ sequences for different weight classes and different values of the linear span is an unsolved problem for $n \geq 7$. In [5], Chan *et al.* have given a span $n$ sequence generation method that uses very simple quadratic functions as the feedback function, which is the sum of a linear function in $n$ variables and a quadratic term for any two variables and reported the number of span $n$ sequences for $5 \leq n \leq 12$. Note that all the methods for finding the number of span $n$ sequences use an exhaustive search.

In this paper, we first analyze the expected period of a sequence generated by a random feedback function of an NLFSR and then present a technique for generating span $n$ sequences using a nonlinear feedback shift register with a particular type of nonlinear feedback function. More specifically, the nonlinear feedback function is the sum of one 1-degree monomial and the nonlinear WG transformation in $t$ variables ($5 \leq t \leq n-1$). In the NLFSR, using the WG transformations as feedback functions and varying all possible $t$-tap positions, new span $n$ sequences can be produced. Our computational results on the enumeration of new span $n$ sequences show that the maximum number of span $n$ sequences can be obtained if the number of inputs to the WG transformation is about half of the length of shift registers. Using this method, the probability for producing a span $n$ sequence is larger than that of a random generation. Moreover, we consider a generalized method for generating span $n$ sequences, in which the WG transformation is replaced by some other functions. Three-term functions, five-term functions, and Kasami functions are used as the nonlinear feedback functions in the generalized method and we report the number of span $n$ sequences for each function. All new span $n$ sequences produced using WG transformations, three-term functions, five-term functions and Kasami functions have an optimal linear span $2^n - 2$ or are lower bounded by suboptimal linear span $2^n - 2 - 4n$, but most new sequences have an optimal linear span.

The rest of the paper is organized as follows. In Section 2, we define and explain some terms which will be used in this paper for producing span $n$ sequences. In Section 3, we prove the period of a sequence generated by a random nonlinear feedback function. In Section 4, we present the technique for generating span $n$ sequences using WG transformations and their success probability and in the same section, we generalize the technique for producing span $n$ sequences by considering three-term, five-term and Kasami functions. In Section 5, we present the linear span property of new span $n$ sequences generated by the aforementioned functions. Finally, in Section 6, we briefly conclude the paper.

## 2 Notations and Preliminaries

In this section, we define and explain some terms and mathematical functions which will be used to produce span $n$ sequences.

- $\mathbb{F}_2 = \{0, 1\}$ - the Galois field with two elements.
- $\mathbb{F}_{2^t} = \{(x_0, x_1, ..., x_{t-1}) \ : x_i \in \mathbb{F}_2\}$ - an extension field which is generated by a primitive element $\alpha$ with $p(\alpha) = 0$, where $p(x) = c_0 + c_1 x + \cdots + c_{t-1} x^{t-1} + x^t$ is a primitive polynomial of degree $t$ ($\geq 2$) over $\mathbb{F}_2$.
- $\mathbb{F}_{2^q}^n = \{(a_0, a_1, ..., a_{n-1}) \ : a_i \in \mathbb{F}_{2^q}\}$ - a vector space over $\mathbb{F}_{2^q}$ ($q \geq 1$) with $2^{nq}$ elements.
- $\mathrm{Tr}(x) = x + x^2 + \cdots + x^{2^{t-1}}$ - the trace function from $\mathbb{F}_{2^t}$ to $\mathbb{F}_2$.
- $D_t = \{d : d \text{ is a coset leader with } \gcd(d, 2^t - 1) = 1\}$. Then the cardinality $|D_t|$ of $D_t$ is given by $\frac{\phi(2^t-1)}{t}$, where $\phi(\cdot)$ is the Euler phi function.

## 2.1 Basic Definitions and Properties

An $n$-stage linear or nonlinear feedback shift register is used to generate a periodic sequence $\mathbf{a} = \{a_i\}$ over the field $\mathbb{F}_{2^q}$ and the recurrence relation for the (N)LFSR is defined as [11]

$$a_{n+k} = f(a_k, a_{k+1}, ..., a_{k+n-1}) = a_k + g(a_{k+1}, ..., a_{k+n-1}),\ a_i \in \mathbb{F}_{2^q},\ k \geq 0$$

where $S_k = (a_k, a_{k+1}, ..., a_{k+n-1})$ is the $k$-th state of the shift register, $S_0$ is the initial state, $f(\cdot)$ is a function from $\mathbb{F}_{2^q}^n$ to $\mathbb{F}_{2^q}$ and $g(\cdot)$ is a function from $\mathbb{F}_{2^q}^{n-1}$ to $\mathbb{F}_{2^q}$. In particular $q = 1$, the function $f$ as well as the function $g$ is called a Boolean function and the sequence $\mathbf{a}$ is called a binary sequence. If the function $f$ is an affine function, then the sequence $\mathbf{a}$ is called an LFSR-sequence; otherwise it is called an NLFSR-sequence.

The complementary binary sequence of a binary sequence $\mathbf{b} = \{b_i\}_{i\geq 0}$, denoted as $\bar{\mathbf{b}}$, is given by $\{\bar{b_i}\}_{i\geq 0}$, where $\bar{b_i} = b_i + 1 \bmod 2$.

A binary sequence with period $2^n$ which satisfies the property that all $n$-tuples in one period are distinct is called a *de Bruijn sequence*. For a binary sequence of period $2^n - 1$, we say that it is *balanced* if there are $2^{n-1}$ 1's $2^{n-1} - 1$ 0's in one period of the sequence. If each nonzero $n$-tuple occurs exactly once in one period of the sequence, then it is called a *modified de Brujin sequence* or *span $n$ sequence*. In the rest of the paper, we use the term span $n$ sequence. The *linear span or linear complexity* of a sequence is the length of the shortest LFSR that will generate the given sequence.

**Property 1** *The linear span of a de Bruijn sequence, denoted as $LS_{db}$, is bounded by [4]*

$$2^{n-1} + n + 1 \leq LS_{db} \leq 2^n - 1. \tag{1}$$

*On the other hand, the linear span of a span $n$ sequence, denoted as $LS_s$, is bounded by*

$$2n < LS_s \leq 2^n - 2. \tag{2}$$

From this property, we say that a span $n$ sequence has the optimal linear span if its linear span is equal to $2^n - 2$.

### The WG Transformation

Let $t$ be a positive integer with $t \bmod 3 \not\equiv 0$ and $3k \equiv 1 \bmod t$. We define the function $h(x)$ from $\mathbb{F}_{2^t}$ to $\mathbb{F}_{2^t}$ by $h(x) = x + x^{q_1} + x^{q_2} + x^{q_3} + x^{q_4}$ and the exponents are given by

$$q_1 = 2^k + 1, q_2 = 2^{2k} + 2^k + 1, q_3 = 2^{2k} - 2^k + 1, q_4 = 2^{2k} + 2^k - 1.$$

Then the function from $\mathbb{F}_{2^t}$ to $\mathbb{F}_{2^t}$ is defined as

$$\mathrm{WGP}(x) = h(x + 1) + 1$$

is known as the *WG permutation* and the functions from $\mathbb{F}_{2^t}$ to $\mathbb{F}_2$ are defined by

$$f_d(x) = \mathrm{Tr}(\mathrm{WGP}(x^d)) \text{ and } g_d(x) = \mathrm{Tr}(h(x^d)),\ d \in D_t$$

are known as the *WG transformation* and *five-term ( or 5-term) function*, respectively [8, 14, 27]. The WG transformation has good cryptographic properties such as high algebraic degree, high nonlinearity, and high linear span.

For a fixed $t$, the number of WG transformations is given by $\left(\frac{\phi(2^t-1)}{t}\right)^2$. We now define a set

$$D_t^* = \{d \,:\, d \in D_t \text{ and } f_d(\cdot) \text{ is a nonlinear function}\}.$$

For such a choice of decimation numbers in $D_t^*$, we take into account all nonlinear WG transformation functions.

### Three-term Functions and Kasami Functions

Let $t = 2k - 1$ and $t \geq 5$. We denote the permutation by $h(x)$ over the field $\mathbb{F}_{2^t}$ and defined by $h(x) = x + x^{2^k+1} + x^{2^k-1}$, which is known as the *three-term permutation* [8]. Then the *three-term (or 3-term) function* from $\mathbb{F}_{2^t}$ to $\mathbb{F}_2$ is defined by

$$T_d(x) = \text{Tr}(h(x^d)),\ d \in D_t,\ x \in \mathbb{F}_{2^t}.$$

Note that the function $T_d(x)$ is a quadratic function.

Let $t$ be an odd positive integer. The *Kasami* exponent is defined as $e = 2^{2s} - 2^s + 1$, where $s < t$ and $\gcd(s, t) = 1$. Then the function

$$h(x) = x^e, x \in \mathbb{F}_{2^t}$$

is called a *Kasami power function* [17].

## 3 Average Period of Sequences Generated by a Random NLFSR

In this section, we first recall the relation between a regular directed graph and an NLFSR over an extension field. We then define the notion of random nonlinear feedback functions. Finally, we prove the expected period of a sequence generated by a random feedback NLFSR with a random starting state (an initial state) using random walks.

