Linear Feedback Shift Registers

Pseudo-Random Sequences

A **pseudo-random sequence** is a periodic sequence of numbers with a very long period.

Golomb's Principles

G1: The # of zeros and ones should be as equal as possible per period.
G2: Half the runs in a period have length 1, one-quarter have length 2, ... , $1/2^i$ have length i. Moreover, for any length, half the runs are blocks and the other half gaps. (A ***block*** is a subsequence of the form ... 011110... and a ***gap*** is one of the form ...10000001...., either type is called a ***run***.)
G3: The out-of-phase autocorrelation AC(k) is the same for all k.

AC(k) = (Agreements - Disagreements)/p where we are comparing a sequence of period p and its shift by k places. The autocorrelation is ***out-of-phase*** if p does not divide k.

Pseudo-Random Sequences

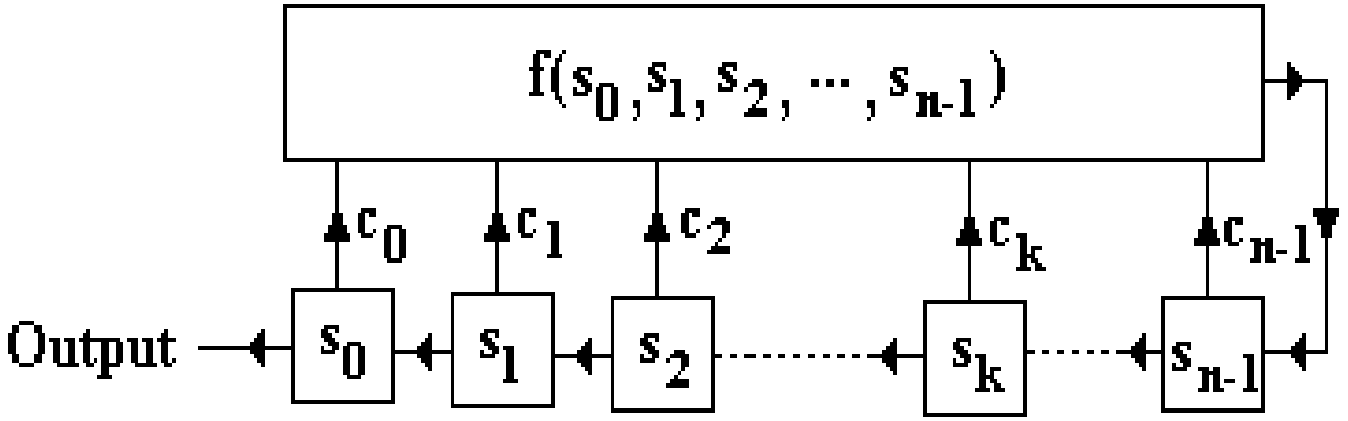
Furthermore, to be of practical use for cryptologists we would require:

C1: The period should be very long ($\sim 10^{50}$ at a minimum).

C2: The sequence should be easy to generate (for fast encryption).

C3: The cryptosystem based on the sequence should be cryptographically secure against chosen plaintext attack. (minimum level of security for modern cryptosystems)

Feedback Shift Registers (FSR's)



The c_i's and s_i's are all 0 or 1. All arithmetic is binary.

An FSR is ***linear*** if the function f is a linear function, i.e.,

$$f(\vec{s}) = \sum_{i=0}^{n-1} c_i s_i$$

LFSR's

The output is determined by the initial values $s_0, s_1, ..., s_{n-1}$ and the linear recursion:

