A short-list of pairing-friendly curves resistant to Special TNFS at the 128-bit security level

Aurore Guillevic

▶ **To cite this version:**

Aurore Guillevic. A short-list of pairing-friendly curves resistant to Special TNFS at the 128-bit security level. PKC 2020 - IACR International Conference on Practice and Theory of Public-Key Cryptography, Jun 2020, Edinburgh, United Kingdom. pp.535-564, 10.1007/978-3-030-45388-6_19 . hal-02396352v2

HAL Id: hal-02396352

https://hal.inria.fr/hal-02396352v2

Submitted on 5 Feb 2020

A short-list of pairing-friendly curves resistant to Special TNFS at the 128-bit security level[*]

Aurore Guillevic

Université de Lorraine, CNRS, Inria, LORIA, Nancy, France
`aurore.guillevic@inria.fr`

Abstract. There have been notable improvements in discrete logarithm computations in finite fields since 2015 and the introduction of the Tower Number Field Sieve algorithm (TNFS) for extension fields. The *Special* TNFS is very efficient in finite fields that are target groups of pairings on elliptic curves, where the characteristic is special (e.g. sparse). The key sizes for pairings should be increased, and alternative pairing-friendly curves can be considered. We revisit the Special variant of TNFS for pairing-friendly curves. In this case the characteristic is given by a polynomial of moderate degree (between 4 and 38) and tiny coefficients, evaluated at an integer (a seed). We present a polynomial selection with a new practical trade-off between degree and coefficient size. As a consequence, the security of curves computed by Barbulescu, El Mrabet and Ghammam in 2019 should be revised: we obtain a smaller estimated cost of STNFS for all curves except BLS12 and BN. To obtain TNFS-secure curves, we reconsider the Brezing–Weng generic construction of families of pairing-friendly curves and estimate the cost of our new Special TNFS algorithm for these curves. This improves on the work of Fotiadis and Konstantinou, Fotiadis and Martindale, and Barbulescu, El Mrabet and Ghammam. We obtain a short-list of interesting families of curves that are resistant to the Special TNFS algorithm, of embedding degrees 10 to 16 for the 128-bit security level. We conclude that at the 128-bit security level, BLS-12 and Fotiadis–Konstantinou–Martindale curves with $k = 12$ over a 440 to 448-bit prime field seem to be the best choice for pairing efficiency. We also give hints at the 192-bit security level.

1 Introduction

A cryptographic pairing is a bilinear non-degenerate map from two groups $\mathbb{G}_1$ and $\mathbb{G}_2$ to a target group $\mathbb{G}_T$, where the three groups share a common prime order r. The first two groups are distinct subgroups of the group of points $E(\mathbb{F}_{p^k})$ of an elliptic curve E defined over a prime field $\mathbb{F}_p$, and the third group is a multiplicative subgroup of order r of a finite field $\mathbb{F}_{p^k}$, where k is the minimal

---

integer such that $r \mid p^k - 1$, and is called the embedding degree. Pairing-friendly curves such that k is small (between 1 and 20 for example) should be designed on purpose, as the embedding degree is usually very large, of the magnitude of r.

Freeman, Scott and Teske presented a taxonomy of pairing-friendly curves in [20]. Until 2015, the size of the target finite field $\mathbb{F}_{p^k}$ was chosen to be the same as a prime field $\mathbb{F}_q$ offering the desired security, that is, a 3072-bit (or 3200-bit) finite field for a 128-bit security level. The size of $\mathbb{F}_q$ is deduced from the asymptotic complexity of the Number Field Sieve $L_p(1/3, c) = \exp((c + o(1))(\ln p)^{1/3}(\ln \ln p)^{2/3})$, where $c = (64/9)^{1/3} \approx 1.923$ for general prime fields and $c = (32/9)^{1/3} \approx 1.526$ for special primes having a very sparse representation. Barreto–Naehrig (BN) curves became very popular. A BN curve defined over a prime field has prime order and embedding degree 12, hence choosing p and r of 256 bits gives 128 bits of security on the curve, and p^k is about 3072-bit long, as desired to match the 128-bit security level in $\mathbb{F}_{p^k}$. But it turned out that prime fields and extension fields of the same total size q and p^k do not offer the same security. The state of affairs for extension fields is complicated, with many different cases.

In 2015 and 2016, Barbulescu, Gaudry and Kleinjung, followed by Kim and Barbulescu and Kim and Jeong [5,26,27] revisited Schirokauer's Tower Number Field Sieve algorithm (TNFS) and applied this new setting to finite fields of composite extension degrees. The asymptotic complexity of this new algorithm decreased significantly, from $L_Q(1/3, 2.201)$ to $L_Q(1/3, 1.526)$ and in particular, below the complexity of a generic DL computation in a prime field, in $L_Q(1/3, 1.923)$. This makes mandatory to revisit the sizes and choices of pairing-friendly curves.

Fotiadis and Konstantinou [17] revisited the Brezing–Weng method to generate families of pairing-friendly curves and identified a list of interesting choices of moderate embedding degrees to match the 128-bit security level. However, they considered the asymptotic complexity of STNFS to deduce the security offered by the curves. It gives a first hint on the sizes of finite fields to choose but is not precise enough. Later Menezes, Sarkar and Singh [29], then Barbulescu and Duquesne [3] and in 2019 Guillevic and Singh [23] refined the analysis of STNFS to obtain more precise sizes of finite fields to match a given security level. Fotiadis and Martindale [18] focused on composite embedding degrees $k \in \{8, 9, 10, 12\}$ for the 128-bit security level, Guillevic, Masson and Thomé [22] considered a modification of the Cocks–Pinch method for $k \in \{5, 6, 7, 8\}$, and Barbulescu, El Mrabet and Ghammam spanned embedding degrees from 9 to 53.

This is an active topic: the standardisation of pairings is under discussions at IETF [32] and at ISO for updating the standard on pairing-friendly curves [24]. Particular pairing-friendly curves (e.g. cycles of curves [12]) are also needed in zero-knowledge proofs and blockchains (ZCash uses a BLS12-381 curve [7,35], Ethereum a BN-256 curve [15], and Zexe a BLS12-377 curve and a Cocks–Pinch curve of embedding degree 6 [8, Table 16]).

Our Contributions

We introduce a practical variant of special polynomial selection for STNFS that applies to target finite fields of pairing-friendly curves. It does not change the asymptotic complexity of STNFS but it changes the estimated cost of STNFS as computed by Barbulescu and Duquesne.

We extend the work of Fotiadis and Konstantinou [17], and identify another criterion to be resistant to STNFS: the polynomial $p(x)$ defining the field characteristic should have no automorphism. Then we build on the work of Guillevic and Singh [23] to estimate finely the cost of a discrete logarithm computation with STNFS. We write a SageMath script to automatically and systematically compare many polynomial selections, and in particular, change of variables on $p(x)$. We consider embedding degrees from 9 to 17 at the 128-bit security level. This is a complement to the work of Fotiadis and Martindale [18], where embedding degrees 8, 9, 10 and 12 are considered at this security level. We also identify non-optimal parameter choices in the recent preprint of Barbulescu, El Mrabet and Ghammam [4], resulting in over-estimated cost of STNFS and under-estimated finite field size. We conclude with a short-list of STNFS-secure pairing-friendly curves of embedding degrees from 10 to 16. The source code of this work is available in Python/SageMath at

https://gitlab.inria.fr/tnfs-alpha/alpha/tree/master/sage

The work in [22] showed that a pairing-friendly curve over a non-special prime, and with a prime embedding degree $k = 5, 7$ gives a slow pairing computation, about three times slower than the best candidate: a BLS12-curve over a 446-bit prime field. Here we estimate that a curve of prime embedding degree $k = 11, 13$ with a special prime will not provide a competitive pairing computation, despite a smaller prime p, of 333 bits for $k = 11$ and 310 bits for $k = 13$, compared to a 446-bit prime p for BLS12 curves, but may provide a faster arithmetic in $\mathbb{G}_1$ (elliptic curve scalar multiplication over $\mathbb{F}_p$) thanks to a smaller finite field.