### 3.1 Generation of Random NLFSR Sequences

Let $S_k = (a_k, a_{k+1}, ..., a_{k+n-1})$ and $S_{k+1} = (a_{k+1}, a_{k+2}, ..., a_{k+n})$ be the $k$-th state and $(k+1)$-th state, respectively. Then $S_{k+1} = f(S_k),\ k \geq 0$, where $f$ is the feedback function of the NLFSR. Let $\mathbb{G} = (V, E)$ be a directed graph, which is defined as: denote each state $S_k$ as a vertex $v_k \in V$ and there exists a directed edge $e_k \in E$ from the state $S_k$ to the state $S_{k+1}^i = (a_{k+1}, a_{k+2}, ..., a_{n+k-1}, a_{n+k}^i),\ a_{n+k}^i \neq a_{n+k}^j, i \neq j,$ for $i = 1, 2, ..., 2^q$. This directed graph $\mathbb{G}$ is known as a de Bruijn graph [2, 16], which is a $2^q$-regular graph with $|V| = 2^{qn}$ and $|E| = 2^{q(n+1)}$.

We now pose the notion of a random NLFSR-sequence. We define the random feedback function $F$ as $F = (f, \Omega)$, where $\Omega$ is a *uniform* probability distribution and the uniform probability is given by $p_j = \frac{1}{2^q},\ j = 1, 2, ..., 2^q$ and $f$ is the feedback function. Then, on the input state $S_k$, $F$ outputs $S^i_{k+1}$, i.e., $S^i_{k+1} = F(S_k)$, for some $i$, where $i$ is chosen according to $\Omega$ and $f$ such that $S^i_{k+1}$ is not already generated by $F$. For any initial state $S_0$, the random NLFSR-sequence generated by $F$ is given by $\mathbf{a} = \{a_0, a_1, ..., a_k, a_{k+1}, ...\}$ and the period of the sequence is $P$ $(> 0)$, if $S_0 = F^P(S_0)$, where $F^P(S) = F^{P-1}(F(S))$.

We shortly recall the definition of a random walk on the directed graph $\mathbb{G}$. Let $F = (f, \Omega)$ be a random nonlinear feedback function. The random walk between the vertices $v_k$ and $v_{k+1}$ with the imposed uniform probability distribution is defined by [20]:

- Let $S_k$ and $S_{k+1}$ be the states corresponding to the vertices $v_k$ and $v_{k+1}$, respectively.

- Then, at the vertex $v_k$, the random walk according to $F$ choses a vertex $v_{k+1}$ randomly under uniform distribution, i.e., $\Pr(S_{k+1} = F(S_k)) = \frac{1}{2^q}$.

We now consider a simple random walk $R(P)$ of length $P$ on the graph $\mathbb{G}$ and which is defined by [3, 20]:

- Starting at any vertex $v_0$

- The vertex $v_1$ is chosen at $v_0$ with $\Pr(S_1 = F(S_0)) = \frac{1}{2^q}$, if $v_1$ is not visited before

- Repeat the above step until it reaches $v_0$

- If it reaches $v_0$ then stop.

From the above, it can be noticed that the generation of sequence $\{a_i\}$ by $F$ is equivalent to a random walk which is performed according to $F$ and consequently, finding the expected period of the random NLFSR-sequence is equivalent to finding the expected value of $P$ in $R(P)$.

Consider the above random walk on a connected directed graph. The *marking time* is defined as the number of steps required to visit all the vertices of the graph [3]. The expected marking time for a connected directed regular graph with $m$ vertices is stated in the following lemma.

**Lemma 1** *[3] For a regular graph $G$, one has expected marking time $E(T) = \frac{mH_m}{p} + O(1)$, where $p$ is uniform probability to choose a vertex.*

**Theorem 1** *Let us consider an $n$-stage NLFSR with a random feedback function, which is defined over the field $\mathbb{F}_{2^q}$, where each cell of the shift register has $q$ bits. The expected period $P$ of an NLFSR-sequence generated by the random feedback NLFSR is lower bounded by $\sqrt{2^{qn}}$.*

**Proof** The directed de Bruijn graph $\mathbb{G} = (V, E)$ of an $n$-stage feedback shift register over the field $\mathbb{F}_{2^q}$ is a $2^q$-regular graph with $N = 2^{qn}$ vertices. Let $\{a_i\}$ be a periodic random NLFSR-sequence with period $P$ generated by the random feedback function $F$ for any initial state $S_0$. Let us consider the above random walk $R(P)$ of length $P$ on $\mathbb{G}$ with a starting vertex $v_0$, which is corresponding to the initial state $S_0$. To prove the expected period of the random NLFSR-sequence is equivalent to finding the expected length of $R(P)$. We now find the lower bound of the expected value of $P$.

For large $N$, the $N$-th harmonic number can be written as $H_N = 1 + \frac{1}{2} + \cdots + \frac{1}{N} \approx \log N$ and $NH_N \approx N \log N$. Assume that in $N$ steps the expected $P = N^x (x \geq 0)$ number of vertices will be visited according to the random walk $R(P)$. Then, according to Lemma 1, the expected number of steps required to visit all $P$ vertices, i.e., the marking time, is given by $2^q P \log P$.

We define the function $f(x)$ as $f(x) = 2^q P \log P - N = 2^q x N^x \log N - N$. We now want to find an approximate value of $x$ such that $f(x) \geq 0$. We can see that at $x = \frac{1}{2}$, $f(x) < 0$ and $f(x') = 0$ for some $1 > x' > \frac{1}{2}$. So the value of $P$ is greater than $\sqrt{N}$. Hence, in $N$ steps, it is possible to visit at least $\sqrt{N}$ vertices (on average) using a random walk $R(P)$. Therefore, the expected period of a sequence generated by a random feedback function in an NLFSR is lower bounded by $\sqrt{N} = \sqrt{2^{qn}}$. □

# 4 Generation of Span $n$ Sequences Using the WG Transformations

In this section, we consider the generation of span $n$ sequences using nonlinear feedback shift registers with the WG transformations as the feedback functions. The WG transformations in $t$ variables are balanced and have even Hamming weight $2^{t-1}$, so the new span $n$ sequences generated using WG transformations belong to the weight class $2^{n-2}$. The new span $n$ sequences have good randomness properties, especially, they have high linear span.

## 4.1 Description of A Span $n$ Sequence Generation Procedure

Let $\mathbf{a} = \{a_i\}$ be a binary sequence generated by an $n$-stage nonlinear recurrence relation, which is defined as

$$a_{n+k} = a_k + f_d(x_k),\ x_k = (a_{r_1+k}, a_{r_2+k}, \cdots, a_{r_t+k}) \in \mathbb{F}_{2^t},\ d \in D_t^*,\ k = 0, 1, 2, \cdots, \quad (3)$$

where $(r_1, r_2, ..., r_t)$ with $0 < r_1 < r_2 < \cdots < r_t \leq n-1$ is called a $t$-tap position of the NLFSR and $f_d(x)$ is the WG transformation. A block diagram of the generation procedure is given in Figure 1. For a proper selection of a $t$-tap position and a feedback function $f_d(x)$, the binary sequence $\mathbf{a}$ can be a span $n$ sequence, which is produced by the WG transformation. Note that if the number of terms in the Boolean representation of the WG transformation $f_d(\cdot)$ is even, then the recurrence relation (3) cannot generate any span $n$ sequence for all $t$-tap positions, since for the all-one state the recurrence relation generates the all-one sequence.

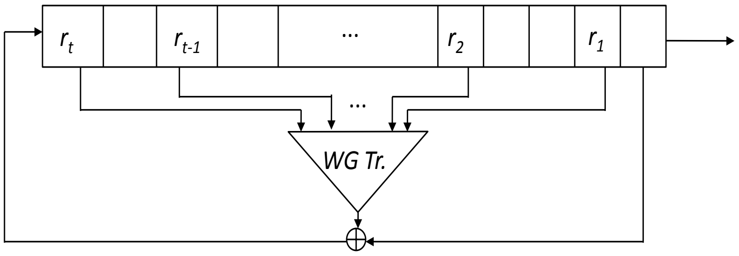

Figure 1: Block diagram of the span $n$ sequence generation procedure

Let $\mathbf{b} = \{b_i\}$ be a binary sequence generated by the following recurrence relation

$$b_{n+k} = 1 + b_k + f_d(x_k),\ x_k = (b_{r_1+k}, b_{r_2+k}, \cdots, b_{r_t+k}) \in \mathbb{F}_{2^t},\ d \in D_t^*,\ k = 0, 1, 2, \cdots. \quad (4)$$

Similarly, for a proper selection of a $t$-tap position and a feedback function $f_d(x)$, the complementary binary sequence $\bar{\mathbf{b}}$ of $\mathbf{b}$ can be a span $n$ sequence, but the sequence $\mathbf{b}$ cannot be a span $n$ sequence since it contains all-zero initial state.

