Protecting ECC Against Fault Attacks: The Ring Extension Method Revisited

Marc Joye

OneSpan
Brussels, Belgium
marc.joye@onespan.com

Abstract. Due to its shorter key size, elliptic curve cryptography (ECC) is gaining more and more popularity. However, if not properly implemented, the resulting cryptosystems may be susceptible to fault attacks. Over the past few years, several techniques for secure implementations have been published. This paper revisits the ring extension method and its adaptation to the elliptic curve setting.

Keywords: Elliptic curves; formal groups; degenerate curves, elliptic curve cryptosystems; fault attacks; countermeasures.

1 Introduction

This paper deals with secure implementations [24] for ECC-based cryptosystems [45,50,49,10,11,20] and, more specifically, with the development of efficient detection methods against fault attacks (or errors) [14]. Practical ways to mount fault attacks are surveyed in [4,27]. See also [41, Part III] for a more recent and complete account.

Fault Attacks and Countermeasures. A fault attack disturbs the expected behavior of a security device and makes it work abnormally so as to infer sensitive data. Since their discovery in 1997, several countermeasures were proposed. The key principle consists in computing a sensitive operation in a redundant way or in exploiting some redundancy already present in the calculation.

Shamir's countermeasure. Most known countermeasures rely on an elegant method first suggested by Shamir [58] for RSA [55] using Chinese remaindering [53]. These include [2,63,12,18,43,61] to name a few.

We follow the general presentation of [38]. Consider the ring $\mathbb{Z}/N\mathbb{Z}$ of integers modulo N where $N = pq$ is the product of two large primes. On input an element $x \in \mathbb{Z}/N\mathbb{Z}$ (for example, x is a ciphertext or the hash value of a message) and a private exponent d, the goal is to compute an RSA exponentiation, $y = x^d \bmod N$, *in the presence of faults*. In order to prevent fault attacks, the evaluation of $y = x^d \bmod N$ is carried out in three steps as follows.

1. Compute $\hat{y} = x^d \bmod rN$ for a (small) integer r;
2. Compute $y' = x^d \bmod r$;
3. Check whether $\hat{y} \equiv y' \pmod{r}$, and
   - if so, output $y = \hat{y} \bmod N$;
   - if not, return `error`.

Shamir's method is an application of the Chinese remainder theorem (CRT). We obviously have $\hat{y} \equiv y \pmod{N}$ and $\hat{y} \equiv y' \pmod{r}$ when the computations are not faulty. In the presence of random faults, the probability that $\hat{y} \equiv y \pmod{r}$ is about $1/r$. Larger values for r imply a higher detection probability, but at the expense of more demanding computations.

VIGILANT'S COUNTERMEASURE. Another method was proposed at CHES 2008 by Vigilant [61]. Again, the goal is to perform a private RSA exponentiation, $y = x^d \bmod N$, in the presence of faults. The method goes as follows.

1. Form $X = \mathrm{CRT}\big(x \pmod N, 1 + r \pmod{r^2}\big)$ for a (small) integer r;
2. Compute $\hat{y} = X^d \bmod r^2 N$;
3. Check whether $\hat{y} \equiv 1 + dr \pmod{r^2}$, and
   - if so, output $y = \hat{y} \bmod N$;
   - if not, return `error`.

In Step 1, $\mathrm{CRT}(\cdot, \cdot)$ denotes an application of the Chinese remainder theorem; namely, the so-constructed X satisfies $X \equiv x \pmod N$ and $X \equiv 1 + r \pmod{r^2}$.

Remark 1. When Vigilant's method is applied to RSA with Chinese remaindering, special care needs to exercised. A number of potential fault attacks against RSA-CRT are presented in [21]; implementation recommendations are also provided.

Elliptic Curve Cryptography. Elliptic curve cryptography [45,50] is an interesting alternative to RSA because the keys are much shorter for a same conjectured security level. Given a point $\boldsymbol{P}$ on an elliptic curve E and a private integer d, the basic operation consists in computing the scalar multiplication $[d]\boldsymbol{P}$, that is, $\boldsymbol{P} \boxplus \boldsymbol{P} \boxplus \cdots \boxplus \boldsymbol{P}$ (d times) where $\boxplus$ denotes the group operation on E. The goal of an attacker is to recover the value of d (or a part thereof) by inducing faults. See [8,1,17,13,57,42,44,51] for examples of fault attacks against elliptic-curve cryptosystems.

Our Contributions. Vigilant's method presents a couple of advantages over Shamir's method. In particular, it trades the small exponentiation $y' = x^d \bmod r$ against the multiplication $1 + dr \bmod r^2 = 1 + r \cdot (d \bmod r)$ in the verification step. This latter operation is much faster. We note however that the evaluation of y' in Shamir's method can be sped up as $x^{d \bmod \varphi(r)} \bmod r$ (where φ denotes Euler's totient function), provided that the value of $\varphi(r)$ is known. The correctness of Vigilant's countermeasure can be seen as a consequence of the binomial theorem. This latter states that $(1+r)^d = \sum_{j=0}^{d} \binom{d}{j} 1^{d-j}\, r^j = \sum_{j=0}^{d} \binom{d}{j} r^j = 1 + dr + \frac{d(d-1)}{2} r^2 + \cdots$. Reducing this identity modulo r^2 yields $(1 + r)^d \equiv 1 + dr \pmod{r^2}$. Hence, since by construction $X \equiv 1 + r \pmod{r^2}$, the following relation holds modulo r^2:

$$\hat{y} \equiv X^d \equiv (1 + r)^d \equiv 1 + dr \pmod{r^2} \ .$$

Shamir's countermeasure generalizes to the elliptic curve scalar multiplication (cf. Section 2). In contrast, Vigilant's method does not readily lend itself to a generalization to elliptic curves. The reason is that there is no equivalent of the binomial theorem. We adopt a different approach and rely on the theory of *formal groups* as put forward in [26] for developing elliptic curve Paillier schemes. Doing so, we obtain a first efficient and versatile method to protect elliptic curve cryptosystems against fault attacks.

It is well known that the singular elliptic curve over the finite prime field $\mathbb{F}_r$ given by the Weierstraß equation $y^2 = x^3$ is isomorphic to the *additive* group $\mathbb{F}_r^+$; e.g., [34, Theorem 7.2]. Neves–Tibouchi [51] take advantage of this property to propose an efficient protection against fault attacks. They also extend their method to other models (including Edwards curves) with [less efficient] multiplicative isomorphisms. As a second contribution, we exhibit efficiently computable isomorphisms to the additive group $(\mathbb{Z}/r\mathbb{Z})^+$ for all elliptic curve models commonly used in cryptographic applications. This results in a second efficient and versatile method to protect against fault attacks.

Organization. The rest of this paper is organized as follows. In the next section, we review variants of Shamir's countermeasure applied to ECC systems. Section 3 describes our general methodology for detecting faults with two possible realizations. Next, in Section 4, we apply it to a variety of elliptic curve models. Finally, we conclude the paper in Section 5.

2 Overcoming Fault Attacks

Shamir's method generalizes to the elliptic curve scalar multiplication. We review hereafter two different implementations. The first countermeasure is due to Blömer–Otto–Seifert and known as the BOS countermeasure [13] while the second one is due to Baek–Vasyltsov [3].

BOS COUNTERMEASURE. As aforementioned, the main operation for elliptic curve cryptography is the scalar multiplication. Specifically, the usual setting is the computation of $\boldsymbol{Q} = [d]\boldsymbol{P}$ on an elliptic curve E defined over the prime field $\mathbb{F}_p$, which is given by the Weierstraß equation $E\colon y^2 = x^3 + ax + b$. The BOS countermeasure proceeds in five steps:

1. For a (small) *prime* r, define an elliptic curve E' over $\mathbb{F}_r$ and a point $\boldsymbol{P'}$ on E';
2. Form the combined curve $\hat{E} = \mathrm{CRT}(E, E')$ over $\mathbb{Z}/pr\mathbb{Z}$ and the combined point $\hat{\boldsymbol{P}} = \mathrm{CRT}(\boldsymbol{P}, \boldsymbol{P'})$;
3. Compute $\hat{\boldsymbol{Q}} = [d]\hat{\boldsymbol{P}}$ on $\hat{E}$;
4. Compute $\boldsymbol{Q'} = [d]\boldsymbol{P'}$ on E';
5. Check whether $\hat{\boldsymbol{Q}} \equiv \boldsymbol{Q'} \pmod{r}$, and
   - if so, output $\boldsymbol{Q} = \hat{\boldsymbol{Q}} \bmod p$;
   - if not, return `error`.

