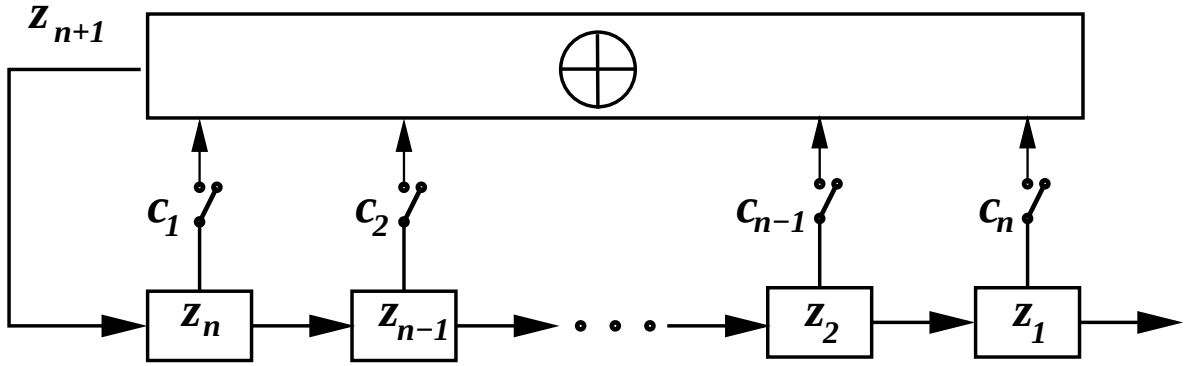
Linear Feedback Shift Registers – LFSR

Stanislav Palúch

Fakula riadenia a informatiky, Žilinská univerzita

23. októbra 2017

Lineárny posuvný register



The sequenc $c_1, c_2, \ldots, c_n$ – is called a **tap sequence**

$$z_{n+1} = c_1 z_n \oplus c_2 z_{n-1} \oplus \cdots \oplus c_{n-1} z_2 \oplus c_n z_1 \qquad (1)$$

Maximum perions of LSFR of the length n is $2^n - 1$.

A connection polynomial – is a polynomial over $\mathbb{Z}_2$:

$$1 + c_1 x + c_2 x^2 + c_3 x^3 + \cdots + c_n x^n$$

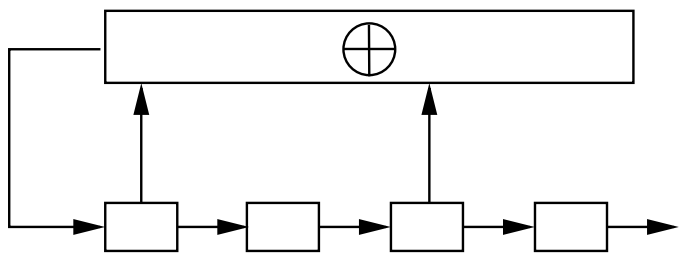
A primitiv polynomial of deree n is such a polynomial of degree n which

- is irredicible
- is a divisor of polynomial $x^{2^{n-1}} + 1$
- does not divide any polynomial of the form $x^d + 1$, where d is a divisor of $2^n - 1$

Connection Polynomial

It holds:
LSFR of the length n has maximum period $2^n - 1$ if and only if its connection polynomial is primitive.

A singular LFSR is such a LFSR whose length is greater then the degree of its connection polynomial.

