JSIAM Letters Vol.6 (2014) pp.37–39 ©2014 Japan Society for Industrial and Applied Mathematics

Scalar multiplication for twisted Edwards curves using the extended double-base number system

Yasunori Mineo[1] and Shigenori Uchiyama[1]

[1] Tokyo Metropolitan University, 1-1 Minami-Osawa, Hachioji, Tokyo 192-0397, Japan

E-mail *mineo-yasunori@ed.tmu.ac.jp*

Received December 13, 2013, Accepted January 17, 2014

Abstract

This paper analyzes the problem of speeding up single-scalar multiplication of a recently introduced type of elliptic curve, so-called “twisted Edwards curve”, and also presents a new construction of addition chains using the extended double-base number system. Our method uses the Fibonacci sequence. It was found through numerical investigation that our double-base chains can save time, compared with other methods in previous work.

Keywords twisted Edwards curves, extended double-base number system, Fibonacci sequence, single-scalar multiplication

Research Activity Group Algorithmic Number Theory and Its Applications

1. Introduction

In [1], Edwards introduced a new normal form for elliptic curves, now known as Edwards curves, for which the addition law is efficient. In [2], Doche and Imbert introduced a new number system called the extended double-base number system. The idea is to expand a positive integer n as a sum $\sum_i d_i 2^{a_i} 3^{b_i}$ of as few terms as possible, with d_i or $-d_i$ which is chosen from a coefficient set S larger than $\{1\}$, and with the restrictions $a_1 \geq a_2 \geq \cdots$ and $b_1 \geq b_2 \geq \cdots$. Then, one can express a scalar multiple $[n]P$ as a sum $\sum_i [d_i 2^{a_i} 3^{b_i}]P$ of very few points. In [3], Bernstein et al. analyzed the best speeds that can be obtained for single-scalar multiplication with various elliptic curves by using the extended double-base number system.

In this paper, we analyze the best speeds with twisted Edwards curves, introduced in [4], using the conventional coefficient set $\mathcal{S}$, as well as another one previously unseen in the literature. Our coefficient set includes a subset of the Fibonacci sequence. By using our new double-base chains, we can speed up for single-scalar multiplication.

The plan of the paper is as follows. In Section 2, we recall the definition of the twisted Edwards curves, and of three coordinate systems on these curves [4,5]: projective twisted Edwards; inverted twisted Edwards; and extended twisted Edwards. We also show new tripling formulas that are needed in making up double-base chains. In Section 3, we review the extended double-base number system, and present a new choice of $\mathcal{S}$. Our experiments and results are described in Section 4 before concluding with Section 5.

2. Twisted Edwards curves

Definition 1 ([4]) *Let k be a field of odd characteristic, and $a, d \in k$ with $ad(a-d) \neq 0$. The twisted Edwards curve with coefficients a and d is the curve $E_{E,a,d} : ax^2 + y^2 = 1 + dx^2y^2$.*

Let $(0,1)$ be the neutral element of the group. Then the inversion of $P = (x_1, y_1)$ is written by $(-x_1, y_1)$.

2.1 Addition law

Let $P = (x_1, y_1)$, $Q = (x_2, y_2)$ be points on the twisted Edwards curve $E_{E,a,d}$. The sum of these points on $E_{E,a,d}$ is

$$P + Q = \left(\frac{x_1y_2 + y_1x_2}{1 + dx_1x_2y_1y_2}, \frac{y_1y_2 - ax_1x_2}{1 - dx_1x_2y_1y_2} \right).$$

This formula also works for doubling, i.e., $P = Q$. Then we can obtain tripling formulas. One can triple a point by first doubling it and then adding the result to itself by applying the curve equation as in doubling. $(x_3, y_3) = [3](x_1, y_1)$, with

$$x_3 = \frac{(ax_1^2 + y_1^2)^2 - (2y_1)^2}{4a(ax_1^2 - 1)x_1^2 - (ax_1^2 - y_1^2)^2} x_1,$$

$$y_3 = \frac{(ax_1^2 + y_1^2)^2 - a(2x_1)^2}{-4(y_1^2 - 1)y_1^2 + (ax_1^2 - y_1^2)^2} y_1.$$

2.2 Coordinates

Bernstein and Lange gave efficient formulas for the group operations. They introduced projective coordinates and inverted coordinates in [4]. In [5], Hisil et al. proposed a new system called *extended twisted Edwards coordinates*. We review these coordinates and show tripling algorithms.

