Embedded Elliptic Curves and Embedded Families for SNARK-Friendly Elliptic Curves

Aurore Guillevic[1] and Simon Masson[2]

[1] Univ Rennes, Inria, CNRS, IRISA, Rennes, France
[2] ZKNOX, Paris, France

Abstract. In 2021, Masson, Sanso, and Zhang introduced the Bandersnatch curve associated to the BLS12-381 pairing-friendly curve, an elliptic curve designed for zero-knowledge proofs requiring circuits with a curve arithmetic. This type of curve is useful for privacy-preserving protocols, and more generally for succinct validity proof using pairing-based SNARKs. An *embedded* curve is defined over a field whose order is the group order of its associated curve. In this way, the pairing-friendly curve is used to express a zero-knowledge proof (such as a SNARK) of a statement taking place on the embedded curve. Contrary to the previous embedded curves (such as CØCØ, JubJub), Bandersnatch was built with the complex multiplication (CM) method, in order to ensure a very small discriminant ($D = -8$, of small magnitude), and thus efficient scalar multiplication thanks to the GLV technique. The algorithm provided by Masson, Sanso, and Zhang for searching this type of curves requires computation of Hilbert class polynomials, making the search of curve slow. It was not known whether Bandersnatch was an exceptional curve or whether comparable curves exist, of larger discriminant. This paper highlights the technicalities of the CM method already in use in the 90s to generate curve parameters of chosen order. This old technique allows revisiting the curve search of Bandersnatch, providing a dramatic speed-up improvement. This paper presents two algorithms: one to generate embedded elliptic curves of SNARK-friendly curves, with a variable discriminant D; a second to generate families (parameterized by polynomials) with a fixed discriminant D. When $D = -3 \bmod 4$ ($D < 0$), it is possible to obtain a prime-order curve, and define a cycle. To illustrate this, we apply the technique first to generate more embedded elliptic curves like Bandersnatch with BLS12-381, such as a curve of discriminant $D = -6673027$, defining a plain twist-secure cycle. We also comment on the scarcity of Bandersnatch-like CM curves, and show that with this generic algorithm, it is only a question of core-hours to find them. Second, we show the link between a paper of Smith in 2015 and the work of Dai, Lin, Zhao, and Zhou in 2023, obtaining prime-order parameterized families of embedded curves of fixed discriminant, such as $D = -3$ for BLS and KSS18 curves. With KSS16 curves, the discriminant $D = -4$ is also possible (the curve has an even order). The technique can work with any KSS, GG (Gasnier–Guillevic), or other fixed-discriminant parameterized family of pairing-friendly curves. This paper provides a more general point of view on embedded curves such as Bandersnatch, putting into perspective the works of Masson, Sanso, and Zhang, and Sanso and El Housni. The Python/SageMath scripts are available at https://gitlab.inria.fr/zk-curves/cm-embedded-curves/.

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

This article is the final version submitted by the authors to the IACR Communications in Cryptology journal on July 29, 2026. The version published by the IACR is available at doi:10.62056/a33zl83y6.

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: 2026-08-04.

1 Introduction

Zero-knowledge proofs have been deployed in a wide range of privacy-preserving protocols, many of them rooted in constructions such as [BCG+14]. In these systems, the statements to be proven typically involve cryptographic computations defined over various finite fields. Among these, verifying a digital signature inside a ZK proof (relative to a given public key) remains one of the most circuit-intensive tasks, and is often intertwined with commitment schemes and Merkle-tree path proofs. In practice, these components rely on hash functions such as Poseidon [GKR+21], themselves defined over finite fields. The resulting mismatch between the first field where the cryptographic statement is defined (for example, a signature scheme or hash-based commitment) and the second field used by the proof system often becomes the primary performance bottleneck. Modern ZK constructions, such as Groth16 [Gro16] and its variants, require circuits to be expressed over the scalar field of a pairing-friendly curve, while other approaches rely on hash functions defined over smaller fields. In curve-based proof systems, this issue is typically mitigated by carefully selecting curves whose statement field (for instance, the base field of the signature curve) coincides with the proof field used for the circuit arithmetic. This has been adopted for example in [BCG+14] where [Hop20b] instantiates a cycle of elliptic curves having these properties. Other constructions are based on BN254 or BLS12-381, pairing-friendly curves designed for SNARKS. For these protocols, the signature curve is selected in order to match with the proof field [ZCa21, MSZ24].

With the development of proof-of-knowledge systems, in particular SNARK (Succinct Non-interactive ARgument of Knowledge), finding new dedicated elliptic curves is needed. 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 (DL) problem [BGH19, Hop20b],
- 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 (ZK) SNARKs are built using arithmetic circuits corresponding to operations involving a secret value. In other words, ZK proofs demonstrate the knowledge of a secret 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 on 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, CBBZ23]. The design was then adapted for the DL polynomial commitment [BGH19]. In most of the constructions, the elliptic curve equipped with a polynomial commitment is defined over a prime field $\mathbb{F}_p$, named *base field* or *field of definition*, and has a cryptographically secure prime subgroup of size q, also called *scalar field*. The arithmetic operations such as point addition and doubling on the curve take place in $\mathbb{F}_p$, while q refers to the order of the points.

ZK proofs built using circuits compute proofs of a set of arithmetic operations in a finite field, that are derived from a statement. 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 that produces the *proof*. In a cryptographic setting, it is interesting to build proofs (of statements) 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 a *statement curve*. Thus, the base field of this elliptic curve needs to be $\mathbb{F}_q$, the scalar field of the proof curve. The naive idea of finding an elliptic curve with identical base field and scalar field is not cryptographically secure [Sma99]. These curves are called anomalous and must be avoided. Hence, a distinct curve for the statement, named embedded curve, is required, whose base field matches 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.

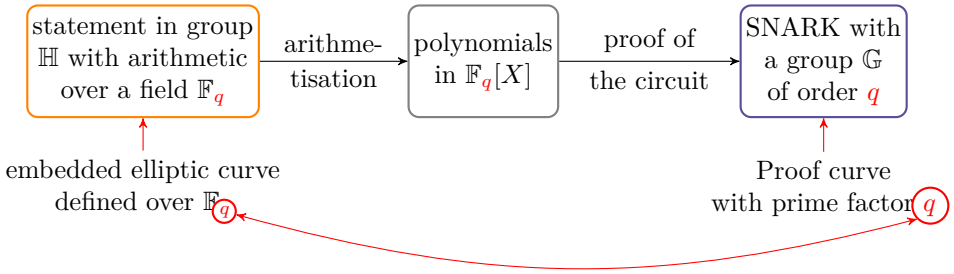
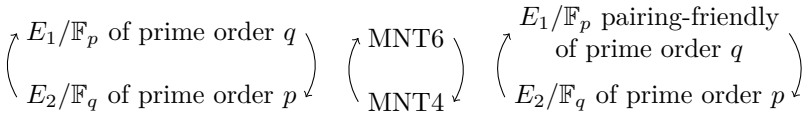


Figure 1: Relation between base field of the embedded curve and scalar field of the proof curve (adapted from [AEG23, Figure 3]).

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 $\mathbb{F}_p$ (Figure 2(b)). This construction, together with a bilinear pairing on each curve, is possible using the Miyaji–Nakabayashi–Takano (MNT) family of curves, but due to the Kim–Barbulescu variants [KB16] of the number field sieve algorithm (NFS) on the discrete logarithm problem (DL) over the pairing output fields, the size of the primes p and q are very large, making the computations very slow. Recent works [CCN24, CK26] 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 a *plain* 2-*cycle* as a cycle of two curves cryptographically secure, without pairing (Figure 2(a)). A plain 2-cycle was originally designed for HALO2 [BGH19, Hop20b]. 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, Figure 2(c) ([Azt] for the BN-254–Grumpkin half-cycle (also named hybrid cycle), [Hop21] for 446-bit Pluto-Eris).



(a) Plain cycle (b) MNT cycle (c) Half cycle

Figure 2: The three known types of cycles of curves.

Contributions. In this paper, we look for embedded curves with different goals. First, we recall a former algorithm from the 90s, applied to finding embedded curves for a fixed elliptic curve. The curves found can be of prime order or with a prescribed cofactor. The strategy iterates over a large set of discriminant values until a curve is found. Then, we apply this algorithm in order to find a prime-order embedded curve for the standardized `ed25519`, hence forming a plain 2-cycle before `ed25519`. We also search for prime-order embedded curves for the pairing-friendly curve BLS12-381. This produces plain 2-cycles, where one of the two curves is a statement curve (embedded) for BLS12-381. We explain the scarcity of such small-discriminant embedded curves and explain that although Bandersnatch is indeed a lucky curve (secure, of very small discriminant *and* twist-secure), it is possible to generate curves with similar properties, considering a larger discriminant. We state a formula from [Mor06] to estimate the number of enumerated discriminant values before such a curve is found. Furthermore, we consider families of embedded curves for a fixed discriminant, generalizing the idea of [SEH24]. In particular, Sanso and El Housni method worked for BLS curves only. Mixing [DLZZ23] and [Smi15], we extend it to all families of curves, in particular KSS and GG curves.

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 expose the 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 for fixed discriminant in Section 4, and provide examples with $D = -3, -4, -20$.

2 Preliminaries

2.1 Notations

Elliptic curves. In this paper, $E\colon y^2 = x^3 + ax + b$ 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 at least 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 $t \neq 0 \bmod q$. 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 $4q = t^2 + dy^2$. In Section 3.2, we will be interested in solving this equation for integers t, y, a prime q and $d > 0$. This is closely related to the theory of imaginary quadratic fields.

We remind the reader that the term embedded curve does not refer to a morphism between two curves, but rather to the fact that the base field of one curve is the scalar field of another.

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 denoted D, where $D = -4d$ if $d = 1, 2 \bmod 4$; $D = -d$ if $d = 3 \bmod 4$. 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:

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

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

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

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

Every order is a suborder of the maximal order. In this paper, to each elliptic curve we associate its 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. 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,

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

2.2 The CM method

The theory of complex multiplication (CM) is related to Deuring [Deu58] for elliptic curves, generalized by Shimura and Taniyama [ST61] for Abelian varieties. We only need a small part of the theory, developed in the 80s and 90s for curves over finite fields to be used in cryptography, at a time when point-counting algorithms were still being improved and required significant computing power. With the development of the Schoof, Elkies, and Atkin algorithm (SEA) for point counting, the Complex Multiplication method 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) 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 $E\colon y^2 = x^3 + \frac{3j}{1728-j}x + \frac{2j}{1728-j}$. 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 $H_D(X)$ mod q are the j-invariants of the elliptic curves over $\mathbb{F}_q$ having discriminant D. In practice, this polynomial can be computed directly modulo p using [Sut11], and the largest computation was done for a discriminant with a dozen of digits. PARI-GP [The23] and Enge's library [Eng09] also implement class polynomial computations.

Remark 1 (Small discriminant 2-cycles). When an elliptic curve $E_1/\mathbb{F}_q$ is of prime order p, there always exists another elliptic curve $E_2/\mathbb{F}_p$ of order q, i.e. a 2-cycle between E_1 and E_2. This classical result comes from the CM method. Let t be the trace of $E_1/\mathbb{F}_q$ so that $q + 1 - t = p$. Let y be an integer such that $4q = t^2 + dy^2$ where d is positive square-free. Writing $p + 1 - (2 - t) = q$, then $4p = (t - 2)^2 + dy^2$ and we obtain using the CM method another curve defined over $\mathbb{F}_p$ of trace $2 - t$ satisfying the Hasse bound $|2 - t| < 2\sqrt{p}$. As long as d is small enough, we can compute the curve coefficients using the Hilbert class polynomial.

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 $D < 0$, K, $\mathcal{O}_K$ as in Section 2.1. Let q be a rational prime. The norm equation $q = N_{K/\mathbb{Q}}(\pi)$ has an algebraic integer solution π in $\mathcal{O}_K$ if, and only if, the ideal (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 $\mathbb{Z}$, $D < 0$.*

Theorem 1 ([AM93, Theorem 3.2])**.** *Let $D < 0$, K, $\mathcal{O}_K$ 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 the ideal (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 rephrase 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 $t^2 + dy^2 = 4q$.*

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 $D \not\equiv -3 \bmod 4$, then t must be even. In consequence, the order of the curve must be even. Reciprocally, the curve will have prime order only if $D \equiv -3 \bmod 4$.

