A comparative Study to calculate the Runs Property in the encryption systems

Dr.Ayad Ghazi Naser Al-Shimmariand* **AmerAbdulmajeedabdulrahman****
dr_ayad64@yahoo.com amer6567@yahoo.com
*Ministry of Education - General Directorate of Vocational Education
**College of Education for Women - Computer Dept.

Abstract

Cryptographic applications demand much more of a pseudo-random-sequence generator than do most other applications. Cryptographic randomness does not mean just statistical randomness, although that is part of it. For a sequence to be cryptographically secure pseudo-random, it must be unpredictable.

The random sequences should satisfy the basic randomness postulates; one of them is the run postulate (sequences of the same bit). These sequences should have about the same number of ones and zeros, about half the runs should be of length one, one quarter of length two, one eighth of length three, and so on.The distribution of run lengths for zeros and ones should be the same. These properties can be measured deterministicallyand then compared to statistical expectations using a chi-square test.

In this paper the Run Criterion, is calculated, it can be calculated for any key generator before it be implemented or constructed (software or hardware).The cryptosystems: Linear, Product and Brüer are chosen as study cases.

Keywords: stream cipher, keygenerator, linear feedback shift register, pseudo random generator, run.

دراسة مقارنة لحساب خاصية الانطلاق في انظمة التشفير

د. اياد غازي ناصر الشمري* **عامر عبد المجيد عبد الرحمن****
*وزارة التربية / المديرية العامة للتعليم المهني
**كلية التربية للبنات - قسم الحاسبات

مستخلص

ان تطبيقات التشفير تتطلب استخدام مولدات متتابعات شبه عشوائية اكثر من اي تطبيقات اخرى. ان عشوائية الشفرة لا تعني فقط الاحصاءات العشوائية، وان كانت هي جزء منه، فعلى المتتابعة شبه العشوائية الشفرية الامينة ان تكون غير قابلة للتخمين.

المتتابعات العشوائية يجب ان تحقق خواص العشوائية، واحدهذه الخواص هي خاصية الانطلاق (سلسلة من الثنائيات المتشابهة). فهذه المتتابعات يجب ان يكون لها نفس العدد من الاصفار والواحدات (0,1)، وحوالي نصف الانطلاقات يجب ان تكون بطول (1) وربعها بطول (2) وثمنها بطول (3) وهكذا. وان توزيع الانطلاقات الصفرية والواحدية يجب ان يكون متساوي.هذه الاختبارات يمكن قياسها نظريا ومن ثم مقارنهاإحصائيا باستخدام اختبار مربع كاي.

في هذا البحث تم حساب مقياس الانطلاق لمولد المفاتيح قبل عملية التنفيذ او الانشاء (برمجيا او ماديا). تم اختيار نظم التشفير: الخطي، الضربيوبرور كحالات دراسية لهذا البحث.

1. Introduction

Cryptography is the study of mathematical techniques which are related to the aspects of information security such as confidentiality, data integrity, entity authentication, and data origin authentication [1].

It is the art or science encompassing the principles and methods of transforming an intelligible message into one that is unintelligible, and then retransforming that message back to its original form [2].

In 1989, Staffelbach and Meier [3] presented two new so-called fast correlation attacks which are more efficient than Siegenthaler's attack in the case where the component LFSRs have sparse feedback polynomials, or if they have low-weight polynomial multiples (e.g., each having fewer than 10 non-zero terms) of not too large a degree.

Gustafson [4] considered alternative statistics for the runs test and the autocorrelation test. Gustafson, Dawson, and Golić proposed a new repetition test which measures the number of repetitions of l-bit blocks. The test requires a count of the number of patterns repeated, but does not require the frequency of each pattern.

The security of GSM conversation is based on usage of the A5 family of stream ciphers. Many hundred million customers in Europe are protected from over-the-air piracy by the stronger version in this family, the A5/1 stream cipher. Other customers on other markets used the weaker version, A5/2. The approximate design of A5/1 leaked in 1994, and in 1999 the exact design of both A5/1 and A5/2 was discovered by Briceno[5]. A lot of investigations of the A5 stream ciphers has been followed.

At FSE 2004, a new stream cipher called VMPC (ventromedial prefrontal cortex) [6] was proposed by BartoszZoltak, which appeared to be a modification of the RC4 stream cipher. In cryptanalysis, a linear distinguishing attack is one of the most common attacks on stream ciphers. In this paper it was claimed that VMPC is designed especially to resist distinguishing attacks [7].

Recently, a new European project eSTREAM [8] has started, and at the first stage of the project 35 new proposals were received in May 2005. Although many previous stream ciphers were broken, collected cryptanalysis experience allowed strengthening new proposals significantly, and there are many of them that are strong against different kinds of attacks. One such good proposal was the new stream cipher Grain.

The stream cipher Grain was developed by a group of researchers M. Hell, T. Johansson, and W. Meier, and was especially designed for being very small and fast in hardware implementation. It uses the key of length 80 bits and the IV is 64 bits, its internal state is of size 160 bits. Grain used a nonlinear feedback shift register (NLFSR) and a linear feedback shift register (LFSR), and the idea to use NLFSR is quite new in modern cryptography [9].

