A Class of FSRs and Their Adjacency Graphs

Ming Li Dongdai Lin

State Key Laboratory of Information Security,
Institute of Information Engineering,
Chinese Academy of Sciences, Beijing 100093, China
E-mail: liming@iie.ac.cn, ddlin@iie.ac.cn

August 24, 2014

Abstract

In this paper, We find a way to construct FSRs. The constructed FSRs can be depicted in many ways. They are just the FSRs whose characteristic polynomial can be written as $g = (x_0 + x_1) * f$ for some f. Their adjacency graphs do not contain self-loops. Further more, we can divide the vertexes in their adjacency graphs into two sets such that the edges are all between the two sets. The number of this class of FSRs is also considered. Besides, some applications in LFSRs and constructing full cycles are presented.

1 Introduction

Feedback shift registers (FSRs) are simple and efficient hardware devices, and have been used and studied for many years. Especially in cryptography, FSRs are the basic component in stream cipher. Despite the widely use and decades of research, some basic theories of FSRs have not been solved. For example, calculate the cycle structures and adjacency graphs of FSRs.

The cycle structure of a FSR determines the period of sequences that the FSR outputs. In cryptography, we need FSRs who output sequences with large period. So calculate the cycle structure is important both from theory and practice. The cycle structures of linear feedback shift registers (LFSRs) have been solved completely [6]. But for non-linear shift registers (NFSRs), we solved this problem only in some special cases [8][9][3]. LFSRs are replaced by NFSRs gradually in cryptography, for numerous attacks that LFSRs may suffer [10][11].

The adjacency graphs of FSRs can be used to construct full cycles. When we change the successor of two states that in different cycles and be conjugate with each other, we get a big cycle from two small cycles [6]. Do it repeatedly, we get a full cycle. But calculate the adjacency graph of a FSR may not be a easy thing. Even for LFSRs, there are no general ways [4]. It is a problem that may be more difficult than determine the cycle structure. Because cycle structures can be derived from adjacency graphs easily, but we have no idea how to do it reversely.

This paper is organized as follows.

In section 2, we present the basic knowledge about boolean functions, feedback shift registers, and adjacency graphs, and explain some notations we will use.

A way to construct $(n+1)$-stage FSRs from n-stage FSRs is presented in section 3. When combine the cycles in FSR_f and FSR_{f+1}, and treat these n-stage cycles as $(n+1)$-stage cycles, we get a $(n+1)$-stage FSR. It can be proved that the $(n+1)$-stage FSR obtained in this way is $\mathrm{FSR}_{(x_0+x_1)*f}$.

In section 4, adjacency graphs are considered. We define the FSR whose adjacency graph has some special property as dividable FSR. Then we show that dividable FSRs are just the FSRs that we constructed in section 3. The number of dividable FSRs is determined. For a dividable FSR, there are

no self-loops in its adjacency graph. But the reverse is not true. We present an example to illustrate it at the end of this section.

Some applications are introduced in section 5. For a linear boolean function f, the number of cycles in FSR_{f+1} is determined. Some ways to construct full cycles are suggested.

At the end, we conclude this paper.

2 Preliminaries

The purpose of this section is to briefly review boolean functions, feedback shift registers, and adjacency graphs respectively and explain some notations that we will use in this paper.

2.1 Boolean functions

Let $\mathbb{F}_2$ be the finite field of two-element, $\mathbb{F}_2^n$ be the vector space of dimension n over $\mathbb{F}_2$. A boolean function (or boolean polynomial) $f(x_0, x_1, \cdots, x_{n-1})$ in n variables is a map from $\mathbb{F}_2^n$ to $\mathbb{F}_2$. Especially, $x_0^{a_0} x_1^{a_1} \cdots x_{n-1}^{a_{n-1}}$ is a boolean function, which takes value 1 at the point $(a_0, a_1, \cdots, a_{n-1})$ and 0 otherwise.

For two boolean functions $f(x_0, x_1, \cdots, x_n)$ and $g(x_0, x_1, \cdots, x_m)$, we denote

$$f * g = f(g(x_0, x_1, \cdots, x_m), g(x_1, x_2 \cdots, x_{m+1}), \cdots, g(x_n, x_{n+1}, \cdots, x_{n+m})). \tag{1}$$

It can be verified, the operation $*$ is not commutative. If $h = f * g$, we say f is a left $*$-factor of h, and g is a right $*$-factor of h [5]. Further more, if f is a linear function, we say f is a left linear $*$-factor of h. Given a boolean function h, it is easy to find all the left linear $*$-factor of h [5]. This fact will be used in section 4.

A generalized division algorithm of Boolean functions was proposed by J.Mykkeltveit *et al* [3].