$$s_{k+n} = \sum_{i=0}^{n-1} c_i s_{k+i}, \quad k \geqslant 0$$

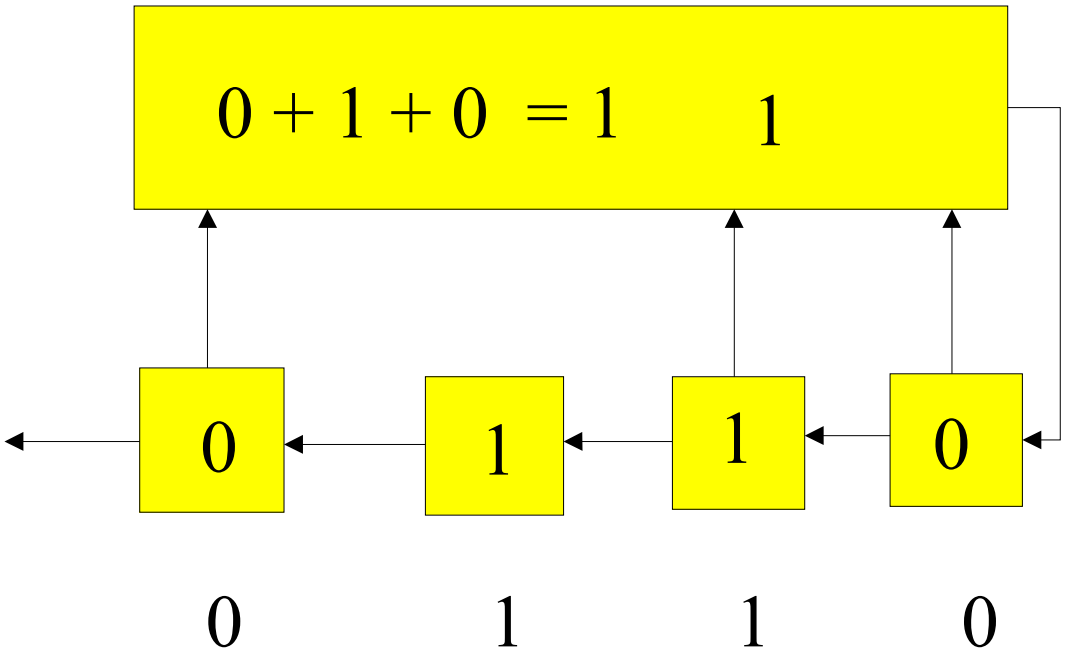
which is equivalent to:

$$\sum_{i=0}^{n} c_i s_{k+i} = 0, \quad k \geqslant 0$$

if we **define** $c_n := 1$.

LFSR's

Example: n = 4 $c_0 = c_2 = c_3 = 1, c_1 = 0$ initial state 0,1,1,0



PN-sequences

A sequence produced by a length n LFSR which has period 2^n-1 is called a ***PN-sequence*** (or a pseudo-noise sequence).

We can characterize the LFSR's that produce PN-sequences. We define the ***characteristic polynomial*** of an LFSR as the polynomial,

$$f(x) = c_0 + c_1 x + c_2 x^2 + \cdots + c_{n-1} x^{n-1} + x^n = \sum_{i=0}^{n} c_i x^i$$

where $c_n = 1$ by definition and $c_0 = 1$ by assumption.

Some Algebra Factoids

Every polynomial f(x) with coefficients in *GF(2)* having f(0) = 1 divides $x^m + 1$ for some m. The smallest m for which this is true is called the ***period*** of f(x).

An ***irreducible*** (can not be factored) polynomial of degree n has a period which divides $2^n - 1$.

An irreducible polynomial of degree n with period $2^n - 1$ is called a ***primitive polynomial***.

Theorem: *A LFSR produces a PN-sequence if and only if its characteristic polynomial is a primitive polynomial.*

Examples

1: The characteristic polynomial of our previous example of an LFSR with n = 4 is: $f(x) = x^4 + x^3 + x^2 + 1 = (x+1)(x^3 + x + 1)$ and so is not irreducible and therefore **not primitive**.

2: $f(x) = x^4 + x^3 + x^2 + x + 1$ is an irreducible polynomial(no linear factors and remainder $x + 1$ when divided by $x^2 + x + 1$). However, $x^5 + 1 = (x+1)f(x)$ and so it has period 5 and is **not primitive**.

3: $f(x) = x^4 + x^3 + 1$ is an irreducible polynomial over GF(2). To find its period, we have to determine the smallest m so that f(x) divides $x^m + 1$. Clearly, $m > 4$ and the period divides $2^4 - 1 = 15$, thus it must be either 5 or 15. By trying the possibilities we get

