Differential Addition on Edwards Curves

Benjamin Justus and Daniel Loebenberger*

b-it
Universität Bonn
D53113 Bonn

Abstract. We give two parametrizations of points on Edwards curves that omit the X coordinate. The first parametrization leads to a differential addition formula that has the cost $5\mathbf{M} + 4\mathbf{S}$, a doubling formula that has the cost $5\mathbf{S}$ and a tripling formula that costs $4\mathbf{M} + 7\mathbf{S}$. The second one yields a differential addition formula with cost $5\mathbf{M} + 2\mathbf{S}$ and a doubling formula with cost $5\mathbf{S}$ both even on generalized Edwards curves. The price to pay for this representation is the extraction of two square roots in the ground field. For both parametrizations the formula for recovering the missing coordinate is also provided. In addition, we give an addition chain for computing the scalar multiple of a point on the Edwards curve.
Keywords. Edwards curve, addition formula, differential addition

1 Introduction

Efficient arithmetic (addition, doubling and scalar multiplication) on elliptic curves is the core requirement of elliptic curve cryptography. They are the cornerstone in applications such as the digital signature algorithms (DSA), see [8], abd Lenstra's elliptic curve factoring method [9], which is a natural adaption of Pollards's $(p-1)$-method [12] to elliptic curves. Various forms of elliptic curves have been proposed for the purpose doing efficient arithmetic. For an overview, the readers can consult the standard reference [1] or the online Explicit-Formulas Database (EFD)[1]. We have selected some of the top candidates and summarized in the table below. Here $\mathbf{M}$ (resp. $\mathbf{S}$) refers to an elementary multiplication (resp. a squaring) in the field. We ignore in this paper the cost induced by multiplication by a constant, since this operation is — especially in hardware — basically for free. Also the cost of an addition in the field will be ignored, since also the cost of this operation is negligible when compared to the cost of multiplication or squaring.

With the advent of Edwards curves [6], extensive recent work [2–5] have been able to show that doing arithmetic on Edwards curves is more efficient in most cases than on other forms of elliptic curve. Edwards curves could well be a leading contender for cryptographic applications involving elliptic curves.

* This work was partially funded by the BSI.
[1] see `http://www.hyperelliptic.org/EFD`

Table 1. Some coordinates with fast speed

Forms	Coordinates	Addition Cost	Doubling Cost
Short Weierstraß	$(X : Y : Z) = (X/Z^2, Y/Z^3)$	$12\mathbf{M} + 4\mathbf{S}$	$4\mathbf{M} + 5\mathbf{S}$
Montgomery curve	$(X : Z)$	$4\mathbf{M} + 1\mathbf{S}$	$2\mathbf{M} + 3\mathbf{S}$
Edwards curve	$(X : Y : Z)$	$10\mathbf{M} + 1\mathbf{S}$	$3\mathbf{M} + 4\mathbf{S}$
Inverted Edwards	$(X : Y : Z) = (Z/X, Z/Y)$	$9\mathbf{M} + 1\mathbf{S}$	$3\mathbf{M} + 4\mathbf{S}$

In this paper, we introduce two parametrizations of points on Edwards curves: In the first parametrization a point on an Edwards curve is represented by the projective coordinate $(Y : Z)$. Notice the X-coordinate is absent, so we can not distinguish P from $-P$ (see next section for details). This is indeed similar to Montgomery's approach [10] where he represented a point with only the x-coordinate. The parametrization leads to a new differential addition formula, a doubling formula and a tripling formula. The addition formula has the cost $6\mathbf{M}+4\mathbf{S}$ ($5\mathbf{M}+4\mathbf{S}$ in the case $c = 1$). The doubling formula has the cost $1\mathbf{M}+4\mathbf{S}$ ($5\mathbf{S}$ when $c = 1$). The tripling formula has the cost $4\mathbf{M} + 7\mathbf{S}$. We also provide methods for recovering the missing x-coordinate.