Organisation of the paper. In Section 2 we recall briefly the special tower number field sieve algorithm and the approximation of running-time made in [23]. We present our variant of special polynomial selection for pairing-friendly curves. In Section 3 we recall the Brezing–Weng construction for pairing-friendly curves, then we list the possible curves for the 128-bit security level, and we present the results of simulation of STNFS for each curve. We select a short-list of nine secure curves. In Section 4 we roughly estimate the cost of the Miller loop for an optimal ate pairing computation on the curves of the short-list that do not appear in previous works. In Section 5 we estimate the cost of STNFS for curves at the 192-bit security level for $k \in \{14, 15, 20, 21, 27, 28\}$. This is more complex than the 128-bit security level. We conclude in Section 6.

2   The Special Tower Number Field Sieve

In this section, we sketch the TNFS algorithm. We refer to [5,26,27,23] for an extended description of TNFS. The TNFS algorithm falls in the broader Number

Field Sieve algorithms. To compute a discrete logarithm in a finite field, one first computes a large amount of precomputed data. A first important ingredient is the *factor base*. A finite field $\mathbb{F}_{p^k}$ has no factorisation of elements into irreducible elements or prime elements. However a number field has a ring of integers, and factorisation of ideals in prime ideals. Equipped with a map from a (sub)ring of integers of a number field to a finite field, one can factor ideals in prime ideals, then map each prime factor to the finite field to obtain a factorisation in $\mathbb{F}_{p^k}$. (There are now well-defined and cheap procedures to handle non-principal ideals and non-torsion units). The factor base is made of the prime ideals (usually of degree one) of small norm, bounded by the *smoothness bound B*. The first step of the algorithm is defining two non-isomorphic number fields with two irreducible polynomials f and g, sharing a common irreducible factor ψ of degree k modulo p (a common root if one targets a prime field $\mathbb{F}_p$), so that one has two maps from the ring of integers of number fields defined by f and g, to the same finite field $\mathbb{F}_{p^k} = \mathbb{F}_p[x]/(\psi(x))$.

The next step is collecting a large number of relations involving the primes of the factor base. We will say that an algebraic integer is B-smooth if it factors in prime ideals of degree one and norm bounded by B (B is an integer). Once enough relations are collected, taking the logarithm of the multiplicative relations, one obtains a large set of linear equations whose unknowns are the discrete logarithms of the prime ideals of the factor base. Solving the system, one obtains the discrete logarithms of the factor base elements. Finally, to compute the discrete logarithm of a given target in the finite field, one lifts the target in the number field, and tries to find a smooth decomposition of this target over the prime ideals whose logarithms are known.

In the Number Field Sieve setting, two distinct number fields are needed, so that their ring of integers can be mapped to the finite field $\mathbb{F}_{p^k}$. In the Tower NFS setting, one consider two extensions of a same number field. Let k be the extension degree, and $k = \eta\kappa$ where η, κ are integers ($\eta = k$ and $\kappa = 1$ if k is prime). One chooses an irreducible monic polynomial $h(Y) \in \mathbb{Z}[Y]$, irreducible modulo p, of degree η and small coefficients. Define the number field $K_h = \mathbb{Q}[Y]/(h(Y))$, and let y denotes a root of h in K_h. Let $\mathcal{O}_h$ denotes the ring of integers of K_h, and let $\mathbb{Z}_y$ be a subring of $\mathcal{O}_h$ (we take the same notations as [23]). Let $\mathfrak{p} = (p, h(Y))$ be the unique prime ideal of $\mathcal{O}_h$ above p. One selects a pair of polynomials $f_y(X), g_y(X)$ so that reduced modulo $(p, h(Y))$, they share a common irreducible factor $\psi_y(X)$ of degree κ. Let $K_{y,f}$ and $K_{y,g}$ be the number fields defined above K_h by $f_y(X)$ and $g_y(X)$ respectively, and $\mathcal{O}_{y,f}$, $\mathcal{O}_{y,g}$ their ring of algebraic integers. Let $x_{y,f}$ be a root of $f_y(X)$ in $K_{y,f}$ and $x_{y,g}$ a root of $g_y(X)$ in $K_{y,g}$. We have the following setting (Figure 1) and commutative diagram (Figure 2).

In the relation collection step, one enumerates all $a(Y) = a_0 + a_1 Y + \ldots + a_{\eta-1} Y^{\eta-1}$, $b(Y) = b_0 + b_1 Y + \ldots + b_{\eta-1} Y^{\eta-1} \in \mathbb{Z}[Y]$ such that integers $|a_i|, |b_i|$ are bounded by the *relation collection bound A*. The aim is to compute the norms of $a(y) + b(y)x_{y,f}$ in $K_{y,f}$ and $a(y) + b(y)x_{y,g}$ in $K_{y,g}$ and store the pairs (a, b) whose norms are B-smooth. Assuming $h(Y), f_Y(X)$ are monic, the norm is

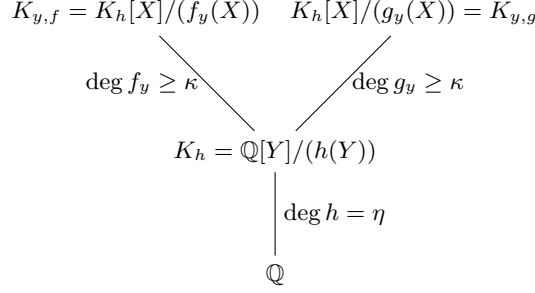$$N_f = \mathrm{Norm}_{K_{y,f}/\mathbb{Q}}(a(y) + b(y)x) = \mathrm{Res}_Y(\mathrm{Res}_X(a(Y) + b(Y)X, f_Y(X)), h(Y))$$

4

$$K_{y,f} = K_h[X]/(f_y(X)) \qquad K_h[X]/(g_y(X)) = K_{y,g}$$

$$\deg f_y \geq \kappa \qquad\qquad \deg g_y \geq \kappa$$

$$K_h = \mathbb{Q}[Y]/(h(Y))$$

$$\deg h = \eta$$

$$\mathbb{Q}$$

Fig. 1. Extensions of number field for TNFS

$$\mathbb{Z}_y[X]$$

$$\mathbb{Z}_y[X]/(f_y(X)) \qquad\qquad \mathbb{Z}_y[X]/(g_y(X))$$

$$\mathrm{mod}\ \mathfrak{p} \qquad\qquad \mathrm{mod}\ \mathfrak{p}$$
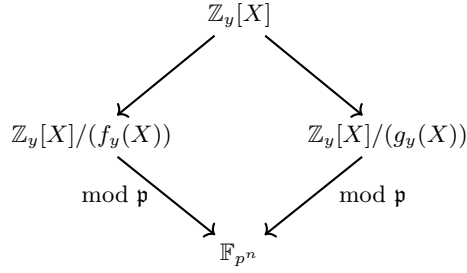
$$\mathbb{F}_{p^n}$$

Fig. 2. Commutative diagram for TNFS.

and for a non-monic $g_y(X)$ of leading coefficient $\mathrm{lc}(g_y)$,

$$N_g = \mathrm{Norm}_{K_{y,g}/\mathbb{Q}}(a(y) + b(y)x_{y,f})|\mathrm{lc}(g_y)|^n \qquad (1)$$
$$= \mathrm{Res}_Y(\mathrm{Res}_X(a(Y) + b(Y)X, g_Y(X)), h(Y)) \ . \qquad (2)$$

The schedule of TNFS can be summarised in four important steps.