Lemma 1. *[3] Let $m, n \in \mathbb{N}^*$ with $m \leqslant n$. Let $g(x_0, x_1, \cdots, x_n)$ and $f(x_0, x_1, \cdots, x_m)$ be two Boolean functions. If $f(x_0, x_1, \cdots, x_m) = f_1(x_0, x_1, \cdots, x_{m-1}) + x_m$, then there exist unique Boolean functions $h_0(x_0, x_1, \cdots, x_{m-1}), h_1(x_0, x_1, \cdots, x_m), \cdots, h_{n-m}(x_0, x_1, \cdots, x_{n-1})$ and $r(x_0, x_1, \cdots, x_{m-1})$ such that*

$$g(x_0, x_1, \cdots, x_n) = \sum_{i=0}^{n-m} h_i(x_0, x_1, \cdots, x_{m+i-1}) f(x_i, x_{i+1}, \cdots, x_{m+i}) + r(x_0, x_1, \cdots, x_{m-1}). \tag{2}$$

If $r = 0$ we say g is divisible by f, denoted by $f \| g$. The following lemma is directly from above.

Lemma 2. *Let $f = f_1(x_0, x_1, \cdots, x_{n-1}) + x_n$, $g = g_1(x_0, x_1, \cdots, x_n) + x_{n+1}$ be two Boolean functions. Then $f \| g, f + 1 \| g$ if and only if $g = (x_0 + x_1) * f$.*

2.2 Feedback Shift Registers

A n-stage feedback shift register (FSR) consists of n binary storage cells and a characteristic polynomial f regulated by a single clock. We denote the FSR with characteristic polynomial f by FSR_f.

Given a initial state $\mathbf{X}_0 = (x_0, \cdots, x_{n-1})$, FSR_f will output a sequence $\underline{x} = x_0 x_1 \cdots$. It is well known that, FSR_f always output the periodic sequences no matter what the initial state is, if and only if f can be written as $f = x_0 + F(x_1, \cdots, x_{n-1}) + x_n$ for some F. In this case, we say FSR_f is nonsingular. Without specification, all the FSRs in this paper is nonsingular. Denote the set of sequence that FSR_f can generate by $G(f)$. It is easy to see, there are 2^n sequences in $G(f)$.

For n-stage FSR_f, when start from a initial state $\mathbf{X}_0$, FSR_f will generate a cycle $C = (\mathbf{X}_0, \mathbf{X}_1, \cdots, \mathbf{X}_l)$, where $\mathbf{X}_{i+1}$ is the next state of $\mathbf{X}_i$ for $i = 1, \cdots, l-1$ and $\mathbf{X}_0$ is the next state of $\mathbf{X}_l$, l is the length of the cycle. For simplicity, the cycle C can be written as $C = (x_0, x_1, \cdots, x_l)$, where x_i is the first component of $\mathbf{X}_i$. We call this notation **sequence-notation**. We warn that, the stage of the

cycle must be known when sequence-notation is used. Cycle C can be seen as an ordered set with element in $\mathbb{F}_2^n$. Sometimes, we do not discriminate between cycle $C = (\mathbf{X}_0, \mathbf{X}_1, \cdots, \mathbf{X}_l)$ and the set $\{\mathbf{X}_0, \mathbf{X}_1, \cdots, \mathbf{X}_l\}$.

From the above discussion, the set $\mathbb{F}_2^n$ is divided into cycles $C_1, \cdots, C_k$ by FSR_f. Reversely, it is easy to see, a division of $\mathbb{F}_2^n$ into cycles determines a n-stage FSR. So we can treat FSR_f as a set of cycles, and use the notation $\mathrm{FSR}_f = \{C_1, \cdots, C_k\}$.

A FSR is called a linear feedback shift register (LFSR) if its feedback function f is linear and nonlinear feedback shift register (NFSR) otherwise.

The maximum length of cycles in FSR_f is 2^n. In this case, FSR_f contains only one cycle. We call FSR_f a maximum-length FSR and the cycle a n-stage M-cycle or a full cycle. The output sequences which corresponding to the M-cycles are called M-sequences or DeBruijn sequences.

In the linear case, the 0-cycle $((0, \cdots, 0))$ which contains only the 0-state $(0, \cdots, 0)$ is always in LFSR_f. So the maximum length of cycles in LFSR_f is $2^n - 1$. In this case, LFSR_f contains two cycles. We call LFSR_f a maximum-length LFSR and the cycle which contains all the state in $\mathbb{F}_2^n$ except the 0-state a n-stage m-cycle. The output sequences which corresponding to the m-cycles are called m-sequences.

The generalized division algorithm introduced in section 2.1 provides an effective way to determine the inclusion relation of the sequence families of FSRs.

Lemma 3. *[3] Let g and f be the characteristic polynomial of two FSRs. Then $f \| g$ if and only if $G(f) \subseteq G(g)$.*

The following lemma can be derived from lemma 2 and lemma 3.

