A Fast Scalar Multiplication Algorithm Based on Alternate-Zeckendorf Representation

Shuang-Gen Liu and Xue-Jing Sun
(Corresponding author: Shuang-Gen Liu)

Department of Information and Communication Engineering, Xi'an University of Posts and Telecommunications
W Chang'an Ave, ChangAnQu XiBu DaXueCheng ShangQuan, Changan Qu, Xi'an, Shaanxi 710121, China
(Email: liusgxupt@163.com)

(Received May 15, 2017; revised and accepted Sept. 25, 2017)

Abstract

This paper proposed a new method to point scalar multiplication on elliptic curve and it is defined in a finite field with a characteristic greater than 3. It is based on the Transformed Fibonacci type sequence like (2P + Q) and it can resist the Simple Power Attack (SPA). Although the sequence quite easier to calculate, expressing any k using the sequence remains a very difficult problem, so we proposed the Alternate-Zeckendorf representation and given the proof of this view.The NewADD algorithm is also added to the new algorithm and in meanwhile we also listed (2P + Q) results as a table to reduce the computation cost.The performance comparisons show that our algorithm is less costly than other algorithms 12.7 % to 27.9 % at least.

Keywords: Addition Chain; Elliptic Curve; Fibonacci Sequence; Scalar Multiplication; Simple Power Attack

1 Introduction

Since the Koblit [8] and Miller [15] firstly applied the elliptic curve in the encryption system, Elliptic Curve Cryptography (ECC) has received more and more attention. It gradually become a mainly standard in public key cryptography and widely used in various areas of information security, such as message encryption, authentication and digital signatures [2, 6, 9, 16, 17, 21]. The literature [20] proposed a combination of RSA and ECC. Compared with RSA, the advantages and benefits of the ECC as follows:

- High security. RSA's security level is sub-exponential and ECC is exponential.
- Shorter key. At the same level of security, the secret key length of ECC is much smaller than that of RSA and ElGamal, which makes the ECC is applied in the storage-constrained environments.
- Small storage space,lower bandwidth requirements. This advantage allows ECC have a good prospect in many limited areas.
- The faster computational speed. Due to the small size of the finite field bottom of ECC and the deepening of related research, it is much faster than RSA's computational speed.

The SET agreement which is introduced by the Visa and Master card has set its default public key cryptography algorithm to ECC. That means with the development of ECC it will be gradually replace the status of RSA mainstream application algorithms.

Although the existing ECC is much faster than RSA, with the development of information technology, the existing computing speed has been unable to meet people's need [22]. So it is imperative to improve the computational efficiency of ECC. However, In elliptic curve calculation the most basic and time-consuming operation is the elliptic curve scalar multiplication (ECSM), which calculate the [k]P, *i.e.*, the computation of the point $kP = P + P + \cdots + P$, where the integer k is in the known domain, the P is a point on the elliptic curve. The literature [19, 23] introduced a fast scalar multiplication algorithm. In fact, the entire process of computing ECSM on two levels: top level and bottom level. Among them, the top level operation refers to the scalar multiplication is converted to the double and addition operation on the elliptic curve. On the other hand, the bottom level operation refers to through the multiplication, square, inversion, addition operation to achieve double and addition operation on top level. Therefore, there are two aspects of the research of scalar multiplication: The top level seeks the efficient representation of scalar k, and the bottom level find the methods achieved the fast calculation of point doubling and addition. Obviously the top level is the operation on the elliptic curve E and the common methods are double-and-add, NAF, sliding window method and so on. The literature [5, 11] were introduced separately improving Miller's Algorithm using NAF, Window NAF algorithm and extend Φ-NAF algorithm. The literature [10] introduced the bottom algorithm on Jacobian coordinates. Our algorithm select the former to research.

In this paper, we will devote to the two levels. Firstly, we proposed a new addition chain based on the deformation of the Fibonacci sequence [12, 13] and prove the theory that any integer can be expressed by the sum of this sequence. Secondly, we presented a new scalar multiplication algorithm based on the transformed Fibonacci sequence and the algorithm form is like (2P + Q). By the method of combining the NewADD algorithm with new algorithm and calculating the results of (2P + Q) form the table we improved the efficiency of the calculation greatly. In addition to that, our algorithm can resist Simple Power Attack (SPA) [7, 18] naturally, SPA is a type of side channel attack discovered by Kocher *et al.* Literature [1] introduced a method that resist SPA using the addition chain.