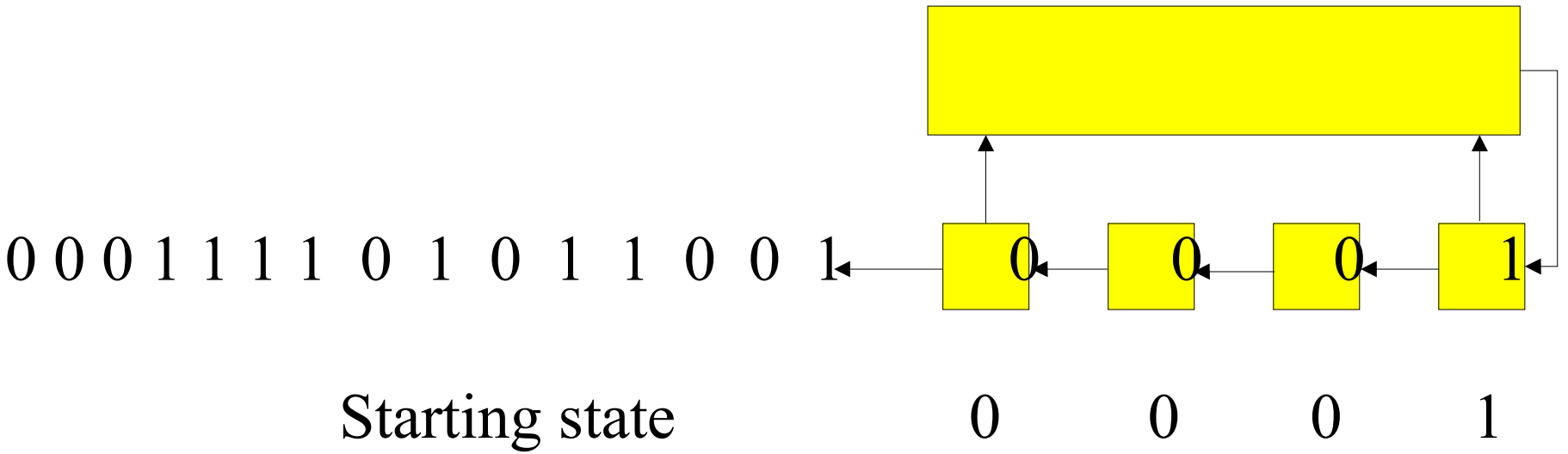
$$x^5 + 1 = (x+1)(x^4 + x^3 + 1) + (x^3 + x)$$

$$x^{15} + 1 = (x^{11} + x^{10} + x^9 + x^8 + x^6 + x^4 + x^3 + 1)(x^4 + x^3 + 1)$$

Thus, f(x) has period 15 and so, is a **primitive polynomial**.

Example

The LFSR with characteristic polynomial $f(x) = x^4 + x^3 + 1$ is



$\Omega(f)$

Let $\Omega(f)$ denote the set of all sequences that can be produced from an LFSR with characteristic polynomial f(x).

Since each starting state produces a different (we are considering shifts as different) sequence, *there are* 2^n *elements in* $\Omega(f)$ since there are that many starting states. The sum of two sequences in $\Omega(f)$ is again in $\Omega(f)$ since the sum will satisfy the same recursion relationship (i.e., the sum corresponds to a different starting state).

The ***reciprocal polynomial*** of f(x) of degree n, denoted f*(x) is:

$$f^*(x) = x^n f(1/x) = c_0 x^n + c_1 x^{n-1} + \ldots + c_n.$$

Note that if $f(x) = g(x)h(x)$ then $f^*(x) = g^*(x)\, h^*(x)$.

Form of Ω(f)

Theorem: Ω(f) = {τ(x)/f*(x), where deg τ(x)< n }.

Pf: We show that each element of Ω(f) can be uniquely expressed in the desired form, and the result will follow since there are exactly 2^n binary polynomials of degree < n.

Let S(x) ∈ Ω(f), where $S(x)=\sum s_i x^i$ and $f^*(x) = c_0 x^n + \ldots + c_n$. Then,

$$S(x) f^*(x)=\left(\sum_{k=0}^{\infty} s_k x^k\right)\left(\sum_{l=0}^{n} c_{n-l} x^l\right)$$

$$=\sum_{j=0}^{\infty}\left(\sum_{l=0}^{min(j,n)} c_{n-l} s_{j-l}\right) x^j$$

$$=\sum_{j=0}^{n-1}\sum_{l=0}^{j} c_{n-l} s_{j-l} x^j+\sum_{j\geq n}\sum_{l=0}^{n} c_{n-l} s_{j-l} x^j$$

$$=\tau(x)+\sum_{j\geq n}\left(\sum_{i=0}^{n} c_i s_{(j-n)+i}\right) x^j \quad =\tau(x)$$

Are PN-sequences psuedo-random?

G1: Since every non-zero state appears once per period and the leftmost bit of the state is the next output value of the sequence, there are 2^{n-1} 1's and 2^{n-1} - 1 0's in any period.

G2: For $k \leq n - 2$, a run of length k will occur in the sequence whenever the leftmost k +2 states are of the form 0111...110 or 100...001. Since all states occur, the number of each of these state sequences is 2^{n-k-2}. There is one state 011..11, and it is followed by state 11..11 since f is primitive f(1) = 1, and that state is followed by 11..110, thus there is no block of size n-1 and one block of size n. Similarly, there is no gap of size n and only one of size n-1. We can therefore calculate the number of runs as

$$2+2\sum_{k=1}^{n-2} 2^{n-k-2}=2+2\left(2^{n-2}-1\right)=2^{n-1}$$

and of these $1/2^k$ of them are of length k.