2.3 Projective twisted Edwards coordinates

To avoid inversions, Bernstein and Lange work on the projective twisted Edwards curve

$$(aX^2 + Y^2)Z^2 = Z^4 + dX^2Y^2. \tag{1}$$

For $Z_1 \neq 0$, the homogeneous point $(X_1 : Y_1 : Z_1)$ represents the affine point $(X_1/Z_1, Y_1/Z_1)$ on $E_{E,a,d}$.

2.4 Inverted twisted Edwards coordinates

Another way to avoid inversions is using a point $(X_1 : Y_1 : Z_1)$, with $X_1Y_1Z_1 \neq 0$ to represent the affine point

$(Z_1/X_1, Z_1/Y_1)$ on $E_{E,a,d}$.

2.5 *Extended twisted Edwards coordinates*

Hisil et al. proposed using a point $(X_1 : Y_1 : T_1 : Z_1)$, with $Z_1 \neq 0$ which satisfies (1) and corresponds to the extended affine point $(X_1/Z_1, Y_1/Z_1, T_1/Z_1)$. Here, $T_1 = X_1Y_1/Z_1$. Next, we show new tripling algorithms for these coordinates. Here, **M** is a field multiplication, **S** is a field squaring, and **D** is a multiplication by a or d.

2.6 *Tripling in inverted twisted Edwards coordinates*

The following sets of formulas compute $(X_3 : Y_3 : Z_3) = [3](X_1 : Y_1 : Z_1)$. The first one costs $9\mathbf{M} + 4\mathbf{S} + 2\mathbf{D}$, while the second needs $7\mathbf{M} + 7\mathbf{S} + 2\mathbf{D}$. Here are $9\mathbf{M} + 4\mathbf{S} + 2\mathbf{D}$ formulas for tripling:

$$\begin{aligned}
&A \leftarrow X_1^2, \quad B \leftarrow aY_1^2, \quad C \leftarrow Z_1^2, \quad D \leftarrow A + B,\\
&E \leftarrow 4(D - d \cdot C), \quad H \leftarrow 2D \cdot (B - A),\\
&P \leftarrow D^2 - A \cdot E, \quad Q \leftarrow D^2 - B \cdot E,\\
&X_3 \leftarrow (H + Q) \cdot Q \cdot X_1, \quad Y_3 \leftarrow (H - P) \cdot P \cdot Y_1,\\
&Z_3 \leftarrow P \cdot Q \cdot Z_1.
\end{aligned}$$

Here are $7\mathbf{M} + 7\mathbf{S} + 2\mathbf{D}$ formulas for tripling:

$$\begin{aligned}
&A \leftarrow X_1^2, \quad B \leftarrow aY_1^2, \quad C \leftarrow Z_1^2, \quad D \leftarrow A + B,\\
&E \leftarrow 4(D - d \cdot C), \quad H \leftarrow 2D \cdot (B - A),\\
&P \leftarrow D^2 - A \cdot E, \quad Q \leftarrow D^2 - B \cdot E,\\
&X_3 \leftarrow (H + Q) \cdot [(Q + X_1)^2 - Q^2 - A],\\
&Y_3 \leftarrow 2(H - P) \cdot P \cdot Y_1,\\
&Z_3 \leftarrow P \cdot [(Q + Z_1)^2 - Q^2 - C].
\end{aligned}$$