Dragon is a word oriented stream cipher submitted to the eSTREAM project, designed by a group of researchers, Ed Dawson et al. It is a word oriented stream cipher that operates on key sizes of 128 and 256 bits. The original idea of the design is to use a Nonlinear Feedback Shift Register (NLFSR) and a linear part (counter), combined by a filter function to generate a new state of the NLFSR and produce the keystream. The internal state of the cipher is 1088 bits, which is updated by a nonlinear function denoted by F. This function is also used as a filter function producing the keystream. The idea to use a NLFSR is quite modern, and there are not many cryptanalysis techniques on NLFSRs yet developed[10].

2. Number Theory

Definition (1)[11]**:**A positive integer n>1 that has only two distinct factors, 1 and n itself (when these are different), is called **prime**; otherwise, it is called **composite**. The first few prime numbers are: 2,3,5,7,11,13,17,….

Theorem (1)[12]: (the fundamental theorem of arithmetic)

Any positive integer n>1 can be written uniquely in the following prime factorization form:

$$n=p_1^{\alpha_1}p_2^{\alpha_2}\cdots p_k^{\alpha_k}=\prod_{i=1}^{k}p_i^{\alpha_i} \quad \ldots(1)$$

where $p_1<p_2<\ldots<p_k$ are primes, and $\alpha_1, \alpha_2,\ldots, \alpha_k$ are positive integers.

Definition (2)[13]: Let a and b be two integers, not both zero. The largest divisor d s.t.d|a and d|b is called the **greatest common divisor** (gcd) of a and b, which is denoted by gcd(a,b).

Definition (3)[13]: Let a and b be two integers, not both zero. d is a common multiple of a and b, the least common multiple (lcm) of a and b, is the **smallest common multiple**, which is denoted by lcm(a,b).

Definition (4)[13]: Integers a and b are called **relatively prime** if gcd(a,b)=1. we say that integers $n_1,n_2,\ldots n_k$ are relatively prime if, whenever $i\neq j$, we have $\gcd(n_i,n_j)=1$, $\forall i,j,\ 1\leq i,j\leq k$.

Theorem (2)[12]: Suppose a and b are two positive integers.

If $a=\prod_{i=1}^{k}p_i^{\alpha_i}$ and $b=\prod_{i=1}^{k}p_i^{\beta_i}$, then

$$\gcd(a,b)=\prod_{i=1}^{k}p_i^{\varepsilon_i}, \text{ where } \varepsilon_i=\min(\alpha_i,\beta_i),\ \forall i,\ 1\leq i\leq k. \ldots(2)$$

$$\text{lcm}(a,b)=\prod_{i=1}^{k}p_i^{\delta_i}, \text{ where } \delta_i=\max(\alpha_i,\beta_i),\ \forall i,\ 1\leq i\leq k.\ldots(3)$$

Theorem (3)[12]: Suppose a and b are two positive integers, then

$$\text{lcm}(a,b)=\frac{a.b}{\gcd(a,b)}\ldots(4)$$

3. Terminology [14]

Cryptography (from the Greek Kryptós, "hidden" and gráphein, "to write") is the study of principles and techniques by which information can be concealed in ciphertexts and later revealed by legitimates users employing the secret key, but in which it is either impossible or computationally infeasible for an unauthorized person to do so. Cryptanalysis (from the Greek Kryptós, and analyéin "to loosen") is the science (and art) of recovering information from ciphertexts without knowledge of the key. Both terms are subordinate to the more general term Cryptology (from the Greek Kryptós, and logos, "word"). The cryptography is concerned in Encryption and Decryption processes.

Now we have to present some important notations:

Message space M: a set of strings (plaintext messages) over some alphabet, that needs to be encrypted.

Ciphertext space C: a set of strings (ciphertexts) over some alphabet that has been encrypted.

Key space K: a set of strings (keys) over some alphabet, which includes the encryption key e_k and the decryption key d_k.

The Encryption process (algorithm) E: $Ee_k(M)=C$.

The Decryption process (algorithm) D: $Dd_k(C)=M$.

The algorithms E and D must have the property that:

$Dd_k(C)=Dd_k(Ee_k(M))=M$.

The above situations shown in figure (1).

Figure (1) Encryption Process.

4. Stream Cipher systems [15, 16]

Stream ciphersare an important class of encryption algorithms. They encrypt individual characters (usually binary digits) of a plaintext message one at a time, using an encryption transformation which varies with time. The main properties of stream ciphers separating them from block ciphers are that the encryption function works on individual symbols (letters) of the underlying alphabet and that the encryption function is time-varying.

In stream ciphers, the message units are bits, and the key is usual produced by a random bit generator. The plaintext is encrypted on a bit-by-bit basis.

The key is fed into random bit generator to create a long sequence of binary signals. This "key-stream" k is then mixed with plaintext m, usually by a bit wise XOR (Exclusive-OR modulo 2 addition) to produce the ciphertext stream, using the same random bit generator and seed.

There is a vast body of theoretical knowledge on stream ciphers, and various design principles for stream ciphers have been proposed and extensively analyzed. However, there are relatively few fully-specified stream cipher algorithms in the open literature. This unfortunate state of affairs can partially be explained by the fact that most stream ciphers used in practice tend to be proprietary and confidential. By contrast, numerous concrete block cipher proposals have been published, some of which have been standardized or placed in the public domain. Nevertheless, because of their significant advantages, stream ciphers are widely used today, and one can expect increasingly more concrete proposals in the coming years.