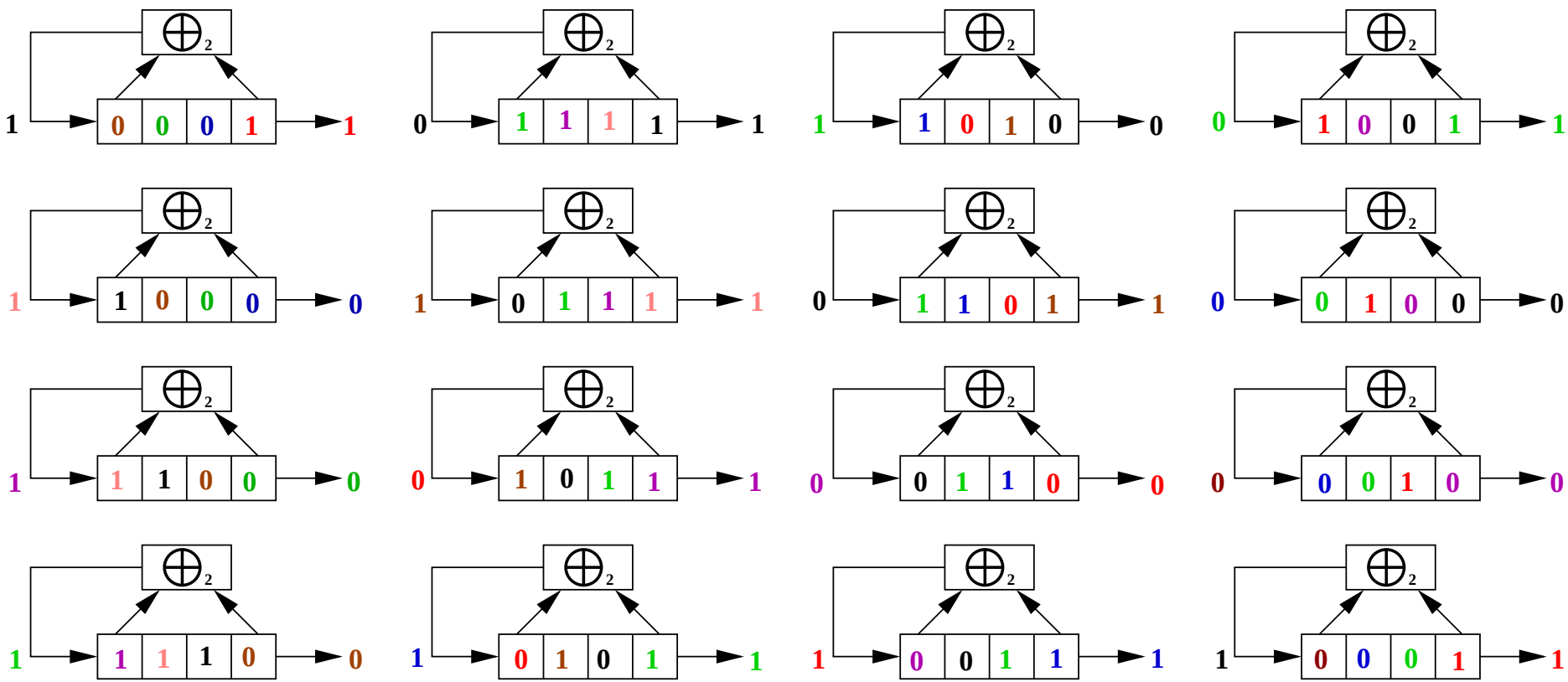


Singular LFSR-s are not used in cryptography.

It is an algoritmically solvable problem to discover whether a given polynomial is a primitive polynomial.

However, finding of primitive polynomials is a hard problem.

An Example How a LSFR Works



LFSR in a Spreadsheet

	A	B	C	D	E
1	0	0	0	0	0
2	=MOD(A1+D1+ E1;2)	=A1	=B1	=C1	=D1

Second row of this table will be copied into several following rows.

The idea:
To use outpit bits of LFSR as a stream ofpseudo-random binary numbers.

Key

- Original setup or LFSR – n bits $z_1, z_2, \ldots, z_n$
- Setup of tap sequence – n bites $c_1, c_2, \ldots, c_n$

If we know tap sequence $c_1, c_2, \ldots, c_n$ and a sequence of n bit $z_1, z_2, \ldots, z_n$ from LFSR, then we can easily compute all following bits using equation (**??**).

$$z_{n+1} = c_1 z_n \oplus c_2 z_{n-1} \oplus \cdots \oplus c_{n-1} z_2 \oplus c_n z_1 \tag{1}$$

Attack Against LFSR If 2n Bits Are Known

If we know only the length of LFSR we proceed as follows:
Suppose that we know the lnegth n of LFSR and $2n$ output bits:

$$z_{2n}, z_{2n-1}, \ldots, z_2, z_1$$

Bits $z_{n+1}, z_{n+2}, \ldots, z_{2n}$ were generate step by step, every one from them using the equation (**??**) and $n-1$ predecessors as follows:

$$\begin{aligned}
z_{n+1} &= c_1 z_n \quad \oplus \ldots \oplus c_{n-1} z_2 \quad \oplus c_n z_1 \\
z_{n+2} &= c_1 z_{n+1} \quad \oplus \ldots \oplus c_{n-1} z_3 \quad \oplus c_n z_2 \\
&\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \\
z_{2n} &= c_1 z_{2n-1} \oplus \ldots \oplus c_{n-1} x_n z n - 1 \oplus c_n z_n
\end{aligned}$$

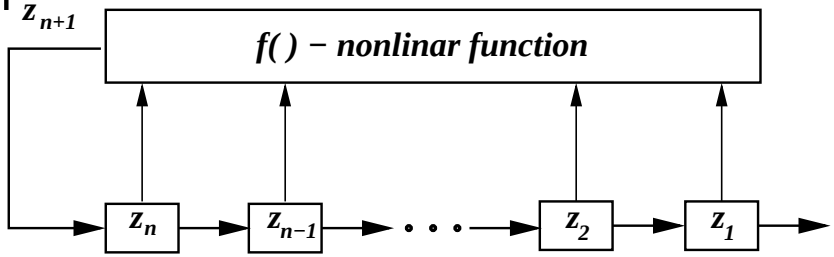
$$\begin{pmatrix} z_n & z_{n-1} & \ldots & z_1 \\ z_{n-1} & z_{n-2} & \ldots & z_2 \\ \ldots & \ldots & \ldots & \ldots \\ z_{2n-1} & z_{2n-2} & \ldots & z_n \end{pmatrix} \begin{pmatrix} c_1 \\ c_2 \\ .. \\ c_n \end{pmatrix} = \begin{pmatrix} z_1 \\ z_2 \\ .. \\ z_n \end{pmatrix}$$

$$\mathbf{Zc} = \mathbf{z} \qquad \mathbf{c} = \mathbf{Z}^{-1}\mathbf{z}$$

Colorary: Cryptography using LFSR is very weak and must not be used.