1. Polynomial selection: choosing $h(Y), f_y(X), g_y(X)$ so as to minimise the integers N_f and N_g;
2. Relation collection: obtaining many $a(y) + b(y)X$ whose absolute norms in $K_{y,f}$ and $K_{y,g}$ w.r.t. $\mathbb{Q}$ are B-smooth. The coefficients a_i, b_i have absolute value bounded by A, where $a(y) = a_0 + a_1 y + \ldots + a_i y^{\eta-1}$, $b(y) = b_0 + b_1 y + \ldots + b_i y^{\eta-1}$;
3. Linear algebra: each relation encodes a row of a large sparse matrix. After a filtering step (preprocessing of the matrix to remove the singletons and small cliques) the right kernel is computed with the Block–Wiedemann algorithm;
4. Individual discrete logarithm computation: obtain the database of discrete logarithms of the prime ideals of factor base. Then given a target in $\mathbb{F}_{p^n}$, lift in one of the number fields, $K_{y,f}$ or $K_{y,g}$, and obtain a smooth decomposition. Sum the discrete logarithms of the factor base involved in the smooth decomposition to obtain the logarithm of the target.

2.1 Estimation of TNFS cost

This is an important concern to know the finite field size needed to match a security level such as 128 bits. Lenstra and Verheul designed an approach to extrapolate prime field sizes from the asymptotic complexity of NFS [28]. This complexity is $L_p(1/3, (64/9)^{1/3})$. First the unknown $o(1)$ in the formula is removed, in other words, it is set to zero. Then one considers a recent record computation, for example the 768-bit DL computation in 2017 for a prime p_{768} required the equivalent of 5300 core-years on Intel Xeon E5-2660 at 2.2GHz, i.e. about 2^{76} clock-cycles. Then one finds a factor δ s.t. $\delta \exp((64/9)^{1/3}(\log p_{768})^{1/3}(\log \log p_{768})^{2/3}) = 2^{76}$: this is $\delta \approx 8.2$. Finally, one extrapolates and finds b s.t. $\delta \exp((64/9)^{1/3}(\log 2^b)^{1/3}(\log \log 2^b)^{2/3}) = 2^{128}$, this is $b = 2940$. It means that a 2940-bit prime field $\mathbb{F}_p$ would offer 128 bits of security. In practice, a 3072-bit prime field is considered safe for the 128-bit security level.

Unfortunately, this extrapolation trick, which is already debatable for prime fields, cannot be applied straightforwardly for extension fields, as shown in [29]. There is no record computation available for scaling the formula, of the form $L_{p^k}(1/3, c)$, with c a constant for the variant of NFS. Moreover the best asymptotic complexities are met for very specific properties of η, κ (Fig. 1) in terms of p^n and this is [26, Table 4]. For Conjugation-TNFS to obtain $c = (48/9)^{1/3}$, one reads $\kappa = (\ln p^n/(12 \ln \ln p^n))^{1/3}$ for instance. For STNFS to obtain $c = (32/9)^{1/3}$, one needs p to be d-SNFS, that is $p = P(u)$ and P is a polynomial of degree d and very small coefficients, with $d = ((2/3)^{1/3} + o(1))(\ln p^n/(\ln \ln p^n))^{1/3}/\kappa$. But in practice, n is fixed to a small integer, for example $n = 12$, and p^n ranges (roughly) from 3072 to 6144 bits. In other words, the optimal case for the parameters n, η, κ, p^n is not necessarily met for a given p^n in practice.

The asymptotic formula for c, κ, η, p is obtained as follows. One chooses a polynomial selection method. The degree of h is η, its coefficient size is negligible, the degrees of f, g depend on κ and the coefficient sizes of f, g depend on η, κ, p^n. With the Conjugation method, we have $(\deg f, \deg g) = (2\kappa, \kappa)$ and $\|f\|_\infty = O(1)$, $\|g\|_\infty = O(\sqrt{p})$. The integers (resultants) of step 2 are bounded by a formula based on eq. (1) in $\eta, \kappa, \|f\|_\infty, \|g\|_\infty, \|h\|_\infty$. The Canfield–Erdös–Pomerance theorem states the proportion of B-smooth integers up to a bound N. Multiplying the proportion of B-smooth integers of Step 2 by the number of valid pairs $(a(y), b(y))$, in other words the volume of the relation collection (this is $\approx A^{2\kappa}$), one gets the expected total number of relations, at a cost dominated by the total number of pairs to proceed ($A^{2\kappa}$). One wants as many relations as possible, at a minimal cost. There are $\approx B/\log B$ primes up to B, and prime ideals of norm up to B. The square matrix has $\approx 2B/\log B$ columns, this is the number of prime ideals in the two factor bases. The linear algebra costs roughly B^2 operations. Then one balances the costs of Step 2 and Step 3 by setting them equal. Then one minimises the cost, obtaining a formula for the parameters A, B in terms of the inputs η, κ, p^n, and this is [26, Tab. 4].

Menezes, Sarkar and Singh observed that the bound on the size of the norms is not tight [29]. There are combinatorial factors in $\eta, \kappa, \|h\|_\infty$ that are removed in the asymptotic estimate as they disappear in the $o(1)$, and the size of coefficients

of h are assumed to tend to 1, but in practice this is not the case. To circumvent this theoretical limitation, Menezes, Sarkar and Singh reconsidered the steps of the algorithm. They bounded the size of norms for a given input (denote the bound N), and optimised the parameters A, B, but for fixed n, η, κ. They used the first-order term of the Canfield–Erdös–Pomerance asymptotic formula to estimate the proportion of B-smooth integers of size up to N, that is, u^{-u} where $u = \log N / \log B$. Later in [3] Barbulescu and Duquesne averaged the size of norms over a sample of about 26000 random inputs (a, b). They replaced the Canfield–Erdös–Pomerance asymptotic formula by the Dickman-ρ function to compute the B-smoothness probability of random integers of bounded size. Then Guillevic and Singh [23] computed the smoothness bias of the resultants with respect to integers of the same size (α value of polynomials), simulated the relation collection of TNFS, and averaged the smoothness probability over random samples, as a TNFS variant of the Murphy E function. This estimate should be done for each set of parameters $(p(x), u, A, B)$. Theory meets practice at this edge: the Murphy-E function was first designed to rank the yield of pairs of polynomials for a given prime p, smoothness bound B and sieving area A^2, and is based on the Dickman-ρ function. It runs as a brute-force search of promising pairs of polynomials in Step 1, for record computations.

We build on these two previous works [3,23]. In particular, we model the relation collection cost as [23, Eq. 6.3] and the linear algebra cost as [23, Eq. 6.5].

$$\text{Cost of relation collection} = \frac{(2A+1)^{2\cdot\eta} \cdot \log(\log(B))}{2 \cdot (\#\operatorname{aut}(h)\gcd(\deg(f), \deg(g)))} \qquad (3)$$

where A is the bound on the coefficients a_i, b_i in the relation collection. The $a(y) = a_0 + a_1 y + \ldots + a_{\eta-1} y^{\eta-1}$ and $b(y) = b_0 + \ldots + b_{\eta-1} y^{\eta-1}$ have coefficients a_i, b_i in $[-A, A]$, and $b_{\eta-1} \geq 0$. There are $(2A+1)^{2\eta}/2$ such pairs $(a(y), b(y))$. For each pair, one computes the norms N_f, N_g and test for B-smoothness, this is estimated as costing $\log\log B$. The process can be faster for specific choices of h, f_y, g_y where automorphisms are available, hence the denominator. We also model

$$\text{Cost of Linear Algebra} = \texttt{cnst} \cdot \texttt{wt} \cdot (\#\mathcal{B} \div \texttt{flt})^2 \qquad (4)$$

where $\texttt{cnst}$ is a constant representing the cost of a multiplication modulo ℓ, $\texttt{wt}$ is the weight per row (number of non-zero entries), $\#\mathcal{B}$ is the total size of the factor base (f-side and g-side), and $\texttt{flt}$ is the reducing factor of the filtering step. Following [23], $\texttt{cnst} = \lfloor \ell/64 \rfloor$ is the machine-word size of ℓ, $\texttt{wt} = 200$ and $\texttt{flt} = 20$.