In algebraic normal form (ANF) representations of WG transformations, we can notice that some WG transformations having an odd number of terms and some having an even number of terms, so using the recurrence relations (3) and (4), all the WG transformations are to be included for producing span $n$ sequences. In the recurrence relations (3) and (4), by varying three parameters, namely, the primitive polynomial $p(x)$, the decimation number $d$, and the $t$-tap position $(r_1, r_2, ..., r_t)$, a number of new span $n$ sequences can be found and that number mainly depends on the length $n$ of the NLFSR and the number $t$ of inputs to the WG transformation. Note that we may not always obtain span $n$ sequences for any length $n$ of the NLFSR. A special case of the recurrence relation (3) with the trace function of $n-1$ variables as the feedback function is defined in [26].

A periodic reverse binary sequence is defined as follows [22, 23]: For a binary sequence $\{a_0, a_1, ..., a_{2^n-2}\}$ with period $2^n - 1$, the reverse sequence of the binary sequence is given by $\{a_{2^n-2}, a_{2^n-3}, ..., a_1, a_0\}$. A reverse sequence of a span $n$ sequence is also a span $n$ sequence, which is not shift equivalent of the original one and the reverse span $n$ sequence can be generated by the same function but with a different $t$-tap position.

Our span $n$ sequences are uniquely determined by three parameters:

1. the decimation number $d$,
2. the primitive polynomial $p(x)$, and
3. the $t$-tap position $(r_1, r_2, ..., r_t)$.

Similarly, the reverse span $n$ sequence of the span $n$ sequence is represented by the same decimation number $d$ and the same primitive polynomial $p(x)$, but with the different $t$-tap position $(n - r_1, n - r_2, ..., n - r_t)$. Table 3 presents a few instances of new span $n$ sequences for $t = 5$ and $n = 7$. For $n > t$, there are different choices of $t$-tap positions, so for a fixed WG transformation $f_d(x)$, a span $n$ sequence generated by $f_d(x)$ is different, if the $t$-tap position is different.

## 4.2 The Complexity/Size of the Search Space

Recall that three parameters are involved in the recurrence relation for finding the number of new span $n$ sequences for fixed $n$ and $t$. The size of the search space or the number of possible span $n$ sequences in terms of $n$ and $t$ is determined in the following proposition.

**Proposition 1** *For any $n > t \geq 6$, the complexity or size of the search space for finding the span $n$ sequences is given by $C = \left(\frac{\phi(2^t-1)}{t}\right)^2 \binom{n-1}{t}$.*

**Proof** In the recurrence relations, the first position is fixed for the sequence to be periodic and any $t$ tap positions is chosen from $n-1$ positions ($n \geq 6$) to form a $t$-tap position, so the number of distinct $t$-tap positions is given by $T = \binom{n-1}{t}$. The total number of nonlinear feedback functions is given by $n_p \cdot |D_t^*|$, where $n_p = \frac{\phi(2^t-1)}{t}$ is the number of $t$ degree primitive polynomials over $\mathbb{F}_2$ and $|D_t^*|$ is the number of decimation numbers for which the feedback function is nonlinear. Hence, for fixed $n$ and $t$, the complexity/size of the search space is

$$\begin{aligned} C = n_p \cdot |D_t^*| \cdot T &= \left(\frac{\phi(2^t-1)}{t}\right)^2 \binom{n-1}{t} \text{ for } t > 5 \\ &= \left(\frac{\phi(2^t-1)}{t}\right)\left(\frac{\phi(2^t-1)}{t} - 1\right)\binom{n-1}{t} \text{ for } t = 5, \end{aligned}$$

since for $d = 5$ the WG transformation is linear. □

### 4.3 New Span $n$ Sequences Generated Using WG transformations

In this subsection, we report the number of new span $n$ sequences obtained using the recurrence relations (3) and (4) for different $t$ and $n$; the new span $n$ sequences are generated by computer simulations. We consider the WG transformations over the field $\mathbb{F}_{2^t}$ for $t = 5, 7, 8, 10$, and 11. Tables 1 and 2 present the number of new span $n$ sequences corresponding to the recurrence relations (3) and (4), respectively ($6 \leq n \leq 20$). However this method can be applied to generate larger length span $n$ sequences. In Tables 1 and 2, "×" represents that for such values of $n$ and $t$ the recurrence relations are not defined and $\sim$ represents that those cases the number of span $n$ sequences is not yet determined. We present some instances of new span $n$ sequences for different $n$ in the Appendix.

Table 1: Number of span $n$ sequences generated using the recurrence relation (3)

| | | $n$ | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $t$ | WG$t$ | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 |
| 5 | WG5 | 0 | 9 | 7 | 14 | 8 | 11 | 17 | 11 | 13 | 10 | 3 | 7 | 7 | 0 | 1 |
| 7 | WG7 | × | × | 3 | 25 | 42 | 63 | 108 | 138 | 138 | 125 | 126 | 111 | 83 | 86 | 63 |
| 8 | WG8 | × | × | × | 3 | 9 | 18 | 34 | 76 | 96 | 104 | 106 | 108 | 110 | 90 | 79 |
| 10 | WG10 | × | × | × | × | × | 5 | 40 | 107 | 246 | 373 | 627 | 819 | 999 | $\sim$ | $\sim$ |
| 11 | WG11 | × | × | × | × | × | × | 31 | 204 | 574 | 1313 | 2539 | 4079 | $\sim$ | $\sim$ | $\sim$ |
| Total | | 0 | 9 | 10 | 42 | 59 | 97 | 230 | 536 | 1067 | 1925 | 3401 | 5124 | – | – | – |

A graphical representation of the number of new span $n$ sequences is provided in Figure 2, which shows that for different $t$ the distribution of the number of new span $n$ sequences has the following property: the number of span $n$ sequences increases as $n$ increases and it reaches the maximum for some value of $n$. After that the number of span $n$ sequences decreases as $n$ increases. At a quick glance, we can see that the number of span $n$ sequences is maximal close to $n = 2t$, which follows from the fact that the complexity of the search space is a multiple of the binomial coefficient. This phenomenon reveals that there exists a tradeoff between $n$ and $t$ to obtain the maximum number of span $n$ sequences.

Table 2: Number of span $n$ sequences generated using the recurrence relation (4)

| | | $n$ | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $t$ | WG-$t$ | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 |
| 5 | WG5 | 1 | 7 | 7 | 10 | 16 | 18 | 10 | 8 | 4 | 10 | 2 | 1 | 3 | 1 | 0 |
| 7 | WG7 | × | × | 4 | 25 | 47 | 59 | 121 | 122 | 137 | 125 | 123 | 98 | 74 | 84 | ∼ |
| 8 | WG8 | × | × | × | 1 | 6 | 35 | 33 | 75 | 73 | 91 | 123 | 115 | 106 | ∼ | ∼ |
| 10 | WG10 | × | × | × | × | × | 4 | 47 | 118 | 270 | 401 | 680 | 863 | ∼ | ∼ | ∼ |
| 11 | WG11 | × | × | × | × | × | × | 33 | 186 | 576 | 1350 | 2522 | ∼ | ∼ | ∼ | ∼ |
| Total | | 1 | 7 | 11 | 36 | 69 | 116 | 244 | 509 | 1060 | 1977 | 3450 | – | – | – | – |

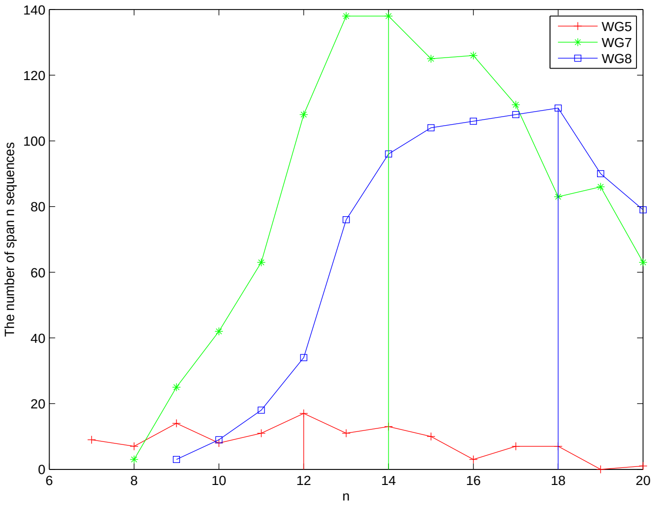

Figure 2: Distribution of the number of span $n$ sequences

**Example 1** The following example describes our span $n$ sequence generation procedure for $t = 5$ and presents new span $n$ sequences for $n = 7$. The WG transformation over $\mathbb{F}_{2^5}$ is given by

$$f(x) = \mathrm{Tr}(x + (x+1)^5 + (x+1)^{13} + (x+1)^{19} + (x+1)^{21}).$$

After simplification, $f(x)$ can be written as

$$f(x) = \mathrm{Tr}(x^{19}),\ x \in \mathbb{F}_{2^5}.$$

For $t = 5$, $D_t = \{1, 3, 5, 7, 11, 15\}$, the set of coset leaders and $D_t^* = \{1, 3, 7, 11, 15\}$, since for $d = 5$, the function $f_d(x)$ is linear. The $d$-th decimation of $f(x)$ is given by

$$f_d(x) = f(x^d) = \mathrm{Tr}(x^{d'}),\ d' = (19 \cdot d) \bmod 2^t - 1,\ d \in D_t^*.$$

The $n$-stage nonlinear recurrence relation with a $t$-tap position is given by

$$a_{n+k} = a_k + f_d(x_k),\ x_k = (a_{r_1+k}, a_{r_2+k}, a_{r_3+k}, a_{r_4+k}, a_{r_5+k}) \in \mathbb{F}_{2^5},\ k \geq 0.$$

For $n = 7$, we have found nine span $n$ sequences by an exhaustive search and present those sequences in Table 3.