Remark 3. If (q) splits as two non-principal ideals $\mathfrak{q}, \tau(\mathfrak{q})$, solving the norm equation will fail, because $\mathfrak{q}$ does not have a unique generator of norm exactly q. But $\mathfrak{q}^h$ is principal, where h is the class number of K. It has norm q^h. It is possible to find a curve defined over $\mathbb{F}_{q^h}$ instead of $\mathbb{F}_q$, but this is not what we are looking for in this paper.

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 efficient implementations, k is chosen to be a product of a power of 2 and a power of 3, for example $k = 12, 24$. In this work, we consider several families of pairing-friendly curves [FST10]: Barreto–Naehrig (BN), Barreto–Lynn–Scott (BLS), Kachisa–Schaefer–Scott (KSS) whose parameter sets are given in Tables 1 and 2, and Scott–Guillevic [SG18], Gasnier–Guillevic [GG25]. Popular seeds are $x_0 =$ `0x44e992b44a6909f1` to generate a BN254 curve, $x_0 =$ `-0xd201000000010000` for BLS12-381, $x_0 =$ `0x8508c00000000001` for BLS12-377. The number after the family name in the label refers to the bitsize of the prime $p(x_0)$. Evaluating the polynomials $p(x)$, $t(x)$ and $q(x)$ at the seed x_0, 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 and selected BLS curve parameters.

	BN $k = 12$	BLS $k = 12$	BLS $k = 24$
D	-3	-3	-3
$q(x)$	$36x^4 + 36x^3 + 18x^2 + 6x + 1$	$\Phi_{12}(x) = x^4 - x^2 + 1$	$\Phi_{24}(x) = x^8 - x^4 + 1$
$t(x)$	$6x^2 + 1$	$x + 1$	$x + 1$
$p(x)$	$36x^4 + 36x^3 + 24x^2 + 6x + 1$	$(x^4 - x^2 + 1)(x - 1)^2/3 + x$	$(x^8 - x^4 + 1)(x - 1)^2/3 + x$
$c(x)$	1	$(x - 1)^2/3$	$(x - 1)^2/3$
$y(x)$	$6x^2 + 4x + 1$	$(2x^2 - 1)(x - 1)/3$	$(2x^4 - 1)(x - 1)/3$
x	any	1 mod 3	1 mod 3

Table 2: Selected KSS curve parameters.

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

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 statement curve compatible with the BLS12 curve and with a fast scalar multiplication. The base field of the new statement curve, called Bandersnatch, is fixed to be the scalar field of BLS12-381. The authors obtain an embedded curve using the CM method of Section 2.2. In [MSZ24, Algorithm 1], the search iterates on various discriminants and computes a Hilbert class polynomial in order to compute the order of

the curves. The authors are able to derive a curve embedded with BLS12-381, with almost optimal properties (very small discriminant, almost prime order, twist-secure).

Parameters. The Bandersnatch curve is defined over $\mathbb{F}_q$ where q is the 255-bit prime factor of the order of BLS12-381. Its j-invariant is 8000, and [MSZ24] provides a representation of the curve in Weierstrass model : $y^2 = x^3 - 3763200000x - 78675968000000$. In practice, other representations are usually preferred for a more efficient elliptic curve group law [CS18]. For Bandersnatch, we suggest the isomorphic curve $y^2 = x^3 + 5x + b$ where b is `0x6a8d275fe8126c2c0022c15e1f181e282fb81761827fdf4ccdf7834600226d91`.

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 scalar multiplications faster than on Jubjub [ZCa21], also defined over the scalar field of BLS12-381, but with a larger discriminant, as it was obtained with the Schoof–Elkies–Atkin (SEA) point counting.

Security. The curve has order $4r$, where r is prime. This allows a Montgomery model, avoiding 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].

Uncommon curve. 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 $q = q(x)$ of Table 1). 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 becomes cumbersome as long as $|D|$ increases. We restate 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 values of discriminant.

3 Embedded curves of variable discriminant

3.1 The problem

In this section, we consider a fixed elliptic curve E defined over a prime field, and we look either for an embedded curve (E is the proof curve), of prime order (Section 3.1.1) or of composite order (Section 3.1.3), or for a proof curve (E is the embedded curve, Section 3.1.2). We target specific properties in terms of security and efficiency:

- In terms of security, we look for curves of prime order or with a large enough prime-order subgroup (such as 4 or 8 times a prime). We also consider the twist-security of the curves, i.e. we have a similar approach on the quadratic twist of the considered curves. This requirement prevents subgroup attacks on the discrete logarithm problem of the quadratic twist of the curve.

- In terms of efficiency, we target a small-enough discriminant (D of magnitude a few bits) so that the GLV technique applies. Alternatively, we consider curves of prime

order and medium-sized discriminant, that is, larger than the limit for GLV usage, but manageable for the Hilbert class polynomial computation, allowing finding the coefficients of the 2-cycle curves.

3.1.1 Prime-order embedded curve for a given proof 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.

$$\begin{array}{c} E_1/\mathbb{F}_q \text{ (statement)} \\ \text{of prime order } r \\ \text{embedded curve} \end{array} \longrightarrow \begin{array}{c} E/\mathbb{F}_p \text{ (proof)} \\ \text{of order } h \cdot q \text{ with prime } q \\ \text{reference curve} \end{array}$$

3.1.2 Prime-order curve to serve as the proof curve of 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, and used for a statement. We look for a proof curve E, defined over a prime field $\mathbb{F}_p$, whose order q matches the base field of the fixed curve. Since E is of prime order, it can form a 2-cycle with a third curve as we will see in the example of Section 3.3.2, where we consider Ed25519 for E_1.

$$\begin{array}{c} E_1/\mathbb{F}_q \text{ (statement)} \\ \text{of order } 4r \text{ with prime } r \\ \text{reference curve} \end{array} \longrightarrow \begin{array}{c} E/\mathbb{F}_p \\ \text{of prime order } q \\ \text{proof curve with } E_1 \end{array}$$

3.1.3 Composite-order embedded curve with a composite-order proof curve

When the reference curve E is defined over $\mathbb{F}_p$ and has a composite order, it cannot constitute a cycle [CCW19, Proposition 7.2]. The embedded curve does not need to be of prime order anymore, it will not define a cycle with E in any way. In this context, we look for an embedded curve E_1 of order $4r$ where r is prime, for a Montgomery representation. Bandersnatch falls in this setting. We provide in Section 3.3.3 an embedded curve with BLS12-381, with similar security features as Bandersnatch, but a larger discriminant. This lets us understand better the scarcity of Bandersnatch.

$$\begin{array}{c} E_1/\mathbb{F}_q \text{ (statement)} \\ \text{of order } 4r \text{ with prime } r \\ \text{embedded curve} \end{array} \longrightarrow \begin{array}{c} E/\mathbb{F}_p \text{ (proof)} \\ \text{of order } h \cdot q \text{ with prime } q \\ \text{reference curve} \end{array}$$

In order to generate these curves, 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 CM 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 90s for the ECPP algorithm [AM93, Section 8.4.2]. We solve this equation using Theorem 2. We find the corresponding element π using 2-dim lattice reduction (Lagrange–Gauss, or

Cornacchia algorithm). Indeed, the algebraic integer of prime norm q is the generator of the ideal of norm q, and is the shortest non-zero element (in terms of coefficient size and algebraic norm) of that ideal. A generator may not exist when the ring $\mathcal{O}_K$ is not principal. Using Cohen's notation [Coh93, Lemma 4.7.9 p. 192], this ideal can be represented with two generators $I = \mathbb{Z}\langle q, X - s\rangle$, where $X - s \in \mathbb{F}_q[X]$ is a factor of $X^2 + d \bmod q$ when $d \not\equiv 3 \bmod 4$, of $X^2 - X + \frac{d+1}{4} \bmod q$ when $d \equiv 3 \bmod 4$. We look for the shortest element of I via reducing the lattice defined by $((q, 0), (-s, 1))$. When an element of norm q exists, we deduce t, y from Theorem 2. Finally, we use the Hilbert class polynomial computation modulo q in order to get the j-invariant and curve coefficients. The method is summarized in Algorithm 1. If $-d$ is not a square modulo q, a supersingular curve exists.

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

Input: prime integer q, minimum and maximum values of $d > 0$
Output: A list of traces, discriminants, and j of embedded elliptic curves for $\mathbb{F}_q$

$\mathcal{L} \leftarrow \{\}$; $\mathcal{L}' \leftarrow \{\}$
for d *from* $d_{\min}$ **to** $d_{\max}$ **do**
  if d *is square-free* **then**
    if $-d$ *is a square modulo* q **then**
      $s \leftarrow \begin{cases} \sqrt{-d} \bmod q & d \not\equiv 3 \bmod 4 \\ \frac{1+\sqrt{-d}}{2} \bmod q & d \equiv 3 \bmod 4 \end{cases}$
      lift s in $\mathbb{Z}$
      $\pi \leftarrow a_0 + a_1 X$ the shortest non-zero element of the lattice $\mathbb{Z}\langle q, X - s\rangle$
      if π *has norm* q **then**
        $(D, t, y) \leftarrow \begin{cases} (-d, 2a_0, a_1) \textbf{ if } d \equiv 3 \bmod 4 \\ (-4d, 2a_0 + a_1, a_1) \textbf{ otherwise} \end{cases}$
        $\mathcal{L} \leftarrow \mathcal{L} \cup \{(D, t, y)\}$
// Post-processing according to additional properties (Section 3.1)
for (D, t, y) *in* $\mathcal{L}$ **do**
  $n \leftarrow q + 1 - t$; $n' \leftarrow q + 1 + t$
  if *the order and twist order* n, n' *satisfy additional properties* **then**
    Compute a root j of $H_D(X) \bmod q$
    $\mathcal{L}' \leftarrow \mathcal{L}' \cup \{(D, t, y, j)\}$
return $\mathcal{L}'$

Algorithm 1 is not conceptually different from [MSZ24, Algorithm 1]. Its main advantage is delaying outside of the loops the slow and resource-consuming Hilbert class polynomial computation. We can enumerate many more and much larger discriminants. The Lagrange–Gauss reduction algorithm in dimension 2 is very efficient and solves for $a_0 + a_1 X$ and the curve order very quickly. Additional conditions on the obtained order (for instance on small co-factors) can be included in the algorithm in order to refine the search for specific curves. Once the best curve is found, fulfilling all constraints, we finally compute one class polynomial modulo q in order to get the curve j-invariant and coefficients. This computation is possible only up to discriminants of magnitude a dozen of digits [Sut11]. As an example, in [BCTV14], the MNT-298 and MNT-753 curves have a 15-digit discriminant $D = -614144978799019$ and $D = -331787862733683$, resp. 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. We also remark that Costello and Korpal [CK26] mention they tweaked the algorithm of [MSZ24] to compute embedded curves (the end of the stitch part in the

lollipop construction, denoted $\boldsymbol{E}_Y/\mathbb{F}_r$). This technique will increase the range of scanned discriminants and speed-up the search.

3.3 Practical examples

In this section, we consider three use-cases related to elliptic curves used in practice. First, we look for an embedded plain cycle with BLS12-381. Then, we consider a similar case with Ed25519. Finally, we look for an embedded curve with the same properties as Bandersnatch in order to understand better the chances to find such curves of small discriminant.

3.3.1 Embedded plain cycle with BLS12-381 as the proof curve

In this section, we look for a plain 2-cycle with BLS12-381, as depicted in Figure 3. From Remark 2, we need $D \equiv -3 \bmod 4$ in order to get a prime-order curve. From Remark 1, we know that a 2-cycle automatically exists. Nitulescu, Paslis, and Ràfols [NPR26] present a scheme for producing a proof with a pairing-based SNARK of a succession of proofs based on the folding technique, with NOVA. Our curve construction could be used as an alternative to a half-cycle for this usecase. The NOVA folding proofs take place on the embedded curve and its associated 2-cycle curve, whose base field is 255-bit long (sclar field of BLS12-381), instead of a half-cycle with fields of about 384 bits due to a BN curve. Finally the SNARK proof takes place on BLS12-381, where the pairing is more efficient than on a prime-order BN curve of about 384 bits.