Remark 1. The arbitrary choice $\texttt{wt} = 200$ and $\texttt{flt} = 20$ is not satisfying, in particular for high security levels. The two parameters would need to increase slowly with the size of inputs. Barbulescu and Duquesne set an upper bound $\texttt{flt} = \log_2 B$ [3, Conjecture 1], but compared to recent record computations made with $\texttt{cado-nfs}$, it is a bit too much. More work is needed to solve this issue.

For each pairing-friendly curve parameters $(p(x), u)$ we run Algorithm 2.1 from [23, Alg. 6.1] to estimate the number of relations obtained for given inputs A, B. The Dickman-ρ function is denoted by D_ρ. We write a SageMath code to automatically adjust the parameters A, B so that enough relations are obtained and the cost of linear algebra and relation collection are finely balanced, in order to minimise the total estimated cost of TNFS.

Algorithm 2.1: Monte-Carlo approximation of Murphy's E for TNFS [23, Alg. 6.1] (computes an estimation of the number of relations)

Input: Polynomials $f_y, g_y, h, \alpha_f, \alpha_g$, parameter $A \in \mathbb{N}$, smoothness bound B, $N \approx 10^5$
Output: Yield estimate (number of relations)

1 $P_{fg} \leftarrow 0$
2 **for** $n := 1$ *to* N **do**
3      $(a_0, \ldots, a_{\eta-1}) \leftarrow$ random tuple in $\{-A, A\}^{2 \deg h}$
4      $(b_0, \ldots, b_{\eta-1}) \leftarrow$ random tuple in $\{-A, A\}^{2 \deg h - 1} \times \{0, A\}$
5   **if** $\gcd(a_0, \ldots, a_{\eta-1}, b_0, \ldots, b_{\eta-1}) \neq 1$ **then**
6   continue
7      $a(Y) \leftarrow \sum_{i=0}^{\eta-1} a_i Y^i$; $\mathfrak{a} \leftarrow a(y)\mathcal{O}_h$; $b(Y) \leftarrow \sum_{i=0}^{\eta-1} b_i Y^i$; $\mathfrak{b} \leftarrow b(y)\mathcal{O}_h$
8   **if** *the ideals* $\mathfrak{a}, \mathfrak{b}$ *are not coprime* ($\mathfrak{a} + \mathfrak{b} \neq 1$) **then**
9   continue
10     $N_f \leftarrow |\operatorname{Res}(h, \operatorname{Res}(f_y, a(Y) - b(Y)X))|$
11     $N_g \leftarrow |\operatorname{Res}(h, \operatorname{Res}(g_y, a(Y) - b(Y)X))|$
12     $u_f \leftarrow (\ln N_f + \alpha_f)/\ln B$; $p_f \leftarrow D_\rho(u_f) + (1 - \gamma)D_\rho(u - 1)/\ln N_f$
13     $u_g \leftarrow (\ln N_g + \alpha_g)/\ln B$; $p_g \leftarrow D_\rho(u_g) + (1 - \gamma)D_\rho(u - 1)/\ln N_g$
14     $P_{fg} \leftarrow P_{fg} + p_f p_g$
15 $P_{fg} \leftarrow P_{fg}/N$
16 $w \leftarrow$ index of group of torsion units of $\mathcal{O}_h$
17 $V \leftarrow (2A + 1)^{2 \deg h}/(2w\zeta_{K_h}(2))$
18 **return** $V \times P_{fg}$

2.2 Special Polynomial Selection

We refine the special polynomial selection introduced in [5] and present a variant particularly suited for certain families of pairing-friendly curves that appear in the recent preprint [4].

Pairing-friendly curves have a special characteristic p, given by a polynomial $p(x)$ of small degree evaluated at an integer u. For BLS12 curves, we have $p(x) = (x^6 - 2x^5 + 2x^3 + x + 1)/3$, and for a 381-bit prime p, $u = -(2^{63} + 2^{62} + 2^{60} + 2^{57} + 2^{48} + 2^{16})$ [7]. Joux and Pierrot introduced a dedicated polynomial selection that takes advantage of the polynomial form $p = p(u)$ [25]. The adaptation to the Tower setting is the following.

Joux–Pierrot polynomial selection for TNFS. Assume there exists an integer $u \approx p^{1/d}$ and a polynomial $P(U)$ of degree d and small coefficients

$\|P(U)\|_\infty = O(1)$, such that $P(u) = 0 \bmod p$. Select a monic polynomial $S_y(X)$ of degree κ and small coefficients $\|S_y(X)\|_\infty = O(1)$, such that $g_y(X) = S_y(X) - u$ and $f_y(X) = P(S_y(X))$ are irreducible. Finally select a monic irreducible $h(Y)$. Then $(h(Y), f_y(X), g_y(X))$ are STNFS polynomials.

Joux–Pierrot polynomial selection for TNFS with automorphism. We recall a variant of the Joux–Pierrot method to obtain a pair of polynomials (f_y, g_y) admitting an automorphism, when k is not prime. First select an auxiliary polynomial with automorphism, for example from the list in [16].

- $\kappa = 2$: $c_t(X) = X^2 - tX + 1$, $\sigma : X \mapsto 1/X$; $c_t(X) = X^2 + t$, $\sigma : X \mapsto -X$;
- $\kappa = 3$: $c_t(X) = X^3 - tX^2 - (t+3)X - 1$, $\sigma : X \mapsto -(X+1)/X$;
- $\kappa = 4$: $c_t(X) = X^4 - tX^3 - 6X^2 + tX + 1$, $\sigma : X \mapsto -(X+1)/(X-1)$;
- $\kappa = 6$: $c_t(X) = X^6 - 2tX^5 - (5t+15)X^4 - 20X^3 + 5tX^2 + (2t+6)X + 1$, $\sigma : X \mapsto -(2X+1)/(X-1)$.

If $\gcd(\kappa, \eta) = 1$, define $f_y(X) = \mathrm{Res}_U(c_U(X), P(U))$ and $g_y(X) = c_u(X)$. If $\gcd(\kappa, \eta) > 1$, define $f_y(X) = \mathrm{Res}_U(c_{Uy}(X), P(U))$ and $g_y(X) = c_{uy}(X)$, or alternatively, $f_y(X) = \mathrm{Res}_U(c_{U+y}(X), P(U))$ and $g_y(X) = c_{u+y}(X)$. If f_y, g_y are irreducible, select a monic irreducible $h(Y)$. Then $(h(Y), f_y(X), g_y(X))$ are STNFS polynomials.

Example 1 ([23, Table 7]). To minimise the size of norms and the total estimated cost of STNFS for BLS12-381 curves, one chooses h of degree 6, and f_y, g_y share a common irreducible factor of degree 2 modulo $(p, h(Y))$. The prime p of BLS12 curves satisfies $p = P(u)/3$, where $P(x) = x^6 - 2x^5 + 2x^3 + x + 1$. The polynomials selected in [23, Table 7] are $h = Y^6 - Y^2 + 1$, $f_y = \mathrm{Res}_U(P(U), X^2 - UY) = X^{12} - 2yX^{10} + 2y^3X^6 + y^5X^2 + y^2 - 1 \bmod h(Y)$ and $g_y = X^2 - uy = X^2 + 15132376222941642752y$.

Improvements on the Joux–Pierrot method. The pairing-friendly curves of Section 3 are defined over prime fields whose characteristic has a polynomial form $p = p(u)$ for an integer seed u, where $p(x)$ has very small coefficients and degree from 4 (BN curves) to 46 (Construction 6.7 for $k = 9$, Table 2). We observed that when the degree of $p(x)$ is larger than 12, the average size of norms obtained with Algorithm 2.1 is not satisfying. In other words, for a same size of finite field $\mathbb{F}_{p^k}$ but different families of curves with $p(x)$ of very different degrees, one obtain very different estimated costs of STNFS. We explain in the following our method to obtain a lower estimated cost of STNFS when the degree of $p(x)$ is too large and the Joux–Pierrot method does not give good enough results.