5. Combination Generator

One approach is to use NLFSRs in parallel; their outputs combined using an n-input binary Boolean function or combining function (CF). [17].

Because LFSRs are inherently linear, one technique for removing the linearity is to feed the outputs of several parallel LFSRs into a non-linear Boolean function to form a combination generator. Various properties of such a combining functionare critical for ensuring the security of the resultant scheme, for example, in order to avoid correlation attacks. [18].

Since a well-designed system should be secure against known plaintext attacks, an LFSR should never be used by itself as a keystream generator. Nevertheless, LFSRs are desirable because of their very low implementation costs[19].

For essentially all possible secret keys, the output sequence of an LFSR based keystream generator should have the following properties:

1. large period.
2. large linear complexity.
3. good statistical properties.

It is emphasized that these properties are only necessaryconditions for a keystream generator to be considered cryptographically secure. Since mathematical proofs of security of such generators are not known, such generators can only be deemed computationally secure after having withstood sufficient public scrutiny [20].

The LFSRs in an LFSR-based keystream generator may have known or secret connection polynomials. For known connections, the secret key generally consists of the initial contents of the component LFSRs. For secret connections, the secret key for the keystream generator generally consists of both the initial contents and the connections.

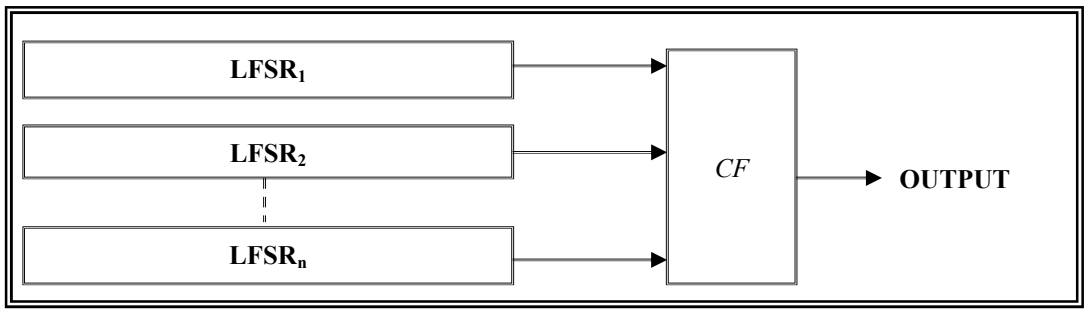

Figure (2) n-LFSR's Generator with Combining Function

5.1Linear Generator[21]

The Linear generator, illustrated in figure (3), is defined by n-maximum-length LFSRs whose lengths $r_1, r_2,\ldots, r_n$, where $n\in Z^+$are pair wise relatively prime, with XOR combining function:

$F(x_1,x_2,..,x_n) = x_1\oplus x_2\oplus\ldots\oplus x_n\ldots$ (5)

This generator is considered weak, despite its good randomness, because of its weak linear complexity.

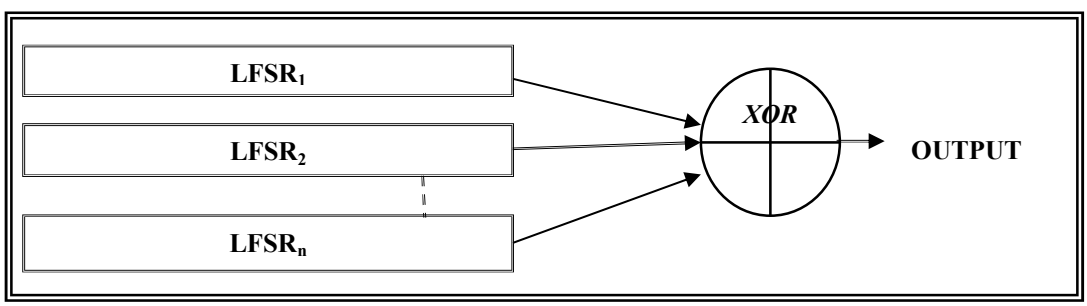

Figure (3) n-Linear Generator.

5.2 Product Generator [21]

The Product generator, illustrated in figure (4), is defined by n-maximum-length LFSRs whose lengths $r_1, r_2,\ldots, r_n$, where $n\in Z^+$are pair wise relatively prime, with AND combining function:

$$F(x_1,x_2,..,x_n) = x_1\bullet x_2\bullet\ldots\bullet x_n = \prod_{i=1}^{n} x_i \ \ldots (6)$$

This generator considered weak, despite its good linear complexity, because of its weak randomness.

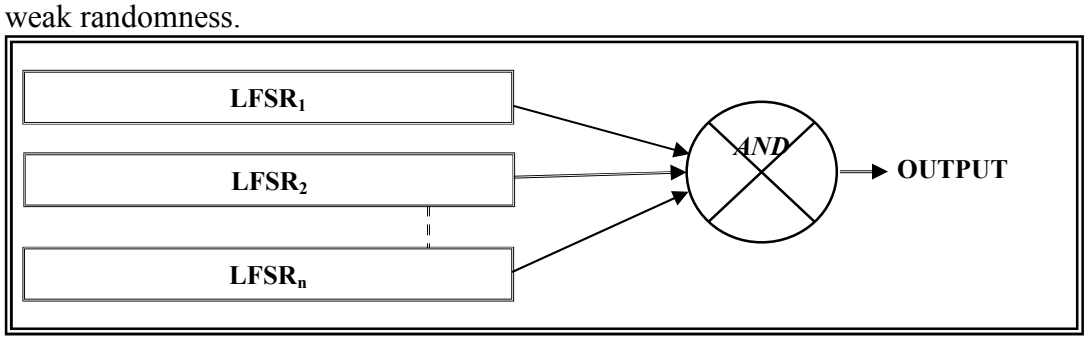

Figure (4) n-Product Generator.

