Associative Rational Points for Improving Random Walkswith Collision–based Attack on Elliptic Curve Discrete Logarithm Problem

Yasuyuki Nogami
Graduate School of Natural Science and Technology, Okayama University
3–1–1 Tsushima–naka,
Kita–ward, Okayama, Okayama, 700–8530, Japan
Email: yasuyuki.nogami [AT] okayama-u.ac.jp

Thomas H. Austin
San Jose State University
United States

Abstract—**Pollard's Rho method is well–known as an efficient method for solving discrete logarithm problems such as the elliptic curve discrete logarithm problem (ECDLP). It consists of two parts: a random walk and collision detection. This paper proposes *associative rational points* for accelerating the random walk procedure. In detail, it considers two associative rational points $T_{i+1}^{+}=T_i-W_i$and $T_{i+1}^{-}=T_i-W_i$, where T_i and W_i are rational points. In order to make the random walk more efficient, random rational points should be efficiently generated to which *associative rational points* has a contribution.$T_{i+1}^{+}$and $T_{i+1}^{-}$are obtained by less computational cost than that of two elliptic curve additions. In order to show the contribution of the proposed idea, this paper experiments with small ECDLPs as examples.**

Keywords--- elliptic curve cryptography, elliptic curve discrete logarithm problem, Pollard's rho method, random walk

I. INTRODUCTION

The security of elliptic curve cryptography (ECC) is guaranteed by the difficulty of the elliptic curve discrete logarithm problem (ECDLP). Let E be an elliptic curve defined over a certain finite field F and let Q be a rational point of order r in E. For a scalar $0 \le s < r$, ECC often calculates a scalar multiplication $R = [s]Q$ in the procedures of encryption and decryption. In practice, it is said that r, that is the order of the cyclic group generated by Q, needs to be more than 160. The scalar multiplication $[s]Q$ is efficiently calculated by the binary method or the improved non–adjacent form (NAF) method [1]; however, its inverse problem of calculating the scalar s from Q and R is difficult.

In general, the security of cryptography is evaluated by attacking the cryptography, such as factoring for RSA cryptography. In the case of ECC, Pollard's rho method [2] is well known as one of the most practical attacking methods. It is basically a collision–detection type method. It generates a lot of random rational points that is often called a *random walk*, then detects a collision from the generated points. In order to improve the rho method, some efficient approaches such as grouping, the Montgomery trick, parallelization, and distinguished points have been proposed. The grouping technique contributes to reduce the size of ECDLP, the Montgomery trick reduces the number of inversions required for random walks, the rho method is efficiently parallelized, and distinguished points reduces the number of stored points for detecting a collision [3]. They are well known as efficient techniques; however, in order to more strictly evaluate the security of ECC, many researchers are still struggling to find more improvements.

This paper proposes *associative rational points* that slightly improves Pollard's rho method on ECDLP. Since the random walk procedure calculates a lot of elliptic curve additions such as T_i+W_i, where T_i and W_i are rational points, it needs a lot of inversions in the base field. Then, in addition to the Montgomery trick, the idea furthermore reduces the number of required inversions. It considers two associative rational points $T_{i+1}^{+}= T_i - W_i$ and $T_{i+1}^{-}= T_i - W_i$. In general, two elliptic curve additions need two inversions in the base field; however, for calculating the two associative rational pointsT_{i+1}^{+} and T_{i+1}^{-}it is enough to calculate only one inversion $(x_{Ti} - x_{Wi})^{-1}$, where x_{Ti} and x_{Wi} are x– coordinates of T_i and W_i, respectively. This efficiency is obtained from a typical feature of elliptic curve subtraction. In addition to the Montgomery trick, the idea contributes to slightly improving a collision–based attack on ECDLP such as Pollard's rho method. In order to show the efficiency of the proposed idea, this paper attacks small ECDLPs as examples.

This work is a theoretically generalized version of our previous work presented at ISCIT2014 [4]. In what follows, F_p, F_{p^m}, and $E(F_{p^m})$ respectively denote a prime field whose modular number is a prime number p, m–th extension field over F_p, and the elliptic curve defined over F_{p^m}.

