Fast and Secure Scalar Multiplication on Elliptic Curves

No Author Given

No Institute Given

Abstract. This paper proposes a new fast and secure point multiplication algorithm for elliptic curves over finite fields. This new scalar multiplication resists against side–channel attacks based on simple power analysis. The algorithm is based on a particular kind of addition-subtraction chain known as *Lucas addition-subtraction chains*. Lucas addition-subtraction chains have been proven to give minimal length chains for infinitely many integers.

Keywords: addition chain; addition-subtraction chain; Lucas chains; Lucas addition-subtraction chains; elliptic curve cryptography; scalar multiplication; elliptic curve system of coordinates; side-channel attacks.

1 Introduction

Elliptic curve cryptography has received much attention because of its short key-length as well as its theoretical robustness. Recall the security of elliptic curve cryptography is based upon the Discrete Logarithm Problem, for which no subexponential attacks are known if parameters are carefully chosen.

Given a point P on an elliptic curve over a finite field, computing the scalar multiple kP is central to the actual implementation of elliptic curve cryptography. Various methods have been proposed to speed up and secure this computation. Exponentiation algorithms have been shown to be vulnerable to side-channel analysis, where an attacker observes the power consumption [14]. This attack is known as *Simple Power Analysis* (SPA). There are several algorithms that have been proposed in the literature [14, 15, 17, 19, 25] to resist against SPA. It should be noted that differential side-channel analysis will not be considered in this paper.

In this work, we propose a new fast and secure point multiplication algorithm, which resists SPA. The algorithm is based upon a particular kind of addition-subtraction chain which are known as *Lucas addition-subtraction chains*. Addition-subtraction chains and Lucas chains have both been studied in connection with speeding up point multiplications [1, 6–8, 11, 12, 17, 26]. The Lucas addition-subtraction algorithm we propose is much simpler and as fast as all the known algorithms that resists SPA.

This paper is organized as follows. In the next section, we provide a brief background on elliptic curves and review Lucas addition-subtraction chains. In Section 3, we present the new scalar multiplication algorithm based on Lucas addition-subtraction chains and show it resists SPA. In Section 4, we compare our proposed algorithm with the classical double-and-add, and NAF scalar multiplication algorithms. Finally, we conclude in Section 5.

2 Background

In this section, we first give a brief overview of addition on elliptic curves. For more details, the reader should consult [7]. We then review Lucas addition-subtraction chains [18].

2.1 Elliptic curves

Definition 1. *An elliptic curve E over a finite field K is given by an equation*

$$E(K): \quad y^2 + a_1xy + a_3y = x^3 + a_2x^2 + a_4x + a_6, \tag{1}$$

where $a_1,\ a_2,\ a_3,\ a_4,\ a_6 \in K$ are such that for each point $(x,\ y)$ on E, the partial derivatives do not simultaneously vanish.

In practice, the equation for an elliptic curve can be simplified into one of the following forms, depending on the field K.

$$y^2 = x^3 + ax + b,$$

where $a,\ b \in K$, $4a^3 + 27b^2 \neq 0$, and K has characteristic greater than 3.

$$y^2 = x^3 + ax^2 + b$$

where $a,\ b \in K$, $a^3b \neq 0$, and K has characteristic 3.

$$y^2 + xy = x^3 + ax^2 + b$$

where $a,\ b \in K$, $b \neq 0$, and the characteristic of K is 2.

The set $E(K)$ of the rational points of an elliptic curve E (defined over K) is an abelian group where the identity element is a special point $\mathcal{O}$, called the point at infinity.