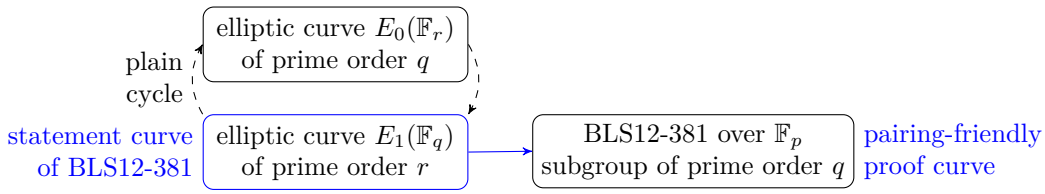


Figure 3: Plain cycle, one curve is a statement curve with the proof curve BLS12-381.

In order to get this curve, we consider various discriminants $|D|$ in the range $1 - 10^8$, with the condition $D \equiv -3 \bmod 4$. Using Algorithm 1, we expect to find a prime-order curve after $\ln(q) \approx 177$ valid curves obtained at Line 8 (Section 3.3.3). After a few hours of computation on one core of an Intel i7-1365U, we obtain the following parameters for a cycle of curves with BLS12-381. We enumerated the roots j of $H_D(X) \bmod q$ until a short Weierstrass curve equation $y^2 = x^3 - 3x + b_q$ 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 its quadratic twist order is 176-bit long. The parameters q, r, b_q and b_r are the following:

q : `0x73eda753299d7d483339d80809a1d80553bda402fffe5bfeffffffff00000001`,
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 a very large degree. This computation can be faster by decomposing the endomorphism as a chain of smaller degree isogenies. Still, this endomorphism would require a lot of operations, 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 a 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 discriminants (trying roughly 2^{32} times more discriminants congruent to -3 mod 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 resource-consuming. 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 Plain-curve cycle with Ed25519 for the statement

Another interesting application is related to the standardized Ed25519 curve. We use Algorithm 1 similarly 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 4. As in Section 3.3.1, we obtain by construction a 2-cycle because $E_1(\mathbb{F}_p) = 2^{255} - 19$ is prime.

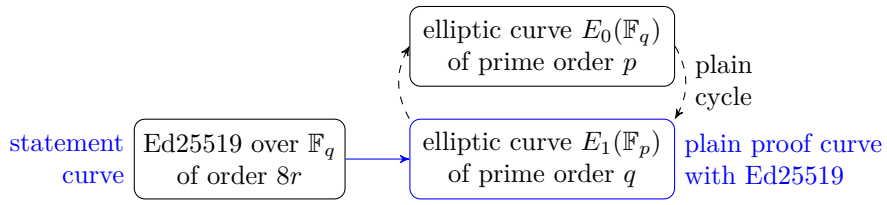


Figure 4: Plain cycle with Ed25519 as the statement curve.

We scan the square-free $d \equiv 3$ mod 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 [The23, Eng09]. 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 : `0x7fffffffffffffffffffffffffffffff34a2208109393ca351aa6d362f601a5f`,
q : `0x7fffffffffffffffffffffffffffffffffffffffffffffffffffffffffffffed`,
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 ring 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.

Remark 4. To the best of our knowledge, the question of finding a proof curve with Ed25519 was raised by Poelstra in 2018 on the Modercrypto/curves mailing list[1]. Finding a cycle of curves with Ed25519 as the statement curve has been investigated by Monero for their privacy-preserving SNARKs. The Helios-Selene curve cycle[2] has these properties, and the security of this cycle of curves have been assessed in [BS25].

3.3.3 Scarcity of subgroup-secure twist-secure embedded curves

In this section, we investigate the scarcity of embedded curves like 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 its discriminant has a very small magnitude $|D|$, 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 discriminant of larger magnitude). 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 small discriminant $D = -40$ for a chosen base field over the Mersenne prime $2^{127} - 1$.

Bandersnatch has a fast endomorphism $\phi = [\sqrt{-2}]$ 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 x_0 *only* when D is a square in $\mathbb{Q}[X]/(q(X))$. In the case of Bandersnatch, -2 is not a square modulo $q(x)$, meaning that there is no *hidden* structure such as a univariate polynomial representation.

While Bandersnatch was found for a small discriminant, it has also good security features: the curve order has a subgroup of 253-bit order, and its quadratic twist has also a subgroup of 244-bit order. This provides a good subgroup and twist security. Meeting these two conditions happens with probability roughly $1/\log(q)^2 \approx 2^{-14.9}$, which is quite small (Line 13 in Algorithm 1). Note that this rough probability estimation does not take into account the fact that discriminants might not lead to a solution of the norm equation in Theorem 2. Estimating this probability depends on the discriminant, more precisely on the class number $h(-D)$ of $\mathcal{O}_K$. We refer to [AM93] for further details. The intuition is to observe that a prime ideal $\mathfrak{q}$ above q in $\mathcal{O}_K$ is usually not principal, but $\mathfrak{q}^{h(D)}$ is principal. The probability that the condition of Line 7 is satisfied is about $1/h(D)$, and the probability that $-d$ is a square modulo q is about $1/2$ (Line 4). Finally, the probability that a prime q splits in the Hilbert class field K_H of discriminant D is $1/(2h(D))$ (FastECPP, [Mor06, Section 2 and 3.7]). To increase the probability, the strategy of ECPP and FastECPP 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 (a large power of 2 divides its order minus one), 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

[1] https://moderncrypto.org/mail-archive/curves/2018/000992.html
[2] https://gist.github.com/tevador/4524c2092178df08996487d4e272b096

for the ECPP method, that does work well in practice, so we again stress that finding CM embedded curves is not a surprise.

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 $D = -4 \cdot 1030258$. Note that $\ln 1030258 = 2^{19.9}$, the difference to $(\ln q)^2 = 2^{14.9}$ comes from the factor $1/(2h(D_i))$ at each d_i, where the D_i are the negative fundamental discriminants of the enumerated $d_i \leq 1030258$. 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 : `0x73eda753299d7d483339d80809a1d80553bda402fffe5bfeffffffff00000001`,
r : `0x1cfb69d4ca675f520cce7602026876011d928688cbe65dba241d31a34658145b`,
b : `0x284504e3feb2feda870885babe7e63bb3a55276335d818514a9313120e044ae9`.

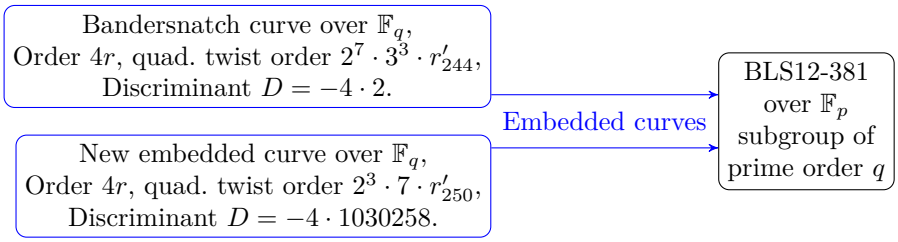


Figure 5: Two secure embedded curves for BLS12-381.

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 of fixed discriminant

4.1 Previous works

The usual pairing-friendly elliptic curves such as Barreto–Naehrig (BN) and Barreto–Lynn–Scott (BLS) curves have parameters in polynomial form (Table 1). This idea dates back to the BLS and Brezing–Weng constructions. Sanso and El Housni in [SEH24] aim at finding families of embedded curves, of small fixed discriminant D, whose parameters have a polynomial form. They present a technique to define families of embedded curves with BLS12 and BLS24. They observe that the scalar field of BLS12 curves $q(x) = x^4 - x^2 + 1$ can be written in the form $q(x) = (t'^2 + 3y'^2)/4$ where $t' = 2x^2 - 1$ and $y' = 1$, forming a parameterized family of embedded curves of trace $t'(x)$, discriminant -3, over a field of characteristic $q(x)$, for seeds $x \in \mathbb{Z}$. It also works with $q(x) = ((2x)^2 + 4(1 - x^2)^2)/4$. These two equations allow solving the norm equation of Section 2.3. Designing a pairing-friendly curve and a corresponding embedded curve boils down to finding a seed $x_0 \in \mathbb{Z}$ such that all the parameters, of the pairing-friendly curve and of the embedded curve, are relevant (integers, primes if required).

We rephrase Sanso and El Housni procedure as Algorithm 2. Someone familiar with the early techniques of pairing-friendly curve generation, named Cocks–Pinch and Brezing–Weng, can make a parallel: moving from integer curve parameters to curve parameters in a polynomial form. On one hand, the flexibility relies on the choice of the variable

discriminant D, on the other hand, it relies on the seed x_0, implying a new pairing-friendly curve tied to a new embedded curve.

The output of Algorithm 2 for BLS12 is Table 3 for embedded curves with j-invariant 0 ($D = -3$). Two families can produce prime-order embedded curves. For $D = -4$, Sanso and El Housni's procedure will output Table 4. curves with j-invariant 1728 cannot have prime order as they always have at least one point of order two and an even order.

Algorithm 2: Generating prime-order small-disc. embedded curves with BLS [SEH24]

Input: parameterized pairing-friendly curve order $q(x)$ that generates primes, discriminant $D < 0$ 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**
  $w(x) \leftarrow \sqrt{D} \bmod q(x)$
  $t_0(x), y_0(x) \leftarrow$ half-gcd$(w(x), q(x))$
  if $t_0(x)^2 - Dy_0(x)^2 = 4q(x)$ **then**
    for *$t'(x)$ in the set of traces of twisted curves with respect to $t_0(x)$* **do**
      $r(x) \leftarrow q(x) + 1 - t'(x)$; $y'(x) \leftarrow$ such that $t'^2(x) - Dy'^2(x) = 4q(x)$
      if *$r(x)$ is irreducible* **then**
        Append (t', y', r) to the list of families
    return the list of families
return $\perp$

Table 3: Embedded curve families above BLS12 with $q(x) = x^4 - x^2 + 1$ and $D = -3$. A first pair is $(t_0, y_0) = (2x^2 - 1, 1)$ and the other pairs are for the quadratic, cubic and sextic twists. The fourth one's order $q + 1 - t' = x^4 - 2x^2 + 4$ is not prime but can give three times a prime (when evaluated at a seed $x = 1 \bmod 3$).

(t', y') s.t. $q = (t'^2 + 3y'^2)/4$		$q + 1 - t'$	family
t_0, y_0	$2x^2 - 1, 1$	$x^4 - 3x^2 + 3$	yes
$-t_0, y_0$	$-2x^2 + 1, 1$	$(x^2 - x + 1)(x^2 + x + 1)$	no
$(t_0 + 3y_0)/2, (t_0 - y_0)/2$	$x^2 + 1, x^2 - 1$	$(x - 1)^2(x + 1)^2$	no
$(t_0 - 3y_0)/2, (t_0 + y_0)/2$	$x^2 - 2, x^2$	$x^4 - 2x^2 + 4$	(yes)
$-(t_0 - 3y_0)/2, (t_0 + y_0)/2$	$-x^2 + 2, x^2$	x^4	no
$-(t_0 + 3y_0)/2, (t_0 - y_0)/2$	$-x^2 - 1, x^2 - 1$	$x^4 + 3$	yes

Limitations. Two steps can fail in Algorithm 2. First, testing if D is a square in the number field $K = \mathbb{Q}(x)/(q(x))$. We note that it is a much stronger condition than asking for D being a square modulo a prime integer $q = q(x_0) \in \mathbb{N}$ for some seed $x_0 \in \mathbb{Z}$. For example, $D = -8$ is not a square in the 12-th cyclotomic field defined by $q(x) = \Phi_{12}(x) = x^4 - x^2 + 1$ however is it a square modulo $q_0 = q(x_0) \in \mathbb{N}$ the order of the scalar field of the BLS12-381 curve. For a fixed number field defined by an irreducible polynomial $q(x)$, only one or two fixed discriminants will work. In particular, choosing the same discriminant as the pairing-friendly curve always works, indeed $(t(x) - 2)/y(x) \bmod q(x)$ corresponds to $\sqrt{D} \bmod q(x)$.

