





Embedded Curves and Embedded Families for SNARK-Friendly Curves

Aurore Guillevic   and Simon Masson  

Univ Rennes, Inria, CNRS, IRISA, Rennes, France

Abstract. Based on the CM method for primality testing (ECP) by Atkin and Morain published in 1993, we present two algorithms: one to generate embedded elliptic curves of SNARK-friendly curves, with a variable discriminant D ; and another to generate families (parameterized by polynomials) with a fixed discriminant D . When $D = 3 \pmod{4}$, it is possible to obtain a prime-order curve, and form a cycle. We apply our technique first to generate more embedded curves like Bandersnatch with BLS12-381 and we propose a plain twist-secure cycle above BLS12-381 with $D = 6673027$. We also devise about the scarcity of Bandersnatch-like CM curves, and show that with our algorithm, it is only a question of core-hours to find them. Second, we obtain families of prime-order embedded curves of discriminant $D = 3$ for BLS and KSS18 curves. Our method obtains families of embedded curves above KSS16 and can work for any KSS family. Our work generalizes the work on Bandersnatch (Masson, Sanso, and Zhang, and Sanso and El Housni).

Keywords: elliptic curves · SNARK · embedded curves · cycles of curves

1 Introduction

With the development of proof-of-knowledge systems, in particular SNARK (Succinct Non-interactive ARgument of Knowledge), new elliptic curves know a recent regain of interest. These curves are defined over a prime field \mathbb{F}_p and equipped with a polynomial commitment. We distinguish two types of constructions:

- Elliptic curves with a polynomial commitment based on the discrete logarithm problem [Hop20b, BGH19],
- Pairing-friendly curves equipped with an efficient bilinear map that pairs points on the curve and outputs a value in a finite field. A polynomial commitment [KZG10] can be obtained from the bilinear map, also called a pairing.

Zero-knowledge SNARKs are built using arithmetic circuits corresponding to operations involving a secret value. In other words, ZK proofs demonstrate the knowledge of a secret information that satisfies a specified list of arithmetic operations. While the arithmetic circuit can be very large in practice, SNARK proofs are small (succinct) so that it is fast enough to verify the proof. Many zero-knowledge constructions are instantiated using a pairing-friendly elliptic curve. In this context, the output proof includes a point of the pairing-friendly elliptic curve, and the verification requires the computation of a cryptographic pairing. In [Gro16], Groth was the first to achieve a cost as small as three pairings and additional multiplications/exponentiations. The construction was later improved and optimized [GWC19, GW20, CBBZ22]. The design was then adapted for

E-mail: aurore.guillevic@inria.fr (Aurore Guillevic), simon.masson@protonmail.com (Simon Masson)

This work is licensed under a “CC BY 4.0” license.

Date of this document: 2024-10-24.



the DL polynomial commitment [BGH19]. In most of the constructions, the elliptic curve equipped with a polynomial commitment is defined over a prime field \mathbb{F}_p , and has a cryptographically secure prime subgroup of size q , also called scalar field \mathbb{F}_q .

ZK proofs built using circuits compute proofs of arithmetic in a finite field. In the context of the designs mentioned above (in particular [GWC19, BGH19]), this field is the scalar field \mathbb{F}_q of the elliptic curve. In a cryptographic setting, it is interesting to build proofs related to cryptographic protocol objects (such as digital signatures, or even other ZK proofs). In practice, elliptic curve point arithmetic is usually computed, involving arithmetic on the base field of the curve. Thus, the base field of this second elliptic curve needs to be \mathbb{F}_q , the scalar field of the first curve. The naive idea of finding an elliptic curve with identical base and scalar fields is not cryptographically secure [Sma99]. These curves are called anomalous and must be avoided. Hence, a second curve, *embedded* on the proof curve is required, and its base field needs to match the scalar field of the proof curve. The relation between base field of the embedded curve and scalar field of the proof curve is summarized in Figure 1. Many ZK projects are based on the pairing-friendly curve BLS12-381. It is possible to obtain an embedded curve for BLS12-381. Jubjub was first proposed [Hop17], and [MSZ24] considers a second curve called Bandersnatch, allowing faster scalar multiplications.

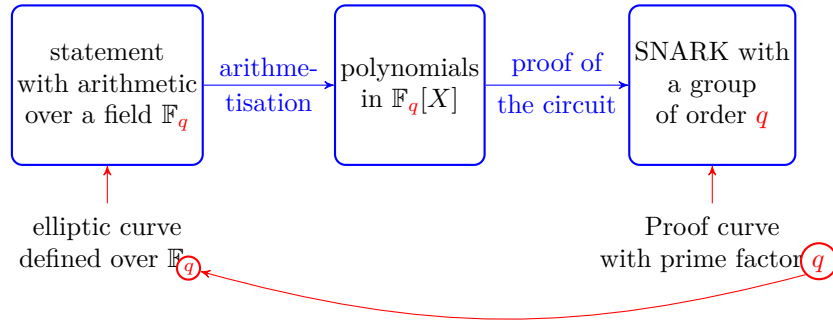


Figure 1: Relation between base field of the embedded curve and scalar field of the proof curve (from [AHG23, Fig. 3]).

Zero-knowledge proofs can be recursively computed in order to keep the proof size small enough and the verification time efficient at a large scale. In this context, the arithmetic circuit includes information on the verification of the previous proof. In other words, if a circuit is defined over \mathbb{F}_q as above, the next proof will be defined using a circuit over \mathbb{F}_p . Using a chain of curves (where the scalar field of a curve is the base field of the previous curve), it is possible to compose proofs in order to achieve a better efficiency. This is also called a one-layer proof composition setting. In order to achieve recursive proofs, the curves involved in the proofs need to be chained so that every proof on a curve can be composed with a proof on the next curve. Cycles of curves make this construction possible, where the scalar field of a curve is the base field of another curve and vice-versa. In particular, a 2-cycle refers to two curves E_1 and E_2 defined respectively over \mathbb{F}_p and \mathbb{F}_q and of respective scalar field \mathbb{F}_q and \mathbb{F}_p . This construction together with the pairing property is possible using MNT curves, but due to NFS variants on the discrete logarithm problem over the pairing output fields, the size of the primes p and q are very large, making the computations very slow. Recent works [CK24] investigate supersingular cycles for a higher security, leading to a better efficiency than MNT cycles. Another approach avoids pairings and considers the DL polynomial commitment scheme. Using this approach, pairing-friendly curves are not needed and one can consider 256-bit primes p and q . We refer here to *plain 2-cycle* as a cycle of two curves cryptographically secure, without the pairing property. A plain 2-cycle was originally designed for HALO2 [Hop20b, BGH19].

The two curves are called Pallas and Vesta, forming the Pasta cycle, used for instance within the Rust implementation of NOVA [KST22]. Although this construction makes recursive proofs possible, the proof size is not constant anymore (but still logarithmic in the size of the inputs). Finally, it is possible to alternate proofs on a pairing-friendly curve and on a plain elliptic curve ([Azt] for the BN-254–Grumpkin half-cycle, [Hop21] for Pluto-Eris). This construction has been considered only theoretically and its efficiency is not well understood.

Our contributions. In this work, we look for embedded curves with different goals. First, we obtain an efficient algorithm for finding embedded curves for a fixed elliptic curve. Then, we apply this algorithm in order to find a plain 2-cycle with the standardized ed255-19 elliptic curve, and for obtaining a plain 2-cycle embedded on the pairing-friendly curve BLS12-381. We investigate the scarcity of such curves and show that despite Bandersnatch is indeed a lucky curve (secure, of very small discriminant *and* twist-secure), it is possible to generate a curve with similar properties (considering a larger discriminant). We also consider families of embedded curves for a fixed discriminant, generalizing the idea of [SEH24].

Organization of the paper. Preliminaries on the CM method, the ECPP algorithm, and the previous works on finding embedded curves are in Section 2. In Section 3, we propose our new efficient algorithm for finding embedded curves, based on algorithmic number theory of imaginary quadratic fields. We apply this method for different zero-knowledge proof use-cases in Section 3.3. Finally, we consider families of embedded curves in Section 4, for fixed discriminants such as $D = 1, 3$.

2 Preliminaries

2.1 Notations

2.1.1 Elliptic curves.

In this paper, $E: y^2 = x^3 + ax + b$ is an elliptic curve ($4a^3 + 27b^2 \neq 0$) defined over a prime finite field \mathbb{F} of large characteristic ≥ 5 and in practice, of cryptographic size of about 256 bits. The above equation is a Weierstrass representation of the curve, and in practice, it is common to use a small a -coefficient. In particular, $a = -3$ leads to an efficient arithmetic on the curve [CS18].

We denote by t the trace of the Frobenius map on E , so that the curve order is $\#E(\mathbb{F}) = \#\mathbb{F} + 1 - t$. From now, we denote q for the characteristic of the base field of the elliptic curve. Here, we consider ordinary elliptic curves, meaning that the endomorphism ring of the curve E is always an order of the imaginary quadratic field $K = \mathbb{Q}[\sqrt{t^2 - 4q}]$. Ordinary curves have a trace $t \not\equiv 0 \pmod{q}$. Thanks to the Hasse–Weil bound, the quantity $t^2 - 4q$ is always negative. Let us denote its square-free factorization as $t^2 - 4q = -dy^2$, where $d > 0$ is square-free, so that

$$4q = t^2 + dy^2. \tag{1}$$

In Section 3.2, we will be interested in solving Equation (1) for integers t, y , a prime q and $d > 0$. This is closely related to the theory of imaginary quadratic fields.

2.1.2 Imaginary quadratic fields.

To keep consistent notations, we define the imaginary quadratic number field K for a square-free positive d :

$$K = \mathbb{Q}[\sqrt{-d}] = \mathbb{Q}[X]/(X^2 + d).$$

The negative fundamental discriminant of K is $-D$, where $D = 4d$ if $d = 1, 2 \pmod{4}$; $D = d$ if $d = 3 \pmod{4}$. The structure of the maximal order of K (its ring of integers \mathcal{O}_K) is $\mathbb{Z}[\omega]$ where ω is defined as follows:

$$\omega = \begin{cases} \sqrt{-D/4} & \text{if } D \not\equiv 3 \pmod{4}, \\ \frac{1+\sqrt{-D}}{2} & \text{if } D \equiv 3 \pmod{4}. \end{cases}$$

The element ω characterizes \mathcal{O}_K and it is a root of the polynomial $P(X)$ where

$$P(X) = \begin{cases} X^2 + d & \text{if } d \equiv 1, 2 \pmod{4}, \\ X^2 - X + (d+1)/4 & \text{if } d \equiv 3 \pmod{4}. \end{cases}$$

Observe that $-D$ is the discriminant of the polynomial $P(X)$.