Remark 2. If $y^2 = x^3 + a'x + b'$ is the equation defining the elliptic curve E' over $\mathbb{F}_r$, $\mathrm{CRT}(E, E')$ denotes the elliptic curve over $\mathbb{Z}/pr\mathbb{Z}$ given by the equation $y^2 = x^3 + \hat{a}x + \hat{b}$ where $\hat{a} = \mathrm{CRT}(a \pmod{p}, a' \pmod{r})$ and $\hat{b} = \mathrm{CRT}(b \pmod{p}, b' \pmod{r})$; i.e., such $\hat{a} \equiv a \pmod{p}$ and $\hat{a} \equiv a' \pmod{r}$, and idem for $\hat{b}$. Point $\hat{\boldsymbol{P}}$ is defined similarly from the coordinates of points $\boldsymbol{P}$ and $\boldsymbol{P'}$.

In a concrete implementation, prime r, curve E' and point $\boldsymbol{P'}$ are precomputed so that the order of point $\boldsymbol{P'}$ on E', $\mathrm{ord}_{E'}(\boldsymbol{P'})$, is maximal. The value of $n' := \mathrm{ord}_{E'}(\boldsymbol{P'})$ together with r, the curve parameters and point $\boldsymbol{P'}$ are stored in non-volatile memory. This presents the further advantage that the computation of $\boldsymbol{Q'}$ in Step 4 can be performed more efficiently as $\boldsymbol{Q'} = [d \bmod n']\boldsymbol{P'}$.

BAEK–VASYLTSOV'S COUNTERMEASURE. Another variant of Shamir's countermeasure was subsequently developed in [3]. Compared to the BOS countermeasure, in a practical setting, it does not require precomputed values and does not assume that the parameter r is prime.

Numerical experiments conducted in [36] however show that a non-negligible proportion of faults is undetected and that larger bit-lengths for r should be used. For example, for a 20-bit randomizer r, the average proportion of undetected faults ranges from 23.2% to 37.3%. Moreover, by construction, Baek–Vasyltsov's countermeasure is restricted to a special Weierstraß model and makes use of less efficient addition formulas.

3 The Ring Extension Method Revisited

In a way similar to Vigilant's countermeasure for RSA, the adaptation of Shamir's method to elliptic curves can be improved by finding a shortcut in the evaluation of $\boldsymbol{Q'} = [d]\boldsymbol{P'}$ on E' by an appropriate choice for E' in the BOS countermeasure. Further, for more versatility and better efficiency, it should work for any randomizer r (i.e., not only prime values) and without the need of pre-computing and pre-storing curve orders.

The core idea is to replace in the BOS countermeasure the combined curve $\hat{E}$ with

$$E(\mathbb{F}_p) \times \mathbb{G}' \cong E(\mathbb{F}_p) \times (\mathbb{Z}/r\mathbb{Z})^+ ,$$

that is, a group isomorphic to the cross product of the groups $E(\mathbb{F}_p)$ and $(\mathbb{Z}/r\mathbb{Z})^+$ and where the group $\mathbb{G}'$ is represented with elements having a group law that *coincides* (i.e., is compatible) with the group law used in the representation of $E(\mathbb{F}_p)$.

We present two such realizations. In the first realization, $\mathbb{G}'$ is chosen as the subgroup of points on an elliptic curve over $\mathbb{Z}/r^2\mathbb{Z}$ that reduce to the neutral point modulo r [37]. The second realization modifies a recent countermeasure due to Neves and Tibouchi [51, § 5]. The proposed methods are generic and can readily be adapted to any elliptic curve model and corresponding addition formulas. Also, although focusing on protecting elliptic curve computations over prime fields for the sake of concreteness, they can be generalized to elliptic curve computations over arbitrary fields, including over binary fields.

3.1 First Realization

It is useful to introduce some notation. Given a commutative ring $\mathcal{R}$ with 1, we let $E(\mathcal{R})$ denote the set of rational points on an elliptic curve E defined over $\mathcal{R}$.

For the ring $\mathcal{R} = \mathbb{Z}/r^2\mathbb{Z}$ (namely, the ring of integers modulo r^2), we define the order-r subgroup

$$\mathbb{G}' \coloneqq E_1(\mathbb{Z}/r^2\mathbb{Z}) = \left\{\boldsymbol{P} \in E(\mathbb{Z}/r^2\mathbb{Z}) \mid \boldsymbol{P} \text{ modulo } r \text{ reduces to } \boldsymbol{O}\right\}$$

where $\boldsymbol{O}$ denotes the identity element on $E(\mathbb{Z}/r\mathbb{Z})$. The analogue of the combined curve $\hat{E}$ becomes

$$E(\mathbb{F}_p) \times E_1(\mathbb{Z}/r^2\mathbb{Z}) \subseteq E(\mathbb{Z}/pr^2\mathbb{Z}) \ .$$

As will be made explicit in Section 4, the so-defined group $\mathbb{G}'$ is isomorphic to $(\mathbb{Z}/r\mathbb{Z})^+$ and the isomorphisms

$$\Upsilon_1 \colon (\mathbb{Z}/r\mathbb{Z})^+ \xrightarrow{\sim} E_1(\mathbb{Z}/r^2\mathbb{Z}), \begin{cases} 0 \longmapsto \Upsilon(0) = \boldsymbol{O} \\ \vartheta \longmapsto \Upsilon(\vartheta) = \boldsymbol{P} \end{cases}$$

and Υ_1^{-1} are efficiently computable.

3.2 Second Realization

The authors of [51] suggest to choose $\mathbb{G}'$ as the group of points on a degenerate curve over $\mathbb{F}_r$. However, most elliptic curve models (the Weierstraß model is a notable exception) do not have an additive degeneration: they either degenerate to the $(r-1)$-order multiplicative group $\mathbb{F}_r^*$ or to the $(r+1)$-order multiplicative subgroup $T_2(\mathbb{F}_r)$ of elements of norm 1 in $\mathbb{F}_{r^2}^*$ [56]. In this case, the shortcut function translates into an exponentiation modulo r (degeneration to $\mathbb{F}_r^*$) or into the evaluation of Lucas sequences modulo r (degeneration to $T_2(\mathbb{F}_r)$).

Actually, it turns out that we can always identify a group $\mathbb{G}' \cong (\mathbb{Z}/r\mathbb{Z})^+$ from the group law in E for a particular choice for the curve parameters; we call E' the corresponding curve. We so define the r-order group

$$\mathbb{G}' \coloneqq E'(\mathbb{Z}/r\mathbb{Z})[r] = \left\{\boldsymbol{P} \text{ satisfying the curve equation } E' \text{ modulo } r \mid [r]\boldsymbol{P} = \boldsymbol{O}\right\}$$

for the particular curve equation E'. Again, this comes with efficiently computable isomorphisms

$$\Upsilon_2 \colon (\mathbb{Z}/r\mathbb{Z})^+ \xrightarrow{\sim} E'(\mathbb{Z}/r\mathbb{Z})[r], \begin{cases} 0 \longmapsto \Upsilon(0) = \boldsymbol{O} \\ \vartheta \longmapsto \Upsilon(\vartheta) = \boldsymbol{P} \end{cases}$$

and Υ_2^{-1}. This will be illustrated in Section 4 with several elliptic curve models commonly used in cryptographic applications.

3.3 Implementation

The computation of $\boldsymbol{Q} = [d]\boldsymbol{P}$ on an elliptic curve $E(\mathbb{F}_p)$ in the presence of faults can be carried out as depicted in Algorithms 1 and 2. Algorithm 1 corresponds to the first realization and Algorithm 2 corresponds to the second realization.

