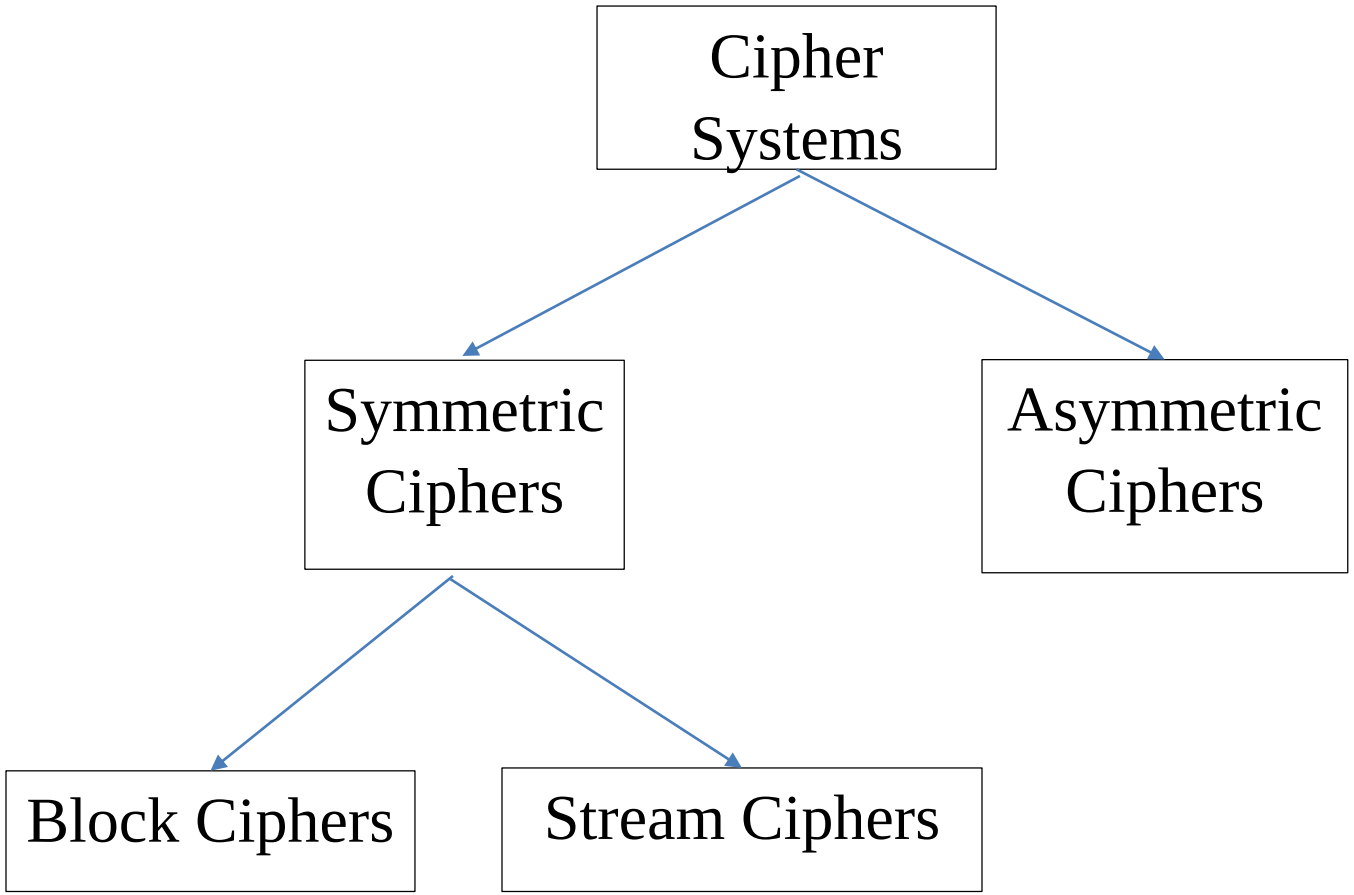
Stream Ciphers

AK Bhateja

Cipher Systems



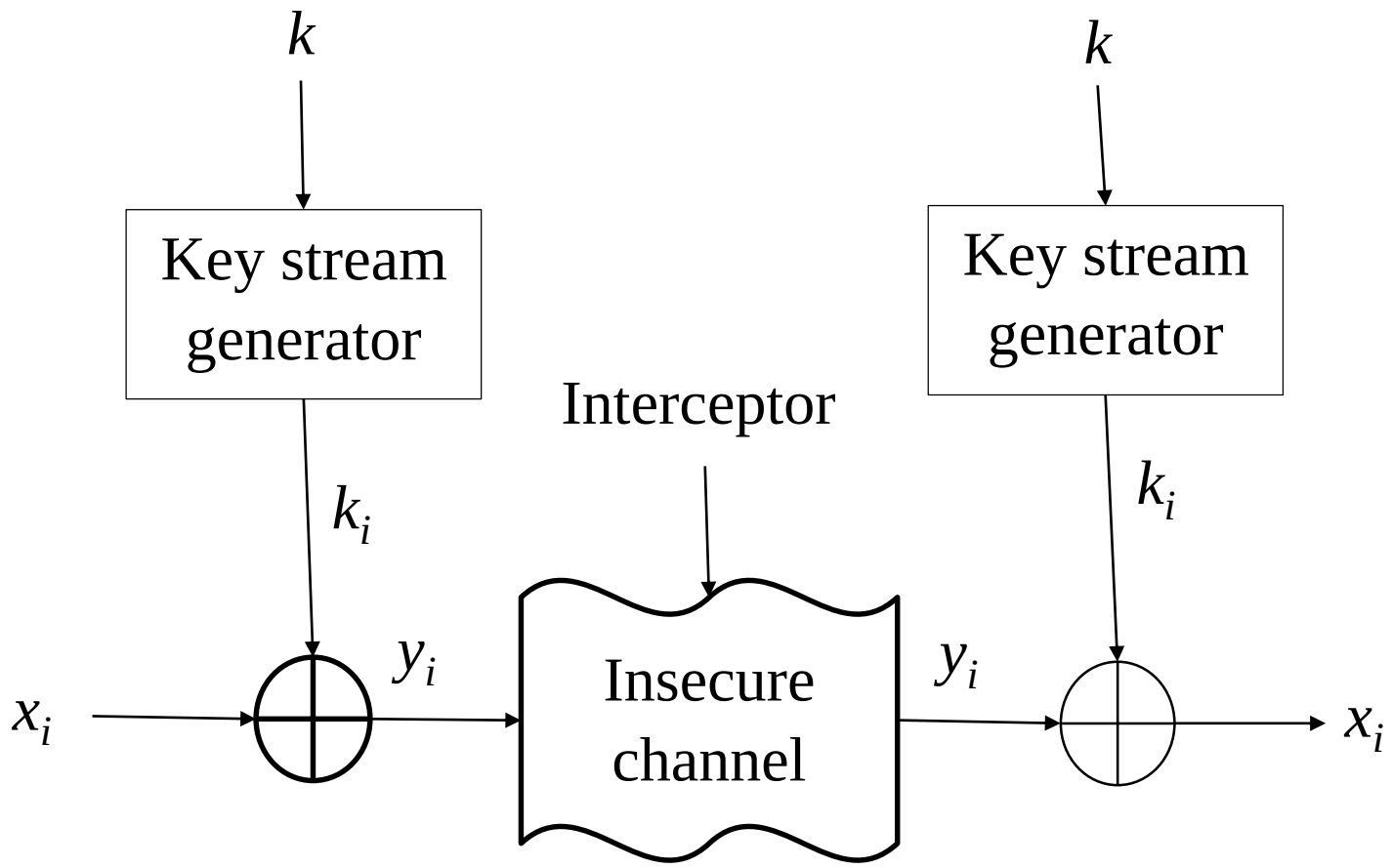
Stream Ciphers

- Encrypt bits individually.
- This is achieved by adding a bit from a key stream to a plaintext bit.
- The key stream depends on the key
- Employ shift registers
- Examples:
  - One time pad – unconditional secure
  - LFSRs - easy to construct with a little electrical engineering knowledge
  - RC4 suited for software & has been used in the security system for wireless local area networks (WLANs)
  - A5/1 is suited to hardware

Practical Stream Ciphers

- One Time Pads are unconditionally secure, but that they have drawbacks which make them impractical.
- Replace the truly random key stream bits by a pseudorandom number generator where the key *k* serves as a seed.
- Practical stream ciphers are not unconditionally secure.
- An elegant way of realizing long pseudorandom sequences is to use LFSRs.