Every order is a suborder of the maximal order. In this paper, we identify elliptic curves to their fundamental discriminant: the endomorphism ring is a suborder of a maximal order of $\mathbb{Q}[\sqrt{t^2 - 4q}]$ of a given fundamental discriminant. In other words, we consider elliptic curve isogeny classes. We refer to [Feo17] for the theory of isogenies (not needed here). Let τ denote complex conjugation. The conjugate of an element $\alpha = a + b\omega$ is $\alpha' = \tau(\alpha) = a + b\tau(\omega)$. The norm of an element $\alpha = a + b\omega$ is $N_{K/\mathbb{Q}}(\alpha) = \alpha\tau(\alpha)$. More precisely,

$$N_{K/\mathbb{Q}}(a + b\omega) = \begin{cases} (a + b\sqrt{-d})(a - b\sqrt{-d}) = a^2 + db^2 & D \not\equiv 3 \pmod{4}, \\ (a + b\frac{1+\sqrt{-d}}{2})(a + b\frac{1-\sqrt{-d}}{2}) = a^2 + ab + \frac{d+1}{4}b^2 & D \equiv 3 \pmod{4}. \end{cases}$$

2.2 The CM method

The theory of Complex Multiplication for curves over finite fields was developed in the 90's, at a time where point counting algorithms were still under improvements and required a huge computing power. With the development of the Schoof, Elkies, and Atkin algorithm (SEA) for point counting, the Complex Multiplication was never massively employed to obtain cryptographic elliptic curves, as it produces curves with a small discriminant. Despite not being directly a weakness, this makes a particularity that one prefers to avoid whenever possible. In the recent survey [BL24], discriminants $|D| \geq 2^{110}$ are considered safe, which is far beyond the feasibility of the CM method.

The CM method is crucial in the Elliptic Curve Primality Proving (ECPP) method. With input a probable prime q whose primality should be established, it consists in enumerating fundamental discriminants of increasing magnitude, until one gives valid parameters (a valid trace) for an elliptic curve over $\mathbb{Z}/q\mathbb{Z}$ (and other requirements on the smoothness of its order are met). Once D is known, the CM method consists in finding the elliptic curve coefficients a, b in the equation $y^2 = x^3 + ax + b$. A modular polynomial (Hilbert or Weber class polynomial computation) is computed. Its roots modulo q are the j -invariants of curves E over $\mathbb{Z}/q\mathbb{Z}$ having CM by $\sqrt{-D}$. The curve coefficients are deduced from j with the formula $E: y^2 = x^3 + \frac{3j}{j-1728}x + \frac{2j}{1728-j}$. The special cases for $j = 0$ and 1728 are done separately: curves with $j = 0$ have equation $y^2 = x^3 + b$ whereas curves with $j = 1728$ have equation $y^2 = x^3 + ax$. The complexity of the CM method comes from the computation of the invariant j from class field theory. Following the notations in [AM93], the Hilbert Class Field of K is the maximal unramified Abelian extension of K and is denoted by K_H . the Hilbert Class Polynomial denoted H_{-D} , is such that K_H is its splitting field. The roots of $H_{-D}(X) \pmod{q}$ are the j -invariants of the elliptic curves over \mathbb{F}_q having discriminant $-D$.

Remark 1 (Small discriminant 2-cycles.). When an elliptic curve E_1/\mathbb{F}_q is of prime order r , there always exists another elliptic curve E_2/\mathbb{F}_r of order q , i.e. a 2-cycle between E_1 and E_2 . This result comes from the CM method. Let t be the trace of E_1/\mathbb{F}_q so that

$q + 1 - t = r$. Writing $r + 1 - (2 - t) = q$, we obtain using the CM method another curve defined over \mathbb{F}_r of trace $2 - t$. As long as D is small enough, we can compute the curve coefficients using the Hilbert class polynomial. In practice, this polynomial can be computed directly modulo q using [Sut11], and the largest computation was done for a discriminant with a dozen of digits.

2.3 Solving norm equations in imaginary quadratic orders

The solution of a norm equation of the form $x^2 + dy^2 = q$ is closely related to number theory. In particular, these equations are related to imaginary quadratic fields, and this theory has been intensively studied in the context of norm equations. Here, we look for class of solutions of equations of the form $x^2 + dy^2 = q$. In particular, we will consider a result of [AM93], that we adapt to our context.

Proposition 1 ([AM93, Proposition 2.3]). *Let $D, K, \mathcal{O}_K, \omega$ as in Section 2.1. Let q be a rational prime. The equation $q = N_{K/\mathbb{Q}}(\pi)$ has a solution π in \mathcal{O}_K if, and only if, (q) splits as the product of two principal ideals $\pi, \tau(\pi)$ in K . In other words: $4q = A^2 + DB^2$ with A and B in \mathbb{Z} .*

Theorem 1 ([AM93, Theorem 3.2]). *Let $D, K, \mathcal{O}_K, \omega$ as in Section 2.1. Let K_H, H_D as in Section 2.2. A rational prime q is a norm in K if and only if (q) splits completely in K_H . This is equivalent to saying that $H_D(X) \pmod{q}$ has only simple roots and they are all in $\mathbb{Z}/q\mathbb{Z}$. Moreover, we have that*

$$4q = A^2 + DB^2$$

has a solution in rational integers (A, B) if and only if $H_D(X)$ splits completely modulo q .

We re-phrase the above results as Theorem 2 to suit our needs.

Theorem 2 (deduced from Proposition 1 and Theorem 1). *Let K, d, D as in Section 2.1. There exists $\pi \in \mathcal{O}_K$ of norm q if, and only if, there exist $t, y \in \mathbb{Z}$ satisfying $t^2 + Dy^2 = 4q$.*

Proof.

(\implies) Suppose that there exists $\pi = a + b\omega \in \mathcal{O}_K$ ($a, b \in \mathbb{Z}$) of norm q . If $D \not\equiv 3 \pmod{4}$, then the norm of π is $q = a^2 + db^2$ where $d = D/4$. Multiplying by 4, we get $4q = 4a^2 + Db^2$. Then, we simply choose $t = 2a$ and $y = b$. If $D \equiv 3 \pmod{4}$, $D = d$ and the norm of π is $q = a^2 + ab + b^2 \frac{D+1}{4} = (a + b/2)^2 + Db^2/4$. Multiplying by 4, we get $4q = (2a + b)^2 + Db^2$. In this case, we choose $t = 2a + b$ and $y = b$.

(\impliedby) Reciprocally, looking at the equation $t^2 + Dy^2 = 4q$ modulo 4, for $D \not\equiv 3 \pmod{4}$, $D = 4d$ and $D = 4, 8 \pmod{16}$, one can show that t should be even, so that we set $\pi = \frac{t}{2} + \omega y \in \mathcal{O}_K$ having norm q . For $D \equiv 3 \pmod{4}$, we show that t, y should have same parity to satisfy $t^2 + Dy^2 = 0 \pmod{4}$. Hence, we set $\pi = \frac{t-y}{2} + \omega y \in \mathcal{O}_K$, having norm $\frac{(t-y)^2}{4} + \frac{t-y}{2}y + \frac{D+1}{4}y^2 = t^2/4 + Dy^2/4 = q$ as required.

□

Remark 2. If a curve defined over \mathbb{F}_q has discriminant $-D$ and trace t , we know that $t^2 - 4q = -Dy^2$ for an integer y . From the proof of Theorem 2, we know that if $D \not\equiv 3 \pmod{4}$, then t must be even. In consequence, the order of the curve must be even. Reciprocally, the curve will have prime order only if $D \equiv 3 \pmod{4}$.

2.4 Pairing-friendly curves

In this section, we consider an elliptic curve E defined over a prime field \mathbb{F}_p . Pairing-friendly curves are such that the Tate or Weil pairings and their variants are computable in reasonable time. For that, the curve *embedding degree* k with respect to a subgroup of points of order q should be small, say $k \leq 54$. The embedding degree k with respect to q is the smallest extension degree of the base field \mathbb{F}_p such that all q -torsion points are \mathbb{F}_{p^k} -rational. Usually for efficiency implementations, k is chosen to be a power of 2 and 3, for example $k = 12, 24$. In this work, we consider several families of pairing-friendly curves: Barreto–Naehrig (BN), Barreto–Lynn–Scott (BLS), Kachisa–Schaefer–Scott (KSS) whose parameter sets are given in Tables 1, 2, and 3. Popular seeds are:

- $u = 0x44e992b44a6909f1$ to generate a BN254 curve,
- $u = -0xd201000000010000$ for a BLS12-381,
- $u = 0x8508c00000000001$ for BLS12-377.

The number after the family name in the label refers to the bitsize of the prime $p(u)$. Evaluating the polynomials $p(x)$, $t(x)$ and $q(x)$ at the seed u , we obtain three parameters p , t and q . We can derive the corresponding elliptic curve using the CM method described in Section 2.2.

Table 1: BN curve parameters

$k = 12$
$-D = -3$
$q(x) = 36x^4 + 36x^3 + 18x^2 + 6x + 1$
$t(x) = 6x^2 + 1$
$p(x) = 36x^4 + 36x^3 + 24x^2 + 6x + 1$
$c(x) = 1$
$y(x) = 6x^2 + 4x + 1$

Table 2: Selected BLS curve parameters

$k = 12$	$k = 24$
$-D = -3$	$-D = -3$
$q(x) = \Phi_{12}(x) = x^4 - x^2 + 1$	$q(x) = \Phi_{24}(x) = x^8 - x^4 + 1$
$t(x) = x + 1$	$t(x) = x + 1$
$p(x) = (x - 1)^2/3(x^4 - x^2 + 1) + x$	$p(x) = (x - 1)^2/3(x^8 - x^4 + 1) + x$
$c(x) = (x - 1)^2/3$	$c(x) = (x - 1)^2/3$
$y(x) = (x - 1)(2x^2 - 1)/3$	$y(x) = (x - 1)(2x^4 - 1)/3$
$u = 1 \pmod{3}$	$u = 1 \pmod{3}$

In the next section, we consider the problem of finding an embedded curve $E_1(\mathbb{F}_q)$ relatively to a fixed elliptic curve $E(\mathbb{F}_p)$ of order divisible by q . We follow the complex multiplication method together with Theorem 2.

