Chapter 4

Stream ciphers

Having met a few basic forms of encryption, we now turn our attention to modern design of symmetric encryption algorithms. The meaning of the word symmetric is that Alice and Bob share the same secret key K, an assumption that we will later change when we study asymmetric encryption.

It is a well established fact that symmetric encryption algorithms are divided into two main categories, *block ciphers and stream ciphers*. Whereas block ciphers tend to encrypt a block of characters of a plaintext message using a fixed encryption transformation, a stream cipher encrypt individual characters of the plaintext using an encryption transformation that varies with time.

There is an extensive amount of theoretical knowledge on stream ciphers and on various design principles for stream ciphers. Going back a few years, most design principles involved socalled linear feedback shift registers, LFSRs. The ideas of modifying, combining, and disrupting LFSR sequences in different ways have been the traditional topic of research during many years. In this chapter we refer to any stream cipher built around LFSRs and producing one bit output on each clock as a *classical stream cipher design*.

The purpose of this chapter is to overview the most relevant design principles, of classes of stream ciphers, as well as overviewing the main cryptanalytical tools available. As a second motivation, we try to assess the strength of different classical designs.

We consider a binary additive stream cipher, i.e., a synchronous stream cipher in which the keystream, the plaintext, and the ciphertext are sequences of binary digits. The output sequence of the keystream generator, $\mathbf{z} = z_1, z_2, \ldots$ is added bitwise to the plaintext sequence $\mathbf{m} = m_1, m_2, \ldots$, producing the ciphertext $\mathbf{c} = c_1, c_2, \ldots$. The keystream generator is initialized through a secret key K, and hence, each key K will correspond to an output sequence. Since the key is shared between the transmitter and the receiver, the receiver can decrypt by adding the output of the keystream generator to the ciphertext and obtain the message sequence, see Figure 4.1.

The design goal is to efficiently produce random-looking sequences that are as "indistinguishable" as possible from truly random sequences. For a synchronous stream cipher, a known-plaintext attack (or chosen-plaintext or chosen-ciphertext) is equivalent to having access to the keystream $\mathbf{z} = z_1, z_2, \ldots, z_N$. So in general we assume that an output sequence $\mathbf{z}$ of length N from the keystream generator is known. The cryptanalyst Eve can

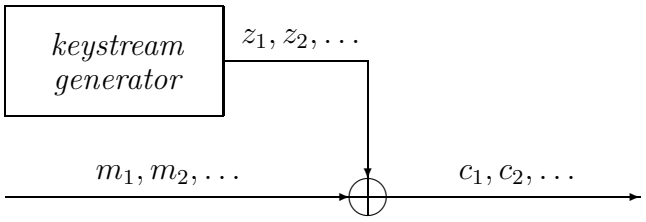

Figure 4.1: A binary additive stream ciphers

then try to perform different types of attacks. Basically, we consider the two main types of attacks.

- *Key recovery attack:* Eve tries to recover the value of the secret key K.
- *Distinguishing attack:* Eve tries to determine whether a given sequence $\mathbf{z} = z_1, z_2, \ldots, z_N$ is likely to have been generated from the considered stream cipher or whether it is just a truly random sequence. If a distinguisher, i.e., a box (algorithm) that can correctly answer the above question with high probability, can be built, we have a distinguishing attack.

It is clear that a distinguishing attack is a much weaker attack compared to a key recovery attack. In short, whereas a key recovery attack allows Eve to get access to any possible plaintext information she wants, the distinguishing attack can only give some limited amount of information to Eve. For example, if the plaintext message is one out of two possible, the distinguishing attack can tell Eve which of the two that was transmitted. Let us formalize a distinguishing attack a bit more.

Let $D(\mathbf{z})$ be an algorithm that takes as input a length N sequence $\mathbf{z}$ and as output gives one out of two possible answers, either "X" or "RANDOM".

Assume that by probability $1/2$ the sequence $\mathbf{z}$ is produced by generator X and by probability $1/2$ it is a purely random sequence. The probability that $D(\mathbf{z})$ correctly determines the origin of $\mathbf{z}$ is written $1/2 + \epsilon$. If ϵ is not very close to zero we say that $D(\mathbf{z})$ is a *distinguisher* for generator X and we have a distinguishing attack on generator X.

Example 4.1. *(Usage of distinguishing attack.) Assume that Alice sends one of N public images $\{I_1, I_2, \ldots, I_N\}$ to Bob. Eve knows the images but she does not know which image Alice sends to Bob because the image is encrypted using cipher X. She only observes the ciphertext* $\mathbf{c}$.

A fully secure cipher would not allow Eve to extract any information about anything useful from the ciphertext. But assume that the used cipher is suceptible to a distinguishing attack where $\epsilon = 1/2$. Then Eve can proceed as follows. Guess that the plaintext is the image I_1, i.e., $\mathbf{m} = I_1$. Calculate $\hat{\mathbf{z}} = \mathbf{m} + \mathbf{c}$ and run the distinguisher with $\hat{\mathbf{z}}$ as input. If the guess $\mathbf{m} = I_1$ was correct then $D(\hat{\mathbf{z}})$ will return the answer "X". If not, $D(\hat{\mathbf{z}})$ will return the answer "RANDOM". In this way Eve can check all images and eventually one of them will be the correct one and then the distinguisher will answer "X".