II. FUNDAMENTALS

This section briefly reviews elliptic curve for cryptography and the basics of Pollard's rho method [2].

A. Elliptic curve

Let us consider an elliptic curve E as

$$E(x,y): y^2 = x^3 + ax + b, \qquad b \in \mathrm{F}_p. \tag{1}$$

The solutions of Eq.(1) are called rational points on the curve. In the case that the definition field is F_{p^m}, the set of rational points on the curve E including the infinity point O is denoted by $E(\mathrm{F}_{p^m})$. It forms an additive group for the following elliptic curve addition (ECA).

For two rational points $R_1(x_1,y_1)$, $R_2(x_2,y_2) \in E(\mathrm{F}_{p^m})$, the elliptic curve addition $R_3(x_3,y_3) = R_1 + R_2$ is given as follows. This operation is called *elliptic curve addition*.

$$\lambda = \begin{cases} \frac{y_2-y_1}{x_2-x_1} & R_1 \neq R_2 \text{and } x_1 \neq x_2 \\ \frac{3x_1^2+a}{2y_1} & R_1 = R_2 \text{and } y_1 \neq 0 \\ \phi & \text{otherwise} \end{cases}, \tag{2a}$$

$$\begin{pmatrix} x_3 \\ y_3 \end{pmatrix} = \begin{cases} \begin{pmatrix} \lambda^2 - x_1 - x_2 \\ (x_1 - x_3)\lambda - x_1 - y_1 \end{pmatrix} & \text{if} \lambda = \phi \\ \mathrm{O} & \text{otherwise} \end{cases} \tag{2b}$$

As shown above, an ECA needs fundamental arithmetic operations in the definition field such as multiplication and inversion (division). Among the arithmetic operations, it is said that an inversion is the most time–consuming.

B. Pollard's rho method

This paper applies the proposed idea to the well–known Pollard's rho method on ECDLP. Let G be a cyclic subgroup of order r in $E(\mathrm{F}_{p^m})$. According to the Lagrange theorem, r divides the order of $E(\mathrm{F}_{p^m})$. Then, an ECDLP is a problem to solve the scalar s from two rational points Q and $R \in \mathrm{G}$ such that $R = [s]Q$, where $0 \leq s < r$. Based on only their x and y coordinates, Pollard's rho method tries to find the scalar s using Alg. 1 [2].

Algorithm 1: Pollard's Rho Method

Input: Q, $R = [s]Q \in \mathrm{G}$, $0 \leq s < r$.
Output: s.
1 for $i = 0$ to $N - 1$ do
2 $\quad T_i \leftarrow [u_i]Q + [v_i]R$,

where u_i and v_i are random numbers less than r.
3 for $i = N$ to $r - 1$ do
4 $\quad l \leftarrow \eta_N(T_{i-1})$.
5 $\quad u_i \leftarrow u_{i-1} + u_l, v_i \leftarrow v_{i-1} + v_l, T_i \leftarrow T_{i-1} + T_l$.
6 $\quad$ if $T_i = T_j\ (0 \leq j < i)$, then exit this loop.
7 $s \leftarrow (u_j - u_i) \cdot (v_i - v_j)^{-1} \pmod r$.

Let x_T be the x–coordinate of rational point T. First, Steps 1 to 2 prepare N random rational points $T_0, \cdots, T_{N-1}$ with Q and R. Then, Steps 3 to 6 iteratively generate a lot of random rational points T_i by using the precomputed random walk table, where η_N is a hash function such as $\eta_N(T) := x_T \bmod N$. It is enough that η_N uniquely determines a certain integer from 0 to $N - 1$ corresponding to the input.

According to the *birthday paradox*, $\sqrt{\pi r/2}$ random points averagely make a collision at Step 7, then the collision leads to the scalar s as Step 8. In detail, the following relations hold when $T_i = T_j$.

$$\begin{aligned} [u_i]Q + [v_i]R &= [u_j]Q + [v_j]R, \\ [u_i + v_i s]Q &= [u_j + v_j s]Q. \end{aligned} \tag{3a}$$

$$\begin{aligned} u_i + v_i s &\equiv u_j + v_j s \bmod r, \\ s &\equiv (u_j - u_i) \cdot (v_i - v_j)^{-1} \bmod r. \end{aligned} \tag{3b}$$

According to the elliptic curve addition Eqs.(2), rho method needs 6 additions, 3 multiplications and an inversion in F_{p^m} to obtain a random rational point T_i. Then, this paper proposes an idea to efficiently obtain another random point called *associative* point at each iteration in order to accelerate the random walk procedure.

