Nonlinear Filter Generator over Elliptic Curve

Tun Myat Aung, Ni Ni Hla

Centre of Advanced Science and Technology,
University of Computer Studies, Yangon, Myanmar
tma.crypto@gmail.com

Abstract. In this paper we propose the construction of nonlinear pseudorandom sequence from the group of points over an elliptic curve. This method is based on an elliptic curve (EC), a linear feedback shift register and a random block. This generator could be used in devices like smart cards which have already been equipped with *EC*-based tools for cryptographic purposes.

Key words: nonlinear pseudorandom sequence, nonlinear filter generator, elliptic curve

1. Introduction

Nonlinear pseudorandom sequences have a significant influence in the development of cryptography, like linear recursive sequences. We introduce a new approach, which generates a nonlinear pseudorandom sequence from elliptic curve. We combine an elliptic curve, a linear feedback shift register and a random block to generate potential nonlinear pseudorandom sequences. This generator logically can be divided into three parts: one that generates a point sequence, one that extracts a nonlinear pseudorandom sequence from the point sequence and one that extracts a bit string from the nonlinear pseudorandom sequence. Our concern is a way of generating the nonlinear pseudorandom sequence from the point sequence. This method can be widely used in the public-key cryptographic applications for generating secure pseudorandom sequences.

The paper is organized as follows. In Section 2 and 3, we give the principle of random block and the idea of nonlinear filter generator. In Section 4, we introduce a method how to extract a nonlinear pseudorandom sequence from the elliptic curve. In Section 5, we give conclusion.

We conclude this section with the definition of elliptic curve over finite field, point addition formula and it's properties.

A. Elliptic Curve over Finite Field GF(p)

We consider finite field **GF**(**p**), where **p**>3 and **p** is a prime number. An elliptic curve **E** over **GF**(**p**) can be written in the following standard form:

$$\mathbf{E : y}^2 = \mathbf{x}^3 + \mathbf{ax} + \mathbf{b} \in \mathbf{GF}(\mathbf{p}),$$

where **a** and **b** are integer numbers over **GF**(**p**) such that $4\mathbf{a}^3 + 27\mathbf{b}^2 \neq 0(\ \mathbf{mod\ p})$. The points **P** = (**x**, **y**), **x**, **y** $\in$ **GF**(**p**), that satisfy this equation, together with *"a point at infinity"* denoted **O**, form an Abelian group (**E**, +,**O**) whose identity element is **O**.

*Addition formula for **E**.*

The operations of addition for two points on an elliptic curve over **GF**(**p**) have the following properties.

1. **P** + **O** = **O** + **P** = **P**, where **P** $\in$ **E**.
2. For any **P** = ($\mathbf{x}_1$, $\mathbf{y}_1$) $\in$ **E**, the point -**P** = ($\mathbf{x}_1$, -$\mathbf{y}_1$). If **Q** = -**P**, **P** + **Q** = **O**.
3. If **P** = ($\mathbf{x}_1$, $\mathbf{y}_1$) $\in$ **E** and **Q** = ($\mathbf{x}_2$, $\mathbf{y}_2$) $\in$ **E**, where **P** or **Q** $\neq$ **O** and **Q** $\neq$ -**P**, then **R** = **P** + **Q** = ($\mathbf{x}_3$, $\mathbf{y}_3$). (see Fig. 1)

$$\mathbf{x}_3 = \lambda^2 - \mathbf{x}_1 - \mathbf{x}_2 \ (\mathrm{mod}\,\mathbf{p}),$$
$$\mathbf{y}_3 = \lambda(\mathbf{x}_1 - \mathbf{x}_3) - \mathbf{y}_1 \ (\mathrm{mod}\,\mathbf{p}),$$

where

$$\lambda = \begin{cases} \dfrac{\mathbf{y}_2 - \mathbf{y}_1}{\mathbf{x}_2 - \mathbf{x}_1} (\mathrm{mod}\,\mathbf{p}), \mathrm{if}\ \mathbf{P} \neq \mathbf{Q} \\ \dfrac{3\mathbf{x}_1^2 + \mathbf{a}}{2\mathbf{y}_1} (\mathrm{mod}\,\mathbf{p}), \mathrm{if}\ \mathbf{P} = \mathbf{Q} \end{cases}$$