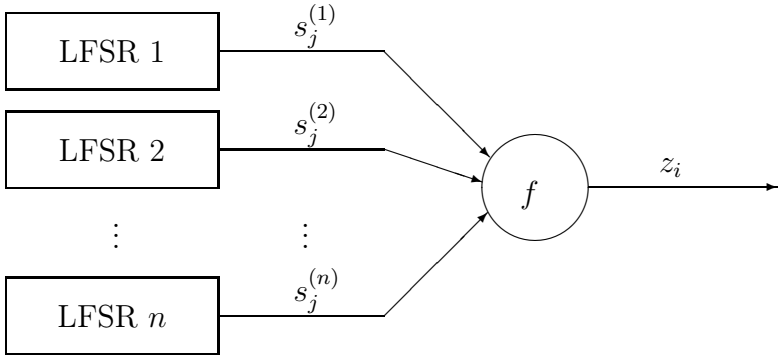

Figure 4.2: A nonlinear combining generator

As the key recovery attack is a special case of a distinguishing attack, we conclude that building a (synchronous) stream cipher reduces to the problem of building a generator that is resistent to all distinguishing attacks.

Note that there are essentially always both distinguishing attacks and key recovery attacks on a cipher. The most basic form of an attack on a cipher is an *exhaustive keysearch*. We try all possible keys and check if that key produces the given keystream sequence. The complexity of this approach is of course high, 2^k (or 2^{k-1} in average), if the number of key bits is k. So an attack is considered successful only if the complexity of performing it is considerably lower than 2^k key tests.

One of the most common building blocks for key stream generators is linear feedback shift registers, or LFSRs for short. A LFSR is a devise that produces a sequence of symbols, where the next symbol is created as a linear combination of previous symbols. Let us illustrate a possible simple design in the following example.

Example 4.2. *(Example of a class of generators) At time j, symbols from n different LFSR sequences, denoted $s_j^{(1)}, s_j^{(2)}, \ldots, s_j^{(n)}$, are used as input to a Boolean function f and it produces the output $z_1, z_2, \ldots$ by*

$$z_j = f(s_j^{(1)}, s_j^{(2)}, \ldots, s_j^{(n)}).$$

This is illustrated in Figure 4.2. The key K is equivalent to the initial states of the LFSRs.

4.1 Linear feedback shift register sequences

There are several reasons for the common use of LFSRs in stream cipher design. One is the fact that they are easy to implement in hardware. Another reason is the fact that LFSR sequences have very nice statistical properties and usually a large period.

4.1.1 Characterization of linear feedback shift register sequences

Definition 4.1. *A* linear feedback shift register *(LFSR) of length L consists of a register of L delay (storage) elements each capable of storing one element from $\mathbb{F}_q$, and a clock*

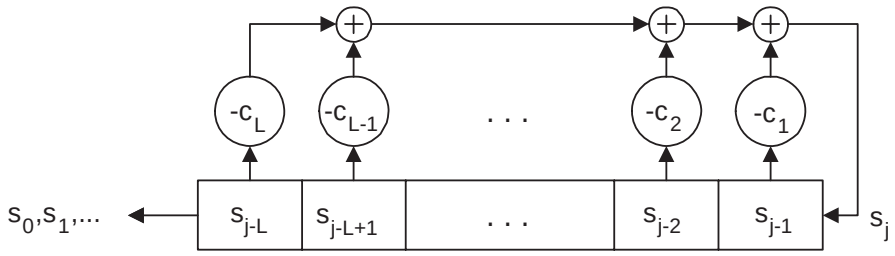

Figure 4.3: A linear feedback shift register

signal. In time unit j the content of the first delay element forms the output symbol s_j. When clocking, the register of delay elements is shifted one step and the new value of the last delay element is calculated as a linear function of the content of the register.

The linear function is described through the coefficients $c_1, c_2, \ldots, c_L \in \mathbb{F}_q$ and the recurrence relation is

$$s_j = -c_1 s_{j-1} - c_2 s_{j-2} - \cdots c_L s_{j-L},$$

for $j = L, L+1, \ldots$. By introducing $c_0 = 1$ we can write

$$\sum_{i=0}^{L} c_i s_{j-i} = 0, \text{ for } j = L, L+1, \ldots.$$

This is referred to as the *shift register equation*. The first L symbols of the sequence $s_0, s_1, \ldots, s_{L-1}$ form the *initial state*. See Figure 4.1.1.

The coefficients $c_0, c_1, \ldots, c_L$ are summarized in the *connection polynomial* $C(D)$ defined by

$$C(D) = 1 + c_1 D + c_2 D^2 + \cdots + c_L D^L.$$

Note that the degree of $C(D)$ is at most L, but can be smaller. When specifying the LFSR we sometimes need to give its length explicitly. We then write $< C(D), L >$ to denote the LFSR with connection polynomial $C(D)$ and length L.

Let us introduce the *D-transform* of a sequence $\mathbf{s} = s_0, s_1, s_2 \ldots$ as

$$S(D) = s_0 + s_1 D + s_2 D^2 + \cdots,$$

assuming $s_i \in \mathbb{F}_q$. The indeterminate D is the "delay" and its exponent indicate time. We assume $s_i = 0$ for $i < 0$. The set of all such sequences having the form

$$f(D) = \sum_{i=0}^{\infty} f_i D^i,$$

$f_i \in \mathbb{F}_q$, is denoted $\mathbb{F}_q[[D]]$ and called the *ring of formal power series*.