Lemma 4. *Let FSR_f be a n-stage FSR, FSR_g be a $(n+1)$-stage FSR. Then $G(f) \subseteq G(g), G(f+1) \subseteq G(g)$ if and only if $g = (x_0 + x_1) * f$.*

2.3 Adjacency Graphs

For a n-stage state $\mathbf{X} = (x_0, x_1, \cdots, x_{n-1})$, its conjugate $\widehat{\mathbf{X}}$ and companion $\widetilde{\mathbf{X}}$ are defined as $\widehat{\mathbf{X}} = (\overline{x}_0, x_1, \cdots, x_{n-1})$ and $\widetilde{\mathbf{X}} = (x_0, x_1, \cdots, \overline{x}_{n-1})$, where $\overline{x}$ denotes the binary complement of x. We call $(\mathbf{X}, \widehat{\mathbf{X}})$ a **conjugate pair**, $(\mathbf{X}, \widetilde{\mathbf{X}})$ a **companion pair**. Two cycles C_1 and C_2 are **adjacent** if they are disjoint and there exists a state $\mathbf{X}$ on C_1 whose conjugate $\widehat{\mathbf{X}}$ (or companion state $\widetilde{\mathbf{X}}$) is on C_2. It is well-known that two adjacent cycles C_1 and C_2 are joined into a single cycle when the successors of $\mathbf{X}$ and $\widehat{\mathbf{X}}$ are interchanged.

This is the basic idea of the cycle joining method introduced in [6]. The problem of determining conjugate pairs between cycles leads to the definition of adjacency graph.

Definition 1. *[12][13] For an FSR, its adjacency graph is an undirected graph where the vertexes correspond to the cycles in it, and there exists an edge between two vertexes if and only if they share a conjugate pair.*

In an adjacency graph, two vertexes may be connected by more than one edge. In this case, the edge between two vertexes sharing exactly m conjugate pairs can be labeled with an integer m [2].

At the end of this section, we present a property of adjacency graph, which will be used in section 4.

Theorem 1. *For any FSR, the adjacency graph is connected.*

Proof. Suppose for FSR_f, the adjacency graph is not connected. Denote the adjacency graph of FSR_f by G. Then we can find a proper subgraph H of G such that, H is a connected graph and there are no edges between the vertexes in H and the vertexes not in H. Using the cycle joining method we can join the cycles in H into a single cycle C. The cycle C has the property that when the state $\mathbf{X}$ is in C then $\widetilde{\mathbf{X}}$ is also in C. In the next paragraph we will show that a cycle with this property is a

full cycle (this conclusion has been left as a exercise in [2]), this will be contradict with H is a proper subgraph, and complete the proof.

For any state $\mathbf{Y} = (y_0, y_1, \cdots, y_{n-1}) \in \mathbb{F}_2^n$, we need to show that $\mathbf{Y}$ is in C. Let $\mathbf{X}_0 = (x_0, x_1, \cdots, x_{n-1})$ be a state in C. Then as the next state of $\mathbf{X}_0$, $(x_1, \cdots, x_{n-1}, 0)$ or $(x_1, \cdots, x_{n-1}, 1)$ is in C. Consider the property C has, we know that both $(x_1, \cdots, x_{n-1}, 0)$ and $(x_1, \cdots, x_{n-1}, 1)$ are in C. So $\mathbf{X}_1 = (x_1, \cdots, x_{n-1}, y_0)$ is in C. Similarly, $\mathbf{X}_2 = (x_2, \cdots, x_{n-1}, y_0, y_1), \cdots, \mathbf{X}_{n-1} = (x_{n-1}, y_0, \cdots, y_{n-2}), \mathbf{Y} = (y_0, y_1, \cdots, y_{n-1})$ is in C. □

3 A Class of FSRs

In this section, we present a way to construct a class of FSRs. First, we introduce some notations of cycles.

Let $C = (\mathbf{X}_0, \mathbf{X}_1, \cdots, \mathbf{X}_{l-1})$ be a n-stage cycle, where l is the length of the cycle and $\mathbf{X}_i = (x_i, x_{i+1}, \cdots, x_{i+n-1})$ is a n-stage state in the cycle for $i = 0, \cdots, l-1$. The subscribes are taken modulo l (similarly hereinafter). When using the sequence-notation, the cycle can be written as $C = (x_0, x_1, \cdots, x_{l-1})$. Now we can construct another cycle $C^+ = (\mathbf{X}_0^+, \mathbf{X}_1^+, \cdots, \mathbf{X}_{l-1}^+)$, where $\mathbf{X}_i^+ = (x_i, x_{i+1}, \cdots, x_{i+n-1}, x_{i+n})$, $i = 0, 1, \cdots, l-1$. It is easy to verify that the definition is meaningful. C^+ is a $(n+1)$-stage cycle of length l. When using sequence-notation, the cycle C^+ can be written as $C^+ = (x_0, x_1, \cdots, x_{l-1})$, the same notation as C. But we note that, C and C^+ are different cycles, for they be of different stages respectively n and $n+1$. We call C^+ the **extended cycle** of C.