In [5, §5.2] and in the SageMath script provided with [4], one observes that when it is possible, the degree of the polynomial P is divided by two without increasing the size of the coefficients. We name it Variant 1.

Variant 1 (Even polynomial $p(x)$) *When $p(x)$ is an even polynomial (that is, with only even degree monomials, and one has $p(x) = p(-x)$), then one*

defines $P(x)$ such that $P(x^2) = p(x)$, and P has degree $\deg(p(x))/2$. The pair of polynomials (for TNFS) $(P(x), x - u^2)$ satisfies $\mathrm{Res}_x(P(x), x - u^2) = P(u^2) = p(u) = p$ as desired.

We adapt this technique to palindrome polynomials (also mentioned in [5, §5.2]).

Variant 2 (Palindrome polynomial $p(x)$) *When $p(x) = p(1/x)x^{\deg p(x)}$, then we define $P(x)$ to be the minimal polynomial of $\alpha + 1/\alpha$ in the number field defined by $p(x)$, $K = \mathbb{Q}[x]/(p(x)) = \mathbb{Q}(\alpha)$. Then $P(x)$ has degree $\deg(p(x))/2$ and small coefficients (as long as $p(x)$ has small coefficients). The pair of polynomials (for TNFS) is $(P(x), ux - (u^2 + 1))$, and $\mathrm{Res}_x(P(x), ux - (u^2 + 1)) = u^{\deg P} P(u + 1/u) \equiv 0 \bmod p(u)$ as desired.*

Variant 3 (Polynomial $p(x)$ with automorphism) *More generally when there is an automorphism available for $p(x)$, say σ, of order two i.e. $\sigma^2(a) = a$, then we define $P(x)$ to be the minimal polynomial of $a + \sigma(a)$ (the trace of the automorphism is invariant). Then $P(x)$ has degree $\deg(p(x))/2$ and small coefficients (as long as $p(x)$ has small coefficients). The second polynomial for TNFS is $x - (u + \sigma(u))$. If $a + \sigma(a)$ does not have a good expression (a fraction of linear polynomials in a), then one computes a half-extended GCD of $p(x)$ and $x + \sigma(x)$ to obtain $x + \sigma(x) = s_1(x)/s_2(x)$. If the degrees of s_1 and s_2 are small, one can define $s_2(u)x - s_1(x)$ as the second polynomial for NFS. We have $\mathrm{Res}_x(P(x), s_2(u)x - s_1(u)) = s_2^{\deg P}(u)P(u + \sigma(u)) \equiv 0 \bmod p(u)$.*

These three variants already allow more possibility of trade-off between f and g in terms of degrees and coefficient size: one divides the degree of f by two and increases the coefficient size of g by a factor two ($\|g_y\|_\infty \approx u^2$ instead of u).

Variant 4 *When $p(x)$ has tiny coefficients and a high degree, it might be worth doing the following transformation, knowing the seed u. Write $p(x) = \sum_{i=0}^{d} p_i x^i$ where $d = \deg p(x)$ and p_i are tiny integer coefficients. Then for an integer l in the range $2 \le l \le d/2$, define*

$$P(x) = \sum_{i=0}^{d} p_i u^{i \bmod l} x^{\lfloor i/l \rfloor} \ .$$

Then $P(x)$ has degree $\lfloor d/l \rfloor$ (the floor integer of the number d/l) and coefficients at most u^{l-1}, and $P(u^l) = p(u)$. The pair of polynomials (for TNFS) is $(P(x), x - u^l)$, and $\mathrm{Res}_x(P(x), x - u^l) = P(u^l) \equiv 0 \bmod p(u)$ as desired.

This is possible to combine Variant 4 with one of Variants 1, 2 or 3. With these alternative pairs of polynomials, we can have more balanced size of norms, hence a higher smoothness probability, and a lower DL cost estimation. Our results are given in the right-most column of Table 3 page 18. It has direct impact on many curves of embedding degrees 9, 10, 11, 13, 14, 17, in particular, the curves whose polynomial $p(x)$ has a high degree.

Example 2. Let us consider a curve of embedding degree $k = 13$, discriminant $D = 3$, following Construction 6.6. The polynomial defining the characteristic is $p(x) = (x^{28}+x^{27}+x^{26}+x^{15}-2x^{14}+x^{13}+x^2-2x+1)/3$. It has no automorphism. We define $P(x) = (u+1)x^9 + u^2 x^8 + x^5 + u(1-2u)x^4 + u^2 - 2u + 1$ such that $P(u^3) = 3p(u)$, and u is a seed for a particular curve. A degree 13 irreducible polynomial $h(Y)$ and the pair $(f, g) = (P(x), x - u^3)$ can be used for polynomial selection with STNFS.

Example 3. Consider a curve of embedding degree 17, named Construction 6.6 in Section 3. It has $p(x) = (x^{36} + x^{35} + x^{34} + x^{19} + 4x^{18} + x^{17} + x^2 + x + 1)/3$ and automorphism $\sigma : x \mapsto 1/x$. Variant 2 gives $P(x) = x^{18} + x^{17} - 17x^{16} - 17x^{15} + 119x^{14} + 119x^{13} - 442x^{12} - 442x^{11} + 935x^{10} + 935x^9 - 1122x^8 - 1122x^7 + 714x^6 + 714x^5 - 204x^4 - 204x^3 + 17x^2 + 18x + 4$ such that $P(x+1/x)x^{18} = 3p(x)$. Applying Variant 4, we obtain $P(x) = u(x^9 + (v-17)x^8 - (17v - 119)x^7 + (119v - 442)x^6 - (442v - 935)x^5 + (935v - 1122)x^4 - (1122v - 714)x^3 + (714v - 204)x^2 - (204v - 17)x + 18v + 4)$, where $v = (u + 1/u) = (u^2 + 1)/u$ (we multiply by u to get integer coefficients). The pair $(P(x), u^2 x - (u^2 + 1)^2)$ can be used for STNFS. Since $\deg p(x) = 36$, the seed u will be very small, and the coefficients of P in u^2 are small.

3 Complete Families of Pairing-Friendly Curves

We will apply our new special polynomial selection to paring-friendly curves whose parameters are given by polynomials, such as BN and BLS12 curves. We recall the generic Brezing–Weng construction of families of pairing-friendly curves. A family will be encoded by three parameters: the embedding degree k, the discriminant D, and a choice e_0 to compute the trace. It allows to capture all cyclotomic constructions of pairing-friendly curves with three parameters. The BN curves, KSS curves [20, §6.2] and Fotiadis–Konstantinou and Fotiadis–Martindale curves [17,18] do not fall in the cyclotomic framework because $r(x)$ is not a cyclotomic polynomial.