The second parametrization also omits the X coordinate. Additionally it uses the squares of the coordinates of the points only. On generalized Edwards curves, addition can be done with $5\mathbf{M} + 2\mathbf{S}$ and point doubling with $5\mathbf{S}$. In order to obtain the coordinates of the resulting point we need to extract at the end of the computation two square roots over the ground field (one for each coordinate). Thus our parametrization is best suited for multiplications of a point P with a large scalar s while working over a (finite) field. In such an application, the additional cost for the extraction of the two square roots is asymptotically negligible. Note that for point doubling we get completely rid of multiplication and employ squarings in the ground field only. This is desireable since squarings can be done slightly faster than generic multiplications, see for example [1].

We are currently carrying out several runtime estimates. Since we do not have comparable results in this direction up to now, we postpose a detailed analysis to the long version of this paper.

The plan of the paper is as follows. We recall the basics of Edwards curves in the next section and describe and prove the addition, doubling and tripling formula in section 3. The formula for recovering the x-coordinate is described in section 4. A parametrization of the points that uses the squares of the coordinates only is introduced in section 5. In section 6, we will describe an addition chain that carries out the task of single scalar multiplication of points on Edwards curves.

2 Edwards Curves

We describe now the basics of Edwards curves. More details can be found in the original papers [4, 5]. A generalized Edwards curve is of the form

$$E_{c,d} : x^2 + y^2 = c^2(1 + dx^2y^2)$$

where c, d are curve parameters in a field k of characteristic different from 2. When $c, d \neq 0$ and $dc^4 \neq 1$, the Edwards addition law is defined by

$$(x_1, y_1), (x_2, y_2) \mapsto \left(\frac{x_1y_2 + y_1x_2}{c(1 + dx_1x_2y_1y_2)}, \frac{y_1y_2 - x_1x_2}{c(1 - dx_1x_2y_1y_2)} \right). \tag{1}$$

For this addition law, the point $(0, c)$ is the neutral element. The inverse of point $P = (x, y)$ is $-P = (-x, y)$. In particular, $(0, -c)$ has order 2; $(c, 0)$ and $(-c, 0)$ are the points of order 4. When the curve parameter d is not a square in k, then the addition law (1) is complete (i.e. defined for all inputs).

3 Representing Points on Edwards Curves

Given a generalized Edwards curve $E_{c,d}$. As explained in the introduction, we represent a point P on the curve by the projective coordinate $P = (Y_1 : Z_1)$. Write $[n]P = (Y_n : Z_n)$. Then we have

Theorem 1. *Let $E_{c,d}$ be a generalized Edwards curve defined over k such that $char(k) \neq 2$, $c, d \neq 0$, $dc^4 \neq 1$ and d is not a square in k. Then the following formulæ hold when $m > n$*

$$\begin{aligned} Y_{m+n} &= Z_{m-n}\left(Y_m^2(Z_n^2 - c^2dY_n^2) + Z_m^2(Y_n^2 - c^2Z_n^2)\right) \\ Z_{m+n} &= Y_{m-n}\left(dY_m^2(Y_n^2 - c^2Z_n^2) + Z_m^2(Z_n^2 - c^2dY_n^2)\right). \end{aligned}$$

with cost $6\mathbf{M} + 4\mathbf{S}$. When $n = m$, the doubling formula is given by

$$\begin{aligned} Y_{2n} &= -c^2dY_n^4 + 2Y_n^2Z_n^2 - c^2Z_n^4 \\ Z_{2n} &= dY_n^4 - 2c^2dY_n^2Z_n^2 + Z_n^4 \end{aligned}$$

having cost $1\mathbf{M} + 4\mathbf{S}$.

Proof. Let $P_1 = (x_1, y_1)$, $P_2 = (x_2, y_2)$ be two points on the Edwards curve $E_{c,d}$ such that $P_1 \neq P_2$. Since the curve parameter d is not a square in k, the addition law (1) is defined for all inputs. Let $P_1 + P_2 = (x_3, y_3)$ and $P_1 - P_2 = (x_4, y_4)$. Then the addition law (1) gives

$$\begin{aligned} y_3c(1 - dx_1x_2y_1y_2) &= y_1y_2 - x_1x_2 \\ y_4c(1 + dx_1x_2y_1y_2) &= y_1y_2 + x_1x_2 \end{aligned}$$