The second challenge is solving the equation $4q(x) = t'^2(x) - Dy'^2(x)$ in $t'(x), y'(x)$ such that the Hasse bound is satisfied. At least, one requires the degrees in x of t' and y' to be half the degree of $q(x)$. Sanso and El Housni suggest computing a half-gcd of $q(x)$ and $w(x) = \sqrt{D} \bmod q(x)$ to obtain candidates for the embedded curve parameters $t'(x), y'(x)$ such that their degree in x is at most half the degree of $q(x)$, to later satisfy the Hasse

Table 4: Embedded curve families above BLS12 with $q(x) = x^4 - x^2 + 1$ and $D = -4$. A first pair is $(t_0, y_0) = (2x^2 - 2, x)$ and the other pairs are for the quadratic and quartic twists. The first one's order $q + 1 - t'$ is not prime but can give two times a prime.

(t', y') s.t. $q = (t'^2 + 4y'^2)/4$		$q + 1 - t'$	family
t_0, y_0	$2x^2 - 2, x$	$x^4 - 3x^2 + 4$	(yes)
$-t_0, y_0$	$-2x^2 + 2, x$	$x^4 + x^2 = x^2(x^2 + 1)$	no
$2y_0, t_0/2$	$2x, x^2 - 1$	$x^4 - x^2 - 2x + 2 = (x - 1)^2(x^2 + 2x + 2)$	no
$-2y_0, t_0/2$	$-2x, x^2 - 1$	$x^4 - x^2 + 2x + 2 = (x + 1)^2(x^2 - 2x + 2)$	no

bound $|t'| \leq 2\sqrt{q}$. 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 $\mathrm{GF}(2^n)$ and dates back to 1984. It is known under the name *Waterloo algorithm* [BFHMV84]. The idea is to express the target (a polynomial $w(x)$ 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 $q(x)$ of odd degree n. The half-gcd algorithm on q, w performs an extended gcd algorithm to get the successive coefficients $a_i(x)q(x) + b_i(x)w(x) = r_i(x)$ where r_i is the remainder at step i. The algorithm stops half-way, when b_i and r_i have degrees $\leq (\deg q)/2$.

In the present case, $q(x)$ has usually an even degree, and a half-gcd algorithm on inputs $(q(x), w(x))$ with $\deg q > \deg w$ outputs three polynomials $g(x), u(x), u(x)$ such that $g(x)q(x) = u(x) - v(x)w(x)$ (in our case, usually g is a constant), $\deg u, \deg v \leq \deg q/2$. Luckily for BLS and BN, $g = 1$ and the equation $t'^2 + 3y'^2 = 4q$ is solved, with $t' = 2u$ and $y' = 2v$. But for KSS18 for example, $w = 2x^3 + 37$, $u = 3$, $v = -2x^3 - 37$, $i = 1372$. As noted in [SEH24], Algorithm 2 fails for KSS curves because it does not find a solution $t'(x), y'(x)$ in polynomial form to the norm equation $4q(x) = t'^2(x) + 3y'^2(x)$, such that $t'(x), y'(x)$ can take integer values, and the Hasse bound is satisfied. For all KSS and GG (Gasnier–Guillevic [GG25]) curve families, the parameters are polynomials with rational coefficients, that can have very large denominators. Sanso and El Housni method will not work on these parameters.

But the solution actually exists. For example, the *halographs* project of Daira Hopwood at [Hop20a] obtained the formulas of prime-order j-invariant 0 embedded curves forming a plain cycle for BLS12 and KSS18, in 2020. A careful look at the SageMath source code shows that the same formulas as [SEH24] for BLS12 are used. For KSS18, the change of variables $x \mapsto 7x$ allowed Hopwood obtaining the formulas, avoiding the denominator issue that Sanso and El Housni faced.

4.2 Generic Method

The generic method presented here brings together two strategies: the Smith technique [Smi15] and the DLZZ technique [DLZZ23], originally designed for two other problems respectively. They both exploit the GLV technique [GLV01] with a fast endomorphism ϕ on the curve, of characteristic polynomial $\chi_\phi = X^2 - \mathrm{tr}_\phi X + \deg_\phi$. If ϕ is on the pairing-friendly curve, then in the cyclic subgroup of order q, the endomorphism ϕ has eigenvalue $\lambda_\phi \bmod q$ and corresponds to the scalar multiplication $[\lambda_\phi]$. A Lagrange–Gauss reduction algorithm on the vectors $(q, 0), (-\lambda_\phi, 1)$ forming a basis of a lattice $\mathcal{L}$, gives the shortest vector (a_0, a_1), satisfying $a_0 + a_1\lambda_\phi \equiv 0 \bmod q$, and $|a_i| \approx \sqrt{q}$. More precisely, the shortest vector has Euclidean norm $\|(a_0, a_1)\| = \lambda_1(\mathcal{L}) \leq (4/3)^{1/4} \mathrm{vol}(\mathcal{L})^{1/2} \approx 1.074\sqrt{q}$. The GLV method computes $[q]P$ as $[a_0]P + [a_1]\phi(P)$ with a multi-scalar multiplication.

4.2.1 DLZZ technique for fast $\mathbb{G}_1$ subgroup membership testing

Dai, Lin, Zhao, and Zhou consider pairing-friendly curves and work over the integer values of the curve parameters [DLZZ23]. One aim is obtaining an optimal formula for $\mathbb{G}_1$ subgroup membership testing, where $\mathbb{G}_1$ is the subgroup of order exactly q of $E(\mathbb{F}_p)$ (remember the pairing-friendly curve cofactor c, where $\#E(\mathbb{F}_p) = c \cdot q$). Given a point P on $E(\mathbb{F}_p)$, the test checks that $[q]P = \mathcal{O}$ without computing the full and costly scalar multiplication by q. The GLV method provides two short scalars a_0, a_1, $|a_i| \approx \sqrt{q}$, and such that $a_0 + a_1\lambda_\phi = 0 \bmod q$. However, as pointed out by Dai, Lin, Zhao, and Zhou, computing $[a_0]P + [a_1]\phi(P)$ might actually output a small multiple $[sq]P$ for some integer s, instead of $[q]P$, and the $\mathbb{G}_1$-membership test is not valid if s is not coprime to the curve cofactor c. The authors develop a criterion to test whether the short scalars (a_0, a_1) give a valid subgroup membership test: for $D = -3$, $\gcd(a_0^2 - a_0a_1 + a_1^2, \#E(\mathbb{F}_p)) = q$, for $D = -4$, $\gcd(a_0^2 + a_1^2, \#E(\mathbb{F}_p) = q)$ ([DLZZ23, Theorem 3]). They propose an algorithm and a Magma implementation to compute the (integer) short scalars (a_0, a_1) that pass the test. Actually, the DLZZ criterion on (a_0, a_1) is very close to the norm equation in Section 2: it requires an equality $a_0^2 - a_0a_1 + a_1^2 = q$, resp. $a_0^2 + a_1^2 = q$. The pair of short scalars for KSS curves actually solve the norm equation and are reproduced in Table 5.

Table 5: Short-scalar pair (a_0, a_1) from [DLZZ23, Table 4]. See Table 2 for KSS parameters.

Curve	D	χ_ϕ	$\lambda_\phi \bmod q(x)$	short vector (a_0, a_1)	DLZZ 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	$(-1+\sqrt{-3})/2 = x^3+18$	$((x/7)^3, -18(x/7)^3 - 1)$	$a_0^2 - a_0a_1 + a_1^2 = q$

We 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's technique [Smi15].

4.2.2 The Smith technique

Smith's technique is intended for fast scalar multiplication on the curve. It is stated in dimension ≥ 2, the number of distinct endomorphisms on the curve, the identity Id included. In our case, there are two endomorphisms on the curve $E(\mathbb{F}_p)$ to consider, Id and ϕ, of eigenvalue λ_ϕ and characteristic polynomial $\chi_\phi(X) = X^2 - \mathrm{tr}_\phi X + \deg_\phi$. We recall the Smith technique with $\dim = 2$. Given any integer scalar $0 < m < q$ and point $P \neq \mathcal{O}$ in the cyclic subgroup of order q, compute a fast multi-scalar multiplication $[m_0]P + [m_1]\phi(P) = [m]P$ by decomposing m as $m_0 + m_1\lambda_\phi \bmod q$, where m_0, m_1 are short ($|m_i| \approx \sqrt{q}$). For a fast decomposition, Smith precomputes a short basis (a_0, a_1) such that $a_0 + a_1\lambda_\phi = 0 \bmod q$, and a_0, a_1 are short. These (a_0, a_1) values are precisely what we are looking for. Smith [Smi15] computes a short basis of the lattice $\mathcal{L}$ whose basis is given by the vectors $\vec{b}_1 = (q, 0)$, $\vec{b}_2 = (-\lambda_\phi, 1)$. This is also the aim of [DLZZ23]. We recall [Smi15, Theorem 2].

Theorem 3 ([Smi15, Theorem 2])**.** *Let ϕ be a non-integer endomorphism of E such that $\mathbb{Z}[\pi] \subset \mathbb{Z}[\phi]$, so $\pi = b + c\phi$ for some integers b and c, where π is the Frobenius endomorphism on the curve E. Suppose that the two endomorphisms on E are (Id, ϕ). 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 $\#E(\mathbb{F}_p)$. If $\mathcal{G} = E(\mathbb{F}_p)$, then $\mathcal{L} = \langle \vec{b}_1, \vec{b}_2 \rangle$.

Smith does not consider pairing-friendly curves such as BLS or KSS, but curves of prime order, or with a very small cofactor such as 2. When the curve has prime order q, the lattice $\mathcal{L} = \langle \vec{b}_1, \vec{b}_2 \rangle$ is already reduced of discriminant q. The exact computations are the following. Let $d > 0$ square-free, such that $t^2 - 4p = -dy^2$. Consider the cyclic subgroup of order q of $E(\mathbb{F}_p)$, where ϕ is the CM endomorphism and acts as a scalar multiplication by its eigenvalue, denoted $[\lambda_\phi]$. For $d = 1, 2 \bmod 4$ and $D = -4d$, $\phi = [\sqrt{-d}]$ has characteristic polynomial $X^2 + d$, $\deg_\phi = d$ and $\mathrm{tr}_\phi = 0$, and $\pi = (t + y\sqrt{-d})/2$. The lattice $\mathcal{L}$ is defined by the matrix

$$B = \begin{bmatrix} \frac{t}{2} - 1 & \frac{y}{2} \\ \frac{y}{2} d & 1 - \frac{t}{2} \end{bmatrix}, \ \mathrm{vol}\,\mathcal{L} = |\det(B)| = \frac{(t-2)^2 + dy^2}{4} = p + 1 - t = \#E(\mathbb{F}_p) \ .$$

For $d = 3 \bmod 4$ and $D = -d$, the CM endomorphism is $\phi = (-1 + \sqrt{-d})/2$, $2\phi + 1 = \sqrt{-d}$, and $\pi = (t + y\sqrt{-d})/2 = (t + y)/2 + y\phi$. $\deg_\phi = (d + 1)/4$ and $\mathrm{tr}_\phi = -1$. The lattice is defined by the matrix

$$B = \begin{bmatrix} \frac{t+y}{2} - 1 & y \\ y\frac{d+1}{4} - \frac{t+y}{2} + 1 & 1 - \frac{t+y}{2} \end{bmatrix}, \ \mathrm{vol}\,\mathcal{L} = |\det(B)| = \frac{(t-2)^2 + dy^2}{4} = p + 1 - t = \#E(\mathbb{F}_p) \ .$$

We now adapt this technique to our case.

4.2.3 Our technique for reducing the lattice $\mathcal{L}$

We begin with the same lattice $\mathcal{L}$ whose volume is $\mathrm{vol}\,\mathcal{L} = |\det B| = \#E(\mathbb{F}_p)$. The main difference is due to the large curve cofactor that we need to take out. We also recall that for pairing-friendly curves, a polynomial $a(x) \in \mathbb{Q}[x]$ of denominator an integer m is said to take integer values if there exists a set of congruences $x_i \bmod m$ such that $a(x_i) \in \mathbb{Z}$. The parameters of KSS and GG curve families always come with a description of these congruence constraints.