2.2 The addition law

The set of points of an elliptic curve forms a group under a certain addition rule. We now give this rule explicitly. Let E be as in (1), and let $P = (x_1,\ y_1),\ Q = (x_2,\ y_2)$ be two points of E. The negative of the point P is given by

$$-P = (x_1,\ -y_1 - a_1x_1 - a_3),$$

and their sum $P + Q = (x_3,\ y_3)$ is defined as follows:

$$x_3 = \lambda^2 + a_1\lambda - a_2 - x_1 - x_2,$$

$$y_3 = \lambda(x_1 - x_3) - y_1 - a_1x_3 - a_3,$$

where λ is:

$$\lambda = \begin{cases} \frac{y_1 - y_2}{x_1 - x_2}, & \text{if } \ P \neq \pm Q, \\ \\ \frac{3{x_1}^2 + 2a_2x_1 + a_4 - a_1y_1}{2y_1 + a_1x_1 + a_3}, & \text{if } \ P = Q. \end{cases}$$

Points on an elliptic curve can be represented in several coordinate systems, such as affine coordinates ($\mathcal{A}$), Jacobian coordinates ($\mathcal{J}$) or projective coordinates ($\mathcal{P}$). The addition law we presented above is for affine coordinates. Coordinate systems like $\mathcal{J}$ and $\mathcal{P}$ allow us to compute the sum while avoiding costly field inversions.

2.3 Lucas addition-subtraction chains

Before we give the definition of Lucas addition-subtraction chains, we first define addition chains, Lucas addition chains, and addition-subtraction chains. As can be inferred from their name, Lucas addition-subtraction chains combine these three types of chains. For more details on these various types of chains, see [1–5, 17, 21, 18].

Definition 2. *Let n be an integer. A sequence $c = \{1 = a_0,\ a_1, \ldots, a_l = n\}$ is called an addition chain for n if and only if for each $a_i \in c$, there exists j, k with $0 \leq j, k < i$ such that:*

$$a_i = a_j + a_k.$$

Example 1 *The sequence $\{1,\ 2,\ 3,\ 5,\ 7,\ 9,\ 14,\ 19\}$ is an addition chain for* 19.

A subset of addition chains are the Lucas addition chains.

Definition 3. *An addition chain $c = \{a_0, a_1, \ldots, a_l\}$ is a Lucas addition chain if and only if:*

$$\textit{if } a_i = a_j + a_k \ \textit{ for some } 0 \leq i, j, k \leq l, \ \textit{ then } a_j = a_k \ \textit{ or } |a_j - a_k| \in c.$$

Example 2 *The sequence $\{1,\ 2,\ 3,\ 5,\ 7,\ 9,\ 14,\ 19\}$ is a Lucas addition chain for* 19.

Notice that, in this example, 14 is obtained by $7 + 7$ and not $9 + 5$.

Definition 4. *A sequence $c = \{1 = a_0,\ a_1,\ \ldots,\ a_l = n\}$ is called an addition-subtraction chain for an integer n if and only if for each $a_i \in c$, then $a_i > 0$ and there exists j, k with $0 \leq j, k < i$ such that*

$$a_i = a_j + a_k \text{ or } a_i = a_j - a_k.$$

Example 3 *The sequence* $\{1,\ 2,\ 4,\ 8,\ 16,\ 24,\ 22\}$ *is an addition-subtraction chain for* 22.

Finally, we come to Lucas addition-subtraction chains.

Definition 5. *Let* n *be a integer. A Lucas addition-subtraction chain for* n *is a sequence* $c = \{a_0 = 1,\ a_1,\ \ldots,\ a_l = n\}$ *such that for each* $a_i \in c$*, there exists* j, k *with* $0 \leq j,\ k < i$ *satisfying*

$$a_i = \begin{cases} a_j + a_k & \text{and } |a_j - a_k| \in c \cup \{0\}, \\ & \text{or} \\ a_j + 1, & \\ & \text{or} \\ a_j - a_k. & \end{cases}$$

Example 4 *Let* F_k *be the* k^{th} *Fibonacci number. Then* $\{F_1,\ F_2,\ \ldots,\ F_l\}$ *is a Lucas addition-subtraction chain for* F_l*. Here*

$$F_k = \begin{cases} 1, & \text{for } k = 0, 1, \\ F_{k-1} + F_{k-2}, & \text{for } k \geq 2. \end{cases}$$

We can see that $F_k - F_{k-1} = F_{k-2}$ *for all* $k > 2$.