3.1 Brezing–Weng constructions of pairing-friendly curves

A set of the complete families presented in the Freeman, Scott and Teske paper [20] are special instances of the generic Brezing–Weng construction [9] that we recall in Algorithm 3.1. In this framework, $r(x)$ is chosen to be a cyclotomic polynomial, and we name it a cyclotomic construction. For BN curves, $r(x)$ is one factor of an Aurifeuillean factorisation of a cyclotomic polynomial. For KSS curves, $r(x)$ is a minimal polynomial of an algebraic element of a cyclotomic field. Freeman, Scott and Teske [20] obtain complete families that correspond to specific choices of trace in Algorithm 3.1. We recall the BLS construction [6], with $D = 3$ in Table 1. The construction is generalised in [20] as Construction 6.6, and gives polynomial families for any k such that $18 \nmid k$, and $D = 3$. Constructions 6.6 and BLS give the same polynomials for $k = 24$, for other embedding degrees, only

---

Algorithm 3.1: CYCLO(k, D, e_0) – Cyclotomic construction of pairing-friendly curves

1 **if** $D = 1$ **then** $m \leftarrow 4/\gcd(4, k)$
2 **else if** $D = 2$ **then** $m \leftarrow 8/\gcd(8, k)$
3 **else if** $D = 3$ **then** $m \leftarrow 3/\gcd(3, k)$
4 **else** $m \leftarrow 1$
5 $r_x \leftarrow \Phi_{km}(x)$; $K \leftarrow \mathbb{Q}[x]/(r(x))$; $\zeta_{km} \leftarrow$ a root of r_x in K
6 **if** $-D$ *is not a square mod* r_x **then return** $\bot$
7 **if** $\gcd(e_0, k) \neq 1$ **then return** $\bot$
8 $t_x \leftarrow x^{me_0} + 1 \bmod r_x$
9 $y_x \leftarrow$ a polynomial in x mapping to $((t_x(\zeta_{km}) - 2)\sqrt{-D}/D)$ in K
10 $p_x \leftarrow (t_x^2 + Dy_x^2)/4$
11 **if** p_x *is not irreducible* **then return** $\bot$
12 **if** p_x *does not represent primes* **then return** $\bot$
13 **return** (p_x, r_x, t_x, y_x, D)

---

the ρ-value is the same. The BLS construction gives a very simple Miller loop of ate pairing, of length $x = t - 1$ (without extra Frobenius and line computation), which is optimal. Constructions 6.2, 6.3 and 6.4 in [20] are polynomial families with $D = 1$ and $k = 1 \bmod 2$, $k = 2 \bmod 4$ and $k = 4 \bmod 8$ respectively. We report the construction number from [20] in Table 2.

Unfortunately, [4] does not consider cyclotomic methods with small discriminants other than 1, 2 and 3. In [9, p.137], Brezing and Weng give alternatives such as $D = 5$ for $k = 10$. Recently, Fotiadis and Konstantinou used the Brezing–Weng method with small discriminants D to generate other pairing-friendly curves whose ρ-value is slightly larger but that are more resistant to TNFS [17]. For $k = 10$, Fotiadis and Konstantinou list alternatives with $D = 5$ and $D = 15$, for $k = 11$, with $D = 11$, for $k = 13$, with $D = 13$. For smaller embedding degrees, between 5 and 8, the ρ value is larger than 2. We refer to [22] for TNFS-resistant curves in this case with a modification of the Cocks–Pinch method.

3.2   Reducing the possibilities

For BLS12 and BN curves, the finite field size identified as secure for 128 bits of security is about $12 \times 448 = 5376$. The arithmetic on these curves is already very well optimised. Hence we decided to reduce the investigation of other families of curves to those where p^k is smaller than 5376 bits.

The minimum size of r is 256 bits to ensure the security on the curve, and the size of p is given by the ρ-value defined as the ratio between the degree of $p(x)$ and $r(x)$. We choose the sharp constraint (at the 128-bit security level)

$$3072 \leq 256\rho k \leq 5376 \tag{5}$$

to reduce the number of families to consider. If $\rho = 1$ we obtain the upper bound $k \leq 21$, and if $\rho = 2$ then we obtain the lower bound $k \geq 6$. We obtain candidates with $9 \leq k \leq 17$, in Table 2 page 15.

Table 1. Polynomials of the BLS families for $k = 3^i$ and $k = 2^i \cdot 3$ (for example $k \in \{6, 9, 12, 24, 27, 48\}$). In practice, it is very popular for $k = 12$.

$$k = 3^i$$
$$r(x) = \Phi_{3^i}(x)/3 = (x^{2 \cdot 3^{i-1}} + x^{3^{i-1}} + 1)/3$$
$$t(x) = x + 1$$
$$c(x) = (x - 1)^2$$
$$y(x) = (x - 1)(2x^{3^{i-1}} + 1)/3$$
$$p(x) = (t^2(x) + 3y^2(x))/4 = (x^2 + x + 1 + (x - 1)^2 x^{3^{i-1}}(x^{3^{i-1}} + 1))/3$$

$$k = 2^i \cdot 3$$
$$r(x) = \Phi_{2^i \cdot 3}(x) = (x^{2^i} - x^{2^{i-1}} + 1)$$
$$t(x) = x + 1$$
$$c(x) = (x - 1)^2/3$$
$$y(x) = (x - 1)(2x^{2^{i-1}} - 1)/3$$
$$p(x) = (t^2(x) + 3y^2(x))/4 = (x^2 + x + 1 + (x - 1)^2 x^{2^{i-1}}(x^{2^{i-1}} - 1))/3$$

Small embedding degrees up to 8. Embedding degree 1 is considered in [10]. Embedding degrees 2 and 3 are obtained with supersingular curves [21, § IX.13 p.204]. Embedding degrees 3, 4, and 6 are obtained with MNT curves. Embedding degrees 5 to 8 were compared in [22]. We focus on embedding degrees 9 to 17 for the 128-bit security level.

Embedding degree 9. There are three families of pairing-friendly curves of embedding degree $k = 9$, discriminant $D = 3$ and $\rho = 4/3$. We focus on $D = 3$ to have a twist of order three since $3 \mid k$. Alternatives are $D = 1$ and $\rho = 11/6$, $D = 2$ and $\rho = 23/12$. Another family with $D = 3$ is given in [33, §4.4] from the Aurifeuillean factorisation of $\Phi_9(-3x^2)$.

Embedding degree 10. We will consider three additional families for $k = 10$: with $D = 1$ and trace $t = x^{18} + 1 \bmod r(x)$ (in [9, p.137] and [20, Construction 6.5]), with $D = 5$, $r(x) = \Phi_{20}(x)$ and $t = x^{18} + 1 \bmod r(x)$ ([17, Table 2 and Example 5]), and with $D = 15$, $r(x) = \Phi_{30}(x)$ and $t(x) = x^3 + 1$ [17, Table 2]. With $D = 3$, no cyclotomic construction is valid, we consider the $\rho = 2$ option in [4]. With $D = 2$, the construction is not interesting: the polynomial $p(x)$ has degree 30 and the choices of seeds u are very limited. There were no choice of u to get a pair of primes (p, r) such that r is 256-bit long or more, and p is at most 512-bit long.

Embedding degree 11. With $D = 1$, $r(x) = \Phi_{44}(x)$ and $t(x) = x^{24} + 1 \bmod r(x)$, the family has $\rho = 1.3$, this is Construction 6.2 in [4]. The other possibilities of $t(x) = x^{4e_0} + 1 \bmod r(x)$ are $e_0 \in \{1, 2, 7\}$. We discard $e_0 = 2$ since no seed u was found so that $p^k \leq 5376$. With $D = 3$, $e_0 = 4$ is Construction 6.6, and $e_0 \in \{8, 1\}$ gives two other valid families of curves. With $D = 11$, we obtain two families of curves with $e_0 \in \{4, 8\}$ ($e_0 = 8$ appears in [17, Table 4]).

Embedding degree 12. For embedding degree 12, we concentrate on $D = 3$ to maximise the twist. The BLS12 and BN curves are the most popular curves of embedding degree 12, and recently Fotiadis and Martindale highlighted a competitive Fotiadis–Konstantinou (FK) curve of embedding degree 12 and discriminant $D = 3$ [17,18]. Curves of discriminant $D = 1$ have a twist of degree 4. Construction 6.4 from [20] produces a family with $\rho = 2$, the size of p is not suited. Applying the Brezing–Weng method, we do not obtain other families ($p(x)$ does not produce primes). With $D = 2$ there is one family of curves and $\rho = 7/4$. Note that in this case, only a quadratic twist is available, the pairing computation will be slower compared to BLS12 curves with $D = 3$ and sextic twists.