Theorem 4. *Consider the lattice* $\mathcal{L} = \big\{(a_0(x), a_1(x)) \in \mathbb{Q}[x]^2, \ a_0 + a_1\lambda_\phi = 0 \bmod q(x)\big\}$. *Consider the matrix made of polynomials* $a_i(x) \in \mathbb{Q}[x]$,

$$B_x = \begin{bmatrix} a_0(x) & a_1(x) \\ a_1(x)\deg_\phi + a_0(x)\,\mathrm{tr}_\phi & -a_0(x) \end{bmatrix}, \ \mathrm{vol}\,\mathcal{L} = |\det B_x| = nq(x) \text{ for a cofactor } n$$

such that $n \in \mathbb{Z}$ *or* $n \in \mathbb{Q}[x]$ *takes integer values at* $x_i \bmod m$. *The first row* (a_0, a_1) *satisfies the norm equation* $a_0^2 + a_0 a_1\,\mathrm{tr}_\phi + a_1^2 \deg_\phi = nq(x)$ *(matrix determinant formula). The second row* $(a_0'(x), a_1'(x))$ *seen as* $a_0' + a_1'\phi \in \mathbb{Z}[\phi]$ *has algebraic norm equation equals to* $\deg_\phi nq(x)$. *Let*

$$N = \begin{bmatrix} b_0 & b_1 \\ b_1\deg_\phi + b_0\,\mathrm{tr}_\phi & -b_0 \end{bmatrix}, \text{ such that } |\det N| = n \ .$$

The first row (b_0, b_1) *has algebraic norm* $b_0^2 + b_0 b_1\,\mathrm{tr}_\phi + b_1^2 \deg_\phi = |\det N| = n$, *the second row has algebraic norm* $\deg_\phi n$. *Assume moreover that* a_1, b_1 *are invertible modulo* nq *and* n *resp., and* $a_0/a_1 = b_0/b_1 = \lambda_\phi \bmod n$.

Then define the matrix

$$M_x = \frac{1}{n} N B_x, \ |\det M_x| = \frac{1}{n^2} \cdot n \cdot nq(x) = q(x) \ .$$

The first row of M_x *has algebraic norm* $q(x)$ *and the second row has algebraic norm* $\deg_\phi q(x)$. *Moreover if* $a_0(x), a_1(x), b_0, b_1$ *take integer values, then the coefficients of* M_x *also take integer values, and the first row of* M_x *provides a solution* (c_0, c_1) *to the norm equation* $c_0^2 + t_\phi c_0 c_1 + c_1^2 \deg_\phi = q(x)$.

Proof. A direct computation of NB_x (double-checked with SageMath) gives

$$nM_x = \begin{bmatrix} a_0b_0 + a_0b_1 \operatorname{tr}_\phi + a_1b_1 \deg_\phi & b_0a_1 - a_0b_1 \\ -\deg_\phi(b_0a_1 - a_0b_1) & a_0b_0 + b_0a_1 \operatorname{tr}_\phi + a_1b_1 \deg_\phi \end{bmatrix} .$$

1. We first need to show that each entry of nM_x is divisible by n. From the assumption $a_0^2 + a_0a_1 \operatorname{tr}_\phi + a_1^2 \deg_\phi = nq(x)$ and moreover if a_1 is invertible modulo $nq(x)$, dividing the equation by a_1^2 we obtain $\chi_\phi(\frac{a_0}{a_1}) = 0 \bmod nq(x)$ hence $\frac{a_0}{a_1} = \lambda_\phi \bmod nq(x)$. We proceed similarly with the equation $b_0^2 + b_0b_1 \operatorname{tr}_\phi + b_1^2 \deg_\phi = n$, dividing by b_1^2 to obtain $\chi_\phi(\frac{b_0}{b_1}) = 0 \bmod n$. Here we use the assumption that $\frac{a_0}{a_1}$ and $\frac{b_0}{b_1}$ coincide to λ_ϕ modulo n. We deduce $a_0b_1 - a_1b_0 = 0 \bmod n$, hence the anti-diagonal terms of nM_x are divisible by n.

   For the first diagonal item $a_0b_0 + a_0b_1 \operatorname{tr}_\phi + a_1b_1 \deg_\phi$, dividing by a_1b_1 and reducing modulo n, we obtain $\frac{a_0}{a_1}\frac{b_0}{b_1} + \frac{a_0}{a_1} \operatorname{tr}_\phi + \deg_\phi = \lambda_\phi^2 + \operatorname{tr}_\phi \lambda_\phi + \deg_\phi = \chi_\phi(\lambda_\phi) = 0 \bmod n$.

   For the last one, $a_0b_0 + b_0a_1 \operatorname{tr}_\phi + a_1b_1 \deg_\phi$, dividing by a_1b_1 and reducing modulo n, we obtain $\frac{a_0}{a_1}\frac{b_0}{b_1} + \frac{b_0}{b_1} \operatorname{tr}_\phi + \deg_\phi = \lambda_\phi^2 + \operatorname{tr}_\phi \lambda_\phi + \deg_\phi = \chi_\phi(\lambda_\phi) = 0 \bmod n$.

   Both diagonal values each divided by a_1b_1 are multiple of n. Finally, $nM_x = NB_x$ can be divided by n so that $M_x = \frac{1}{n}NB_x$ has entries that take integer values.

2. By multiplicativity of the determinant, $|\det M_x| = (1/n)^2 |\det(N)\det(B_x)| = q(x)$.

3. A direct calculus (double-checked with SageMath) shows that the first row (c_0, c_1) of M_x satisfies $c_0^2 + c_0c_1 \operatorname{tr}_\phi + c_1^2 \deg_\phi = -\det(B_x)/n = q(x)$. The second row (d_0, d_1) of M_x satisfies $d_0^2 + d_0d_1 \operatorname{tr}_\phi + d_1^2 \deg_\phi = -\deg_\phi \det(B_x)/n = \deg_\phi q(x)$.

The SageMath script is provided in Section A. □

4.2.4 Our technique for parameterized families of pairing-friendly curves

We consider the pairing-friendly curve subgroup order $q(x)$ where $q(x)$ is a polynomial that takes prime integer values at $x_i \bmod m$. Let d be a positive square-free integer that has a square root modulo $q(x)$. The embedded curve discriminant is $D = -4d$ if $d = 1, 2 \bmod 4$, $D = -d$ if $d = 3 \bmod 4$. Note that D can be distinct from the pairing-friendly curve discriminant.

We aim at reducing the lattice $\mathcal{L}$ defined by the matrix B_x of polynomials in $\mathbb{Q}[x]$

$$B_x = \begin{bmatrix} q(x) & 0 \\ -\lambda_\phi(x) & 1 \end{bmatrix} \text{ where } \lambda_\phi = \begin{cases} \sqrt{-d} \bmod q \text{ if } d = 1, 2 \bmod 4, \\ \frac{-1+\sqrt{-d}}{2} \bmod q \text{ if } d = 3 \bmod 4 . \end{cases}$$

Following Sanso and El Housni idea, we compute $\lambda_\phi(x) = \sqrt{-d}$, resp. $\lambda_\phi(x) = (-1 + \sqrt{-d})/2$ modulo $q(x)$ in polynomial form. Then we run a half-gcd algorithm to obtain $-\lambda_\phi(x) = a_0(x)/a_1(x)$ satisfying $\deg(a_0), \deg(a_1) \leq (\deg q)/2$ and a_0, a_1 coprime in $\mathbb{Q}[x]$. The basis B_x becomes

$$B_x = \begin{bmatrix} a_0(x) & a_1(x) \\ \deg_\phi a_1(x) + \operatorname{tr}_\phi a_0(x) & -a_0(x) \end{bmatrix}, \; |\det(B_x)| = a_0^2 + a_0a_1 \operatorname{tr}_\phi + a_1^2 \deg_\phi = 0 \bmod q(x) .$$

We know that the cofactor n in $|\det(B_x)| = nq(x)$ is an integer not a polynomial, because the degrees of $a_i(x)$ are no more than $\deg q(x)/2$, and $\operatorname{tr}_\phi, \deg_\phi \in \mathbb{N}$.

At this point we exploit Theorem 4. Note that $n \in \mathbb{Z}$ (this is not a polynomial). If $\lambda_\phi(x) = -(a_0/a_1)(x) \bmod n$ has entries in $\mathbb{Z}$ (all the terms in x are multiple of n), we apply a lattice reduction algorithm to the basis made of the vectors $(n, 0), (-\lambda_\phi(x) \bmod n, 1)$ to obtain N. Otherwise, we consider separately the congruence constraints $x_i \bmod m$, we take $\lambda_\phi(mx + x_i) \bmod n$, and consider recursively smaller factors n_i of n such that

$\lambda_\phi(mx+x_i) = -(a_0/a_1)(mx+x_i) \bmod n_i$ has entries in $\mathbb{Z}$. At the end of this process we obtain a reduced basis whose determinant is exactly $q(x)$ and whose first row provides a solution to the norm equation. We formalize it as Algorithm 3.

Algorithm 3: Embedded-curve families

Input: Embedded curve field characteristic $q(x)$, congruence constraints $\{x_i \bmod m\}_i$, positive square-free integer d being a square mod $q(x)$

Output: Solutions (a_0, a_1) satisfying the algebraic norm equation for d, for each x_i.

$L \leftarrow \{\}$
if $d = 1, 2 \bmod 4$ **then** $\lambda_\phi(x) \leftarrow \sqrt{-d} \bmod q(x)$
else $\lambda_\phi(x) \leftarrow (-1+\sqrt{-d})/2 \bmod q(x)$ $(d = 3 \bmod 4)$
$(a_0, a_1) \leftarrow$ Half-GCD$(-\lambda_\phi, q(x))$ $(a_0/a_1 = -\lambda_\phi \bmod q(x)$, $\deg a_0, a_1 \leq \deg q(x)/2)$
$B_x \leftarrow \begin{bmatrix} a_0(x) & a_1(x) \\ \deg_\phi a_1(x) + \mathrm{tr}_\phi\, a_0(x) & -a_0(x) \end{bmatrix}$
$n \leftarrow |\det B_x|/q(x) \in \mathbb{Z}$
for $x_i \bmod m$ **do**
  for n_i *such that* $\prod_i n_i = n$ *and* $((a_0/a_1)(mx+x_i) \bmod n_i) \in \mathbb{Z}$ **do**
    $N \leftarrow \begin{bmatrix} n_i & 0 \\ (a_0/a_1)(mx+x_i) \bmod n_i & 1 \end{bmatrix}$
    $N \leftarrow$ Lagrange-Gauss-Reduce(N) such that $|\det N| = n_i$
    $B_x \leftarrow \frac{1}{n_i} N B_x$
  $(b_0, b_1) \leftarrow$ first row of B_x ; $L \leftarrow L \cup \{(b_0, b_1, x_i)\}$
return the list L of (b_0, b_1, x_i) for each $x_i \bmod m$

4.3 Practical Results

Example 1 (KSS18, $D = -3$)**.** Table 2 gives the KSS18 parameters. The KSS18 solution is provided in [DLZZ23, Remark 4] (see Table 5). We reproduce the results with our technique. From $q(x) = (x^6 + 37x^3 + 343)/343$, the eigenvalue of ϕ is $\lambda_\phi = x^3 + 18$ and satisfies $\lambda_\phi^2 + \lambda_\phi + 1 = 343q(x)$.

$$B_x = \begin{bmatrix} -\lambda_\phi & 1 \\ 1+\lambda_\phi & \lambda_\phi \end{bmatrix} = \begin{bmatrix} -x^3-18 & 1 \\ x^3+19 & x^3+18 \end{bmatrix}, |\det B_x| = 343q(x) = 7^3 q(x)\ .$$

Observe that with x multiple of 7 (indeed, the KSS18 constraint is $x = 14 \bmod 21$), then $N = B_x \bmod 343$, and $M_x = \frac{1}{343} N B_x$ has exactly determinant $q(x)$, where

$$N = \begin{bmatrix} -18 & 1 \\ 19 & 18 \end{bmatrix}, |\det N| = 343\ ;\ M_x = \frac{1}{343} N B_x = \begin{bmatrix} 19x'^3+1 & x'^3 \\ -x'^3 & 18x'^3+1 \end{bmatrix},\ x' = \frac{x}{7}\ .$$