Stream Cipher



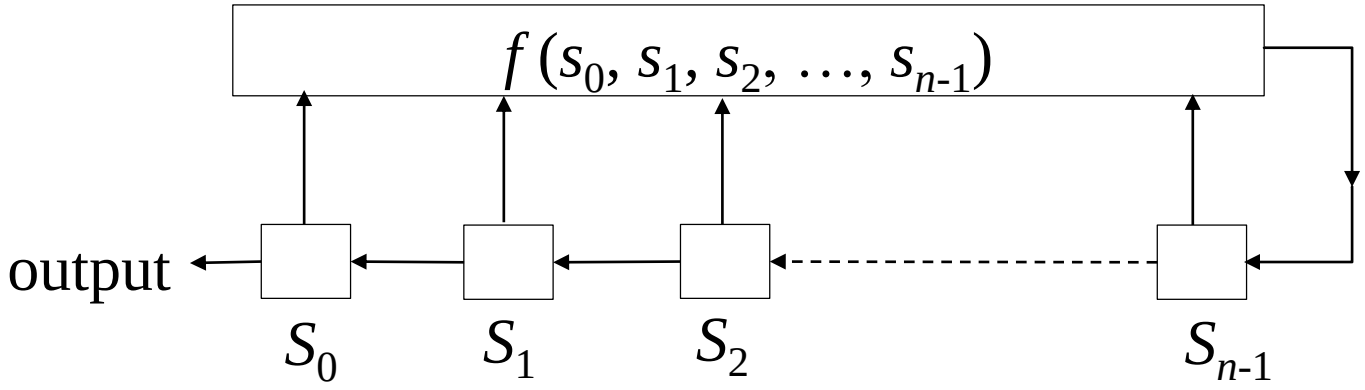
- The two main requirements for a secure stream cipher based cryptosystem
  - The sequence which is to be added in the plaintext must have a guaranteed minimum length for its period.
  - The cipher must appear to be random

- Assumptions (worst case conditions) in order to access the security of a system:
  - The cryptanalyst has a complete knowledge of the cipher system
  - The cryptanalyst has obtained a considerable amount of ciphertext
  - The cryptanalyst knows the plaintext equivalent of a certain amount of the ciphertext.

Feedback Shift Registers

- Consists of clocked storage elements (flip-flops) and a feedback function.
- The number of storage elements is the degree of the shift register.
- The feedback network computes the input for the last flip-flop
- The binary storage elements are called the stages of the shift register & at any given time their contents are called its state.
- A shift register with n stages has 2^n possible states.
- $f(s_0, s_1, s_2, \ldots, s_{n-1})$ is feedback function

Linear Feedback Shift Registers



- Feedback function is linear

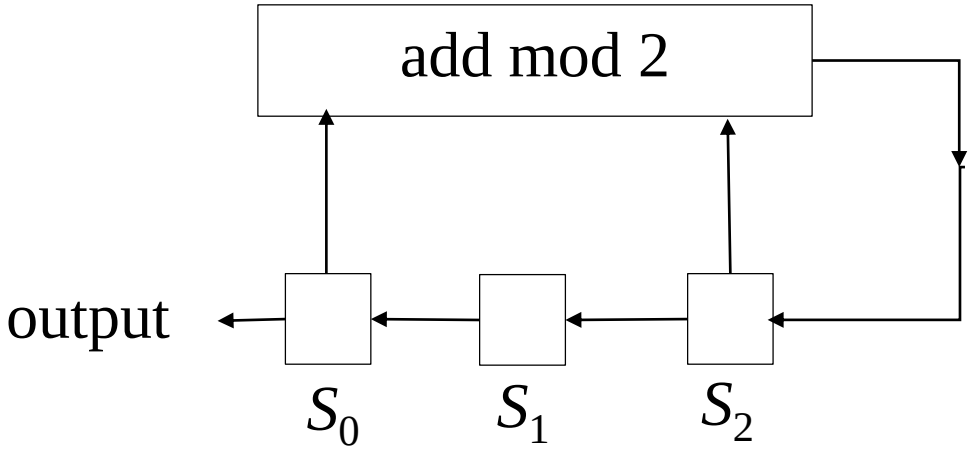
  $f(s_0, s_1, s_2, \ldots, s_{n-1}) = c_1 s_{n-1} + c_2 s_{n-2} + \ldots + c_n s_0$

  i.e. $s_n = c_1 s_{n-1} + c_2 s_{n-2} + \ldots + c_n s_0$
- where each c_i is 0 or 1 and all addition is over $GF(2)$.
- The constants $c_1, c_2, \ldots, c_n$ are called the feedback coefficients,

  i.e. coefficients of polynomial $c_0 + c_1 x + c_2 x^2 \ldots + c_n x^n$, here $c_0 = 1$.
- More generally, the contents of the shift register are $(s_t, s_{t+1}, \ldots, s_{t+n-1})$, the bit s_{t+n} is the output

$$s_{t+n} = c_1 s_{t+n-1} + c_2 s_{t+n-2} + \ldots + c_{n-1} s_{t+1} + c_n s_t = \sum_{i=1}^{n} c_i s_{t+n-i}$$