The paper is organized as follows. Section 2 gives some relevant conception and definition for the ECC. Section 3 is about addition chain definition. Section 4 introduce our new algorithm and some examples to description. Section 5 we compare and analysis with the other algorithms. And the summary is given in Section 6.

2 Elliptic Curve Arithmetic

2.1 Definition of Elliptic Curve

An elliptic curve E over a finite field K is defined by the equation:

$$y^2 + a_1xy + a_3y = x^3 + a_2x^2 + a_4x + a_6. \quad (1)$$

where $a_1, a_2, a_3, a_4, a_6 \in K$, and $\Delta \neq 0$. It is defined as

$$\begin{aligned} \Delta &= d_2^2d_8 - 8d_4^3 - 27d_6^2 + 9d_2d_4d_6 \\ d_2 &= a_1^2 + 2a_2 \\ d_4 &= 2a_4 + a_1a_3 \\ d_6 &= a_3^2 + 4a_6 \\ d_8 &= a_1^2a_6 + 4a_2a_6 - 4a_1a_3a_4 + a_3a_2^2 - a_4^2. \end{aligned}$$

When the characteristic of K is not equal 2 or 3, the equation can be written in another form, such that

$$y^2 = x^3 + a_4x + a_6. \quad (2)$$

where $a_4, a_6 \in K$, and $\Delta = -16(4a_4^3 + 27a_6^2) \neq 0$.

In practice, we simplified the formula to

$$y^2 = x^3 + ax + b. \quad (3)$$

where $a, b \in K$ and $4a^3 + 27b^2 \neq 0$, over the characteristic greater than 3. The set of points of $E(K)$ is an Abelian group.

2.2 Addition on Elliptic Curve

People proposed several algorithms to compute the addition of two points and also found the many coordinate systems to improve the addition efficiency, you can refer to [4]. Algorithm 1 is executed in the Jacobian coordinates.

Algorithm 1 EACCD

```
**Input:** $P = (X_1, Y_1, Z_1)$ , $Q = (X_2, Y_2, Z_2)$
**Output:**$P + Q$
$A \leftarrow X_1Z_2^2, B \leftarrow X_2Z_1^2, C \leftarrow Y_1Z_2^2, D \leftarrow Y_2Z_1^2$
$E \leftarrow B - A, F \leftarrow D - C$
$X_3 \leftarrow F^2 - E^3 - 2AE^2$
$Y_3 \leftarrow F(AE^2 - X^3 - CE^3)$
$Z_3 \leftarrow Z_1Z_2E$
**Return**$(X_3, Y_3, Z_3)$
```

If one of the point given the form like $(X, Y, 1)$, we can obtain the addition cost is 12 multiplications(M) and 4 square(S), the cost of the doubling is 4 multiplications(M) and 6 square(S).

However, if we assume that P and Q sharing the same Z-coordinate, P = (X_1, Y_1, Z) and Q = (X_2, Y_2, Z). Then P+Q=$(X_3, Y_3, Z_3) = (X_3^{'}Z^6, Y_3^{'}Z^9, Z3^{'}Z^3) \sim (X_3^{'}, Y_3^{'}, Z_3^{'})$. Where $A = (X_2 - X_1)^2, B = X_1A, C = X_2A, D = (Y_2 - Y_1)^2$, therefore

$$\begin{aligned} X_3^{'} &= D - B - C \\ Y_3^{'} &= (Y_2 - Y_1)(B - X_3) - Y_1(C - B) \\ Z_3^{'} &= Z(X_2 - X_1) \end{aligned}$$

The cost is reduced to 5M+2S.

3 Addition Chain Theory

Addition Chain: The addition chain is defined as a sequence $v = (v_1, \cdots, v_l)$, where $v_1 = 1, v_l = k, v_i = v_{i-1} + v_{i-2}(1 \leq i \leq l)$. And the l is a length of the addition chain [3].