2.5 Bandersnatch: a curve embedded on BLS12-381 scalar field

In [MSZ24], the authors investigate an embedded curve for the BLS12-381 curve. In this context, they look for a curve above the BLS12 curve with a fast scalar multiplication, and so the base field of the new curve, called Bandersnatch, is fixed to be the scalar field

Table 3: Selected KSS curve parameters

$k = 16$	$k = 18$
$-D = -4$	$-D = -3$
$q(x) = (x^8 + 48x^4 + 625)/61250$	$q(x) = (x^6 + 37x^3 + 343)/343$
$t(x) = (2x^5 + 41x + 35)/35$	$t(x) = (x^4 + 16x + 7)/7$
$p(x) = (x^{10} + 2x^9 + 5x^8 + 48x^6 + 152x^5$ $+ 240x^4 + 625x^2 + 2398x + 3125)/980$	$p(x) = (x^8 + 5x^7 + 7x^6 + 37x^5 + 188x^4$ $+ 259x^3 + 343x^2 + 1763x + 2401)/21$
$c(x) = 125(x^2 + 2x + 5)/2$	$c(x) = 49(x^2 + 5x + 7)/3$
$y(x) = (x^5 + 5x^4 + 38x + 120)/35$	$y(x) = (5x^4 + 14x^3 + 94x + 259)/21$
$u = 24, 45 \pmod{70}$	$u = 14 \pmod{21}$

of BLS12-381. The authors obtain an embedded curve using the Complex Multiplication method of Section 2.2. In [MSZ24], Algorithm 1 iterates on various discriminants and computes the Hilbert Class polynomials in order to compute the order of the curves. Using their technique, they are able to derive an elliptic curve embedded above BLS12-381, with almost optimal structure.

2.5.1 Parameters.

The Bandersnatch curve is defined over \mathbb{F}_q where q is the large prime factor of the order of BLS12-381. Its j -invariant is 8000, and [MSZ24] provides a representation of the curve in Weierstrass model with the equation $y^2 = x^3 - 3763200000x - 7867596800000$. In practice, other representations are usually preferred [CS18] for a more efficient elliptic curve group law. For Bandersnatch, we obtain an isomorphic curve with the equation $y^2 = x^3 + 5x + b$ where b is `0x6a8d275fe8126c2c0022c15e1f181e282fb81761827fdf4ccdf7834600226d91`.

2.5.2 Endomorphism.

The curve has endomorphism ring $\mathbb{Z}[\sqrt{-2}]$, i.e. a fundamental discriminant $-D = -8$, and the endomorphism of multiplication by $\sqrt{-2}$ can be computed efficiently. This enables faster scalar multiplications than on Jubjub [ZCa21], also defined over the scalar field of BLS12-381, but with a larger discriminant (as it was obtained with SEA point counting).

2.5.3 Security.

The curve has order $4r$, where r is prime. This lets us represent the curve in the Montgomery model, and avoids subgroup attacks with appropriate strategies such as Decaf [Ham15] or Ristretto [Val21]. The quadratic twist has order $2^7 \cdot 3^3 \cdot r'_{244}$ for a prime r'_{244} of 244 bits. This avoids twist subgroup attacks [BL24, Section 9].

It seems unlikely to have all these conditions at a time for one curve, given the base field \mathbb{F}_q (tiny D , order of the form $4r$ with r prime, twist-security). In Section 3.3.3, we provide arguments for showing that there is no polynomial structure on Bandersnatch parameters (except $q = q(u)$ of Table 2). Moreover, we explain the scarcity of such curves and prove that although it was lucky to find one with a small discriminant $-D = -8$, we are able to find another one with the same properties on its order and twist order, for a larger discriminant, as expected.

Originally, [MSZ24, Algorithm 1] searches for secure embedded curves by computing the candidate curve orders using the Hilbert class polynomial. This computation become cumbersome as long as D increases. We adapt the original algorithm into Algorithm 1 in order to obtain a faster search in Section 3. Our algorithm produces the same output as

in [MSZ24, Table 2], but the computation is much faster and thus we can consider much larger discriminants.

3 Generation of embedded curves for a variable discriminant

3.1 The problem

In this section, we consider a fixed elliptic curve defined over a prime field, and we look for an embedding above or under this curve. More precisely, we consider two cases where a curve is fixed and we look for an embedded curve above it, and one case where we look for a curve under the fixed curve.

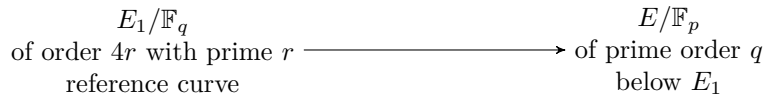
3.1.1 Prime order embedded curve above a given curve.

Given an elliptic curve E defined over \mathbb{F}_p of composite order hq where q is prime, find an embedded curve E_1 of prime order r defined over \mathbb{F}_q . From Remark 1, this automatically leads to a 2-cycle with E_1 and another curve E_2 (defined over \mathbb{F}_r of order q), if the discriminant is small enough. We provide in Section 3.3.1 an example when E is the pairing-friendly curve BLS12-381.



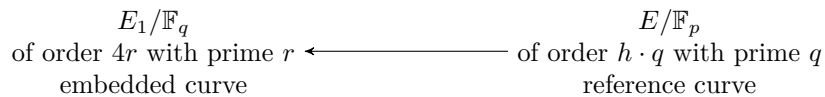
3.1.2 Prime order curve below a given curve.

We also consider the case where E_1 , defined over \mathbb{F}_q and of order $4r$ for a prime r , is fixed. In this case, we look for a curve E defined over a prime field \mathbb{F}_p of order q . Again, this produces a 2-cycle with E as we will see in the example of Section 3.3.2, where we consider Ed255-19 for E_1 .



3.1.3 Composite order embedded curve above a composite order curve.

When E is defined over \mathbb{F}_p and has a composite order, it is not possible to obtain a 2-cycle, and so in this context, we look for a curve E_1 of order $4r$ where r is prime, above E . Bandersnatch falls in this setting. We provide in Section 3.3.3 a curve embedded above BLS12-381, with similar security features as Bandersnatch, but a larger discriminant. This lets us understand better the scarcity of Bandersnatch.



In order to generate these embeddings, we start with the idea of [MSZ24, SEH24], and we modify the original algorithm in Section 3.2 in order to improve the exhaustive search of curves.

3.2 The method

Given a prime q , we consider the Complex Multiplication method. In other words, we look for a discriminant $-D$ such that there exists an elliptic curve whose endomorphism ring is an order of discriminant $-D$. From the CM theory, the elliptic curve trace t will satisfy $t^2 - 4q = -Dy^2$ for an integer y . We find it interesting to remember that solving a CM equation was needed in the 90's for the ECPP algorithm [AM93, §8.4.2]. We solve this equation using Theorem 2. We find the corresponding element π using lattice reduction. Indeed, an algebraic integer of norm q is a generator of the ideal of norm q , and is a shortest non-zero element (in terms of coefficient size) of that ideal. A generator may not exist when the ring \mathcal{O}_K is not principal. Using Cohen's notation, this ideal can be represented by $I = \mathbb{Z}\langle q, X - s \rangle$, where $X - s \in \mathbb{F}_q[X]$ is a factor of $X^2 + d \pmod q$ when $d \not\equiv 3 \pmod 4$, of $X^2 - X + \frac{d+1}{4} \pmod q$ when $d \equiv 3 \pmod 4$. We look for the shortest element of I by reducing the lattice defined by I . When an element of norm q exists, we deduce t, y as we did in the proof of Theorem 2. Finally, we use the Hilbert class polynomial computation modulo q in order to get the j -invariant and curve coefficients. Our method is summarized in Algorithm 1.

Algorithm 1: EmbeddedCurve(q, d_{\min}, d_{\max})

Input: prime integer q , minimum and maximum values of $d > 0$

Output: A list of traces and discriminants of embedded elliptic curves for \mathbb{F}_q

$\mathcal{L} \leftarrow \{\}$

for d from d_{\min} **to** d_{\max} **do**

if d is square-free and $-d$ is a square modulo q **then**

$$s \leftarrow \begin{cases} \sqrt{-d} \pmod q & d \not\equiv 3 \pmod 4 \\ \frac{1+\sqrt{-d}}{2} \pmod q & d \equiv 3 \pmod 4 \end{cases}$$

 lift s in \mathbb{Z}

$\pi \leftarrow a + bX$ the shortest non-zero element of the lattice $\mathbb{Z}\langle q, X - s \rangle$

if π has norm q **then**

$$(t, y) \leftarrow \begin{cases} (2a, b) & \text{if } d \equiv 3 \pmod 4 \\ (2a + b, b) & \text{otherwise} \end{cases}$$

$\mathcal{L} \leftarrow \mathcal{L} \cup \{(d, t, y)\}$

return \mathcal{L}

While the order of the embedded curve is known (it is $q + 1 - t$), we need to compute the Hilbert class polynomial modulo q in order to get the actual curve coefficients, and this is way much slower (and resource-consuming) than finding (t, y) . This computation is possible only up to discriminants of a dozen of digits [Sut11]. We remark that for $D \not\equiv 3 \pmod 4$, the resulting curves have even trace and so the order of the curve is always even. Additional conditions on the obtained order (for instance on its factorization) can be included in Algorithm 1 in order to refine the search for specific curves. In Sections 3.3.1 and 3.3.2, we apply Algorithm 1 for two base fields corresponding to practical use-cases of zero-knowledge proofs.

3.3 Practical results

In this section, we consider three use-case related to elliptic curves used in practice. First, we look for an embedded plain cycle above BLS12-381. Then, we consider a similar case with Ed255-19. Finally, we look for an embedded curve with the same properties as Bandersnatch in order to understand better its scarcity.

3.3.1 Embedded plain cycle above BLS12-381.

In this section, we look for a plain 2-cycle above BLS12-381, as depicted in Figure 2. From Remark 2, we need $D \equiv 3 \pmod{4}$ in order to get a prime order curve. From Remark 1, we know that a 2-cycle automatically exists. We observe that such a setting may become needed in future settings [NPR24].

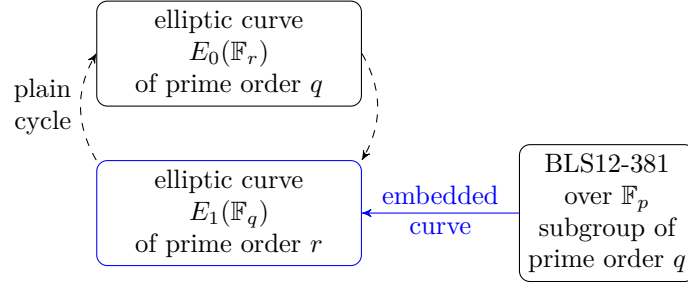


Figure 2: Plain cycle above BLS12-381.