5.3 Threshold Generator [22].

This generator as usual uses combining function called Majority function which is balance and symmetric (which expect that this generator produces pseudo random generator). This generator illustrated in figure (5) tries to get around the security problems by using a variable number of LFSR's. The theory is that if you use a lot of LFSRs, it's harder to break the cipher.

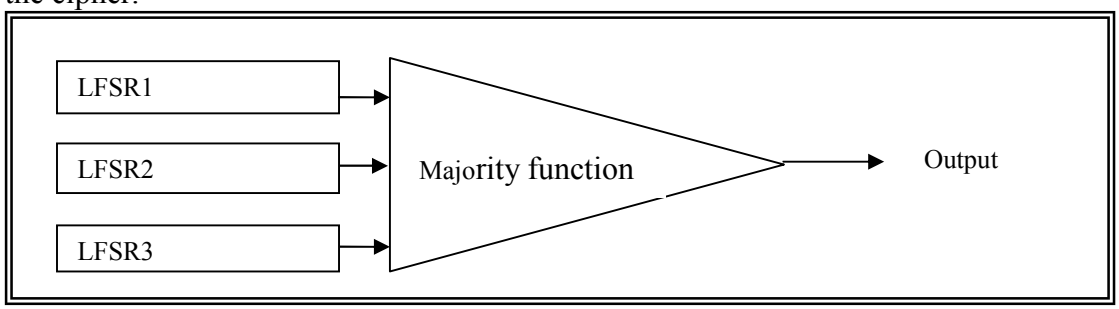

Figure (5) Threshold generator

Take the output of a large number of LFSRs (use an odd number of them). Make sure that the lengths of all the LFSRs are relatively prime and all the feedback polynomials are primitive: maximize the period. If it is more than the half the output bits are 1, then the output of the generator is 1. If more than half the output bits are 0, then the output of the generator is 0.

With three LFSRs, the output generator can be written as:

The Threshold generator using the non-linear combining function s.t:

$F_3(x_1,x_2,x_3)=x_1x_2\oplus x_1x_3\oplus x_2x_3$.

From the combing function of this generator, we expect that it has a larger linear complexity (LC) [21]:

$LC(S) = r_1r_2 + r_1r_3 + r_2r_3$

where r_1, r_2, and r_3 are the lengths of the first, second, and third LFSRs.

6. Implementation of Run Postulate

Lemma (1) shows a linear relation between the periodicity of n-KG and the periodicity of combination of one or number of combined LFSR in the system.

$$P(S)=\sum_{k=0}^{n-1}(-1)^k\sum_{t=1}^{C_k^n}P(R^t_{n-k}) \quad \ldots(7)$$

Such that $P(R^t_m)=2^{R^t_m}-1=2^{R^t_m-1}-1+2^{R^t_m-1}=\sum_{a=0}^{1}N_{R^t_m}(a)$ …(8)

$N_{R^t_m}(a)$ denotes the number of binary of kind "a" of the sequence which is generated from the component R^t_m.

Depending on equation (7) and equation (5) we can find $N_S(a)$, s.t.

$$N_S(a)=\sum_{k=0}^{n-1}(-1)^k\sum_{t=1}^{C_k^n}N_{R^t_{n-k}}(a)\ ,\ a=0,\ 1 \quad \ldots(9)$$

where $C_k^n=\dfrac{n!}{k!(n-k)!}$

Equation (9) shows the relation between $N_S(a)$ and number of 0's (1's) obtained from the possible combination of the length of combined LFSR's.

Example (1):

Let $r_1=2$ and $r_2=3$, then $C_0^2=1, C_1^2=2,\ R_1^1=r_1=2,\ R_1^2=r_2=3,\ R_2^1=r_1+r_2=5$.

$$N_S(a)=\sum_{k=0}^{1}(-1)^k\sum_{t=1}^{C_k^2}N_{R_{2-k}^t}(a),\ a=0,1 \quad \ldots(10)$$

$$N_S(a)=N_{R_2^1}(a)+(-1)[N_{R_1^1}(a)+N_{R_1^2}(a)]=N_5(a)+(-1)[N_2(a)+N_3(a)]$$

The results of calculating $N_S(a)$ from equation (9) are as follows:

a	$N_2(a)$	$N_3(a)$	$N_5(a)$	$N_S(a)$
0	1	3	15	11
1	2	4	16	10

Now we can calculate the runs of S depending on the runs of the maximal sequences which are generated from the combinations R_m^t which are known.

Let $N_j^{R_m^t}(a)$ be the number of runs (a=0 for gaps and a=1 for blocks) with length j for the combinations R_m^t, and $N_j^{S}(a)$ be the number of runs of kind *a* with length *j* for the sequence S.