Euclidean Addition Chain: The Euclidean addition chain (EAC) of k is defined as an addition chain which satisfies $v_1 = 1, v_2 = 2, v_3 = v_1 + v_2$ and $\forall 3 \leq i \leq s-1$, if $v_i = v_{i-1} + v_j$, some $j < i-1$, then $v_{i+1} = v_i + v_{i-1}$ or $v_{i+1} = v_i + v_j$ [14].

Theorem 1. *We denotes the average number of steps to compute gcd*(k, g) *using the subtraction Euclid's algorithm when* g *is uniformly distributed in the range* $1 \leq \mathrm{g} \leq \mathrm{k}$ *[14].*

$$S(\mathrm{k}) = 6\pi^{-2}(ln\mathrm{k})^2 + O(log\mathrm{k}(logk)^2). \quad (4)$$

This formula shown that the chain length of the EAC is determined by random g and the chain length is about $(lnk)^2$, the chain length is too long to practical application. Even if we choose the g close to the k/ϕ, where $\phi = (1 + \sqrt{5}) \div 2$ is the golden section, the average chain length is still nearly 1100. So we need to find the appropriate addition chain, however how to seek the shortest addition chain is NP complete problem and solving this problem is very difficult for us. In order to avoid the effect of g, we choose the Fibonacci sequence and the definition is shown as follows.

Fibonacci Sequence: The Fibonacci sequence is defined as $F_0 = 0, F_1 = 1, F_i = F_{i-2} + F_{i-1}$ [13].

Fibonacci Type Sequence: All sequences satisfied the Fibonacci condition $F_i = F_{i-2} + F_{i-1}$ collectively called Fibonacci type sequence.

We use the example to illustrate the difference between the Fibonacci Sequence and the Fibonacci Type Sequence, the sequence $\{0, 1, 1, 2, 3, 5, 8, \cdots\}$ is the Fibonacci sequence, but the Fibonacci type sequence like $\{4, 5, 9, 14, 23, \cdots\}$, this means that the Fibonacci sequence is a special form of the Fibonacci type sequence. Fibonacci type sequences have a greater range and form. Thus we can compute the Fibonacci type sequence of arbitrary integers.

4 A New Algorithm about 2P+Q

In previous section we have presented various addition chain forms. Notice that the Fibonacci type sequence is a special EAC and no gap in the application, and inherited all the advantages of EAC. Based on this idea, we proposed a new Transformed Fibonacci type sequence in this section and introduced how to apply this algorithm in the elliptic curve.

4.1 Transformed Fibonacci Type Addition Chain

Transformed Fibonacci Type Addition Chain: A sequence which satisfied the formula $T_i = 2T_{i-2} + T_{i-1}$ $(4 \leq i \leq l)$, where $T_1 = 1$, $T_2 = 2$, $T_3 = 3$. We called this sequence as Transformed Fibonacci Type Addition Chain(TFTAC), marked as T. For example $T = \{1, 2, 3, 7, 13, \cdots\}$.

Alternate-Zeckendorf Representation: Let k be an integer and $T_i (i \geq 0)$ is the Transformed Fibonacci Type (TFT) sequence, the k can be written in the form:

$$k = \Sigma d_i [2T_{i-2} + T_{i-1}]. \quad (5)$$

with $d_i \in \{0, 1\}$. This representation is similar to that the Zeckendorf representation, we called it Alternate-Zeckendorf (A-Z) representation method.

Proof. We use the inductive deduction to prove it. For the $i = 1, 2, 3$, there is no doubt that it is correct since these are Transformed Fibonacci definition numbers, for $i = 4$ we get $7 = 2 * 2 + 3$. Now suppose each $i \leq k$ can be represented correctly. If $k + 1$ is a Transformed Fibonacci number then we're done. Otherwise hypothesis there exists j and $T_j < k + 1 < T_{j+1}$, making the $a = k + 1 - T_j$. Since $a < k$ and a can be represented, so $k + 1 = a + T_j$. At the same time $T_j + a < T_{j+1}$, $a < T_{j+1} - T_j = 2T_{j-1}$. Taking into account the relationship between $2T_{j-1}$ and T_j we obtained that the $a \leq T_j$, now consider the following two cases:

Case 1: where $a = T_j$, a is represented as $a = 2T_{j-2} + T_{j-1}$, the $k + 1$ can be represented as a and T_j.

