A New Method for Speeding Up Arithmetic on Elliptic Curves over Binary Fields

Kwang Ho Kim and So In Kim

Department of Algebra, Institute of Mathematics
National Academy of Sciences, Pyongyang, D. P. R. of Korea
kimkhhj1980@yahoo.com.cn

Abstract: Now, It is believed that the best costs of a point doubling and addition on elliptic curves over binary fields are $4M+5S$ (namely, four finite field multiplications and five field squarings) and $8M+5S$, respectively. In this paper we reduce the costs to less than $3M+3S$ and $8M+1S$, respectively, by using a new projective coordinates we call PL-coordinates and rewriting the point doubling formula. Combining some programming skills, the method can speed up a elliptic curve scalar multiplication by about 15～20 percent in practice.

Keywords: ECC(elliptic curve cryptosystem), binary field, point doubling, point addition.

1. Introduction

As well known, elliptic curve cryptosystem(ECC) is recognized to be strongest among modern cryptosystems.

Nevertheless, it seems the amount of operations needed for practice to be yet large.

So, the investigation for speeding up the arithmetic on elliptic curves has been steadily working since ECC was appeared. We call the point addition(Ep) and the point doubling(Ed) on the elliptic curve as elliptic operations.

In the case of characteristic 2, the elliptic operation algorithm which was introduced in international standard in 1999, consums $15M+5S$, i.e. 15 multiplications and 5 squarings in finite field, per Ep.(cf. [1])

After a while, a new algorithm has occurred, which required $14M+6S$ per Ep.(cf. [2])

Also, a new Ep-algorithm was announced which required $13M+4S$, in 2000.(cf. [3])

But all these algorithms have the cost of at most $9M+4S$, in the case of $a=1$ and $Z=1$ which is most important in the cryptographic practice.(cf. [1,4])

After them, in 2002, an Ep algorithm for the case $a=1$ and $Z=1$ was proposed, which required $8M+5S$.(cf. [4]) Since M is more expensive than S, it turns out that this is improvement of [2].

On the other hand, the Ed algorithm had not been improved over $4M+5S$ which is original, during all these time.

In sum, the best costs for Ed and Ep attained by now are $4M+5S$, $8M+5S$ respectively.(cf. [1-5])

In this paper we propose a new method to reduce the costs to less than $3M+3S$ and $8M+1S$, respectively. Combining some programming skills, the method gives a speed-enhancement in scalar multiplication, by 15~20 percent.

2. Algorithm

In this paper we suppose that elements of the binary field $GF(2^n)$ are represented in polynomial basis, because the representation generally gives faster implementation of arithmetic on the field than normal basis representation.

Lets denote the modulation (by the definition polynomial) of an polynomial a by $[a]$ and polynomial multiplication of $a \in GF(2^n)$ and $b \in GF(2^n)$ (without the modulation), by $a*b$.

Then it is obvious that the cost of a field multiplication is equal to the sum of the costs for those two operations.

Since the binary field addition is so cheap that is always negligible in considerations for implementation, we think that the cost for a modulation $[a]$ is approximately equal to the cost of a field squaring.

In below estimations we will be keeping in mind them.

There are many type of projective coordinates used for speeding up the scalar multiplication on elliptic curves.(cf. [1])

We employ a 4-D projective coordinates (X,Y,Z,T), corresponding to an affine representation (x,y) by: $x = X/Z, y = [Y]/T, T = Z^2$

to represent points on the elliptic curve. In this paper we will call this projective coordinates as PL-coordinates.

As shown, the difference of the coordinates from Lopez-Dahab coordinates(cf. [2]) is that an expansion variable T is added and the variant Y is without modulation.

Lets consider non-supersingular elliptic curve: $y^2 + xy = x^3 + x^2 + b$, where $b \in GF(2^n)'$,

over $GF(2^n)$. It is known that this type of curves covers the half of ordinary elliptic curves over $GF(2^n)$.(cf. [4])

In the PL-coordinates, the curve equation is represented as: $[Y]^2 + XYZ = X^3Z + X^2Z^2 + bZ^4$.

[THEOREM 1] The Ed (X_1,Y_1,Z_1,T_1) of a point (X,Y,Z,T) in the PL-coordinates is obtained by following.

Order	Operation	Cost
1	$A=X^2, B=[Y]^2$	$3S$
2	$Z_1=TA, \quad T_1=Z_1^2$	$1M+1S$
3	$X_1=[A*A+b*T^2]$	$1M+1S$
4	$Y_1=B*(B+X_1+Z_1)+b*T_1+T_1$	$2M-2S$
Total		$4M+3S$

(proof)

By the definition of elliptic curve group law, the doubling point $(x_{1,}y_1)$ of affine point (x,y) is given by:

$$\lambda = y/x + x\,,$$

$$x_1 = \lambda^2 + \lambda + 1\,,$$

$$y_1 = x_1(\lambda+1) + x^2\,.$$

We can easily derive from the curve equation that

$$y^2 + b = x^2(\lambda+1)\,,$$

$$x^6 = (y^2 + xy + x^2 + b)^2\,.$$

Using them, we obtain that

$$x_1 = y^2/x^2 + x^2 + (y^2+b)/x^2 = x^2 + b/x^2\,,$$

$$y_1 = (x^4+b)(y^2+b)/x^4 + x^2 = [y^2(y^2+x^2+x^4+b) + x^4(b+1)]/x^4\,.$$

In the PL-coordinates, above expressions are equivalent to:

$$x_1 = (X^4 + bZ^4)/X^2Z^2\,,$$

$$y_1 = ([Y]^2([Y]^2 + X^2Z^2 + X^4 + bZ^4) + X^4Z^4(b+1))/X^4Z^4\,.$$

Then, the algorithm is obtained, putting $Z_1 = X^2Z^2$. □

It is noticed that two among four field multiplications in above algorithm are by a fixed constant b and without modulation.