Example 5 $\{1,\ 2,\ 3,\ 5,\ 10,\ 20,\ 19\}$ *is a Lucas addition-subtraction chain for* 19.

Example 6 $\{1,\ 2,\ 3,\ 4,\ 7,\ 10,\ 11,\ 9\}$ *is a Lucas addition-subtraction chain for* 9

Throughout the remainder of the paper, we will use the shorthand LASC to denote a Lucas addition-subtraction chain. We now give a simple way to create short LASCs with the following theorem.

Theorem 7. *Let* n *be an integer. A Lucas addition-subtraction chain for* n *can be obtained recursively in the following way:*

1. *If* n *is even, then append* n *to a chain for* $\frac{n}{2}$.
2. *If* $n \equiv 1 \bmod 4$*, then append* n *to a chain for* $n - 1$.
3. *If* $n \equiv 3 \bmod 4$*, then append* n *to a chain for* $n + 1$.

Proof. We need only show that each of the three steps given above satisfy the criteria for LASCs. If c is a LASC for $\frac{n}{2}$, then appending n to c will still be a valid LASC as $n = \frac{n}{2} + \frac{n}{2}$. If instead c is a valid LASC for $n - 1$, then the definition for LASCs allows for adding 1 to an element of c, so we can append n to c. Finally, if c is a LASC for $n + 1$, then always we have $1 \in c$, and we may append $(n + 1) - 1 = n$ to c.

For ease of notation, we label the steps in Theorem 7 as DBL, ADD, and SUB. Notice a DBL step is a doubling of the previous final element of the chain, an ADD step is an addition of 1 to the previous final element of the chain, and a SUB step is a subtraction by 1. We illustrate the theorem in the next two examples.

Example 8 *1. A Lucas addition-subtraction chain for* 124 *can be obtained as follows:*

$$\begin{aligned} \mathbf{124} &= 62 \cdot 2, \\ \mathbf{62} &= 31 \cdot 2, \\ \mathbf{31} &= 32 - 1, \\ 32 &= \mathbf{16} \cdot 2, \\ 16 &= \mathbf{8} \cdot 2, \\ &\vdots \\ \mathbf{2} &= 1 \cdot 2. \end{aligned}$$

The corresponding Lucas addition-subtraction chain is:

$$\{1,\ 2,\ 4,\ 8,\ 16,\ 32,\ 31,\ 62,\ 124\}.$$

2. A Lucas addition-subtraction chain for 242 *can be obtained as follows:*

$$\begin{aligned} \mathbf{242} &= 121 \cdot 2, \\ \mathbf{121} &= 120 + 1, \\ \mathbf{120} &= 60 \cdot 2, \\ \mathbf{60} &= 30 \cdot 2, \\ 30 &= \mathbf{15} \cdot 2, \\ \mathbf{15} &= 16 - 1, \\ 16 &= \mathbf{8} \cdot 2, \\ &\vdots \\ \mathbf{2} &= 1 \cdot 2. \end{aligned}$$

The corresponding Lucas addition-subtraction chain is:

$$\{1,\ 2,\ 4,\ 8,\ 16,\ 15,\ 30,\ 60,\ 120,\ 121,\ 242\}.$$

We note that there are other approaches to finding Lucas addition-subtraction chains. However, this algorithm is significant because of its simplicity and the short length of the chains produced. As seen above, each step is either a doubling, or an addition or subtraction by 1. Whenever we perform an ADD or SUB step, notice that we come to a value which is $\equiv 0 \bmod 4$, and so we can always then perform two successive doubling steps. This makes the computation very efficient. We now present this method in algorithmic form.

Algorithm 1 LASC(n)

```
Input: $n$ : integer
Output: $c$ : a Lucas addition-subtraction chain for $n$
$c = \{n\}$
if $n$ is even then
    add $LASC(n/2)$ to the chain $c$
else
    if $n \equiv 1 \bmod 4$ then
        add $LASC(n-1)$ to the chain $c$
    else
        if $n \equiv 3 \bmod 4$ then
            add $LASC(n+1)$ to the chain $c$.
        end if
    end if
end if
return c
```

3 The new scalar multiplication algorithm

As mentioned before, one of the key operations in the implementation of elliptic curve cryptography is computing scalar multiples of points. Let P be a point on an elliptic curve, and k be the scalar we wish to use. The following algorithm computes kP by constructing a Lucas addition-subtraction chain for k.

