N-adic Summation-Shrinking Generator

Basic properties and empirical evidences

Zhaneta Tasheva

Assistant Prof. Eng. PhD.
NMU "V. Levski"
Faculty of Artillery and Air Defense, Shoumen, Bulgaria
Phone: +359 54 5 23 71
e-mail: tashevi86@yahoo.com

Borislav Bedzhev

Assoc. Prof. Eng. DSc.
NMU "V. Levski"
Faculty of Artillery and Air Defense, Shoumen, Bulgaria
Phone: +359 54 4 64 38
e-mail: bedzhev@mail.pv-ma.bg

Borislav Stoyanov

Assistant Prof. Mag. PhD. Student
Shoumen University
Faculty of Computer Informatics, Shoumen, Bulgaria
Phone: +359 54 4 78 48
e-mail: bpstoyanov@abv.bg.

ABSTRACT

The need of software-flexible stream ciphers has led to several alternative proposals in the last few years. One of them is a new Pseudo Random Number Generator (PRNG), named N-adic Summation-Shrinking (NSumSG), which architecture is described in this paper. It uses N-1 parallel working slave summation generators and one N-adic summation generator, controlling the nonlinearity in the generator. The implementation, some properties and statistical tests of NSumSG are given.

The results from statistical analysis show that the sequence generated by NSumSG is uniform, scalable, uncompressible, whit large period; consistent and unpredictable. This gives the reason consider the NSumSG as suitable for a particular cryptographic application.

KEY WORDS

Cryptography, Encryption Algorithm, Shrinking Generator, Summation Generator, Stream Ciphers, PRNG, FCSRs.

SECTION 1

Introduction

The proliferation of computers and communications systems in the 1960s brought with it a demand from the private sector for means to protect information in digital form and to provide security services. The stream ciphers are an important tool for solving this problem. Despite of their large application, it is very hard or may be impossible to describe all factors, which influence over the performance quality of the stream ciphers. Anyway, surely it depends on their crypto resistance, velocity and effectiveness of hardware implementation. Mostly the crypto resistance of a stream cipher is connected with it ability to generate pseudo random sequence (*PRS* or *gamma*) with following properties:

(1) it should have enormous period;

(2) it should demonstrate uniform distribution of *d*-tuples (for a large range of *d*);

(3) it should exhibit a good structure (usually a lattice structure) in high dimensions.

Unfortunately, the mentioned factors are in contradiction, because if the structure of the stream cipher is simple in order to provide high performance velocity and cost-effective hardware implementation, then the crypto reliability is low. For instance, the classical fast and cheap *Linear Feedback Shift Registers (LFSRs)* are vulnerable to the so - named "Berlekamp–Massey crypto attack" [4], [5], [8]. This attack allows finding of all bits of a *LFSR* output sequence, if *2n* its consequent bits are known. Here *n* is the number of the cells connected in the *LFSR*. Having in mind the advantages of the stream ciphers with simple structure, recently some theorists [3], [4], [6] proposed a new approach to stream cipher design. The basic idea of this approach is building devices with high crypto reliability combining in some appropriate way crypto vulnerable, but fast and cheap elements (including *LFSR*). This meaning of stream cipher design leaded to introducing of a few new architectures. It should be mentioned the so-named summation generator, shrinking generator and *N-adic Feedback with Carry Shift Register (N-FCSR*) [2], [3], [13]. They are promising candidates for high-speed encryption applications due to their simplicity and provable properties.

With regard to positive features of the summation generator, shrinking generator and *N-FCSRs*, our paper is focused on the problem of synthesis of a derivative structure, named summation-shrinking generator.

The paper is organized as follows. First, the basics of the summation generator and shrinking generator are recalled. Second one their derivative structure, called *N*-adic Summation-Shrinking Generator (*NSumSG*) is presented. After then, the implementation and statistical analysis of *NSumSG* properties are given. Finally, the advantages and possible areas of application of our algorithm are discussed.

SECTION 2

Basic theory of the summation and shrinking generators

Principally the crypto resistance of a stream cipher, based on *LFSR*s, can be enhanced by two alternative methods. The first method uses an appropriate combining of the outputs of some *LFSR*s, as it is shown on Fig.1a. These gamma generators are called "Combination Generators". The other alternative is to generate the gamma as a non-linear function from conditions of the single *LFSR* triggers (Fig.1b). In this case the gamma generators are named "Filter Generators".

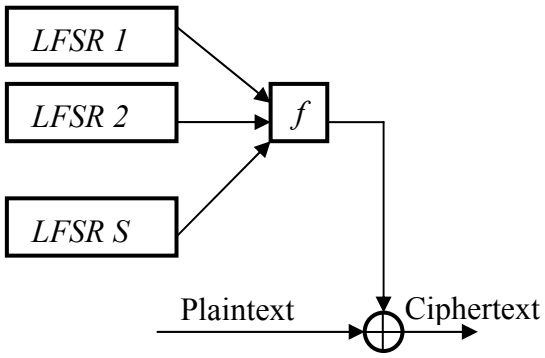

***Fig. 1a*:** *Combination generator*

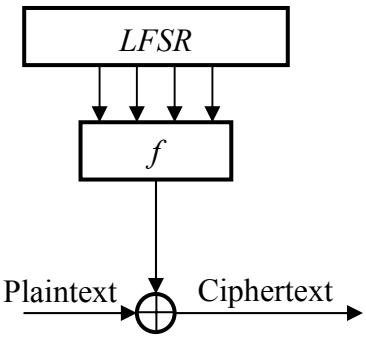

Fig. 1b:*Filter-generator*

Having in minded that:

- the filter-generators could be studied as a particular case of the combination generators when $S = 1$ on Fig. 1a;

- the combination generators are still being applied in some real communication and information systems [5], [7];

in the rest part of this report our attention shall be focused on the derivative structures of the combination generator.

As mentioned, the basic idea of the combination generator method is to create a hard-to-crack gamma sequence by an appropriate combining of some *PRSs*, whose technical realization is simple and cost-effective. The scheme, shown on Fig.1a, is an object of intensive research since 1984, because it is easy to generate *PRSs* with *LFSRs*. As a result of these efforts [6] the cryptologist Rueppel has proved that the combination generators have maximal linear complexity *L(x)* if:

- the all *LFSR*s have a feed-back loop, described with primitive irreducible polynomial (i.e. the created *PRSs* are *maximum length sequences* (shortly *m-sequences*));

- the periods T_i, $i = 1, 2, \ldots, s$ of the *PRSs*, generated by *LFSR*s, are different.

Here linear complexity $L(x)$ means the length of the binary *LFSR*, which can be constructed in the result of the Berlekamp-Massey crypto attack.

The Rueppel conditions are easy to realizing as a s-bit adder. This means that ***f*** from Fig.1a must be a full adder, which has $\log_2(s+1)$ triggers. In order to simplify the explanation, we shall suppose, that the *LFSRs* are only two. In this case, during the time interval from $j.\tau_0$ to $(j+1).\tau_0$ (here τ_0 is the period of the *LFSR*s clock-pulses) in *LFSR* triggers the sequences $A = a_j, a_{j+1}, ..., a_{j+r-1}$ and $B = b_j, b_{j+1}, ..., b_{j+r-1}$ are placed. In the adder the numbers, corresponding to the sequences *A* and *B*:

$$\begin{aligned} a &= a_{j+r-1}.2^{r-1} + ... + a_{j+1}.2 + a_j, \\ b &= b_{j+r-1}.2^{r-1} + ... + b_{j+1}.2 + b_j, \end{aligned} \tag{1}$$

are summed with carry. Then in the outputs of the adder the total sum $z = a + b$ is obtained. Here:

$$\begin{aligned} &Z = z_j, z_{j+1}, ..., z_{j+r-1}, \\ &z = z_{j+r-1}.2^{r-1} + ... + z_{j+1}.2 + z_j, \\ &z_i = a_i + b_i + \sigma_{i-1}, \quad i = j, j+1, ..., j+r-1 \\ &\sigma_i = a_i.b_i + a_i.\sigma_{i-1} + b_i.\sigma_{i-1}, \quad i = j, j+1, ..., j+r-1, \end{aligned} \tag{2}$$

and:

- z_j is the j^{th} element of combination generator output sequence;
- σ_i is the carry from the $(i-1)^{th}$ digit.

The basic idea of the combining generator can be realized as a shrinking generator also. In the shrinking generator, a control *LFSR* R_0 is used to select a portion of the output sequence of a second *LFSR* R_1. Therefore, the produced gamma (or the *keystream*) is a *shrunken* version (also known as an *irregularly decimated subsequence*) of the output sequence of R_1, as depicted in Fig. 2.

The algorithm of shrinking generator consists of the following steps:

(1) Registers R_0 and R_1 are clocked.