III. PROPOSAL OF ASSOCIATIVE POINT

This paper proposed to calculate the associative point T_i^- in addition to T_i for each iteration in Alg. 1 as follows:

$$T_i = T_i^+ = T_{i-1} + T_l = T_{i-1} + (x_{T_l}, y_{T_l}), \tag{4a}$$

$$T_i^- = T_{i-1} - T_l = T_{i-1} + (x_{T_l}, -y_{T_l}). \tag{4b}$$

Calculating these two points separately by Eqs.(2) doubles the computational cost; however, the associative point T_i^- eliminates the work of one inversion by simultaneously computing them. In detail, since the x–coordinates shown in Eqs.(4) are equal, the inverses $(x_{T_{i-1}} - x_{T_l})^{-1}$ for Eq.(2a) become the same as each other. Since the inversion is the most time–consuming operation among the fundamental arithmetic operations in F_{p^m}, applying the associative rational point leads to the acceleration of the random walk.

Applying the associative rational point that is the proposal of this paper, our improvement of the rho method is given by Alg.2. An important point is that the index l for T_l in the random walk table is determined by T_{i-1}^+ in this algorithm.
In other words, T_{i-1}^- does not affect the selection of T_l.

Algorithm 2: Proposed random walk using associative rational points

Input: $Q, R = [s]Q \in \mathrm{G}$, $0 \le s < r$.
Output: s.
1for $i = 0$ to $N - 1$ do
2 $T_i^+ \leftarrow [u_i]Q + [v_i]R$,
where u_i and v_i are random numbers less than r.
3 for $i = 0$ to $N - 1$ do
4 $l \leftarrow \eta_N(T_{i-1}^+)$.
5$u_i^+ \leftarrow u_{i-1} + u_l, v_i^+ \leftarrow v_{i-1} + v_l, T_i^+ \leftarrow T_{i-1} + T_l$.
6$u_i^- \leftarrow u_{i-1} - u_l, v_i^- \leftarrow v_{i-1} - v_l, T_i^- \leftarrow T_{i-1} - T_l$.
7if a collision occurs such as
$T_i^+ = T_j^- (0 \le j < i)$†, then exit this loop.
8 $s \leftarrow \left(u_j^- - u_i^+\right) \cdot \left(v_i^+ - v_j^-\right)^{-1}$†.

† When $T_i^+ = T_j^+$ as another case, $\leftarrow \left(u_j^+ - u_i^+\right) \cdot \left(v_i^+ - v_j^+\right)^{-1}$.

IV. EXPERIMENTAL RESULTS

Tables IV and V show small experimental results of the proposed method compared with the original rho method corresponding to the parameter settings given in Table II and Table III, respectively. In this simulation, let G be a cyclic subgroup in $E(\mathrm{F}_p)$ that is defined over a prime field F_p. Note that A, M, and I respectively denote the calculation costs of an addition, a multiplication, and an inversion in F_p.

According to the results, the computation time for solving an ECDLP on G were averagely reduced by about 33%. As an important point, the total numbers of the generated random points between the original rho and proposed method are almost the same.

V. ALGORITHMIC COOPERATIVENESS FOR OTHER TECHNIQUES

This section discusses some other efficient techniques and their cooperativeness with the proposed idea.

Table I

Experimental Environment	
CPU	Intel Core i7–870 2.93GHz†
OS	Windows 7 Professional (64–bit)
GCC	ver. 4.5.0
GMP	ver. 6.0.0

† The CPU had 4 cores but this experiment used only 1 core.