After multiplying the two equations above, we obtain

$$y_3y_4c^2(1 - d^2x_1^2x_2^2y_1^2y_2^2) = y_1^2y_2^2 - x_1^2x_2^2. \tag{2}$$

Next we substitute $x_1^2 = \frac{c^2-y_1^2}{1-c^2dy_1^2}$ and $x_2^2 = \frac{c^2-y_2^2}{1-c^2dy_2^2}$ (obtained from the curve equation) in (2) to obtain

$$y_3y_4(-dy_1^2y_2^2 + c^2dy_1^2 + c^2dy_2^2 - 1) = c^2dy_1^2y_2^2 - y_1^2 - y_2^2 + c^2 \tag{3}$$

After changing (3) into projective coordinates, we see that the formula for adding $[m]P = (Y_m, Z_m)$ and $[n]P = (Y_n, Z_n)$ $(m > n)$ becomes

$$\frac{Y_{m+n}}{Z_{m+n}}\frac{Y_{m-n}}{Z_{m-n}} = \frac{Y_m^2(Z_n^2 - c^2dY_n^2) + Z_m^2(Y_n^2 - c^2Z_n^2)}{dY_m^2(Y_n^2 - c^2Z_n^2) + Z_m^2(Z_n^2 - c^2dY_n^2)}. \tag{4}$$

This proves the addition formula. If $P_1 = P_2$, we obtain by the Edwards addition law (1)

$$y_3c(1 - dx_1^2y_1^2) = y_1^2 - x_1^2.$$

Similarly, we substitute $x_1^2 = \frac{c^2-y_1^2}{1-c^2dy_1^2}$ into the equation above to obtain

$$y_3(cdy_1^4 - 2c^3dy_1^2 + c) = -c^2dy_1^4 + 2y_1^2 - c^2$$

This proves the doubling formula in Theorem 1 after changing into projective coordinates. □

We obtain additional savings in the case $c = 1$:

Corollary 1. *With the same assumptions and notations as in Theorem 1. If $c = 1$ we have for $m > n$*

$$\begin{aligned} Y_{m+n} &= Z_{m-n}\left((Y_m^2 - Z_m^2)(Z_n^2 - dY_n^2) - (d-1)Y_n^2Z_m^2\right), \\ Z_{m+n} &= -Y_{m-n}\left((Y_m^2 - Z_m^2)(Z_n^2 - dY_n^2) + (d-1)Y_m^2Z_n^2\right) \end{aligned}$$

with costs $5\mathbf{M} + 4\mathbf{S}$. For doubling we obtain

$$\begin{aligned} Y_{2n} &= -(Y_n^2 - Z_n^2)^2 - (d-1)Y_n^4, \\ Z_{2n} &= (dY_n^2 - Z_n^2)^2 + (d - d^2)Y_n^4. \end{aligned}$$

having cost $5\mathbf{S}$. □

Remark 1. A simple induction argument also shows that the computation of the 2^k-fold of a point costs $5k\mathbf{S}$.

3.1 A Tripling Formula

One also obtains a tripling formula that has the cost $4\mathbf{M} + 7\mathbf{S}$. This is cheaper than by doing a doubling then an addition. We restrict ourselves to the case $c = 1$. The calculation is similar in the case $c \neq 1$.

Proposition 1. *With the same assumptions and notations as in Theorem 1. Furthermore, we assume $char(k) \neq 3$. Then we have*

$$Y_{3n} = Y_n \left((dY_n^4 - 3Z_n^4)^2 - Z_n^4 \left((2Z_n^2 - (1+d)Y_n^2)^2 + 8Z_n^4 - (1+d)^2 Y_n^4\right)\right)$$
$$Z_{3n} = Z_n \left((Z_n^4 - 3dY_n^4)^2 - dY_n^4 \left((2Z_n^2 - (1+d)Y_n^2)^2 - 4Z_n^4 + (12d - (1+d)^2)Y_n^4\right)\right)$$

Proof. We sketch the proof. It is similar as before. Let $(x_3, y_3) = 3(x, y) = 2(x, y) + (x, y)$. Using the addition law (1), we obtain an expression for y_3. Inside the expression, make the substitution $x^2 = \frac{1-y^2}{1-dy^2}$ and simplify to obtain an expression in y only.

$$y_3 = \frac{y(d^2y^8 - 6dy^4 + (4d+4)y^2 - 3)}{-3d^2y^8 + (4d^2+4d)y^6 - 6dy^4 + 1}.$$

Substitute into projective coordinates $y = Y/Z$ and rearrange terms. The formula follows. □

4 Recovering the x-coordinate

In some cryptographic applications it is important to have at some point both x and y coordinates. This is possible as shown in the next result. We also mention there have been results [11, 7] in this direction for other forms of elliptic curve.