(2) If the output of R_0 is 1, the output bit of R_1 forms a part of the keystream.

(3) If the output of R_0 is 0, the output bit of R_1 is discarded.

Let R_0 and R_1 be maximum-length *LFSRs* of lengths L_0 and L_1, respectively, and let z be an output sequence of the shrinking generator formed by R_0 and R_1. If gcd(L_0, L_1) = 1, the z has period $(2^{L_1} - 1).\ 2^{L_0 - 1}$ [7]. The linear complexity $L(z)$ of z satisfies Eq. (3) [7]:

$$L_1.2^{L_0-2} < L(z) \le L_1.2^{L_0-1} \tag{3}$$

Suppose that the connection polynomials of R_0 and R_1 are chosen uniformly at random from the set of all primitive polynomials of degrees L_0 and L_1 over $\boldsymbol{Z_2}$. Then the distribution of patterns in z is almost uniform [7].

For maximum security, R_0 and R_1 should be maximum-length *LFSRs*, and their lengths should satisfy the condition $gcd(L_0, L_1) = 1$. Moreover, secret connection should be used. Subject to these constraints, if $L_0 \approx m$ and $L_1 \approx m$, the shrinking generator has a security level approximately equal to 2^{2m}. Thus, if $L_0 \approx 64$ and $L_1 \approx 64$, the generator appears to be secure against all presently known attacks [5], [7].

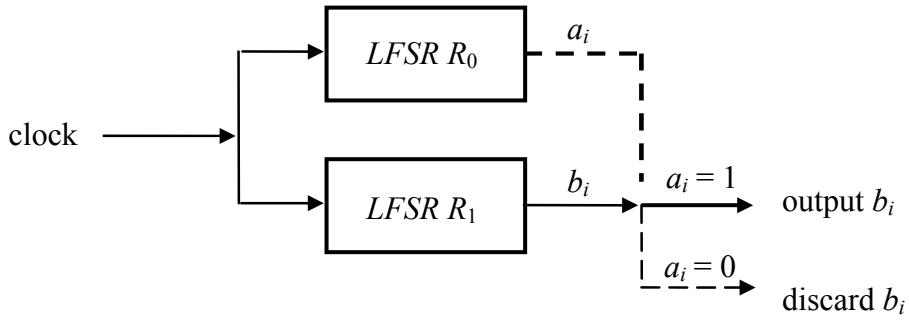

Fig. 2*: *Shrinking generator

SECTION 3

N-adic Summation-Shrinking Generator Architecture

In this section the basic architecture of new *N*-adic Summation-Shrinking Generator (*NSumG*) and some basic *NSumG* properties will be present.

The *NSumG* architecture, proposed recently in [12], uses an increased number of slaved registers in comparison with Shrinking Generator as in the Shrinking-Multiplexing Generator [11]. The control and slave registers in shrinking-multiplexing generator are replaced with *N*-adic and 2-adic summation generators in the *NSumG* (fig. 3) respectively. The using of *N*-adic control summation generator enhances the number of the used 2-adic slave summation generators from 1 in shrinking generator to $N-1$ in *NSumG*.

Every summation generator consists of two *FCSRs*, depicted as $R_{j1} \div R_{j2}$, $(j = 0, 1, ..., N-1)$. It ought to be underlined that slave *FCSRs* $R_{j1} \div R_{j2}$ $(j = 1, 2, ...N-1)$ are *2–FCSRs* and hence, the corresponding adders m_j consist one bit for m_j and one bit for sign. The control *FCSRs* R_{01} and R_{02} are *N-FCSRs* and their adder m_0 have $ind_N(m_j^0)+1$ bits for $|m_0|$ and an extra bit for sign.

As shown, a summation generator selects a portion of the output sequences of several summation generators.

Definition 1. The algorithm of the *N*-adic Summation-Shrinking Generator consists of the following steps:

(1) All *FCSRs* from $R_{01} \div R_{02}$ to $R_{N\text{-}1\,1} \div R_{N\text{-}1\,2}$ are clocked with clock sequence with period τ_0.

(2) If the *N*-adic output $b_i = j$ of the control summation generator is not equal to 0, the output bit of j^{th} slave summation generator forms a part of the keystream. Otherwise, if the output $b_i = 0$ of the control summation generator is equal to 0, the all output bits of slaved summation generators are discarded (fig. 3).

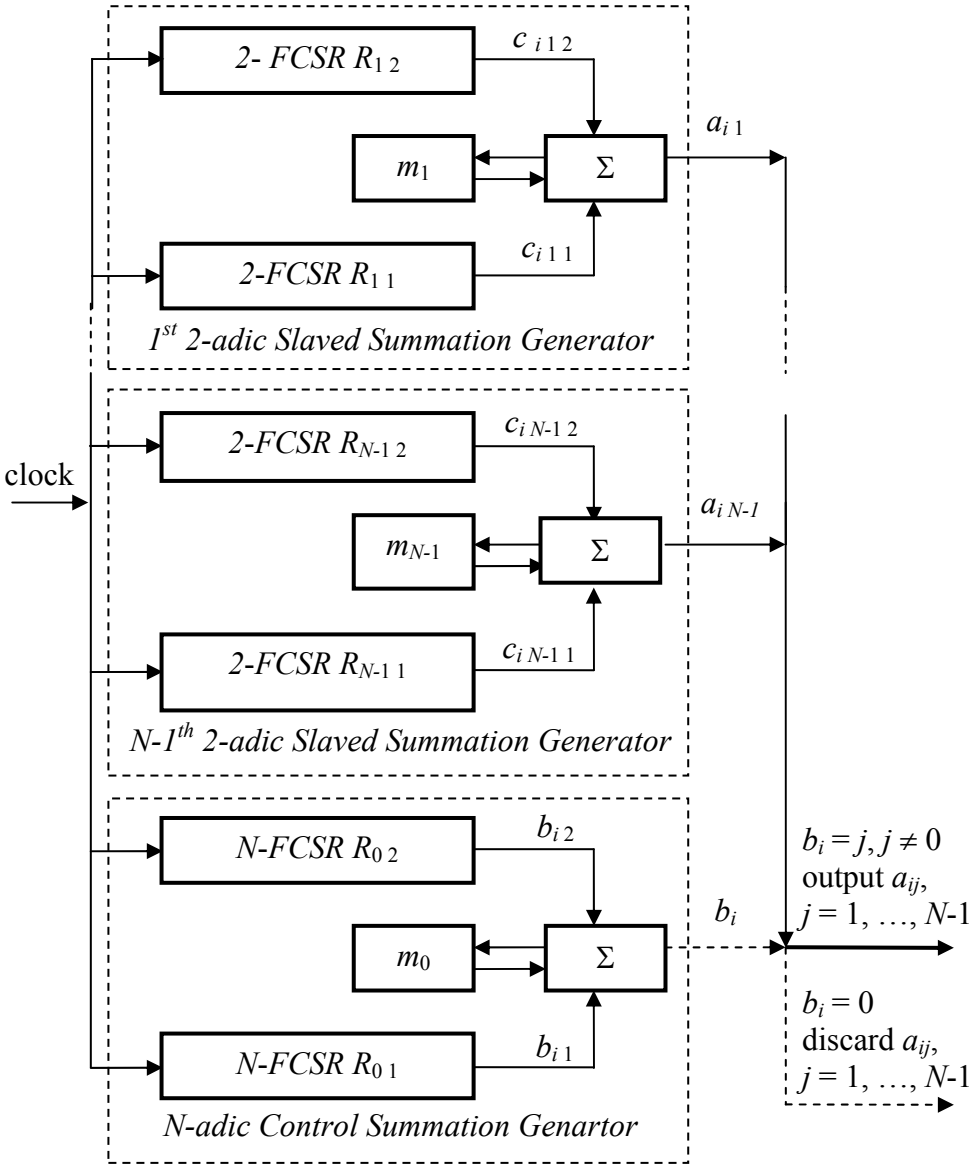

Fig. 3: *N-adic Summation-Shrinking Generator*

Therefore, the produced keystream is a *shrunken and mixed* version of the output sequences a_{ij}, $i = 1, 2, ..., N-1$ of the N-1 slaved summation generators.

It is straightforward that the N-adic Summation-Shrinking Generator succeeds all positive features of the summation generator, shrinking generator and N-adic *FCSR*.

The proposed new pseudo random number generator architecture takes advantages of feedback with carry shift registers over $Z/(N)$ for any integer $N > 1$ (N-FCSRs) (see fig. 4).