In order to get this curve, we consider various discriminants in the range $1 - 10^8$, with the condition $D \equiv 3 \pmod{4}$. Using the method of Algorithm 1, we expect to find a prime order curve after $\ln(q) \approx 177$ discriminants. After a few hours of computation on one core of an Intel i7-1365U, we obtain the following parameters for a cycle of curves above BLS12-381. We enumerated the roots j of $H_{-D}(X) \pmod{q}$ until choosing $a = -3$ was possible.

We obtain a plain cycle of elliptic curves for $D = 6673027$. The curve $E_1/\mathbb{F}_q : y^2 = x^3 + b_q$ (resp. $E_0/\mathbb{F}_r : y^2 = x^3 - 3x + b_r$) is defined over \mathbb{F}_q (resp. \mathbb{F}_r) and has order r (resp. q). By default, the curves are subgroup secure because they are prime order curves. While E_1 is also twist-secure (its quadratic twist order is $3^2 \cdot 19^2 \cdot 953 \cdot r_{234}$ where r_{234} is a prime of 234 bits), E_0 does not fulfill this requirement, and the largest factor of the order of the quadratic twist of E_0 is 176-bit long. The parameters q , r , b_q and b_r are the followings:

```

q : 0x73eda753299d7d483339d80809a1d80553bda402fffe5bfeffffffffff00000001,
r : 0x73eda753299d7d483339d80809a1d80496b5714d26546fcc43d6b3e6dd7e79ed,
b_q : 0x181db5d04907341a0a65de398d54bad311b5cf755ded77e2b1e865971c5d3bd4,
b_r : 0x4c4e0682a1ae35f11ce41de53abb7e6cb6f90abc7280629d0fed4785715a4468.

```

We mention that while these curves are obtained using the CM method, the corresponding endomorphism might be expensive to compute. Thus, the GLV acceleration, at the origin of [MSZ24], would probably not apply and scalar multiplications would probably be slower than on Bandersnatch. The rational functions defining the endomorphism have degree $o(D)$, and the endomorphism might be defined over an extension of \mathbb{F}_q . This computation can be fastened by decomposing the endomorphism as a chain of smaller degree isogenies. Still, this computation would require a lot of computation, and the GLV technique would not improve the scalar multiplication.

In many proof systems, the Fast Fourier Transform is used in order to improve the efficiency of the polynomial multiplications. This requires additional conditions on the fields \mathbb{F}_q and \mathbb{F}_r . More precisely, implementations usually require 2^{32} -th roots of unity defined over these two fields. This property, available for instance in [Hop20b], adds an additional condition on q and r : $q - 1$ and $r - 1$ must be divisible by 2^{32} . While q is fixed in our context (and 2^{32} divides $q - 1$), this condition on r happens with probability 2^{-32} . In order to obtain a prime order curve with this 2-adicity condition, we would iterate on larger discriminant (try roughly 2^{32} times more discriminants congruent to $3 \pmod{4}$). We would obtain a curve with a discriminant of more than twelve digits,

reaching the records of Hilbert class polynomial computation. We did not investigate further this computation, as it is very cumbersome. Moreover, this condition on 2-adicity might be avoided using [BCKL22], a novel technique for computing on-circuit polynomial multiplications. In this case, the 2-adicity condition on r is not required.

3.3.2 Embedded plain cycle above Ed255-19.

Another interesting application is related to the standardized Ed25519 curve. We use Algorithm 1 in a similar way to Section 3.3.1, in order to generate a prime order elliptic curve whose scalar field is $\mathbb{F}_{2^{255}-19}$, the base field of Ed25519, as depicted in Figure 3. As in Section 3.3.1, we obtain by construction a 2-cycle because $E_1(\mathbb{F}_p) = 2^{255} - 19$ is prime.

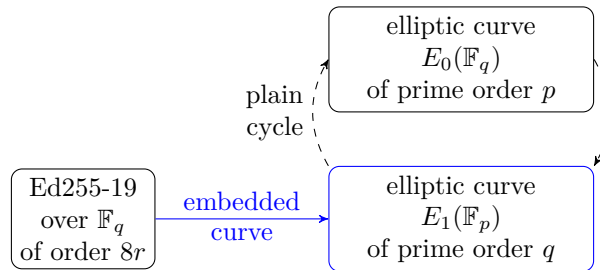


Figure 3: Plain cycle under Ed255-19.

We scan the square-free discriminants $D \equiv 3 \pmod{4}$ in the range $1 - 10^{10}$. After a day of computation, we obtain an elliptic curve for $D = -65012179$. The discriminant is larger than in the case of Section 3.3.1. We were able to compute the Hilbert class polynomial in a few seconds using PARI-GP. The curve coefficients were deduced by finding the roots of the polynomial, following the method of Section 2.2. Finally, the two curves of the cycles are $E_1/\mathbb{F}_p : y^2 = x^3 - 3x + b_p$ (of order q) and $E_0 : \mathbb{F}_q : y^2 = x^3 - 3x + b_q$ (of order p). The coefficients defining the curves are:

```
p : 0x7fffffffffffffffffffffffffffffffff34a2208109393ca351aa6d362f601a5f,
q : 0x7fffffffffffffffffffffffffffffffff8c9789795714e497c009e88009eb98a3e,
b_p : 0x1d426f89bb7e48f1cc1a5ec1b850994254b762353cab0b1a0bca0ea8d50ad73c,
b_q : 0x475335deba31abe6a6e06ea61b48032598e7920645cdb8f81f2444aaa8cb0345.
```

Note that E_1 is also twist secure. More precisely, the order of its quadratic twist is prime. E_2 is also twist secure in the sense that its quadratic twist has a prime factor of 215 bits. While this curve has nice security features (prime order curve, and twist security), it has the same issue as in Section 3.3.1: the fields \mathbb{F}_q and \mathbb{F}_p do not allow fast polynomial multiplications ($p - 1$, $q - 1$ do not have large 2-valuation). We did not investigate further this additional property as it requires a much larger discriminant. As well as for the embedded cycle above BLS12-381, the endomorphism is expensive to compute and so the GLV technique cannot be applied for fast scalar multiplications. As mentioned in Section 3.3.1, for this size of discriminant, the endomorphism degree is too large, and evaluating this endomorphism requires too many operations. However, we stress that without the CM method with a manageable-size discriminant, it would not be possible to obtain the curve parameters of the second cycle curve E_0 .

3.3.3 Scarcity of subgroup-secure twist-secure embedded curves.

In this section, we investigate the scarcity of Bandersnatch. As we have seen in Section 2.5, Bandersnatch has several properties related to security and efficiency. In this section, we prove that finding such a curve was unlikely because it has a very small discriminant,

allowing fast scalar multiplications. However, we also show that it is just a matter of few hours of computation in order to get a similar curve (with a larger discriminant). This result is also showing that Bandersnatch is not *too good to be true*. This expression was used to qualify the BN curves by Menezes in 2007, and then Aranha in 2017: “These curves should not exist, they are too good to be true”. Later when Costello and Longa came with the FourQ curve, again it was said to be a *too good to be true* curve. FourQ was lucky to have a tiny discriminant for a chosen base field over the Mersenne prime $2^{127} - 1$.

Bandersnatch has a fast endomorphism $\sqrt{-2}$ that enables fast scalar multiplications. While this endomorphism was targeted by the authors of [MSZ24], it avoids any polynomial structure on the parameters of the curve. As we will see in Section 4, we can represent the parameters of the embedded curve as polynomials of $\mathbb{Q}[X]$, evaluated at the seed u only when $-D$ is a square in $\mathbb{Q}[X]/(q(X))$. In the case of Bandersnatch, -2 is not a square, meaning that there is no *hidden* structure as one variable polynomial representation.

While Bandersnatch was found for a small discriminant, it has also good security features: the order has a subgroup of 254-bit order, and its quadratic twist has also a subgroup of 253-bit order. This provides a good subgroup and twist security. Meeting these two conditions happens with probability $1/\log(q)^2 \approx 2^{-14}$, which is quite small. Note that this rough probability estimation does not take into account the fact that discriminants might not lead to solution of the norm equation in Theorem 2. Estimating this probability depends on the discriminant, more precisely on the genera number $g(-D)$ and the class number $h(-D)$. We refer to [AM93] for further details. More precisely, [Mor06, §2, §3.7] (FastECP) recalls that the probability that a prime q splits in the Hilbert class field K_H of discriminant $-D$ is $1/(2h(-D))$ where $h(-D)$ denotes the class number. To increase the probability, the strategy of ECP and FastECP is to select a subset of small primes d_i and combine them to form smooth discriminants. We do not go further in this direction as we were not limited by the class number (a basic iteration over discriminants of increasing magnitude was enough). However for finding a twist-secure embedded curve with high 2-adicity, it can become useful to apply this strategy. In conclusion, given a prime q , finding an embedded curve with the CM method seems not harder than finding an appropriate curve for the ECP method, that does work well in practice, so we again stress that finding CM embedded curves is not a surprise. In addition, ECP works for very large primes such as 2000 decimal digits, while here q is only 256 bits long.

With the technique presented in Section 3, we are able to iterate on many discriminants until we find an embedded curve. We obtain a curve similar to Bandersnatch with a larger discriminant $D = 4 \cdot 1030258$. The curve is $E_1 : \mathbb{F}_q : y^2 - 3x + b$ and has order $4r$ and its quadratic twist has order $2^3 \cdot 7 \cdot r'_{250}$ where r'_{250} is a prime number of 250 bits. The coefficients of the curve are the following:

```
q : 0x73eda753299d7d483339d80809a1d80553bda402fffe5bfeffffffffff00000001,
r : 0x1cfb69d4ca675f520cce7602026876011d928688cbe65dba241d31a34658145b,
b : 0x284504e3feb2feda870885babe7e63bb3a55276335d818514a9313120e044ae9.
```

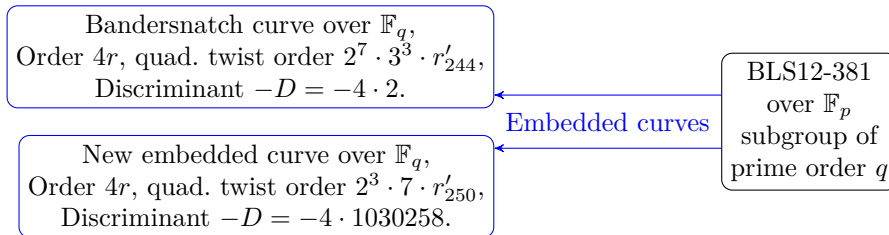


Figure 4: Two secure embedded curves above BLS12-381.