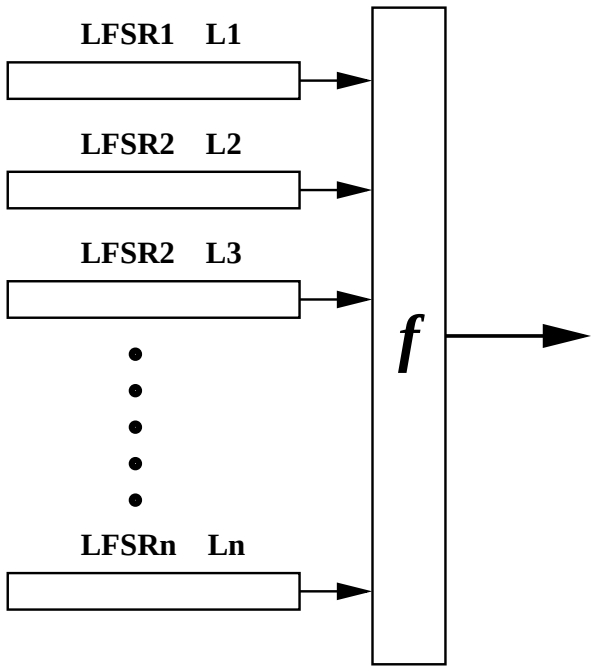
Attempths to Improve the Safety of LFSR

Replacement of $\oplus$ by a non linear function:

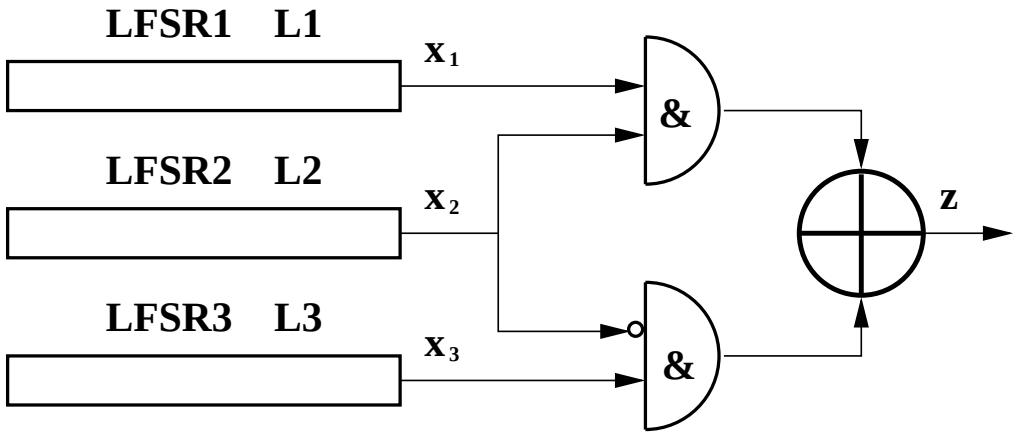


Drawback: Such nelinear registers ara hard to study. It is difficult to prove their properties e.g. existence of short cycles, e.t.c.

To use otuputs from several LSFRs as inputs into non linear function.



Geffe Generator



$$z = x_1.x_2 \oplus (1 \oplus x_2).x_3$$

$$P[z = x_1] = \underbrace{P[x_2 = 1]}_{=\frac{1}{2}} + \underbrace{P[x_2 = 0]}_{=\frac{1}{2}} . \underbrace{P[x_3 = x_1]}_{=\frac{1}{2}} = \frac{1}{2} + \frac{1}{2}.\frac{1}{2} = \frac{3}{4}$$

$$P[z = x_3] = \underbrace{P[x_2 = 0]}_{=\frac{1}{2}} + \underbrace{P[x_2 = 1]}_{=\frac{1}{2}} . \underbrace{P[x_3 = x_1]}_{=\frac{1}{2}} = \frac{1}{2} + \frac{1}{2}.\frac{1}{2} = \frac{3}{4}$$

Geffe Generator

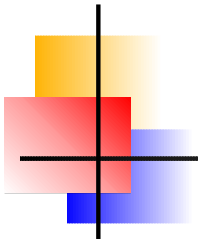
Other way how to determine probabilities $P[x_i = z]$.

Table of Otupt Function $z = x_1.x_2 \oplus (1 \oplus x_2).x_3$

	x_1	x_2	x_3	$z = x_1.x_2 \oplus (1 \oplus x_2).x_3$	$x_1 = z$	$x_3 = z$
	0	0	0	0	+	+
	0	0	1	1	-	+
	0	1	0	0	+	+
	0	1	1	0	+	-
	1	0	0	0	-	+
	1	0	1	1	+	+
	1	1	0	1	+	-
	1	1	1	1	+	+

It follows from this table that searched probabilities are

$$P[x_1 = z] = \frac{6}{8} = \frac{3}{4}, \qquad P[x_3 = z] = \frac{6}{8} = \frac{3}{4}.$$



Correlation Attack

Key of Geffe Generator – Start up contents of registers LFSR1, LFSR2 and LFSR3 – i.e. $(2^{L_1}-1)(2^{L_2}-1)(2^{L_3}-1)$ possibilities.
Therefore a brute force attack would require at most the same number of attempts.

Correlation attack:

We have a sequence $\mathbf{z} = z_1, z_2, \ldots, z_k, \ldots$ from output of Geffe generator.

Step 1.:

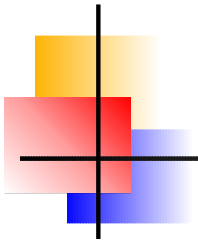
Fill LFSR2 and LFSR3 with arbitrary sequences of bits and then set up LFSR1 from $00\ldots01$ up to $11\ldots11$ untill the number of equalities with sequence $\mathbf{z} = z_1, z_2, \ldots, z_k, \ldots$ rises to approximately $\frac{3}{4}$.
In this moment LFSR1 will be set up exactly alike as it was in the beginning of sequence $\mathbf{z}$.