A three stage LFSR

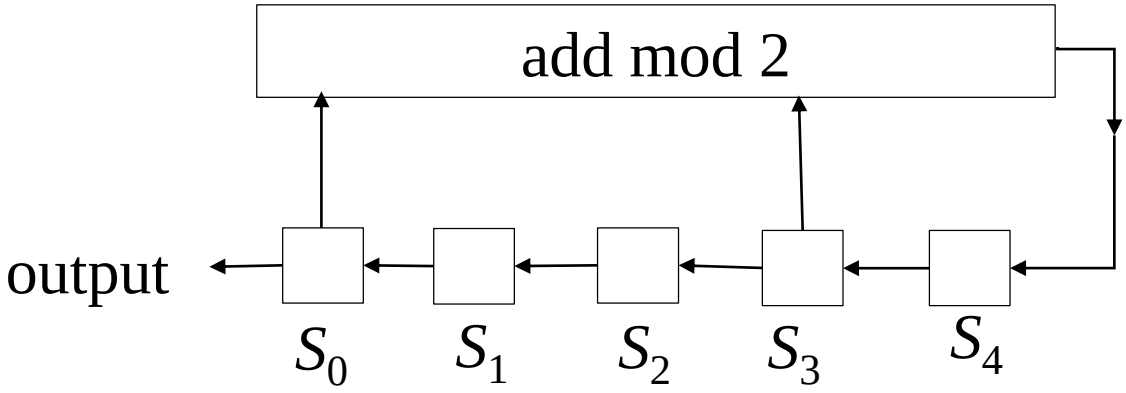


Successive states:

1 1 1	1 0 0
1 1 0	0 0 1
1 0 1	0 1 1
0 1 0	1 1 1

Sequence generated: 1 1 1 0 1 0 0

Five stage shift register with feedback function $s_0 + s_3$



- State at $t = 0$ is 0 1 0 1 0

  $t = 1$ 1 0 1 0 1 etc.

  it will have 31 different states.
- If all the feedback coefficients are 0, then the sequence of states will begin with 01010, 10100, 01000, 10000, 00000, then every subsequent state will be null state.

Theorem: The succession of states in a shift register is periodic, with a period $p \leq 2^n - 1$, where n is the number of stages.

Proof. Each state of the shift register is completely determined by the previous state. Hence, if it ever happens that a state is the same as some earlier state, then the following states are the same.

With n stages in a shift register, there are only 2^n possible states.

Finally, if the state "all 0's" ever occurs, the subsequent states will also consist of "all 0's", and the periodicity is $p = 1$. Thus a long period cannot include this state and $p \leq 2^n - 1$.

Note: This theorem holds no matter what the initial state of the shift register is.

The output sequence of n stage LFSR, the maximum possible length i.e. $2^n - 1$ is called an m-sequence.

Theorem: Let p be a prime and let $q = p^n$ be a power of p with $n \geq 1$. Then every irreducible polynomial of degree n in $F_p[x]$ is a factor of $x^{q-1} - 1$.

Proof: Let $f(x)$ be an irreducible polynomial of degree n in $F_p[x]$.

It has a root α in the extension field $K = F_p(\alpha)$ of F_p.

Therefore K has order $q = p^n$ and the multiplicative group K^* has order $q - 1$.

Therefore the order of any element $\alpha \in K^*$ divides $q - 1$.

So in particular $\alpha^{q-1} = 1$.

This means that α is a root of the polynomial $x^{q-1} - 1 = 0$.

Hence $f(x) \mid x^{q-1} - 1$.

Exponent of a polynomial: Smallest p such that $f(x)$ divides $x^p - 1$ is called the exponent of $f(x)$.

$f(x)$ has exponent e if $f(x) \mid x^e - 1$ but $f(x) \nmid x^r - 1$ for any r satisfying $0 < r < e$.

Primitive polynomial: An irreducible polynomial of degree n over Z_p is called primitive if its exponent is $p^n - 1$.

i.e. polynomial $f(x)$ of degree n is primitive if $f(x) \mid x^{p^n-1} - 1$ but $f(x) \nmid x^r - 1$ for any r satisfying $0 < r < p^n - 1$.