In this context, the FFT condition is not really an issue, as we are not considering a 2-cycle of curves, but simply an embedded curve as in the case of Bandersnatch. The existence of such a curve provides some evidence that while Bandersnatch is a “lucky” curve, the properties on its order and the quadratic twist order were possible to achieve at the cost of a slower endomorphism. In practice, Bandersnatch is of course preferred, for its fast scalar multiplications.

4 Families of embedded curves for a fixed discriminant

4.1 Previous works

4.1.1 Embedded families for BLS12.

In [SEH24], Sanso and El Housni introduce a technique to obtain families of endomorphism-equipped embedded curves with BLS. They apply the technique to BLS12 and BLS24. They observe that the scalar field of BLS12 curves $q(u) = u^4 - u^2 + 1$ can be written in the form $q(u) = (t_e^2 + 3y_e^2)/4 = ((2u^2 - 1)^2 + 3(1)^2)/4$ to generate an embedded curve family with $-D = -3$, and $q(u) = (t_e^2 + 4y_e^2)/4 = ((2u)^2 + 4(1 - u^2)^2)/4$ to generate an embedded curve family with $-D = -4$. Observe that these two equations allow to solve the norm equation of Section 2.3.

We rephrase Sanso and El Housni procedure as Algorithm 2. The output for BLS12 is Table 4 for embedded curves with j -invariant 0 ($-D = -3$). Two families can produce prime-order embedded curves. For $-D = -4$, Sanso and El Housni procedure will output Table 5. curves with j -invariant 1728 cannot have prime order as they always have at least one point of order two and an even order. One can note that the order is $u^4 - 3u^2 + 4 = u^2(u^2 - 1) - 2u^2 + 4$ which is always even whenever the parity of u .

Algorithm 2: Generating prime-order endomorphism-equipped embedded curves with BLS or KSS [SEH24]

Input: parameterized pairing-friendly curve order $q(u)$ that generates primes, discriminant $-D$ for the embedded curve

Output: Embedded curve families of discriminant $-D$ or \perp

if $-D$ is a square in $\mathbb{Q}[x]/(q(x))$ **then**

$W(x) \leftarrow \sqrt{-D} \bmod q(x)$

$(t(x), y(x)) \leftarrow \text{half-gcd}(W(x), q(x))$

if $t(x)^2 + Dy(x)^2 = 4q(x)$ **then**

for $t_e(x)$ in the set of traces of twisted curves with respect to $t(x)$ **do**

$q_e(x) \leftarrow q(x) + 1 - t_e(x)$

if $q_e(u)$ is irreducible **then**

Append (t_e, y_e, q_e) to the list of families

return the list of families

return \perp

4.1.2 Finding a square root of $-D$ modulo $q(x)$ in Algorithm 2.

Looking at Algorithm 2, there are two steps that can fail. The first is testing if $-D$ is a square in $\mathbb{Q}(x)/(q(x))$. We note that it is a much stronger condition than asking for $-D$ being a square modulo a prime integer $q = q(u)$ for some seed u . For example, $-D = -2$ is not a square modulo $q(x) = \Phi_{12}(x) = x^4 - x^2 + 1$ however is it a square modulo $q(u_0)$ where $u_0 = -0\text{xd}20100000010000 = -(2^{63} + 2^{62} + 2^{60} + 2^{57} + 2^{48} + 2^{16})$ is the seed of the BLS12-381 curve. Considering the Legendre symbol and the law of quadratic reciprocity, -2 is a square modulo a prime q if and only if $q = \pm 1 \pmod 8$. Back to the polynomial form

Table 4: Embedded curve families above BLS12 with $q(x) = x^4 - x^2 + 1$ and $-D = -3$. A first pair is $(t_e, y_e) = (2x^2 - 1, 1)$ and the other pairs are for the quadratic, cubic and sextic twists. The fourth one's order $q + 1 - t_e = x^4 - 2x^2 + 4$ is not prime but can give three times a prime (when evaluated at a seed $u = 1 \pmod{3}$).

(t_e, y_e) s.t. $q = (t_e^2 + 3y_e^2)/4$	$q + 1 - t_e$	family	
t, y	$2x^2 - 1, 1$	$x^4 - 3x^2 + 3$	yes
$-t, y$	$-2x^2 + 1, 1$	$(x^2 - x + 1)(x^2 + x + 1)$	no
$(t + 3y)/2, (t - y)/2$	$x^2 + 1, x^2 - 1$	$(x - 1)^2(x + 1)^2$	no
$(t - 3y)/2, (t + y)/2$	$x^2 - 2, x^2$	$x^4 - 2x^2 + 4$	(yes)
$-(t - 3y)/2, (t + y)/2$	$-x^2 + 2, x^2$	x^4	no
$-(t + 3y)/2, (t - y)/2$	$-x^2 - 1, x^2 - 1$	$x^4 + 3$	yes

Table 5: Embedded curve families above BLS12 with $q(x) = x^4 - x^2 + 1$ and $-D = -4$. A first pair is $(t, y) = (2x^2 - 2, x)$ and the other pairs are for the quadratic and quartic twists. The first one's order $q + 1 - t_e$ is not prime but can give two times a prime.

(t_e, y_e) s.t. $q = (t_e^2 + 4y_e^2)/4$	$q + 1 - t_e$	family	
t, y	$2x^2 - 2, x$	$x^4 - 3x^2 + 4$	(yes)
$-t, y$	$-2x^2 + 2, x$	$x^4 + x^2 = x^2(x^2 + 1)$	no
$2y, t/2$	$2x, x^2 - 1$	$x^4 - x^2 - 2x + 2 = (x - 1)^2(x^2 + 2x + 2)$	no
$-2y, t/2$	$-2x, x^2 - 1$	$x^4 - x^2 + 2x + 2 = (x + 1)^2(x^2 - 2x + 2)$	no

of $q(x)$, we deduce that $q(u) \equiv 1 \pmod{4}$ for any u , and $q(u) \equiv 1 \pmod{8} \iff u \not\equiv 2 \pmod{4}$. However, this does not make a family. To design a family of embedded curves with $-D = -2$ for BLS12 curves, one example could be to write $q(x^2) = x^8 - x^4 + 1$ (replace the variable x by x^2 everywhere i.e. assume the seed is a square) then apply Algorithm 2 with $\sqrt{-2} \equiv x^5 + x^3 - x \pmod{q(x)}$, a half-gcd gives directly $q(x) = (x^4 - x^2 + 1)^2 + 2(x^3 - x)$, and $(t, y) = (2(x^4 - x^2 + 1), 2(x^3 - x))$.

4.1.3 Solving for polynomials $(t(x), y(x))$ in the norm equation $q(x) = (t(x)^2 + Dy(x)^2)/4$.

Sanso and El Housni suggest to compute a half-gcd of $q(x)$ and $W(x)$ to obtain candidates for $t(x), y(x)$ such that their degree is at most half the degree of $q(x)$. We recall that this strategy is well-known for example in cryptanalysis, in the descent step of a discrete logarithm computation. The first occurrence of this technique (applied to polynomials) is for the initial splitting step of discrete logarithm computation in $\text{GF}(2^n)$ and dates back to 1984. It is known under the name *Waterloo algorithm* from the University of Waterloo, ON, Canada, where the authors are from [BFHMV84, BMV84]. The idea is to express the target (a polynomial in $\mathbb{F}_2[x]$ of even degree $n - 1$) as the ratio of two polynomials of degree $(n - 1)/2$, modulo an irreducible polynomial of odd degree n . The aim is to increase the smoothness probability.

In the present case q has usually an even degree, and a half-gcd algorithm on inputs $(q(x), W(x))$ with $\deg q > \deg W$ outputs three polynomials $I(x), U(x), V(x)$ such that $I(x)q(x) = U(x) - V(x)W(x)$ with usually $\deg(I) = 1$, $\deg U, \deg V \leq \deg q/2$. Luckily for BLS and BN, $I = 1$ and the equation $t^2 + Dy^2 = 4q$ is solved, with $t = 2U$ and $y = 2V$. But for KSS18 for example, $W = 2x^3 + 37$, $U = 3$, $V = -2x^3 - 37$, $I = 1372$.

4.2 Our Generic Method

4.2.1 A particular case: embedded curves above KSS18.

Building on Algorithm 2, Sanso and El Housni looked at KSS18 curves. The difficulty comes from finding a generic formula to express the parameterized KSS18 order $q = (x^6 + 37x^3 + 343)/343$ as a sum of two squares $q(x) = (t^2(x) + Dy^2(x))/4$. From Section 2, we can write

$$q(u) = (t^2 + 3y^2)/4 = ((t + y)/2)^2 - y(t + y)/2 + y^2 = a_0^2 - a_0a_1 + a_1^2 \quad (2)$$

and deduce that $(a_0, a_1) = ((t + y)/2, y)$. In other words, $(t, y) = (2a_0 - a_1, a_1)$. Then we recognize that (2) is exactly the formula of Dai, Lin, Zhao, and Zhou [DLZZ23, Remark 4] for \mathbb{G}_1 subgroup membership testing. Then we deduce that the formula Sanso and El Housni were looking for is

$$(a_0, a_1) = ((x/7)^3, -18(x/7)^3 - 1) \iff (t, y) = (20(x/7)^3 + 1, -18(x/7)^3 - 1) . \quad (3)$$

We deduce Algorithm 3 and run it to obtain the prime-order endomorphism-equipped embedded curves with KSS18 (Fig. 8).

Algorithm 3: Generating prime-order endomorphism-equipped embedded curve families with KSS18 and $-D = -3$

$q(x) \leftarrow (x^6 - 37x^3 + 343)/343$, a KSS18 curve order

$(t(x), y(x)) \leftarrow (20(x/7)^3 + 1, -18(x/7)^3 - 1)$

for $(t_e(x), y_e(x))$ in the set of 6 twist parameters of $(t(x), y(x))$ **do**

$q_e(x) \leftarrow q + 1 - t_e$

if $q_e(x)$ is irreducible **then**

Append (t_e, y_e, q_e) to the list of families

return the list of families

Table 6: Embedded curves above KSS18, $-D = -3$. A first pair is $(t, y) = (20(x/7)^3 + 1, -18(x/7)^3 - 1)$ and the other pairs are for the quadratic, cubic and sextic twists. The first and fifth one's order $q_e = q + 1 - t_e$ are irreducible but multiple of 3.