Step 2.:

The initial state of LFSR3 set by the same way.

Step 3.:

Calculate initial state of LFSR2 using initial states of LSFR1 and LSFR3 and the equation $z = x_1.x_2 \oplus (1 \oplus x_2).x_3$.



Correlation Attack

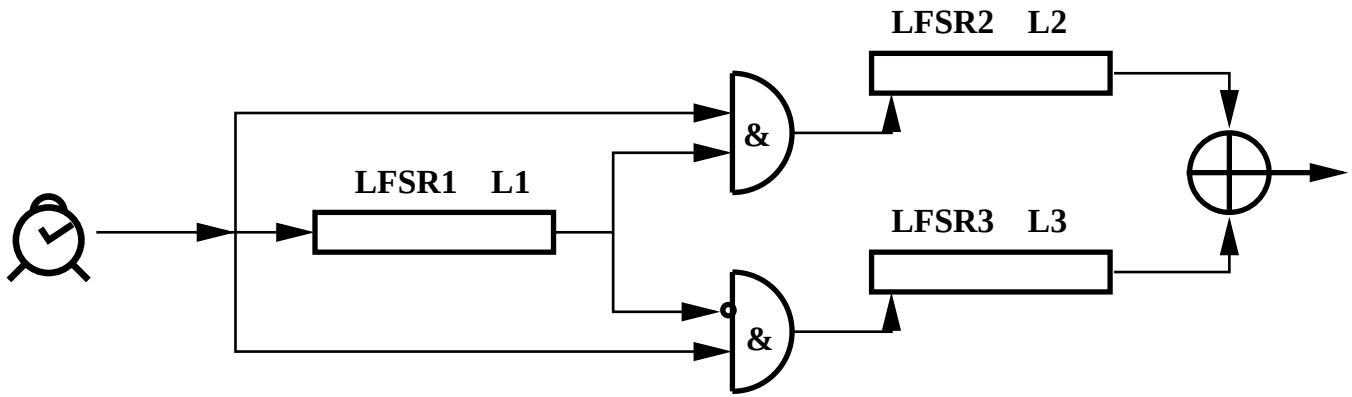
Correlation attack requires at most $(2^{L_1} - 1) + (2^{L_3} - 1)$ attemptes, instead of trying at most $(2^{L_1} - 1)(2^{L_2} - 1)(2^{L_3} - 1)$ possibilities of initial set up of all three registers.

If registers LFSR1 and LFSR3 are long enough this attack becomes impracticable, however, existence of an attack with complexity $(2^{L_1} - 1) + (2^{L_3} - 1)$ compared to complexity of brute attack $(2^{L_1} - 1)(2^{L_2} - 1)(2^{L_3} - 1)$ is considered as a serious weakness of this system.

This principle can be used for arbitrary cryptosystem based on LFSRs with arbitrary output function, if for output x_i of i-th register and output of cryptosystem it holds $P[x_i = z] \neq \frac{1}{2}$.

Alternating Stop-and-go Generator – ASG

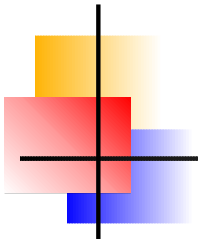
Another pseudorandom bits generator based on LFSFR-s is the alternating stop-and-go generator.
Its design was published in 1987 by C. G. Günther.



All LFSRs used should be regular registers all with maximum period.
Output of LFSR1 defines which of registers LFSR2, LFSR3 will shift in this clock impus.
If the oputput of LFSR1 is 1, LFSR2 is clocked. Otherwise LFSR3 is clocked.
If LFSR1 is modified in such a way that after $(L_1 - 1)$ zeros it produces one more zero, then the periond of this generator will be equal to

$$2^{L_1}.(2^{L_2} - 1).(2^{L_3} - 1)$$

provided that L_1, L_2, L_3 are two by two coprime.



Alternating Stop-and-go Generator – ASG

Khazaei, S., Fischer, S., and Meier, W. published in 2007 an attack against ATG with time complexity $O(L^2.2^{2L/3})$ and the amount of output needed to mount the attack $O(2^{2L/3})$ bits, where L is the size of the shortest of the three LFSRs.

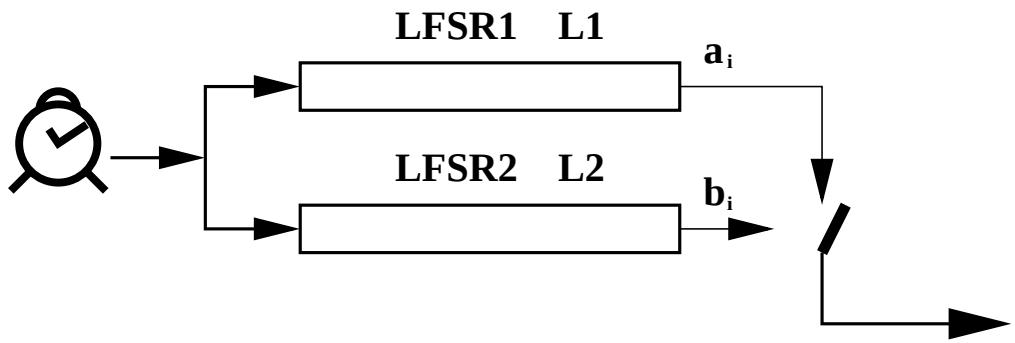
For L_1, L_2, L_3 coprime, $L_1 \approx L_2 \approx L_3 \approx 256$ is the corresponding time complexity of ASG equal to

$$O(256^2.2^{512/3}) \approx O(2^{16}.2^{170}) = O(2^{186}).$$

ASG was patented, patent expired in Jun 15, 1993to failure to pay maintenance fee.

