High Efficient Sequences Generate from Developed MCG Generator

Faez Hassan Ali Al_Azawi / Mathematics Dept. /College of Science /Al_Mustansiriyah University, Email: faezhasan@yahoo.com

Aseel Ghazi Mahmoud Al-Tai /College of Nursing / Baghdad University.

Abstract

It has been before using the mathematical system – the Multipicative Cyclic Group (MCG) - to construct the MCG unit, which is a base unit of MCG systems [1]. In this paper we develop the MCG unit to gain random digital sequences with high efficiency to be used in Stream Cipher Systems. The basic criteria of efficiency are applied to test the digital sequence results from the system which is combined from the proposed unit.

مستخلص

لقد سبق وان تم استخدام النظام الرياضي–الزمرة الضربية الدوارة–لإنشاء وحدة MCG، وهي الوحدة الأساسية
لمنظومة MCG [1]. في هذا البحث سوف نقوم بتطوير وحدة MCG للحصول على متتابعات رقمية عشوائية
ذات كفاءة عالية تستخدم في نظم التشفير الانسيابي. لقد تم تطبيق مقاييس الكفاءة الأساسية على المتتابعات
الرقمية الناتجة من أنظمة مركبة من الوحدات الجديدة المطورة لمعرفة مدى كفاءة الوحدة الجديدة.

Keywords: *Multipicative cyclic group, Stream cipher systems, Random digital sequences, Basic criteria of efficiency.*

1. Introduction

- Let $\langle G,*\rangle$ be a MCG, G with order q-1 where q is prime, let $\alpha\in G$ be a generator element iff $\beta_i=f(\alpha,i,q)=\alpha^i$ (mod q), $1\le i\le q-1$, $\forall\beta_i\in G$. If $(k,q-1)=1$ then β_k is another generator. its cleat that f is 1-1 and onto function. We know that there are $g(q)=\Phi(q-1)$ generators, where Φ is Euler function [2]. The **FIND-GEN(α) algorithm** introduced before to find another generator from the generator α [1].
- The sequence $S=\{s_k\}_{k=1}^{q-1}$, $0\le s_k\le m-1$, $m\ge 2$, s.t. $s_k=(m\cdot\beta_k)$ div q, k=1,…,q-1, (div gives the quotient) can be generates from one generator element α where $\beta_k=f(\alpha,k,q)$ is any element in G. Its clear that the period P(S) of S is (q-1). The **ONE-GEN algorithm** introduced before to generate S by one generator α [1].
- The sequence $S=\{s_k\}_{k=1}^{q-1}$, $0\le s_k\le m-1$, is a sequence generate by two generators $\alpha_t,\alpha_r\in G$, choosing $\alpha_t\neq\alpha_r$ let $x=f(\alpha_t,k,q)$ and $y= f(\alpha_r,x,q)$, $1\le k,x\le q-1$, so we can defined the function h s.t. $y=f(\alpha_r,f(\alpha_t,k,q),q) = h(\alpha_r,\alpha_t,k,q)$, where $h:G\to G$, its cleat that h is 1-1 and onto function. The **TWO-GEN algorithm** introduced before to generate S by two different generators α_r and α_t [1].
- MCG unit which generate the sequence S is a function of five independent variables s.t $S = MCGU(q, \alpha_t, \alpha_r, \gamma, m)$, where $1\le\gamma\le q-1$ is an arbitrarily chosen start point [1].
- The **MCGU algorithm** introduced before to generate S [1], know the steps of this algorithm is as follows:

```
*MCGU Algorithm*
INPUT      : READ $q$ , $\alpha_t$ , $\alpha_r$ , $\gamma$ , $m$ , $L$ ;
PROCESS    : $i := \gamma-1$ ; $k := 0$ ;
               REPEAT
                   $i := i \ (\mathrm{mod}(q-1))+1$ ; $k := k+1$ ;
                   $y := h(\alpha_t , \alpha_r , i, q)$ ;
                   $s_k := (m.y) \ \mathrm{div} \ q$ ;
               UNTIL  $k = L$ ;
OUTPUT     : the sequence S ;
END.
```

The good efficiency statistical results of the MCGU are proved using the basic criteria of efficiency; these results are discussed in [1].