Table 3: Span $n$ sequences generated using WG5 for $n = 7$

| Decimation<br>$d$ | Polynomial<br>$(c_0, c_1, c_2, c_3, c_4)$ | $t$-tap position<br>$(r_1, r_2, r_3, r_4, r_5)$ |
|---|---|---|
| 1 | 1 1 1 0 1 | 1 2 3 4 5 |
| 1 | 1 1 0 1 1 | 1 3 4 5 6 |
| 7 | 1 0 0 1 0 | 1 2 3 4 6 |
| 7 | 1 0 1 0 0 | 1 2 4 5 6 |
| 7 | 1 0 1 1 1 | 2 3 4 5 6 |
| 11 | 1 0 0 1 0 | 1 2 4 5 6 |
| 11 | 1 1 1 1 0 | 1 2 4 5 6 |
| 11 | 1 1 1 0 1 | 1 2 4 5 6 |
| 15 | 1 1 1 1 0 | 1 2 4 5 6 |

### 4.4 Successful Probability and Comparisons with Existing Approaches

Tables 1 and 2 show that at $t = \lceil \frac{n}{2} \rceil$ or $t$ is close to $\lceil \frac{n}{2} \rceil$, the maximum number of span $n$ sequences can be obtained, which motivates us to compute the search complexity at $t = \lceil \frac{n}{2} \rceil$ for finding the maximum number of span $n$ sequences. Assume that we use NLFSRs defined by relations (3) and (4) for $t = \lceil \frac{n}{2} \rceil$. Let $N$ be the number of span $n$ sequences (including reverse span $n$ sequences) obtained using the relations (3) and (4). Then we have the following theorem.

**Theorem 2** *The number of instances in the search space for possible span n sequences is given by $C_0$, where $C_0 \approx \left( \frac{\phi(2^{\lceil \frac{n}{2} \rceil} - 1)}{\lceil \frac{n}{2} \rceil} \right)^2 \cdot \frac{2^{n-1}}{\sqrt{\pi \cdot \frac{n-1}{2}}}$ and $C_0 \approx \frac{2^{2n-1} - 2^{\frac{3n}{2}+1}}{\sqrt{\pi} \cdot (\lceil \frac{n}{2} \rceil)^{5/2}}$, if $2^t - 1$ is a Mersenne prime, and the success probability of obtaining such a span n sequence is given by $\frac{N}{C_0}$.*

**Proof** We recall that the complexity of search space is

$$C = \left( \frac{\phi(2^t - 1)}{t} \right)^2 \binom{n-1}{t}, \quad \text{for } t > 5.$$

Putting $t = \lceil \frac{n}{2} \rceil$ in the above formula, then we get

$$\begin{aligned} C_0 &= \left( \frac{\phi(2^{\lceil \frac{n}{2} \rceil} - 1)}{\lceil \frac{n}{2} \rceil} \right)^2 \cdot \binom{n-1}{\lceil \frac{n}{2} \rceil} \\ &= \left( \frac{\phi(2^{\lceil \frac{n}{2} \rceil} - 1)}{\lceil \frac{n}{2} \rceil} \right)^2 \cdot \binom{n-1}{\lfloor \frac{n-1}{2} \rfloor + 1}, \text{ for positive } n \\ &= \left( \frac{\phi(2^{\lceil \frac{n}{2} \rceil} - 1)}{\lceil \frac{n}{2} \rceil} \right)^2 \cdot \frac{(n - \lfloor \frac{n-1}{2} \rfloor - 1) \cdot \binom{n-1}{\lfloor \frac{n-1}{2} \rfloor}}{(\lfloor \frac{n-1}{2} \rfloor + 1)}. \end{aligned}$$

By Stirling's formula

$$\binom{m}{\lfloor \frac{m}{2} \rfloor} \sim \frac{2^m}{\sqrt{\pi m/2}},$$

the above equation can be written as

$$\begin{aligned} C_0 &\sim \left( \frac{\phi(2^{\lceil \frac{n}{2} \rceil} - 1)}{\lceil \frac{n}{2} \rceil} \right)^2 \cdot \frac{\lfloor \frac{n-1}{2} \rfloor \cdot 2^{n-1}}{(\lfloor \frac{n-1}{2} \rfloor + 1) \cdot \sqrt{\pi \cdot \frac{n-1}{2}}} \\ &\sim \left( \frac{\phi(2^{\lceil \frac{n}{2} \rceil} - 1)}{\lceil \frac{n}{2} \rceil} \right)^2 \cdot \frac{2^{n-1}}{\sqrt{\pi \cdot \frac{n-1}{2}}}. \\ &\approx \frac{2^{2n-1} - 2^{\frac{3n}{2}+1}}{\sqrt{\pi} \cdot (\lceil \frac{n}{2} \rceil)^{5/2}}, \text{ if } 2^t - 1 \text{ is a Mersenne prime.} \end{aligned}$$

Hence, the result is proved. □

Note that the success probability to obtain a randomly generated span $n$ sequence is $\frac{1}{2^{n-3}}$ [24]. We compared the success probability to obtain a span $n$ sequence (including reverse sequences) with our approach and with random span $n$ sequence generation method for $t = 5, 7, 8$ ( for $t \approx \lceil \frac{n}{2} \rceil$), 10 and 11 (for $13 \leq n \leq 17$) and the comparison shows that our approach is better than the random span $n$ sequence generation method. Moreover, in our approach a span $n$ sequence is generated according to a construction of a nonlinear recurrence relation, but not in a random way.

As mentioned earlier, most existing techniques for generating span $n$ sequences conduct an exhaustive search over all Boolean functions or very simple types of quadratic functions [5, 21, 23]. Considering the algebraic normal form of an arbitrary Boolean function, it is not easy to understand the construction of that function. However, we use a particular type of nonlinear feedback functions which have compact representation and these nonlinear feedback functions can be studied apart from their ANF representations. One may analyze feedback functions, i.e., WG transformations with a $t$-tap position from the point of view of the bases of finite fields, the decimation numbers, the permutations and the selection of $t$-tap positions. For example, there exist many span $n$ sequences whose $t$-tap positions, the bases of the finite field are the same but their decimation numbers are different. In addition, the WG transformations have good cryptographic properties such as the algebraic degree, high nonlinearity, and high linear span. Using this type of functions, we can generate a number of span $n$ sequences for a proper choice of parameters $t$ and $n$, for example, $t = \frac{n}{2}$. Again, for a fixed $n$, by varying $t$, that is, by considering all the WG transformations over different fields, one can obtain more span $n$ sequences.

## 4.5 On the Reduction of Search Complexity

This subsection discusses how to reduce the size of the search space when not all new span $n$ sequences are aimed to be obtained. The idea of reducing the size of the search space is the following: by restricting the exhaustive search over a particular type of decimation numbers and over a selection of $t$-tap positions. If one or a few span $n$ sequences are aimed to obtain, then a search might be performed according to some patterns of decimation numbers or/and $t$-tap positions. Below we list a type of decimation number and $t$-tap positions. In some cases, we may not find any span $n$ sequence. However, according to our observations based on the above heuristic, it is possible to obtain many span $n$ sequences with reduced search complexity.

### 4.5.1 Observations on Decimation Numbers

We have performed a search on the following type of decimation numbers for different $n$

$$D_{dec} = \{d : \ d \in D_t^* \text{ and } d = 2^i - 1, i = 1, 2, \cdots, t-1\}$$

for $t = 7$, 8, and 10 and the result shows that there exist many span $n$ sequences whose decimation numbers are of the above type. The complexity of the search space for this type of decimation numbers is given by

$$C_{dec} = \frac{\phi(2^t - 1)}{t}(t-1)\binom{n-1}{t} \approx \phi(2^t - 1)\binom{n-1}{t}.$$

Obviously, the reduced complexity $C_{dec}$ is less than the original complexity $C$.