Shrinking Generator

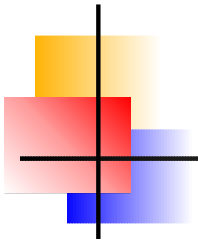
The shrinking generator was published in Crypto 1993 by
D. Coppersmith, H. Krawczyk, and Y. Mansouris.
The shrinking generator uses two LFSRs.
LFSR1 of the size L_1 generates output bits,
LFSR2 of the size L_2 controls output of LFSR1.
Both LFSR1 and LFSR2 accept clock impulses at the same time and produce bits a_i, b_i.



If $b_i = 1$ then the output bit of shrinking generator is a_i.
If $b_i = 0$ then the the bit a_i is discarded, nothing is output.
The period of this generator is

$$(2^{L_1} - 1).(2^{L_2} - 1)$$

provided that L_1 and L_2 are coprime



Shrinking Generator

The great the disadvantage if shrinking enerator is that the generator's output rate varies irregularly.

Those irregularities can be used directly to determine the state of LFSR2.

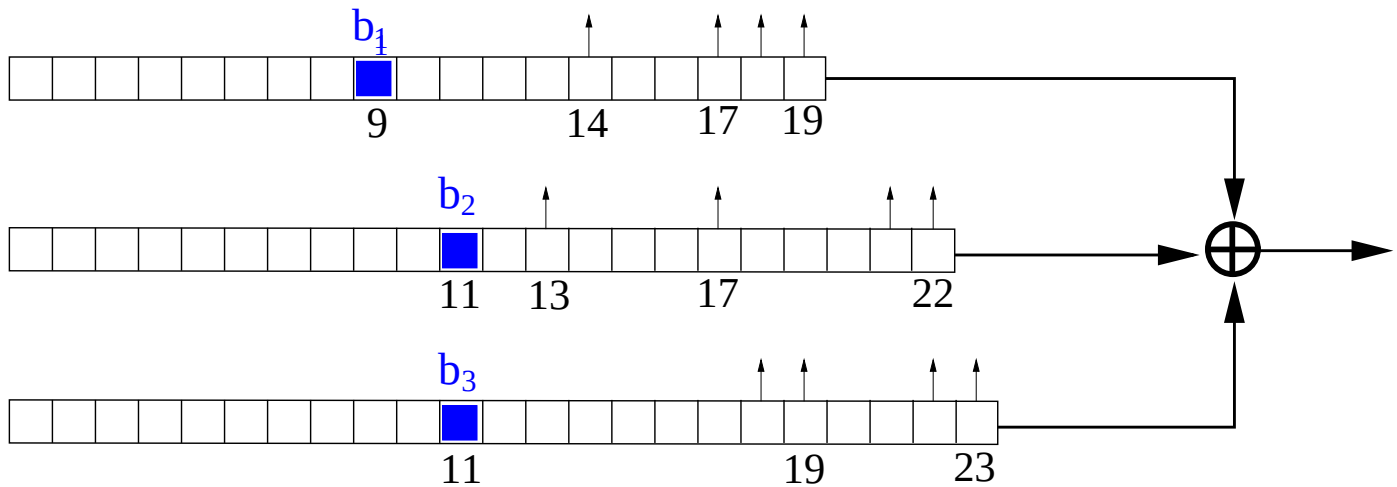
This problem can be overcome by buffering the output.

There are currently no known attacks better than brute attack when the feedback polynomials are secret.

If the feedback polynomials are known, the best known attack requires less than $L_1.L_2$ bits of output.

GSM A5 algoritmus

LFSR1 – (19, 18, 17, 14, 0)
LFSR2 – (22, 21, 17, 13, 0)
LFSR3 – (23, 22, 19, 18, 0)



$$posun(i) = b_i \oplus \overline{T(b_{i+1}, b_{i+2}, b_{i+3})}$$

$$\overline{T(b_1, b_2, b_3)} = \begin{cases} 0 & \text{ak } (b_1 + b_2 + b_3) \geq 2 \\ 1 & \text{ak } (b_1 + b_2 + b_3) \leq 1 \end{cases}$$

$$shift(i) = b_i \oplus \overline{T(b_1, b_2, b_3)}$$

Blum - Micalli generator, RSA generator

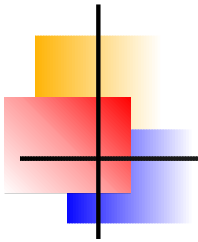
Blum - Micalli generator:
g, p two great secret primes

$$\begin{aligned} x_{i+1} &= g^{x_i} \mod p \\ b_i &= \begin{cases} 1 & \text{ak } x_i < \frac{p-1}{2} \\ 0 & \text{inak} \end{cases} \end{aligned}$$

RSA generator:

p, q dve great secret primes
$N = p.q$
e coprime with s $(p-1)(q-1)$

$$\begin{aligned} x_{i+1} &= x_i^e \mod N \\ b_i &= x_i \mod 2 \text{ (– the least sinificant bit } x_i) \end{aligned}$$



Statistical Hypothesis Testing

Suppose we are given a sequence of bits

$$\mathbf{b} = b_1, b_2, \ldots, b_n$$

gained from a genuin random or pseudorandom number generator.
Our goal is to find if this sequence is usable in one time pad cryptography.