Definition 2 [13]. Let $N > 1$ be an integer and $S = \{a : 0 \le a \le N-1\}$. For any integer $r \ge 1$, the state of a feedback with carry shift register over $Z/(N)$ consist of r integers $a_0, a_1, \ldots, a_{r-1} \in S$ and arbitrary integer $M = M_{r-1}$, the memory. The state change function is determined by $r+1$ integers $g, d_1, d_2, \ldots, d_r \in S$, such that $gcd(g, N) = 1$ and $d_r \ne 0$ as follows (fig. 4):

(1) Compute the integer sum $\sigma = M_{r-1} + a_{r-1}d_1 + a_{r-2}d_2 + \ldots + a_0 d_r$;

(2) Compute $a_r \in S$, $M_r \in Z$ such that $\sigma = g a_r + M_r N$;

(3) Change the memory M_{r-1} to M_r;

(4) Output the cell a_0 and use the cell a_r to shift the register loading cells, replacing $(a_{r-1}, \ldots, a_0)$ by $(a_r, \ldots, a_1)$.

For $n \ge r$, a_n is defined by both the memory and the running register cells. In the entire operating $(g, d_1, d_2, \ldots, d_r)$ are fixed. The following integer $d = -g + d_1 N + d_2 N^2 + \ldots + d_r N^r$ is called the connection number. Consequently, $d_0 = -g$ and $d = \sum_{i=0}^{r} d_i N^i$.

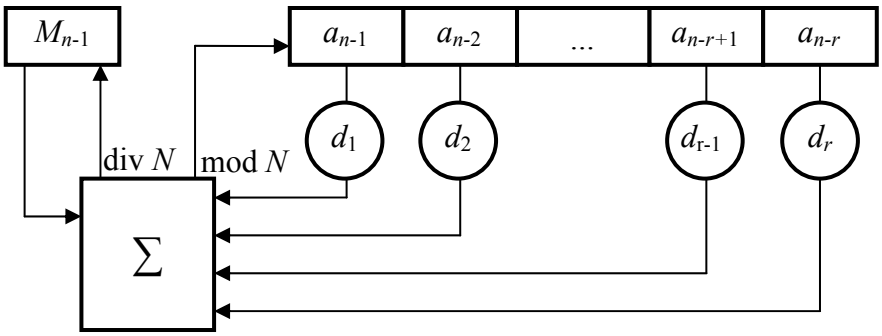

Fig. 4: *N-adic Feedback with Carry Shift Register*

For maximum security one must choose the triples of integers (d, p, N) satisfying the next conditions:

(1) d is prime;

(2) $d = 2p + 1$ and p is odd prime;

(3) N is prime;

(4) N is primitive modulo d and primitive modulo p.

In particular case when $N = 2$ the *2SumSG* consists of only one slave *2*-adic summation generator. Let the connection integers of two *2-FCSRs* $R_{01} \div R_{02}$ of control summation generator be d_{01} and d_{02}. Let the slave summation generator combines two *2-FCSRs* $R_{11} \div R_{12}$ with connection numbers d_{11} and d_{12}. The period of control summation generator is

$$T_0 = \frac{(d_{01} - 1)(d_{02} - 1)}{\gcd((d_{01} - 1), (d_{02} - 1))} = \mathrm{lcm}((d_{01} - 1), (d_{02} - 1)) \tag{4}$$

and the period of slave summation generator is

$$T_1 = \frac{(d_{11} - 1)(d_{12} - 1)}{\gcd((d_{11} - 1), (d_{12} - 1))} = \mathrm{lcm}((d_{11} - 1), (d_{12} - 1)), \tag{5}$$

according to the [6] and the using of triples (d, p, N) with properties mentioned above.

Then the period S_2 of the *2*–adic Summation-Shrinking Generator is:

$$S_2 = \frac{T_0^* T_1}{\gcd(T_0, T_1)}. \tag{6}$$

Here the T_0^* denotes the total number of ones of the control summation generator.

According to [6] the linear complexities L_0 and L_1 of the summation generators are close to their periods, i.e. $L_0 = \frac{(d_{01} - 1)(d_{02} - 1)}{\gcd((d_{01} - 1)(d_{02} - 1))}$, $L_1 = \frac{(d_{11} - 1)(d_{12} - 1)}{\gcd((d_{11} - 1)(d_{12} - 1))}$.

Then from [1] the linear complexity L of the *2SumSG* is at most

$$L = T_1 . T_0^*. \tag{7}$$

As one can see from equation (4)÷(7), the proposed new architecture of pseudorandom number generator even with $N = 2$ allows to produce *PRSs* with period and linear complexity larger than the respective parameters of the *PRSs* formed by a classic shrinking generator [1].

SECTION 4

Implementation and output files generation

The *N*SumSG is software implemented in Visual C++ 6.0 environment for Windows/32 bits. There are used the class *p_adic* to produce the output *N*SumSG sequence. The application and *N*SumSG statistical tests were executed on PC AMD Athlon™ XP 2200+ / 256 MB RAM.

Two different setups are applied to generate 1 000 sequences by 1 000 000 bits each to test the *N*-adic Summation-Shrinking Generator:

(1) $N = 2$. Thereby the *N*SumSG consists of one controlling *2*-adic summation generator with connection integers $d_{01} = 10\,000\,139$ and $d_{02} = 10\,000\,189$. The slave *2*-adic summation generator has first connection number $d_{11} = 10\,000\,229$. The second connection number d_{12} is in every 1 000 000 bits, taking consequently 1 000 values, which are strong *2*-primes [9] in the range [81 467, 2 283 803]. So the seed of constructed *N*SumSG is different at every 1 000 000 bits. The size of generated *N*SumSG output file is 983 Mbytes.

(2) $N = 3$. In this configuration the controlling *3*-adic summation generator gets two connection numbers $d_{01} = 5\,000\,011$ and $d_{02} = 5\,000\,201$. The first slave summation *2*-adic generator has a seed comprising the numbers $d_{11} = 10\,000\,139$ and $d_{12} = 10\,000\,189$. The second summation generator has the first connection number $d_{21} = 10000229$. The second connection number d_{22} is changed in every 1 000 000 bits, taking consequently 1 000 values, which are strong *2*-primes in the range [981 467, 2 283 803]. In this way were generated 1 000 sequences by 1 000 000 bits, in which the seed were changed at every 1 000 000 bits. The size of generated *N*SumSG output file is 983 Mbytes.

The connection *FCSR* numbers were chosen randomly in the two above mention setups.

SECTION 5

Statistical analysis and interpretation of empirical results

To test the randomness of binary sequences generated by *N*SumSG the so-named NIST suite, proposed by National Institute of Standards and Technology, is used. The NIST suite [7], [10] includes sixteen tests. The tests fix on a variety of different types of non-randomness that could exist in a sequence. These tests are: frequency (monobit), frequency within a block, runs, longest-run-of-ones in a block, binary matrix rank, discrete Fourier transform (spectral), non-overlapping template matching, overlapping template matching, Maurer's "Universal statistical", Lempel-Ziv compression, linear complexity, serial, approximate entropy, cumulative sums, random excursions, random excursions variant.

The testing process consists of the following steps [7], [10]:

(1) State the null hypothesis. Assume that the binary sequence is random.

(2) Compute a sequence test statistic. Testing is carried out at the bit level.

(3) Compute the p-value, $\mathrm{p-value} \in [0, 1]$.

(4) Compare the $\mathrm{p-value}$ to α. Fix α, where $\alpha \in (0.0001, 0.01]$. *Success* is declared whenever $\mathrm{p-value} \geq \alpha$; otherwise, *failure* is declared.

Given the empirical results for a particular statistical test, the NIST suite computes the proportion of sequences that pass. The range of acceptable proportion is determined using the confidence interval defined as, $\hat{p} \pm 3\sqrt{\frac{\hat{p}(1-\hat{p})}{m}}$, where $\hat{p} = 1 - \alpha$, and m is the number of binary tested sequences. In our two setups $m = 1000$. Thus the confidence interval is $0.99 \pm 3\sqrt{\frac{0.99(0.01)}{1000}} = 0.99 \pm 0.0094392$. The proportion should lie above 0.9805607.

The distribution of p-values is examined to ensure uniformity. The interval between 0 and 1 is divided into 10 sub-intervals, and the p-values that lie within each sub-interval are counted. Uniformity may also be specified trough an application of a χ^2 test and the determination of a p-value corresponding to the Goodness-of-Fit Distributional Test on the p-values obtained for an arbitrary statistical test, p-value of the p-values. This is implemented by computing $\chi^2 = \sum_{i=1}^{10} \frac{(F_i - m/10)^2}{m/10}$, where F_i is the number of p-values in sub-interval i, and m is the number of tested sequences. A p-value is calculated such that $\mathrm{p\text{-}value}_T = igamc(9/2, \chi^2/2)$. If $\mathrm{p\text{-}value}_T > 0.0001$, then the sequences can be regarded to be uniformly distributed.