We call a cycle C **prime cycle**, if there is no conjugate pair (companion pair) in C. For a prime cycle C, we can construct a $(n-1)$-stage cycle: $C^- = (\mathbf{X}_0^-, \mathbf{X}_1^-, \cdots, \mathbf{X}_{l-1}^-)$, where $\mathbf{X}_i^- = (x_i, x_{i+1}, \cdots, x_{i+n-2})$, $i = 0, 1, \cdots, l-1$. The definition is meaningful, because the states in C^- are all different from each other, and $\mathbf{X} \to \mathbf{Y}$ implies $\mathbf{X}^- \to \mathbf{Y}^-$. We warn that, C^- is meaningful if and only if C is a prime cycle. We call C^- the **reduced cycle** of C.

Theorem 2. *Let $FSR_f = \{C_1, C_2, \cdots, C_k\}, FSR_{f+1} = \{D_1, D_2, \cdots, D_t\}$ be two FSRs, then*

$$\{C_1^+, C_2^+, \cdots, C_k^+, D_1^+, D_2^+, \cdots, D_t^+\}$$

is a FSR whose characteristic polynomial is $g = (x_0 + x_1) * f$.

Proof. Let n be the stage of FSR_f and FSR_{f+1}. Let $\mathbf{X} \in \mathbb{F}_2^{n+1}$ be a $(n+1)$-stage state, write $\mathbf{X} = (x_0, x_1, \cdots, x_n)$. Define $\mathbf{X}^- = (x_0, x_1, \cdots, x_{n-1})$. Suppose for FSR_f, state $\mathbf{X}^-$ is in cycle C_i, and for FSR_{f+1} state $\mathbf{X}^-$ is in cycle D_j. If $f(x_0, x_1, \cdots, x_n) = 0$, then the state $\mathbf{X}$ is in C_i^+. If $f(x_0, x_1, \cdots, x_n) = 1$, then $f(x_0, x_1, \cdots, x_n) + 1 = 0$, so the state $\mathbf{X}$ is in D_j^+. In any case, $\mathbf{X}$ belong to some cycle of $\{C_1^+, C_2^+, \cdots, C_k^+, D_1^+, D_2^+, \cdots, D_t^+\}$. So $\{C_1^+, C_2^+, \cdots, C_k^+, D_1^+, D_2^+, \cdots, D_t^+\}$ is a division of $\mathbb{F}_2^{n+1}$ into cycles. Such division corresponding to a $(n+1)$-stage FSR, denoted as FSR_g

Consider the output sequences of FSR_g, FSR_f and FSR_{f+1}, we have $G(g) = G(f) \cup G(f+1)$. According to lemma 4, $g = (x_0 + x_1) * f$.

□

We present an example as the end of this section.

Example 1. *Let $f(x) = x_0 + x_1x_2 + x_3$, then*

$$\begin{aligned}
&FSR_f = \{C_1 = (000), C_2 = (001, 010, 100), C_3 = (011, 111, 110, 101)\} \\
&FSR_{f+1} = \{D_1 = (000, 001, 011, 110, 100),\ D_2 = (010, 101),\ D_3 = (111)\} \\
&FSR_g = \{C_1^+ = (0000),\ C_2^+ = (0010, 0100, 1001),\ C_3^+ = (0111, 1110, 1101, 1011), \\
&\qquad D_1^+ = (0001, 0011, 0110, 1100, 1000),\ D_2^+ = (0101, 1010),\ D_3^+ = (1111)\}
\end{aligned} \tag{3}$$

where $g = (x_0 + x_1) * f = x_0 + x_1 + x_1x_2 + x_2x_3 + x_3 + x_4$.

4 The Adjacency Graph of Dividable FSRs

In this section, we consider the adjacency graph of FSRs, whose characteristic polynomial g can be written as $g = (x_0 + x_1) * f$ for some f.

Definition 2. *A FSR is called dividable if we can divide the vertexes in the adjacency graph of the FSR into two sets, such that the edges are all between the two sets.*

Let FSR_g be a n-stage dividable FSR, the cycles in FSR_g can be divide into two sets $\mathcal{A}$ and $\mathcal{B}$ such that the edges in the adjacency graph of FSR_g all between $\mathcal{A}$ and $\mathcal{B}$, we denote $\mathrm{FSR}_g = (\mathcal{A}|\mathcal{B})$. Write $\mathcal{A} = \{C_1, C_2, \cdots, C_k\}, \mathcal{B} = \{D_1, D_2, \cdots, D_t\}$. Let $A = C_1 \cup C_2 \cup \cdots \cup C_k, B = D_1 \cup D_2 \cup \cdots \cup D_t$. Since there are 2^{n-1} edges in the adjacency graph of n-stage FSRs, and the edges are all between $\mathcal{A}$ and $\mathcal{B}$, we get $|A| = |B| = 2^{n-1}, B = \{\widehat{\mathbf{X}}|\mathbf{X} \in A\}, B = \{\widetilde{\mathbf{X}}|\mathbf{X} \in A\}$.