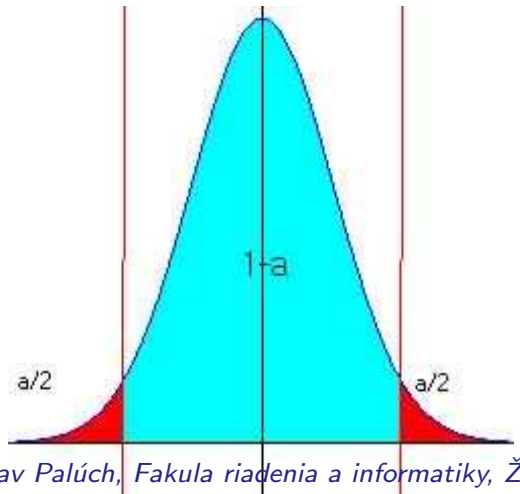
Folowint test can exclude sequences which are not suitable for enciphering.

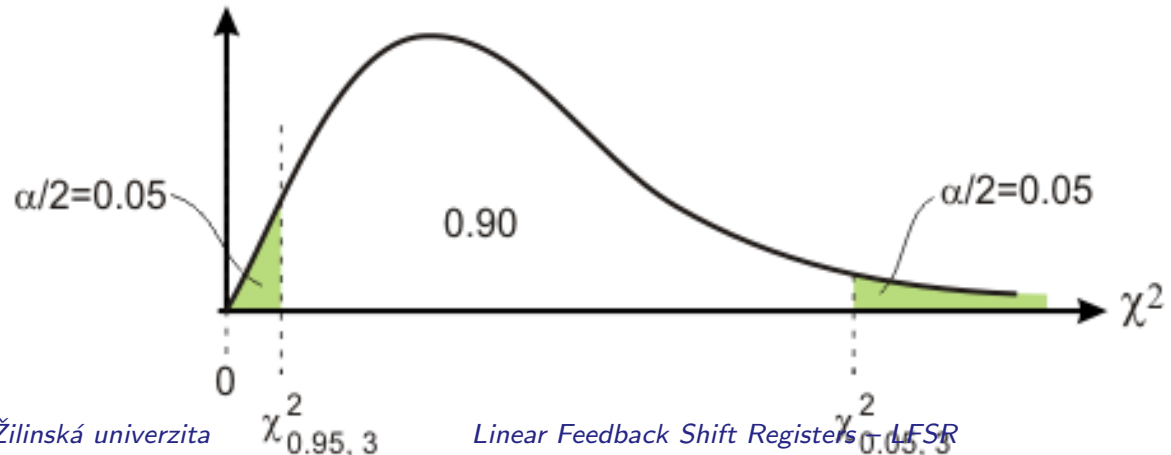
Principle of all text is as follows:

- A hypotheses H is specified (e.g. "$P[b_i = 1] = P[b_i = 0] = \frac{1}{2}$" – i. e. the probablility of 0 equals to the probablity of 1).
- Select a significance level α, a probability threshold below which the hypothesis H will be rejected in spite of the fact that it holds. Common values are 5% and 1%.
  (this is so calld an error of the first type).