Table 1 lists the results from the NIST test suite with input file from the first setup ($N = 2$). The detailed result of Non-overlapping template matching test, Random excursion test and Random excursion – variant test and the numbers of the p-values in the subintervals, when $N = 2$, can be found in Appendix 1.

Table 1: *The results from NSumSG statistical tests, when N = 2*

Statistical Test	**Result**	**Proportion**	**P-value$_T$**	**Comment**
Frequency (monobit)	Pass	0.9920	0.260930	
Frequency within a block	Pass	0.9810	0.896345	
Cumulative sums	Pass	0.9870	0.524101	
	Pass	0.9910	0.832561	
Runs	Pass	0.9830	0.326749	
Longest-run-of-ones in a block	Pass	0.9850	0.465415	
Binary matrix rank	Pass	0.9890	0.757790	
Discrete Fourier transform (spectral)	Pass	0.9970	0.186566	
Non-overlapping template matching	Pass	0.9894	0.531028	Avg. values

Statistical Test	Result	Proportion	P-value$_T$	Comment
Overlapping template matching	Pass	0.9940	0.618385	
Maurer's "Universal statistical"	Pass	0.9880	0.086634	
Approximate entropy	Pass	0.9890	0.476911	
Random excursions	Pass	0.9870	0.598233	Avg. values
Random excursions variant	Pass	0.9901	0.431378	Avg. values
Serial	Pass	0.9930	0.227180	
	Pass	0.9910	0.849708	
Lempel-Ziv compression	Pass	0.9960	0.037320	
Linear complexity	Pass	0.9960	0.355364	
The minimum pass rate for the Random Excursion - (variant) test is approximately 0.977854.				
The minimum pass rate for each statistical test with the exception of the Random Excursion - variant test is approximately = 0.980561.				

The Table 2 lists the results from the NIST test suite with input file from the second setup ($N = 3$). The detailed result of Non-overlapping template matching test, Random excursion test and Random excursion – variant test and the numbers of the p-values in the subintervals, when $N = 3$, can be found in Appendix 2.

Table 2: *The results from NSumSG statistical tests, when N = 3*

Statistical Test	Result	Proportion	P-value$_T$	Comment
Frequency (monobit)	Pass	0.9890	0.881662	
Frequency within a block	Pass	0.9880	0.254411	
Cumulative sums	Pass	0.9820	0.534146	
	Pass	0.9850	0.827279	
Runs	Pass	0.9930	0.428095	
Longest-run-of-ones in a block	Pass	0.9870	0.187581	
Binary matrix rank	Pass	0.9860	0.618385	
Discrete Fourier transform (spectral)	Pass	0.9910	0.647530	
Non-overlapping template matching	Pass	0.9899	0.476221	Avg. values
Overlapping template matching	Pass	0.9900	0.045088	
Maurer's "Universal statistical"	Pass	0.9850	0.662091	
Approximate entropy	Pass	0.9950	0.508172	
Random excursions	Pass	0.9907	0.476154	Avg. values
Random excursions variant	Pass	0.9895	0.461205	Avg. values
Serial	Pass	0.9880	0.672470	
	Pass	0.9940	0.159910	
Lempel-Ziv compression	Pass	0.9820	0.532132	
Linear complexity	Pass	0.9900	0.869278	
The minimum pass rate for the Random Excursion - (variant) test is approximately 0.978117.				
The minimum pass rate for each statistical test with the exception of the Random Excursion - variant test is approximately = 0.980561.				

CONCLUSIONS AND FUTURE WORKS

The results from statistical analysis show that the sequence generated by *NSumSG* is uniform, scalable, uncompressible, whit large period; consistent and unpredictable. This gives the reason to consider that the *NSumSG* as a very interesting pseudorandom generator and it can be useful as a part of stream ciphers.

We will be glad to thanks everyone who helps us to make some strong cryptanalysis of *NSumSG.*

References:

[1] D. Coppersmith, H. Krawczyk, Y. Mansour, "The Shrinking Generator", *Proceedings of Crypto 93*, Springer-Verlag, 1994., pp. 22-39

[2] A. Klapper, M. Goresky, "*2*-adic Shift Register. Fast Software Encryption", *Second International Workshop*. (*Lecture Notes in Computer Science*, vol. 950, Springer Verlag, N. Y., 1994.) pp.174-178

[3] A. Klapper, J. Xu, "Algebraic Feedback Shift Registers" (submitted to *Elsevier Preprint*), 2003.

[4] R. Lidl, H. Niederreiter, "*Finite Fields*", Addison – Wesley Publishing Company, London, England, 1983.

[5] P. van Oorshot, A. Menezes, S. Vanstone, "*Handbook of Applied Cryptography*", CRC Press, 1997.

[6] R. Rueppel, "*Analysis and Design of Stream Siphers*", Springer Verlag, N. Y., 1986.

[7] A. Rukhin, J. Soto, J. Nechvatal, M. Smid, E. Barker, S. Leigh, M. Levenson, M. Vangel, D. Banks, A. Heckert, J. Dray, S. Vo, "A Statistical Test Suite for Random and Pseudo-Random Number Generators for Cryptographic Application", *NIST Special Publication 800-22* (with revision May 15, 2001) http://csrc.nist.gov/rng/.

[8] B. Schneier, "*Applied Cryptography*", John Wiley & Sons, New York, 1996.

[9] Ch. Seo, S. Lee, Y. Sung, K. Han, S. Kim, "A Lower Bound on the Linear Span an FCSR", *IEEE Transaction on Information Theory*, Vol. 46, No 2, March 2000.

[10] J. Soto, "Statistical Testing of Random Number Generators", http://csrc.nist.gov/rng/.

[11] Zh. N. Tasheva, B. Y. Bedzhev, V. A. Mutkov, "An Shrinking Data Encryption Algorithm with *p*-adic Feedback with Carry Shift Register", *XII International Symposium of Theoretical Electrical Engineering ISTET 03*, Warsaw, Poland, 6-9 July, 2003., Conference Proceedings, vol.II, pp. 397–400.

[12] Zh. N. Tasheva, B. Y. Bedzhev, B. P. Stoyanov, "Summation-Shrinking Generator", *Conference Proceeding of International Conference "Information Technology and Sequrity ITS – 2004"*, June 22-26, 2004, Partenit, Crimea, Ukraine, pp.119-127.

[13] Xu, J., "Stream Cipher Analysis Based on FCSRs", *PhD Dissertation*, University of Kentucky, 2000.

APPENDIX 1

Results from setup 1

The Uniformity of *p*-values and the Proportion of passing sequences

C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	p-values$_T$	Proportion	Test
113	117	91	111	85	97	86	100	96	104	0.260930	0.9920	frequency
91	98	112	101	111	96	101	98	93	99	0.896345	0.9810	block-frequency
114	105	96	91	82	95	107	97	106	107	0.524101	0.9870	cumulative-sums
106	104	109	97	88	92	96	94	108	106	0.832561	0.9910	cumulative-sums
122	90	108	104	99	108	86	92	96	95	0.326749	0.9830	runs
108	95	94	96	118	94	84	110	103	98	0.465415	0.9850	longest-run
109	106	97	95	99	86	102	114	95	97	0.757790	0.9890	rank
94	107	109	109	121	100	93	99	84	84	0.186566	0.9970	fft
102	92	99	113	83	92	90	115	121	93	0.120207	0.9900	nonperiodic-templates
108	118	90	95	96	104	95	111	96	87	0.459717	0.9860	nonperiodic-templates
99	95	87	101	106	106	96	101	90	119	0.589341	0.9910	nonperiodic-templates
106	112	101	96	108	107	101	81	85	103	0.431754	0.9940	nonperiodic-templates
104	86	98	101	102	104	120	67	111	107	0.025535	0.9840	nonperiodic-templates
97	117	106	93	79	99	92	109	92	116	0.167184	0.9900	nonperiodic-templates
90	92	121	96	121	120	87	85	87	101	0.016374	0.9910	nonperiodic-templates
108	133	91	92	89	94	112	101	86	94	0.031637	0.9810	nonperiodic-templates
83	109	122	99	95	91	101	98	98	104	0.361938	0.9910	nonperiodic-templates
89	109	108	93	100	106	105	104	104	82	0.603841	0.9890	nonperiodic-templates
122	91	92	111	89	99	98	106	103	89	0.317565	0.9840	nonperiodic-templates
108	105	83	97	120	88	101	94	107	97	0.329850	0.9860	nonperiodic-templates
89	116	101	95	105	93	97	99	90	115	0.522100	0.9930	nonperiodic-templates
94	90	113	91	93	109	112	101	100	97	0.668321	0.9860	nonperiodic-templates
90	94	93	115	101	108	103	100	100	96	0.834308	0.9920	nonperiodic-templates
85	99	106	106	100	98	116	95	111	84	0.383827	0.9910	nonperiodic-templates
97	101	103	111	106	81	96	101	112	92	0.572847	0.9910	nonperiodic-templates
107	89	94	95	113	103	103	94	103	99	0.864494	0.9900	nonperiodic-templates
103	111	101	96	95	98	78	99	102	117	0.388990	0.9940	nonperiodic-templates
99	89	100	106	99	90	106	100	103	108	0.931185	0.9870	nonperiodic-templates
99	99	98	84	102	101	104	105	104	104	0.946308	0.9900	nonperiodic-templates
105	98	97	111	107	97	82	109	90	104	0.597620	0.9860	nonperiodic-templates
98	91	79	88	111	102	107	117	102	105	0.235589	0.9910	nonperiodic-templates
94	107	115	94	98	109	105	105	86	87	0.488534	0.9910	nonperiodic-templates
102	93	99	114	98	98	108	96	91	101	0.897763	0.9910	nonperiodic-templates
106	113	92	101	95	111	112	89	93	88	0.461612	0.9930	nonperiodic-templates