This was the missing step in Algorithm 2 of Sanso and El Housni. With $(a_0, a_1) = (19(x/7)^3 + 1, (x/7)^3)$ the first row of M_x, one embedded curve family has trace $t'(x) = 2a_0 - a_1 = 37(x/7)^3 + 2$ and $y'(x) = a_1 = (x/7)^3$, such that $4q(x) = t'^2(x) + 3y'^2(x)$. The second row gives the trace $t''(x) = -20(x/7)^3 - 1$, $y''(x) = 18(x/7)^3 + 1$. The formulas of quadratic, cubic and sextic twists give the other families of embedded curves (Table 6). The seed $x_0 = -\texttt{0xfdde07f8000} = -2^{44} + 2^{37} + 2^{33} + 2^{29} - 2^{23} + 2^{15}$ with the 2nd row of Table 6 provides a KSS18 curve $y^2 = x^3 + 13$ over $\mathbb{F}_p$ with a 348-bit p, a 256-bit prime subgroup order q, 2-valuation $2^{45} | q - 1$. The embedded curve E_1 over $\mathbb{F}_q$ has equation $y^2 = x^3 + 13$, prime-order $r = (x_0^6 + 57x_0^3 + 1029)/343$ of 256 bits. The prime-order curve $E_0/\mathbb{F}_r$ has equation $y^2 = x^3 - 4$, order q and forms a plain cycle with E_1. The seed $x_0 = -\texttt{0xfd7ffdee000} = -2^{44} + 2^{37} + 2^{35} + 2^{21} + 2^{16} + 2^{13}$ with the 3rd row of Table 6

provides a KSS18 curve $y^2 = x^3 + 2$ over $\mathbb{F}_p$ of 348 bits, it has a 256-bit prime subgroup order q, 2-valuation $2^{39}|q-1$. The embedded curve E_1 over $\mathbb{F}_q$ has equation $y^2 = x^3 + 7$, prime-order $r = (x_0^6 + 54x_0^3 + 1029)/343$ of 256 bits. The prime-order curve $E_0/\mathbb{F}_r$ has equation $y^2 = x^3 + 2$, order q and forms a plain cycle with E_1.

Table 6: Embedded curves with KSS18, $D = -3$. A first pair is $(t_0, y_0) = (20(x/7)^3 + 1, 18(x/7)^3 + 1)$ and the other pairs are for the quadratic, cubic and sextic twists. The first and fifth one's order $r = q + 1 - t'$ are irreducible but multiple of 3.

(t', y') s.t. $q = (t'^2 + 3y'^2)/4$		$r = q + 1 - t'$	family
t_0, y_0	$20(x/7)^3 + 1, 18(x/7)^3 + 1$	$(x^6 + 17x^3 + 343)/343$	(yes, 3)
$-t_0, y_0$	$-20(x/7)^3 - 1, 18(x/7)^3 + 1$	$(x^6 + 57x^3 + 1029)/343$	yes
$(t_0 - 3y_0)/2, (t_0 + y_0)/2$	$-17(x/7)^3 - 1, 19(x/7)^3 + 1$	$(x^6 + 54x^3 + 1029)/343$	yes
$(t_0 + 3y_0)/2, (t_0 - y_0)/2$	$37(x/7)^3 + 2, (x/7)^3$	$x^6/7^3$	no
$-(t_0 + 3y_0)/2, (t_0 - y_0)/2$	$-37(x/7)^3 - 2, (x/7)^3$	$(x^6 + 74x^3 + 1372)/343$	(yes, 3)
$-(t_0 - 3y_0)/2, (t_0 + y_0)/2$	$17(x/7)^3 + 1, 19(x/7)^3 + 1$	$(x^2 - 4x + 7)(x^2 - x + 7)(x^2 + 5x + 7)/343$	no

Example 2 (KSS16 and $D = -4$)**.** On the KSS16 curves, $q(x) = (x^8 + 48x^4 + 625)/61250$ and the endomorphism has characteristic polynomial $\chi = X^2 + 1$. A solution is already provided in Table 5. We reproduce it with $\lambda_\phi = (x^4 + 24)/7$, $\mathrm{tr}_\phi = 0$, $\deg_\phi = 1$, $(a_0, a_1) = (-\lambda_\phi, 1)$ (because $q(x)$ has degree 8 and λ_ϕ has degree 4, we don't reduce further the degree of λ_ϕ), $B_x = \begin{bmatrix} a_0 & a_1 \\ a_1 & -a_0 \end{bmatrix}$ has determinant $|\det B_x| = nq(x)$, $n = 1250 = 2 \cdot 5^4$. The KSS16 parameters take integer values for $x = 25, 45 \bmod 70$. With the change of variables $x \leftarrow 70x + 25$, the denominator 7 vanishes and the x-coefficient of $(a_0/a_1)(x) = -\lambda_\phi$ is a multiple of n. Reducing $\begin{bmatrix} 1250 & 0 \\ -807 & 1 \end{bmatrix}$, $N = \begin{bmatrix} 31 & 17 \\ -17 & 31 \end{bmatrix}$ has determinant exactly 1250. One row of $M_x = \frac{1}{1250} N B_x$ (up to sign) is $(a_0, a_1) = ((31(\frac{x}{5})^4 + 1)/14, (17(\frac{x}{5})^4 + 1)/14)$ such that $a_0^2 + a_1^2 = q(x)$ exactly, as in Table 5. The quadratic and quartic twists give the other solutions in Table 7.

Table 7: Embedded curves for KSS16, parameters (t', y') such that $q = (t'^2 + 4y'^2)/4$ with $D = -4$. A first pair is $(t_0, y_0) = ((31(x/5)^4 + 1)/7, (17(x/5)^4 + 1)/14)$ 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', y') s.t. $q = (t'^2 + 4y'^2)/4$		$r = q + 1 - t'$	family
t_0, y_0	$(31(x/5)^4 + 1)/7, (17(x/5)^4 + 1)/14$	$(x^8 - 386x^4 + 5^5 \cdot 17)/61250$	(yes, 2)
$-t_0, y_0$	$(-31(x/5)^4 - 1)/7, (17(x/5)^4 + 1)/14$	$(x^8 + 482x^4 + 5^4 \cdot 113)/61250$	(yes, 2)
$2y_0, t_0/2$	$(17(x/5)^4 + 1)/7, (31(x/5)^4 + 1)/14$	$(x^8 - 190x^4 + 5^5 \cdot 17)/61250$	(yes, 20)
$-2y_0, t_0/2$	$(-17(x/5)^4 - 1)/7, (31(x/5)^4 + 1)/14$	$(x^8 + 286x^4 + 5^4 \cdot 113)/61250$	(yes, 32)

Example 3 (GG20a and $d = 1$)**.** The parameters are $q(x) = (x^8 + 4x^7 + 11x^6 + 24x^5 + 41x^4 + 120x^3 + 275x^2 + 500x + 625)/25625$, $\lambda_\phi = (x^5 + 38)/41 \bmod q(x)$ such that $\lambda_\phi^2 + 1 = 0 \bmod q(x)$. The degree of λ_ϕ is too large, and we obtain instead $\lambda_\phi = \frac{x^4+2x^3+3x^2+2x-7}{x^3+4x^2+11x+24} = \frac{a_0}{a_1} \bmod q(x)$, that satisfies $a_0^2 + a_1^2 = 41 \cdot 5^4 q(x)$. The congruence constraints are $x = 1715, 1815 \bmod 2050$ for $q(x)$ and the other parameters to take integer values. On one hand, for $x = 1715 \bmod 2050$, the matrix

$$M_x = \frac{1}{25625} \begin{bmatrix} 4 & -5 \\ 5 & 4 \end{bmatrix} \begin{bmatrix} 4 & -3 \\ 3 & 4 \end{bmatrix}^2 B_x \text{ where } B_x = \begin{bmatrix} a_0 & a_1 \\ a_1 & -a_0 \end{bmatrix}$$

Table 8: 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 x_0	L $2^L \mid q-1$	equation $E_{\mathrm{KSS}}/\mathbb{F}_p$	p (bits)	q (bits)	embedded curve equation $E_1/\mathbb{F}_q$	h
$r = (x^8 - 386x^4 + 5^5 \cdot 17)/61250$ (row 1 in Table 7)						
`0x37effef25` $= 45 \bmod 70$ $2^{34} - 2^{31} - 2^{24} - 2^{12} - 2^8 + 2^5 + 2^2 + 1$	5	$y^2 = x^3 + 25x$	329	255	$y^2 = x^3 + 3x$	2
`0x36007bf3f` $= 25 \bmod 70$ $2^{34} - 2^{31} - 2^{29} + 2^{19} - 2^{14} - 2^8 + 2^6 - 1$	4	$y^2 = x^3 + 11x$	328	255	$y^2 = x^3 + 3x$	2
$r = (x^8 - 190x^4 + 5^5 \cdot 17)/61250$ (row 3 in Table 7)						
`0x3dee0008d` $= 25 \bmod 70$ $2^{34} - 2^{29} - 2^{24} - 2^{21} + 2^7 + 2^4 - 2^2 + 1$	6	$y^2 = x^3 + 2x$	330	256	$y^2 = x^3 + 3x$	20

after re-arranging the signs, gives the solution (a_0, a_1) such that $a_0^2 + a_1^2 = q(x)$, $(a_0, a_1) =$

$$\left(\frac{1}{41}\left(131x'^4 + 34x'^3 + x'^2 - 6x' - 5\right), \frac{1}{41}\left(92x'^4 + 63x'^3 + 32x'^2 + 13x' + 4\right)\right),\ x' = \frac{x}{5}\ .$$

The seed $x_0 = 9712385115$ produces p of 384 bits, q of 251 bits, and $(q + 1 - 2a_1)/2$ prime of 250 bits.

On the other hand, for $x = 1815 \bmod 2050$, the matrix

$$M_x = \frac{1}{25625}\begin{bmatrix}5 & -4\\ 4 & 5\end{bmatrix}\begin{bmatrix}4 & -3\\ 3 & 4\end{bmatrix}^2 B_x$$

after re-arranging the signs, gives a solution (a_0, a_1) such that $a_0^2 + a_1^2 = q(x)$,

$$\left(\frac{1}{41}\left(148x'^4 + 47x'^3 + 8x'^2 - 3x' - 4\right), \frac{1}{41}\left(61x'^4 + 54x'^3 + 31x'^2 + 14x' + 5\right)\right),\ x' = \frac{x}{5}\ .$$

The seed $x_0 = 9825684615$ produces p of 384 bits, q of 251 bits, and $(q + 1 + 2a_0)/2$ prime of 250 bits.

Example 4 (GG20a and $d = 5$)**.** To highlight the possibility of choosing a distinct discriminant for the embedded curve, we give embedded curve families with $d = 5$ for the pairing-friendly curve family GG20a. The integer $d = 5$ is a square modulo $q(x)$ (Example 3). To start, $\sqrt{-5} = \frac{s_0}{s_1} = \frac{x^4+2x^3+x^2+10x+25}{x^3-5x} \bmod q(x)$. Both s_0 and s_1 are multiple of 25 for x multiple of 5. Assuming $x = 1715 \bmod 2050$, define

$$M_x = \frac{1}{41}NB_x = \frac{1}{41}\begin{bmatrix}6 & 1\\ 5 & -6\end{bmatrix} \cdot \begin{bmatrix}s_0/25 & s_1/25\\ s_1/5 & -s_0/25\end{bmatrix}\ , |\det M_x| = q(x)\ .$$

The first row (a_0, a_1) of M_x satisfies $a_0^2 + 5a_1^2 = q(x)$, where $(a_0, a_1) =$

$$\left(\frac{1}{41}\left(150x'^4 + 85x'^3 + 6x'^2 + 7x' + 6\right), \frac{1}{41}\left(25x'^4 - 20x'^3 + x'^2 + 8x' + 1\right)\right)\ , x' = \frac{x}{5}\ .$$

One embedded curve family has order $q(x) + 1 - 2a_0(x) = 10r'(x)$ always multiple of 10, and $r'(x)$ generates primes. The quadratic twist has order $q(x)+1+2a_0(x) = 2r(x)$ always even, and $r(x)$ generates primes. The seed $x_0 = 9527661665$ produces q of 251 bits, p of 383 bits, and the embedded curve order is $(q + 1 + 2a_0) = 2r_0$ with r_0 prime of 250 bits.