Theorem 2. *Let $E_{c,d}$ be a generalized Edwards curve defined over k such that $char(k) \neq 2$, $c, d \neq 0$, $dc^4 \neq 1$ and d is not a square in k. Let (x, y) be a point whose order does not divide 4. Let y_n, y_{n+1} be the affine y-coordinates of the points $[n]P, [n+1]P$ respectively. Then the following formula holds*

$$x_n = \frac{2yy_ny_{n+1} - c\Omega_n - cy_{n+1}^2}{cdxyy_n\left(\Omega_n - y_{n+1}^2\right)},$$

where

$$\Omega_n = \frac{Ay_n^2 + B}{dBy_n^2 + A}$$
$$A = 1 - c^2dy^2$$
$$B = y^2 - c^2.$$

In order to prove Theorem 2, we need the following result.

Proposition 2. *Fix an Edwards curve $E_{c,d}$ such that $char(k) \neq 2$, $c, d \neq 0$, $dc^4 \neq 1$ and d is not a square in k. Let $Q = (x, y)$, $P_1 = (x_1, y_1)$ be two points on $E_{c,d}$. Define $P_2 = (x_2, y_2)$ and $P_3 = (x_3, y_3)$ by $P_2 = P_1 + Q$ and $P_3 = P_1 - Q$. Then we have*

$$x_1 = \frac{2yy_1 - cy_2 - cy_3}{cdxyy_1(y_3 - y_2)} \tag{5}$$

provided the denominator does not vanish.

Proof. By the addition law (1), we have

$$\begin{aligned} c(1 - dxx_1yy_1)y_2 &= yy_1 - xx_1 \\ c(1 + dxx_1yy_1)y_3 &= yy_1 + xx_1. \end{aligned}$$

Add the two equations and solve for x_1, the Proposition follows. □

The following lemma tells us when the formula (5) is valid.

Lemma 1. *With the same assumptions as in Proposition 2. Furthermore, let P_1, Q be points whose order does not divide 4. Then the formula (5) always holds.*

Proof. The points P_1 and Q have orders that are not $1, 2, 4$, so $x, y, y_1 \neq 0$. Suppose now $y_2 = y_3$ (i.e. y-coordinates of $P_1 + Q$ and $P_1 - Q$ are the same). By the addition Law (1), this implies

$$\frac{yy_1 - xx_1}{c(1 - dxx_1yy_1)} = \frac{yy_1 + xx_1}{c(1 + dxx_1yy_1)}$$

whence $dy^2y_1^2 = 1$. But d is not a square in k, so a contradiction is arrived.

Proof (Theorem 2). Let $[n]P = (x_n, y_n)$ where P is not a 4-torsion point on $E_{c,d}$. Our task is to recover the x_n. By Proposition 2, we may write

$$x_n = \frac{2yy_n - cy_{n-1} - cy_{n+1}}{cdxyy_n(y_{n-1} - y_{n+1})} \tag{6}$$

where y_{n-1},y_{n+1} are the y-coordinates of the points $[n-1]P$ and $[n+1]P$ respectively. Now the variable y_{n-1} can be eliminated because of (4). Indeed we may write using (4) in affine coordinate

$$y_{n-1}y_{n+1} = \frac{Ay_n^2 + B}{dBy_n^2 + A} \tag{7}$$

where

$$A = 1 - c^2dy^2, \quad B = y^2 - c^2.$$

Now from (7), y_{n-1} can be isolated and put back in (6). This gives

$$x_n = \frac{2yy_ny_{n+1}(dBy_n^2 + A) - c(Ay_n^2 + B) - cy_{n+1}^2(dBy_n^2 + A)}{cdxyy_n\left(Ay_n^2 + B - y_{n+1}^2(dBy_n^2 + A)\right)}.$$