The cost of a constant multiplication can be much more decreased by block control using the MMX instructions, in software implementations and by filter operations using DSP, in hardware implementations.

In other words, we can implement a constant multiplication by less than half of the cost of a general multiplication , in either case of software or hardware.

Therefore the cost of above algorithm is cheaper than $3M+3S$, in the actual cryptographic practice.

Next, we propose a mixed point addition algorithm in PL-coordinates.

[THEOREM 2]

The Ep (X_2, Y_2, Z_2, T_2) of an affine point (x, y) and a PL-projective point (X_1, Y_1, Z_1, T_1) is obtained by:

Order	Operation	Cost
1	$A = X_1 + xZ_1$	$1M$
2	$B = [Y_1 + y*T_1]$	$1M$
3	$C = AZ_1$	$1M$
4	$D = C(B+C)$	$1M$
5	$Z_2 = C^2, \quad T = Z_2{}^2$	$2S$
6	$X_2 = [B*B + C*A^2 + D]$	$1M+1S$
7	$Y_2 = (X_2 + xZ_2)*D + (x+y)*T_2$	$3M-2S$
Total		$8M+1S$

We abbreviate the proof of the theorem, since it is quite similar to one of theorem 1. □

As well known in communication practice, all Ed’s can be eliminated in the case of transmission.

In the case, we can eliminate the modulation of the X - coordinate as well as the one of the Y - coordinate, in the algorithm, using analogue of PL-coordinates. It can be easily checked that this also decreases by $1S$ the cost of above algorithm.

Therefore the cost of an Ep is estimated to be $8M$, in the transmission.

Summing up, theorems 1 and 2 show that when using the PL- coordinates , the costs for Ed and Ep are less than $3M+3S$ and $8M+1S$, respectively.

Finally, we estimate the cost of an scalar multiplication on non-supersingular elliptic curve in practice. Let length of the scalar for point multiplication is N . In practice, the length also is the size of base field or the strength of security. When using the algorithms proposed above and the window method with width 4, cost needed for a key agreement is estimated as follows.

Case of transmission : $W_1 = Ep*N/16 = N*8M/16 = N*0.5M$

Case of reception : $W_2 = Ed*N + Ep*N/4 = N*(3M+3S+2M+0.25S)$

$= N*(5M+3.25S)$

Total : $W = W_1 + W_2 = N * (5.5M + 3.25S)$

If the length N is about 200 bit, a usual P4-2.4GHz computer can compute four million of finite field multiplications per second by using the method for speeding-up proposed in [8]. Thereby, according to above discussion we can generate more than three thousand ECC keys of 200 bit long per second.

4. References

[1] I. Blake, G. Seroussi, and N. Smart. Elliptic Curves in cpyptography, pp. 60-72, Cambridge, U.K.: Cambridge Univ. Press, 1999.

[2] J. Lopez and R. Dahab. “Improved Algorithms for Elliptic Curve Arithmetic in $GF(2^n)$ ”, Proc. Selected Areas in Cryptography-SAC`98, pp. 201-212, 1998.

[3] A. Higuchi and N. Takagi. “A Fast Addition Algorithm for Elliptic Curve Arithmetic in $GF(2^n)$ Using Projective Coordinates”, Information Processing Letters, vol. 76, pp. 101-103, 2000.

[4] E. Al-Daoud, R. Mahmed, M. Rushdan, and A. Kilioman. “A New Addition Formula for Elliptic Curves over $GF(2^n)$ ”, IEEE Transactions on computers, vol. 51, no. 8, pp. 972-975, 2002.

[5] A. Satoh and K. Takano. “A Scalable Dual –Field Elliptic Curve cryptographic processor”, IEEE Transactions on Computers, vol. 52, no. 4, pp. 449-460, 2003.

[6] M. Aydos, T. Yanik and Ç. K. Koç. “High -Speed Implementation of an ECC-based Wireless authentication protocol on an ARM Microprocessor”, IEE Proc. Commun., vol. 148, no. 5, pp. 273-279, 2001.

[7] R. M. Avanzi. A Note on Square Roots in Binary Fields. Cryptology ePrint Archive: Report 2007/103, 2007, http:// eprint.iacr.org/2007/103.

[8] K. H. Kim, S. I. Kim and C. S. Sin. Fractal-structured Karatsuba’s algorithm for binary field multiplication. Kwahagwonthongbo(Bulletin of the Academy of Sciences, DPRK), 1, 6-8, 2005.