It is easy to see, a dividable FSR contains only prime cycles.

Theorem 3. *FSR_g is dividable if and only if $g = (x_0 + x_1) * f$ for some f.*

Proof. Suppose FSR_g is dividable. Let $\mathrm{FSR}_g = (\mathcal{A}|\mathcal{B})$, where $\mathcal{A} = \{C_1, C_2, \cdots, C_k\}, \mathcal{B} = \{D_1, D_2, \cdots, D_t\}$. Define $\mathcal{A}^- = \{C_1^-, C_2^-, \cdots, C_k^-\}, \mathcal{B}^- = \{D_1^-, D_2^-, \cdots, D_t^-\}$. Since the cycles in $\mathcal{A}$ and $\mathcal{B}$ are all prime cycles, the definition is meaningful. Let $A = C_1 \cup C_2 \cup \cdots \cup C_k, B = D_1 \cup D_2 \cup \cdots \cup D_t$. Because there is no companion pair in A (or B), we have the following conclusion:

$$\text{Let } \mathbf{X}_1, \mathbf{X}_2 \text{ be two states in } A \text{ (or } B\text{), then } \mathbf{X}_1 \neq \mathbf{X}_2 \text{ implies } \mathbf{X}_1^- \neq \mathbf{X}_2^-.$$

This means $\mathcal{A}^-$ and $\mathcal{B}^-$ are two divisions of $\mathbb{F}_2^{n-1}$ into cycles. So we get two $(n-1)$-stage FSRs, write $\mathcal{A}^- = \mathrm{FSR}_f$, $\mathcal{B}^- = \mathrm{FSR}_{f'}$.

Let $\mathbf{X} \in \mathbb{F}_2^{n-1}$ be a $(n-1)$-stage state. Suppose $\mathbf{X} \in C_i^-, \mathbf{X} \in D_j^-$ for some i and j. Let $\mathbf{Y}_1 \in C_i, \mathbf{Y}_2 \in D_j$ such that $\mathbf{Y}_1^- = \mathbf{X}, \mathbf{Y}_2^- = \mathbf{X}$. Then we have $\mathbf{Y}_1 = \mathbf{Y}_2$ or $\mathbf{Y}_1 = \widetilde{\mathbf{Y}}_2$. Since $\mathbf{Y}_1 \in A, \mathbf{Y}_2 \in B$ and $A \cap B = \emptyset$, we get $\mathbf{Y}_1 = \widetilde{\mathbf{Y}}_2$. This means the next state of $\mathbf{X}$ in C_i^- is companion with the next state of $\mathbf{X}$ in D_j^-. Since $\mathbf{X}$ is a arbitrary state in $\mathbb{F}_2^{n-1}$, we get $f' = f + 1$.

It is easy to see, $C_i = (C_i^-)^+, i = 1, 2, \cdots, k$, and $D_j = (D_j^-)^+, j = 1, 2, \cdots, t$. According to theorem 2, $g = (x_0 + x_1) * f$.

Conversely, suppose $g = (x_0 + x_1) * f$ for some f. Denote $\mathrm{FSR}_f = \{C_1, C_2, \cdots, C_k\}, FSR_{f+1} = \{D_1, D_2, \cdots, D_t\}$. According to theorem 2, we have $\mathrm{FSR}_g = \{C_1^+, C_2^+, \cdots, C_k^+, D_1^+, D_2^+, \cdots, D_t^+\}$. Define $\mathcal{A} = \{C_1^+, C_2^+, \cdots, C_k^+\}, \mathcal{B} = \{D_1^+, D_2^+, \cdots, D_t^+\}$, $A = C_1^+ \cup C_2^+ \cup \cdots \cup C_k^+, B = D_1^+ \cup D_2^+ \cup \cdots \cup D_t^+$.

Suppose there are two states $\mathbf{X}, \mathbf{Y}$ in A (or B) such that $\mathbf{X} = \widetilde{\mathbf{Y}}$. Then the state $\mathbf{X}^-$ $(= \mathbf{Y}^-)$ would appear twice in FSR_f (FSR_{f+1}), which is impossible. Further more, it is easy to see that $|A| = |B| = 2^n$. So we get $B = \{\widehat{\mathbf{X}}|\mathbf{X} \in A\}, B = \{\widetilde{\mathbf{X}}|\mathbf{X} \in A\}$. This implies that the edges in the adjacency graph of FSR_g are all between $\mathcal{A}$ and $\mathcal{B}$. So FSR_g is dividable and $\mathrm{FSR}_g = (\mathcal{A}|\mathcal{B})$. □

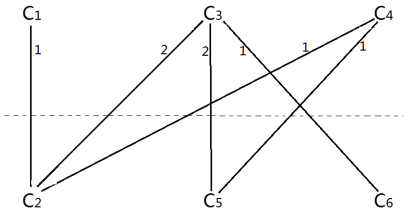
For a boolean function g, we can determine whether $x_0 + x_1$ is a linear $*$-factor of g or not easily [5]. So it is easy to determine whether FSR_g is dividable or not.