Are PN-sequences psuedo-random?

G3: Let $\{s_i\}$ be a PN-sequence and $\{s_{i+k}\}$ be the same sequence shifted k places. The sum of these two sequences satisfies the same recursion relation as the both of them do and so is a PN-sequence as well. The number of agreements in the two sequences will be the number of 0's in the sum and the number of disagreements is the number of 1's in the sum. So by G1,

$$AC(k) = \frac{(2^{n-1}-1)-(2^{n-1})}{2^n-1} = \frac{-1}{2^n-1}$$

for all $1 \le k < 2^n - 1$.

Thus we see that PN-sequences satisfy all of Golomb's conditions for pseudo-randomness.

Crypto Properties of LFSR's

C1: One can obtain sufficiently large periods by taking n large enough. In fact, n = 166 will give a period of $2^{166} - 1 > 10^{50}$.

C2:Being simple Boolean circuits, LFSR's are extremely easy to implement and are very fast.

C3: Zilch. Given 2n consecutive bits of the sequence, s_k, s_{k+1}, ..., s_{k+2n-1} (**which can be obtained in a known plaintext attack**) we can write down a system of n equations in the n unknowns c_0, ..., c_{n-1} which is non-degenerate and so has a unique solution. This gives the characteristic polynomial and so the LFSR to the cryptanalyst.

Linear Equivalence

Thus, linear feedback shift registers should **not** be used in cryptographic work (despite this, LFSR's are still the most commonly used technique). However, this argument does not apply to non-linear FSR's so we need to examine them next.

An FSR with a possibly non-linear feedback function will still produce a periodic sequence (with a possible non-periodic beginning). If the period is p, then the LFSR with characteristic function $1 + x^p$ and starting state equal to the period of the sequence, will produce the same sequence; possibly other LFSR's will also. Hence, the following definition makes sense.

The ***linear equivalence*** of a periodic sequence S(x) is the length n of the smallest LFSR that can generate S(x).

More Theory

Lemma 1 : Let h(x) and f(x) be the characteristic polynomials of an m-stage and respectively n-stage LFSR. Then $\Omega(h)$ is contained in $\Omega(f)$ iff $h(x) \mid f(x)$.

Lemma 2: Let S(x) be in $\Omega(f)$ with $S(x) = \tau(x)/f^*(x)$. Then there exists an h(x) with $h(x) \mid f(x)$ and $h(x) \neq f(x)$ with $S(x)$ in $\Omega(h)$ iff $\gcd(\tau(x), f^*(x)) \neq 1$.

Theorem: Let S(x) be the generating function of a periodic binary sequence with period p. Let $S^{(p)}(x)$ be the truncated polynomial of degree p-1. Then there exists a unique polynomial m(x) with

a) $S(x) \in \Omega(m)$, and
b) if $S(x) \in \Omega(h)$ then $m(x) \mid h(x)$.

Minimal Characteristic Polynomials

m(x) is called the ***minimal characteristic polynomial*** of S(x), and

$$m^*(x) = \frac{1 + x^p}{\gcd\left(S^{(p)}(x), 1 + x^p\right)}$$

Proof. Let S(x) ∈ Ω(m), but S(x) ∉ Ω(f) for any proper divisor f of m. We shall prove that m is unique. Note that $S(x) = S^{(p)}(x)/ (1 + x^p)$. Since S(x) ∈ Ω(m), we know that there exists a τ(x) with degree < degree of m, such that $S(x) = \tau(x)/m^*(x)$. By Lemma 2, gcd $(m^*(x),\tau(x)) = 1$, so

$$\gcd\left(m^*(x), \frac{S^{(p)}(x) m^*(x)}{1 + x^p}\right) = 1$$

$$\gcd\left(m^*(x)(1 + x^p), S^{(p)}(x) m^*(x)\right) = 1 + x^p$$

$$m^*(x) \gcd\left(1 + x^p, S^{(p)}(x)\right) = 1 + x^p$$

Example

Cor: The linear equivalence of S(x) with period p is the degree of m(x) (= deg m*(x)) above.

Consider the following non-linear feedback function (3-stage):

$$f(s_0,s_1,s_2) = s_0 + s_1 + s_1s_2 + 1.$$

With starting state 1 0 1 we get the following sequence:

1 0 1	-
0 1 0	1
1 0 0	0
0 0 0	1
0 0 1	0
0 1 1	0
1 1 1	0
1 1 0	1
1 0 1	1

The period 8 sequence produced has the generating polynomial:

$$S^{(8)}(x) = 1 + x^2 + x^6 + x^7.$$

Now gcd $(S^{(8)}(x), 1 + x^8) = 1 + x$, so $m^*(x) = (1 + x)^7 = 1 + x + x^2 + x^3 + x^4 + x^5 + x^6 + x^7$, and $m(x) = m^*(x)$. With starting state 1 0 1 0 0 0 1, this 7-stage LFSR produces the same sequence. The linear equivalence of our starting sequence is thus 7.