2. Generating All Generators of G

Let us choose the generator $\alpha \in G$, let A(q) be the set of all possible generators $\alpha_i \in G$ s.t. $1 \le i \le g(q)$, $A(q)=\{\alpha_1,\alpha_2,\ldots,\alpha_{g(q)}\}$ generated by α using the **FIND-ALL-GEN(α) algorithm** mentioned below.

```
*FIND-ALL-GEN($\alpha$) Algorithm*
INPUT      : $q$ ,$\alpha$ ;
PROCESS    : $A(q)=\{\}$ ;$k := 1$;  $i := 0$;
               REPEAT
                   $k := k + 1$;
                   IF $\gcd(k,q-1)=1$ THEN
                       $i := i + 1$ ;
                       $\alpha_i= f(\alpha_i,k,q)$ {*is a generator of MCG*}
                       $A(q):=A(q)+\{\alpha_i\}$;
                   ENDIF
               UNTIL $k > q-2$ ;
OUTPUT     : The set A(q) of all generators of MCG;
END.
```

Its obvious that we can choose α in g(q) ways, so there are $\phi(q-1)$ ways of A(q) elements arrangements.

3. Developing MCGU

Let $\langle G,*\rangle$ be a MCG, G with order q-1, choose α_t as generator of MCG to construct the set A(q), let $x=f(\alpha_i,k,q)$ and $y=f(\alpha_j,x,q)$, that implies $y= f(\alpha_i,f(\alpha_j,k,q),q)= h(\alpha_i,\alpha_j,k,q)$, $1 \le k,x \le q-1$. Its proved that $h:G \rightarrow G$ is 1-1 and onto function [1].

Now we want to introduce the following variables, which are, being useful in our work:

1. Choose q prime number, γ, m and L.
2. Choose α_r to construct the set A(q) of all possible generators α_i of G.
3. Choose the initial value γ, s.t. $1 \le \gamma \le q-1$.
   Take the cyclic value $l = \gamma,\ldots,q-1,1,2,\ldots,\gamma-1$; $k=1,\ldots,q-1$.
   Calculate $y = h(\alpha_i ,\alpha_j , l , q)$, $i=1,\ldots,q-1$, $i \ne j$.

Calculate s_k = (m.y) div q.

We want to formulate these choices in a new algorithm to introduce a new unit in order to generate sequence of period larger than q-1, which we call it, a **Modified MMCG Unit (MMCGU)**.

We introduce a new algorithm we call it **Modified MCG (MMCG) algorithm.** Let S be the sequence generate from MMCG Unit (MMCGU) with length L s.t. S=MMCG(q, m, α_i, α_j, γ), $1\leq\gamma\leq q-1$, $\alpha_i,\alpha_j\in A(q)$, $1\leq i\neq j\leq g(q)$.

```
MMCGU Algorithm
INPUT        : READ q , $\gamma$ , m , L ;
PROCESS      : r := 0;
               REPEAT
                 r := r + 1;
                 READ $\alpha_r$;
                 CALL FIND-ALL-GEN($\alpha$);
                 Find A={$\alpha_1$, $\alpha_2$,…, $\alpha_{g(q)}$};
                 i := 0 ; t := 0;
                 REPEAT
                      i := i + 1 ;
                      l:= $\gamma$-1 ; k := 0 ; j := 0;
                      REPEAT
                          j := j + 1 ;
                          IF i$\neq$j THEN
                              t := t + 1;
                              l:= l (mod(q-1))+1 ; k := k +1 ;
                              y := h($\alpha_i$ ,$\alpha_j$ , l , q ) ;
                              $s_k$ := (m.y) div q ;
                          ENDIF;
                      UNTIL  k = q ;
                 UNTIL  (i = g(q))
               UNTIL (r=g(q)) OR (t =L) ;
OUTPUT       : the sequence S ;
END.
```