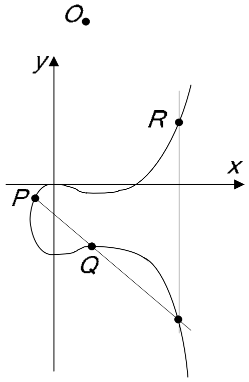


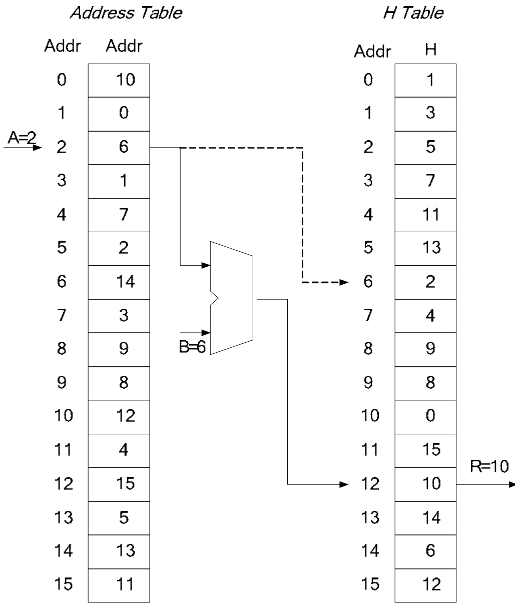
Fig. 1. Adding **P** and **Q**

2. Principle of Random Block (R-block)

Random block (***R***-block) is a nonlinear substitution method. The logic work of ***R***-block and its conditional graphic symbol are shown in figure 2 and 3. ***H*** table is a substitution box, which contains the permutation of elements over $\boldsymbol{GF}\left(2^{\boldsymbol{m}}\right)$, i.e, $\boldsymbol{H} = \{\boldsymbol{e}\},\ \boldsymbol{e} \in \boldsymbol{GF}(2^{\boldsymbol{m}})$. The address table is filled with addresses of corresponding cells containing elements in the ***H*** table. The formula of a ***substitution*** transformation of ***R***-block is as following:

$$\boldsymbol{R_H}\left(\boldsymbol{A,B}\right) = \boldsymbol{H}\left(\left(\boldsymbol{e_A} + \boldsymbol{B}\right)\ \boldsymbol{mod}\ 2^{\boldsymbol{m}}\ \right)$$

where $\boldsymbol{A}, \boldsymbol{B}$ are input elements $\in \boldsymbol{GF}\left(2^{\boldsymbol{m}}\right)$, $\boldsymbol{e_A}$ is an address of cell containing element ***A*** in ***H*** table, i.e., $\boldsymbol{H}(\boldsymbol{e_A}) = \boldsymbol{A}$. The output of ***R***-block is the essence of reading the contents of cells in ***H*** table after summation of an input element ***B*** and an address of cell containing input element ***A***.



Address Table

Addr	Addr
0	10
1	0
2	6
3	1
4	7
5	2
6	14
7	3
8	9
9	8
10	12
11	4
12	15
13	5
14	13
15	11

H Table

Addr	H
0	1
1	3
2	5
3	7
4	11
5	13
6	2
7	4
8	9
9	8
10	0
11	15
12	10
13	14
14	6
15	12

Fig. 2. The logic work of ***R***-block ∈ **GF**(2^4)

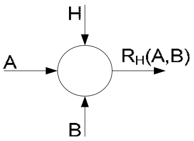


Fig. 3. Conditional graphic symbol for ***R***-block

3. Principle of Nonlinear Filter Generator

A linear feedback shift register (LFSR) should not be used in cryptographic work because the outputs are completely linear, leading to fairly easy cryptanalysis. Therefore a better approach is to use a nonlinear transformation in LFSR. A way to destroy the linearity in LFSR is to use a nonlinear filter function $\boldsymbol{f}$ with single output $\boldsymbol{z}$ and $\boldsymbol{j}$ inputs. It is called a nonlinear filter generator(NFG) (see Fig. 4). It uses a single LFSR. Its keystream is generated by a nonlinear function $\boldsymbol{f}$ of the stages of the LFSR. Practical implementation of nonlinear filters uses S-boxes which have essentially the same requirements as those used in block ciphers.