### 4.5.2 Observations on $t$-tap Positions

The search complexity can also be reduced by fixing a few tap positions among $t$ positions. Assume that it is possible to fix, let's say, $k$ tap positions $(1 \leq k \leq t)$. Then, the total number of fixed tap positions is $k+1$ and we only need to choose $t-k$ positions out of $n-1-k$ positions. So, for $k$ fixed choices of tap positions, the search complexity is reduced to

$$C_{tap} = \left(\frac{\phi(2^t - 1)}{t}\right)^2 \binom{n-1-k}{t-k}.$$

We have done an investigation on the $t$-tap positions for $t = 7, 8,$ and 10 and the result shows that the following types of $t$-tap positions are effective when the slope of the curves in Figure 2 increases gradually. For example, in case of $t = 7$, $n = 11, 12, 13$ and 14 and in case of $t = 8$, $n = 13, 14, 15, 16, 17$ and 18, the $t$-tap positions are given by:

$$\{1, 2, 3, 4, \cdots\},\ \{1, 2, 3, \cdots, n-1\},\ \{1, 2, \cdots, n-2, n-1\},\ \{1, \cdots, n-3, n-2, n-1\},$$

where the numbers in the tap positions represent fixed positions in a $t$-tap position (i.e., $k = 4$ fixed positions) and "$\cdots$" represents any other combinations of $n - k - 1$ tap positions. With respect to the above approach, we can reduce the size of the search space by some factors if we aim at finding only a few span $n$ sequences.

## 4.6 A Generalized Method for Span $n$ Sequences Generation Using NLFSRs

In this subsection, we present a generalized technique for generating span $n$ sequences, which is similar to the span $n$ sequence generation technique using WG transformations. A natural generalization is done by taking into account any permutation over the field $\mathbb{F}_{2^t}$ instead of the WG permutation or by considering any other functions which have a trace representation. In the generalized method, we have considered three-term, five-term, and Kasami functions for different $t$, $5 \leq t \leq 11$. Tables 4 - 9 present the number of span $n$ sequences obtained using the recurrence relations (3) and (4) for three-term functions, five-term functions, and Kasami functions. (Note that, when $t = 5$, the WG transformations and Kasami functions degenerate to the same functions and similarly, three-term functions and five-term functions degenerate to the same functions.)

Table 4: Number of three-term span $n$ sequences generated using rec. rel. (3)

| $n$ | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $t$ | T3-$t$ | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 |
| 5 | T3-5$^*$ | 1 | 3 | 9 | 8 | 9 | 8 | 4 | 3 | 5 | 2 | 3 | 1 |
| 7 | T3-7 | × | × | 6 | 25 | 51 | 89 | 103 | 150 | 131 | 128 | 127 | 123 |
| 9 | T3-7 | × | × | × | × | 8 | 52 | 104 | 223 | 391 | 549 | 710 | 770 |
| 11 | T3-11 | × | × | × | × | × | × | 35 | 190 | 624 | 1323 | 2580 | 4056 |
| Total | – | 1 | 3 | 15 | 33 | 68 | 149 | 246 | 566 | 1151 | 2002 | 3420 | 4950 |

Table 5: Number of three-term span $n$ sequences generated using rec. rel. (4)

| $n$ | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $t$ | T3-$t$ | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 |
| 5 | T3-5$^*$ | 1 | 2 | 2 | 5 | 10 | 5 | 6 | 5 | 3 | 1 | 3 | 5 |
| 7 | T3-7 | × | × | 4 | 24 | 44 | 84 | 98 | 122 | 133 | 146 | 128 | 111 |
| 9 | T3-7 | × | × | × | × | 12 | 47 | 109 | 237 | 361 | 553 | 694 | 823 |
| 11 | T3-11 | × | × | × | × | × | × | 34 | 186 | 578 | 1416 | 2554 | $\sim$ |
| Total | – | 1 | 3 | 6 | 29 | 66 | 136 | 247 | 550 | 1075 | 2116 | 3379 | – |

Table 6: Number of five-term span $n$ sequences generated using rec. rel. (3)

| | | $n$ | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $t$ | FT-$t$ | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 |
| 5 | FT-5* | 1 | 3 | 9 | 8 | 9 | 8 | 4 | 4 | 5 | 2 | 3 | 1 | 0 | 1 |
| 7 | FT-7 | × | × | 5 | 22 | 44 | 66 | 118 | 131 | 115 | 135 | 124 | 118 | 99 | 90 |
| 8 | FT-8 | × | × | × | 1 | 9 | 18 | 37 | 56 | 88 | 101 | 104 | 86 | 92 | 90 |
| 10 | FT-10 | × | × | × | × | × | 9 | 37 | 116 | 246 | 411 | 621 | 797 | 943 | ∼ |
| 11 | FT-11 | × | × | × | × | × | × | 25 | 171 | 590 | 1443 | 2618 | 4194 | ∼ | ∼ |
| Total | | 1 | 3 | 14 | 31 | 62 | 101 | 221 | 478 | 1044 | 2092 | 3470 | 5196 | – | – |

Table 7: Number of five-term span $n$ sequences generated using rec. rel. (4)

| | | $n$ | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $t$ | FT-$t$ | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 |
| 5 | T3-5* | 1 | 2 | 2 | 5 | 10 | 5 | 6 | 5 | 3 | 1 | 3 | 5 | 0 | ∼ |
| 7 | FT-7 | × | × | 8 | 19 | 43 | 74 | 108 | 138 | 138 | 127 | 117 | 102 | 84 | 91 |
| 8 | FT-8 | × | × | × | 0 | 6 | 22 | 38 | 54 | 66 | 116 | 89 | 106 | 83 | ∼ |
| 10 | FT-10 | × | × | × | × | × | 7 | 47 | 119 | 223 | 443 | 627 | 861 | ∼ | ∼ |
| 11 | FT-11 | × | × | × | × | × | × | 20 | 172 | 609 | 1397 | 2558 | ∼ | ∼ | ∼ |
| Total | | 1 | 2 | 10 | 24 | 59 | 108 | 219 | 488 | 1039 | 2084 | 3394 | – | – | – |

## 5 Linear Span of New Span $n$ Sequences

In this section, we study the linear span of new span $n$ sequences generated using the WG transformations, five-term functions, three-term functions, and Kasami functions. We note that all the nonlinear feedback functions have a trace representation. The linear span of a sequence is an important randomness property that is considered as an upper bound on sequence unpredictability because using only twice the linear span many consecutive bits of the sequence one can certainly predict the remaining bits of the sequence by the Berlekamp-Massey algorithm [1, 19]. There is no theoretical result on the linear span of span $n$ sequences generated by a nonlinear feedback shift register. What we know is the bounds presented in Property 1 in Section 2.

We compute the linear span of new span $n$ sequences by the Berlekamp-Massey algorithm and our computational results show that the linear spans attained by new sequences are the optimal $2^n-2$, the suboptimal $2^n-2-4n$ and between the optimal and suboptimal. Tables 10 and 11 present a summary of the linear spans of WG span $n$ sequences generated by the recurrence relations (3) and (4), respectively. Moreover, Tables 12, 13, and 14 exhibit a summary of the linear spans of the span $n$ sequences generated by Kasami functions, three-term functions, and five-term functions, respectively for different values of $t$. There exists only one span $n$ sequence whose linear span lies in the range from $2^n-2-4n$ to $2^n-2-3n$ and for all other

Table 8: Number of Kasami span $n$ sequences generated using rec. rel. (3)

| $n$ | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $t$ | KP-$t$ | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 |
| 5 | KP-5 | 0 | 9 | 7 | 14 | 8 | 11 | 17 | 11 | 13 | 10 | 3 | 7 | 7 | 0 |
| 7 | KP-7 | × | × | 6 | 17 | 41 | 76 | 79 | 118 | 108 | 99 | 125 | 78 | 88 | 72 |
| 9 | KP-7 | × | × | × | × | 10 | 43 | 120 | 258 | 410 | 519 | 662 | 788 | ∼ | ∼ |
| 11 | K-11 | × | × | × | × | × | × | 26 | 188 | 604 | 1423 | 2491 | ∼ | ∼ | ∼ |
| Total | – | 0 | 9 | 13 | 31 | 59 | 130 | 242 | 575 | 1135 | 2051 | 3281 | – | – | – |

Table 9: Number of Kasami span $n$ sequences generated using rec. rel. (4)

| $n$ | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $t$ | KP-$t$ | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 |
| 5 | KP-5 | 1 | 7 | 7 | 10 | 16 | 18 | 10 | 8 | 4 | 10 | 2 | 1 | 3 | 1 |
| 7 | KP-7 | × | × | 4 | 25 | 45 | 60 | 98 | 117 | 114 | 104 | 116 | 96 | 86 | 77 |
| 9 | KP-7 | × | × | × | × | 6 | 37 | 131 | 239 | 367 | 558 | 740 | 860 | ∼ | ∼ |
| 11 | K-11 | × | × | × | × | × | × | 32 | 184 | 596 | 1403 | 2547 | ∼ | ∼ | ∼ |
| Total | – | 1 | 7 | 11 | 35 | 67 | 115 | 271 | 548 | 1081 | 2075 | 3405 | – | – | – |

sequences the linear span is lower bounded by $2^n - 2 - 3n$. Our computational results also show that most of new sequences obtain the optimal linear span value $2^n - 2$, only very few span $n$ sequences obtain the linear span value $2^n - 2 - 3n$ and in some cases all the linear spans are greater than $2^n - 2 - 3n$. We summarize the above discussions in the following two properties.