C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	p-values$_T$	Proportion	Test
99	115	80	92	104	125	102	94	87	102	0.079538	0.9910	nonperiodic-templates
93	113	89	108	115	90	88	106	99	99	0.428095	0.9920	nonperiodic-templates
92	104	98	105	91	93	123	109	92	93	0.382115	0.9890	nonperiodic-templates
97	112	102	101	113	90	99	107	81	98	0.492436	0.9910	nonperiodic-templates
94	87	113	107	97	109	96	98	96	103	0.781106	0.9890	nonperiodic-templates
104	98	86	99	94	105	107	92	116	99	0.691081	0.9910	nonperiodic-templates
104	105	98	91	99	90	111	90	96	116	0.616305	0.9890	nonperiodic-templates
102	106	105	95	94	94	107	106	94	97	0.967382	0.9930	nonperiodic-templates
103	100	91	103	92	100	96	112	105	98	0.940080	0.9830	nonperiodic-templates
104	107	107	93	98	93	98	105	79	116	0.399442	0.9910	nonperiodic-templates
104	87	97	107	98	111	110	84	95	107	0.536163	0.9960	nonperiodic-templates
104	107	106	83	100	102	101	104	91	102	0.837781	0.9900	nonperiodic-templates
96	96	123	84	89	97	106	101	106	102	0.331408	0.9950	nonperiodic-templates
100	115	98	97	96	103	84	100	107	100	0.771469	0.9830	nonperiodic-templates
110	99	106	117	85	93	102	112	88	88	0.251837	0.9920	nonperiodic-templates
91	103	101	94	96	103	105	103	93	111	0.937919	0.9870	nonperiodic-templates
103	74	94	108	99	102	96	99	116	109	0.246750	0.9920	nonperiodic-templates
101	102	86	100	108	100	112	108	94	89	0.709558	0.9850	nonperiodic-templates
105	97	98	100	97	122	95	91	92	103	0.626709	0.9880	nonperiodic-templates
92	110	103	99	95	102	102	95	98	104	0.980341	0.9920	nonperiodic-templates
103	100	102	100	90	115	95	90	107	98	0.820143	0.9860	nonperiodic-templates
104	111	93	104	82	81	118	90	108	109	0.103753	0.9810	nonperiodic-templates
105	116	85	89	96	96	106	100	95	112	0.471146	0.9840	nonperiodic-templates
91	104	107	95	96	90	106	108	107	96	0.873987	0.9830	nonperiodic-templates
110	100	106	104	107	96	99	98	103	77	0.574903	0.9890	nonperiodic-templates
125	91	107	94	101	111	90	91	100	90	0.216713	0.9890	nonperiodic-templates
96	93	112	94	97	109	91	101	95	112	0.753844	0.9920	nonperiodic-templates
102	100	95	107	106	104	99	106	84	97	0.889118	0.9870	nonperiodic-templates
104	98	119	103	99	94	85	90	100	108	0.518106	0.9930	nonperiodic-templates
95	84	113	95	91	101	113	98	108	102	0.536163	0.9890	nonperiodic-templates
106	106	90	89	113	105	98	92	98	103	0.771469	0.9880	nonperiodic-templates
97	101	99	95	110	90	95	93	123	97	0.486588	0.9900	nonperiodic-templates
101	91	100	99	97	104	90	113	113	92	0.729870	0.9910	nonperiodic-templates
110	97	101	79	104	105	100	115	76	113	0.075254	0.9860	nonperiodic-templates
107	86	105	115	91	97	89	107	98	105	0.550347	0.9850	nonperiodic-templates
88	111	102	100	94	96	100	102	114	93	0.769527	0.9950	nonperiodic-templates
105	111	98	94	94	96	99	89	108	106	0.867692	0.9870	nonperiodic-templates
86	117	99	113	100	96	120	94	91	84	0.107512	0.9900	nonperiodic-templates
105	107	100	112	98	92	95	107	94	90	0.837781	0.9920	nonperiodic-templates
79	111	97	104	98	100	113	105	89	104	0.417219	0.9930	nonperiodic-templates
86	81	112	104	115	104	106	85	111	96	0.138860	0.9920	nonperiodic-templates
92	91	107	95	114	100	101	114	89	97	0.593478	0.9900	nonperiodic-templates
111	99	99	107	95	97	95	115	97	85	0.647530	0.9900	nonperiodic-templates
90	117	83	115	96	96	100	107	92	104	0.301194	0.9910	nonperiodic-templates
100	94	102	105	96	108	92	103	91	109	0.924076	0.9880	nonperiodic-templates
94	121	88	100	105	82	99	108	110	93	0.222480	0.9950	nonperiodic-templates
116	93	120	105	91	94	116	79	99	87	0.046568	0.9890	nonperiodic-templates
106	85	89	100	93	116	102	115	100	94	0.390721	0.9870	nonperiodic-templates
102	92	99	114	82	93	89	117	119	93	0.103138	0.9900	nonperiodic-templates
103	95	101	127	102	84	100	89	92	107	0.182550	0.9900	nonperiodic-templates

C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	p-values$_T$	Proportion	Test
100	105	91	125	99	107	83	95	101	94	0.254411	0.9900	nonperiodic-templates
75	108	120	104	118	86	113	94	88	94	0.017305	0.9900	nonperiodic-templates
106	117	91	88	111	102	88	109	96	92	0.366918	0.9900	nonperiodic-templates
101	97	105	93	86	100	117	99	103	99	0.759756	0.9920	nonperiodic-templates
92	78	101	97	113	113	105	115	87	99	0.155499	0.9950	nonperiodic-templates
107	98	93	103	110	89	96	115	86	103	0.556460	0.9930	nonperiodic-templates
95	110	99	98	104	101	100	103	88	102	0.962688	0.9910	nonperiodic-templates
92	96	90	96	115	109	99	99	103	101	0.821937	0.9880	nonperiodic-templates
99	105	98	106	88	101	107	96	98	102	0.970302	0.9910	nonperiodic-templates
98	87	96	103	115	103	100	96	104	98	0.861264	0.9860	nonperiodic-templates
92	109	95	100	107	115	91	118	89	84	0.199045	0.9930	nonperiodic-templates
110	105	104	101	108	96	95	92	100	89	0.889118	0.9900	nonperiodic-templates
107	127	93	73	99	100	120	81	97	103	0.005436	0.9840	nonperiodic-templates
88	103	116	100	82	105	104	76	118	108	0.043087	0.9970	nonperiodic-templates
100	108	85	104	93	88	99	101	115	107	0.560545	0.9890	nonperiodic-templates
97	105	111	88	111	84	97	104	106	97	0.589341	0.9890	nonperiodic-templates
122	99	87	106	106	103	78	104	101	94	0.175691	0.9850	nonperiodic-templates
101	94	90	102	119	96	88	98	114	98	0.469232	0.9910	nonperiodic-templates
96	94	102	91	102	96	95	108	106	110	0.922855	0.9890	nonperiodic-templates
102	98	88	101	101	102	88	98	112	110	0.788728	0.9920	nonperiodic-templates
107	107	110	91	106	81	99	92	102	105	0.564639	0.9880	nonperiodic-templates
105	102	111	97	98	96	92	101	100	98	0.981417	0.9910	nonperiodic-templates
93	89	103	97	102	119	92	103	105	97	0.678686	0.9890	nonperiodic-templates
107	96	113	87	110	77	90	115	103	102	0.133404	0.9910	nonperiodic-templates
96	107	98	128	88	110	106	95	93	79	0.057510	0.9930	nonperiodic-templates
91	99	89	105	124	99	98	98	95	102	0.492436	0.9900	nonperiodic-templates
106	96	115	95	76	98	101	107	111	95	0.291091	0.9940	nonperiodic-templates
98	90	106	96	127	97	93	108	89	96	0.246750	0.9940	nonperiodic-templates
99	104	106	92	103	116	83	90	99	108	0.498313	0.9930	nonperiodic-templates
112	98	95	108	82	102	100	103	95	105	0.715679	0.9800	nonperiodic-templates
93	104	90	98	103	101	112	83	104	112	0.562591	0.9840	nonperiodic-templates
109	89	116	99	104	94	105	89	88	107	0.484646	0.9920	nonperiodic-templates
96	107	99	100	99	99	115	97	103	85	0.801865	0.9930	nonperiodic-templates
110	102	99	97	99	79	126	87	100	101	0.121616	0.9910	nonperiodic-templates
101	88	110	88	111	106	96	104	107	89	0.587274	0.9930	nonperiodic-templates
87	105	119	84	100	103	110	91	97	104	0.329850	0.9890	nonperiodic-templates
109	109	89	111	99	93	96	95	89	110	0.620465	0.9880	nonperiodic-templates
104	104	88	113	96	104	98	97	93	103	0.877083	0.9960	nonperiodic-templates
104	93	104	91	110	109	94	104	100	91	0.854708	0.9910	nonperiodic-templates
110	110	99	92	105	94	105	87	104	94	0.767582	0.9890	nonperiodic-templates
100	93	94	98	97	107	112	94	117	88	0.574903	0.9870	nonperiodic-templates
93	103	98	94	92	102	117	100	100	101	0.870856	0.9880	nonperiodic-templates
109	94	99	106	100	108	101	105	93	85	0.818343	0.9950	nonperiodic-templates
89	114	85	118	78	112	106	92	90	116	0.021262	0.9920	nonperiodic-templates
95	93	125	116	97	94	96	97	75	112	0.035876	0.9920	nonperiodic-templates
99	98	91	89	97	99	105	110	104	108	0.896345	0.9930	nonperiodic-templates
94	105	90	98	111	102	105	99	90	106	0.873987	0.9910	nonperiodic-templates
94	101	112	113	89	107	87	82	100	115	0.192724	0.9900	nonperiodic-templates
90	87	98	109	105	113	93	108	91	106	0.556460	0.9920	nonperiodic-templates
89	91	115	102	109	94	105	100	102	93	0.713641	0.9910	nonperiodic-templates