(t_e, y_e) s.t. $q = (t_e^2 + 3y_e^2)/4$	$q_e = q + 1 - t_e$	family
t, y	$20(x/7)^3 + 1, -18(x/7)^3 - 1$	$(x^6 + 17x^3 + 343)/343$ (yes, 3)
$-t, y$	$-20(x/7)^3 - 1, -18(x/7)^3 - 1$	$(x^6 + 57x^3 + 1029)/343$ yes
$(t + 3y)/2, (t - y)/2$	$-17(x/7)^3 - 1, 19(x/7)^3 + 1$	$(x^6 + 54x^3 + 1029)/343$ yes
$(t - 3y)/2, (t + y)/2$	$37(x/7)^3 + 2, (x/7)^3$	$x^6/7^3$ no
$-(t - 3y)/2, (t + y)/2$	$-37(x/7)^3 - 2, (x/7)^3$	$(x^6 + 74x^3 + 1372)/343$ (yes, 3)
$-(t + 3y)/2, (t - y)/2$	$17(x/7)^3 + 1, 19(x/7)^3 + 1$	$(x^2 - 4x + 7)(x^2 - x + 7)(x^2 + 5x + 7)/343$ no

To conclude we mention the *halographs* project of Daira Hopwood at [Hop20a], who already in 2020 obtained the formulas of prime-order j -invariant 0 embedded curves forming a plain cycle for BLS12 and KSS18. A careful look at the SageMath source code shows that it uses the same formulas as [SEH24] for BLS12. For KSS18, the change of variables $x \mapsto 7x$ allowed to obtain the formulas, avoiding the denominator issue that Sanso and El Housni faced.

4.2.2 Our general solution.

We stick together different pieces that come from the literature about elliptic curves and cryptography. In particular, we will explain the link with Smith technique [Smi15] and Dai, Lin, Zhao, and Zhou work [DLZZ23].

Dai, Lin, Zhao, and Zhou work over the integer values of the curve parameters. Their aim is to obtain an optimal formula for \mathbb{G}_1 subgroup membership testing that is, given a point P on $E(\mathbb{F}_p)$, check that $[q]P = \mathcal{O}$ without computing the full and costly scalar multiplication by q . For that, the endomorphism ϕ on the curve of characteristic polynomial χ_ϕ is used. This technique is known as the GLV method [GLV01]. The endomorphism ϕ has eigenvalue $\lambda_\phi \bmod q$. A Gaussian reduction gives two shorter scalars $a_0 + a_1\lambda_\phi \equiv 0 \bmod q$ however, as pointed out by Dai, Lin, Zhao, and Zhou, $[a_0]P + [a_1]\phi(P)$ might actually compute a small multiple $[sq]P$ instead of $[q]P$ and the test is not valid if s is not coprime to the curve cofactor. The authors of [DLZZ23] develop a criterion to test whether the short scalars (a_0, a_1) give a valid subgroup membership test. They propose an algorithm and a Magma implementation to compute the short scalars that pass the test.

We then observe that we face a very similar problem: with an elementary change of variables, finding (t, y) to define embedded curves correspond to finding the short scalars (a_0, a_1) to design a valid and optimal \mathbb{G}_1 subgroup membership testing. However as we are interested in defining families of embedded curves, we are interested in finding the scalars generically, parameterized by polynomials. For that we exploit Smith technique that dates back to an AGCT workshop at CIRM in Marseille Luminy in 2015 [Smi15].

We present our technique based on Smith idea for KSS16 and KSS18 curves. The general strategy follows the same procedure for other pairing-friendly curves. For these two curves the output is exactly what Dai, Lin, Zhao, and Zhou found with a Gaussian reduction on integers (Table 7).

Table 7: From [DLZZ23, Table 4], with $q = (x^8 + 48x^4 + 625)/61250$ for KSS16, $q = (x^6 + 37x^3 + 343)/343$ for KSS18.

Curve	$-D$	χ_ϕ	$\lambda \bmod q$	short vector (a_0, a_1)	criterion
KSS16	-4	$X^2 + 1$	$\sqrt{-1} = (x^4 + 24)/7$	$((31x^4 + 625)/8750, -(17x^4 + 625)/8750)$	$a_0^2 + a_1^2 = q$
KSS18	-3	$X^2 + X + 1$	$(-1 + \sqrt{-3})/2 = x^3 + 18$	$((x/7)^3, -18(x/7)^3 - 1)$	$a_0^2 - a_0a_1 + a_1^2 = q$

4.2.3 Smith technique.

Smith [Smi15] is interested in computing a ready-made short basis of the lattice whose long basis is given by the following \vec{b}_i , where λ_{ϕ_i} stands for the eigenvalue of the i -th endomorphism ϕ_i on the curve E .

$$\begin{cases} \vec{b}_1 &= (q, 0, \dots, 0) \\ \vec{b}_2 &= (-\lambda_{\phi_2}, 1, 0, \dots, 0) \\ \vec{b}_3 &= (-\lambda_{\phi_3}, 0, 1, 0, \dots, 0) \\ &\vdots \\ \vec{b}_d &= (-\lambda_{\phi_d}, 0, \dots, 0, 1) \end{cases}$$

In our case, there are two endomorphisms, $\phi_1 = \text{Id}$ and $\phi_2 = \phi$, of characteristic polynomial $\chi(T) = T^2 - t_\phi T + n_\phi$. We recall [Smi15, Theorem 2].

Theorem 3 ([Smi15, Th. 2]). *Let ϕ be a non-integer endomorphism of \mathcal{E} such that $\mathbb{Z}[\pi] \subset \mathbb{Z}[\phi]$, so $\pi = c\phi + b$ for some integers c and b . Suppose that we are in the situation of §1 with $\mathcal{A} = \mathcal{E}$ and $(\phi_1, \phi_2) = (1, \phi)$. The vectors*

$$\vec{b}_1 = (b - 1, c) \text{ and } \vec{b}_2 = (c \deg(\phi) + (b - 1)t_\phi, 1 - b)$$

generate a sublattice of \mathcal{L} of determinant $\#\mathcal{E}(\mathbb{F}_p)$. If $\mathcal{G} = \mathcal{E}(\mathbb{F}_p)$, then $\mathcal{L} = \langle \vec{b}_1, \vec{b}_2 \rangle$.

In [Smi15, Sect. 4], Smith provides a way for reducing the basis (\vec{b}_1, \vec{b}_2) in case of small co-factors $h = 2$ for example, and provides a general framework for the technique.

We clarify that Smith's technique starts from the curve endomorphism and the curve coefficients and defines the basis in a context where the curve is of prime order. In our case, we know the pairing-friendly curve coefficients and we are looking for the embedded curve coefficients.

Another point of view is to look for a generator of a principal ideal in $\mathbb{Q}(\sqrt{-D})$ of norm q . It will be of the form $\tau = c\omega + b$. But again as we are working with parameters in polynomial form, we follow Smith technique.

We consider the pairing-friendly curve parameters (p, t, q, y) where p defines the field characteristic, t the curve trace, q the prime order of the subgroup of embedding degree k , and y such that $t^2 - 4p = -Dy^2$ with square-free D . We compute $\sqrt{-D}$ modulo $q(x)$ in polynomial form. Actually $\#E(\mathbb{F}_p) = cq = ((t-2)^2 + Dy^2)/4$ so $\sqrt{-D} = (t-2)/y \pmod{q(x)}$. Inverting $y(x)$ is done with an extended Euclidean algorithm on $q(x), y(x)$. Then we run a half-gcd algorithm to obtain $\sqrt{-D} \equiv U(x)/V(x)$ of reduced degrees and U, V coprime. At this point we introduce Smith basis reduction technique. The first vector of the basis is $\vec{b}_1 = (U(x), -V(x))$. We need to complete the basis: the second vector is $(DV(x), -U(x))$. Observe that the determinant of

$$B = \begin{bmatrix} U(x) & -V(x) \\ DV(x) & -U(x) \end{bmatrix}$$

is $\det(B) = U^2(x) + DV^2(x)$ and is a multiple of $q(x)$. For each factor ℓ of the determinant, we reduce the basis. It consists in finding a left kernel of B in $\mathbb{Z}/\ell\mathbb{Z}$. At the end of this process we expect to obtain a reduced basis whose determinant is exactly $q(x)$.

For $D \equiv 3 \pmod{4}$ and characteristic polynomial $\chi = X^2 - t_\phi X + \deg_\phi$ of discriminant $t_\phi^2 - 4 \deg_\phi = -D$ with $t_\phi = -1$ and $\deg_\phi = (D+1)/4$, a variant can be used (to avoid a factor 4). Compute $(t_\phi + \sqrt{-D})/2 = \lambda$ as $U(x)/V(x)$ modulo $q(x)$. The first vector is $(U(x), -V(x))$. Multiply $U(x) - V(x)\lambda$ by the negative of the conjugate root $\lambda - t_\phi$ and observe that $-\lambda(\lambda - t_\phi) = \deg_\phi$: one obtains $U(x)\lambda - U(x)t_\phi + \deg_\phi V(x)$. The second vector is $(-t_\phi U(x) + \deg_\phi V(x), U(x))$ so that

$$B = \begin{bmatrix} U(x) & -V(x) \\ -t_\phi U(x) + \deg_\phi V(x) & U(x) \end{bmatrix}$$

and the determinant of the basis matrix B is $U^2(x) - t_\phi U(x)V(x) + \deg_\phi V^2(x)$. Once the matrix is reduced of determinant exactly q , we obtain the embedded curve coefficients from the formulas (2).