**Property 2** *For all newly found span $n$ sequences (including reverse span $n$ sequences), for $7 \leq n \leq 20$ and $n$ is a prime number, then the WG, 5-term, 3-term, and Kasami span $n$ sequences have the linear spans which take the following three values*

$$\{2^n - 2 - 2n, 2^n - 2 - n, 2^n - 2\}.$$

**Property 3** *For $7 \leq n \leq 20$ and all the other cases, except for those in Property 2, listed below, the linear span, denoted as $LS$, is bounded by*

$$2^n - 2 - 4n \leq LS \leq 2^n - 2$$

*for all WG span $n$ sequences, 5-term span $n$ sequences, 3-term span $n$ sequences and Kasami span $n$ sequences when $n$ is a composite number and their respective reverse span $n$ sequences for any $n$.*

## 6 Conclusions & Discussions

In this paper, we first introduced the notion of random NLFSR-sequences and proved the expected period of a random NLFSR-sequence for any initial state. We then studied a span $n$ sequence generation technique using a nonlinear feedback shift register with the nonlinear WG transformation as the feedback function. Our computational results show that using WG transformations and varying all $t$-tap positions, a number of new span $n$ sequences can be generated by an exhaustive search and the maximum number of span $n$ sequences can be obtained if about half of the length of the shift register many tap positions participate in the WG transformation. In this approach, a span $n$ sequence is searched according to a construction of a nonlinear feedback function and the success probability to obtain a span $n$ sequence using the WG transformations is greater than the success probability to obtain a span $n$ sequence in a random way for $n \approx 2t$. Moreover, we generalized the technique for producing span $n$ sequences by considering any permutation instead of the WG permutation or by considering other functions instead of the WG transformations. In the generalized method, three-term functions, five-term functions, and Kasami functions have been considered and the number of span $n$ sequences is reported for each function. Finally, we presented the linear span property of newly generated span $n$ sequences using the aforementioned functions and gave a summary of the bounds of the linear span for different values of $t$. The linear span of the new span $n$ sequences lies between the suboptimal $2^n - 2 - 4n$ and optimal $2^n - 2$. We have noticed that the majority of span $n$ sequences have an optimal linear span. The new span $n$ sequences with optimal linear span or span $n$ sequences produced using the given method can be used as building blocks to design PRSGs and stream ciphers.

## References

[1] E.R. Berlekamp. *Algebraic Coding Theory*, McGraw-Hill, New York, ch. 7, 1968.

[2] N.G. de Bruijn. A Combinatorial Problem, *Proc. Koninklijke Nederlandse Akademie v. Wetenschappen*, Vol. 49 pp. 758 –764, 1946.

[3] C. Banderier, and R.P. Dobrow. A Generalized Cover Time for Random Walks on Graphs, *In Formal Power Series and Algebraic Combinatorics (Moscow)*, pp. 113-124, Springer, 2000.

[4] A.H. Chan, R.A. Games, and E.L. Key. On the Complexities of de Bruijn Sequences, *Journal of Combinatorial Theory, Series A*, Vol. 33, No. 3, pp. 233 - 246, 1982.

[5] A.H. Chan, R.A. Games, and J.J. Rushanan. On Quadratic $m$-sequences, In *IEEE International Symposium on Information Theory* , pp. 364, July 1994.

[6] C. De Canniére, O. Dunkelman and M. Knězević. KATAN and KTANTAN A Family of Small and Efficient Hardware-Oriented Block Ciphers, *Cryptographic Hardware and Embedded Systems - CHES 2009*, LNCS, Vol. 5747, pp. 272-288, Springer, 2009.

[7] L. Chen, and G. Gong. Communication System Security, Lecture notes, will be published in CRC Press.

[8] J. Dillon and H. Dobbertin. New Cyclic Difference sets with Singer parameters, *Finite Fields and Their Application,* 10(2004), pp. 342-389, August 1999.

[9] H. Dobbertin. Kasami Power Functions, Permutation Polynomials and Cyclic Difference Sets, *Proceedings of the NATO-A.S.I. Workshop "Difference sets, sequences and their correlation properties"*, Bad Windsheim, August 3-14, 1998, Kluwer, Dordrecht, pp. 133 – 158, 1999.

[10] eSTREAM - The ECRYPT Stream Cipher Project, `http://www.ecrypt.eu.org/stream/`.

[11] S.W. Golomb. *Shift Register Sequences*, Aegean Park Press, Laguna Hills, CA, USA, 1981.

[12] S.W. Golomb. On the Classification of Balanced Binary Sequences of Period $2^n - 1$, *IEEE Transformation on Information Theory,* Vol. 26, No. 6, pp. 730-732, November 1980.

[13] S.W. Golomb, and G. Gong. *Signal Design for Good Correlation: For Wireless Communication, Cryptography, and Radar.* Cambridge University Press, New York, NY, USA, 2004.

[14] G. Gong, and A. Youssef. Cryptographic Properties of the Welch-Gong Transformation Sequence Generators, *IEEE Transactions on Information Theory*, Vol. 48, No. 11, pp. 2837-2846, November 2002.

[15] G. Gong. Randomness and Representation of Span-n-sequences, In *Proceedings of the 2007 international conference on Sequences, subsequences, and consequences*, SSC'07, pp. 192–203, Springer-Verlag, 2007.

[16] I.J. Good. Normal Recurring Decimals, *Journal of London Math. Soc.*, Vol. 21(Part 3), 1946.

[17] T. Kasami. Weight Distributions of Bose-Chaudhuri-Hocquenghem Codes, *Combinatorial Mathematics and its Applications*, (Bose, R.S. and Dowling, T.A., eds.), Univ. of North Carolina Press, Chapel Hill, NC, 1969.

[18] R. Lidl, and H. Niederreiter. *Finite Fields*, Cambridge University Press, 1997.

[19] J.L. Massey. Shift-Register Synthesis and BCH Decoding, *IEEE Transactions on Information Theory* Vol. 15, No. 1, pp. 122–127, 1969.

[20] R. Motwani, and P. Raghavan. *Randomized algorithms*, Cambridge University Press, 1995.

[21] G.L. Mayhew, and S.W. Golomb. Linear Spans of Modified de Bruijn Sequences, *IEEE Transactions on Information Theory*, Vol. 36, No. 5, pp. 1166 –1167, September 1990.

[22] G.L. Mayhew. Weight Class Distributions of de Bruijn Sequences, *Discrete Math.*, Vol. 126, pp. 425–429, March 1994.

[23] G.L. Mayhew, and S.W. Golomb. Characterizations of Generators for Modified de Bruijn Sequences, *Adv. Appl. Math.*, Vol. 13, pp. 454–461, December 1992.

[24] G.L Mayhew. Clues to the Hidden Nature of de Bruijn Sequences, *Computers and Mathematics with Applications*, Vol. 39, No. 11, pp. 57-65, June 2000.

[25] A.J. Menezes, P.C. van Oorschot, and S.A. Vanstone. *Handbook of Applied Cryptography*, CRC Press, 1997.

[26] J. Ling-Fung Ng. Binary Nonlinear Feedback Shift Register Sequence Generator using the Trace Function, Master Thesis, the University of Waterloo, 2005.

[27] J.S. No, S.W. Golomb, G. Gong, H.K. Lee, and P. Gaal. New Binary Pseudorandom Sequences of Period $2^n - 1$ with Ideal Autocorrelation, *IEEE Transactions on Information Theory*, Vol. 44, No. 2, pp. 814817, March 1998.

[28] R.A. Rueppel. *Analysis and Design of Stream Ciphers*, Springer-Verlag, 1986.

[29] `http://www.comsec.uwaterloo.ca/~kmandal/WG-Span-n/index.html`.

# A Appendix

In this appendix, we present the upper and lower bounds of the linear span of new span $n$ sequences generated using WG transformations, three-term, five-term, and Kasami functions for different $n$ and $t$ and give all new span $n$ sequences generated using WG transformations for $t = 5$. All new span $n$ sequences generated using WG transformations with $t = 7, 8, 10,$ and 11 is presented in [29].