Embedding degree 13. Since -13 is not a square in $\mathbb{Q}(\zeta_{13})$, we concentrate on $D = 1$ with $r(x) = \Phi_{4 \times 13}(x)$ and $D = 3$ with $r(x) = \Phi_{3 \times 13}(x)$. For $D = 1$, the trace is $x^{4e_0} + 1$ where $e_0 \in \{1, 7\}$ give valid families of curves, and $e_0 = 7$ corresponds to Construction 6.2. For $D = 3$, the trace is $t(x) = x^{3e_0} + 1$ and $e_0 = 9$ corresponds to Construction 6.6. We also consider $e_0 \in \{1, 2, 10\}$.

Embedding degree 14. We concentrate on Construction 6.3 and 6.6. The other choices of e_0 in the Brezing–Weng construction do not produce families of curves satisfying the bounds on the size of p^k. In particular, $D = -7$ produces an alternative family whose ρ-value is too large.

Embedding degree 17. In addition to Construction 6.2 and 6.6, we consider $D = 3$ and trace $t(x) = x^{3 \times 12} + 1 \mod r(x)$ where $r(x) = \Phi_{3 \times 17}$. Actually because of the very large degree of $p(x)$ (36 and 38), it was not possible to find a seed u so that p^k is smaller than 5376 bits. However for a comparison to [4], we include the three families of curves in our security estimate.

Other embedding degrees. For embedding degree 16 we take the KSS-16 curves, these are (6.11) in [20]. Embedding degrees 15 and above 17 do not satisfy the conditions (5), however we include $k = 15$ to compare to [4].

For $9 \le k \le 17$, we list in Table 2 the available families satisfying Equation (5). Moreover we will later restrict to $D = 3$ when $3 \mid k$ and $D = 1$ when $4 \mid k$ to ensure the higher degree of twist.

3.3 Security estimate for the finite field

The next step is to determine the size of the finite field $\mathbb{F}_{p^k}$ to ensure the required security w.r.t. a DL computation with any variant of the NFS algorithm.

Refinement of Barbulescu–El Mrabet–Ghammam results. In the preprint [4], Barbulescu, El Mrabet and Ghammam presented a consequent list of pairing-friendly curves of embedding degrees 6 to 53 for the three common security levels of 128, 192 and 256 bits. There were about 150 distinct curves. We compare the curves of [4] that are listed in Table 2.

Table 2. Pairing-friendly Constructions for $9 \le k \le 17$ such that $3072 \le 256\rho k \le 5376$. Parameters m and e_0 match Algorithm 3.1. The value 256ρ is an approximation of the minimal bit-size of p required to ensure r to be of 256 bits, so that the curve $E(\mathbb{F}_p)$ offers 128 bits of security. We include $k = 12$ Construction 6.4, and $k = 15$ although $256\rho k$ is too large, for they are refereed in Tables 3 and 4.

k	Construction	D	m	e_0	ρ	deg $p(x)$	$\sigma_p(x)$	$\lceil 256\rho \rceil$	$\lceil 256\rho k \rceil$
9	Cyclo (BLS)	3	1	1	$1.33 = 4/3$	8	$\frac{x^4-x^3-1}{x^2+x}$	342	3072
9	Cyclo	3	1	4	$1.33 = 4/3$	8	$1/x$	342	3072
9	Cyclo (6.6)	3	1	7	$1.33 = 4/3$	8	$\frac{x^4+x^3+x^2+x-1}{1-x^2}$	342	3072
9	Cyclo (6.2)	1	4	5	$1.83 = 11/6$	22	$-x$	470	4224
9	Cyclo (6.7)	2	8	1	$1.92 = 23/12$	46	$-x$	491	4416
9	Cyclo (FM10)	3	1	5	$2.00 = 2$	12	Id	512	4608
10	Cyclo (6.5)	1	2	9	$1.50 = 3/2$	12	$-x$	384	3840
10	Cyclo (6.3) (FM13)	1	2	1	$1.75 = 7/4$	14	$-x$	448	4480
10	Cyclo (FM16)	2	4	9	$1.88 = 15/8$	30	$-x$	480	4800
10	(Cyclo) 6.6	3	3	1	$2.00 = 2$	16	Id	512	5120
10	Cyclo (FM14)	5	2	9	$1.75 = 7/4$	14	$-x$	448	4480
10	Cyclo (FM15)	15	3	1	$1.75 = 7/4$	14	Id	448	4480
11	Cyclo (6.2)	1	4	6	$1.30 = 13/10$	26	$-x$	333	3661
11	Cyclo	1	4	1	$1.50 = 3/2$	30	$-x$	384	4224
11	Cyclo	1	4	7	$1.70 = 17/10$	34	$-x$	436	4788
11	Cyclo (6.6)	3	3	4	$1.20 = 6/5$	24	$1/x$	308	3380
11	Cyclo	3	3	8	$1.30 = 13/10$	26	Id	333	3661
11	Cyclo	3	3	1	$1.40 = 7/5$	28	Id	359	3943
11	Cyclo	11	1	4	$1.60 = 8/5$	16	Id	410	4506
11	Cyclo	11	1	8	$1.60 = 8/5$	16	$1/x$	410	4506
12	BN (6.8)	3	1	1	$1.00 = 1$	4	$1/(6x)$	256	3072
12	Cyclo (BLS)	3	1	1	$1.50 = 3/2$	6	Id	384	4608
12	FK12 (FM17)	3	1	–	$1.50 = 3/2$	6	Id	384	4608
12	FM19	3	1	–	$1.50 = 3/2$	6	Id	384	4608
12	FM20	3	1	–	$1.50 = 3/2$	6	Id	384	4608
12	Cyclo (6.7) (FM18)	2	2	1	$1.75 = 7/4$	14	$-x$	448	5376
12	(Cyclo) 6.4	1	1	1	$2.00 = 2$	8	$-1/x$	512	*6144*
13	Cyclo (6.2)	1	4	7	$1.25 = 5/4$	30	$-x$	320	4160
13	Cyclo	1	4	1	$1.42 = 17/12$	34	$-x$	363	4715
13	Cyclo (6.6)	3	3	9	$1.17 = 7/6$	28	Id	299	3883
13	Cyclo	3	3	1	$1.33 = 4/3$	32	Id	342	4438
13	Cyclo	3	3	10	$1.42 = 17/12$	34	Id	363	4715
13	Cyclo	3	3	2	$1.58 = 19/12$	38	Id	406	5270
14	Cyclo (6.3)	1	2	1	$1.50 = 3/2$	18	$-x$	384	5376
14	Cyclo (6.6)	3	3	5	$1.33 = 4/3$	16	Id	342	4779
15	Cyclo (BLS)	3	1	1	$1.50 = 3/2$	12	Id	384	*5760*
15	Cyclo (6.6)	3	1	11	$1.50 = 3/2$	12	Id	384	*5760*
16	KSS16 (6.11)	1	–	–	$1.25 = 5/4$	10	Id	320	5120
17	Cyclo (6.2)	1	4	9	$1.18 = 19/16$	38	$-x$	304	5168
17	Cyclo (6.6)	3	3	6	$1.12 = 9/8$	36	$1/x$	288	4896
17	Cyclo	3	3	12	$1.19 = 19/16$	38	Id	304	5168

We obtain lower DL cost estimates in the embedding field of these curves, except for $k = 9$ construction LZZW (that we set in the BLS framework). Sometimes the cost for STNFS is not given in [4], we give our estimate. We investigated these differences by running the scripts provided with [4] and developing a second implementation based on the SageMath code available with [23,22]. We develop the following improvements.

1. Given p^k as input, for each possible integer factorisation $k = \eta\kappa$ with $\eta > 1$, we generate many irreducible polynomials h of degree η and pairs of polynomials (f_y, g_y).
2. For each set of polynomials (h, f_y, g_y), the code iterates and adjusts automatically the parameters A, B (sieving bound, smoothness bound) in order to find the best combination that balances the costs of relation collection and linear algebra, so that the total cost is minimised. When plugging these values into the former scripts and adding a tiny offset if needed, one obtains the new results.
3. We implement the improvements of the Special setting described in the variants 1-4: automorphisms and changes of variables on $p(x)$ to minimise the average size of norms.
4. We compute the joint average size of norms and smoothness probabilities for $K_{y,f}$ and $K_{y,g}$ simultaneously. This allows to compute the ratio of non-coprime ideals $a(y)\mathcal{O}, b(y)\mathcal{O}$ and validates the formula $1/\zeta_{K_h}(2)$.