4.3 Practical results

4.3.1 Application to KSS18.

A curve like KSS18 with j -invariant 0 has complex multiplication (CM) by $\mathbb{Z}[(-1 + \sqrt{-3})/2]$. The Frobenius is $\pi = (-t + y\sqrt{-3})/2$ so that $\pi\bar{\pi} = (t^2 + 3y^2)/4$. For the embedded curve parameters we are looking for (t_e, y_e) such that $(t_e^2 + 3y_e^2)/4 = q$. We denote $\tau = (t_e + y_e\sqrt{-3})/4$. The endomorphism ϕ on KSS18 has characteristic polynomial $\chi = X^2 + X + 1$ and its eigenvalue is $\lambda_\phi = (-1 + \sqrt{-3})/2$. We obtain $\lambda = x^3 + 18$, already of degree $\deg q/2$. No half-gcd is required. The first basis vector is $\vec{b}_1 = (x^3 + 18, -1)$ and a second vector can be $\vec{b}_2 = (1, x^3 + 19)$. We define the basis

$$\begin{bmatrix} \lambda & -1 \\ \deg \phi & \lambda + 1 \end{bmatrix} = \begin{bmatrix} x^3 + 18 & -1 \\ 1 & x^3 + 19 \end{bmatrix}$$

whose determinant is $343q(x) = 7^3 \cdot q$. The aim is to reduce this basis by a factor 7^3 . We are looking for a linear combination

$$(\vec{b}_1 + j\vec{b}_2)/343 = ((j + 18i + i \cdot x^3)/343, (19j - i + j \cdot x^3)/343)$$

such that the denominator 343 will simplify and the coefficients will be integers. Note that $x \equiv 14 \pmod{21}$ hence $7 \mid x$, $343 \mid x^3$ and we are looking for $i, j \in \mathbb{Z}/343\mathbb{Z}$ satisfying

$$j + 18i \equiv 0 \pmod{343} \iff 19j - i = 0 \pmod{343} \text{ indeed } 1/18 = -19 \pmod{343} .$$

We have a degree of freedom on j as $i = 19j \pmod{343}$. We test all $1 \leq j < 343$, and keep the pairs such that $\vec{b}_{i,j} = (i\vec{b}_1 + j\vec{b}_2)/343 = (a_0, a_1)$ satisfies $a_0^2 + a_0a_1 + a_1^2 = q$ (with exactly q , not a multiple). Finally we obtain a solution whose coefficients are integer-valued assuming $x \equiv 14 \pmod{21}$ like for KSS18 curves.

$$\begin{aligned} (i, j) &= (19, 1) \\ \vec{b} &= (19\vec{b}_1 + \vec{b}_2)/343 = ((1 + 19\lambda)/7^3, (\lambda + 1) - 19) \\ &= (19(x/7)^3 + 1, (x/7)^3) . \end{aligned}$$

The pair $(a_0, a_1) = (19(x/7)^3 + 1, (x/7)^3)$ corresponds to a twist of the embedded curve given by Dai, Lin, Zhao, and Zhou parameters.

Table 8: Seeds u of Hamming weight ≤ 6 such that the KSS18 curve E/\mathbb{F}_p has a high 2-valuation $2^L \mid q - 1$ and admits a prime-order embedded curve E_1/\mathbb{F}_q of j -invariant 0 that has a plain cycle curve E_0/\mathbb{F}_{q_e} . All curves have $-D = -3$.

seed	L	equation $E_{\text{KSS}}/\mathbb{F}_p$	p (bits)	q (bits)	embedded curve equation E_1/\mathbb{F}_q	plain cycle curve equation E_0/\mathbb{F}_{q_e}
$q = (u^6 + 57u^3 + 1029)/343$						
-0xfdde07f8000 $-2^{44} + 2^{37} + 2^{33} + 2^{29} - 2^{23} + 2^{15}$	45	$y^2 = x^3 + 13$	348	256	$y^2 = x^3 + 13$	$y^2 = x^3 - 4$
$q = (u^6 + 54u^3 + 1029)/343$						
-0xfd7ffdee000 $-2^{44} + 2^{37} + 2^{35} + 2^{21} + 2^{16} + 2^{13}$	39	$y^2 = x^3 + 2$	348	256	$y^2 = x^3 + 7$	$y^2 = x^3 + 2$

4.3.2 Application to KSS16.

For KSS16 curves, the endomorphism has characteristic polynomial $\chi = X^2 + 1$. One obtains, with $\lambda_\phi = (x^4 + 24)/7$,

$$\vec{b}_1 = (1, \lambda_\phi) = (1, (x^4 + 24)/7), \quad \vec{b}_2 = (\lambda_\phi, -1) = ((x^4 + 24)/7, -1) .$$

The determinant of the matrix made of \vec{b}_1, \vec{b}_2 is $-1250q(x)$ and we are looking for a linear combination to simplify by $1250 = 2 \cdot 5^4$,

$$(i\vec{b}_1 + j\vec{b}_2)/1250 = (i + j(x^4 + 24)/7, i(x^4 + 24)/7 - j)/1250$$

such that the denominator 1250 will simplify and the coefficients will be integers. Note that $x \equiv 25, 45 \pmod{70}$ hence $x \equiv 5 \pmod{10}$, $5^4 \mid x^4$. With $x = 10x_0 + 5 = 5(2x_0 + 1)$,

$$\begin{aligned} (i\vec{b}_1 + j\vec{b}_2) &= (i + j(5^4(2x_0 + 1)^4 + 24)/7, i(5^4(2x_0 + 1)^4 + 24)/7 - j) \\ &= (i + j(5^4 + 24)/7, i(5^4 + 24)/7 - j) \pmod{1250} \end{aligned}$$

and we are looking for $i, j \in \mathbb{Z}/2 \cdot 5^4\mathbb{Z}$ satisfying

$$i + (5^4 + 24)/7j \equiv 0 \pmod{2 \cdot 5^4} \iff i + 807j \equiv 0 \pmod{2 \cdot 5^4} .$$

(Note that $((5^4 + 24)/7)^2 = -1 \pmod{2 \cdot 5^4}$ so that the two constraints are equivalent). We have a degree of freedom on j as $i = -807j = 443j \pmod{2 \cdot 5^4}$. We test the pairs (i, j)

and keep those such that $(a_0, a_1) = (i\vec{b}_1 + j\vec{b}_2)$ satisfies $a_0^2 + a_1^2 = q(x)$. We obtain integer valued parameters for $x \equiv \pm 25 \pmod{70}$ for KSS16:

$$\begin{aligned} (i, j) &= (31, 17), \\ \vec{b} &= (31\vec{b}_1 + 17\vec{b}_2)/1250 = ((31 + 17\lambda_\phi)/1250, (31\lambda_\phi - 17)/1250) \\ &= ((17(x/5)^4 + 1)/14, (31(x/5)^4 + 1)/14). \end{aligned} \quad (4)$$

Algorithm 4: Generating embedded curve families with KSS16 and $-D = -4$

```

 $q(x) \leftarrow (x^8 + 48x^4 + 625)/61250$ , a KSS16 curve order
 $(t(x), y(x)) \leftarrow ((31(x/5)^4 + 1)/7, -(17(x/5)^4 + 1)/14)$ 
for  $(t_e(x), y_e(x))$  in the set of 4 twist parameters of  $(t(x), y(x))$  do
   $q_e(x) \leftarrow q + 1 - t_e$ 
  if  $q_e(x)$  is irreducible then
    Append  $(t_e, y_e, q_e)$  to the list of families
return the list of families

```

We give in Table 9 the results of Alg. 4 applied to KSS16 parameters.

Table 9: Embedded curves for KSS16, parameters (t_e, y_e) such that $q = (t_e^2 + 4y_e^2)/4$ with $-D = -4$. A first pair is $(t, y) = ((31(x/5)^4 + 1)/7, -(17(x/5)^4 + 1)/14)$ and the other pairs are for the quadratic and quartic twists. The polynomials for the orders are all irreducible but have cofactors 2, 2, 32, and 20.

	(t_e, y_e) s.t. $q = (t_e^2 + 4y_e^2)/4$	$q_e = q + 1 - t_e$	family
t, y	$(31(x/5)^4 + 1)/7, (-17(x/5)^4 - 1)/14$	$(x^8 - 386x^4 + 5^5 \cdot 17)/61250$	(yes, 2)
$-t, y$	$(-31(x/5)^4 - 1)/7, (-17(x/5)^4 - 1)/14$	$(x^8 + 482x^4 + 5^4 \cdot 113)/61250$	(yes, 2)
$2y, t/2$	$(-17(x/5)^4 - 1)/7, (31(x/5)^4 + 1)/14$	$(x^8 + 286x^4 + 5^4 \cdot 113)/61250$	(yes, 32)
$-2y, t/2$	$(17(x/5)^4 + 1)/7, (31(x/5)^4 + 1)/14$	$(x^8 - 190x^4 + 5^5 \cdot 17)/61250$	(yes, 20)

Table 10: Seeds u of Hamming weight ≤ 8 such that the KSS16 curve E/\mathbb{F}_p admits an embedded curve E_1/\mathbb{F}_q of j -invariant 1728 and order $h \cdot s$ with s prime and even h tiny. All curves have $-D = -4$.

seed u	L	equation $E_{\text{KSS}}/\mathbb{F}_p$	p (bits)	q (bits)	embedded curve equation E_1/\mathbb{F}_q	h
$q = (u^8 - 386u^4 + 5^5 \cdot 17)/61250$ (row 1 in Table 9)						
0x37effef25 = 45 mod 70 $2^{34} - 2^{31} - 2^{24} - 2^{12} - 2^8 + 2^5 + 2^2 + 1$	5	$y^2 = x^3 + 25x$	329	255	$y^2 = x^3 + 3x$	
0x36007bf3f = 25 mod 70 $2^{34} - 2^{31} - 2^{29} + 2^{19} - 2^{14} - 2^8 + 2^6 - 1$	4	$y^2 = x^3 + 11x$	328	255	$y^2 = x^3 + 3x$	
$(u^8 - 190u^4 + 5^5 \cdot 17)/61250$ (row 4 in Table 9)						
0x3dee0008d = 25 mod 70 $2^{34} - 2^{29} - 2^{24} - 2^{21} + 2^7 + 2^4 - 2^2 + 1$	6	$y^2 = x^3 + 2x$	330	256	$y^2 = x^3 + 3x$	

4.4 Better seeds of embedded curves with BLS12

In [SEH24], Sanso and El Housni propose the seed 0xb504f33499580000 that generates a BLS12-380 curve and a prime-order embedded curve. Alternatively we generated the seeds in Table 11 of Hamming weight up to 6 in signed binary representation.

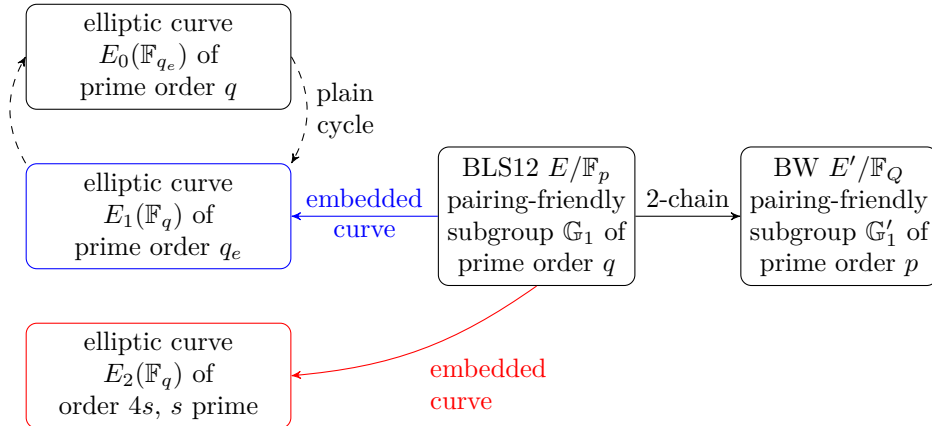


Figure 5: Plain embedded cycle and embedded curve above an inner BLS12 curve, and an outer curve forming a 2-chain.