We can reformulate the Golomb's second postulate by:

$$\left.\begin{aligned} &N_{R_m^t}^{R_m^t}(1)=N_{R_m^t-1}^{R_m^t}(0)=1\\ &N_{R_m^t}^{R_m^t}(0)=N_{R_m^t-1}^{R_m^t}(1)=1\\ &N_j^{R_m^t}(a)=2^{R_m^t-j-2}\quad \text{When } 1\le j\le R_m^t-2,\ a=0,1 \end{aligned}\right\} \ldots(11)$$

The next lemma discuss the relation between the $P(R_m^t)$ and the elements of the equation (11).

Lemma (2):

$$P(R_m^t)=\sum_{j=1}^{R_m^t}j\sum_{a=0}^{1}N_j^{R_m^t}(a) \qquad \ldots(12)$$

Proof:

$$N_{R_m^t}(a)=1\cdot N_1^{R_m^t}(a)+2\cdot N_2^{R_m^t}(a)+\cdots+R_m^t\cdot N_j^{R_m^t}(a) \ldots(13)$$

$$N_{R_m^t}(a)=\sum_{j=1}^{R_m^t}j\cdot N_j^{R_m^t}(a) \ldots(14)$$

and since

$$P(R_m^t)=2^{R_m^t}-1=N_{R_m^t}(0)+N_{R_m^t}(1)=\sum_{a=0}^{1}N_{R_m^t}(a) \ldots(15)$$

using equation(14)

$$P(R_m^t)=\sum_{a=0}^{1}\sum_{j=1}^{R_m^t}j\cdot N_j^{R_m^t}(a)=\sum_{j=1}^{R_m^t}j\sum_{a=0}^{1}N_j^{R_m^t}(a)$$

■

Linear System (n-LKG)

Now we are ready to calculate the runs of the sequence S generated from linear system.

First, let's rearrange the LFSR combined in the linear system (of course the rearrangement does not effect on any results, since the XOR function is symmetric and commutative) in increasing (decreasing) order. Its naturally that $R_m^1 \le R_m^2 \le .... \le R_m^{C_m^n}$, $\forall 1 \le m \le n-1$ and $R_m^t < R_n^1$, $\forall 1 \le t \le C_m^t$, $CD(P(r_i))=1$, $\forall i$.

The length j of runs of the sequence S will be in the range $1 \le j \le R_n^1$ passing all the combination less than R_n^1. So, when using equations (15) we get:

$$P(S) = \sum_{k=0}^{n-1} (-1)^k \sum_{t=1}^{C_k^n} \sum_{j}^{R_{n-k}^t} j \cdot \sum_{a=0}^{1} N_j^{R_{n-k}^t}(a) \quad \ldots(16)$$

If we extend the sum R_{n-k}^t to R_n^1 (since the maximum run of S is R_n^1) considering that $N_j^{R_m^t}(a) = 0$, when $j > R_m^t$, then equation (16) will be:

$$P(S) = \sum_{k=0}^{n-1} (-1)^k \sum_{t=1}^{C_k^n} \sum_{j}^{R_n^1} j \cdot \sum_{a=0}^{1} N_j^{R_{n-k}^t}(a) \quad \ldots(17)$$

Since the maximum run of S is R_n^1, then it's obvious that:

$$P(S) = \sum_{j=0}^{R_n^1} j \cdot \sum_{a=1}^{1} N_j^S(a) \quad \ldots(18)$$

Comparing equation (17) and (18) we get:

$$\sum_{j=1}^{R_n^1} j \cdot \sum_{a=0}^{1} N_j^S(a) = \sum_{k=0}^{n-1} (-1)^k \sum_{t=1}^{C_k^n} \sum_{j=1}^{R_n^1} j \cdot \sum_{a=0}^{1} N_j^{R_{n-k}^t}(a) \quad \ldots(19)$$

$$N_j^S(a) = \sum_{k=0}^{n-1} (-1)^k \sum_{t=1}^{C_k^n} N_j^{R_{n-k}^t}(a) \quad \ldots(20)$$

From equation (20) we can calculate the runs j of S, $1 \le j \le R_n^1$ for the linear system.

Table (1) shows the calculating of $N_j^S(a)$, for S generated from linear system.

From equation (13) we can calculate $N_S(a)$ by

$$N_S(a) = \sum_{j=1}^{R_n^1} j \cdot N_j^S(a) \quad \ldots(21)$$

Example (2):

Table (1) describes the calculating of runs j of the sequence S generated from the linear system using $r_1=2$, $r_2=3$.

Table (1) Calculates of runs j of S generated from the n-LKG

		+1	-1			
j	a	R_2^1=5	R_1^1=2	R_1^2=3	$\overset{S}{N}_j(a)$	$N_S(a)$
1	0	4	1	1	2	2
	1	4	0	1	3	3
2	0	2	0	1	1	2
	1	2	1	0	1	2
3	0	1	0	0	1	3
	1	1	0	1	0	0
4	0	1	0	0	1	4
	1	0	0	0	0	0
5	0	0	0	0	0	0
	1	1	0	0	1	5
Sum		2^5-1=31	2^2-1=3	2^3-1=7	P(S)=21	

Notice that from table (1) and equation (20), there is a little difference between $\overset{S}{N}_j(0)$ and $\overset{S}{N}_j(1)$ values that will give a balance output which implies that S will pass the run test successfully. The next two lemmas prove the 2nd Golomb randomness for runs of linear system.

The next lemma proves the double relation between $\overset{S}{N}_{j+1}(a)$ and $\overset{S}{N}_j(a)$.