Case 2: where $a < T_j$, so representation a does not contain T_j. As a result, $k + 1$ can be represented as the sum of T_j and representation a.

□

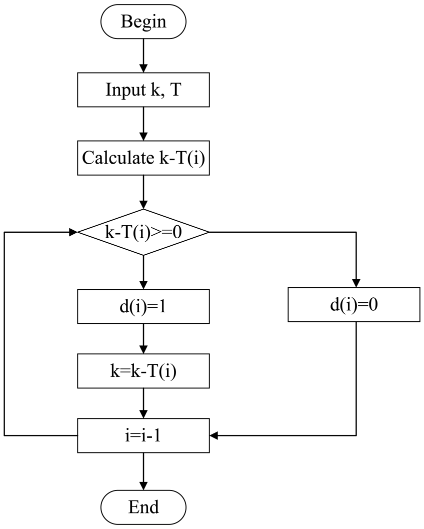

Figure 1: The flow chart of the addition chain

In summary, $k + 1$ can be represented with this method. Therefore, for any k, it can be represented using A-Z representation.

4.2 Specific Algorithm

4.2.1 Generated the A-Z Representation Sequence

Now, we discuss how to generate a sequence represented by A-Z. Firstly, we set the third number of the sequence is equal to the second number plus the doubling of the first number , in this way we can gain the next result until all the T number are calculated. Then we calculate the addition chain of the k. For any integer k, we calculate the $k - T_i$, where i is the last element of T. Case 1: if $k - T_i \geq 0$, then set $k = k - T_i$, mark $d_i = 1$, until the $k = 0$. Case 2: if $k - T_i < 0$, mark $d_i = 0$ then calculate $k - T_{i-1}$, till the $k - T_i \geq 0$ using Case 1. The flow chart is shown in Figure 1.

Algorithm 2 is a specific TFTAC algorithm. Through a large number of experiments we found that the number of T is generally related to the number of the binary bits of k. For example, if we select 160-bits integer to

Algorithm 2 Transformed Fibonacci Addition Chain

```
Input: A positive integer $k$ and $n$
Output: d=$\{d_1, d_2, \cdots, d_n\}$.
$T = \{1, 2, 3\}$, d=$\{\}$,Set $i = 4$
while $i \leq n$ do
    $T_i = T_{i-2} + T_{i-1}$
    Output T.
end while
while $j \geq 1$ and $j \leq n$ do
    if $k - T_n < 0$ then
        Output n is not appropriate, return 1.
    end if
    if $k - T_j \geq 0$ then
        $d_j = 1$
        $d = d \bigcup d_j$
        $k = k - T_j$
        $j = j - 1$
    end if
    if $k - T_j < 0$ then
        $d_j = 0$
        $d = d \bigcup d_j$
        $j = j - 1$
    end if
end while
Output d.
```

execute our experiment, the number of the T is 160 as well. That means the chain length of the k obtained by this method is the same as the binary length. But if we use the Zenkendorf method to represent the k, the chain length is about 230. It needs 44 % more digits than the A-Z representation and the binary representation. In fact, we can see that the definition of the A-Z representation is similar to the binary method. Here we use an example to illustrate our algorithm in Example 1.

Example 1. k = 567, n = 10, T ={1,2,3}

$T_4 = 2T_2 + T_3 = 7$
$T_5 = 2T_3 + T_4 = 13$
$T_6 = 2T_4 + T_5 = 27$
$T_7 = 2T_5 + T_6 = 53$
$T_8 = 2T_6 + T_7 = 107$
$T_9 = 2T_7 + T_8 = 213$
$T_{10} = 2T_8 + T_9 = 427$
We get the $T = \{1, 2, 3, 27, 53, 107, 217, 427\}$, then
$k - T_{10} = 567 - 427 > 0, d_{10} = 1, k = 567 - 427 = 140$
$k - T_9 = 140 - 213 < 0, d_9 = 0$
$k - T_8 = 140 - 107 > 0, d_8 = 1, k = 140 - 107 = 33$
$k - T_7 = 33 - 53 < 0, d_7 = 0$
$k - T_6 = 33 - 27 > 0, d_6 = 1, k = 33 - 27 = 6$
$k - T_5 = 6 - 13 < 0, d_5 = 0$
$k - T_4 = 6 - 7 < 0, d_4 = 0$
$k - T_3 = 6 - 3 > 0, d_3 = 1, k = 6 - 3 = 3$
$k - T_2 = 3 - 2 > 0, d_2 = 1, k = 3 - 2 = 1$
$k - T_1 = 1 - 1 = 0, d_1 = 1, k = 1 - 1 = 0$
Output $d = \{1010100111\}$