4. Efficiency Criterions Calculations of MMCGU

- **Periodicity P(S):** Period of S with every two different generator elements is q-1, since we have $P_2^{g(q)}$ generators then the period of MMCGU is: P(S)= $P_2^{g(q)}$ *(q-1), while the period of MCG unit is just q-1.
- **General Complexity GC(S):** Let T be the number of primes q in some range or in available data base, we have g(q) ways to choose α to construct A(q), $P_2^{g(q)}$ ways to choose two different generators in A(q), and q-1 ways to choose γ then: GC(S) = T*ϕ(q-1)* $P_2^{g(q)}$ *(q-1), while the general complexity of MCG unit is T* $P_2^{g(q)}$ *(q-1).
- **Randomness:** Since every subsequence generates from single generator has good randomness properties of S then we expect S has good properties (the good randomness of MCGU is proved in [1]).
- **Linear Complexity (LC):** it's proved before that the linear complexity is large as possible for MCGU [1] because of the high non-linearity of the MCG unit. Since we don't change the strategy of constructing the function to generate the sequence S

then we expect the proposed new unit has good linear complexity and this is shown in table (1).

Table (1) shows The Periodicity, General Complexity and Linear Complexity [2] tests for some binary sequences (m=2) which are generate from MMCGU for different primes compared with MCG unit (the maximum length of tested sequences not exceed 16000 bits for primes with P(S) larger than this number).

Table (1) efficiency criterions for MMCG output results.

	MCG			**MMCG**		
Primes	**P(S)**	**GC(S)**	**LC(S)**	**P(S)**	**GC(S)**	**LC(S)**
101	100	156000T	50	156000	6240000T	1950
1009	1008	83225520T	506	83225520	23968949760T	8000
4111	4110	4860716160T	2055	4860716160	5288459182080T	8001

Its important to mentioned that the linear complexity is the minimum equavilant Linear Feedback Shift Register (LFSR) generate the given sequence [3].

5. MMCG System (MMCGS)

We can use MMCG unit as a construction unit in MMCG system with Combining Function (CF). If S is the sequence which is generate from MMCG system has a F_n as a combining function with n_MMCG units then:

$S=F_n(S_1,S_2,\ldots,S_n)$ s.t. $S_i=MMCGU_i(q_i,\alpha_{ti},\alpha_{ri},\gamma_i,m)$, where $1\le i\le n$.

S_i represents the sequence i generate from the MMCG unit i.

Its important to mention that:

$$\left.\begin{array}{l} s_j = s_{ij} + s_{kj} \pmod m \\ s_j = s_{ij} * s_{kj} \pmod m \end{array}\right\} \begin{array}{l} s_j\in S,\ j=1,2,\ldots \\ s_{ij}\in S_i \text{ and } s_{kj}\in S_k,\ 1\le i,k\le n. \end{array}$$

Table (2) shows output results of various MMCG systems tested by CRYPT-X'98 using Period, Linear Complexity, Frequency (FT), Binary Derivative (BDT), Change Point (CPT), Subblock (SBT), Run (RT) and Sequence Complexity (SCT) test [4].

Table (2) output tests results of MMCG linear systems for m=2, n=2 and 3 .

		CRYPT -X'98 tests							
n	**Primes**	**P(S)**	**LC**	**FT**	**BDT**	**CPT**	**SBT**	**RT**	**SCT**
2	101 997	1.0585×10^8	8001	P	P	P	P	P	P
3	17 31 43	18480	8001	P	P	P	P	P	P

In General, the periodicity of the sequence S of a system can be obtained by the least common multiple (lcm) of all combined sequences [5]:

$P(S) = \text{lcm}\,(P(S_1), P(S_2), \ldots, P(S_n))$.

6. More Digital Randomness Tests

The sequences S generated from MMCG system is being tested for binary randomness test (m=2) only. In this section we can test these sequences by using the Main Digital Standard Randomness Tests (MDSRT) Three basic tests are used; Frequency (Freq), Run and Auto Correlation tests (AC) [6].

Now we will test three different digital sequences for m=3, 5 and 7, with different length L=2000, 5000 and 8000 digits respectively. All these sequences are generated from different linear MMCGS's (CF is XOR function) have the initial keys described in table (3).

Table (3) the Three MCGS's initial key.