2.7 *Tripling in extended twisted Edwards coordinates*

The following sets of formulas compute $(X_3 : Y_3 : T_3 : Z_3) = [3](X_1 : Y_1 : T_1 : Z_1)$. The first one costs $11\mathbf{M} + 4\mathbf{S} + 1\mathbf{D}$, while the second needs $9\mathbf{M} + 7\mathbf{S} + 1\mathbf{D}$. Here are $11\mathbf{M} + 4\mathbf{S} + 1\mathbf{D}$ formulas for tripling:

$$\begin{aligned}
&A \leftarrow aX_1^2, \quad B \leftarrow Y_1^2, \; C \leftarrow (2Z_1)^2, \quad D \leftarrow A + B,\\
&E \leftarrow D^2, \quad F \leftarrow 2D \cdot (A - B), \quad G \leftarrow E - B \cdot C,\\
&H \leftarrow E - A \cdot C, \quad I \leftarrow F + H, \quad J \leftarrow F - G,\\
&X_3 \leftarrow G \cdot J \cdot X_1, \quad Y_3 \leftarrow H \cdot I \cdot Y_1,\\
&T_3 \leftarrow G \cdot H \cdot T_1, \quad Z_3 \leftarrow I \cdot J \cdot Z_1.
\end{aligned}$$

Here are $9\mathbf{M} + 7\mathbf{S} + 1\mathbf{D}$ formulas for tripling:

$$\begin{aligned}
&A \leftarrow aX_1^2, \quad B \leftarrow Y_1^2, \quad C \leftarrow Z_1^2, \quad D \leftarrow A + B,\\
&E \leftarrow D^2, \quad F \leftarrow 2D \cdot (A - B), \quad K \leftarrow 4C,\\
&L \leftarrow E - B \cdot K, \quad M \leftarrow E - A \cdot K, \quad N \leftarrow F + M,\\
&O \leftarrow N^2, \quad P \leftarrow F - L, \; X_3 \leftarrow 2L \cdot P \cdot X_1,\\
&Y_3 \leftarrow M \cdot [(N + Y_1)^2 - O - B], \quad T_3 \leftarrow 2L \cdot M \cdot T_1,\\
&Z_3 \leftarrow P \cdot [(N + Z_1)^2 - O - C].
\end{aligned}$$

When one computes tripling in projective twisted Edwards coordinates, one can compute in extended twisted Edwards coordinates by simply ignoring T. The costs in projective coordinates can be reduced $2\mathbf{M}$ rather than the costs of using extended coordinates. By using tripling formulas, we can reduce multiplication costs less than the costs of using mixing doubling and addition formulas. Extended coordinates are faster than projective coordinates and inverted coordinates, while in doubling and tripling, projective coordinates and inverted coordinates are faster than extended coordinates.

3. Extended double-base number system

This section reviews the extended double-base number system (extended DBNS, for short) for computing $[n]P$ given P. Let $\mathcal{S}$ be a set containing 1. Every positive integer n can be represented as $n = \Sigma_{i=1}^m d_i 2^{a_i} 3^{b_i}$, with $|d_i| \in \mathcal{S}$, $a_0 \geq a_1 \geq a_2 \geq \cdots \geq a_m \geq 0$, and $b_0 \geq b_1 \geq b_2 \geq \cdots \geq b_m \geq 0$. This approach is called extended DBNS.

This representation is not unique. Bernstein et al. optimized single-scalar multiplication using extended DBNS. They analyzed various elliptic curves containing Edwards curves. However, twisted Edwards curves are not contained. We analyze projective twisted Edwards coordinates, inverted twisted Edwards coordinates, and extended twisted Edwards coordinates. We also propose a new precomputation set $\mathcal{S}$ for single-scalar multiplication, and optimize the best speeds that can be obtained.

3.1 *A new choice of a coefficient set $\mathcal{S}$*

We reviewed extended DBNS above. It is significant to choose a proper coefficient set $\mathcal{S}$. If one chooses a better set, single-scalar multiplication can be computed faster. We propose a new coefficient set containing $F_2 = 1, F_3 = 2, \ldots$. Here, F_i is i-th Fibonacci number. For example, if $\#\mathcal{S} = 6$, we make up $\mathcal{S} = \{1, 2, 3, 5, 8, 13\}$. If the number of elements in $\mathcal{S}$ is larger, one can choose larger coefficients, and it is possible to make up more efficient extended DBNS expansions. Moreover, by calculating the precomputation points $[2]P, [3]P, [5]P = [2]P + [3]P, [8]P = [3]P + [5]P, \ldots$, in order, the initial computation of $[c]P$ for each $c \in \mathcal{S}$ can be calculated efficiently.