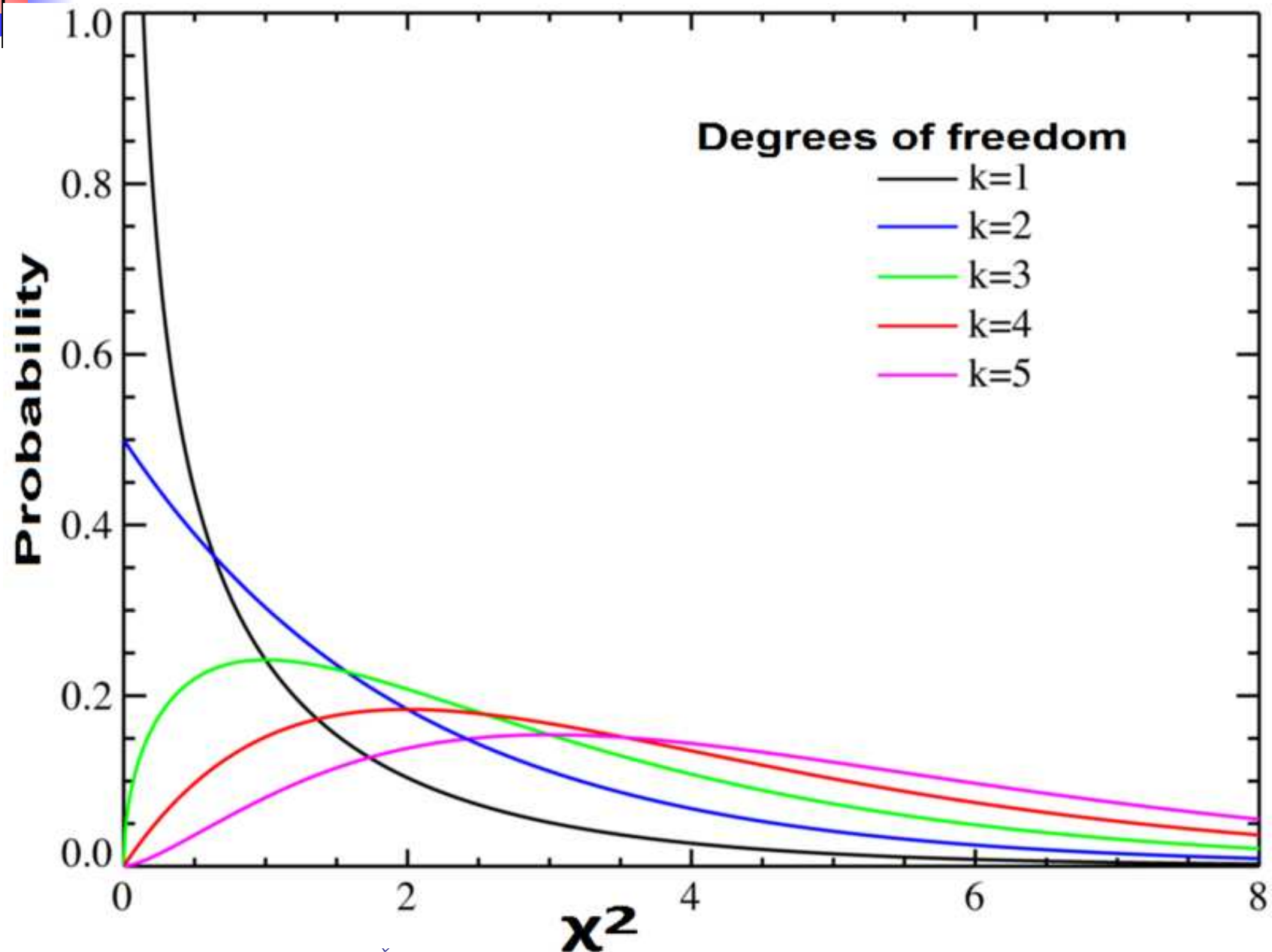
Statistical Hypothesis Testing

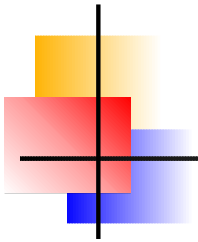
- A random function $X = f(b_1, b_2, \ldots, b_n)$ (called also a statisics) is determined, which has a known probablility distribution f under the assumptoin of validity of hypothes H
  (wery often we use Student distribution $f = \chi^2(k)$ with k degrees of freedom or normal distribution $f = N(0, 1)$
- An interval (a, b) – so called confidence interval is determined such that $P[X \in (a, b)] = 1 - \alpha$.
  A part of real axis $(-\infty, a\rangle \cup \langle b, \infty)$ is called a critical region.
- If X falls into the critical region, then we reject hypothesis H since a not avaited event occured, provided H is valid.
- If X falls into interval (a, b), then we do not reject hypotesis H.





Density of Distribution χ^2 for Various Degrees of Freedom





Frequency Test

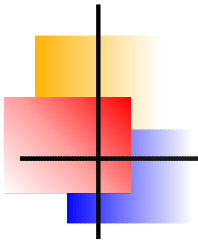
Let us have a sequence of bits $\mathbf{b} = b_1, b_2, \ldots, b_n$.

$$n_0 \text{ – number of zeros} \quad n_1 \text{ – number of ones} \quad n = n_o + n_1$$

Assume that **b** is a random sequence with the same probability of zeros and ones. The statistics

$$X_1 = \frac{(n_0 - n_1)^2}{n}$$

has distribution $\chi^2(1)$ with one degree of freedom and tested hypothesis is that $X_1 = 0$.



Two-Bit Serial Test

Let us have a sequence of bits $\mathbf{b} = b_1, b_2, \ldots, b_n$.

Let $n_{00}, n_{01}, n_{10}, n_{11}$ – numbers of appearence tuples 00, 01, 10, 11 in the sequence $\mathbf{b}$.

It holds: $n_{00} + n_{01} + n_{10} + n_{11} = n - 1$.

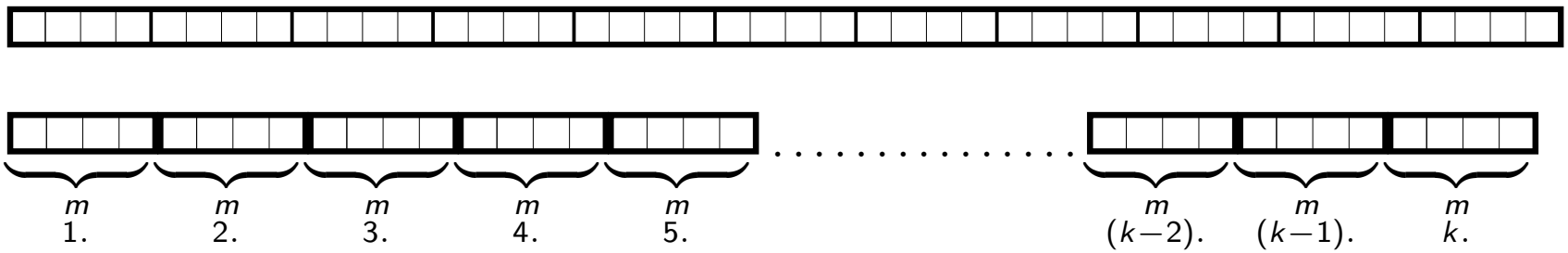
$$X_2 = \frac{4}{n-1}\left(n_{00}^2 + n_{01}^2 + n_{10}^2 + n_{11}^2\right) - \frac{2}{n}\left(n_0^2 + n_1^2\right) + 1$$

Statistics X_2 has probablility distribution $\chi^2(2)$ with two degrees of freedom for $n \geq 21$ under assumption that probablilities of all mentioned couples are equal.