Fact: If p is prime and $r_1, r_2, \ldots, r_t$ be the district prime factors of $p^n - 1$. Then an irreducible polynomial $f(x)$ of degree n is primitive if for each $0 < i < t$:

$$x^{\frac{p^n-1}{r_i}} \not\equiv 1 \pmod{f(x)}$$

i.e. x is an element of order $p^n - 1$ in the field $Z_p[x] / f(x)$.

Definition: A primitive polynomial $f(x)$ is an irreducible polynomial of degree n in $F_{p^n}[x]$ with the property that each root of f is a generator of $F_{p^n}^*$, the multiplicative group of F_{p^n}.

For any n-stage register with feedback constants $c_0, c_1, c_2, \ldots, c_n$ the characteristic polynomial $f(x)$ is $f(x) = 1 + c_1 x + \cdots + c_{n-1} x^{n-1} + x^n$

If (s_t) is the output sequence generated from an initial state of $s_0, s_1, s_2, \ldots, s_{n-1}$ then the recurrence relation is

$$s_{t+n} = c_1 s_{t+n-1} + c_2 s_{t+n-2} + \ldots + c_n s_t$$

In LFSR, an irreducible polynomial $f(x)$ of degree n is primitive if for each $0 < i < t$

$$x^{\frac{2^n-1}{r_i}} \not\equiv 1 \pmod{f(x)}$$

Where $r_1, r_2, \ldots, r_t$ be the district prime factors of $2^n - 1$.

Fact: A LFSR produces a maximum period sequence (PN-sequence) if its connection (characteristic) polynomial is a primitive polynomial.

Theorem: For any positive integer n the number of primitive polynomials of degree n over F_{p^n} is $\lambda(n) = \frac{\varphi(p^n-1)}{n}$.

Proof: The multiplicative group of F_{p^n} is cyclic with order p^n-1.

We know that the number of generators of multiplicative group of F_{p^n} is $\varphi(p^n-1)$.

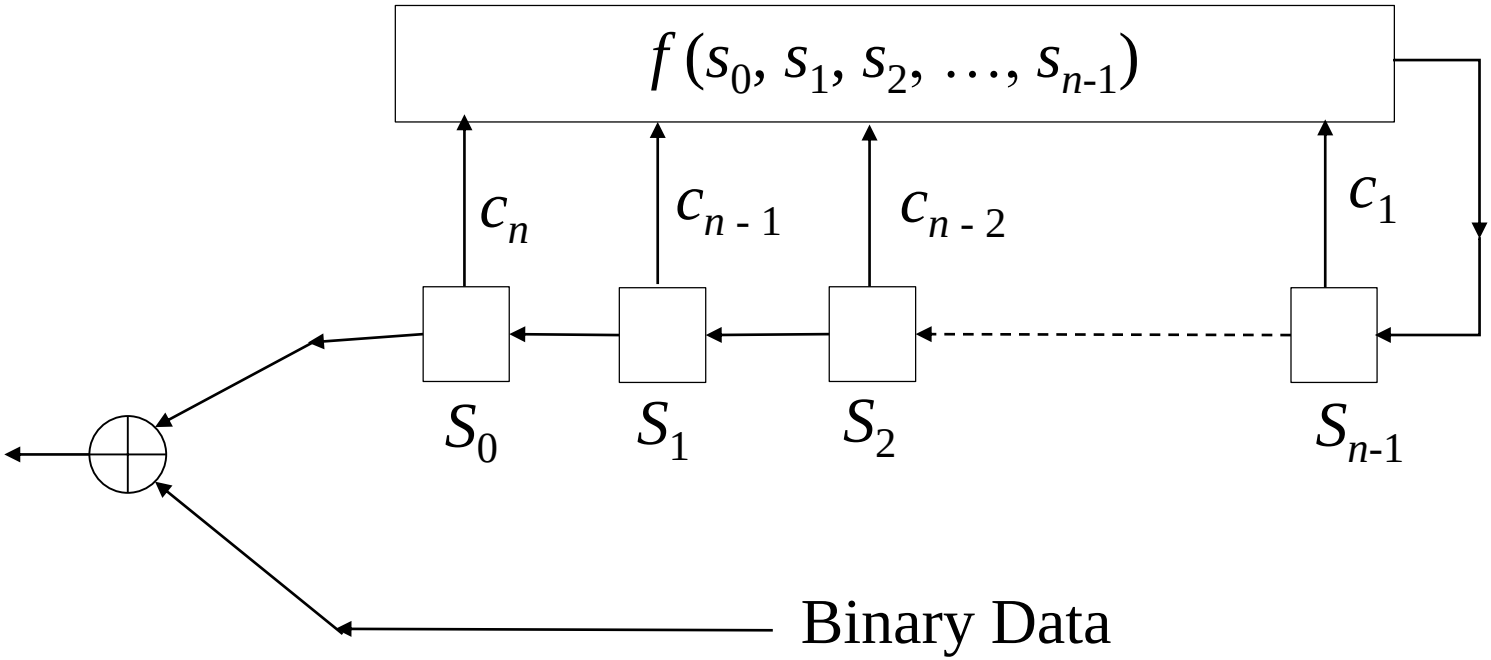
Since they generate all of $F^*_{p^n}$, they can't be elements of a proper subfield so their minimal polynomials all have degree n.