Non-Linear Functions ☹

We see that the use of non-linear functions does not gain any cryptographic security since we can always find a LFSR to give the same sequence. In an attempt to get this security, various means of combining the outputs of LFSR's in a non-linear way have been attempted. Clearly, sums, shifts and products of outputs don't work. Most of the information on these techniques is classified (so, *someone* does believe that the required security can be obtained this way).

One such approach which is in the public domain is the multiplexing algorithm of Jennings.

Jennings' Multiplexing Algorithm

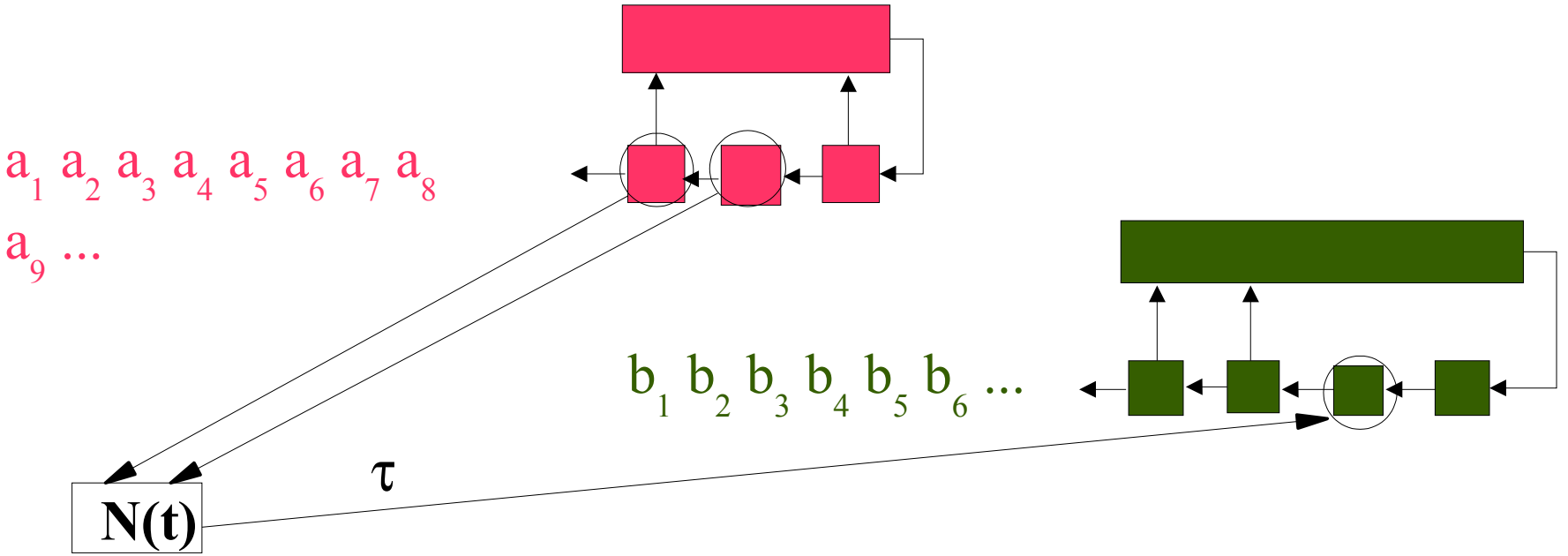
Take an m-stage (the a_i) and an n-stage (the b_j) LFSR with primitive characteristic polynomials and non-zero starting states. Choose $h \leq \min(m, \log_2 n)$ entries from the set of subscripts $\{0, 1, ..., m-1\}$ and order them $0 \leq i_1 < i_2 < ... < i_h < m$. At time t, define

$$N(t) = \sum_{j=1}^{h} a_{i_j}(t) 2^{j-1}$$

Let τ be any one-one mapping from $\{0,1,...,2^h - 1\}$ into $\{0,1, ..., n-1\}$. Define the output u(t) of the multiplexed sequence to be:

$$u(t) = b_{\tau(N(t))}(t)$$

Jennings' Multiplexing Algorithm



Thm: *If (m,n) = 1 this multiplexed sequence has period* $(2^m - 1)(2^n - 1)$.

Thm: *If (m,n) = 1 and h = m - 1, then the sequence has linear equivalence* $n(2^m - 1)$.