4.2.2 Application of Elliptic Curve

In previous section, we discussed how to compute the addition chain sequence of k, and now we describe the application of this algorithm on elliptic curves. It is shown in Algorithm 3.

Algorithm 3 Scalar Multiplication Algorithm using TF-TAC

```
Input: $P \in E(K)$, $k = (d_l, \cdots, d_1)$
Output: W = $[k]P \in E(K)$
Begin W=0
if $d_1 = 1$ then
    W ← W + P
end if
(U, V)←(2P, 3P)
for $i = 2, \cdots, l$ do
    if $d_i = 1$ then
        W←W + U
    end if
    (U,V)←(V, 2U+V)
end for
Return W
```

In Section 2 we know that the NewADD algorithm can be used as long as two points sharing the same Z-coordinates. In the later content, we will use this theory and the specific algorithm is shown in Algorithm 4.

Algorithm 4 Scalar Multiplication Algorithm using NewADD

```
Input: $P \in E(K)$, $k = (d_l, \cdots, d_1)$
Output: W = $[k]P \in E(K)$
Begin W=0
if $d_1 = 1$ then
    W← P
end if
(U, V)←(2P, 3P)
for $i = 2, \cdots, l$ do
    if $d_i = 1$ then
        upgrade W
        ($\cdots$,W)← NewADD(W,U)
    end if
    Calculate 2U
    upgrade V
    (U,V)← NewADD(2U,V)
end for
($\cdots$,W)← NewADD(W,U)
Return W
```

Now we need to analyze the cost of the algorithm. Supposing the U= (X_U, Y_U, Z) , V= (X_V, Y_V, Z) and P= $(x, y, 1)$. When d_1 =1, W= $(x, y, 1)$, upgrade the W= (xZ^2, yZ^3, Z) and the cost is 3M+S. At this time we can add W and U using the NewADD algorithm with the cost is 5M+2S. And then doubling the U needs 4M+4S. Since the $Z_{2U} = 2Y_U Z$, thus the $X_V^{'} = (X_V(2Y_U)^2$, $Y_V(2Y_U)^3, 2Y_U Z)$ only costs 2M. Because of the density of "1" is about 0.5, the add step cost is 4M+1.5S. Finally, the total cost is 15M+7.5S.

Example 2. Computation of $[39]P = 27 + 7 + 3 + 2 = (101110)_{A-Z}$:

initialization : W= 0

$d_2 = 1$: W = 0 + 2P = 2P, (U, V) ← (3P, 7P)

$d_3 = 1$: W = 2P + 3P = 5P, (U, V) ← (7P, 13P)

$d_4 = 1$: W = 5P + 7P = 12P, (U, V) ← (13P, 27P)

$d_5 = 0$: (U, V) ← (27P, 53P)

$d_6 = 1$: W = 12P + 27P = 39P

Return W = [39]P

Example 3. Computation of $[67]P = 53 + 13 + 1 = (1010001)_{A-Z}$:

initialization : W= 0

$d_1 = 1$: W = 0 + P = P

$d_2 = 0$: (U, V)← (3P, 7P)

$d_3 = 0$: (U, V)← (7P, 13P)

$d_4 = 0$: (U, V)← (13P, 27P)

$d_5 = 1$: W = P + 13P = 14P , (U, V)← (27P, 53P)

$d_6 = 0$: (U, V)← (53P, 107P)

$d_7 = 1$: W = 14P + 53P = 69P

Return W = [67]P

Example 2 and Example 3 are illustrated for Algorithm 4. In Algorithm 4, the elliptic curve scalar multiplication process has a double and a addition each time.