Table 10: The bounds of the linear span of WG span $n$ sequences using rec. rel. (3)

| Range on $n$ | $t$ | Upper bound of LS | Lower bound of LS |
|---|---|---|---|
| $7 \leq n \leq 20$ | 5 | $2^n - 2$ | $2^n - 2 - 2n$ |
| $8 \leq n \leq 20$ | 7 | $2^n - 2$ | $2^n - 2 - 2n$ |
| $9 \leq n \leq 20$ | 8 | $2^n - 2$ | $2^n - 2 - 3n$ |
| $11 \leq n \leq 17$ | 10 | $2^n - 2$ | $2^n - 2 - 3n$ |
| $12 \leq n \leq 17$ | 11 | $2^n - 2$ | $2^n - 2 - 2n$ |

Table 11: The bounds of the linear span of WG span $n$ sequences using rec. rel. (4)

| Range on $n$ | $t$ | Upper bound of LS | Lower bound of LS |
|---|---|---|---|
| $7 \leq n \leq 20$ | 5 | $2^n - 2$ | $2^n - 2 - 2n$ |
| $8 \leq n \leq 20$ | 7 | $2^n - 2$ | $2^n - 2 - 3n$ |
| $9 \leq n \leq 20$ | 8 | $2^n - 2$ | $2^n - 2 - 2n$ |
| $11 \leq n \leq 17$ | 10 | $2^n - 2$ | $2^n - 2 - 4n$ |
| $12 \leq n \leq 16$ | 11 | $2^n - 2$ | $2^n - 2 - 2n$ |

Table 12: The bounds of the linear span of Kasami span $n$ sequences using rec. rel. (3)

| Range on $n$ | $t$ | Upper bound of LS | Lower bound of LS |
|---|---|---|---|
| $7 \le n \le 19$ | 5 | $2^n - 2$ | $2^n - 2 - 2n$ |
| $8 \le n \le 19$ | 7 | $2^n - 2$ | $2^n - 2 - 3n$ |
| $8 \le n \le 17$ | 9 | $2^n - 2$ | $2^n - 2 - 3n$ |
| $12 \le n \le 16$ | 11 | $2^n - 2$ | $2^n - 2 - 3n$ |

Table 13: The bounds of the linear span of three-term span $n$ sequences using rec. rel. (3)

| Range on $n$ | $t$ | Upper bound of LS | Lower bound of LS |
|---|---|---|---|
| $7 \le n \le 17$ | 5 | $2^n - 2$ | $2^n - 2 - 2n$ |
| $8 \le n \le 17$ | 7 | $2^n - 2$ | $2^n - 2 - 3n$ |
| $8 \le n \le 17$ | 9 | $2^n - 2$ | $2^n - 2 - 3n$ |
| $12 \le n \le 17$ | 11 | $2^n - 2$ | $2^n - 2 - 3n$ |

Table 14: The bounds of the liner span of five-term span $n$ sequences using rec. rel. (3)

| Range on $n$ | $t$ | Upper bound of LS | Lower bound of LS |
|---|---|---|---|
| $7 \le n \le 19$ | 5 | $2^n - 2$ | $2^n - 2 - 2n$ |
| $8 \le n \le 19$ | 7 | $2^n - 2$ | $2^n - 2 - 2n$ |
| $9 \le n \le 19$ | 8 | $2^n - 2$ | $2^n - 2 - 3n$ |
| $11 \le n \le 17$ | 10 | $2^n - 2$ | $2^n - 2 - 3n$ |
| $12 \le n \le 16$ | 11 | $2^n - 2$ | $2^n - 2 - 2n$ |

Table 15: Span $n$ sequences generated using WG7

| Length | Decimation | Polynomial | $t$-tap position |
|---|---|---|---|
| $n$ | $d$ | $(c_0, c_1, ..., c_5, c_6)$ | $(r_1, r_2, ..., r_6, r_7)$ |
| 8 | 5 | 1 1 0 0 0 0 0 | 1 2 3 4 5 6 7 |
| 9 | 1 | 1 0 1 1 1 1 1 | 1 2 3 4 5 6 7 |
| 10 | 27 | 1 1 1 1 0 1 1 | 1 2 3 4 5 6 7 |
| 11 | 1 | 1 1 1 1 0 1 1 | 1 2 3 5 8 9 10 |
| 12 | 1 | 1 0 1 1 1 0 0 | 1 2 4 5 8 10 11 |
| 13 | 9 | 1 1 0 0 1 0 1 | 1 2 3 4 5 6 8 |
| 14 | 43 | 1 1 1 0 1 1 1 | 1 2 3 4 5 6 7 |
| 15 | 31 | 1 1 0 0 0 0 0 | 1 2 3 4 7 12 14 |
| 16 | 27 | 1 1 1 1 0 1 1 | 1 2 3 5 6 8 14 |
| 17 | 1 | 1 0 1 1 1 0 0 | 1 2 3 4 7 9 13 |
| 18 | 1 | 1 0 1 1 1 0 0 | 1 2 3 4 6 9 16 |
| 19 | 3 | 1 1 1 1 1 1 0 | 1 2 3 5 7 15 17 |
| 20 | 31 | 1 1 1 1 1 1 0 | 1 2 3 7 8 12 15 |

Table 16: Span $n$ sequences generated using WG8

| Length | Decimation | Polynomial | $t$-tap position |
|---|---|---|---|
| $n$ | $d$ | $(c_0, c_1, ..., c_6, c_7)$ | $(r_1, r_2, ..., r_7, r_8)$ |
| | | | |
| 9 | 13 | 1 1 1 1 0 0 1 1 | 1 2 3 4 5 6 7 8 |
| 10 | 1 | 1 1 1 0 0 1 1 1 | 1 2 3 5 6 7 8 9 |
| 11 | 7 | 1 0 1 1 0 0 0 1 | 1 2 5 6 7 8 9 10 |
| 12 | 1 | 1 1 1 0 0 1 1 1 | 1 2 3 4 6 8 9 11 |
| 13 | 11 | 1 0 1 0 1 1 1 1 | 1 2 3 4 5 6 8 10 |
| 14 | 1 | 1 0 0 1 0 1 1 0 | 1 4 5 6 7 11 12 13 |
| 15 | 11 | 1 0 1 1 0 0 0 1 | 1 2 5 6 7 9 10 12 |
| 16 | 19 | 1 1 1 0 0 1 1 1 | 1 2 3 4 8 10 13 14 |
| 17 | 23 | 1 1 1 1 0 0 1 1 | 2 5 6 7 8 11 12 15 |
| 18 | 37 | 1 0 1 1 1 0 0 0 | 1 2 3 5 6 10 11 17 |
| 19 | 127 | 1 1 0 0 0 1 1 0 | 1 2 5 9 13 15 16 18 |
| 20 | 53 | 1 0 1 0 1 1 1 1 | 1 2 3 6 7 10 17 19 |

Table 17: Span $n$ sequences of stage $n = 8$ generated using rec. rel. (3)

| Decimation | Polynomial | Tap position |
|---|---|---|
| $d$ | $(c_0, c_1, c_2, c_3, c_4)$ | $(r_1, r_2, r_3, r_4, r_5)$ |
| 1 | 1 0 1 0 0 | 1 2 4 5 7 |
| 1 | 1 1 1 1 0 | 1 3 4 5 6 |
| 1 | 1 1 1 1 0 | 2 4 5 6 7 |
| 3 | 1 1 0 1 1 | 1 2 3 5 6 |
| 7 | 1 0 1 1 1 | 1 2 3 5 7 |
| 7 | 1 0 1 0 0 | 2 3 4 6 7 |
| 15 | 1 1 1 1 0 | 2 3 4 6 7 |

Table 18: Span $n$ sequences of stage $n = 10$ generated using rec. rel. (3)

| Decimation | Polynomial | Tap position |
|---|---|---|
| $d$ | $(c_0, c_1, c_2, c_3, c_4)$ | $(r_1, r_2, r_3, r_4, r_5)$ |
| 1 | 1 1 0 1 1 | 1 2 4 5 8 |
| 1 | 1 1 1 0 1 | 1 3 4 6 7 |
| 1 | 1 1 1 0 1 | 1 3 4 6 9 |
| 3 | 1 1 0 1 1 | 1 2 3 4 8 |
| 7 | 1 0 0 1 0 | 1 2 4 7 8 |
| 11 | 1 0 1 1 1 | 1 2 3 4 5 |
| 11 | 1 0 0 1 0 | 1 2 3 7 8 |
| 11 | 1 1 1 1 0 | 1 4 5 8 9 |

Table 19: Span $n$ sequences of stage $n = 9$ generated using rec. rel. (3)

| Decimation | Polynomial | Tap position |
|---|---|---|
| $d$ | $(c_0, c_1, c_2, c_3, c_4)$ | $(r_1, r_2, r_3, r_4, r_5)$ |
| 1 | 1 1 1 0 1 | 1 2 5 6 8 |
| 1 | 1 1 1 0 1 | 1 3 6 7 8 |
| 1 | 1 1 1 1 0 | 2 3 5 7 8 |
| 1 | 1 1 1 0 1 | 4 5 6 7 8 |
| 3 | 1 1 0 1 1 | 1 2 4 5 6 |
| 3 | 1 0 1 0 0 | 1 2 4 5 8 |
| 3 | 1 0 1 0 0 | 2 4 6 7 8 |
| 7 | 1 0 1 0 0 | 1 2 3 4 6 |
| 11 | 1 1 1 0 1 | 1 4 6 7 8 |
| 11 | 1 1 1 1 0 | 2 4 5 6 7 |
| 11 | 1 1 1 1 0 | 2 4 5 6 8 |
| 11 | 1 1 1 0 1 | 2 4 6 7 8 |
| 15 | 1 1 1 1 0 | 1 2 3 4 6 |
| 15 | 1 1 1 0 1 | 1 2 5 7 8 |