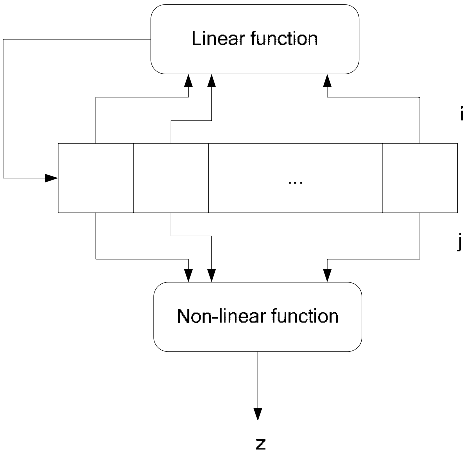


Fig. 4. General design of nonlinear filter generator

4. Construction of Nonlinear Pseudorandom Sequence from Elliptic Curve

G. Gong and C. Lam proposed the linear sequence over elliptic curve, based on the construction of LFSR. This sequence is well known as LFSR-EC sequence. Now we introduce a method how to extract nonlinear pseudorandom sequence from an elliptic curve, based on ***R***-block and LFSR. This sequence is an NFG sequence based on an elliptic curve.

Definition 1. Let **E** be an elliptic curve over **GF(p)** of order **r**, $f(x) = x^n - c_{n-1}x^{n-1} - \cdots - c_1 x - c_0 \in Z_r[x]$ be a monic polynomial over Z_r and **P** = { P_k } where P_k = (**x_k**, **y_k**) ∈ **E**. For fixed $c_{-1} \in Z_r$ and given (**n** + 1) – tuple $(Q, P_0, \ldots, P_{n-1}) \in E^{n+1}$, if the sequence satisfies the following linear recursive relation:

$$P_{n+k} = \sum_{i=0}^{n-1} c_i P_{i+k} + c_{-1} Q, \; k = 0, 1, \ldots,$$

then **P** is called an **n**th order linear recursive sequence over **E** or simply EC sequence for short; $f(x)$ is called a characteristic polynomial of **P** over Z_r and $(Q, P_0, P_1, \ldots, P_{n-1})$ is called an initial state of **P**.

Firstly, we construct a linear recursive sequence over **E** that is called LFSR-EC sequence **P** according to definition (1). Then we choose a nonlinear function f which is called ***R***-block that is built of elements over **GF**($2^{\mathbf{m}}$). The elements of **P** are input to the nonlinear function f and it generates a nonlinear pseudorandom sequence $\underline{z} = \{z \in GF(2^m)\}$ as an output keystream of generator. The formula:

$$R_H(A,B) = H(e_A + B) \in GF(2^m)$$

is defined as a nonlinear function $f : P \to z$, where A and B are the values (i.e., $\mathbf{x_k}$, $\mathbf{y_k}$) of coordinates of the element P_k over LFSR-EC sequence **P**. If the elements A and B that is input to the nonlinear function f are out of the field $GF(2^m)$, these elements will be transformed to the elements of the field $GF(2^m)$ according to the modular arithmetic. The binary pseudorandom sequence **b** is obtained by using the function ***h***. The function ***h*** maps the elements of $GF(2^m)$ to the elements of $GF(2)$. The function ***h*** is defined as following.

$$h(z)=\begin{cases}0, 0\le z\le \dfrac{2^m-1}{2}\\ 1, \quad \text{otherwise}\end{cases}$$

Example 1. (curve defined on ***GF(p)*** *).* Consider the curve $\boldsymbol{E}:\boldsymbol{y}^2=\boldsymbol{x}^3+\boldsymbol{x}+4$ over **GF**(23).