Algorithm 1: Fault-protected scalar multiplication on elliptic curves (1)

Data: Point $\boldsymbol{P} \in E(\mathbb{F}_p)$ and private scalar $d \in \mathbb{Z}$
Result: Point $\boldsymbol{Q} = [d]\boldsymbol{P} \in E(\mathbb{F}_p)$ or "`error`"

1 Randomly select a small integer r and define the point $\boldsymbol{P'} \leftarrow \Upsilon_1(\vartheta) \in E(\mathbb{Z}/r^2\mathbb{Z})$ for some $\vartheta \in \mathbb{Z}/r\mathbb{Z}$;
2 Form the point $\hat{\boldsymbol{P}} \leftarrow \mathrm{CRT}(\boldsymbol{P}, \boldsymbol{P'}) \in E(\mathbb{Z}/pr^2\mathbb{Z})$;
3 Compute $\hat{\boldsymbol{Q}} \leftarrow [d]\hat{\boldsymbol{P}} \in E(\mathbb{Z}/pr^2\mathbb{Z})$;
4 Compute $\boldsymbol{Q'} \leftarrow \Upsilon_1(d \cdot \vartheta \bmod r) \in E(\mathbb{Z}/r^2\mathbb{Z})$;
5 If $(\hat{\boldsymbol{Q}} \bmod r^2) \neq \boldsymbol{Q'}$ return "`error`";
6 Return $\hat{\boldsymbol{Q}} \bmod p$.

Algorithm 2: Fault-protected scalar multiplication on elliptic curves (2)

Data: Point $\boldsymbol{P} \in E(\mathbb{F}_p)$ and private scalar $d \in \mathbb{Z}$
Result: Point $\boldsymbol{Q} = [d]\boldsymbol{P} \in E(\mathbb{F}_p)$ or "`error`"

1 Randomly select a small integer r and define the point $\boldsymbol{P'} \leftarrow \Upsilon_2(\vartheta) \in E'(\mathbb{Z}/r\mathbb{Z})$ for some $\vartheta \in \mathbb{Z}/r\mathbb{Z}$;
2 Form the curve equation $\hat{E} \leftarrow \mathrm{CRT}(E, E')$ and point $\hat{\boldsymbol{P}} \leftarrow \mathrm{CRT}(\boldsymbol{P}, \boldsymbol{P'}) \in \hat{E}(\mathbb{Z}/pr\mathbb{Z})$;
3 Compute $\hat{\boldsymbol{Q}} \leftarrow [d]\hat{\boldsymbol{P}} \in \hat{E}(\mathbb{Z}/pr\mathbb{Z})$;
4 Compute $\boldsymbol{Q'} \leftarrow \Upsilon_2(d \cdot \vartheta \bmod r) \in E'(\mathbb{Z}/r\mathbb{Z})$;
5 If $(\hat{\boldsymbol{Q}} \bmod r) \neq \boldsymbol{Q'}$ return "`error`";
6 Return $\hat{\boldsymbol{Q}} \bmod p$.

Implementation notes. It is worth noting that in the "redundancy" step (i.e., Step 4 in Algorithms 1 and 2) the resulting point $\boldsymbol{Q'} = [d \bmod r]\Upsilon_1(\vartheta)$ (resp. $\boldsymbol{Q'} = [d \bmod r]\Upsilon_2(\vartheta)$) is viewed as an element of $\mathbb{G}' = E_1(\mathbb{Z}/r^2\mathbb{Z})$ (resp. of $\mathbb{G}' = E'(\mathbb{Z}/r\mathbb{Z})[r]$). This is much faster than computing a scalar multiplication in $E(\mathbb{Z}/r^2\mathbb{Z})$ (resp. in $E'(\mathbb{Z}/r\mathbb{Z})$). This also allows the reduction of d modulo r, the group order of $\mathbb{G}'$.

Furthermore, single points of failure like conditional branchings should be avoided in fault-resistant implementations. The verification step (i.e., Step 5 in Algorithms 1 and 2) involves an **if**-branching. By inducing a fault during the comparison

$$(\hat{\boldsymbol{Q}} \bmod r^2) \stackrel{?}{\neq} \boldsymbol{Q'} \qquad (\text{resp. } (\hat{\boldsymbol{Q}} \bmod r) \stackrel{?}{\neq} \boldsymbol{Q'}),$$

an attacker may hope to force the comparison bit to 0 (i.e., `false`) and therefore get the value of $\hat{\boldsymbol{Q}} \bmod p$. The **if**-branching can however be avoided by making use of the so-called "infective computation" technique [63].

For better fault coverage, it is recommended to choose ϑ in Step 1 of Algorithms 1 and 2 so that $\boldsymbol{P'}$ is of maximal order (i.e., of order r). Obtaining a generator for the additive group $(\mathbb{Z}/r\mathbb{Z})^+$ is fairly easy since any non-zero integer co-prime to r generates $(\mathbb{Z}/r\mathbb{Z})^+$. Two possible strategies are:

1. Take 1 as a generator or fix *a priori* a prime ϑ larger than the maximum value for r. Then $(\mathbb{Z}/r\mathbb{Z})^+ = \langle 1 \rangle$ or $(\mathbb{Z}/r\mathbb{Z})^+ = \langle \vartheta \rangle$.
2. Select r as a prime number. Then any non-zero integer $0 < \vartheta < r$ is a generator of $(\mathbb{Z}/r\mathbb{Z})^+$.

The first strategy is preferred as it does not impose conditions on r.

4 Illustration

The proposed methods apply to many elliptic curve models. This is illustrated below with the (twisted) Edwards model and the Weierstraß model. Applications to further models can be found in Appendix A.

4.1 Edwards Model

In [23], Edwards proposed a normal form for elliptic curves. It was later extended in [7] and subsequently in [5] (see also [30]). The latter form, referred to as the *twisted Edwards form*, is given by the equation

$$E_{\mathcal{E}_{a,d}}\colon ax^2 + y^2 = 1 + dx^2y^2 \ . \tag{1}$$

The neutral element is $\boldsymbol{O} = (0, 1)$. The addition law is unified. Given two points (x_1, y_1) and (x_2, y_2), their sum $(x_3, y_3) = (x_1, y_1) \boxplus (x_2, y_2)$ is given by

$$(x_3, y_3) = \left(\frac{x_1y_2 + x_2y_1}{1 + dx_1x_2y_1y_2}, \frac{y_1y_2 - ax_1x_2}{1 - dx_1x_2y_1y_2} \right) \ . \tag{2}$$

First Realization. We define:

$$E_{\mathcal{E}_{a,d},1}(\mathbb{Z}/r^2\mathbb{Z}) = \left\{ \Upsilon_1(\vartheta) = (\vartheta \cdot r, 1) \mid \vartheta \in \mathbb{Z}/r\mathbb{Z} \right\} \cong (\mathbb{Z}/r\mathbb{Z})^+ \ . \tag{3}$$

In words, the group $\mathbb{G}' = E_{\mathcal{E}_{a,d},1}(\mathbb{Z}/r^2\mathbb{Z})$ is the set of points $(x, 1) = (\vartheta \cdot r, 1)$ on an Edwards curve (1) over the ring $\mathbb{Z}/r^2\mathbb{Z}$, equipped with the addition law (2) . It is easily verified that:

1. $\Upsilon_1(\vartheta) \equiv (\vartheta \cdot r, 1) \equiv (0, 1) \equiv \Upsilon_1(0) \equiv \boldsymbol{O} \pmod{r}$, and
2. $\Upsilon_1(\vartheta_1) \boxplus \Upsilon_1(\vartheta_2) = (\vartheta_1 \cdot r, 1) \boxplus (\vartheta_2 \cdot r, 1) = \left(\frac{\vartheta_1 \cdot r \cdot 1 + \vartheta_2 \cdot r \cdot 1}{1}, \frac{1 \cdot 1}{1} \right) = \left((\vartheta_1 + \vartheta_2) \cdot r, 1 \right)$
$= \Upsilon_1(\vartheta_1 + \vartheta_2)$

as desired. We also have $E_{\mathcal{E}_{a,d},1}(\mathbb{Z}/r^2\mathbb{Z}) = \langle (r, 1) \rangle$.

Second Realization. We have:

$$(\mathbb{Z}/r\mathbb{Z})^+ \cong E'_{\mathcal{E}_{0,0)}}(\mathbb{Z}/r\mathbb{Z})[r] = \left\{ \Upsilon_2(\vartheta) = (\vartheta, 1) \mid \vartheta \in \mathbb{Z}/r\mathbb{Z} \right\} \subseteq \left\{ (x, y) \in E'_{\mathcal{E}_{0,0}}(\mathbb{Z}/r\mathbb{Z}) \right\} \ . \tag{4}$$

In more detail, the group $\mathbb{G}' = E'_{\mathcal{E}_{0,0)}}(\mathbb{Z}/r\mathbb{Z})[r]$ is the set of points $(x, 1)$ on an Edwards curve E' given by Eq. (1) with parameters $a = d = 0$, over the ring $\mathbb{Z}/r\mathbb{Z}$, equipped with the addition law (2). When $a = d = 0$, it immediately follows that:

1. $\Upsilon_2(0) = (0, 1) = \boldsymbol{O}$, and
2. $\Upsilon_2(\vartheta_1) \boxplus \Upsilon(\vartheta_2) = (\vartheta_1, 1) \boxplus (\vartheta_2, 1) = \left(\frac{\vartheta_1 \cdot 1 + \vartheta_2 \cdot 1}{1}, \frac{1 \cdot 1}{1} \right) = (\vartheta_1 + \vartheta_2, 1) = \Upsilon_2(\vartheta_1 + \vartheta_2)$;

and $E'_{\mathcal{E}_{0,0)}}(\mathbb{Z}/r\mathbb{Z})[r] = \langle (1, 1) \rangle$.