The claim follows. □

5 A parametrization using squares only

The formulae in Theorem 1 show that for the computation of Y_{m+n}^2 and Z_{m+n}^2 it is sufficient to know the squares of the coordinates of the points (Y_m, Z_m), (Y_n, Z_n) and (Y_{m-n}, Z_{n-m}) only. This gives

Theorem 3. *With the same assumptions as in Theorem 1, the following formulæ hold when $m > n$*

$$\begin{aligned} Y_{m+n}^2 &= Z_{m-n}^2\left((A+B)/2\right)^2 \\ Z_{m+n}^2 &= Y_{m-n}^2\left((A-B)/2 + (d-1)Y_m^2(Y_n^2 - c^2Z_n^2)\right)^2. \end{aligned}$$

with

$$\begin{aligned} A &:= (Y_m^2 + Z_m^2)((1-dc^2)Y_n^2 + (1-c^2)Z_n^2), \\ B &:= (Y_m^2 - Z_m^2)((1+c^2)Z_n^2 - (1+dc^2)Y_n^2). \end{aligned}$$

The cost of this addition is $5\mathbf{M} + 2\mathbf{S}$ if one stores the squares of the coordinates only. When $n = m$, we obtain

$$\begin{aligned} Y_{2n}^2 &= \left((1-c^2d)Y_n^4 + (1-c^2)Z_n^4 - (Y_n^2 - Z_n^2)^2\right)^2, \\ Z_{2n}^2 &= \left(dc^2(Y_n^2 - Z_n^2)^2 - d(c^2-1)Y_n^4 + (c^2d-1)Z_n^4\right)^2 \end{aligned}$$

with cost $5\mathbf{S}$.

Proof. This follows directly from Theorem 1 and elementary calculus. □

We will now sketch the computation of a scalar multiple $[s]P$ in this parametrization. Assume P has affine coordinates $(x : y)$. Then one would proceed as follows: After changing to projective coordinates $(X : Y : Z)$, two squares (one for each of the coordinates Y and Z) have to be computed. Now a differential addition chain (such as the one described in the next section) is employed to compute the multiple $[s]P$. During the computation we store the squares of the coordinates of the intermetiate points only. After the computation two square roots have to be extracted. Due to the contruction both computations are possible, i.e. the inputs are indeed squares in the ground field. We now end up with the Y and the Z coordinate of the point $[s]P$. Note that the computation of the square roots is only feasible if we are working over a field. This makes the proposed parametrization unsuitable for the elliptic curve factoring method.

Also the trippling formula given in Proposition 1 can be adapted to this second parametrization. Namely we have

Corollary 2. *With the same assumptions and notations as in Theorem 1. Furthermore, we assume* $char(k) \neq 3$. *Then we have*

$$Y_{3n}^2 = Y_n^2 \left((dY_n^4 - 3Z_n^4)^2 - Z_n^4 \left((2Z_n^2 - (1+d)Y_n^2)^2 + 8Z_n^4 - (1+d)^2 Y_n^4\right)\right)^2,$$
$$Z_{3n}^2 = Z_n^2 \left((Z_n^4 - 3dY_n^4)^2 - dY_n^4 \left((2Z_n^2 - (1+d)Y_n^2)^2 - 4Z_n^4 + (12d - (1+d)^2)Y_n^4\right)\right)^2$$

with cost $4\mathbf{M} + 5\mathbf{S}$. □

6 Addition chains for scalar multiplication

To compute the scalar multiple $[s]P$ using the addition and doubling formula developed in section 3, one can apply Montgomery's ladder (see [1]). The algorithms has the indistinguishability feature that prevents side channel attacks.

We present in the following an alternative algorithm (Algorithm 1) for computing $[s]P$. The algorithms uses tripling and quadrupling in addition to additions and doubling. The algorithm can be applied in any additive groups. Let us briefly analyze the cost for computing $[s]P$ in our context. We treat terms like $3P_1 + P_2$ by computing $(P_1 + P_2)$, $(P_1 + P_2) + 2P_1$ successively. This is feasible because the difference between P_1 and P_2 is always P. The cost of addition using our addition formula (Theorem 1) becomes $3\mathbf{M}+ 4\mathbf{S}$. The quadrupling of a point is done by doubling the point twice. In summary the algorithm uses 2 additions and 2 doubling for each 2-bits of s. The length of the addition chain is $\log_4 s$. So if s has n-bits, the total cost for computing $[s]P$ is n additions and n doublings.