Lemma (3):

For the sequence S generated from linear system:

$$\overset{S}{N}_{j+1}(a) \approx \frac{1}{2}\overset{S}{N}_j(a), \text{ for } 1 \le j \le R_n^1 \ldots (22)$$

Proof:

From equation (21) we have:

$$\overset{S}{N}_{j+1}(a) = \sum_{k=0}^{n-1}(-1)\sum_{t=1}^{C_k^n}\overset{R_{n-k}^t}{N}_{j+1}(a) \quad (23)$$

Since $\overset{R_m^t}{N}_{j+1}(a) \approx \frac{1}{2}\overset{R_m^t}{N}_j(a)$ from 2nd Golomb postulate, then:

$$\overset{S}{N}_{j+1}(a) \approx \sum_{k=0}^{n-1}(-1)\sum_{t=1}^{C_k^n}\frac{1}{2}\overset{R_{n-k}^k}{N}_j(a) = \frac{1}{2}\sum_{k=0}^{n-1}(-1)\sum_{t=1}^{C_k^n}\overset{R_{n-k}^t}{N}_j(a) = \frac{1}{2}\overset{S}{N}_j(a) \quad \blacksquare$$

The next lemma describes the approximation frequency of gaps and blocks which are from the same length.

Lemma (4):

For the sequence S generated from linear system

$$\overset{S}{N}_j(0) \approx \overset{S}{N}_j(1) \quad \text{, for } 1 \le j \le R_n^1 - 2$$

Proof:
From equation (21) we have:

$$\overset{S}{N}_j(0)=\sum_{k=0}^{n-1}(-1)\sum_{t=1}^{C_k^n}\overset{R_{n-k}^t}{N}(0)\ldots(24)$$

From 2nd Golomb postulate, we have $\overset{R_m^t}{N}_j(0)\approx\overset{R_m^t}{N}_j(1)$, for $1\le j\le R_n^1-2$ then:

$$\overset{S}{N}_j(0)=\sum_{k=0}^{n-1}(-1)\sum_{t=1}^{C_k^n}\overset{R_{n-k}^t}{N}_j(1)=\overset{S}{N}_j(1)$$ ■

Product System (n-PKG)

If the equivalent LFSR with length L, then there is no guarantee that the maximum gap is with length L-1, since the generated S is not maximalsequence, while we may find gap with length Gms.t.

$$G_m=\sum_{i=1}^{n}G_{mi}=\sum_{i=1}^{n}(r_i-1)=\sum_{i=1}^{n}r_i-n\ldots(25)$$

Where G_{mi} denotes length of the maximum gap in the LFSR number i, and not repeated again. G_m happened from the union of all the maximum gaps in every combined LFSR in n-PKG, and it's obvious that $\overset{S}{N}_{G_m}(a)=1$.

The maximum block B_m happened from the intersection of all maximum block in every combined LFSR in n-PKG, so if B_{mi} is the length of the maximum block of the LFSR number i, then

$B_m=\min(B_{m1},B_{m2},\ldots,B_{mn})=\min(r_1,r_2,\ldots,r_n)=r_m$

Where r_m is the minimum LFSR in n-PKG, it happened more than one time. In fact $G_m\neq B_m$, so we can't compare them.

Example (3):
Let $r_1=2$, $r_2=3$ and $r_3=5$, table (2) shows the statistics of the number of runs (gaps and blocks) of the sequence S generated from of 3-PKG for three different initial values of combined LFSR's.

Table (2) the statistics of the number of runs of 3-PKG

Initial value	a	Number of runs																
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
I_1	0	16	28	8	12	16	9	4	5	2	1	3	0	2	3	1	1	1
	1	96	16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
I_2	0	16	29	9	12	15	9	4	5	2	1	3	0	2	3	1	1	1
	1	96	16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
I_3	0	16	29	9	13	16	9	3	5	2	1	3	0	2	3	1	1	1
	1	96	16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

From table (2) we notice the unbalance between the numbers of gaps with each other, from a side, and with numbers of blocks from the same length, from another side. Notice that the maximum block is of length 2, and no relation of double between the number of runs with length j and length j+1. These results will make the sequence S generated from n-PKG not satisfies the 2nd Golomb's postulate.

Brüer System (3-BKG)

Since the output of the majority function is balance so it includes strings with closed distribution, and since the runs are part from these strings, then they will be distributed closely.

Example (4):

Let $r_1=5$, $r_2=7$ and $r_3=11$, table (3) shows the statistics of the number of runs (gaps and blocks) of the sequence S generated from 3-BKG.

Table (3) the statistics of the number of runs of 3-BKG

j	Number of runs	
	0	1
1	1032527	1003599
2	512688	509442
3	252154	260403
4	139916	113067
5	46064	52952
6	29353	32882
7	17702	20539
8	4194	6037
9	2787	3054
10	974	808
11	414	381
12	255	227
13	127	97

From table (3) we notice the semi-balance between the numbers of gaps with each other, from a side, and with numbers of blocks from the same length, from another side. Notice that the relation of double between the number of runs with length j and length j+1. These results will make the sequence S generated from 3-BKG satisfies the 2nd Golomb's postulate.

7. Applying of Chi-Square of Run Test on Study Cases

In this section we will apply chi-square test on the results gotten from calculations of the run postulates on three study cases.