4.2 Weierstraß Model

The *Weierstraß model* (e.g., [59, Chapter III]) is the most common way to represent an elliptic curve. It is given by the equation $E_{\mathcal{W}_{a,b}}\colon y^2 = x^3 + ax + b$ or, using projective coordinates,

$$E_{\mathcal{W}_{a,b}}\colon Y^2Z = X^3 + aXZ^2 + bZ^3 \ . \tag{5}$$

The neutral element is the point at infinity $\boldsymbol{O} = (0 : 1 : 0)$. A unified addition formula [46, Section 3] (see also [47,15,35,54]) for adding two projective points $(X_1 : Y_1 : Z_1)$ and $(X_2 : Y_2 : Z_2)$ is given by $(X_3 : Y_3 : Z_3) = (X_1 : Y_1 : Z_1) \boxplus (X_2 : Y_2 : Z_2)$ where

$$\begin{cases} X_3 = (Y_1Z_2 + Y_2Z_1)A + (X_1Y_2 + X_2Y_1)B \\ Y_3 = (X_1Z_2 + X_2Z_1)M + (Y_1Y_2 + 3bZ_1Z_2)(Y_1Y_2 - 3bZ_1Z_2) - aN \\ Z_3 = (X_1Y_2 + X_2Y_1)(aZ_1Z_2 + 3X_1X_2) + (Y_1Z_2 + Y_2Z_1)V \end{cases} \tag{6}$$

with $A = a(aZ_1Z_2 - X_1X_2) - 3b(X_1Z_2 + X_2Z_1)$, $B = Y_1Y_2 - a(X_1Z_2 + X_2Z_1) - 3bZ_1Z_2$, $M = 3b(3X_1X_2 - aZ_1Z_2) - a^2(X_1Z_2 + X_2Z_1)$, $N = (aZ_1Z_2 + 3X_1X_2)(aZ_1Z_2 - X_1X_2)$, and $V = Y_1Y_2 + 3bZ_1Z_2 + a(X_1Z_2 + X_2Z_1)$. As detailed in [54, Algorithm 1], this can be evaluated with only 12 general multiplications plus 5 multiplications by constants (namely, a and $3b$).

First Realization. We define:

$$E_{\mathcal{W}_{a,b},1}(\mathbb{Z}/r^2\mathbb{Z}) = \left\{\Upsilon_1(\vartheta) = (\vartheta \cdot r : 1 : 0) \mid \vartheta \in \mathbb{Z}/r\mathbb{Z}\right\} \cong (\mathbb{Z}/r\mathbb{Z})^+ \ . \tag{7}$$

Here again, it can be verified that $\Upsilon_1(\vartheta) \equiv (\vartheta \cdot r : 1 : 0) \equiv (0 : 1 : 0) \equiv \Upsilon_1(0) \equiv \boldsymbol{O} \pmod{r}$ and that the addition formula (6) yields $\Upsilon_1(\vartheta_1) \boxplus \Upsilon_1(\vartheta_2) = (\vartheta_1 \cdot r : 1 : 0) \boxplus (\vartheta_2 \cdot r : 1 : 0) = \big((\vartheta_1 + \vartheta_2) \cdot r : 1 : 0\big) = \Upsilon_1(\vartheta_1 + \vartheta_2)$ by observing that we then have $A = 0$, $B = 1$, $M = N = 0$, $V = 1$ and thus $(X_3 : Y_3 : Z_3) = \big((\vartheta_1 \cdot r \cdot 1 + \vartheta_2 \cdot r \cdot 1) \cdot 1 : 1 \cdot 1 : (\vartheta_1 \cdot r \cdot 1 + \vartheta_2 \cdot r \cdot 1) \cdot 0 + 0 \cdot 1\big) = \big((\vartheta_1 + \vartheta_2) \cdot r : 1 : 0\big)$. We also have $E_{\mathcal{W}_{a,b},1}(\mathbb{Z}/r^2\mathbb{Z}) = \langle (r : 1 : 0) \rangle$.

Second Realization. We have:

$$(\mathbb{Z}/r\mathbb{Z})^+ \cong E'_{\mathcal{W}_{0,0}}(\mathbb{Z}/r\mathbb{Z})[r] = \left\{\Upsilon_2(\vartheta) = (\vartheta : 1 : \vartheta^3) \mid \vartheta \in \mathbb{Z}/r\mathbb{Z}\right\} \subseteq \left\{(X : Y : Z) \in E'_{\mathcal{W}_{0,0}}(\mathbb{Z}/r\mathbb{Z})\right\} \ . \tag{8}$$

Similarly to the first realization, it can be verified that $\Upsilon_2(0) = (0 : 1 : 0) = \boldsymbol{O}$ and, when $a = b = 0$, that the addition formula (6) yields $\Upsilon_2(\vartheta_1) \boxplus \Upsilon_2(\vartheta_2) = (\vartheta_1 : 1 : {\vartheta_1}^3) \boxplus (\vartheta_2 : 1 : {\vartheta_2}^3) = \big(\vartheta_1 + \vartheta_2 : 1 : (\vartheta_1 + \vartheta_2)3\vartheta_1\vartheta_2 + {\vartheta_1}^3 + {\vartheta_2}^3\big) = \big(\vartheta_1 + \vartheta_2 : 1 : (\vartheta_1 + \vartheta_2)^3\big) = \Upsilon_2(\vartheta_1 + \vartheta_2)$; and $E'_{\mathcal{W}_{0,0}}(\mathbb{Z}/r\mathbb{Z})[r] = \langle (1 : 1 : 1) \rangle$.

4.3 Comparison

The proposed methods share the advantages (but not the weaknesses!) of the Baek–Vasyltsov's countermeasure: they do not require pre-computed values in some non-volatile memory and do not suppose randomizer r to be prime. Furthermore, they are more general as they are not restricted to a special type of Weierstraß parametrization.

Another advantage of the proposed methods is that they carry the completeness of the addition law, whatever the choice of the parameters. For example, for twisted Edwards curves, completeness is guaranteed provided that curve parameter a is a square and curve parameter d is a non-square. Further, twisted Edwards curves as given by Eq. (1) do not hold in characteristic 2. There are no such restrictions on $\mathbb{G}' = E_{\mathcal{E}_{a,d},1}(\mathbb{Z}/r^2\mathbb{Z})$ (cf. (3)) or on $\mathbb{G}' = E'_{\mathcal{E}_{0,0}}(\mathbb{Z}/r\mathbb{Z})[r]$ (cf. (4)). Indeed, for any points $\boldsymbol{P_1} = (\vartheta_1 \cdot r, 1)$ and $\boldsymbol{P_2} = (\vartheta_2 \cdot r, 1)$ in $E_{\mathcal{E}_{a,d},1}(\mathbb{Z}/r^2\mathbb{Z})$ (resp. $\boldsymbol{P_1} = (\vartheta_1, 1)$ and $\boldsymbol{P_2} = (\vartheta_2, 1) \in E'_{\mathcal{E}_{0,0}}(\mathbb{Z}/r\mathbb{Z})[r]$), the addition formula given by Eq. (2) remains always valid since the denominators always are equal to 1 and thus are invertible modulo r^2 (resp. modulo r), including for even values for r. The same conclusion holds true for the completeness of the Weierstraß model as described in § 4.2 and the other models given in appendix.

Furthermore, unlike the BOS countermeasure, the order of the small curve is known in advance: by construction, we have $\#\mathbb{G}' \cong (\mathbb{Z}/r\mathbb{Z})^+$. Scalar d in the computation of $\boldsymbol{Q'}$ in $E_1(\mathbb{Z}/r^2\mathbb{Z})$ (resp. $E'(\mathbb{Z}/r\mathbb{Z})$) can therefore be reduced modulo r. Because the BOS countermeasure makes use of general groups of points on an elliptic curve, the group order is not so easily obtained; this is addressed by fixing randomizer r once and for all and by pre-computing (and storing) the group order for the curve modulo r. In our case, randomizer r can be freely selected on the fly, with a fresh value for each execution. In addition to better efficiency and easier implementation, this offers better security guarantees and fault coverage.