C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	p-values$_T$	Proportion	Test
106	100	87	93	121	108	107	91	101	86	0.285427	0.9860	nonperiodic-templates
89	101	108	94	91	103	103	111	108	92	0.769527	0.9910	nonperiodic-templates
84	101	115	97	100	108	96	95	99	105	0.717714	0.9910	nonperiodic-templates
97	86	99	100	94	97	110	106	107	104	0.873987	0.9920	nonperiodic-templates
119	85	84	88	103	100	108	99	120	94	0.092041	0.9830	nonperiodic-templates
101	114	97	95	94	94	93	102	102	108	0.894918	0.9800	nonperiodic-templates
115	95	102	101	104	103	99	107	84	90	0.651693	0.9820	nonperiodic-templates
117	94	106	112	90	92	86	106	98	99	0.431754	0.9890	nonperiodic-templates
104	92	96	102	93	98	91	96	108	120	0.622546	0.9870	nonperiodic-templates
102	83	115	83	109	107	117	93	102	89	0.122325	0.9850	nonperiodic-templates
116	100	91	99	81	111	103	83	110	106	0.184549	0.9860	nonperiodic-templates
97	108	104	100	94	103	90	95	103	106	0.962688	0.9860	nonperiodic-templates
80	107	96	109	104	98	100	103	113	90	0.490483	0.9930	nonperiodic-templates
102	104	99	101	95	91	103	106	110	89	0.915317	0.9900	nonperiodic-templates
117	94	96	113	105	90	103	97	82	103	0.361938	0.9810	nonperiodic-templates
108	108	103	94	99	102	101	97	84	104	0.867692	0.9900	nonperiodic-templates
90	93	110	95	95	111	108	112	88	98	0.558502	0.9950	nonperiodic-templates
111	102	85	88	94	94	94	110	111	111	0.397688	0.9880	nonperiodic-templates
102	113	109	117	93	97	93	92	88	96	0.461612	0.9810	nonperiodic-templates
106	95	101	103	106	96	104	83	99	107	0.853049	0.9890	nonperiodic-templates
101	87	108	112	102	97	104	97	92	100	0.851383	0.9860	nonperiodic-templates
106	85	89	99	94	114	104	114	102	93	0.455937	0.9870	nonperiodic-templates
103	103	101	111	90	101	112	100	82	97	0.618385	0.9940	overlapping-templates
103	99	87	115	115	109	77	86	109	100	0.086634	0.9880	universal
108	101	105	106	102	78	99	89	101	111	0.476911	0.9890	apen
53	65	64	62	64	46	57	69	62	62	0.664861	0.9868	random-excursions
61	62	55	71	52	57	61	62	57	66	0.870331	0.9834	random-excursions
67	75	66	62	60	61	58	43	52	60	0.262249	0.9834	random-excursions
64	66	56	61	57	63	67	46	60	64	0.759756	0.9868	random-excursions
72	54	68	58	53	68	56	55	67	53	0.500934	0.9851	random-excursions
58	58	68	63	45	64	60	65	69	54	0.561227	0.9917	random-excursions
61	63	64	73	65	55	50	71	58	44	0.207730	0.9868	random-excursions
62	60	61	59	65	67	60	62	50	58	0.958773	0.9917	random-excursions
44	50	64	58	58	48	72	75	63	72	0.042347	0.9851	random-excursions-variant
49	56	55	59	57	68	51	60	73	76	0.193419	0.9901	random-excursions-variant
47	55	61	61	61	55	70	61	61	72	0.554420	0.9884	random-excursions-variant
53	52	50	61	61	61	66	67	72	61	0.592098	0.9917	random-excursions-variant
59	41	62	56	52	65	65	66	66	72	0.220931	0.9884	random-excursions-variant
60	53	59	43	70	59	55	61	63	81	0.083867	0.9934	random-excursions-variant
62	57	53	59	68	56	62	63	65	59	0.962959	0.9901	random-excursions-variant
63	56	66	56	65	62	54	72	58	52	0.749884	0.9851	random-excursions-variant
62	64	60	48	61	63	48	62	70	66	0.571477	0.9785	random-excursions-variant
57	50	63	54	68	80	51	65	53	63	0.161104	0.9884	random-excursions-variant
49	58	60	63	66	63	68	56	61	60	0.888137	0.9950	random-excursions-variant
48	64	61	71	44	69	50	64	70	63	0.119973	0.9934	random-excursions-variant
51	70	50	73	66	47	56	61	68	62	0.195163	0.9917	random-excursions-variant
49	63	76	62	66	41	60	57	62	68	0.108791	0.9983	random-excursions-variant
47	64	70	62	56	65	56	65	60	59	0.719747	0.9934	random-excursions-variant
54	48	74	72	57	68	58	49	61	63	0.205897	0.9884	random-excursions-variant
54	61	66	65	60	74	45	56	61	62	0.437274	0.9917	random-excursions-varian

C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	p-values$_T$	Proportion	Test
54	57	65	70	59	61	61	57	58	62	0.957319	0.9901	t random-excursions-variant
106	100	90	103	87	88	97	112	94	123	0.227180	0.9930	serial
89	93	102	114	103	96	95	103	98	107	0.849708	0.9910	serial
99	74	114	121	115	94	98	95	103	87	0.037320	0.9960	lempel-ziv
83	119	107	89	99	106	100	97	92	108	0.355364	0.9960	linear-complexity