Theorem 4.1. *The set of sequences that can be generated by the LFSR with connection polynomial $C(D)$ is the set of sequences that have D-transform*

$$S(D) = \frac{P(D)}{C(D)},$$

where $P(D)$ is an arbitrary polynomial of degree at most $L-1$,

$$P(D) = p_0 + p_1 D + \ldots + p_{L-1} D^{L-1}.$$

Furthermore, the relation between the initial state of the LFSR and the $P(D)$ polynomial is given by the linear relation

$$\begin{pmatrix} p_0 \\ p_1 \\ \vdots \\ p_{L-1} \end{pmatrix} = \begin{pmatrix} 1 & 0 & \cdots & 0 \\ c_1 & 1 & \ldots & 0 \\ \vdots & \vdots & \vdots & \vdots \\ c_{L-1} & c_{L-2} & \ldots & 1 \end{pmatrix} \begin{pmatrix} s_0 \\ s_1 \\ \vdots \\ s_{L-1} \end{pmatrix}.$$

Proof. We start by assuming a sequence with D-transform $S(D)$ generated by the LFSR with connection polynomial $C(D)$. Then

$$\begin{aligned} C(D)S(D) &= \sum_{j=0}^{L} c_j D^j \sum_{i=0}^{\infty} s_i D^i && (4.1) \\ &= \sum_{j=0}^{L} \sum_{i=0}^{\infty} c_j s_i D^{i+j} = \sum_{i=0}^{\infty} \sum_{j=0}^{L} c_j s_i D^{i+j} = && (4.2) \\ \{k = i+j\} &= \sum_{i=0}^{\infty} (\sum_{j=0}^{L} c_j s_{k-j}) D^k. && (4.3) \\ & && (4.4) \end{aligned}$$

By observing that $\sum_{j=0}^{L} c_j s_{k-j} = 0$ whenever $k > L$ we can rewrite the above as

$$C(D)S(D) = \sum_{k=0}^{L-1} (\sum_{j=0}^{L} c_j s_{k-j}) D^k.$$

By introducing the notation $p_k = \sum_{j=0}^{L} c_j s_{k-j}$, where $0 \leq k \leq L-1$ we obtain

$$C(D)S(D) = \sum_{k=0}^{L-1} p_k D^k.$$

The right hand side is a degree $L-1$ polynomial that we write

$$P(D) = \sum_{j=0}^{L-1} p_j D^j.$$

This proves that every LFSR sequence $\mathbf{s}$ has a D-transform of the form $S(D) = P(D)/C(D)$.

We next need to prove that the opposite holds, i.e., for every polynomial $P(D)$ and connection polynomial $C(D)$ there is a unique sequence $\mathbf{s}$ that can be generated by the LFSR with connection polynomial $C(D)$. It is then enough to show that each choice of

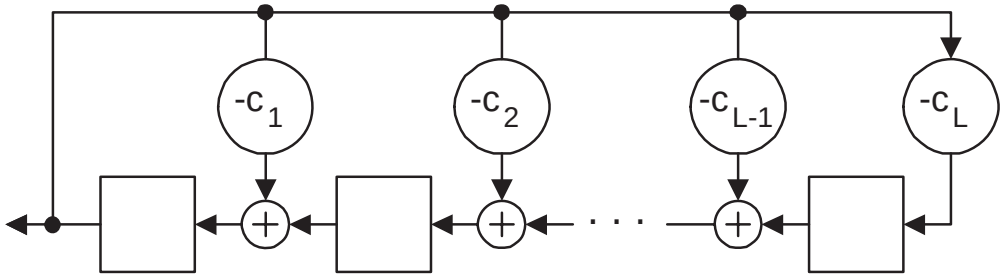

Figure 4.4: Multiplication of β by α.

$P(D)$ corresponds to a unique starting state $(s_0, s_1, \ldots, s_{L-1})$ for the LFSR. The relation is given above as $p_k = \sum_{j=0}^{L} c_j s_{k-j}$, which can be expressed as

$$\begin{pmatrix} p_0 \\ p_1 \\ \vdots \\ p_{L-1} \end{pmatrix} = \begin{pmatrix} 1 & 0 & \cdots & 0 \\ c_1 & 1 & \ldots & 0 \\ \vdots & \vdots & \vdots & \vdots \\ c_{L-1} & c_{L-2} & \ldots & 1 \end{pmatrix} \begin{pmatrix} s_0 \\ s_1 \\ \vdots \\ s_{L-1} \end{pmatrix}.$$

Clearly, the matrix is invertible and there is a unique solution for the starting state. □

We will next derive a link between LFSR sequences and extension fields, so let us return shortly to finite fields. Let $\pi(x)$ be an irreducible polynomial over $\mathbb{F}_q$ and assume that its coefficients are

$$\pi(x) = x^L + c_1 x^{L-1} + \cdots + c_L.$$

This means that $\pi(x)$ is the *reciprocal* polynomial of $C(D)$.

This polynomial can be used to construct the extension field $\mathbb{F}_{q^L}$ through $\pi(\alpha) = 0$ and the methods of Chapter 1. This means that an arbitrary element β from $\mathbb{F}_{q^L}$ can be expressed in a polynomial basis as

$$\beta = \beta_0 + \beta_1 \alpha + \cdots + \beta_{L-1} \alpha^{L-1},$$

where $\beta_0, \beta_1, \ldots \beta_{L-1} \in \mathbb{F}_q$.

Assume that the (unknown) element β is multiplied by the fixed element α. The result is

$$\alpha\beta = \beta_0 \alpha + \beta_1 \alpha^2 + \cdots + \beta_{L-1} \alpha^L.$$

Reducing α^L using $\pi(\alpha) = 0$ gives

$$\alpha\beta = -c_L \beta_{L-1} + (\beta_0 - c_{L-1}\beta_{L-1})\alpha + \cdots + (\beta_{L-2} - c_1 \beta_{L-1})\alpha^{L-1}.$$

Figure 4.4 describes an implementation.

If we consider the contents of the memory elements in the figure as a sequence $s_0, s_1, \ldots$, it is quickly checked that

$$s_j = -c_1 s_{j-1} - c_2 s_{j-2} - \cdots c_L s_{j-L},$$

when $j \geq L$. Furthermore, we have $p_0 = s_0$, $p_1 = s_1 + c_1 s_0$, etc, where $p_0, p_1, \ldots, p_{L-1}$ is the initial state in Figure 4.4. The sequence fulfills the shift register equation, but uses $p_0, p_1, \ldots p_{L-1}$ as initial state. This leads to the conclusion that the set of LFSR sequences, when $C(D)$ is irreducible, is exactly the set of sequences possible to produce by the implementation of multiplication of an element β by the fixed element α in $\mathbb{F}_{q^L}$.