Finally, the verification step essentially boils down to a mere modular multiplication modulo r rather than a full scalar multiplication on an elliptic curve.

5 Conclusion

This paper revisited the ring extension method over elliptic curves as presented in [13,3]. The proposed approaches apply to a variety of elliptic models and provide more practical countermeasures against fault attacks.

References

1. Antipa, A., Brown, D.R.L., Menezes, A., Struik, R., Vanstone, S.A.: Validation of elliptic curve public keys. In: Desmedt, Y. (ed.) Public Key Cryptography − PKC 2003. Lecture Notes in Computer Science, vol. 2567, pp. 211–223. Springer (2003). doi:10.1007/3-540-36288-6_16
2. Aumüller, C., Bier, P., Fischer, W., Hofreiter, P., Seifert, J.P.: Fault attacks on RSA with CRT: Concrete results and practical countermeasures. In: Kaliski Jr., B.S., Koç, Ç.K., Paar, C. (eds.) Cryptographic Hardware and Embedded Systems − CHES 2002. Lecture Notes in Computer Science, vol. 2523, pp. 260–275. Springer (2002). doi:10.1007/3-540-36400-5_20
3. Baek, Y.J., Vasyltsov, I.: How to prevent DPA and fault attacks in a unified way for ECC scalar multiplication: Ring extension method. In: Dawson, E., Wong, D. (eds.) Information Security Practice and Experience − ISPEC 2007. Lecture Notes in Computer Science, vol. 4464, pp. 225–237. Springer (2007). doi:10.1007/978-3-540-72163-5_18
4. Bar-El, H., Choukri, H., Naccache, D., Tunstall, M., Whelan, C.: The sorcerer's apprentice guide to fault attacks. Proceedings the IEEE **94**(2), 370–382 (2006). doi:10.1109/JPROC.2005.862424
5. Bernstein, D.J., Birkner, P., Joye, M., Lange, T., Peters, C.: Twisted Edwards curves. In: Vaudenay, S. (ed.) Progress in Cryptology − AFRICACRYPT 2008. Lecture Notes in Computer Science, vol. 5023, pp. 389–405. Springer (2008). doi:10.1007/978-3-540-68164-9_26
6. Bernstein, D.J., Chuengsatiansup, C., Kohel, D., Lange, T.: Twisted Hessian curves. In: Lauter, K.E., Rodrìguez-Henrìquez, F. (eds.) Progress in Cryptology − LATINCRYPT 2015. Lecture Notes in Computer Science, vol. 9230, pp. 269–294. Springer (2015). doi:10.1007/978-3-319-22174-8_15
7. Bernstein, D.J., Lange, T.: Faster addition and doubling on elliptic curves. In: Kurosawa, K. (ed.) Advances in Cryptology − ASIACRYPT 2007. Lecture Notes in Computer Science, vol. 4833, pp. 29–50. Springer (2007). doi:10.1007/978-3-540-76900-2_3
8. Biehl, I., Meyer, B., Müller, V.: Differential fault attacks on elliptic curve cryptosystems. In: Bellare, M. (ed.) Advances in Cryptology − CRYPTO 2000. Lecture Notes in Computer Science, vol. 1880, pp. 131–146. Springer (2000). doi:10.1007/3-540-44598-6_8
9. Billet, O., Joye, M.: The Jacobi model of an elliptic curve and side-channel analysis. In: Fossorier, M., Høholdt, T., Poli, A. (eds.) Applied Algebra, Algebraic Algorithms and Error-Correcting Codes. Lecture Notes in Computer Science, vol. 2643, pp. 34–42. Springer (2003). doi:10.1007/3-540-44828-4_5
10. Blake, I., Seroussi, G., Smart, N.: Elliptic Curves in Cryptography. No. 265 in London Mathematical Society Lecture Note Series, Cambridge University Press (1999). doi:10.1017/CBO9781107360211
11. Blake, I.F., Seroussi, G., Smart, N.P. (eds.): Advances in Elliptic Curve Cryptography, London Mathematical Society Lecture Note Series, vol. 317. Cambridge University Press (2005). doi:10.1017/CBO9780511546570
12. Blömer, J., Otto, M., Seifert, J.P.: A new CRT-RSA algorithm secure against Bellcore attack. In: 10th ACM Conference on Computer and Communications Security (CCS 2003). pp. 311–320. ACM Press (2003). doi:10.1145/948109.948151
13. Blömer, J., Otto, M., Seifert, J.P.: Sign change fault attacks on elliptic curve cryptosystems. In: Breveglieri, L., et al. (eds.) Fault Diagnosis and Tolerance in Cryptography − FDTC 2006. Lecture Notes in Computer Science, vol. 4236, pp. 36–52. Springer (2006). doi:10.1007/11889700_4
14. Boneh, D., DeMillo, R.A., Lipton, R.J.: On the importance of eliminating errors in cryptographic computations. J. Cryptology **14**(2), 101–119 (2001). doi:10.1007/s001450010016, extended abstract in Proc. of EUROCRYPT '97
15. Bosma, W., Lenstra Jr, H.W.: Complete systems of two addition laws for elliptic curves. J. Number Theor. **53**(2), 229–240 (1995). doi:10.1006/jnth.1995.1088
16. Chudnovsky, D.V., Chudnovsky, G.V.: Sequences of numbers generated by addition in formal groups and new primality and factorization tests. Adv. Appl. Math. **7**, 385–434 (1986). doi:10.1016/0196-8858(86)90023-0
17. Ciet, M., Joye, M.: Elliptic curve cryptosystems in the presence of permanent and transient faults. Designs, Codes and Cryptography **36**(1), 33–43 (2005). doi:10.1007/s10623-003-1160-8
18. Ciet, M., Joye, M.: Practical fault countermeasures for Chinese remaindering based RSA. In: 2nd Workshop on Fault Diagnosis and Tolerance in Cryptography (FDTC 2005). pp. 124–132. Edinburgh, UK (2005), http://conferenze.dei.polimi.it/FDTC05/Joye%20-%20publisheable.pdf
19. Ciss, A.A., Sow, D.: On a new generalization of Huff curves. Cryptology ePrint Archive, Report 2011/580 (2011), http://eprint.iacr.org/2011/580
20. Cohen, H., Frey, G. (eds.): Handbook of Elliptic and Hyperelliptic Curve Cryptography, Discrete Mathematics and Its Applications, vol. 34. Chapman & Hall/CRC (2005)