Tested hypothesis is $X_2 = 0$.

Poker Test

Let us have a sequence of bits $\mathbf{b} = b_1, b_2, \ldots, b_n$.



We divide the examined sequence **b**of the length n into k m-tuples. Clearly $k.m \leq n$.

Number m must be selekted inorder $k \geq 5.2^m$.

Every m-tuple of bits represents a number from 0 to $2^m - 1$.
Let us assign by n_i the number m-tuples such they represent the number i in binary notation for $i = 0, 1, 2, \ldots, 2^m - 1$.

$$X_3 = \frac{2^m}{k} . \left(\sum_{i=0}^{2^m - 1} n_i^2 \right) - k$$

Štatistics X_3 has distribution $\chi^2(2^m - 1)$ and tested hypothesis is $X_3 = 0$.

Runs Test

Block of the length n is a sequence of n ones in the sequence **b** fenced from both sides by zero or beginning or end of the sequence **b**.
Gap of the length n is a sequence of n zeros in the sequence **b** fenced from both sides by one or beginning or end of the sequence **b**.
Probablity of occurence of a block of the length i $\ldots 0 \underbrace{1\,1\,\ldots\,1}_{i} 0 \ldots$ in an endless random sequence with the same probablity of zeros and ones is : $\frac{1}{2^{i+2}}$. The same number hold for probablility o occurence of a gap of the length i.

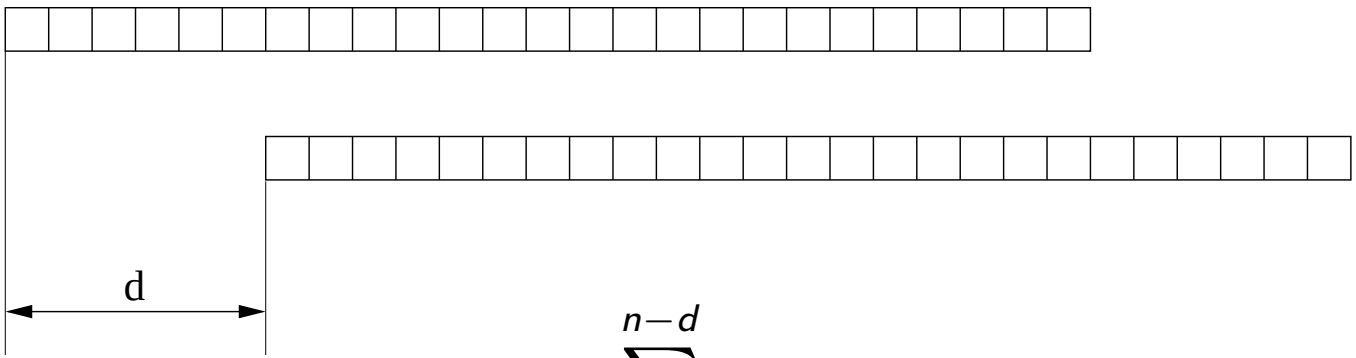
Awaited number of blocks (resp. gaps) of the length i in a n-element sequence **b** is $e_i = \frac{n-i+3}{2^{i+2}}$.
Define statistics

$$X_4 = \sum_{i=1}^{k} \frac{(B_i - e_i)^2}{e_i} + \sum_{i=1}^{k} \frac{(G_i - e_i)^2}{e_i}$$

where k is the greatest number such that $e_i \geq 5$ and where B_i, G_i are real numbers of blocks resp. gaps in the sequence **b**.
Statistics X_4 has distribution $\chi^2(2k-2)$, tested hypothesis is $X_4 = 0$.

Autokorelation Test

Let us have a sequence of bits $\mathbf{b} = b_1, b_2, \ldots, b_n$.
This test can reveal wheather the sequence **b** contains a periodical component with period d.
Let d – be a fixed number, let $1 \leq d \leq [\frac{n}{2}]$



$$A(d) = \sum_{i=1}^{n-d} b_i \oplus b_{i+d}$$

$$X_5 = 2.\frac{A(d) - \frac{n-d}{2}}{\sqrt{n-d}}$$

Statistics X_5 has a normal distribution $N(0, 1)$.

Tested hypothesis is $X_5 = 0$.

FIPS 140–1 Statistic Test

Test is determined for string **b** containing 20000 bits.

1. Monobit test: $1 < n_1 < 10346$
2. Poker test pre $m = 4$: $1.03 < X_3 < 57.4$
3. Runs test.
   For $i = 1, 2, 3, 4, 5$ B_i resp. G_i – the number of bloks resp. gaps of the length i.
   For $i = 6$ B_6 resp. G_6 the number of bloks resp. gaps of the length 6 and more.

i	Dovolený rozsah B_i, G_i
1	2267 – 2733
2	1079 – 1421
3	502 – 748
4	223 – 402
5	90 – 223
6	90 – 223

4. Long run test. There must not exist a block or gap of the length 34 or more.