3.2 *Example.*

Take the integer $n = 264290$. We consider two coefficient sets $\mathcal{S}_1 = \{1, 2, 3, 5, 7, 9\}$ and $\mathcal{S}_2 = \{1, 2, 3, 5, 8, 13\}$. The extended DBNS expansion with $\mathcal{S}_1$ can be written as $5 \cdot 2^{11}3^3 - 7 \cdot 2^6 3^3 - 5 \cdot 2^1 3^2 - 2 \cdot 2^1 3^0$. Assuming that $[2]P$, $[5]P$, and $[7]P$ are precomputed, it is possible to obtain $[264290]P$ as $[2]([3^2]([2^5 3]([2^5][5]P - [7]P) - [5]P) - [2]P)$ with 11 doublings, 3 triplings, and 3 additions. On the other hand, the extended DBNS expansion with $\mathcal{S}_2$ can be written as $13 \cdot 2^8 3^4 - 8 \cdot 2^3 3^4 - 8 \cdot 2^2 3^1 + 2 \cdot 2^0 3^0$. Assuming that $[2]P$, $[8]P$, and $[13]P$ are precomputed, one can obtain $[264290]P$ as $[2^2 3^1]([2^1 3^3]([2^5][13]P - [8]P) - [8]P) + [2]P$ with 8 doublings, 4 triplings, and 3 additions. The latter expansion can be computed faster.

To compute extended double-base chains, we used the greedy type algorithm in [2] (Algorithm 1). In this algorithm, one chooses the best approximation $d_1 2^{a_1} 3^{b_1}$ of given integer n first. Total costs for single-scalar multiplication using extended DBNS depend on a_1, b_1, and the length of the chain. If these values can be reduced, it is possible to compute with less costs. Let d_1 be the largest number in $\mathcal{S}$. Other d_i's are chosen in $\mathcal{S}$ properly. Then, we can reduce a_1 and b_1 as in the above example. We carried out the experiments and showed the results in the next section.

Table 1. Total multiplication counts for each curve shape.

Shape	l (bit)	**M** (New results)	**M** (Bernstein)	a_0
Proj	160	1142.08706	1149.18034	156
Proj	200	1392.19286	1402.05392	196
Proj	256	1739.61144	1749.63448	252
Proj	300	2012.27620	2022.30108	296
Proj	400	2653.80934	2682.76172	396
Proj	500	3273.85562	3302.77824	496
Inv	160	1126.75536	1134.59694	156
Inv	200	1376.36036	1386.94202	196
Inv	256	1723.79004	1734.50618	252
Inv	300	1996.43160	2007.16798	296
Inv	400	2633.84604	2663.45662	396
Inv	500	3253.88962	3283.49154	496
Ext	160	1272.09880	1283.00072	156
Ext	200	1560.78106	1574.78950	196
Ext	256	1964.27612	1978.37394	252
Ext	300	2280.83532	2295.00180	296
Ext	400	3011.23480	3047.20314	396
Ext	500	3731.27792	3767.28818	496

Table 2. Choices of the sets.

l (bit)	$\#\mathcal{S}$	$\mathcal{S}$ (New results)	$\mathcal{S}$ (Bernstein)
160	8	$\{1, 2, 3, 5, 8, \dots, 34\}$	$\{1, 2, 3, 5, 7, \dots, 13\}$
200	9	$\{1, 2, 3, 5, 8, \dots, 55\}$	$\{1, 2, 3, 5, 7, \dots, 15\}$
256	9	$\{1, 2, 3, 5, 8, \dots, 55\}$	$\{1, 2, 3, 5, 7, \dots, 15\}$
300	9	$\{1, 2, 3, 5, 8, \dots, 55\}$	$\{1, 2, 3, 5, 7, \dots, 15\}$
400	14	$\{1, 2, 3, 5, 8, \dots, 610\}$	$\{1, 2, 3, 5, 7, \dots, 25\}$
500	14	$\{1, 2, 3, 5, 8, \dots, 610\}$	$\{1, 2, 3, 5, 7, \dots, 25\}$