In summary, the two implementations in Figure 4.3 and Figure 4.4 produce the same set of sequences in the case $C(D)$ irreducible. For a specific sequence specified as $S(D) = P(D)/C(D)$ the initial state in Figure 4.3 is the first L symbols whereas the same sequence is produced in Figure 4.4 if the initial state is $p_0, p_1, \ldots, p_{L-1}$.

A sequence $\mathbf{s} = \ldots, s_0, s_1, \ldots$ is called *periodic* if there is a positive integer T such that $s_i = s_{i+T}$, for all $i \geq 0$. The *period* is the least such positive integer T for which $s_i = s_{i+T}$, for all $i \geq 0$.

The state of our sequence generator in Figure 4.4 (equivalent to Figure 4.3) runs through different values. Clearly, the initial state will appear again after visiting a number of states. If $C(D)$ is irreducible the state corresponds to an element in $\mathbb{F}_{q^L}$, say β. The sequence of different states that we are entering is then

$$\beta, \alpha\beta, \alpha^2\beta, \ldots, \alpha^{T-1}\beta, \alpha^T\beta = \beta,$$

where T is the order or α.

If α is a primitive element (its order is $q^L - 1$), then obviously we will go trough all $q^L - 1$ different states and the sequence will have period $q^L - 1$. Such sequences are called *m-sequences* and they appear if and only if the polynomial $\pi(x)$ is a primitive polynomial. Recall that $\pi(x)$ is a primitive polynomial over $\mathbb{F}_q$ if α is a primitive element in $\mathbb{F}_{q^L}$ where $\pi(\alpha) = 0$.

Clearly, considering LFSR sequences where $\deg C(D) = L$, the period of a sequence is the same as the number of different states visited, before returning to the initial state.

Example 4.3. *Consider the LFSR sequences generated by a length* 4 *LFSR with connection polynomial* $C(D) = 1 + D + D^2 + D^3 + D^4$ *in* $\mathbb{F}_2$. *The polynomial* $C(D)$ *is irreducible. Starting in initial state* (0001)*, we return to the same state after 5 clockings of the LFSR. Examining this further we see that there are three cycles of length 5 and one of length one (the zero cycle).*

If we switch representation to express the sequence generation through multiplication in $\mathbb{F}_{2^4}$*, we have* $\pi(x) = x^4 + x^3 + x^2 + x + 1$ *and* $\pi(\alpha) = 0$. *As an explanation to the behaviour above we can see that* $\alpha^5 = 1$ *and* $\mathrm{ord}(\alpha) = 5$. *So starting in any nonzero state* $\beta \in \mathbb{F}_{2^4}$*, we will jump between the states*

$$\beta, \alpha\beta, \alpha^2\beta, \alpha^3\beta, \alpha^4\beta, \alpha^5\beta = \beta,$$

and after 5 clockings we are back where we started. The length one cycle corresponds to the zero state.

Example 4.4. *Consider the LFSR sequences generated by the LFSR connection polynomial* $C(D) = 1 + D + D^4$, *in* $\mathbb{F}_2$. *The polynomial* $C(D)$ *is irreducible. Starting in an initial state, we return after 15 clockings of the LFSR. The polynomial* $C(D)$ *is a primitive polynomial.*

Switching representation to multiplication in $\mathbb{F}_{2^4}$*, we have* $\pi(x) = x^4+x^3+1$ *and* $\pi(\alpha) = 0$*. We can see that* $\text{ord}(\alpha) = 15$*. So starting in any nonzero state* $\beta \in \mathbb{F}_{2^4}$*, we will jump between the all nonzero states before returning to where we started.*

4.1.2 Determining the cycle set for an LFSR

We will now take a closer look at the different state cycles that will appear for an arbitrary LFSR.

Let the notation $[s_0, s_1, \ldots, s_{T-1}]^\infty$ denote the periodic and causal sequence

$$s_0, s_1, \ldots, s_{T-1}, s_0, s_1, \ldots, s_{T-1}, s_0, \ldots,$$

where $s_i \in \mathbb{F}_q$, $i = 0, 1, \ldots, T-1$. Similarly, let $(s_0, s_1, \ldots, s_{N-1})$ denote a sequence where the first N symbols are $s_0, s_1, \ldots, s_{N-1}$ (and the upcoming symbols are not defined), where $s_i \in \mathbb{F}_q$, $i = 0, 1, \ldots, N-1$.

We note that if $\mathbf{s} = [1, 0, 0, \ldots, 0]^\infty$ then

$$S(D) = 1 + D^T + D^{2T} + \cdots = \frac{1}{1 - D^T}.$$

Similar, if $\mathbf{s} = [0, 1, 0, \ldots, 0]^\infty$ then

$$S(D) = D + D^{T+1} + D^{2T+1} + \cdots = \frac{D}{1 - D^T}$$

and, in general, if $\mathbf{s} = [s_0, s_1, \ldots, s_{T-1}]^\infty$ then

$$S(D) = \frac{s_0}{1 - D^T} + \frac{s_1 D}{1 - D^T} + \ldots = \frac{s_0 + s_1 D + s_2 D^2 + \ldots s_{T-1} D^{T-1}}{1 - D^T}. \tag{4.5}$$

Example 4.5.
a)

$$S(D) = \frac{D}{1 + D^2}$$

is the D-tranform of the sequence $[0, 1]^\infty$ *in* $\mathbb{F}_2$*.*

b)

$$S(D) = \frac{D^2}{1 + D^2} = D \cdot \frac{D}{1 + D^2}$$

is the D-tranform of the sequence $0, [0, 1]^\infty$ *in* $\mathbb{F}_2$*.*

c)
In $\mathbb{F}_2$ *the sequence* $[1, 1, 0]^\infty$ *has D-transform*

$$\frac{1 + D}{1 + D^3} = \frac{1}{1 + D + D^2}.$$

d)
However, In $\mathbb{F}_3$ *the sequence* $[1, 1, 0]^\infty$ *has D-transform*

$$\frac{1 + D}{1 - D^3} = \frac{1 + D}{(1 - D)^3}.$$

Definition 4.2. *The* period of a polynomial $C(D)$ *is the least positive number* T *such that* $C(D)|(1-D^T)$.