Algorithm 1 INPUT: A point P on E and a positive integer $s = (n_{l-1} \ \ldots n_0)_4$. OUTPUT: $[s]P$.

1: $P_1 \leftarrow [n_{l-1}]P$ and $P_2 \leftarrow [n_{l-1} + 1]P$
2: **for** $i = l - 2$ down to 0 **do**
3: **if** $n_i = 0$ **then**
4: $P_1 = 4P_1$ and $P_2 = 3P_1 + P_2$
5: **else if** $n_i = 1$ **then**
6: $P_1 = 3P_1 + P_2$ and $P_2 = 2(P_1 + P_2)$
7: **else if** $n_i = 2$ **then**
8: $P_1 = 2(P_1 + P_2)$ and $P_2 = P_1 + 3P_2$
9: **else**
10: $P_1 = P_1 + 3P_2$ and $P_2 = 4P_2$
11: **end if**
12: **end for**
13: **return** P_1

7 Acknowledgements

This work was funded by the b-it foundation and the state of North Rhine-Westphalia.

References

1. R. M. Avanzi, H. Cohen, C. Doche, G. Frey, T. Lange, K. Nguyen, and F. Vercauteren. *Handbook of Elliptic and Hyperelliptic Curve Cryptography.* Discrete Mathematics and its Applications. Chapman & Hall/CRC, 2006.
2. D. J. Bernstein, P. Birkner, M. Joye, T. Lange, and C. Peters. Twisted edwards curves. In S. Vaudenay, editor, *Progress in Cryptology: Proceedings of AFRICACRYPT 2008,* Casablanca, Morocco, volume 5023 of *Lecture Notes in Computer Science*, pages 389–405, 2008.
3. D. J. Bernstein, P. Birkner, T. Lange, and C. Peters. ECM using edwards curves. 2008.
4. D. J. Bernstein and T. Lange. Faster addition and doubling on elliptic curves. In K. Kurosawa, editor, *Advances in Cryptology: Proceedings of ASIACRYPT 2007,* Kuching, Sarawak, Malaysia, volume 4833 of *Lecture Notes in Computer Science*, pages 29–50, June 2007.
5. D. J. Bernstein and T. Lange. Inverted edwards coordinates. In S. Boztas and H. feng Lu, editors, *Applied Algebra, Algebraic Algorithms and Error-Correcting Codes, 17th International Symposium, AAECC-17, Bangalore, India, December 16-20, 2007, Proceedings*, volume 4851 of *Lecture Notes in Computer Science*, pages 20–27, 2007.
6. H. M. Edwards. A normal form for elliptic curves. *Bulletin of the American Mathematical Society*, 44(3):393–422, July 2007.
7. É. Brier and M. Joye. Weierstraß elliptic curves and side-channel attacks. In D. Naccache and P. Paillier, editors, *Public Key Cryptography*, number 2274 in Lecture Notes in Computer Science, pages 183–194, Berlin, Heidelberg, 2002. Springer-Verlag.
8. Information Technology Laboratory. Fips 186-3: Digital signature standard (dss). Technical report, National Institute of Standards and Technology, June 2009.
9. H. W. Lenstra, Jr. Factoring integers with elliptic curves. *Annals of Mathematics*, 126:649–673, 1987.
10. P. L. Montgomery. Speeding the pollard and elliptic curve methods of factorization. *Mathematics of Computation*, 48(177):243–264, January 1987.
11. K. Okeya and K. Sakurai. Efficient elliptic curve cryptosystem from a scalar multiplication algorithm with recovery of the y-coordinate on a montgomery-form elliptic curve. In Ç. K. Koç, D. Naccache, and C. Paar, editors, *Cryptographic Hardware and Embedded Systems, Workshop, CHES'01,* Paris, France, number 2162 in Lecture Notes in Computer Science, pages 126–141, Berlin, Heidelberg, 2001. Springer-Verlag.
12. J. M. Pollard. Theorems on factorization and primality testing. *Proceedings of the Cambridge Philosophical Society*, 76:521–528, 1974.