Algorithm 2 scalarMultiplication(k, P)

```
Input: $k$ : integer, $P$: a point of an elliptic curve $E$
Output: $kP$ : a point of $E$
if $k = 2a$ then
    return $Double(LascExp(a, P))$
else
    if $k = 2^m - 1$ then
        return $LascExp(k+1, P) - P$
    else
        if $\lfloor k/2 \rfloor$ even then
            return $Double(LascExp(\lfloor k/2 \rfloor, P)) + P$
        else
            return $Double(LascExp(\lfloor k/2 \rfloor + 1, P)) - P$
        end if
    end if
end if
```

The algorithm is at the worst case $2/3(\lambda(k))DBL + (\lambda(k)/3)ADD$ which is the average cost of the double-and-add scalar multiplication, where $\lambda(k) = \lfloor \log_2(k) \rfloor$.

3.1 Side-Channel Analysis

Side-channel attacks[16, 24] are any attacks based on *side-channel information*. Side-channel information refers to information that can be gained from the physical encryption device. This includes, for example, timing information, power consumption, and electromagnetic leaks. In particular, since the computational cost of addition and doubling of points on elliptic curves are distinguishable by measuring the power consumption, an attacker can use SPA to exploit this information. That is, addition and doubling are distinguishable by one–tile measurement of power consumption, whereas addition and subtraction are indistinguishable.

Several counter-measures have been proposed against these attacks [14], [19], [20], [21]. In this work, the new scalar multiplication avoids simple power analysis (SPA) by taking advantage of the indistinguishability of addition and subtraction. We assume that an attacker can use the power consumption to determine the sequence of addition and the doubling steps of our algorithm. However, this will not produce enough information about the binary expansion of the scalar k. If the attacker knows that there are m addition steps, then if we have used Algorithm 1 there are roughly 2^m possibilities for k. This follows because each addition step could be either an ADD or SUB step. The attacker cannot distinguish between any two possible candidates for k.

We illustrate this concept with an example. Suppose an SPA attack yields the sequence of doublings and additions used to compute a Lucas addition-subtraction chain for an integer n. We list such a sequence in the second column below. The third and fourth columns show how knowing this sequence does not determine n.

Step	Operation	Chain for **1917**	Chain for **2171**
1	4 DBL	$\{1,\ 2,\ 4,\ 8,\ 16\}$	$\{1,\ 2,\ 4,\ 8,\ 16\}$
2	1 ADD	$15 = (1111)_2$	$17 = (10001)_2$
3	5 DBL	$\{30,\ 60,\ 120,\ 240,\ 480\}$	$\{34,\ 68,\ 136,\ 272,\ 544\}$
4	1 ADD	$479 = (111011111)_2$	$543 = (1000011111)_2$
5	2 DBL	$\{958,\ 1916\}$	$\{1086,\ 2172\}$
6	1 ADD	**1917**	**2171**

Fig. 1. xxx

4 Comparisons with classical algorithms

In this section, our new proposed scalar multiplication algorithm will be compared to the classic double-and-add (or binary) and the NAF methods. We implemented each method with 1000000 random 160-bit primes, and display the average number of addition and doubling steps required. The next three tables do the same for 384-bit , 512-bit, and 1024-bit integers.

This table is for scalars of 160–bits.

Method	binary	NAF	LASC
Addition	88	52	55
Doubling	159	160	156
Total	247	212	211

Fig. 2. xxx

This table is for scalars of 384–bits.

Method	binary	NAF	LASC
Addition	202	117	130
Doubling	383	384	384
Total	585	501	524

Fig. 3. xxx

This table is for scalars of 512–bits.

Method	binary	NAF	LASC
Addition	265	168	173
Doubling	511	512	511
Total	776	680	684

Fig. 4. xxx

This table is for scalars of 1024–bits.