The period of a polynomial is easily calculated by division of 1 by $C(D)$ and continuing until the we receive the first remainder of the form $1 \cdot D^N$. Then the period is $T = N$.

Example 4.6. *Calculate the period of the polynomial* $C(D) = 3 + D^2$ *in* $\mathbb{F}_5$.

Solution: *We divide 1 by* $C(D)$ *and cancel out the lowest degree terms. The first reminder of the form* $1 \cdot D^N$ *indicate the period. Long division written in the following form:*

$$\begin{aligned} 1 &= 2 \cdot (3 + D^2) + (-2D^2) \\ 3D^2 &= D^2 \cdot (3 + D^2) + (-D^4) \\ 4D^4 &= 3D^4 \cdot (3 + D^2) + (-3D^6) \\ 2D^6 &= 4D^6 \cdot (3 + D^2) + (-4D^8) \end{aligned}$$

The remainder $(-4D^8) = D^8$ *has the required form* $1 \cdot D^N$. *The period of* $C(D)$ *is* 8.

Theorem 4.2. *If* $\gcd(C(D), P(D)) = 1$ *then the connection polynomial* $C(D)$ *and the sequence* $\mathbf{s}$ *with D-transform*

$$S(D) = \frac{P(D)}{C(D)}$$

have the same period (the period of $\mathbf{s}$ *is the same as the period of the polynomial* $C(D)$*).*

Proof. to be added. □

Example 4.7. *Determine the shortest LFSR generating the periodic sequence* $[1010011]^\infty$ *in* $\mathbb{F}_2$.

Solution: *The shortest LFSR will have a connection polynomial* $C(D)$ *and the D-transform of the periodic sequence* $[1010011]^\infty$ *can be written as* $S(D) = \frac{P(D)}{C(D)}$. *On the other hand, the D-transform can also be written as in (4.5), so we must have*

$$\frac{s_0 + s_1 D + s_2 D^2 + \ldots s_{T-1} D^{T-1}}{1 - D^T} = \frac{P(D)Q(D)}{C(D)Q(D)},$$

for some polynomial $Q(D)$. *Furthermore* $Q(D) = \gcd(s_0 + s_1 D + s_2 D^2 + \ldots s_{T-1} D^{T-1}, 1 - D^T)$. *This means that after computing* $Q(D)$, *the shortest LFSR is described by the connection polynomial* $C(D) = (1 - D^T)/Q(D)$.

In our case

$$S(D) = \frac{1 + D^2 + D^5 + D^6}{1 + D^7}.$$

Using Euclidean algorithm, we compute

$$Q(D) = \gcd(1 + D^2 + D^5 + D^6, 1 + D^7) = 1 + D + D^2 + D^4$$

in $\mathbb{F}_2$. *The connection polynomial for the shortest LFSR generating the sequence is then*

$$C(D) = \frac{1 + D^7}{1 + D + D^2 + D^4} = 1 + D + D^3,$$

and the D-transform of the sequence can be written as $S(D) = (1 + D + D^2)/(1 + D + D^3)$.

Theorem 4.3. *If two sequences,* $\mathbf{s}_A$ *and* $\mathbf{s}_B$*, with periods* T_A *and* T_B *have D-transforms*

$$S_A(D) = \frac{P_A(D)}{C_A(D)}, S_B(D) = \frac{P_B(D)}{C_B(D)},$$

then the sum of the sequences $\mathbf{s} = \mathbf{s}_A + \mathbf{s}_B$ *has D-transform* $S(D) = S_A(D) + S_B(D)$ *and period* $\mathrm{lcm}(T_A, T_B)$*, assuming* $\gcd(P_A(D), C_A(D)) = 1$*,* $\gcd(P_B(D), C_B(D)) = 1$*,* $\gcd(C_A(D), C_B(D)) = 1$*.*

Proof. to be added. □

Example 4.8. *Let* $\mathbf{s}_A$ *be the periodic LFSR sequence generated by* $C(D) = 1 + D + D^4$ *and initial state* $(s_0, s_1, \ldots, s_3) = 1000$*. Let* $\mathbf{s}_B$ *be the periodic LFSR sequence generated by* $C(D) = 1+D+D^6$ *and initial state* $(s_0, s_1, \ldots, s_5) = 00001$*. Compute the D-transform of the sequence* $\mathbf{s} = \mathbf{s}_A + \mathbf{s}_B$ *and its period.*

Solution: *We start by computing the* $P_A(D)$ *polynomial. We get* $p_0 = s_0 = 1$*,* $p_1 = c_1 s_0 + s_1 = 1$*, and then* $p_2 = 0, p_3 = 0$*. We have*

$$S_A(D) = \frac{1 + D}{1 + D + D^4}.$$

For the second sequence we proceed in the same way and get coefficients of the $P_B(D)$ *polynomial as* $p_0 = 0, p_1 = 0, \ldots, p_4 = 0$ *and* $p_5 = 1$*. We have*

$$S_B(D) = \frac{D^5}{1 + D + D^6}.$$

This gives $S(D)$ *as*

$$S(D) = S_A(D) + S_B(D) = \frac{(1 + D)(1 + D + D^6) + D^5(1 + D + D^4)}{(1 + D + D^6)(1 + D + D^4)}.$$

As $1+D+D^4$ *as well as* $1+D+D^6$ *are primitive polynomials,* $\mathbf{s}_A$ *has period* 15 *and* $\mathbf{s}_B$ *has period* 63*. Since* $\gcd(P_A(D), C_A(D)) = 1$*,* $\gcd(P_B(D), C_B(D)) = 1$*,* $\gcd(C_A(D), C_B(D)) = 1$ *we immediately get the period of the sequence* $\mathbf{s}$ *as* $\mathrm{lcm}(15, 63) = 105$*.*