4. Experiments and results

In this section, we explain our experiments and summarize our results. Our experiments are based on what Bernstain et al. carried out in [3]. We generated 10000 uniform random integers $n \in \{0, 1, \dots, 2^l - 1\}$. Here, l's are several bit sizes, namely, 160, 200, 256, 300, 400, and 500. Next, we converted each integer into a double-base chain as specified by a_0 and $\mathcal{S}$. Note that we can obtain b_0 by calculating $\lceil (\log_2 n - a_0) \log_2 3 \rceil$. In our experiments, we chose $\mathcal{S}$ optimized by them. In addition, we included the sets $\{F_2, F_3, \dots\}$. Finally, we checked that the constructed chain indeed computed n starting the chain from 1, and counted the number of triplings, doublings, and additions for those 10000 choices of n. Our experiments included the three curve shapes: Projective twisted Edwards(Proj); Inverted twisted Edwards(Inv); and Extended twisted Edwards(Ext). We follow the standard practice of counting $\mathbf{S} = 0.8\mathbf{M}$, and disregarding other field operations. We included the multiplication counts for the initial computation of $[c]P$ for each $c \in \mathcal{S}$. The results of the experiments are presented as tables. Table 1 shows total multiplication counts for each curve shape and each l. We describe our choices of the coefficient sets for each l in Table 2. The multiplication counts can be reduced approximately 1% for each curve shape and each l rather than that of Bernstein et al. For larger l, our sets can be more efficient. Extended twisted Edwards coordinates are slower than the other coordinates in our experiment. That is because the number of doublings occupies most of the multiplication counts. The costs of the doubling for extended coordinates are larger than those for projective coordinates and inverted coordinates.

5. Conclusion

In this paper, we showed explicit tripling formulas for twisted Edwards curves. We also proposed a new coefficient set for extended DBNS chains, and optimized single-scalar multiplication for twisted Edwards curves. Future works are as follows:

- analyzing mixed coordinates introduced in [5] for twisted Edwards curves,
- optimizing single-scalar multiplication for other elliptic curve shapes by using our sets,
- speeding up the algorithm which makes up extended DBNS chains.

Acknowledgments

We would like to thank the anonymous reviewer for his/her valuable comments. This work was supported in part by Grant-in-Aid for Scientific Research (C) (20540135).

References

[1] H. M. Edwards, A normal form for elliptic curves, B. Am. Math. Soc., **44** (2007), 393–422.

[2] C. Doche and L. Imbert, Extended double-base number system with applications to elliptic curve cryptography, in: Proc. of INDOCRYPT 2006, Rana Barua et al. eds., LNCS, Vol. 4329, pp. 335–348, Springer-Verlag, Berlin, 2006.

[3] D. J. Bernstein, P. Birkner, T. Lange and C. Peters, Optimizing double-base elliptic-curve single-scalar multiplication, in: Proc. of INDOCRYPT 2007, K. Srinathan et al. eds., LNCS, Vol. 4859, pp. 167–182, Springer-Verlag, Berlin, 2007.

[4] D. J. Bernstein, P. Birkner, T. Lange and C. Peters, Twisted Edwards curves, in: Proc. of AFRICACRYPT 2008, Serge Vaudenay ed., LNCS, Vol. 5023, pp. 389–405, Springer-Verlag, Berlin, 2008.

[5] H. Hisil, K. K. Wong, G. Carter and E. Dawson, Twisted Edwards curves revisited, in: Proc. of ASIACRYPT 2008, Josef Pieprzyk ed., LNCS, Vol. 5350, pp. 326–343, Springer, Berlin, 2008.