From the tables we see that Algorithm 2 (which uses LASCS) is comparable in efficiency to the NAF method, while both are more efficient than the classical binary double–and–add technique. We also observe that in comparison to the NAF method, Algorithm 2 requires much fewer doublings, at the trade-off of requiring more additions and subtractions. The greater number of addition steps is actually helpful, however, since as discussed previously, it provides resistance to SPA.

We further analyzed Algorithm 2 to determine the distribution of ADD versus SUB steps occurring in the addition steps. When we have roughly the same

Method	binary	NAF	LASC
Addition	530	350	455
Doubling	1023	1024	912
Total	1553	1374	1367

Fig. 5. xxx

number of additions as subtractions, it decreases the chance of an attacker finding the right value for k. In each table we display the average number of DBL, ADD, and SUB steps required in Algorithm 1.

Method	For 1000000 random prime numbers of 160–bits
ADD	28.06 additions (+1's)
DBL	159.33 doublings
SUB	26.73 subtractions (-1's)

Fig. 6. xxx

Method	For 1000000 random prime numbers of 256–bits
ADD	44.05 additions (+1's)
DBL	255.33 doublings
SUB	42.73 subtractions (-1's)

Fig. 7. xxx

A simple explanation for the observation that we have roughly the same number of ADD and SUB steps is the following. We expect the odd values in the chain computed by Algorithm 1 to be uniformly distributed mod 4. That is, we expect about half of them to be $\equiv 1$ mod 4, while half are $\equiv 3$ mod 4. From this it follows that the number of additions and subtractions (whch are just +1's and -1's) to be approximately equal. The data supports this conclusion.

5 Conclusion

This paper has presented a new algorithm to compute scalar multiplication on elliptic curves. Our new algorithm is fast, much simpler, and as secure as previously known algorithms in protecting against SPA. The key tool used for the algorithm is Lucas addition-subtraction chains. Generaly, these chains have shorter length than the traditional Lucas addition chains [1, 18] and have the

Method	For 500000 random prime numbers of 384–bits
ADD	65.06 additions (+1's)
DBL	383.33 doublings
SUB	65.73 subtractions (-1's)

Fig. 8. xxx

Method	For 250000 random prime numbers of 512–bits
ADD	86.72 additions (+1's)
DBL	511.33 doublings
SUB	85.39 subtractions (-1's)

Fig. 9. xxx

same properties. We leave it as future work to examine the potential use of Lucas addition-subtraction chains in the elliptic curve factoring technique [27–29].

References

1. Peter L. Montgomery. *Evaluating recurrences of form* $X_{m+n} = f(X_m,\ X_n,\ X_{m-n})$ *via Lucas Chains* January 1992.
2. H. Volger. *Some results on addition-subtraction chains Information Processing Letters* **20** (3) (8 April 1985) 155-160.
3. F. Morrain, J. Olivos Speeding up the computation on an elliptic curve using addition-subtaction chains. *Informatique théorique et applications* **24**, (6) (1990) 531-543.
4. C. Wang, C. Lin and C. Chang, *A method for computing Lucas sequences . Computers and Mathematics with Applications* **38** (1999) 187-196.
5. M. Mignotte, A. Tall, *A note on addition chains. International Journal of Algebra* **5** (6) (2011) 269 - 274.
6. K. Koyama, Y Tsuruoka, *Speeding elliptic cryptosystems using a signed binary window method. Advances in cryptology* **740** (1992) 345 - 357.
7. J. H. Silverman, *The arithmetic of elliptic curves. Springer-verlag* (1986).
8. R. M. Avanzi, H. Cohen, C. Doche, G. Frey, T. Lange, K. Nguyen, and F. Vercauteren. Handbook of Elliptic and Hyperelliptic Curve Cryptography. CRC Press, 2005.
9. Duc-Phong Le. *Fast quadrupling of a point in elliptic curve cryptography preprint Information Processing Letters* (23 March 2011).
10. N. Koblitz, *Elliptic curve cryptosystems. Mathematics of computation* **48** (117) (1987).
11. CY. Sakari, K. Sakurai, *Efficient scalar multiplication on elliptic curve with direct computations of several doublings. IEICE Transactions Fundamentals* **E84-A(1)** (2001) 120-129.
12. M. Ciet, M. Joye, K. Lauter, P. L. Montgomery, *Trading Inversions for Multiplications in elliptic curve cryptography.*