Table II

Parameters	
$E(x,y)$	$y^2 = x^3 + 7$
p	298464723373 (39 bits)
r	298464176149 (39 bits)

Table III

Parameters	
$E(x,y)$	$y^2 = x^3 + 17$
p	3055438451161 (42 bits)
r	3055436701561 (42 bits)

A. Montgomery trick

The Montgomery trick is well used for reducing the number of inefficient arithmetic operations such as inversion. Consider the calculation of a^{-1}, b^{-1}, and c^{-1}. They are of course obtained by three inversions; however, if a multiplication is much more efficient than an inversion, the following calculation is better for obtaining them.

$$M_{ab} = ab,\ M_{abc} = cM_{ab},\ I_{abc} = M_{abc}^{-1}, \tag{5a}$$

$$c^{-1} = I_{abc}M_{ab},\ I_{ab} = cI_{abc}, \tag{5b}$$

$$b^{-1} = aM_{ab},\ a^{-1} = bI_{ab}. \tag{5c}$$

In general, when we calculate n inverses, the Montgomery trick just requires one inversion and $3n$ multiplications. Applying our technique in addition to the Montgomery trick, the number of inversions and multiplications required for the random walk are twice more reduced.

B. Grouping

Suppose that the rational points on a target ECDLP are divided into groups such that each group has n rational points. If the n rational points in each group are connected to each other by a certain efficient mapping together with an explicit scalar multiplication, the average number for findinga collision becomes much smaller as $\sqrt{\pi r/2n}$.

As an example, the negation mapping is well known. Let Neg $(\cdot)$ be the negation mapping, an arbitrary rational point $R(x_R, y_R)$ of order r satisfies the following relation.

$$\mathrm{Neg}(R) = -R = (x_R, -y_R) = [r-1]R. \tag{6}$$

In case of the negation mapping, the number n of points in each group is equal to 2. In other words, R and $\mathrm{N}(R) = -R$ constitute a group. Thus, the grouping technique reduces the size of ECDLP. It is easily found the proposed idea, that is *associative points*, is cooperative to the grouping technique. It is also noted that the grouping technique needs to determine a representative among the rational points in the group; however, the determination does not cause a large overhead in general.

The GLV technique [5] is available for the grouping. The combination of grouping and representative determination can cause *fruitless cycle*. As an example on the negation mapping, when $T_{\eta_N(T_{i+1})}$is occasionally equal to $T_{\eta_N(T_i)}$, a fruitless cycle of cycle 2 occurs as follows.

$$T_{i+1} = T_i + T_{\eta_N(T_i)}, \quad (7a)$$

$$T_{i+2} = \mathrm{Neg}\big(\mathrm{Neg}(T_{i+1})\big) + T_{\eta_N(T_{i+1})} = T_i. \quad (7b)$$

Eqs.(7) find a collision such as $T_{i+2} = T_i$; however, they cannot solve the discrete logarithm problem. Thus, detecting and then escaping such a fruitless cycle are problems in practice and need additional care [6].

on are iteratively generated. Just changing the initial points for the k random walks efficiently generates random rational points, where the initial point corresponds to T_N in the cases of Alg.1 and Alg.2.

It is found that the associative point is cooperative to the parallelization. The reason why the random walks running in parallel need to share one random walk table comes from adapting the *distinguished point* technique descried below.

D. Distinguished point

In practice, a collision–based attack needs a large storage space if all of the generated random points are kept in the storage. Correspondingly, the time for detecting a collision also

Table IV

EXPERIMENTAL RESULTS OF THE PROPOSED METHOD WITH THE PARAMETERS SHOWN IN Table II

	original rho method	method
Calculation cost for generating *two* random points†	$16A + 6M + 2I$	$\cdot I$
# of the expected random points for a collision	$\sqrt{\pi r/2} \approx 684535$	
# of the generated points until a collision††	688597	
Calculation time for solving an ECDLP††	0.91 sec.	

†: For the proposed method, it denotes the calculation cost for generatingT_i^+ and its associative T_i^-.

††: They are respectively the average number and calculation time by solving 1000 ECDLPs with 1000 random scalars.

Table V

	original rho method	proposed method
Calculation cost for generating *two* random points†	$16A + 6M + 2I$	$14A + 6M + I$
# of the expected random points for a collision	$\sqrt{\pi r/2} \approx 2190769$	
# of the generated points until a collision††	2,182,561	2,166,959
Calculation time for solving an ECDLP††	3.15sec.	2.13sec.