4.2.3 Improve Algorithm

By the large of statistical analysis, we get the density of "1" in the A-Z representation is approximate 0.5. That means representing a 160-bits integer needs 80 Transformed Fibonacci number and the total multiplication is about 3360. In this part we will introduce how to improve Algorithm 4 to reduce the cost of multiplication. We known that each iteration of the algorithm needs to calculate 2P + Q, so we can calculate the results of 2P + Q in advance and recorded it as a table. That means we do not need to calculate 2P + Q in the algorithm. The part table shown in Table 1.

Table 1: Transformed fibonacci type number

Tth num.	T_1	T_2	T_3	T_4	T_5
TFT num.	1	2	3	7	13
Tth num.	T_6	T_7	T_8	T_9	$\cdots$
TFT num.	27	53	107	213	$\cdots$

Now, we recalculate Example 3 using improved Algorithm. As you can see in Example 4 only the $d_i = 1$ is performed, and the amount of computation is reduced by half.

Example 4. Computation of $[67]P = 53 + 13 + 1 = (1010001)_{A-Z}$:

initialization : W= 0

$d_1 = 1$: TFT num.=1, then W = 0 + P = P

$d_2 = 0$: Next

$d_3 = 0$: Next

$d_4 = 0$: Next

$d_5 = 1$: TFT num.=13, then W = P + 13P = 14P

$d_6 = 0$: Next

$d_7 = 1$: TFT num.=53, then W = 14P + 53P = 69P

return W = [67]P

5 Comparison with Other Algorithms

In this section we will compare with other algorithms and give some practical results about new algorithm. We choose mixed coordinates to calculate the cost of the TFTAC and the cost of the various mixed coordinates is shown in Table 2 [4]. In addition, we will also introduce how to resist the SPA attacks using the new algorithm.

Where A is the Affine coordinates, the J is Jacobian, J^c is Chudnovsky Jacobian, J^m is modified Jacobian, the P is Projective coordinate.

5.1 Scalar Multiplication Analysis

In Table 3, We compared our algorithm with others, such as NAF, 4-NAF and Double-and-add and so on on mixed coordinates. In Table 4, we compared several algorithms which used the NewADD. Table 5 shown the chain length of several algorithms.

In Section 3, we know that there is unnecessary to calculate the cost of (2P+Q). What we just need to do is to calculate the remaining addition when 1 appears. So the average cost is 4M+1.5S, the final multiplications are 832. However, the number of occurrences of 1 dependent on the specific number, in order to illustrate the advantages of the TFTAC algorithm clearly, we choose the largest number to do comparison and the cost is 8M+3S, the final field multiplications is 1664. On mixed coordinates, TFTAC is faster at least 6.5 % than NAF, 20.9 % than double-and-add, 12.7 % than GRAC-258 , 21 % DFAC-160. Although slower 3.8 % than 4-NAF, most of the TFT numbers are faster than it.

Table 4 shown the comparisons algorithm using the NewADD algorithm. From 27.9 % more than the Fibonacci-and-add, 20.3 % more than the Signed Fib-and-add and 15.1 % more than the Window Fib-and-add. So we can see the new algorithm significantly reduces the addition computation.

Table 5 shown the chain length comparison of the various algorithms with the 160-bits, the A-Z representation chain length is 160 and it is same as the binary counterpart. And compared with other algorithm chain length the A-Z representation chain length is quite shorter than

Table 2: The cost of various mixed coordinate

doubling	
operation	**costs**
$2A = J$	$2[M] + 4[S]$
$2J^m = J$	$3[M] + 4[S]$
$2A = J^m$	$3[M] + 4[S]$
$2J^m$	$3[M] + 5[S]$
$2J^m = J^c$	$4[M] + 4[S]$
$2J$	$4[M] + 6[S]$
$2J^c$	$5[M] + 6[S]$
$2P$	$7[M] + 5[S]$
addition	
operation	**costs**
$P + P$	$12[M] + 2[S]$
$J^m + J^m$	$13[M] + 6[S]$
$J + A$	$8[M] + 3[S]$
$J^m + A = J^m$	$9[M] + 5[S]$
$J^m + A = J$	$8[M] + 3[S]$
$J^c + J = J$	$11[M] + 3[S]$
$J^c + J^c = J^m$	$11[M] + 4[S]$
$J^c + J^c = J$	$10[M] + 2[S]$
$J^c + J^c$	$11[M] + 3[S]$
$J^c + A = J^m$	$8[M] + 4[S]$
$J^c + A = J^c$	$8[M] + 3[S]$
$J + A = J^m$	$9[M] + 5[S]$
$A + A = J^m$	$5[M] + 4[S]$
$A + A = J^c$	$5[M] + 3[S]$
$J + J$	$12[M] + 4[S]$
$J^c + J = J^m$	$12[M] + 5[S]$
$J^m + J^c = J^m$	$12[M] + 5[S]$