13. D. Hankerson, A. J. Menezes, S. Vanstone, *Guide to elliptic curve cryptography. Springer-Verlag New York, Inc, Secawcus NJ, USA* (2003).
14. B. Chevallier-Mames, M. Ciet and M. Joye. *Low-cost solutions for preventing simple side-channel analysis: Side-channel atomicity IEEE Transactions on Computers* **53** (6) (2004) 760-768.
15. M. Hebadou, P. Pinel and L. Beneteau. *A comb method to render ECC resistant against side channel attacks Report 2004/342, Cryptology ePrint Archive, 2004* **http;//eprint.iacr.org**.
16. P. Kocher, J. Jaffe and B. Jun. *Differential power analysis M.J. Wiener, editor, Advances in Cryptology – CRYPTO '99, volume 1666 of Lecture Notes in Computer Science* **(Springer 1999)** 388-397.
17. D.M. Gordon. *A survey of fast exponentiation methods Journal of Algorithms* **27** (1) (1998) 129-146.
18. Tall, A., A generalization of Lucas addition chains, Cryptology ePrint Archive, Report 2011/378, 2011. Available at http://eprint.iacr.org/2011/378.pdf.
19. M. Joye and J.-J. Quisquater, Hessian elliptic curves and side-channel attacks. Cryptographic Hardware and Embedded Systems-CHESS 2001, volume 2162 of Lecture Notes in Computer Science. Springer-Verlag, 2001.
20. V. Dimitrov, L. Imbert, and P. K. Mishra. Efficient and secure Elliptic Curve Point Multiplication Using Double-Base Chains. ASIACRYPT 2005, volume 3788 of Lecture Notes in Computer Science, pages 59-78, 2005.
21. Alfred Brauer, On addition chains, Bulletin of the American Mathematical Society 45 (1939), 736739. ISSN 02730979. MR 1,40a.
22. V. Dimitrov, L. Imbert, P. K. Mishra., Efficient and secure elliptic curve point multiplication using double-base chains, Advances in CryptologyASIACRYPT 2005, 11th International Con- ference on the Theory and Application of Cryptology and Information Security, Chennai, India, December 48, 2005, Proceedings, Lecture Notes in Computer Science, 3788, Springer, Berlin, 2005, 5978.
23. D. J. Bernstein, T. Lange, Faster addition and doubling on elliptic curves, Advances in cryptologyASIACRYPT 2007, Lecture Notes in Computer Science, 4833, Springer, (2007), 2950.
24. P. C. Kocher. Timing attacks on implementations of Di?e-Hellman, RSA, DSS, and other systems. In N. Koblitz, editor, Advances in Cryptology - CRYPTO96, volume 1109 of Lecture Notes in Computer Science, pages 104-113. Springer-Verlag, Aug. 1996.
25. V. Dimitrov, L. Imbert, and P. K. Mishra. Efficient and secure Elliptic Curve Point Multiplication Using Double-Base Chains. ASIACRYPT 2005, volume 3788 of Lecture Notes in Computer Science, pages 59-78, 2005.
26. F. Morain, J. Olivos. Speeding up the computations on an elliptic curve using addition-subtraction chains, RAIRO Informatique thoretique et application 24, pp. 531-543 (1990).
27. Chelli, I., Fully Determinist ECM, Available at http://caramel.loria.fr/sem-slides/200909251030.pdf.
28. Kruppa, A., Factoring into Large primes with $p-1$, $p+1$, and ECM, Available at http://cado.gforge.inria.fr/workshop/slides/kruppa.pdf.
29. Kruppa, A., A software implementation of ECM for NFS, Available at http://hal.archives-ouvertes.fr/docs/00/41/90/94/PDF/RR-7041.pdf.