Let us now focus on establishing exactly which different lengths of cycles that will appear for an LFSR with connection polynomial $C(D)$ over $\mathbb{F}_q$. To do this we introduce the *cycle set* for $C(D)$ (assuming $L = \deg C(D)$). The cycle set is written in the form $n_1(T_1) \oplus n_2(T_2) \oplus \ldots$. For example $1(1) \oplus 3(5)$, means that there is one cycle of length one and three cycles of length 5. Note that $n_1(T) \oplus n_2(T) = (n_1 + n_2)(T)$. Let us recall some already established facts.

1. If $C(D)$ is a primitive polynomial of degree L over $\mathbb{F}_q$ then the cycle set is $1(1) \oplus (1)(q^L - 1)$.

2. If $C(D)$ is an irreducible polynomial then the cycle set is $1(1) \oplus \frac{(q^L-1)}{T}(T)$, where T is the period of the polynomial $C(D)$ (or the order of α when $\pi(\alpha) = 0$).

We now give formulas for the remaining cases.

Theorem 4.4. *If $C(D) = C_1(D)^n$ then the cycle set of $C(D)$ is*

$$1(1) \oplus \frac{(q^{L_1}-1)}{T_1}(T_1) \oplus \frac{q^{L_1}(q^{L_1}-1)}{T_2}(T_2) \oplus \cdots \frac{q^{(n-1)L_1}(q^{L_1}-1)}{T_n}(T_n),$$

where $\deg C(D) = L$ *and* T_j *is the period of the polynomial* $C_1(D)^j$.

Theorem 4.5. *If $C_1(D)$ is irreducible with period T_1, then the period of the polynomial $C_1(D)^j$ is $T_j = p^m T_1$ where p is the characteristic of the field and m the integer satisfying $p^{m-1} < j \leq p^m$.*

Example 4.9. *Compute the cycle set for the connection polynomial $C(D) = (1+D+D^2)^3$ in $\mathbb{F}_2$.*

Solution: *The connection polynomial is in the form $C(D) = C_1(D)^3$ where $C_1(D) = 1 + D + D^2$ is irreducible (and primitive). The period of $C_1(D)$ is then $T_1 = 3$. Using the above theorem we compute the period of $C_1(D)^2$ to be $T_2 = 2T_1 = 6$, since $2^0 < 2 \leq 2^1$ and $m = 1$. Similarly, $T_3 = 2^2 T_1 = 12$. Finally, the formula in Theorem 4.4 gives a cycle set of the form*

$$1(1) \oplus \frac{(2^{L_1}-1)}{T_1}(T_1) \oplus \frac{2^{L_1}(2^{L_1}-1)}{T_2}(T_2) \oplus \frac{2^{(3-1)L_1}(2^{L_1}-1)}{T_3}(T_3),$$

or

$$1(1) \oplus 1(3) \oplus 2(6) \oplus 4(12).$$

The cycle set for an arbitrary polynomial is given by the following general result.

Theorem 4.6. *For a connection polynomial $C(D)$ factoring like*

$$C(D) = C_1(D)^{n_1} C_2(D)^{n_2} \cdots C_m(D)^{n_m},$$

$C_i(D)$ irreducible, has cycle set $S_1 \times S_2 \times \cdots S_m$, where S_i is the cycle set for $C_i^{n_i}$, and $(n_1)T_1 \times (n_2)(T_2) = (n_1 n_2 \cdot \gcd(T_1, T_2)(\mathrm{lcm}(T_1, T_2))$ and the distributive law holds for $\times$ and $\oplus$.

Example 4.10. *For a connection polynomial $C(D) = (1+D)^2(1+D+D^2+D^3+D^4)(1+D+D^4)$ the cycle set is*

$$[2(1) \oplus 1(2)] \times [1(1) \oplus 3(5)] \times [1(1) \oplus 1(15)].$$

Using the rule for $\times$ we get the cycle set

$$2(1) \oplus 1(2) \oplus 6(5) \oplus 3(10) \oplus 32(15) \oplus 16(30).$$

We can verify that $2 \cdot 1 + 1 \cdot 2 + 6 \cdot 5 + 3 \cdot 10 + 32 \cdot 15 + 16 \cdot 30 = 2^{10}$, i.e., the cycle set contains all possible states.

4.1.3 Decimation

If we have an m-sequence $s = s_0, s_1, \ldots$ we might want to consider the sequence $\mathbf{s}'$ obtained through *decimation* by k, defined as the sequence

$$\mathbf{s}' = s_0, s_k, s_{2k}, s_{3k}, \ldots.$$

As the sequence $\mathbf{s}$ corresponded to multiplication of β by the fixed element α, it is clear that $\mathbf{s}'$ corresponds to multiplication of β by the fixed element α^k, i.e, the cycle of different states correspond to the sequence

$$\beta, \alpha^k\beta, \alpha^{2k}\beta, \ldots, \alpha^{(T-1)k}\beta, \alpha^{Tk}\beta = \beta.$$

The length of this cycle which equals the period of $\mathbf{s}'$ is $\mathrm{ord}(\alpha^k)$. As we assumed that α was a primitive element, $\mathrm{ord}(\alpha^k) = q^L - 1/\gcd(q^L - 1, k)$.