- An elliptic curve embedded under Ed255-19. This curve is useful in order to prove arithmetic circuit on the curve Ed255-19. Moreover, it is of prime order, enabling recursive proofs as in the previous curve.
- An embedded curve above BLS12-381 similar to [MSZ24]. The endomorphism of this curve is not as fast as in the case of Bandersnatch, but it helps us understand the sparsity of Bandersnatch (high security and small discriminant). This second curve fulfills the security requirements of Bandersnatch, for another (larger) discriminant.

Although these curves open new directions for recursive proofs, we mention that in the context of ZK proofs, high 2-adicity is usually required for the proof generation. We did not investigate this property as it probably requires a new record of computation for the Hilbert class polynomial.

In the second part, we consider families of embedded curves. Instead of considering a field \mathbb{F}_q , we consider $\mathbb{Q}[X]/(q(x))$ and our results let us generate embedded curves for a generic seed u (so that \mathbb{F}_q can be defined using $q(u)$). In particular, we obtain families of prime-order embedded curves of discriminant 3 that form a family of plain cycles above BLS12, KSS16 and KSS18 curves.

Acknowledgments. A. G. involvement in this work follows a discussion with Carla Ràfols at the IMACC'23 conference at Royal Holloway in December 2023. A. G. would like to thank Carla Ràfols, Anca Nitulescu, Javier Silva, and Nikitas Paslis for the fruitful discussions and many shared references. The authors thank Youssef El Housni and Antonio Sanso for discussions on embedded curves.

References

- [AHG23] Diego F. Aranha, Youssef El Housni, and Aurore Guillevic. A survey of elliptic curves for proof systems. *DCC*, 91(11):3333–3378, 2023. doi:10.1007/s10623-022-01135-y.
- [AM93] A. O. L. Atkin and F. Morain. Elliptic curves and primality proving. *Mathematics of Computation*, 61(203):29–68, July 1993. doi:10.1090/S0025-5718-1993-1199989-X.

- [Azt] Aztec Protocol. Aztec connect specifications. <https://aztecprotocol.github.io/aztec-connect/primitives.html>.
- [BCKL22] Eli Ben-Sasson, Dan Carmon, Swastik Kopparty, and David Levit. Scalable and transparent proofs over all large fields, via elliptic curves - (ECFFT part II). In Eike Kiltz and Vinod Vaikuntanathan, editors, *TCC 2022, Part I*, volume 13747 of *LNCS*, pages 467–496. Springer, Cham, November 2022. doi:10.1007/978-3-031-22318-1_17.
- [BFHVMV84] Ian F. Blake, Ryoh Fuji-Hara, Ronald C. Mullin, and Scott A. Vanstone. Computing logarithms in finite fields of characteristic two. *SIAM Journal on Algebraic Discrete Methods*, 5(2):276–285, 1984. doi:10.1137/0605029.
- [BGH19] Sean Bowe, Jack Grigg, and Daira Hopwood. Halo: Recursive proof composition without a trusted setup. Cryptology ePrint Archive, Report 2019/1021, 2019. URL: <https://eprint.iacr.org/2019/1021>.
- [BL24] Daniel J. Bernstein and Tanja Lange. Safe curves for elliptic-curve cryptography. Cryptology ePrint Archive, Report 2024/1265, 2024. URL: <https://eprint.iacr.org/2024/1265>.
- [BMV84] Ian F. Blake, Ronald C. Mullin, and Scott A. Vanstone. Computing logarithms in $\text{GF}(2^n)$. In G. R. Blakley and David Chaum, editors, *CRYPTO'84*, volume 196 of *LNCS*, pages 73–82. Springer, Berlin, Heidelberg, August 1984. doi:10.1007/3-540-39568-7_8.
- [CBBZ22] Binyi Chen, Benedikt Bünz, Dan Boneh, and Zhenfei Zhang. HyperPlonk: Plonk with linear-time prover and high-degree custom gates. Cryptology ePrint Archive, Report 2022/1355, 2022. URL: <https://eprint.iacr.org/2022/1355>.
- [CK24] Craig Costello and Gaurish Korpak. Lollipops of pairing-friendly elliptic curves for composition of proof systems. *ePrint 2024/1627*, 10 2024.
- [CS18] Craig Costello and Benjamin Smith. Montgomery curves and their arithmetic - the case of large characteristic fields. *Journal of Cryptographic Engineering*, 8(3):227–240, September 2018. doi:10.1007/s13389-017-0157-6.
- [DLZZ23] Yu Dai, Kaizhan Lin, Chang-An Zhao, and Zijian Zhou. Fast subgroup membership testings for \mathbb{G}_1 , \mathbb{G}_2 and \mathbb{G}_T on pairing-friendly curves. *DCC*, 91(10):3141–3166, 2023. doi:10.1007/s10623-023-01223-7.
- [Feo17] Luca De Feo. Mathematics of isogeny based cryptography. *CoRR*, abs/1711.04062, 2017. URL: <http://arxiv.org/abs/1711.04062>, [arXiv:1711.04062](https://arxiv.org/abs/1711.04062).
- [GLV01] Robert P. Gallant, Robert J. Lambert, and Scott A. Vanstone. Faster point multiplication on elliptic curves with efficient endomorphisms. In Joe Kilian, editor, *CRYPTO 2001*, volume 2139 of *LNCS*, pages 190–200. Springer, Berlin, Heidelberg, August 2001. doi:10.1007/3-540-44647-8_11.
- [Gro16] Jens Groth. On the size of pairing-based non-interactive arguments. In Marc Fischlin and Jean-Sébastien Coron, editors, *EUROCRYPT 2016, Part II*, volume 9666 of *LNCS*, pages 305–326. Springer, Berlin, Heidelberg, May 2016. doi:10.1007/978-3-662-49896-5_11.

- [GW20] Ariel Gabizon and Zachary J. Williamson. plookup: A simplified polynomial protocol for lookup tables. Cryptology ePrint Archive, Report 2020/315, 2020. URL: <https://eprint.iacr.org/2020/315>.
- [GWC19] Ariel Gabizon, Zachary J. Williamson, and Oana Ciobotaru. PLONK: Permutations over Lagrange-bases for oecumenical noninteractive arguments of knowledge. Cryptology ePrint Archive, Report 2019/953, 2019. URL: <https://eprint.iacr.org/2019/953>.
- [Ham15] Mike Hamburg. Decaf: Eliminating cofactors through point compression. In Rosario Gennaro and Matthew J. B. Robshaw, editors, *CRYPTO 2015, Part I*, volume 9215 of *LNCS*, pages 705–723. Springer, Berlin, Heidelberg, August 2015. doi:10.1007/978-3-662-47989-6_34.
- [Hop17] Daira Hopwood. Jubjub supporting evidence. <https://github.com/daira/jubjub>, 2017.
- [Hop20a] Daira Hopwood. Halo optimizations and constructing graphs of elliptic curves. <https://github.com/daira/halographs/>, 2020.
- [Hop20b] Daira Hopwood. The pasta curves for halo 2 and beyond. <https://electriccoin.co/blog/the-pasta-curves-for-halo-2-and-beyond/>, 2020.
- [Hop21] Daira Hopwood. Pluto-eris hybrid cycle of elliptic curves, 2021. <https://github.com/daira/pluto-eris>.
- [KST22] Abhiram Kothapalli, Srinath Setty, and Ioanna Tzialla. Nova: Recursive zero-knowledge arguments from folding schemes. In Yevgeniy Dodis and Thomas Shrimpton, editors, *CRYPTO 2022, Part IV*, volume 13510 of *LNCS*, pages 359–388. Springer, Cham, August 2022. doi:10.1007/978-3-031-15985-5_13.
- [KZG10] Aniket Kate, Gregory M. Zaverucha, and Ian Goldberg. Constant-size commitments to polynomials and their applications. In Masayuki Abe, editor, *ASIACRYPT 2010*, volume 6477 of *LNCS*, pages 177–194. Springer, Berlin, Heidelberg, December 2010. doi:10.1007/978-3-642-17373-8_11.
- [Mor06] François Morain. Implementing the asymptotically fast version of the elliptic curve primality proving algorithm. *Mathematics of Computation*, 76(257):493–505, 2006. hal-00004136. doi:10.1090/S0025-5718-06-01890-4.
- [MSZ24] Simon Masson, Antonio Sanso, and Zhenfei Zhang. Bandersnatch: a fast elliptic curve built over the BLS12-381 scalar field. *Designs, Codes and Cryptography*, 2024. <https://doi.org/10.1007/s10623-024-01472-0>. doi:10.1007/s10623-024-01472-0.
- [NPR24] Anca Nitulescu, Nikitas Paslis, and Carla Ràfols. FLIP-and-prove R1CS. *ePrint:2024/1364*, 2024.
- [SEH24] Antonio Sanso and Youssef El Housni. Families of prime-order endomorphism-equipped embedded curves on pairing-friendly curves. <https://link.springer.com/article/10.1007/s00145-024-09514-5>, 2024. doi:10.1007/s00145-024-09514-5.
- [Sma99] Nigel P. Smart. The discrete logarithm problem on elliptic curves of trace one. *Journal of Cryptology*, 12(3):193–196, June 1999. doi:10.1007/s001459900052.

-
- [Smi15] Benjamin Smith. Easy scalar decompositions for efficient scalar multiplication on elliptic curves and genus 2 Jacobians. *Contemporary mathematics*, 637:15, May 2015. HAL:00874925. doi:10.1090/conm/637/12753.
- [Sut11] Andrew V. Sutherland. Computing Hilbert class polynomials with the chinese remainder theorem. *Mathematics of Computation*, 80(273):501–538, 2011. <https://arxiv.org/abs/0903.2785>. doi:10.1090/S0025-5718-2010-02373-7.
- [Val21] Henry De Valence. The ristretto group. <https://ristretto.group>, 2021.
- [ZCa21] ZCash. What is jubjub? <https://z.cash/technology/jubjub/>, 2021.