EXPERIMENTAL RESULTS OF THE PROPOSED METHOD WITH THE PARAMETERS SHOWN IN Table III

†: For the proposed method, it denotes the calculation cost for generatingT_i^+ and its associativeT_i^-.

††: They are respectively the average number and calculation time by solving 1000 ECDLPs with 1000 random scalars.

C. Parallelization

The rho method is efficient for parallel computing. When there are k calculation cores, running k random walks in parallel leads to finding a collision k times faster for which a precomputed random walk table needs to be shared in every random walk process. In the cases of Alg.1 and Alg.2, the precomputed random walk table means the N rational points T_0 to T_{N-1} by which the following random rational points T_N and so

becomes large. Suppose that a collision $T_i = T_j, i \neq j$ occurs in the procedure. Then, their following points also give collisions as

$$T_{i+t} = T_{j+t}, \ t \geq 0. \quad (8)$$

Based on this property, the number of random points to be stored in the storage can be reduced. In detail, even if T_i and T_j such that $T_i = T_j, i \neq j$ are not stored in the storage, a certain pair

of their following points will be stored and give a collision as Eq.(8).In order to distinguish a rational point that is to be stored, a positive integer parameter θ is introduced. As an example, if the *x*–coordinate of the point is divisible by θ on a usual integer division, the point is stored. When θ is equal to 10, the number of stored points will be reduced by $1/10$ times. This technique is less cooperative to the proposed *associative point* as the parameter θ becomes larger.

According to the proposed algorithm Alg.2, the index l for T_l is determined byT_{i-1}^{+}as Step 4. In other words, itsassociative pointT_{i-1}^{-}does not affect to the selection ofT_l. Thus, when a collision such as$T_i^{+} = T_i^{-}$has occurred but not been stored as distinguished points, it comes to nothing as if no collisions have occurred. It is because the following points do not keep on collisions such as Eq.(8). In detail, it is easily found that

$$T_{i+t}^{+} \neq T_{j+t}^{-},\ t \geq 0. \qquad (9)$$

Thus, as the parameter θ for distinguishing points becomes larger, the efficiency of the proposed idea, that is associative points, becomes smaller.

VI. CONCLUSION

This paper has proposed *associative rational points* in order to improve collision–based attacks such as Pollard's rho method on the elliptic curve discrete logarithm problem (ECDLP). Then, the efficiency of the proposal was evaluated experimentally. It was shown that it achieved a 33% reduction of calculation time for solving an ECDLP.

The distinguished point technique that substantially reduces the number of points stored in memory to find a collision is the most important tool for targeting large ECDLPs; however, the proposed idea does not cooperatively work with this technique as it is. To overcome this inconvenience will be an important future work.

ACKNOWLEDGMENT

This work was partially supported by JSPS KAKENHI Grant Number 25280047.

REFERENCES

[1] D. Hankerson, A. Menezes, and S. A. Vanstone, Guide to Elliptic Curve Cryptography, Springer–Verlag, p. 98, 2004.

[2] J. M. Pollard, "Monte Carlo Methods for Index Computation (mod p)," Math. Comp., vol. 32, no. 143, pp. 918–924, 1978.

[3] H. Cohen and G. Frey ed., "Handbook of Elliptic and Hyperelliptic Curve Cryptography," Chapman & Hall, 2006.

[4] Y. Kono, Y. Nogami, and T. Kusaka, "Experimental Evaluation of the Efficiency of Associative Rational Points for Random Walks on ECDLP," 2014 International Symposium on Communications and Information Technologies (ISCIT), R4–C–4, pp. 366–367, 2014.

[5] R. P. Gallant, R. J. Lambert, and S. A. Vanstone, " Faster Point Multiplication on Elliptic Curves with Efficient Endomorphisms," CRYPTO 2001, LNCS 2139, pp. 190–200, Springer–Verlag, 2001.

[6] D. J. Bernstein, T Lange, and P. Schwabe, "On the Correct Use of the Negation Map in the Pollard rho Method," PKC 2011, LNCS 6571, pp. 128–146, Springer–Verlag, 2011.