APPENDIX 2

Results from setup 2

The Uniformity of p-values and the Proportion of passing sequences

C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	p-values$_T$	Proportion	Test
94	105	98	96	91	91	101	106	109	109	0.881662	0.9890	frequency
90	91	101	100	98	86	120	95	118	101	0.254411	0.9880	block-frequency
101	99	89	96	107	87	91	109	105	116	0.534146	0.9820	cumulative-sums
94	93	101	102	96	108	112	87	104	103	0.827279	0.9850	cumulative-sums
98	87	103	97	90	120	103	108	105	89	0.428095	0.9930	runs
112	111	103	110	87	84	104	82	110	97	0.187581	0.9870	longest-run
101	115	113	102	95	91	92	102	101	88	0.618385	0.9860	rank
111	108	110	88	97	92	91	95	109	99	0.647530	0.9910	fft
105	101	97	98	89	120	91	97	99	103	0.678686	0.9920	nonperiodic-templates
100	107	88	96	95	103	93	93	107	118	0.622546	0.9940	nonperiodic-templates
111	101	103	89	114	78	104	107	93	100	0.314544	0.9910	nonperiodic-templates
95	98	85	115	98	95	124	96	107	87	0.154629	0.9870	nonperiodic-templates
98	95	117	105	93	87	125	111	73	96	0.014961	0.9850	nonperiodic-templates
83	97	114	100	86	95	100	100	113	112	0.328297	0.9930	nonperiodic-templates
106	100	106	100	100	101	95	93	97	102	0.996335	0.9820	nonperiodic-templates
101	105	98	96	91	78	106	113	112	100	0.383827	0.9930	nonperiodic-templates
90	99	98	102	95	118	86	87	101	124	0.122325	0.9920	nonperiodic-templates
110	100	95	113	96	86	100	99	101	100	0.827279	0.9860	nonperiodic-templates
92	88	113	97	93	102	88	103	106	118	0.390721	0.9910	nonperiodic-templates
88	98	111	91	115	86	108	103	91	109	0.345650	0.9930	nonperiodic-templates
99	87	94	130	94	97	92	110	100	97	0.152044	0.9910	nonperiodic-templates
105	98	98	113	100	104	109	99	71	103	0.242986	0.9860	nonperiodic-templates
104	86	98	119	87	108	99	101	98	100	0.518106	0.9940	nonperiodic-templates
89	97	119	93	101	108	100	91	108	94	0.548314	0.9870	nonperiodic-templates
97	100	107	101	98	93	91	98	113	102	0.930026	0.9880	nonperiodic-templates
98	91	107	99	106	121	100	99	88	91	0.496351	0.9880	nonperiodic-templates
109	112	97	112	98	84	78	105	102	103	0.236810	0.9850	nonperiodic-templates
129	78	93	95	96	91	97	107	109	105	0.055361	0.9890	nonperiodic-templates
96	100	95	109	103	106	106	89	99	97	0.949278	0.9880	nonperiodic-templates
96	109	89	117	88	101	84	111	108	97	0.274341	0.9910	nonperiodic-templates
93	122	103	81	100	101	84	109	109	98	0.151190	0.9880	nonperiodic-templates
94	96	97	113	105	94	108	98	100	95	0.921624	0.9880	nonperiodic-templates
99	92	86	102	99	116	103	115	92	96	0.498313	0.9860	nonperiodic-templates

C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	p-values$_T$	Proportion	Test
87	110	107	107	94	99	103	91	97	105	0.809249	0.9890	nonperiodic-templates
91	90	100	115	104	101	106	105	95	93	0.781106	0.9930	nonperiodic-templates
102	100	100	92	109	103	102	93	116	83	0.579021	0.9840	nonperiodic-templates
96	91	97	83	101	97	117	100	103	115	0.411840	0.9950	nonperiodic-templates
100	96	101	93	89	108	117	108	86	102	0.530120	0.9850	nonperiodic-templates
90	87	114	125	102	98	92	97	92	103	0.189625	0.9930	nonperiodic-templates
103	87	93	108	112	83	96	112	93	113	0.274341	0.9880	nonperiodic-templates
94	94	96	109	77	109	106	112	109	94	0.292519	0.9930	nonperiodic-templates
86	98	110	102	102	93	114	92	112	91	0.492436	0.9930	nonperiodic-templates
95	107	103	87	103	97	100	96	119	93	0.641284	0.9880	nonperiodic-templates
94	100	93	103	114	98	102	86	102	108	0.777265	0.9910	nonperiodic-templates
96	109	89	110	103	106	88	91	96	112	0.587274	0.9930	nonperiodic-templates
85	105	109	95	109	97	102	95	97	106	0.816537	0.9980	nonperiodic-templates
96	114	110	119	113	84	92	92	82	98	0.077131	0.9890	nonperiodic-templates
111	98	97	90	89	108	97	99	113	98	0.737915	0.9900	nonperiodic-templates
101	99	100	83	100	93	116	111	113	84	0.248014	0.9910	nonperiodic-templates
99	97	106	94	98	107	101	90	95	113	0.890582	0.9900	nonperiodic-templates
97	96	96	100	99	98	92	115	104	103	0.935716	0.9880	nonperiodic-templates
105	112	97	88	100	95	101	88	106	108	0.747898	0.9900	nonperiodic-templates
81	95	108	119	115	94	99	110	94	85	0.110734	0.9960	nonperiodic-templates
98	94	106	111	86	95	93	94	107	116	0.526105	0.9940	nonperiodic-templates
100	103	102	102	102	107	89	93	99	103	0.980883	0.9920	nonperiodic-templates
113	85	97	106	101	102	114	99	89	94	0.536163	0.9880	nonperiodic-templates
97	97	114	111	96	91	111	89	102	92	0.593478	0.9900	nonperiodic-templates
105	99	86	105	93	109	101	110	88	104	0.701366	0.9880	nonperiodic-templates
94	105	112	96	109	113	93	105	88	85	0.424453	0.9880	nonperiodic-templates
101	89	101	124	98	86	92	99	113	97	0.248014	0.9880	nonperiodic-templates
100	102	90	103	109	121	88	96	104	87	0.366918	0.9840	nonperiodic-templates
92	96	114	107	91	108	101	93	96	102	0.798139	0.9930	nonperiodic-templates
104	103	105	94	92	111	102	95	89	105	0.878618	0.9880	nonperiodic-templates
92	94	112	98	101	100	100	100	97	106	0.966626	0.9940	nonperiodic-templates
92	89	120	129	105	97	82	95	102	89	0.022452	0.9960	nonperiodic-templates
95	89	114	105	88	77	106	108	96	122	0.058984	0.9910	nonperiodic-templates
111	97	104	95	98	97	91	99	102	106	0.961869	0.9840	nonperiodic-templates
93	113	100	93	99	119	105	89	98	91	0.474986	0.9850	nonperiodic-templates
116	95	79	100	96	114	112	91	92	105	0.187581	0.9890	nonperiodic-templates
109	99	85	107	88	96	109	94	108	105	0.614226	0.9860	nonperiodic-templates
90	115	84	91	105	102	102	97	118	96	0.316052	0.9910	nonperiodic-templates
108	92	87	94	99	102	119	100	99	100	0.637119	0.9960	nonperiodic-templates
99	89	113	94	89	106	113	104	96	97	0.643366	0.9930	nonperiodic-templates
101	98	88	116	100	88	111	100	95	103	0.632955	0.9910	nonperiodic-templates
102	86	103	99	103	123	97	105	96	86	0.355364	0.9880	nonperiodic-templates
97	113	97	88	97	125	87	91	99	106	0.185555	0.9930	nonperiodic-templates
96	86	116	97	109	99	86	92	108	111	0.347257	0.9970	nonperiodic-templates
91	113	94	94	106	110	100	106	95	91	0.739918	0.9930	nonperiodic-templates
90	108	117	90	105	106	85	105	113	81	0.124476	0.9930	nonperiodic-templates
110	86	99	86	90	105	104	103	108	109	0.546283	0.9850	nonperiodic-templates
102	91	114	94	108	119	88	90	105	89	0.254411	0.9860	nonperiodic-templates
86	119	89	101	99	116	109	88	102	91	0.188601	0.9910	nonperiodic-templates
105	101	97	97	90	120	91	98	98	103	0.697257	0.9920	nonperiodic-templates