Table 20: Span $n$ sequences of stage $n = 11$ generated using rec. rel. (3)

| Decimation | Polynomial | Tap position |
|---|---|---|
| $d$ | $(c_0, c_1, c_2, c_3, c_4)$ | $(r_1, r_2, r_3, r_4, r_5)$ |
| 1 | 1 1 1 0 1 | 1 2 7 8 10 |
| 1 | 1 1 1 1 0 | 3 4 5 8 10 |
| 1 | 1 1 1 0 1 | 6 7 8 9 10 |
| 7 | 1 0 1 1 1 | 1 2 3 6 7 |
| 7 | 1 0 0 1 0 | 1 3 7 8 10 |
| 7 | 1 0 1 1 1 | 2 3 4 7 10 |
| 7 | 1 1 0 1 1 | 2 3 7 9 10 |
| 7 | 1 0 0 1 0 | 2 4 5 6 10 |
| 7 | 1 1 0 1 1 | 3 4 5 8 9 |
| 11 | 1 1 1 1 0 | 1 2 4 5 8 |
| 11 | 1 1 1 0 1 | 1 3 4 6 10 |

Table 21: Span $n$ sequences of stage $n = 12$ generated using rec. rel. (3)

| Decimation | Polynomial | Tap position |
|---|---|---|
| $d$ | $(c_0, c_1, c_2, c_3, c_4)$ | $(r_1, r_2, r_3, r_4, r_5)$ |
| 1 | 1 1 1 1 0 | 2 3 4 5 6 |
| 1 | 1 0 1 0 0 | 2 3 4 5 8 |
| 1 | 1 1 1 0 1 | 2 3 5 7 9 |
| 1 | 1 0 1 0 0 | 2 3 6 9 10 |
| 1 | 1 1 1 0 1 | 4 6 9 10 11 |
| 3 | 1 1 0 1 1 | 1 2 3 4 5 |
| 3 | 1 1 0 1 1 | 2 5 7 8 10 |
| 3 | 1 0 1 0 0 | 4 5 6 9 11 |
| 7 | 1 0 1 0 0 | 1 2 4 7 8 |
| 7 | 1 1 0 1 1 | 1 2 5 6 8 |
| 11 | 1 0 0 1 0 | 1 3 4 6 10 |
| 11 | 1 1 1 0 1 | 1 3 4 9 11 |
| 11 | 1 1 1 1 0 | 1 4 5 8 9 |
| 11 | 1 1 1 0 1 | 2 3 6 7 10 |
| 11 | 1 1 1 1 0 | 3 5 7 8 9 |
| 11 | 1 1 1 1 0 | 4 6 7 9 10 |
| 15 | 1 1 1 1 0 | 1 2 4 7 8 |

Table 22: Span $n$ sequences of stage $n = 13$ generated using rec. rel. (3)

| Decimation | Polynomial | Tap position |
|---|---|---|
| $d$ | $(c_0, c_1, c_2, c_3, c_4)$ | $(r_1, r_2, r_3, r_4, r_5)$ |
| 1 | 1 0 1 0 0 | 1 3 4 5 9 |
| 1 | 1 0 1 0 0 | 5 8 9 11 12 |
| 3 | 1 1 0 1 1 | 5 6 10 11 12 |
| 7 | 1 0 1 0 0 | 1 2 3 6 8 |
| 7 | 1 1 0 1 1 | 3 5 7 10 12 |
| 7 | 1 1 0 1 1 | 6 7 9 10 12 |
| 11 | 1 0 0 1 0 | 1 2 3 5 10 |
| 11 | 1 1 1 0 1 | 1 2 5 10 12 |
| 11 | 1 1 1 0 1 | 1 5 6 10 12 |
| 11 | 1 1 1 0 1 | 4 5 7 8 9 |
| 15 | 1 1 1 1 0 | 1 2 3 6 8 |

Table 24: Span $n$ sequences of stage $n = 15$ generated using rec. rel. (3)

| Decimation | Polynomial | Tap position |
|---|---|---|
| $d$ | $(c_0, c_1, c_2, c_3, c_4)$ | $(r_1, r_2, r_3, r_4, r_5)$ |
| 1 | 1 1 1 0 1 | 4 5 12 13 14 |
| 3 | 1 0 1 0 0 | 2 6 8 9 10 |
| 3 | 1 0 1 0 0 | 4 5 6 7 14 |
| 7 | 1 0 1 1 1 | 2 5 7 10 13 |
| 7 | 1 0 1 1 1 | 2 5 8 11 14 |
| 7 | 1 0 0 1 0 | 3 4 5 7 12 |
| 11 | 1 0 0 1 0 | 2 3 6 7 13 |
| 11 | 1 1 1 0 1 | 2 4 9 11 13 |
| 11 | 1 0 1 1 1 | 2 9 10 11 12 |
| 15 | 1 1 1 0 1 | 1 2 3 5 6 |

Table 25: Span $n$ sequences of stage $n = 18$ generated using rec. rel. (3)

| Decimation | Polynomial | Tap position |
|---|---|---|
| $d$ | $(c_0, c_1, c_2, c_3, c_4)$ | $(r_1, r_2, r_3, r_4, r_5)$ |
| 1 | 1 1 1 0 1 | 1 2 12 13 14 |
| 3 | 1 1 0 1 1 | 4 7 8 10 15 |
| 3 | 1 1 0 1 1 | 5 10 11 14 17 |
| 7 | 1 0 0 1 0 | 1 2 5 7 11 |
| 7 | 1 1 0 1 1 | 5 7 8 11 17 |
| 11 | 1 0 0 1 0 | 1 8 9 11 15 |
| 15 | 1 1 1 0 1 | 2 9 12 15 17 |

Table 23: Span $n$ sequences of stage $n = 14$ generated using rec. rel. (3)

| Decimation | Polynomial | Tap position |
|---|---|---|
| $d$ | $(c_0, c_1, c_2, c_3, c_4)$ | $(r_1, r_2, r_3, r_4, r_5)$ |
| 1 | 1 0 1 0 0 | 1 3 5 7 9 |
| 1 | 1 1 1 1 0 | 2 6 8 9 13 |
| 1 | 1 1 1 0 1 | 3 4 6 8 10 |
| 1 | 1 1 1 0 1 | 3 5 8 10 13 |
| 3 | 1 1 0 1 1 | 1 8 10 11 13 |
| 7 | 1 0 0 1 0 | 1 2 6 9 12 |
| 7 | 1 0 0 1 0 | 1 3 10 12 13 |
| 7 | 1 0 0 1 0 | 1 6 9 12 13 |
| 7 | 1 0 1 0 0 | 3 5 7 8 9 |
| 11 | 1 1 1 1 0 | 1 2 4 11 12 |
| 11 | 1 1 1 1 0 | 1 2 9 10 11 |
| 15 | 1 1 1 0 1 | 3 5 6 8 13 |
| 15 | 1 1 1 1 0 | 3 5 7 8 9 |

Table 26: Span $n$ sequences of stage $n = 16$ generated using rec. rel. (3)

| Decimation | Polynomial | Tap position |
|---|---|---|
| $d$ | $(c_0, c_1, c_2, c_3, c_4)$ | $(r_1, r_2, r_3, r_4, r_5)$ |
| 1 | 1 1 0 1 1 | 1 10 11 12 14 |
| 1 | 1 1 1 0 1 | 1 10 11 12 14 |
| 15 | 1 1 1 0 1 | 3 6 9 12 14 |

Table 27: Span $n$ sequences of stage $n = 17$ generated using rec. rel. (3)

| Decimation | Polynomial | Tap position |
|---|---|---|
| $d$ | $(c_0, c_1, c_2, c_3, c_4)$ | $(r_1, r_2, r_3, r_4, r_5)$ |
| 3 | 1 0 1 0 0 | 1 6 7 8 9 |
| 3 | 1 1 0 1 1 | 4 7 8 9 12 |
| 7 | 1 0 1 0 0 | 1 3 12 13 14 |
| 7 | 1 1 0 1 1 | 1 4 10 11 13 |
| 7 | 1 0 0 1 0 | 1 5 11 12 13 |
| 11 | 1 1 1 0 1 | 1 3 6 12 13 |
| 15 | 1 1 1 1 0 | 1 3 12 13 14 |

Table 28: Span $n$ sequences of stage $n = 20$ generated using rec. rel. (3)

| Decimation | Polynomial | Tap position |
|---|---|---|
| $d$ | $(c_0, c_1, c_2, c_3, c_4)$ | $(r_1, r_2, r_3, r_4, r_5)$ |
| 1 | 1 1 1 0 1 | 5 10 12 18 19 |