Also, if one of the roots of a polynomial is primitive, all of them have same multiplicative order.

(Given $\alpha \in F_{p^n}$, an arbitrary conjugate is a α^{p^i} for some integer i. If $\alpha^k = 1$ then $\left(\alpha^{p^i}\right)^k = \alpha^{kp^i} = \left(\alpha^k\right)^{p^i} = 1$. If $\alpha^k \neq 1$ then likewise $\left(\alpha^{p^i}\right)^k = \left(\alpha^k\right)^{p^i}$. Therefore conjugates have the same multiplicative order.)

So the $\varphi(p^n-1)$ generators are accounted for by their $\lambda(n) = \frac{\varphi(p^n-1)}{n}$ minimal polynomials, each of which is primitive.

LFSR based Stream Cipher



The matrix approach of LFSR based stream cipher

Initial state: $s_0, s_1, s_2, \ldots, s_{n-1}$

Feedback constants: $c_1, c_2, \ldots, c_n$

For any t let s_t be the state vector at time t,

$$\mathbf{s}_t = (s_t \quad s_{t+1} \quad \cdots \quad s_{t+n-1}), \qquad \mathbf{s}_{t+1} = (s_{t+1} \quad s_{t+2} \quad \cdots \quad s_{t+n})$$

$s_{t+n} = c_1 s_{t+n-1} + c_2 s_{t+n-2} + \ldots + c_{n-1} s_{t+1} + c_n s_t$

Matrix form $\mathbf{s}_{t+1} = \mathbf{s}\,\boldsymbol{C}$ where $\boldsymbol{C}$ is $n \times n$ matrix

$$\boldsymbol{C} = \begin{bmatrix} c_1 & 1 & 0 & \cdots & 0 \\ c_2 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ c_{n-1} & 0 & 0 & 0 & 1 \\ c_n & 0 & 0 & \cdots & 0 \end{bmatrix}$$

Here $|\boldsymbol{C}| = c_n$, Thus $\boldsymbol{C}$ has inverse iff $c_n \neq 0$, i.e. iff $c_n = 1$. $\mathbf{s}_t = \mathbf{s}_{t+1}\,\boldsymbol{C}^{-1}$

Which implies that each state vector uniquely determines its predecessor as well as its successor.

The matrix approach of LFSR based stream cipher

The characteristic equation of the matrix $\boldsymbol{C}$ is

$$\det(\boldsymbol{C}-\lambda\boldsymbol{I}) = \begin{vmatrix} c_1-\lambda & 1 & 0 & \cdots & 0 \\ c_2 & -\lambda & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ c_{n-1} & 0 & 0 & \cdots & 1 \\ c_n & 0 & 0 & \cdots & -\lambda \end{vmatrix}$$

$$= (-\lambda)^{n-1}(c_1-\lambda) - c_2(-\lambda)^{n-2} + c_3(-\lambda)^{n-3} \ldots (-1)^r c_n$$

$$= -(-\lambda)^n \left[1 - \frac{c_1}{\lambda} - \frac{c_2}{\lambda^2} - \frac{c_3}{\lambda^3} - \cdots - \frac{c_n}{\lambda^n}\right]$$

$$= \frac{(-1)^{n+1}}{x^n}\left[1 - (c_1 x + c_2 x^2 + c_3 x^3 + \cdots + c_n x^n)\right]$$

where the substitution $x = 1/\lambda$ has been made. Thus, except for the factor of $(-1)^{n+1}/x^n$, the characteristic equation of $\boldsymbol{C}$ is the characteristic polynomial of the shift register.