Table 3: Comparison the classical algorithm for the 160-bits

Algorithm	**Costs**
4-NAF	1600
NAF	1780
Double-and-add	2104
GRAC-258	1907
DFAC-160	2016

Table 4: Comparisons with different algorithm using the NewADD

Algorithm	**Chain length**
Fibonacci-and-add	2311
Signed Fib-and-add	2088
Window Fib-and-add	1960
TFTAC	1664

Table 5: Comparisons with the chain length for 160-bit

Algorithm	**Chain length**
Fibonacci-and-add	358
Signed Fib-and-add	322
Window Fib-and-add	292
Binary representation	160
Zenkendorf representation	230
TFTAC	160

others. It is 55 % shorter than the Fibonacci-and-add, 50.3 % shorter than Signed Fib-and-add, 45 % shorter than Window Fib-and-add, and 44 % shorter than Zenkendorf representation.

5.2 SPA Analysis

SPA is a technology which is a direct interpretation of energy consumption measured value. The system consumption of energy is different that mainly depending on the instructions executed by the microprocessor. When the microprocessor operation performed at different part of the encryption algorithm, some of the energy consumption of the system is very obvious. With this feature, the attacker can distinguish a single instruction to achieve the purpose of breaking the algorithm.

We already know that the attacker obtained the key information by observing the energy curve changing. Therefore we can utilize the method of fixed sequence to resist SPA. Our algorithm based on the Transformed Fibonacci sequence and it is a fixed sequence, therefore it can resist the SPA as well.

6 Conclusion

In this paper we proposed a new algorithm TFTAC which is based on the Fibonacci deformation sequence, the new algorithm combined with the advantages of NewADD and through the method of generating tables reduced the cost of scalar multiplication significantly. This method utilized the space exchange for time achieve the purpose of saving resources effectively. Among them the multiplication is less at least 12.7 % than others, the chain length also reduced from 45 % to 55 %. In addition, the algorithm against the SPA as well.

Acknowledgments

The support of NSFC (National Natural Science Foundation of China, No.61272525), Jiangxi Natural Science Foundation (No.2009GQN0094Scientific Research program funded by Shaanxi Provincial Education Department (No.2013JK1097) and the Youth Foundation of Xi'an University of Posts and Telecommunications (No.103-0454) are gratefully acknowledged.

References

[1] A. Byrne, N. Meloni, F. Crowe, W. P. Marnane, A. Tisserand, and E. M. Popovici, "Spa resistant elliptic curve cryptosystem using addition chains," in *Fourth International Conference on Information Technology (ITNG'07)*, pp. 995–1000, 2007.

[2] K. Chatterjee, A. De, and D. Gupta, "Mutual authentication protocol using hyperelliptic curve cryptosystem in constrained devices," *International Journal of Network Security*, vol. 15, no. 1, pp. 9–15, 2013.

[3] D. V. Chudnosky and G. V. Chudnovsky, "Sequences of numbers generated by addition in formal groups and new primarily and factorization tests," *Advances in Applied Mathematics*, vol. 7, no. 4, pp. 385–434, 1986.

[4] H. Cohen, A. Miyaji, and T. Ono, "Efficient elliptic curve exponentiation using mixed coordinates," *Lecture Notes in Computing Science*, vol. 1514, pp. 51–65, 1998.

[5] S. Ezzouak, M. E. Amrani, and A. Azizi, "Improving miller's algorithm using the naf and the window NAF," *Lecture Notes in Computer Science*, vol. 7853, pp. 279–283, 2013.