This curve has order 29 and is cyclic. Let ***P*** = (4, 7) be a point on ***E***. Choose $\boldsymbol{f}(\boldsymbol{x})=\boldsymbol{x}^2-28\boldsymbol{x}-26\in\boldsymbol{Z}_{29}[\boldsymbol{x}]$. Taking an initial state ***Q*** = ***O***, $\boldsymbol{P_0}$ = ***P*** and $\boldsymbol{P_1}$ = 2***P***, then the recursive relation is:

$$\boldsymbol{P_k}=28\boldsymbol{P_{k-1}}+26\boldsymbol{P_{k-2}}=\boldsymbol{a_k}\boldsymbol{P},\boldsymbol{k}=2,3,\ldots$$

where

$$\boldsymbol{a_k}=28\boldsymbol{a_{k-1}}+26\boldsymbol{a_{k-2}},\boldsymbol{k}=2,3,\ldots$$

which is an LFSR sequence over $\boldsymbol{Z}_{29}$. This recursive relation gives the following LFSR-EC sequence **P.**

P = (P, 2P, 24P, 28P, 16P, 16P, 23P, 16P, 2P, 8P, 15P, 19P, 23P, 7P, 11P, 26P, 28P, 10P, 22P, 6P, 15P, 25P, 17P, 24P, 12P, …).

Let's generate a nonlinear pseudorandom sequence $\underline{z}=\{z\in \boldsymbol{GF}(2^4)\}$. Construct ***R***-block that consists of the permutation of elements over **GF**(2^4) (as shown in Fig. 2). The ***R***-block, that is a nonlinear function $\boldsymbol{f}$, gives the nonlinear pseudorandom sequence with related to the cyclic group of points over the given elliptic curve (see Table. 1). The nonlinear pseudorandom sequence obtained from **P** by applying the nonlinear function $\boldsymbol{f}$ which is called ***R***-block is:

z = { 6, 6, 9, 4, 12, 12, 10, 12, 6, 12, 1, 5, 10, 4, 8, 3, 4, 14, 2, 15, 1, 10, 5, 9, 12, …}.

The binary sequence obtained from **z** by applying the mapping function ***h*** is:

b = { 0, 0, 1, 0, 1, 1, 1, 1, 0, 1, 0, 0, 1, 0, 1, 0, 0, 1, 0, 1, 0, 1, 0, 1, 1, …}.

Table 1.

	iP= (x, y)	**R_H(iP)**	**i**	**iP= (x, y)**	**R_H(iP)**	**i**	**iP= (x, y)**	**R_H(iP)**	**i**	**iP= (x, y)**	**R_H(iP)**
	(4, 7)	6	9	(9, 12)	11	17	(14, 5)	5	25	(15, 17)	10
	(10, 18)	6	10	(11, 9)	14	18	(17, 14)	6	26	(13, 12)	3
	(13, 11)	1	11	(17, 9)	8	19	(11, 14)	5	27	(10, 5)	3
	(15, 6)	3	12	(14, 18)	12	20	(9, 11)	7	28	(4, 16)	4
	(8, 8)	3	13	(0, 2)	10	21	(18, 14)	11	29	O	0
	(1, 11)	15	14	(22, 5)	7	22	(7, 3)	2			
	(7, 20)	4	15	(22, 18)	1	23	(1, 12)	10			
	(18, 9)	12	16	(0, 21)	12	24	(8, 15)	9			

5. Conclusion

This paper presents the new concept of using elliptic curves for generating secure pseudorandom numbers. This sequence has the uniform distribution, the unpredictability and no dependences between elements of a sequence. This generator may not lead to fairly easy cryptanalysis. Thus NFG over elliptic curve is a cryptographically strong generator.

References

1. B. Schoenmakers, A. Sidorenko. Cryptanalysis of the Dual Elliptic Curve Pseudorandom Generator. 2006.
2. D. Hankerson, A. Menezes, S. Vanstone. Guide to Elliptic Curve Cryptography, Springer, 2004.
3. E. Barker, J. Kelsey. Recommendation for Random Number Generation Using Deterministic Random Bit Generators, NIST Special Publication 800-90, 2006.
4. E. Shparlinski. On The Naor-Reingold Pseudorandom Function From Elliptic Curves, 1999.
5. G. Gong, C. Lam. Linear Recursive Sequences Over Elliptic Curves, 2001.
6. G. Gong, T. Berson, D. Stinson. Elliptic Curve Pseudorandom Sequence Generators. 1998. Technical Report.
7. P. H. T. Beelen, J. M. Doumen. Pseudorandom sequences from elliptic curves, 2002.
8. S. Hallgren. Linear Congruential Generators Over Elliptic Curves, 1994.
9. М. А. Иванов, И. В. Чугунков, Теория, применение и оценка качества генераторов псевдослучайных последовательностей, 2003.