MMCGS	n	q_i	α_{ti}	α_{ri}	γ_i	m
1	2	101	2	8	1	3
		997	7	855	1	
2	3	199	3	44	1	5
		1103	5	125	1	
		3607	5	3125	1	
3	5	149	2	8	1	7
		509	2	8	1	
		1051	7	567	1	
		1301	2	8	1	
		2003	5	125	1	

The following three tables (table (4), (5) and (6)) are show the randomness test results of the three digital sequences mentioned above by using MDSRT (where T^* is chi-square test value, T_0 is the tabled pass value and υ degree of freedom).

Table (4) MDSRT results of MMCGS output with L=2000 for m=3.

Test	T^* Value	υ	Pass Value T_0
Frequency	2.428	2	6.01
Run	6.971	6	12.31
	7.229	6	12.31
	6.63	5	10.97
Auto Correlation	No# of fail values $0.0\le T(\tau)\le 14.238$ 0.05% for 500 shift	1	3.81

Table (5) MDSRT results of MMCGS output with L=5000 for m=5.

Test	T^* Value	υ	Pass Value T_0
Frequency	2.294	4	9.52
Run	1.4	3	7.84
	3.49	4	9.52
	5.73	4	9.52
	6.62	3	7.84
	10.99	4	9.52
Auto Correlation	No# of fail value $0.0\le T(\tau)\le 9.465$ 0.07% for 500 shift	1	3.81

Table (6) MDSRT results of MMCGS output with L=8000 for m=7.

Test	T^* Value	υ	Pass Value T_0
Frequency	6.992	6	12.309
Run	2.997	4	9.52
	3.458	3	7.84
	7.088	4	9.52
	5.982	3	7.84
	6.283	3	7.84
	1.823	3	7.84
	3.429	3	7.84
Auto Correlation	No# of fail values $0.0\le T(\tau)\le 15.899$ 0.068% for 500 shift	1	3.81

7. Conclusions & Recommendations

1. If we compare the MMCGU and LFSR, we get the following differences:
   i. For unknown algorithm, the LFSR variables are the length, tap and initial values are unknown, but MMCGU variables are α_t,α_r,q,γ and m are all unknowns.
   ii. For known algorithm, the initial value (basic key) is unknown only, but in MMCGU, α_t, α_r, q, γ are unknown which are can be considered as initial values.
   iii. The periodicity of the sequence generated from LFSR with length r is 2^r-1, but the period of the sequence generated from MMCG is $P_2^{g(q)}$ *(q-1), where $P_2^{g(q)}$ is the permutation of 2 from g(q).
   iv. The common generated sequence from LFSR is binary, but in MMCGU, the sequence is digital (1<m≤q-1).
   v. The length, tap and initial values of LFSR can be found from some available length of the generated sequence by using Massey algorithm [3], but it's not easy to find the initial value of the MMCGU in spite the availability of the generated sequence because of the high non-linearity of the function h.
2. The MMCGU can be developed to increase its periodicity, complexity and randomness by using other non-used generators of G to generate new A(q).

References

[1]. Ali, F. H., "**Use the Multiplicative Cyclic Group to Generate Pseudo Random Sequences**", Al-Rafidain University College Journal for Sciences, No#20, 2006.

[2]. Gilbert, W. J. "**Modern Algebra with Applications**", Wiley-Interscince, March 2002.

[3]. Massey, J.L., "**Shift Register Synthesis and BCH Decoding**" IEEE Transaction on Information Theory, Vol. IT-15, No.1, 1969.

[4]. Gustafson, H., Dawson, E. "**A Computer Package for Measuring the Strength of Encryption Algorithm**", Computer & Security Vol.13, No.8, 1996.

[5]. Dr. Rahma, Abdul Monem, Dr. Nadia, M. G. and Nasser, A. G., "**The Theoretic Estimation of the Basic Criterions to Evaluate the Key Generator Efficiency before the Practical Construction**", The 1st Information Technology Conference, Iraq, April, 2009.

[6]. Ali, F. H., Mohammed S. A. and Shamran M. A., "**Generalize the Randomness Tests to Test the Digital Sequences Produced from Digital Stream Cipher Systems**", Journal of College of Science / Baghdad University, 2009.