Example 2. *Let $g = x_0 + x_1x_2 + x_2x_3 + x_4$ be the characteristic polynomial of FSR_g. It is easy to see $g = (x_0 + x_1) * f$, where $f = x_0 + x_1 + x_2 + x_1x_2 + x_3$. So FSR_g is dividable. The cycles in FSR_g is as follows*

$$C_1 = (0000), C_2 = (0001, 0010, 0100, 1000), C_3 = (0011, 0111, 1110, 1100, 1001)$$

$$C_4 = (0101, 1010), C_5 = (0110, 1101, 1011), C_6 = (1111)$$

We can divide the cycles into two sets $\{C_1, C_3, C_4\} \cup \{C_2, C_5, C_6\}$. The adjacency graph of FSR_g is shown as follows.



We can see that there are no edges in $\{C_1, C_3, C_4\}$ *and in* $\{C_2, C_5, C_6\}$. *The edges are all between* $\{C_1, C_3, C_4\}$ *and* $\{C_2, C_5, C_6\}$.

Further more, we can calculate the number of dividable FSRs.

Theorem 4. *There are* $2^{2^{n-2}-1}$ *dividable FSRs in the* n*-stage FSRs.*

Proof. Let FSR_{f_1},FSR_{f_2} be two $(n-1)$-stage FSRs. We have

$$(x_0 + x_1) * f_1 = (x_0 + x_1) * f_2 \Leftrightarrow f_1 - f_2 = x_1 * (f_1 - f_2) \Leftrightarrow f_1 = f_2 \text{ or } f_1 = f_2 + 1.$$

So $(x_0 + x_1) * f_1 \neq (x_0 + x_1) * f_2$ if and only if $f_1 \neq f_2$ and $f_1 \neq f_2 + 1$.

Define a map ψ from the $(n-1)$-stage FSRs to the n-stage FSRs: $\psi(\mathrm{FSR}_f) = \mathrm{FSR}_{(x_0+x_1)*f}$. Then ψ is a two-to-one map. The image of ψ is just the n-stage dividable FSRs. So there are $2^{2^{n-2}-1}$ dividable FSRs in the n-stage FSRs. □

Theorem 5. *For a dividable FSR, there is only one way for us to divide the cycles in the FSR.*

Proof. Let FSR_g be dividable. Suppose $\mathrm{FSR}_g = (\mathcal{A}|\mathcal{B}) = (\mathcal{A}'|\mathcal{B}')$, and $\mathcal{A}' \neq \mathcal{A}$, $\mathcal{A}' \neq \mathcal{B}$. Define $\mathcal{C}_1 = \mathcal{A} \cap \mathcal{A}', \mathcal{C}_2 = \mathcal{A} \cap \mathcal{B}', \mathcal{C}_3 = \mathcal{B} \cap \mathcal{A}', \mathcal{C}_4 = \mathcal{B} \cap \mathcal{B}'$. The cycles in FSR_g are divided into four sets $\mathcal{C}_1, \mathcal{C}_2, \mathcal{C}_3, \mathcal{C}_4$. Consider the adjacency graph of FSR_g, because there are no edges in $\mathcal{A}$, and $\mathcal{C}_1, \mathcal{C}_2$ are subsets of $\mathcal{A}$, so there are no edges between $\mathcal{C}_1$ and $\mathcal{C}_2$. Similarly there are no edges between $\mathcal{C}_1$ and $\mathcal{C}_3$, $\mathcal{C}_4$ and $\mathcal{C}_2$, $\mathcal{C}_4$ and $\mathcal{C}_3$ in the adjacency graph. Define $\mathcal{D}_1 = \mathcal{C}_1 \cup \mathcal{C}_4, \mathcal{D}_2 = \mathcal{C}_2 \cup \mathcal{C}_3$. The above discussion shows that there are no edges between $\mathcal{D}_1$ and $\mathcal{D}_2$ in the adjacency graph. So the adjacency graph of FSR_g is not connected. This is contradict with Theorem 1. □

Next, we propose another class of FSRs which take dividable FSRs as its subclass.

Definition 3. *A FSR is called prime if there is no self-loop in the adjacency graph of the FSR.*

The prime FSRs are precisely the FSRs which contain only prime cycles.

It can be seen that, a dividable FSR is always prime. In reverse, whether a prime FSR is always dividable? For stage $n = 2, 3, 4$, the answer is yes. But for $n = 5$, we find a negative example.