[6] G. Hou and Z. Wang, "A robust and efficient remote authentication scheme from elliptic curve cryptosystem," *International Journal of Network Security*, vol. 19, no. 6, pp. 904–911, 2017.

[7] F. Jia and D. Xie, "A unified method based on spa and timing attacks on the improved RSA," *China Communications*, vol. 13, no. 4, 2016.

[8] N. Koblitz, "Elliptic curve cryptosystems," *Mathematics of Computation*, vol. 48, no. 177, pp. 203–209, 1987.

[9] F. Laguillaumie and D. Vergnaud, "Time-selective convertible undeniable signatures with short conversion receipts," *Information Sciences*, vol. 180, no. 12, pp. 2458–2475, 2010.

[10] Z. X. Lai and Z. J. Zhang, "Research on elliptic curve bottom level algorithm in jacobian coordinate system," *Bulletin of Science and Technology*, vol. 31, no. 10, pp. 244–248, 2015.

[11] T. C. Lin, "Algorithms on elliptic curves over fields of characteristic two with non-adjacent forms," *International Journal of Network Security*, vol. 9, no. 2, pp. 117–120, 2009.

[12] S. G. Liu and Y. P. Hu, "Fast and secure elliptic curve scalar multiplication algorithm based on special addition chains," *Journal of Southeast University*, vol. 24, no. 1, pp. 29–32, 2008.

[13] S. G. Liu, G. L. Qi, and X. A. Wang, "Fast and secure elliptic curve scalar multiplication algorithm based on a kind of deformed fibonacci-type series," in *10th International Conference on P2P, Parallel, Grid, Cloud and Internet Computing (3PGCIC'15)*, pp. 398–402, 2015.

[14] N. Meloni, "New point addition formulae for ecc applications," *Lecture Notes in Computing Science*, vol. 4547, 2007.

[15] V. S. Miller, "Use of elliptic curves in cryptography," *Lecture Notes in Computer Science*, vol. 218, pp. 417–426, 1986.

[16] V. S. Naresh and N. V. E. S. Murthy, "Elliptic curve based dynamic contributory group key agreement protocol for secure group communication over ad-hoc networks," *International Journal of Network Security*, vol. 17, no. 3, pp. 328–339, 2015.

[17] B. Rashidi, S. M. Sayedi, and R. R. Farashahi, "Efficient and low-complexity hardware architecture of gaussian normal basis multiplication over GF(2m) for elliptic curve cryptosystems," *IET Circuits, Devices and Systems*, vol. 11, no. 2, 2017.

[18] Reddy and E. Kesavulu, "Elliptic curve cryptosystems and side-channel attacks," *International Journal of Network Security*, vol. 12, no. 3, pp. 151–158, 2011.

[19] Y. Sakemi, T. Izu, and M. Shirase, "Faster scalar multiplication for elliptic curve cryptosystems," in *16th International Conference on Network-Based Information Systems*, pp. 523–527, Sep. 2013.

[20] A. Thomas and M. Manuel, "Embedment of montgomery algorithm on elliptic curve cryptography over rsa public key cryptography," *International Conference on Emerging Trends in Engineering, Science and Technology*, vol. 24, pp. 911–917, 2016.

[21] M. Toorani and A. A. B. Shirazi, "Cryptanalysis of an elliptic curve-based signcryption scheme," *International Journal of Network Security*, vol. 10, no. 1, pp. 51–56, 2010.

[22] S. F. Tzeng, M. S. Hwang, "Digital signature with message recovery and its variants based on elliptic curve discrete logarithm problem", *Computer Standards & Interfaces*, vol. 26, no. 2, pp. 61–71, 2004.

[23] D. Yong, Y. F. Hong, W. T. Wang, Y. Y. Zhou, and X. Y. Zhao, "Speeding scalar multiplication of elliptic curve over GF(2)," *International Journal of Network Security*, vol. 11, no. 10, pp. 70–77, 2010.

Biography

Shuang-Gen Liu was born in 1979, associate professor. He graduated from Xidian University in 2008 with a major in cryptography, PhD, a member of the Chinese Institute of computer science, and a member of the Chinese code society.

Xue-Jing Sun is a graduate student of Xi'an University of post and telecommunications. She is mainly engaged in the research of elliptic curve cryptosystem.