With $x = 1815 \bmod 2050$, the matrix N changes, define

$$M_x = \frac{1}{41}NB_x = \frac{1}{41}\begin{bmatrix}-6 & 1\\ 5 & 6\end{bmatrix} \cdot \begin{bmatrix}s_0/25 & s_1/25\\ s_1/5 & -s_0/25\end{bmatrix}\ , |\det M_x| = q(x)\ .$$

The first row (a_0, a_1) of M_x satisfies $a_0^2 + 5a_1^2 = q(x)$, where $(a_0, a_1) =$

$$\left(\frac{1}{41}\left(150x'^4 + 35x'^3 + 6x'^2 + 17x' + 6\right), \frac{1}{41}\left(25x'^4 + 40x'^3 + x'^2 - 4x' + 1\right)\right), x' = \frac{x}{5}.$$

One embedded curve family has order $q(x) + 1 - 2a_0(x) = 10r'(x)$ always multiple of 10, and $r'(x)$ generates primes. The quadratic twist has order $q(x) + 1 + 2a_0(x) = 2r(x)$ always even, and $r(x)$ generates primes. The seed $x_0 = 9760945615$ gives q prime of 251 bits, p prime of 384 bits, and the embedded curve order is $(q + 1 + 2a_0) = 2r_0$ with r_0 prime of 250 bits.

Another example with GG22D7 is given in appendix, Example 5.

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 9 of Hamming weight up to 6 in signed binary representation.

Moreover, with a larger search space (Hamming weight 7), we were able to obtain seeds in Table 10 such that the BLS12 curve E admits at the same time a prime-order embedded curve E_1 (with its cycle plain curve E_0) and a second embedded curve E_2 of order 4 times a prime (like in the CØCØ construction), see Figure 6. We think it can be of interest for interoperability purposes.

Table 9: Seeds x_0 of Hamming weight ≤ 6 such that the BLS12 curve $E/\mathbb{F}_p$ has a high 2-valuation $2^L \mid p-1$, $2^L \mid q-1$ and admits a prime-order embedded curve $E_1/\mathbb{F}_q$ of j-invariant 0 that has a plain cycle curve $E_0/\mathbb{F}_r$, and $2^L \mid r-1$. For $2^L \mid x_0 - 1$, x_0 is odd and the order is necessarily $r = x_0^4 - 3x_0^2 + 3$ because $r' = x_0^4 + 3$ is even for odd seeds x_0. All curves have $D = -3$.

seed	L	equation $E_{\mathrm{BLS}}/\mathbb{F}_p$	p (bits)	q (bits)	embedded curve equation $E_1/\mathbb{F}_q$	plain cycle curve equation $E_0/\mathbb{F}_r$
`0x9ffc012000000001` $2^{63} + 2^{61} - 2^{50} + 2^{40} + 2^{37} + 1$	37	$y^2 = x^3 + 1$	379	254	$y^2 = x^3 + 7$	$y^2 = x^3 + 15$
`-0xff97ffdfffffffff` $-2^{64} + 2^{55} - 2^{53} + 2^{51} + 2^{37} + 1$	37	$y^2 = x^3 + 1$	383	256	$y^2 = x^3 + 11$	$y^2 = x^3 + 7$
`0x87fbc01000000001` $2^{63} + 2^{59} - 2^{50} - 2^{46} + 2^{36} + 1$	36	$y^2 = x^3 + 1$	377	253	$y^2 = x^3 + 13$	$y^2 = x^3 + 11$
`0x80067fff00000001` $2^{63} + 2^{51} - 2^{49} + 2^{47} - 2^{32} + 1$	32	$y^2 = x^3 + 1$	377	253	$y^2 = x^3 + 15$	$y^2 = x^3 + 5$

Table 10: Seeds x_0 of Hamming weight 7 such that the BLS12 curve $E/\mathbb{F}_p$ has a high 2-valuation, a prime-order embedded curve $E_1/\mathbb{F}_q$ with a plain cycle curve $E_0/\mathbb{F}_r$ and a second embedded curve $E_2/\mathbb{F}_q$ of order $x^4 + 3 = 4s$ where s is prime. All curves have $D = -3$.

seed x_0	L $2^L \mid q-1$	equation $E_{\mathrm{BLS}}/\mathbb{F}_p$	p (bits)	q (bits)	embedded curve equation $E_{1,2}/\mathbb{F}_q$	plain cycle curve equation $E_0/\mathbb{F}_r$
`0xffff007fda000001` $2^{64} - 2^{48} + 2^{39} - 2^{29} - 2^{27} + 2^{25} + 1$	25	$y^2 = x^3 + 1$	383	256	E_1: $y^2 = x^3 + 19$ E_2: $y^2 = x^3 + 17$	$y^2 = x^3 + 7$
`0xfc3ec00400000001` $2^{64} - 2^{58} + 2^{54} - 2^{48} - 2^{46} + 2^{34} + 1$	34	$y^2 = x^3 + 1$	383	256	E_1: $y^2 = x^3 + 23$ E_2: $y^2 = x^3 + 29$	$y^2 = x^3 + 29$
`-0xef000ffefdffffff` $-2^{64} + 2^{60} + 2^{56} - 2^{44} + 2^{32} + 2^{25} + 1$	25	$y^2 = x^3 + 1$	382	256	E_1: $y^2 = x^3 + 11$ E_2: $y^2 = x^3 + 17$	$y^2 = x^3 + 17$
`0xdf07fffdfc000001` $2^{64} - 2^{61} - 2^{56} + 2^{51} - 2^{33} - 2^{26} + 1$	26	$y^2 = x^3 + 1$	382	256	E_1: $y^2 = x^3 + 11$ E_2: $y^2 = x^3 + 23$	$y^2 = x^3 + 7$

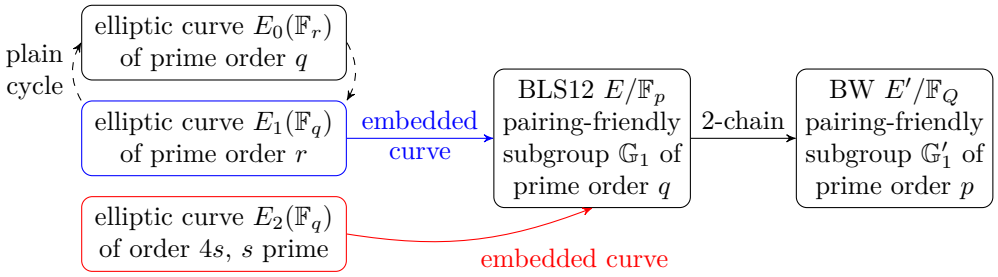


Figure 6: Plain embedded cycle and embedded curve associated to an inner BLS12 curve, and an outer curve forming a 2-chain.

5 Conclusion

In this paper, we considered the search of embedded curves in the context of zero-knowledge proofs. The CM method implemented in [AM93] provides an optimization of the algorithm in [MSZ24], accelerating the curve search. The strategy avoids computing Hilbert class polynomials, but instead gets the curve parameters (up to isomorphism) using imaginary quadratic field results. However, in order to compute the curve coefficients, the computation of the Hilbert class polynomial (and its factorization) is still needed, as we follow the complex multiplication method.

In a first part, we obtain new embedded elliptic curves using this algorithm, considering fixed prime fields:

- A prime-order embedded elliptic curve for statements, with BLS12-381 as the proof curve. It leads to a plain cycle with another curve, enabling recursive ZK proofs.
- An elliptic curve for proving arithmetic circuits derived from statements expressed in Ed25519. Moreover, it is of prime order, enabling a 2-cycle for recursive proofs as in the previous case.
- An embedded curve with BLS12-381 similar to [MSZ24]. The endomorphism of this curve is not competitive with 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 possibilities 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 x (so that $\mathbb{F}_q$ can be defined using $q(x)$). In particular, we obtain a family of prime-order embedded curves of discriminant $D = -3$, defining a family of plain 2-cycle curves, with the pairing-friendly curves KSS18 and SG54 respectively, and even-order embedded families with KSS16, GG20, and GG22 curves.

Acknowledgments. This work started after a discussion with Carla Ràfols at the IMACC'23 conference at Royal Holloway in December 2023. The authors thank Carla Ràfols, Anca Nitulescu, Javier Silva, and Nikitas Paslis for the fruitful discussions and many shared references on proof systems. The authors thank Youssef El Housni and Antonio Sanso for discussions on embedded curves. The authors thank the reviewers for their useful feedback on an earlier version of the paper.

References

[AEG23] 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.

[BCG+14] Eli Ben-Sasson, Alessandro Chiesa, Christina Garman, Matthew Green, Ian Miers, Eran Tromer, and Madars Virza. Zerocash: Decentralized anonymous payments from Bitcoin. In *2014 IEEE Symposium on Security and Privacy*, pages 459–474. IEEE Computer Society Press, May 2014. doi:10.1109/SP.2014.36.

[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.

[BCTV14] Eli Ben-Sasson, Alessandro Chiesa, Eran Tromer, and Madars Virza. Scalable zero knowledge via cycles of elliptic curves. In Juan A. Garay and Rosario Gennaro, editors, *CRYPTO 2014, Part II*, volume 8617 of *LNCS*, pages 276–294. Springer, Berlin, Heidelberg, August 2014. doi:10.1007/978-3-662-44381-1_16.

[BFHMV84] 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.

[BS25] Alp Bassa and Ben Sepanski. Security assessment for the helios–selene curve cycle. https://moneroresearch.info/index.php?action=resource_RESOURCEVIEW_CORE&id=279&list=1, 2025. Accessed: 2025-12-09.

[CBBZ23] Binyi Chen, Benedikt Bünz, Dan Boneh, and Zhenfei Zhang. HyperPlonk: Plonk with linear-time prover and high-degree custom gates. In Carmit Hazay and Martijn Stam, editors, *EUROCRYPT 2023, Part II*, volume 14005 of *LNCS*, pages 499–530. Springer, Cham, April 2023. doi:10.1007/978-3-031-30617-4_17.

[CCN24] Maria Corte-Real Santos, Craig Costello, and Michael Naehrig. On cycles of pairing-friendly abelian varieties. In Leonid Reyzin and Douglas Stebila, editors, *CRYPTO 2024, Part IX*, volume 14928 of *LNCS*, pages 221–253. Springer, Cham, August 2024. doi:10.1007/978-3-031-68400-5_7.

[CCW19] Alessandro Chiesa, Lynn Chua, and Matthew Weidner. On cycles of pairing-friendly elliptic curves. *SIAM Journal on Applied Algebra and Geometry*, 3(2):175–192, 2019. doi:10.1137/18M1173708.

[CK26] Craig Costello and Gaurish Korpal. Cycles of supersingular elliptic curves for pairing-based proof systems. *CiC*, 2(4):29, 2026. doi:10.62056/ahmpdk5vt.

[Coh93] Henri Cohen. *A course in computational algebraic number theory*, volume 138 of *Graduate Texts in Mathematics*. Springer, 1993. doi:10.1007/978-3-662-02945-9.

[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.

[Deu58] Max Deuring. Die klassenkörper der komplexen multiplikation. In *Enzyklopädie der Mathematischen Wissenschaften mit Einschluss ihrer Anwendungen*. Teubner, Stuttgart, 1958. Bd. 1, H. 10, T. 2.

[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.

[Eng09] Andreas Enge. The complexity of class polynomial computation via floating point approximations. *Mathematics of Computation*, 78(266):1089–1107, 2009. https://www.multiprecision.org/cm/index.html. doi:10.1090/S0025-5718-08-02200-X.

[FST10] David Freeman, Michael Scott, and Edlyn Teske. A taxonomy of pairing-friendly elliptic curves. *Journal of Cryptology*, 23(2):224–280, April 2010. doi:10.1007/s00145-009-9048-z.

[GG25] Jean Gasnier and Aurore Guillevic. An algebraic point of view on the generation of pairing-friendly curves. *SIAM Journal on Applied Algebra and Geometry*, 9(2):456–480, 2025. doi:10.1137/23M1601961.

[GKR+21] Lorenzo Grassi, Dmitry Khovratovich, Christian Rechberger, Arnab Roy, and Markus Schofnegger. Poseidon: A new hash function for zero-knowledge proof systems. In Michael Bailey and Rachel Greenstadt, editors, *USENIX Security 2021*, pages 519–535. USENIX Association, August 2021. URL: https://www.usenix.org/conference/usenixsecurity21/presentation/grassi.