Let M be the number of categories in the sequence S, c_i be the category i, $N(c_i)$ be the observed frequency of the category c_i, Pr_i the probability of occurs of the category c_i, then the expected frequency E_i of the category c_i is $E_i=P(S)\cdot Pr_i$, the T (chi-square value) can be calculated as follows:

$$T=\sum_{i=1}^{K}\frac{(N(c_i)-E_i)^2}{E_i} \quad \ldots(26)$$

Assuming that T distributed according to chi-square distribution by $\upsilon=M-1$ freedom degree by α as significance level (as usual $\alpha=0.05\%$), which it has T_0 as a pass mark. If $T\leq T_0$ then the hypothesis accepted and the sequence pass the test, else we reject the hypothesis and the sequence fails to pass the test, this mean that T not distributed according to chi-square distribution.

In order to test our results we have to suggest an example suitable to our three studied cases. Let n=3, $r_1=7$, $r_2=9$ and $r_3=11$. P(S)= 132844159, $E_i=66422079.5$.

In this test $\alpha=0.05\%$, with $\upsilon=2(Maxl-1)$ freedom degrees, $E_j=(P(S)-j+3)/2^{j+2}\approx P(S)/2^{j+2}$, where j is the length of run (gap or block), and maxl=27 is the maximum length.

Note: We will apply this test and auto correlation test on 3-LKG only since the applications of these test on other cases studies are same.

For the 3-LKG, Table (4) shows the results of frequencies of run taken from equation (26).

Table (4) Run frequencies using equation (15)

j	1	2	3	4	5	6	7	8	9
$\overset{S}{N}(a)$	16597328	8298664	4149332	2074666	1037333	518666	259461	129731	64665
E_j	16605520	8302760	4151380	2075690	1037845	518923	259461	129731	64865
j	10	11	12	13	14	15	16	17	18
$\overset{S}{N}(a)$	32433	16216	8108	4054	2027	1013	507	253	127
E_j	32433	16216	8108	4054	4054	1013	507	253	127
j	19	20	21	22	23	24	25	26	27
$\overset{S}{N}(a)$	63	32	16	8	4	2	1	0	1
E_j	63	32	16	8	4	2	1	0	1

$$T=\sum_{i=1}^{K}\frac{(N(c_i)-E_i)^2}{E_i}=\sum_{i=1}^{27}\sum_{a=0}^{1}\frac{(\overset{S}{N}_j(a)-E_i)^2}{E_i} \quad \ldots(27)$$

$=4.04+2.02+1.01+0.505+0.251+0.127+0+\cdots+0=7.953.$

$T=7.953<T_0=40.1$, then S generated from 3-LKG passes this test.

Runs Test

The run test counts the number of runs of ones (blocks) and runs of zeros (gaps) for each possible run length. For random data there should be an equal number of blocks and gaps. The expected number of blocks (gaps) of length i is:

$$e_i=\frac{\frac{n-i-1}{2}+2}{2^{i+1}} \quad \ldots(28)$$

A chi-squared test is performed on the bit stream to test for the goodness-of-fit of the number of blocks and gaps to this distribution. The chi-squared statistic is calculated as:

$$\chi^2=\sum_{i=1}^{k}\frac{(B_i-e_i)^2}{e_i}+\sum_{i=1}^{k}\frac{(G_i-e_i)^2}{e_i} \quad \ldots(29)$$

Where B_i is the number of blocks of length i, G_i is the number of gaps of length i, and Σ denotes the summation over all possible run k of length i such that $e_i \geq 5$. This is compared with 2k-2 degree of freedom.

8. Evaluation the results

8.1 Basic Evaluation

We can define the estimation process of the basic efficiency using the four basic efficiency criterions by the Basic evaluation. In this section we will evaluate the three study cases in two ways; theoretically and practically, we assume that these two ways must be identical. The evaluation process in this paper is relative evaluation between the three study cases but not absolute.

In order to obtain fair evaluation we assume having three relatively prime MLFSR's with length r_1, r_2 and r_3, combined with each other by combining function of each case study.

8.2 Theoretical Evaluation

The evaluation process in this subsection is done by theoretical estimation. Let S_L, S_P and S_B be the sequences generated from the linear, product and Brüer systems respectively. The evaluation process have been done by the following aspects:
Three sequences will have the same period, s.t.

$$P(S)=\prod_{i=1}^{3} P(S_i).$$

The product system has the largest linear complexity, s.t.
$LC(S_P) > LC(S_B) > LC(S_L)$.

The linear andBrüer systems will produce good statistical random sequences since the given balance string of bits, while the sequence generates from the product system will fail to pass the randomness tests.

The linear system has the largest correlation immunity since it has 0.5 correlation probability for all its combined LFSR's. But that is not true for the product andBrüer systems because of their high non-linearity order.
$CI(S_L) > CI(S_P) = CI(S_B)$.

The decision is that, the Brüer system will be the best of the other two systems if the weak point of correlation can be manipulated.

8.3 Practical Evaluation

The KG can be evaluated practically if it passes (or not) some successful value, this has been done by using weight level value for each efficiency criterion. These weights can be estimated according to cryptanalysis or attack means done on KG's.

For instance, we may suggest to use the passing value 51%, giving the following weights for each criterion; randomness (W_R=50%), periodicity (W_P=10%), linear complexity (W_{LC}=25%) and correlation immunity (W_{CI}=15%). Table (5) shows the results of practical evaluation for the three study cases using four efficiency criterions.