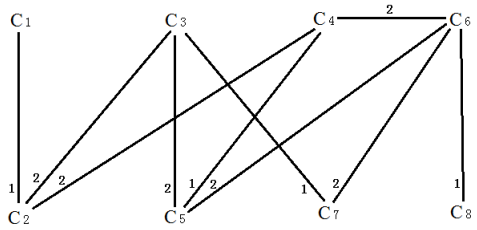
Example 3. *Let* $g = x_0 + x_1x_2x_4 + x_1x_3x_4 + x_5$, *the cycles in* FSR_g *are as follows*

$$C_1 = (00000), C_2 = (00001, 00010, 00100, 01000, 10000), C_3 = (00011, 00110, 01100, 11000, 10001),$$

$$C_4 = (00101, 01010, 10100, 01001, 10010), C_5 = (00111, 01110, 11100, 11001, 10011),$$

$$C_6 = (01011, 10111, 01111, 11110, 11101, 11010, 10101), C_7 = (01101, 11011, 10110), C_8 = (11111).$$

It can be verified that FSR_g *is a prime FSR, for there are no conjugate pairs in* C_i, $i = 1, 2, \cdots, 8$. *But* FSR_g *is not dividable, because* $x_0 + x_1$ *is not a left* $*$*-factor of* g. *The adjacency graph of* FSR_g *is shown below.*



Corollary 1. *There are at least* $2^{2^{n-2}-1}$ *prime FSRs in the n-stage FSRs.*

Calculate the number of prime FSRs and find a way to determine whether a FSR is prime or not is the next work we need to do.

5 Some Applications

5.1 Applications in LFSRs

In [1], D-morphism was proposed to construct FSRs. We restate it briefly in the way useful to us. D-morphism is defined as follows.

$$\begin{aligned} D: \quad & \mathbb{F}_2^{n+1} \to \mathbb{F}_2^n \\ (x_0, x_1, \cdots, x_n) & \mapsto (x_0 + x_1, x_1 + x_2, \cdots, x_{n-1} + x_n). \end{aligned} \tag{4}$$

Let C be a n-stage cycle, define $D^{-1}(C) = \{\mathbf{X} | D(\mathbf{X}) \in C\}$. It can be verified, for any state $\mathbf{X} \in D^{-1}(C)$ there is one and only one state $\mathbf{Y}$ in $D^{-1}(C)$ can be the successor of $\mathbf{X}$. Define $\mathbf{X} \to \mathbf{Y}$, the states in $D^{-1}(C)$ form cycles. Let $\mathrm{FSR}_f = \{C_1, C_2, \cdots, C_k\}$ be a n-stage FSR. Combine all the cycles in $D^{-1}(C_i), i = 1, \cdots, k$, we get a division of $\mathbb{F}_2^{n+1}$ into cycles. So we get a $(n+1)$-stage FSR, denoted as FSR_g. It was proved in [1] that: $g = f * (x_0 + x_1)$.

For a cycle $C = (\mathbf{X}_1, \cdots, \mathbf{X}_k)$, the weight of C is defined to be $W(C) = \sum_{i=1}^{k} x_i$, where x_i is the first component of $\mathbf{X}_i$. We have the following lemma, which can be derived from [1].

Lemma 5. *[1] Suppose there are s cycles of odd weight, t cycles of even weight in* FSR_f*. Then there are* $s + 2t$ *cycles in* $\mathrm{FSR}_{f*(x_0+x_1)}$*.*

Since the operation $*$ is not commutative, $(x_0 + x_1) * f \neq f * (x_0 + x_1)$ generally. But when f is a linear boolean function, we have $(x_0 + x_1) * f = f * (x_0 + x_1)$. So in the linear case, combine the conclusion in [1] with our conclusion, we can get more results.

Theorem 6. *Let f be a linear boolean function. Suppose there are t cycles of even weight in* FSR_f*. Then there are t cycles in* FSR_{f+1}*.*

Proof. Suppose there are s cycles of odd weight in FSR_f, and there are u cycles in FSR_{f+1}. Then there are $s + 2t$ cycles in $\mathrm{FSR}_{f*(x_0+x_1)}$ according to lemma 5. Further more, according to theorem 2 there are $s+u$ cycles in $\mathrm{FSR}_{(x_0+x_1)*f}$. Since f is a linear boolean function, $(x_0+x_1)*f = f*(x_0+x_1)$. So $\mathrm{FSR}_{f*(x_0+x_1)} = \mathrm{FSR}_{(x_0+x_1)*f}$, and $s + 2t = s + t + u$. Therefore $u = t$. □

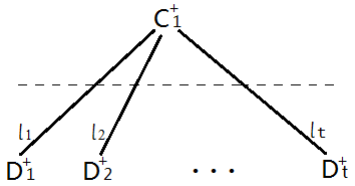
5.2 Applications in constructing full cycles

The adjacency graph of dividable FSR has a very good property, so we can use it to construct maximum-length FSRs.

First, we show a way to construct $(n+1)$-stage maximum-length FSRs from n-stage maximum-length FSRs.