[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.

[KB16] Taechan Kim and Razvan Barbulescu. Extended tower number field sieve: A new complexity for the medium prime case. In Matthew Robshaw and Jonathan Katz, editors, *CRYPTO 2016, Part I*, volume 9814 of *LNCS*, pages 543–571. Springer, Berlin, Heidelberg, August 2016. doi:10.1007/978-3-662-53018-4_20.

[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. *DCC*, 92(12):4131–4143, 2024. doi:10.1007/s10623-024-01472-0.

[NPR26] Anca Nitulescu, Nikitas Paslis, and Carla Ràfols. FLIP-and-prove R1CS. *IACR Communications in Cryptology*, 2(4), 2026. doi:10.62056/ayc3tx4e-.

[SEH24] Antonio Sanso and Youssef El Housni. Families of prime-order endomorphism-equipped embedded curves on pairing-friendly curves. In Carla Rafols and Benoit Libert, editors, *Journal of Cryptology*, number 37 in Topical Collection on Modern Zero-Knowledge Protocols. Springer, October 8 2024. ePrint:2023/1662. doi:10.1007/s00145-024-09514-5.

[SG18] Michael Scott and Aurore Guillevic. A new family of pairing-friendly elliptic curves. In Lilya Budaghyan and Francisco Rodríguez-Henríquez, editors, *Arithmetic of Finite Fields*, pages 43–57, Cham, 2018. Springer. ePrint:2018/193. doi:10.1007/978-3-030-05153-2_2.

[Sma99] Nigel P. Smart. Elliptic curve cryptosystems over small fields of odd characteristic. *Journal of Cryptology*, 12(2):141–151, March 1999. doi:10.1007/PL00003820.

[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.

[ST61] Gorō Shimura and Yutaka Taniyama. Complex multiplication of abelian varieties and its applications to number theory. In *Publications of the Mathematical Society of Japan*, volume 6. Mathematical Society of Japan, Tokyo, 1961.

[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.

[The23] The PARI Group, Univ. Bordeaux. *PARI/GP version 2.15.5*, 2023. available from https://pari.math.u-bordeaux.fr/.

[Val21] Henry De Valence. The ristretto group. https://ristretto.group, 2021.

[ZCa21] ZCash. What is jubjub? https://z.cash/technology/jubjub/, 2021.

A Sagemath verification script for Theorem 4

```
QQab.<a0,b0,a1,b1,deg_,tr_> = QQ[]
Bx = Matrix(QQab, 2, 2, [a0, a1, a1*deg_ + a0*tr_, -a0])
N = Matrix(QQab, 2, 2, [b0, b1, b1*deg_ + b0*tr_, -b0])
Mn = N * Bx
Mn
n = N.determinant()
assert Mn.determinant() == n * Bx.determinant() # trivial
Mn.determinant().factor()
c0, c1 = Mn[0][0], Mn[0][1]
(c0^2 + c0*c1*tr_ + c1^2*deg_).factor()
assert (c0^2 + c0*c1*tr_ + c1^2*deg_) == n * Bx.determinant()
d0, d1 = Mn[1][0], Mn[1][1]
(d0^2 + d0*d1*tr_ + d1^2*deg_).factor()
assert (d0^2 + d0*d1*tr_ + d1^2*deg_) == n * Bx.determinant() * deg_
```

B Other examples of families of embedded curves

Example 5 (GG with $d = 7$)**.** To highlight the case $\deg_\phi > 1$, we consider the GG22D7 curve where $d = 7$, $\mathrm{tr}_\phi = -1$ and $\deg_\phi = 2$, $q(x) = (x^{20} - x^{19} - x^{18} + 3x^{17} - x^{16} - 5x^{15} + 7x^{14} + 3x^{13} - 17x^{12} + 11x^{11} + 23x^{10} + 22x^9 - 68x^8 + 24x^7 + 112x^6 - 160x^5 - 64x^4 + 384x^3 - 256x^2 - 512x + 1024)/23$. The congruence constraints are $x = 32, 151 \bmod 161$. The eigenvalue $\lambda_\phi^2 + \lambda_\phi + 2 = 0$ is $\lambda_\phi = (x^{11} + 22)/23$ of degree 11 while $\deg q(x)/2 = 10$. A half-GCD gives $a_0 = x^{10} - 2x^8 + 2x^7 + 2x^6 - 6x^5 + 2x^4 + 10x^3 - 14x^2 - 6x + 34$, $a_1 = x^9 - x^8 - x^7 + 3x^6 - x^5 - 5x^4 + 7x^3 + 3x^2 - 17x + 11$ satisfying $a_0/a_1 = -\lambda_\phi \bmod q(x)$, $a_0^2 - a_0a_1 + 2a_1^2 = 23q(x)$. Define $B_x = \begin{bmatrix} a_0 & a_1 \\ 2a_1 - a_0 & -a_0 \end{bmatrix}$ whose determinant is $|\det B_x| = 23q(x)$. We want to remove the factor $n = 23$. for $x = 32 \bmod 161$, we have $a_0/a_1(161x + 32) = 10 \bmod 23$ and we deduce the matrix N to be reduced, made of the row vectors $(23, 0), (10, 1)$. We obtain

$$M_x = \frac{1}{23}\begin{bmatrix} 3 & -2 \\ 4 & 5 \end{bmatrix} B_x \ , \ |\det M_x| = q(x) \ .$$

The first row (a_0, a_1) of M_x satisfies $a_0^2 - a_0a_1 + 2a_1^2 = q(x)$, with

$$\begin{aligned} a_0 &= \left(5x^{10} - 4x^9 - 6x^8 + 14x^7 - 2x^6 - 26x^5 + 30x^4 + 22x^3 - 82x^2 + 38x + 126\right)/23, \\ a_1 &= \left(2x^{10} + 3x^9 - 7x^8 + x^7 + 13x^6 - 15x^5 - 11x^4 + 41x^3 - 19x^2 - 63x + 101\right)/23 \ . \end{aligned}$$

With $t'(x) = 2a_0 - a_1$, $y'(x) = a_1$, we have $4q(x) = t'^2(x) + 7y'^2(x)$.

For $x = 151 \bmod 161$, we have $a_0/a_1(161x + 151) = 18/21 = 14 \bmod 23$ and we deduce the matrix N to be reduced, made of the row vectors $(23, 0), (14, 1)$. We obtain

$$M_x = \frac{1}{23}\begin{bmatrix} -5 & -2 \\ 4 & -3 \end{bmatrix} B_x \ , \ |\det M_x| = q(x) \ .$$

The first row (a_0, a_1) of M_x satisfies $a_0^2 - a_0a_1 + 2a_1^2 = q(x)$, with

$$\begin{aligned} a_0 &= \left(-3x^{10} - 4x^9 + 10x^8 - 2x^7 - 18x^6 + 22x^5 + 14x^4 - 58x^3 + 30x^2 + 86x - 146\right)/23, \\ a_1 &= \left(2x^{10} - 5x^9 + x^8 + 9x^7 - 11x^6 - 7x^5 + 29x^4 - 15x^3 - 43x^2 + 73x + 13\right)/23 \ . \end{aligned}$$

With $t'(x) = 2a_0 - a_1$, $y'(x) = a_1$, we have $4q(x) = t'^2(x) + 7y'^2(x)$.

Example 6 (Scott–Guillevic $k = 54$ curve [SG18] with $D = -3$)**.** The $k = 54, D = -3$ Scott–Guillevic curves have $q(x) = 3^9x^{18} + 3^5x^9 + 1$ (there are 3 distinct families, of trace $3^5x^{10} + 1$, $-2 \cdot 3^5x^{10} - 3x + 1$, and $3^5x^{10} + 3x + 1$, and cofactor $3x^2 + 3x + 1$, $3x^2 + 1$, and $3x^2 - 3x + 1$ respectively). The eigenvalue is $\lambda_\phi = \sqrt{-3} = 243x^9 + 1 \bmod q(x)$. It satisfies $\lambda_\phi^2 + \lambda_\phi + 1 = 3q(x)$. We define B_x from λ_ϕ satisfying $|\det B_x| = 3q(x)$, and N from $n = 3$ and $\lambda_\phi = 1 \bmod 3$, $|\det N| = 3$, reduce N and obtain

$$M_x = \frac{1}{3}NB_x = \frac{1}{3}\begin{bmatrix} -1 & 1 \\ 1 & 2 \end{bmatrix}\begin{bmatrix} -\lambda_\phi & 1 \\ 1 + \lambda_\phi & \lambda_\phi \end{bmatrix} = \begin{bmatrix} 162x^9 + 1 & 81x^9 \\ 81x^9 + 1 & 162x^9 + 1 \end{bmatrix} \ .$$

Alternatively one can choose $(a_0, a_1) = (81x^9, -81x^9 - 1)$ satisfying $a_0^2 - a_0a_1 + a_1^2 = q(x)$. The six possible orders of embedded curves are given in Table 11. For $r(x) = 3^9x^{18} + 2 \cdot 3^5x^9 + 4$ to be prime, we require $x = 1 \bmod 2$. With the original SG54 curve of trace $t(x) = 3^5x^{10} + 1$, when the seed is odd, $p(x)$ is multiple of 4 and cannot be a prime.

With the choice of SG-54 curve having trace $t(x) = -2 \cdot 3^5x^{10} - 3x + 1$ and cofactor $c(x) = 3x^2 + 1$, The seed $x_0 = -18429$ provides a prime p of 300 bits, a prime q of 270 bits, and a prime $r = 3^9x_0^{18} + 2 \cdot 3^5x_0^9 + 4$ of 270 bits. The SG54 pairing-friendly curve has equation $E\colon y^2 = x^3 + 8$ defined over $\mathbb{F}_p$, of order cq, where $c = c(x_0) = 3x_0^2 + 1$. The

prime-order embedded curve is $E_1\colon y^2 = x^3 + 2$ defined over $\mathbb{F}_q$ and of prime order r. The 2-cycle curve is $E_0\colon y^2 = x^3 + 15$ defined over $\mathbb{F}_r$ of prime order q.

With the choice of SG-54 curve having trace $t(x) = 3^5x^{10} + 3x + 1$ and cofactor $c(x) = 3x^2 - 3x + 1$, The seed $x_0 = -1499617$ provides a prime p of 423 bits, a prime q of 384 bits, and a prime $r = 3^9x_0^{18} + 2 \cdot 3^5x_0^9 + 4$ of 384 bits. The SG54 pairing-friendly curve has equation $E\colon y^2 = x^3 - 4$ defined over $\mathbb{F}_p$, of order cq, where $c = c(x_0) = 3x_0^2 - 3x_0 + 1$. The prime-order embedded curve is $E_1\colon y^2 = x^3 + 11$ defined over $\mathbb{F}_q$ and of prime order r. The 2-cycle curve is $E_0\colon y^2 = x^3 + 22$ defined over $\mathbb{F}_r$ of prime order q.

Table 11: Embedded curves with SG54, $D = -3$. A first pair is $(t_0, y_0) = (243x^9 + 2, 81x^9)$ and the other pairs are for the quadratic, cubic and sextic twists.

(t', y') s.t. $q = (t'^2 + 3y'^2)/4$		$r = q + 1 - t'$	family
t_0, y_0	$3^5x^9 + 2, 3^4x^9$	3^9x^{18}	no
$-t_0, y_0$	$-3^5x^9 - 2, 3^4x^9$	$r = 3^9x^{18} + 2 \cdot 3^5x^9 + 4$	yes
$(t_0 + 3y_0)/2, (t_0 - y_0)/2$	$3^5x^9 + 1, 3^4x^9 + 1$	$(3x^2 + 1)(3x^2 \pm 3x + 1)(27x^6 \pm 9x^3 + 1)$	no
$-(t_0 + 3y_0)/2, (t_0 - y_0)/2$	$-3^5x^9 - 1, 3^4x^9 + 1$	$3(3^4x^9 + 1)^2$	no
$(t_0 - 3y_0)/2, (t_0 + y_0)/2$	$1, 2 \cdot 3^4x^9 + 1$	$q(x)$	no (anomalous)
$-(t_0 - 3y_0)/2, (t_0 + y_0)/2$	$-1, 2 \cdot 3^4x^9 + 1$	$3(3^8x^{18} + 3^4x^9 + 1)$	(yes, 3)