Table (5) The results of practical evaluation for the three study cases

Study Cases	weight levels				Results of evaluation
	W_R%	W_P%	W_{LC}%	W_{CI}%	
Linear system	50	10	0	15	75%
Product system	0	10	25	0	35%
Brüer system	50	10	20	10	90%

Notice that Brüer system is the best between other two systems since it gets 90% as pass value.

Example(5): (basic statistical tests)

Consider the (non-random) sequence s of length n = 160 obtained by replicating the following sequence four times: 11100 01100 01000 10100 11101 11100 10010 01001.
Frequency test: n_0=84, n_1=76, and the value of the statistic X_1 is 0.4.
Autocorrelation test: If τ=3, $n_0(3)$=80 and $n_1(3)$=77. The value of the statistic X_5 is 0.115.
Runs test: Here E_1=20.25, E_2=10.0625, E_3=5, and k=3. There are 25, 4, 5 blocks of lengths 1, 2, 3, respectively, and 8, 20, 12 gaps of lengths 1, 2, 3, respectively. The value of the statistic X_4 is 31.7913.
For a significance level of α=0.05, the threshold values for X_1, X_2, X_3, X_4, and X_5 are 3.8415, 7.8415, 7.8415, 9.4877, and 0.115, respectively.

9. Conclusions

The following are some points concluded from this paper:

1- In literature survey the linear and Brüer cryptosystems are proved generates good randomness sequences statistically, while in this paper we prove that these cryptosystems generates really good random sequences deterministically.

2 - Golomb proves that the LFSR generates random sequence deterministically. In this paper we notice that the theoretical and the practical evaluation which have been done in the three studies cases are identical to each other.

3 - Although the linear system passes the theoretical and the practical evaluation, we don't recommend using it as cryptosystems, because the weakness of its linear complexity.

4 - New efficient criterions may be discovered for KG. These new criterions may depend or not on earlier known criterions since these criterions appeared by the developing of cryptanalysis and design which may be applied on KG's.

References

[1]. Stinson, D. R., "Cryptography: Theory and Practice", CRC Press, 1995.

[2]. Xiang-Yang Li "Cryptography and Network Security",1995

[3]. Staffelbach, O. and Meier, W., "Fast Correlation Attacks on Certain Stream Ciphers", Journal of Cryptology, 1 (1989), 159–176.

[4].Gustafson, H., "Statistical Analysis of Symmetric Ciphers", PhD thesis, Queensland University of Technology, 1996.

[5]. Briceno, M, Goldberg, I., and Wagner, D., "A Pedagogical Implementation of A5/1", 1999.

[6]. Zoltak, B., "VMPC One-Way Function and Stream Cipher", In B. Roy and W. Meier, editors, Fast Software Encryption 2004, volume 3017 of Lecture Notes in Computer Science, pages 210–225. Springer-Verlag, 2004.

[7].Paul, S. and Prenel, B., "A New Weakness in the RC4 Keystream Generator and an Approach to Improve the Security of the Cipher", In B. Roy and W. Meier, editors, Fast Software Encryption 2004, volume 3017 of Lecture Notes in Computer Science, pages 245–259. Springer-Verlag, 2004.

[8]. eSTREAM: ECRYPT Stream Cipher Project, IST-2002-507932. Available at http://www.ecrypt.eu.org/stream/ (accessed September 29, 2005), 2005.

[9]. Johnson, D. W. and Johnson, F. P., "Joining Together: Group Theory and Group Skills", Allyn& Bacon, July 2002.

[10]. Chen, K. and et al, "Dragon: A Fast Word Based Stream Cipher", eSTREAM, ECRYPT Stream Cipher Project, Report 2005/006 (2005-04-29), 2005.

[11]. Ribenboim, P., "The New Book of Prime Number Records", Springer-Verlag, 1996.

[12]. Apostol, T. M., "Introduction to Analytic Number Theory", Corrected 5th Printing, Undergraduate Texts in Mathematics, Springer-Verlag, 1998.

[13]. Andrews, G. G, "Number Theory", Dover Publications, Oct. 1994.

[14]. Motwani, R. and Raghavan, P., "Randomized Algorithms", Cambridge University Press, 1995.

[15]. Ekdhal, P., "On LFSR based Stream Ciphers Analysis and Design", Ph.D. Thesis, November 21, 2003.

[16]. Rivest, R. L., "Hand Book of Applied Cryptography", John Wiley & Sons,1997.

[17]. Rechberger, C., "Side Channel Analysis of Stream Ciphers", Master's Thesis, Graz, Austria, 2004.

[18]. Mister S., "Cryptanalysis of RC4-like Stream Ciphers", M.Sc. thesis,Queen's University, May 1998.

[19]. Matthews, R. A. J., "The Use of Genetic Algorithms in Cryptanalysis", Cryptologia, vol. XVII, no 2, pp. 187-201, 1993.

[20]. Menezes, A. P. van Oorschot, P. and Vanstone, S., "Handbook of Applied Cryptography", CRC Press, 1996.

[21]. Schneier B., "Applied Cryptography", John Wiley & Sons, 1996.

[22]. Brüer, J.O., "On Pseudo Random Sequences as Crypto Generators", Proceedings of the International Zurich Seminar on Digital Communication, Switzerland, 1984.