Theorem 7. *Let FSR_f be a n-stage maximum-length FSR. Suppose $D_1, \cdots, D_t$ are the cycles in FSR_{f+1}. Let $(a_{i,1}, \cdots, a_{i,n})$ be a state in D_i, $i = 1, \cdots, t$. Define $g = (x_0+x_1)*f+\sum_{i=1}^{t} x_1^{a_{i,1}} \cdots x_n^{a_{i,n}}$. Then FSR_g is a $(n+1)$-stage maximum-length FSR.*

Proof. Let l_i be the number of states in $D_i, i = 1, \cdots, t$. Denote the full cycle in FSR_f by C_1. Then the adjacency graph of $\mathrm{FSR}_{(x_0+x_1)*f}$ can be depicted as follows.

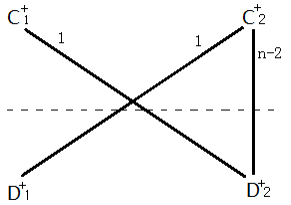


So if we choose a state from each cycle $D_1^+, \cdots, D_t^+$ arbitrarily, and change the successor of this state with its conjugate state, we get a full cycle. □

Next, we show another way to construct maximum-length FSRs, which start from maximum-length LFSRs. The conclusion we will get can be found in [3][7], but we use a totally different way, and our method is more simple and direct.

Let FSR_f be a n-stage maximum-length LFSR. In FSR_f, there are two cycles. One cycle is 0-cycle $((0, \cdots, 0))$ which contains only the 0-state $(0, \cdots, 0)$, denoted as C_1. The other cycle contains all the states except 0-state, denoted as C_2. The two cycles are all of even weight. According to theorem 6, there are two cycles in FSR_{f+1}. It is easy to see, the 1-cycle $((1, \cdots, 1))$ which contains only the 1-state $(1, \cdots, 1)$ is in FSR_{f+1}, denoted as D_1. So the other cycle in FSR_{f+1} contains all the states except 1-state, denoted as D_2.

$\mathrm{FSR}_{(x_0+x_1)*f} = (\{C_1^+, C_2^+\}|\{D_1^+, D_2^+\})$ is dividable. The adjacency graph of $\mathrm{FSR}_{(x_0+x_1)*f}$ can be depicted as follows.



Change the successor of conjugate pairs properly, we get full cycles.

Theorem 8. *[3] Let FSR_f be a n-stage maximum-length LFSR. Define*

$$g = (x_0 + x_1) * f + x_1^0 \cdots x_n^0 + x_1^1 \cdots x_n^1 + x_1^{a_1} \cdots x_n^{a_n}$$

where $(a_1, \cdots, a_n)$ is a arbitrary n-stage state except $(0, \cdots, 0)$ and $(1, \cdots, 1)$. Then FSR_g is a $(n+1)$-stage maximum-length FSR.

6 Conclusion

We find a way to construct FSRs. The cycle structure and adjacency graph of the constructed FSRs are considered. We also calculate the number of this FSRs. Besides, some applications in LFSRs and constructing full cycles are suggested.

References

[1] Abraham Lempel, On a Homomorphism of the de Bruijn Graph and Its Applications to the Design of Feedback Shift Registers. IEEE Transactions on computer. December 1970

[2] Harold Fredricksen, A Survey of Full Length Nonlinear Shift Register Cycle Algorithms. Society for Industrial and Applied Mathematics. April 1982

[3] Johannes Mykkeltveit, On the Cycle Structure of Some Nonlinear Shift Register Sequences. Information and Control. 1979

[4] Chaoyun Li, Xiangyong Zeng, Tor Helleseth, Chunlei Li, Lei Hu, The Properties of a Class of Linear FSRs and Their Applications to the Construction of Nonlinear FSRs. IEEE Transactions on Information Theory. May 2014

[5] Tian T, Qi W F, On decomposition of an NFSR into a cascade connection of two smaller NFSRs[J]. Submitted to Applicable Algebra in Engineering, Communication and Computing. 2014

[6] Solomon W. Golomb, Shift Register Sequences. San Francisco, Calif. Holden-Day, 1967

[7] Farhad Hemmati, A Large Class of Nonlinear Shift Register Sequences. IEEE Transactions on Information Theory. March 1982

[8] K. Kjeldsen, On the Cycle Structure of a Set of Nonlinear Shift Registers with Symmetric Feedback Functions. Journal of Combinatorial Theory. 1976

[9] Jan Soreng, The Periods of the Sequences Generated by Some Symmetric Shift Registers. Journal of Combinatorial Theory. 1976

[10] T. Siegenthaler, Decrypting a class of Stream Ciphers Using Ciphertext Only. IEEE Trans. Computers. Jan. 1985.

[11] N. Courtois and W. Meier, Algebraic attacks on stream ciphers with linear feedback. EUROCRYPT, 2003.

[12] E. R. Hauge and J. Mykkeltveit, On the classification of deBruijn sequences. Discrete Math. Jan. 1996.

[13] K. B. Magleby, The synthesis of nonlinear feedback shift registers. Stanford Electron. 1963.