21. Coron, J.S., Giraud, C., Morin, N., Piret, G., Vigilant, D.: Fault attacks and countermeasures on Vigilant's RSA-CRT algorithm. In: Breveglieri, L., et al. (eds.) 7th Workshop on Fault Diagnosis and Tolerance in Cryptography (FDTC 2010). pp. 89–96. IEEE Computer Society (2010). doi:10.1109/FDTC.2010.9
22. Duquesne, S.: Improving the arithmetic of elliptic curves in the Jacobi model. Information Processing Letters **104**(3), 101–105 (2007). doi:10.1016/j.ipl.2007.05.012
23. Edwards, H.M.: A normal form for elliptic curves. Bull. Am. Math. Soc. **44**(3), 393–422 (2007). doi:10.1090/S0273-0979-07-01153-6
24. Fan, J., Verbauwhede, I.: An updated survey on secure ECC implementations: Attacks, countermeasures and cost. In: Naccache, D. (ed.) Cryptography and Security: From Theory to Applications (Quisquater Festschrift). Lecture Notes in Computer Science, vol. 6805, pp. 265–282. Springer (2012). doi:10.1007/978-3-642-28368-0_18
25. Farashahi, R.R., Joye, M.: Efficient arithmetic on Hessian curves. In: Nguyen, P.Q., Pointcheval, D. (eds.) Public Key Cryptography – PKC 2010. Lecture Notes in Computer Science, vol. 6056, pp. 243–260. Springer (2010). doi:10.1007/978-3-642-13013-7_15
26. Galbraith, S.D.: Elliptic curve Paillier schemes. J. Cryptology **15**(2), 129–138 (2002). doi:10.1007/s00145-001-0015-6
27. Giraud, C., Thiebeauld, H.: A survey on fault attacks. In: Quisquater, J.J., et al. (eds.) Smart Card Research and Advanced Applications VI (CARDIS 2004). pp. 159–176. Kluwer (2004). doi:10.1007/1-4020-8147-2_11
28. Hesse, O.: Über die Elimination der Variabeln aus drei algebraischen Gleichungen vom zweiten Grade mit zwei Variabeln. J. Reine Angew. Math. **10**, 68–96 (1844). doi:10.1515/crll.1844.28.68
29. Hışıl, H., Carter, G., Dawson, E.: New formulae for efficient elliptic curve arithmetic. In: Srinathan, K., Rangan, C.P., Yung, M. (eds.) Progress in Cryptology – INDOCRYPT 2007. Lecture Notes in Computer Science, vol. 4859, pp. 138–151. Springer (2007). doi:10.1007/978-3-540-77026-8_11
30. Hışıl, H., Wong, K.K.H., Carter, G., Dawson, E.: Twisted Edwards curves revisited. In: Pieprzyk, J. (ed.) Advances in Cryptology – ASIACRYPT 2008. Lecture Notes in Computer Science, vol. 5350, pp. 326–343. Springer (2008). doi:10.1007/978-3-540-89255-7_20
31. Hışıl, H., Wong, K.K.H., Carter, G., Dawson, E.: Jacobi quartic curves revisited. In: Boyd, C., Nieto, J.M.G. (eds.) Information Security and Privacy (ACISP 2009). Lecture Notes in Computer Science, vol. 5594, pp. 452–468. Springer (2009). doi:10.1007/978-3-642-02620-1_31
32. Hışıl, H., Wong, K.K.H., Carter, G., Dawson, E.: An exploration of affine group laws for elliptic curves. J. Math. Cryptol. **5**(1), 1–50 (2011). doi:10.1515/jmc.2011.005
33. Huff, G.B.: Diophantine problems in geometry and elliptic ternary forms. Duke Math. J. **15**, 443–453 (1948). doi:10.1215/S0012-7094-48-01543-9
34. Husemöller, D.: Elliptic Curves, Graduate Texts in Mathematics, vol. 111. Springer (1987). doi:10.1007/978-1-4757-5119-2
35. Joye, M.: Complete addition formulæ for elliptic curves. Tech. rep., Technicolor, Rennes (Oct 2008), http://joye.site88.net/techreps/complete.pdf
36. Joye, M.: On the security of a unified countermeasure. In: Breveglieri, L., et al. (eds.) 5th Workshop on Fault Diagnosis and Tolerance in Cryptography (FDTC 2008). pp. 87–91. IEEE Computer Society (2008). doi:10.1109/FDTC.2008.8
37. Joye, M.: Edwards curves and fault attacks. Presented at the rump session of CRYPTO 2012, Santa Barbara, USA (Aug 21, 2012), http://crypto.2012.rump.cr.yp.to/
38. Joye, M., Paillier, P., Yen, S.M.: Secure evaluation of modular functions. In: Hwang, R., Wu, C. (eds.) 2001 International Workshop on Cryptology and Network Security. pp. 227–229. Taipei, Taiwan (Sep 2001), http://joye.site88.net/papers/JPY01dfa.pdf
39. Joye, M., Quisquater, J.J.: Hessian elliptic curves and side-channel attacks. In: Koç, Ç.K., Naccache, D., Paar, C. (eds.) Cryptographic Hardware and Embedded Systems – CHES 2001. Lecture Notes in Computer Science, vol. 2162, pp. 402–410. Springer (2001). doi:10.1007/3-540-44709-1_33
40. Joye, M., Tibouchi, M., Vergnaud, D.: Huff's model for elliptic curves. In: Hanrot, G., Morain, F., Thomé, E. (eds.) Algorithmic Number Theory (ANTS-IX). Lecture Notes in Computer Science, vol. 6197, pp. 234–250. Springer-Velag (Jul 2010). doi:10.1007/978-3-642-14518-6_20
41. Joye, M., Tunstall, M. (eds.): Fault Analysis in Cryptography. Information Security and Cryptography, Springer (2012). doi:10.1007/978-3-642-29656-7
42. Karabina, K., Ustaoglu, B.: Invalid-curve attacks on (hyper)elliptic curve cryptosystems. Adv. Math. Commun. **4**(3), 307–321 (2010). doi:10.3934/amc.2010.4.307
43. Kim, C.H., Quisquater, J.J.: How can we overcome both side channel analysis and fault attacks on RSA-CRT? In: Breveglieri, L., et al. (eds.) 4th Workshop on Fault Diagnosis and Tolerance in Cryptography (FDTC 2007). pp. 21–29. IEEE Computer Society (2007). doi:10.1109/FDTC.2007.11

44. Kim, T., Tibouchi, M.: Bit-flip faults on elliptic curve base fields, revisited. In: Boureanu, I., Owesarski, P., Vaudenay, S. (eds.) Applied Cryptography and Network Security (ACNS 2014). Lecture Notes in Computer Science, vol. 8479, pp. 163–180. Springer (2014). doi:10.1007/978-3-319-07536-5_11
45. Koblitz, N.: Elliptic curve cryptosystems. Math. Comp. **48**(177), 203–209 (1987). doi:10.2307/2007884
46. Lange, H., Ruppert, W.: Complete systems of addition laws on abelian varieties. Invent. Math. **79**(3), 603–610 (1985). doi:10.1007/BF01388526
47. Lange, H., Ruppert, W.: Addition laws on elliptic curves in arbitrary characteristics. Journal of Algebra **107**(1), 106–116 (1987). doi:10.1016/0021-8693(87)90077-9
48. Liardet, P.Y., Smart, N.P.: Preventing SPA/DPA in ECC systems using the Jacobi form. In: Koç, Ç.K., Naccache, D., Paar, C. (eds.) Cryptographic Hardware and Embedded Systems — CHES 2001. Lecture Notes in Computer Science, vol. 2162, pp. 391–401. Springer (2001). doi:10.1007/3-540-44709-1_32
49. Menezes, A.: Elliptic Curve Public Key Cryptosystems. Kluwer Academic Publishers (1993). doi:10.1007/978-1-4615-3198-2
50. Miller, V.: Use of elliptic curves in cryptography. In: Williams, H.C. (ed.) Advances in Cryptology — CRYPTO '85. Lecture Notes in Computer Science, vol. 218, pp. 417–426. Springer (1986). doi:10.1007/3-540-39799-X_31
51. Neves, S., Tibouchi, M.: Degenerate curve attacks: Extending invalid curve attacks to Edwards curves and other models. IET Information Security **12**(3), 217–225 (2018). doi:10.1049/iet-ifs.2017.0075
52. Orhon, N.G., Hışıl, H.: Speeding up Huff form of elliptic curves. Designs, Codes and Cryptography **86**(12), 2807–2803 (2018). doi:10.1007/s10623-018-0475-4
53. Quisquater, J.J., Couvreur, C.: Fast decipherment algorithm for RSA public-key cryptosystem. Electronics Letters **18**(21), 905–907 (1982). doi:10.1049/el:19820617
54. Renes, J., Costello, C., Batina, L.: Complete addition formulas for prime order elliptic curves. In: Fischlin, M., Coron, J.S. (eds.) Adv ances in Cryptology — EUROCRYPT 2016, Part I. Lecture Notes in Computer Science, vol. 9665, pp. 403–428. Springer (2016). doi:10.1007/978-3-662-49890-3_16
55. Rivest, R.L., Shamir, A., Adleman, L.M.: A method for obtaining digital signatures and public-key cryptosystems. Commun. ACM **21**(2), 120–126 (1978). doi:10.1145/359340.359342
56. Rubin, K., Silverberg, A.: Torus-based cryptography. In: Boneh, D. (ed.) Advances in Cryptology — CRYPTO 2003. Lecture Notes in Computer Science, vol. 2729, pp. 349–365. Springer (2000). doi:10.1007/978-3-540-45146-4_21
57. Schmidt, J.M., Medwed, M.: A fault attack on ECDSA. In: Breveglieri, L., et al. (eds.) 6th Workshop on Fault Diagnosis and Tolerance in Cryptography (FDTC 2009). pp. 93–99. IEEE Computer Society (2009). doi:10.1109/FDTC.2009.38
58. Shamir, A.: How to check modular exponentiation. Presented at the rump session of EUROCRYPT '97, Konstanz, Germany (May 13, 1997), https://www.iacr.org/conferences/ec97/rump.html
59. Silverman, J.H.: The Arithmetic of Elliptic Curves, Graduate Texts in Mathematics, vol. 106. Springer (1986). doi:10.1007/978-0-387-09494-6
60. Smart, N.P.: The Hessian form of an elliptic curve. In: Koç, Ç.K., Naccache, D., Paar, C. (eds.) Cryptographic Hardware and Embedded Systems — CHES 2001. Lecture Notes in Computer Science, vol. 2162, pp. 118–125. Springer (2001). doi:10.1007/3-540-44709-1_11
61. Vigilant, D.: RSA with CRT: A new cost-effective solution to thwart fault attacks. In: Oswald, E., Rohatgi, P. (eds.) Cryptographic Hardware and Embedded Systems — CHES 2008. Lecture Notes in Computer Science, vol. 5154, pp. 130–145. Springer (2008). doi:10.1007/978-3-540-85053-3_9
62. Wu, H., Feng, R.: Elliptic curves in Huff's model. Wuhan University Journal of Natural Sciences **17**(6), 473–480 (2012). doi:10.1007/s11859-012-0873-9
63. Yen, S.M., Kim, S., Lim, S., Moon, S.: RSA speedup with Chinese remainder theorem immune against hardware fault cryptanalysis. IEEE Trans. Computers **52**(4), 461–472 (2003). doi:10.1109/TC.2003.1190587