C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	p-values$_T$	Proportion	Test
109	97	107	100	107	86	81	107	101	105	0.514124	0.9910	nonperiodic-templates
93	97	91	92	111	92	109	106	101	108	0.769527	0.9950	nonperiodic-templates
94	95	94	121	92	102	108	99	103	92	0.591409	0.9970	nonperiodic-templates
92	93	102	100	112	105	85	97	106	108	0.719747	0.9910	nonperiodic-templates
121	118	84	100	120	90	102	90	90	85	0.021262	0.9920	nonperiodic-templates
117	96	87	100	110	101	98	87	94	110	0.452173	0.9910	nonperiodic-templates
91	106	103	88	96	101	109	103	99	104	0.915317	0.9910	nonperiodic-templates
84	96	98	98	127	89	101	110	101	96	0.187581	0.9950	nonperiodic-templates
95	91	116	97	99	93	103	89	126	91	0.159020	0.9900	nonperiodic-templates
132	102	83	102	80	100	100	107	104	90	0.026410	0.9820	nonperiodic-templates
107	105	110	83	112	87	99	95	109	93	0.408275	0.9900	nonperiodic-templates
84	90	95	90	110	96	107	83	112	133	0.009950	0.9910	nonperiodic-templates
106	95	110	112	104	94	102	78	101	98	0.484646	0.9900	nonperiodic-templates
107	104	99	99	79	114	105	92	103	98	0.528111	0.9860	nonperiodic-templates
108	116	93	92	103	83	93	100	96	116	0.310049	0.9880	nonperiodic-templates
92	96	83	107	103	107	117	105	90	100	0.446556	0.9970	nonperiodic-templates
90	103	89	111	107	88	98	101	91	122	0.266235	0.9950	nonperiodic-templates
122	102	109	80	104	93	106	82	116	86	0.030197	0.9930	nonperiodic-templates
110	105	95	93	102	103	97	93	100	102	0.973718	0.9900	nonperiodic-templates
95	97	103	97	97	108	88	110	110	95	0.839507	0.9910	nonperiodic-templates
127	95	104	94	99	94	96	103	93	95	0.399442	0.9890	nonperiodic-templates
102	98	89	104	110	83	95	101	119	99	0.417219	0.9850	nonperiodic-templates
96	118	108	111	92	94	103	90	97	91	0.510153	0.9920	nonperiodic-templates
87	90	106	94	107	103	93	94	129	97	0.147815	0.9860	nonperiodic-templates
118	106	110	97	105	94	83	91	95	101	0.413628	0.9810	nonperiodic-templates
102	108	110	95	115	93	95	84	102	96	0.566688	0.9900	nonperiodic-templates
87	116	107	94	103	94	87	107	101	104	0.544254	0.9920	nonperiodic-templates
84	102	100	102	91	115	105	92	106	103	0.632955	0.9960	nonperiodic-templates
98	93	95	114	105	86	110	104	104	91	0.628790	0.9870	nonperiodic-templates
93	110	103	105	92	92	94	93	115	103	0.709558	0.9870	nonperiodic-templates
83	91	118	123	106	82	103	96	91	107	0.043087	0.9900	nonperiodic-templates
107	106	95	95	107	79	96	108	112	95	0.461612	0.9880	nonperiodic-templates
85	99	114	93	108	79	90	118	113	101	0.073417	0.9930	nonperiodic-templates
89	87	100	105	116	103	97	97	112	94	0.556460	0.9900	nonperiodic-templates
102	95	118	102	98	98	102	87	106	92	0.701366	0.9870	nonperiodic-templates
112	96	103	93	117	81	110	87	96	105	0.225998	0.9860	nonperiodic-templates
99	96	87	91	100	95	102	106	103	121	0.552383	0.9880	nonperiodic-templates
92	88	107	88	105	100	100	103	110	107	0.755819	0.9900	nonperiodic-templates
97	115	106	95	118	98	98	87	90	96	0.426272	0.9910	nonperiodic-templates
91	108	90	103	98	91	109	90	114	106	0.583145	0.9910	nonperiodic-templates
103	101	116	104	87	93	106	108	82	100	0.415422	0.9910	nonperiodic-templates
103	105	85	101	104	92	107	81	115	107	0.331408	0.9870	nonperiodic-templates
113	102	107	81	87	93	107	109	105	96	0.373625	0.9860	nonperiodic-templates
110	108	94	108	101	90	97	101	102	89	0.834308	0.9810	nonperiodic-templates
105	109	91	96	108	100	102	90	96	103	0.914025	0.9810	nonperiodic-templates
111	88	106	106	97	97	107	101	103	84	0.668321	0.9900	nonperiodic-templates
92	97	104	97	110	111	97	99	87	106	0.803720	0.9950	nonperiodic-templates
118	86	94	96	103	102	98	110	104	89	0.508172	0.9890	nonperiodic-templates
94	97	94	93	105	113	90	124	81	109	0.108150	0.9960	nonperiodic-templates
89	107	88	109	105	110	110	80	117	85	0.082010	0.9930	nonperiodic-templates

C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	p-values$_T$	Proportion	Test
106	101	91	90	107	107	108	86	95	109	0.655854	0.9890	nonperiodic-templates
87	114	91	101	107	104	107	85	110	94	0.417219	0.9900	nonperiodic-templates
108	89	100	108	95	93	88	104	101	114	0.657933	0.9890	nonperiodic-templates
95	105	101	95	82	107	95	102	117	101	0.566688	0.9960	nonperiodic-templates
100	103	91	97	106	108	99	95	108	93	0.947308	0.9870	nonperiodic-templates
96	110	99	89	96	96	104	98	101	111	0.903338	0.9910	nonperiodic-templates
118	103	100	113	91	91	89	95	91	109	0.373625	0.9850	nonperiodic-templates
99	97	100	108	106	109	98	86	91	106	0.827279	0.9910	nonperiodic-templates
91	105	96	116	99	107	109	87	101	89	0.534146	0.9880	nonperiodic-templates
95	109	93	84	99	124	107	92	87	110	0.133404	0.9900	nonperiodic-templates
80	111	98	123	99	87	88	95	108	111	0.071620	0.9960	nonperiodic-templates
96	96	100	97	111	93	102	102	102	101	0.987079	0.9850	nonperiodic-templates
92	86	82	119	89	86	111	122	116	97	0.011303	0.9900	nonperiodic-templates
101	104	118	88	102	95	111	79	91	111	0.172816	0.9860	nonperiodic-templates
88	120	101	96	97	103	96	102	105	92	0.649612	0.9910	nonperiodic-templates
114	93	112	83	91	108	101	118	87	93	0.134944	0.9870	nonperiodic-templates
92	98	105	105	121	103	92	96	93	95	0.614226	0.9880	nonperiodic-templates
98	80	106	93	105	106	103	83	112	114	0.220159	0.9900	nonperiodic-templates
99	108	103	108	92	95	84	109	91	111	0.568739	0.9880	nonperiodic-templates
91	108	119	94	100	97	93	106	104	88	0.538182	0.9920	nonperiodic-templates
107	72	74	104	106	124	94	97	116	106	0.003273	0.9890	nonperiodic-templates
100	89	116	111	81	108	102	100	109	84	0.179584	0.9910	nonperiodic-templates
86	119	91	98	98	119	108	89	101	91	0.174728	0.9920	nonperiodic-templates
122	92	103	82	102	117	101	78	96	107	0.045088	0.9900	overlapping-templates
109	108	91	101	90	100	87	108	110	96	0.662091	0.9850	universal
99	100	93	105	97	86	96	100	123	101	0.508172	0.9950	apen
74	58	56	68	61	67	69	59	65	54	0.741506	0.9826	random-excursions
64	73	60	67	64	51	68	65	57	62	0.802867	0.9921	random-excursions
79	54	54	71	72	61	48	62	73	57	0.098152	0.9905	random-excursions
63	45	60	52	59	78	74	52	73	75	0.027441	0.9905	random-excursions
47	69	62	60	61	63	67	60	72	70	0.604104	0.9968	random-excursions
62	63	60	57	82	63	60	64	59	61	0.643730	0.9952	random-excursions
68	63	63	55	59	61	70	68	69	55	0.876153	0.9905	random-excursions
61	44	82	63	60	70	51	53	78	69	0.015275	0.9873	random-excursions
56	70	65	71	75	56	59	58	68	53	0.500934	0.9889	random-excursions-variant
63	66	61	76	70	59	56	68	54	58	0.660243	0.9873	random-excursions-variant
65	66	71	67	64	60	51	62	56	69	0.808725	0.9905	random-excursions-variant
65	58	77	60	60	75	62	53	65	56	0.476207	0.9889	random-excursions-variant
52	70	70	75	50	75	55	59	60	65	0.201925	0.9921	random-excursions-variant
58	68	58	55	76	66	74	52	55	69	0.314520	0.9889	random-excursions-variant
65	64	51	72	66	62	69	50	80	52	0.138408	0.9889	random-excursions-variant
64	63	52	58	71	67	78	57	61	60	0.538952	0.9921	random-excursions-variant
48	64	69	68	69	63	69	61	58	62	0.722324	0.9968	random-excursions-variant
50	78	61	60	79	60	54	63	70	56	0.140937	0.9921	random-excursions-variant
75	64	59	69	65	51	52	51	76	69	0.165500	0.9905	random-excursions-variant
75	64	55	54	66	58	71	50	70	68	0.349202	0.9889	random-excursions-variant
66	66	56	59	65	63	60	67	71	58	0.956262	0.9857	random-excursions-variant
54	62	76	65	54	67	65	63	67	58	0.709396	0.9873	random-excursions-variant
51	68	63	64	74	56	62	63	51	79	0.227067	0.9889	random-excursions-variant
47	66	73	61	67	58	63	68	58	70	0.526168	0.9889	random-excursions-variant

C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	p-values$_T$	Proportion	Test
47	72	67	60	73	63	59	57	72	61	0.391767	0.9873	random-excursions-variant
52	61	70	69	78	61	63	57	55	65	0.473156	0.9873	random-excursions-variant
93	103	95	108	92	87	99	102	116	105	0.672470	0.9880	serial
91	105	119	89	104	109	86	113	84	100	0.159910	0.9940	serial
104	94	94	121	93	105	99	107	93	90	0.532132	0.9820	lempel-ziv
101	97	105	110	104	89	102	87	103	102	0.869278	0.9900	linear-complexity