Furthermore, let us review some further facts on finite fields. Assume that $\mathbb{F}_{q^L}$ was constructed through a degree L polynomial $\pi(x) \in \mathbb{F}_q[x]$ with $\pi(\alpha) = 0$. Let $\beta \in \mathbb{F}_q$ and consider the set of polynomials

$$\mathcal{F}(\beta) = \{f(x) \in \mathbb{F}_q[x] : f(\beta) = 0\}.$$

The set will contain at least one polynomial of degree $\leq L$. Let $f_0(x)$ be the polynomial in $\mathcal{F}(\beta)$ of lowest degree. Then any other polynomial $f(x)$ in $\mathcal{F}(\beta)$ can be written as $f(x) = q(x)f_0(x) + r(x)$, $\deg r(x) < \deg f_0(x)$ and

$$0 = f(\beta) = q(\beta)f_0(\beta) + r(\beta) = r(\beta).$$

So $r(\beta) = 0$ and this means that $f_0(x)|f(x)$ for all polynomials $f(x)$ in $\mathcal{F}(\beta)$.

The polynomial $f_0(x)$ is called the *minimal polynomial* of the element β.

In fact, the minimal polynomial to β, now denoted $\pi_\beta(x)$, can be calculated as

$$\pi_\beta(x) = (x - \beta)(x - \beta^q)(x - \beta^{q^2}) \cdots (x - \beta^{q^{d-1}}),$$

where d is the smallest integer such that $q^d \equiv 1 \bmod \mathrm{ord}(\beta)$ (d is the number of conjugates of β).

Just as the reciprocal of $\pi(x)$ gives the connection polynomial for a minimal LFSR producing a sequence corresponding to the state sequence

$$\beta, \alpha\beta, \alpha^2\beta, \ldots, \alpha^{(T-1)}\beta, \alpha^T\beta = \beta,$$

the the reciprocal of the minimal polynomial $\pi_\beta(x)$ gives the connection polynomial for a minimal LFSR producing a sequence corresponding to the state sequence

$$\beta, \alpha^k\beta, \alpha^{2k}\beta, \ldots, \alpha^{(T-1)k}\beta, \alpha^{Tk}\beta = \beta.$$

This means that the decimated sequence $\mathbf{s}'$ can be generated by an LFSR with a connection polynomial being the reciprocal of $\pi_{\alpha^k}(x)$.

Example 4.11. *Let* $\mathbf{s}$ *be the sequence generated by a LFSR with connection polynomial* $C(D) = 1 + D + D^4$ *in* $\mathbb{F}_2$*. The first four symbols are* $\mathbf{s} = 0, 0, 0, 1, \ldots$*. Assume that we decimate* $\mathbf{s}$ *by 3, obtaining* $\mathbf{s}'$*. Then* $\mathbf{s}'$ *will have period* $15/\gcd(15, 3) = 5$*. The reciprokal of* $C(D)$ *is* $\pi(x) = 1+x^3+x^4$*. Let* $\pi(\alpha) = 0$*. We calculate the minimal polynomial* $\pi_{\alpha^3}(x)$ *as*

$$\pi_{\alpha^3}(x) = (x + \alpha^3)(x + (\alpha^3)^2)(x + (\alpha^3)^4)(x + (\alpha^3)^8).$$

Note that the minimal polynomial will have coefficients in the ground field, in this case $\mathbb{F}_2$*. Calculating the above gives*

$$\pi_{\alpha^3}(x) = (x + \alpha^3)(x + \alpha^6)(x + \alpha^{12})(x + \alpha^{24}) = 1 + x + x^2 + x^3 + x^4.$$

The starting state for this LFSR is $0, 1, 1, 0$*.*

Then $\mathbf{s}'$ *is generated by an LFSR with connection polynomial* $C(D) = 1+D+D^2+D^3+D^4$*.*

4.1.4 Statistical properties of LFSR sequences

The importance of LFSR sequences in general and m-sequences in particular is due to their pseudo randomness properties. In many ways (but not all), m-sequences behave like sequences whose elements are chosen at random.

If $\mathbf{s} = s_0, s_1, \ldots$ is an m-sequence, recall that an r-gram is a subsequence of length r,

$$(s_t, s_t + 1, \ldots, s_{t+r-1}),$$

for $t = 0, 1, \ldots$.

Theorem 4.7. *Among the* $q^L - 1$ *L-grams that can be constructed for* $t = 0, 1, \ldots, q^L - 2$*, every nonzero vector appears exactly once.*

Proof. A repeated L-gram would in the sequence would mean a repeated state, violating the period of $q^L - 1$. Also, the all zero L-gram cannot occur because that would mean that the state is all zero. □

The above observation gives the interesting *run-distribution properties* of m-sequences. We view a periodic sequence $\mathbf{s}$ as a cycle. Also, we explain this for the binary case, so assume $q = 2$. A *run of length* r in a sequence $\mathbf{s}$ is a subsequence of *exactly* r zeros (or ones). This means that the r zeros must have a one before. It turns out that m-sequence have a very regular distribution of different run lengths.

Theorem 4.8. *The run distribution of any m-sequence of length* $2^L - 1$ *is given as*

length	0-*runs*	1-*runs*
1	2^{L-3}	2^{L-3}
2	2^{L-4}	2^{L-4}
$\vdots$	$\vdots$	$\vdots$
$L-2$	1	1
$L-1$	1	0
L	0	1
Total	2^{L-2}	2^{L-2}

We leave the proof out.

It is interesting to note that the run distribution of any m-sequence is almost exactly what would be expected from a purely random sequence.