A Further Models

A.1 Jacobi Quartic Model

The (extended) Jacobi quartic model is presented in [9] (see also [16,22,29,31]). Its curve equation is given by

$$E_{\mathcal{J}_{a,d}} : y^2 = dx^4 + 2ax^2 + 1$$

with $\boldsymbol{O} = (0,1)$ as the neutral element. The unified addition of two points (x_1, y_1) and (x_2, y_2), $(x_3, y_3) = (x_1, y_1) \boxplus (x_2, y_2)$, is given by

$$(x_3, y_3) = \left(\frac{x_1 y_2 + x_2 y_1}{1 - d {x_1}^2 {x_2}^2}, \frac{(1 + d {x_1}^2 {x_2}^2)(y_1 y_2 + 2a x_1 x_2) + 2d x_1 x_2 ({x_1}^2 + {x_2}^2)}{(1 - d {x_1}^2 {x_2}^2)^2} \right) .$$

[The original Jacobi quartics correspond to the case $d = k^2$ and $-2a = 1 + k^2$ for some parameter k.]

First Realization. We define:

$$E_{\mathcal{J}_{a,d},1}(\mathbb{Z}/r^2\mathbb{Z}) = \left\{ \Upsilon_1(\vartheta) = (\vartheta \cdot r, 1) \mid \vartheta \in \mathbb{Z}/r\mathbb{Z} \right\} \cong (\mathbb{Z}/r\mathbb{Z})^+ .$$

Analogously to the Edwards model (cf. § 4.1), it is easily verified that $\Upsilon_1(\vartheta) \equiv (\vartheta \cdot r, 1) \equiv (0,1) \equiv \Upsilon_1(0) \equiv \boldsymbol{O} \pmod{r}$ and that $\Upsilon_1(\vartheta_1) \boxplus \Upsilon_1(\vartheta_2) = (\vartheta_1, 1) \boxplus (\vartheta_2, 1) = \left(\frac{\vartheta_1 \cdot r \cdot 1 + \vartheta_2 \cdot r \cdot 1}{1}, \frac{1 \cdot 1}{1^2} \right) = \left((\vartheta_1 + \vartheta_2) \cdot r, 1 \right) = \Upsilon_1(\vartheta_1 + \vartheta_2)$.

Second Realization. We have:

$$(\mathbb{Z}/r\mathbb{Z})^+ \cong E'_{\mathcal{J}_{0,0}}(\mathbb{Z}/r\mathbb{Z})[r] = \left\{ \Upsilon_2(\vartheta) = (\vartheta, 1) \mid \vartheta \in \mathbb{Z}/r\mathbb{Z} \right\} \subseteq \left\{ (x, y) \in E'_{\mathcal{J}_{0,0}}(\mathbb{Z}/r\mathbb{Z}) \right\} .$$

As for the Edwards model, we have $\Upsilon_2(0) = (0,1) = \boldsymbol{O}$ and, when $a = d = 0$, $\Upsilon_2(\vartheta_1) \boxplus \Upsilon_2(\vartheta_2) = (\vartheta_1, 1) \boxplus (\vartheta_2, 1) = \left(\frac{\vartheta_1 \cdot 1 + \vartheta_2 \cdot 1}{1}, \frac{1 \cdot 1}{1^2} \right) = (\vartheta_1 + \vartheta_2, 1) = \Upsilon_2(\vartheta_1 + \vartheta_2)$.

A.2 Jacobi Quadrics Intersection Model

Another way to represent an elliptic curve is as the intersection of two quadrics in $\mathbb{P}^3$ (see, e.g., [16]). Applications to cryptography are discussed in [16,29,48]. The most general form [32] reads as

$$E_{\mathcal{Q}_{a,b}} : \begin{cases} a x^2 + y^2 = 1 \\ b x^2 + z^2 = 1 \end{cases} .$$

The neutral element is $\boldsymbol{O} = (0,1,1)$. The unified sum of two points (x_1, y_1, z_1) and (x_2, y_2, z_2) is given by $(x_3, y_2, z_3) = (x_1, y_1, z_1) \boxplus (x_2, y_2, z_2)$ where

$$(x_3, y_3, z_3) = \left(\frac{x_1 y_2 z_2 + x_2 y_1 z_1}{1 - ab {x_1}^2 {x_2}^2}, \frac{y_1 y_2 - a x_1 z_1 x_2 z_2}{1 - ab {x_1}^2 {x_2}^2}, \frac{z_1 z_2 - b x_1 y_1 x_2 y_2}{1 - ab {x_1}^2 {x_2}^2} \right) .$$

First Realization. We define:

$$E_{\mathcal{Q}_{a,b},1}(\mathbb{Z}/r^2\mathbb{Z}) = \left\{ \Upsilon_1(\vartheta) = (\vartheta \cdot r, 1, 1) \mid \vartheta \in \mathbb{Z}/r\mathbb{Z} \right\} \cong (\mathbb{Z}/r\mathbb{Z})^+ .$$

A straightforward calculation shows that $\Upsilon_1(\vartheta) \equiv (\vartheta \cdot r, 1, 1) \equiv (0,1,1) \equiv \Upsilon_1(0) \equiv \boldsymbol{O} \pmod{r}$ and that $\Upsilon_1(\vartheta_1) \boxplus \Upsilon_1(\vartheta_2) = (\vartheta_1 \cdot r, 1, 1) \boxplus (\vartheta_2 \cdot r, 1, 1) = \left(\frac{\vartheta_1 \cdot r \cdot 1 \cdot 1 + \vartheta_2 \cdot r \cdot 1 \cdot 1}{1}, \frac{1 \cdot 1}{1}, \frac{1 \cdot 1}{1} \right) = \left((\vartheta_1 + \vartheta_2) \cdot r, 1, 1 \right) = \Upsilon_1(\vartheta_1 + \vartheta_2)$.

Second Realization. We have:

$$(\mathbb{Z}/r\mathbb{Z})^+ \cong E'_{\mathcal{Q}_{0,0}}(\mathbb{Z}/r\mathbb{Z})[r] = \left\{ \Upsilon(\vartheta) = (\vartheta, 1, 1) \mid \vartheta \in \mathbb{Z}/r\mathbb{Z} \right\} \subseteq \left\{ (x, y, z) \in E_{\mathcal{Q}_{0,0}}(\mathbb{Z}/r\mathbb{Z}) \right\} .$$

It can be checked that $\Upsilon_2(0) = (0,1,1) = \boldsymbol{O}$ and, when $a = b = 0$, that $\Upsilon_2(\vartheta_1) \boxplus \Upsilon_2(\vartheta_2) = (\vartheta_1, 1, 1) \boxplus (\vartheta_2, 1, 1) = \left(\frac{\vartheta_1 \cdot 1 \cdot 1 + \vartheta_2 \cdot 1 \cdot 1}{1}, \frac{1 \cdot 1}{1}, \frac{1 \cdot 1}{1} \right) = (\vartheta_1 + \vartheta_2, 1, 1) = \Upsilon_2(\vartheta_1 + \vartheta_2)$.

A.3 Hessian Model

Hessian curves [28] were generalized, modified, and extended for cryptographic applications in several works, including [39,60,6,25]. We follow the presentation of [6] where the neutral element is $\boldsymbol{O} = (0, -1)$. The curve equation is

$$E_{\mathcal{H}_{a,d}} \colon ax^3 + y^3 + 1 = dxy \ .$$

The unified sum $(x_3, y_3) = (x_1, y_1) \boxplus (x_2, y_2)$ of two points (x_1, y_1) and (x_2, y_2) is given by

$$(x_3, y_3) = \left(\frac{x_1 - {y_1}^2 x_2 y_2}{a {x_1} y_1 {x_2}^2 - y_2}, \frac{y_1 {y_2}^2 - a {x_1}^2 x_2}{a {x_1} y_1 {x_2}^2 - y_2} \right) \ .$$

First Realization. We define:

$$E_{\mathcal{H}_{a,d},1}(\mathbb{Z}/r^2\mathbb{Z}) = \left\{ \Upsilon_1(\vartheta) = (3\vartheta \cdot r, -1 - d\vartheta \cdot r) \mid \vartheta \in \mathbb{Z}/r\mathbb{Z} \right\} \cong (\mathbb{Z}/r\mathbb{Z})^+ \ .$$

Again, it can be verified that $\Upsilon_1(\vartheta) \equiv (3\vartheta \cdot r, -1 - d\vartheta \cdot r) \equiv (0, -1) \equiv \Upsilon_1(0) \equiv \boldsymbol{O} \pmod{r}$. A quick inspection shows that the above addition law for computing $\Upsilon_1(\vartheta_1) \boxplus \Upsilon_1(\vartheta_2) = (3\vartheta_1 \cdot r, -1 - d\vartheta_1 \cdot r) \boxplus (3\vartheta_2 \cdot r, -1 - d\vartheta_2 \cdot r)$ incurs the value of $-(-1 - d\vartheta_2 \cdot r) = 1 + d\vartheta_2 \cdot r$ in the denominator. We therefore must have $d\vartheta_2 \cdot r \neq -1 \pmod{r^2}$. This is always satisfied since $d\vartheta_2 \cdot r \equiv -1 \pmod{r^2}$ would imply $0 \equiv -1 \pmod{r}$. Hence, the sum is always defined and is given by as $\Upsilon_1(\vartheta_1) \boxplus \Upsilon_1(\vartheta_2) = \left(\frac{3\vartheta_1 \cdot r - (-1 - d\vartheta_1 \cdot r)^2 \cdot (3\vartheta_2 \cdot r) \cdot (-1 - d\vartheta_2 \cdot r)}{-(-1 - d\vartheta_2 \cdot r)}, \right.$ $\left. \frac{(-1 - d\vartheta_1 \cdot r) \cdot (-1 - d\vartheta_2 \cdot r)^2}{-(-1 - d\vartheta_2 \cdot r)} \right) = \left(\frac{3\vartheta_1 \cdot r + 3\vartheta_2 \cdot r}{1 + d\vartheta_2 \cdot r}, \frac{-1 - d\vartheta_1 \cdot r - d\vartheta_2 \cdot r}{1} \right) = \left(3(\vartheta_1 + \vartheta_2) \cdot r, -1 - d(\vartheta_1 + \vartheta_2) \cdot r \right) = \Upsilon_1(\vartheta_1 + \vartheta_2)$, noting that $(1 + d\vartheta_2 \cdot r)^{-1} = 1 - d\vartheta_2 \cdot r$ and that $(3\vartheta_1 \cdot r + 3\vartheta_2 \cdot r)(1 - d\vartheta_2 \cdot r) = 3(\vartheta_1 + \vartheta_2) \cdot r$.

Second Realization. We have:

$$(\mathbb{Z}/r\mathbb{Z})^+ \cong E'_{\mathcal{H}_{0,0}}(\mathbb{Z}/r\mathbb{Z})[r] = \left\{ \Upsilon_2(\vartheta) = (\vartheta, -1) \mid \vartheta \in \mathbb{Z}/r\mathbb{Z} \right\} \subseteq \left\{ (x, y) \in E'_{\mathcal{H}_{0,0}}(\mathbb{Z}/r\mathbb{Z}) \right\} \ .$$

Likewise, we have $\Upsilon_2(0) = (0, -1) = \boldsymbol{O}$ and the addition law when $a = d = 0$ yields $\Upsilon_2(\vartheta_1) \boxplus \Upsilon_2(\vartheta_2) = (\vartheta_1, -1) \boxplus (\vartheta_2, -1) = \left(\frac{\vartheta_1 - (-1)^2 \vartheta_2 (-1)}{-(-1)}, \frac{(-1)(-1)^2}{-(-1)} \right) = (\vartheta_1 + \vartheta_2, -1) = \Upsilon_2(\vartheta_1 + \vartheta_2)$.

A.4 Huff's Model

Huff curves, after [33], were introduced for cryptographic applications in [40]. The most general form as presented in [52] (see also [19,62]) is given by the equation

$$E_{\mathcal{H}_{a,c,d}} \colon y(ax^2 + 1) = cx(dy^2 + 1)$$

with neutral element $\boldsymbol{O} = (0, 0)$. The unified addition formula of points (x_1, y_1) and (x_2, y_2) is given by $(x_3, y_3) = (x_1, y_1) \boxplus (x_2, y_2)$ where

$$(x_3, y_3) = \left(\frac{(x_1 + x_2)(1 - d y_1 y_2)}{(1 - a x_1 x_2)(1 + d y_1 y_2)}, \frac{(y_1 + y_2)(1 - a x_1 x_2)}{(1 + a y_1 y_2)(1 - d x_1 x_2)} \right) \ .$$

First Realization. We define:

$$E_{\mathcal{H}_{a,c,d},1}(\mathbb{Z}/r^2\mathbb{Z}) = \left\{ \Upsilon_1(\vartheta) = (\vartheta \cdot r, c\,\vartheta \cdot r) \mid \vartheta \in \mathbb{Z}/r\mathbb{Z} \right\} \cong (\mathbb{Z}/r\mathbb{Z})^+ \ .$$

The correctness follows by observing that $\Upsilon_1(\vartheta) \equiv (\vartheta \cdot r, c\,\vartheta \cdot r) \equiv (0, 0) \equiv \Upsilon_1(0) \equiv \boldsymbol{O} \pmod{r}$ and that the addition law leads to $\Upsilon_1(\vartheta_1) \boxplus \Upsilon_1(\vartheta_2) = (\vartheta_1 \cdot r, c\,\vartheta_1 \cdot r) \boxplus (\vartheta_2 \cdot r, c\,\vartheta_2 \cdot r) = \left(\frac{(\vartheta_1 \cdot r + \vartheta_2 \cdot r) \cdot 1}{1 \cdot 1}, \frac{(c\,\vartheta_1 \cdot r + c\,\vartheta_2 \cdot r) \cdot 1}{1 \cdot 1} \right) = \left((\vartheta_1 + \vartheta_2) \cdot r, c(\vartheta_1 + \vartheta_2) \cdot r \right) = \Upsilon_1(\vartheta_1 + \vartheta_2)$.

Second Realization. Fix $\bar{c} \in \mathbb{Z}/r\mathbb{Z}$. We have:

$$(\mathbb{Z}/r\mathbb{Z})^+ \cong E'_{\mathcal{H}_{0,\bar{c},0}}(\mathbb{Z}/r\mathbb{Z})[r] = \left\{ \Upsilon_2(\vartheta) = (\vartheta, \bar{c} \cdot \vartheta) \mid \vartheta \in \mathbb{Z}/r\mathbb{Z} \right\} \subseteq \left\{ (x, y) \in E'_{\mathcal{H}_{0,\bar{c},0}}(\mathbb{Z}/r\mathbb{Z}) \right\} \ .$$

We observe that $\Upsilon_2(0) = (0, 0) = \boldsymbol{O}$ and, when $(a, c, d) = (0, \bar{c}, 0)$, the addition law gives $\Upsilon_2(\vartheta_1) \boxplus \Upsilon_2(\vartheta_2) = (\vartheta_1, \bar{c} \cdot \vartheta_1) \boxplus (\vartheta_2, \bar{c} \cdot \vartheta_2) = (\vartheta_1 + \vartheta_2, \bar{c} \cdot \vartheta_1 + \bar{c} \cdot \vartheta_2) = \Upsilon_2(\vartheta_1 + \vartheta_2)$.