The next statistical property we consider is the *autocorrelation function*. Let $\mathbf{x}, \mathbf{y}$ be two binary sequences of the same length n. The correlation $C(\mathbf{x}, \mathbf{y})$ between the two sequences is defined as the number of positions of agreements minus the number of disagreements.

The autocorrelation function $C(\tau)$ is defined to be the correlation between a sequence $\mathbf{x}$ and its τth cyclic shift, i.e.,

$$C(\tau) = \sum_{i=1}^{n} (-1)^{x_i + x_{i+\tau}}, \tag{4.6}$$

where subscripts are taken modulo n and addition in the exponent is mod 2 addition.

Theorem 4.9. *If* $\mathbf{s}$ *is an* m*-sequence of length* $2^L - 1$*, then*

$$C(\tau) = \begin{cases} 2^L - 1 & \text{if } \tau \equiv 0 \pmod{n} \\ -1 & \text{otherwise} \end{cases}$$

We skip the proof. Again we can see that the autocorrelation function for an m-sequence is very close to the autocorrelation function for a truly random sequence. In fact, if $X = (X_1, X_2, \ldots, X_n)$ is a random sequence, then X_i and X_j are independent if $i \neq j$. Then if $\tau \neq 0$

$$E[C(\tau)] = E[\sum_{i=1}^{n} (-1)^{x_i} (-1)^{x}_{i+\tau}] = \sum_{i=1}^{n} E[(-1)^{x_i}] E[(-1)^{x}_{i+\tau}] = 0.$$

So the expected value for a random sequence would be

$$C(\tau) = \begin{cases} n & \text{if } \tau \equiv 0 \pmod{n} \\ 0 & \text{otherwise} \end{cases}$$

We have seen a few properties of m-sequences that are very close to the behaviour of random sequences. We can continue and show more such things. For example, the decimation of an m-sequence or the sum of two different m-sequences are (under some assumptions) again m-sequences. So the run properties and the autocorrelation function for a decimated m-sequence is again as before.

However, recall that there is one property that is completely away from the properties of random sequences. Let the binary m-sequence be generated by the recursion $s_j = \sum_{i=1}^{L} c_i s_{j-i}$. By forming a set of random variables $X_j = \sum_{i=0}^{L} c_i s_{j-i}, j \leq L$ we see that $P(X_j = 0) = 1$. For a random sequence $x_1, x_2, \ldots$ and summing in the same way $X_j = \sum_{i=0}^{L} c_i x_{j-i}, j \leq L$ we would have $P(X_j = 0) = 1/2$. This is an extreme point of nonrandomness.

4.2 Exercises

Exercise 4.1. *Show that the polynomial $x^4 + x + 1$ is irreducible over the field GF(2).*

Exercise 4.2. *Find* $(0110)^{1/2}$. $\pi(\alpha) = 0, \pi(x) = x^4 + x + 1$.

Exercise 4.3. *Determine the elements in GF(3^2). $\pi(x) = x^2 + x + 2$ is irreducible over GF(3)*

Exercise 4.4. *Give the addition and multiplication tables for*

a) *GF(5)*

b) *GF(7)*

Exercise 4.5. *Give the multiplication table for GF(2^2), $\pi(x) = x^2 + x + 1$.*

Exercise 4.6. *Show that in all fields of characteristic p the following equality holds.* $(x + y)^p = x^p + y^p$

Exercise 4.7. *Find gcd($1 + D^7, 1 + D^2 + D^5 + D^6$) over GF(2).*

Exercise 4.8. The stop-and-go-generator is built using two linear feedback shift registers, L_1 and L_2. If L_2 outputs 1, then L_2 is clocked and the output bit of L_2 is the new keystream bit. If L_1 outputs 0 then L_2 is not clocked and the new keystream bit is the previous bit from L_2. As an example, if L_1 produces (101001...) and L_2 produces $(a_0, a_1, a_2, \ldots)$ then the keystream sequence is $(a_0, a_0, a_1, a_1, a_1, a_2, \ldots)$.

a) Describe a distinguishing attack on the generator.

b) One day, you manage to intercept a transmission that you know have been encrypted using the stop-and-go-generator. The intercepted bitstream is

$$C = (1001\ 1101\ 0110\ 1001\ 0001\ 0010\ 1110\ 0101).$$

You know that the intercepted bitstream is the encryption of one of the messages m_1, m_2 or m_3. Determine which message was encrypted.

m_1 : (1110 1000 1011 1111 1101 0001 0111 1100)

m_2 : (1101 1010 1100 0101 1001 1101 1011 1111)

m_3 : (1010 1100 1001 0110 1000 0101 0010 0101)

Exercise 4.9. *Determine the sequence corresponding to each of the following D-transforms*

a) $\frac{1}{D^{-1}+2}$ *in GF(3),*

b) $\frac{D^{-1}}{D^{-1}+2}$ *in GF(3),*

c) $\frac{1}{D^{-2}+D^{-1}+1}$ *in GF(2).*

Exercise 4.10. *Determine the period for each of the following polynomials*

a) $1 + D + D^2 + D^3 + D^4$ *in GF(2),*

b) $1 + D + D^2 + D^4$ *in GF(2),*

c) $1 + D^3 + D^6$ *in GF(2),*

d) $1 + 3D^2$ *in GF(5),*

e) $1 + 3D + 3D^2$ *in GF(5).*

Exercise 4.11. *Determine the cycle set for each of the following connection polynomials*

a) $1 + D^2 + D^4$ *in GF(2),*

b) $1 + D^4$ *in GF(2),*

c) $1 - D - 2D^2$ *in GF(3),*

d) $1 - 2D - D^2$ *in GF(3),*

e) $1 - D - D^2$ *in GF(3).*

Exercise 4.12. *Determine the cycle set for* $C(D) = 1 - 8D + 2D^2$ *in GF(13).*