We obtain the results of Tables 3, 4 and 5. In Table 3, we reproduce the results of Barbulescu, El Mrabet and Ghammam [4, §3.4]. We hereafter make the following remarks.

Remark 2.

- We do not consider even embedding degrees k with Construction 6.2. As explained in [20], 6.2 is valid for odd embedding degrees, 6.3 is for $k = 2 \bmod 4$, and 6.4 for $k = 4 \bmod 8$. Hence we do not report even k with 6.2 in Table 3.
- For $k = 10$ and construction 6.3, we obtained a lower DL cost with $\eta = 10$ instead of $\eta = 5$. We obtained 2^{122} instead of 2^{134}.
- For all curves but BN and BLS12, we obtain a lower estimated cost with optimised parameters A, B and § 2.1.
- When the degree of $p(x)$ is large, we apply one of the variants using automorphisms 1, 2 or 3 if applicable, so that $\deg P = \deg p(x)/2$. We compared without the polynomial variants and observed a lower DL cost estimate with variants 1 or 2 when the degree of $p(x)$ is more than 12. Note that the variant 2 is commented in the Python script of [4] for $k = 17(6.6)$.
- We observed that when the degree of P is more than 12 (after applying variants 1, 2 or 3 if applicable), applying our improvement 4 reduces further the estimated complexity of STNFS. We obtained the smallest cost with P of degree between 4 and 12. This case is reported in the right-most column of Tables 3 and 4. The curves involved with this improvement are $k = 9(6.7)$, $k = 10(6.6)$, $k = 11(6.2)$, $k = 13(6.2)$ and $k = 13(6.6)$, $k = 14(6.6)$ and $k = 17(6.2)$, $k = 17(6.6)$.

16

Moreover, we applied our work to the parameter seeds of [4]. The previous remarks apply: we do not consider the seeds of even k with Construction 6.2 ([4, Table 10]). We identified five seeds that produce insecure curves because the STNFS estimated cost in $\mathbb{F}_{p^k}$ is below 2^{128}: these are $k = 9$ BLS (denoted LZZW in [4, Table 23]), $k = 9$ (6.2), $k = 10$ (6.3), $k = 11$ (6.2) and $k = 11$ (6.6). Our DL security estimate is given in bold coloured font in Table 4.

For $k = 10$ (6.3), the size of u is smaller than the minimum size recommended in [4, §3.4] ($p(u)$ is 433-bit long instead of 446, and $r(u)$ is 249-bit long, smaller than 256 bits). For $k13(6.2)$, the minimum size recommended in [4, §3.4] is $p(u)$ of 329 bits, but the seed produces a 599-bit prime p. The security is much larger than 2^{128}. These two cases are reported in italic coloured font in Table 4.

Our results. In Table 5 we present our estimations of STNFS security. For each curve family in Table 2, we first generate seeds and parameters so that r is at least a 256-bit prime. Then we run our estimation of STNFS, trying many combinations of degrees of $h(Y)$ and of $P(x)$. When the cost is smaller than 2^{128}, we increase the size of the seed u and generate larger parameters $r(u)$ and $p(u)$. We report the minimum size of p so that r is at least 256-bit long, and the security in $\mathbb{F}_{p^k}$ is at least 2^{128}.

For each embedding degree k, we highlight in coloured background the family that has no automorphism available in $p(x)$ so that the variants 1, 2 and 3 do not apply, and so that $p(u)$ has minimal possible size. We eliminate the embedding degree $k = 17$. Since $p(x)$ has large degree of 36 or 38, it was not possible to find a seed u so that $p(u)$ and $r(u)$ are prime, and $p^k(u)$ is less than 5376 bits (constrain of Eq. (5)). We eliminate embedding degree $k = 9$: the curves whose $p(x)$ has no automorphism do not satisfy $p^k(u) \leq 5376$.

There are eleven highlighted families in Table 5. The families of Fotiadis and Martindale [18] with $k = 12$ and $D = 3$ (denoted FM17, FM19 and FM20) have very similar properties and like in [18], we only include FM17 in our final short list (for the same bitsize of $p(u)$, FM17 produces $r(u)$ one bit larger than FM19 and four bits larger than FM20).

We are left with a final short-list of nine STNFS-secure pairing-friendly curves that we summarise in Table 6. We give the polynomials $p(x), r(x), t(x)$ as Curves 1, 2, 3, 4, 5. We add the modified Cocks–Pinch curve with $k = 8$ from [22] as it looks quite promising in terms of pairing efficiency [1].

Remark 3. The curves listed below all admit a fast endomorphism from the complex multiplication, because their discriminant $-D$ is small. For curves with $-D = -4$ and j-invariant 1728, the endomorphism is $(x, y) \mapsto (-x, iy)$, where $i^2 = -1$ (in short Weierstrass representation). The curves are ordinary, $p \equiv 1 \bmod 4$, and there exists $i \in \mathbb{F}_p$ such that $i^2 = -1 \bmod p$. More precisely, we can easily precompute i. The characteristic p has form $p = (t^2 + y^2)/4$ where t is the trace, and $t^2 - 4p = -y^2$. Then $\sqrt{-1} \equiv t/y \bmod p$. The endomorphism has characteristic polynomial $x^2 + 1$, and eigenvalue $\sqrt{-1} \bmod r$, where $r \mid p + 1 - t$ and r is prime. Writing $p+1-t = ((t-2)^2+y^2)/4$, one has $\sqrt{-1} \equiv (t-2)/y \bmod r$. This is explained in details in [34]. When the cofactor c of the elliptic curve such

Table 3. Pairing-friendly Constructions for $9 \leq k \leq 17$ from Table 2 and their security estimate in [4], with $\eta = \deg h$, $n = \eta\kappa$ (see Fig. 2). In several cases the data in [4] was missing or unexpected (it seems that the parameters A, B were not enough optimised). The polynomial $P(x^i)$ equals $3p(x)$ for $D = 3$, $4p(x)$ for $D = 1$, $8p(x)$ for $D = 2$, $p(x)$ for BN, and $P(x + 1) = 980p(x)$ for KSS16.

k	Construction	D,m,e_0	deg $p(x)$	p bits	p^k bits	r bits	η	poly P	deg P	DL cost [4]	§2.1	new params	deg P	DL cost
9	Cyclo (BLS)	3,1,1	8	591	5314	443	9	$P(x)$	8	128	**128**			
9	Cyclo (6.6)	3,1,7	8	535	4810	401	9	$P(x)$	8	129	**122**			
9	Cyclo (6.2)	1,4,5	22	484	4356	266	9	$P(x^2)$	11	134	**116**			
9	Cyclo (6.7)	2,8,1	46	520	4672	273	9	$P(x^2)$	23	266	220	$P(u^4)$	11	140
10	Cyclo (6.3)	1,2,1	14	446	4460	256	5	$P(x^2)$	7	134	133	$\eta\!=\!10$	7	**121**
10	(Cyclo) 6.6	3,3,1	16	511	5104	256	10	$P(x)$	16	166	152	$P(u^2)$	8	150
11	Cyclo (6.2)	1,4,6	26	337	3698	*189*	11	$P(x^2)$	13	173	**118**			
11	Cyclo (6.6)	3,3,4	24	311	3421	*78*	11	$P(x)$	24	$\emptyset$	232	$P(x\!+\!1/x)$	12	**114**
12	BN (6.8)	3,–,–	4	462	5534	462	6	$P(x)$	4	128	135			
12	Cyclo (BLS)	3,1,1	6	461	5525	308	6	$P(x)$	6	128	135			
12	Cyclo (6.7)	2,2,1	14	445	5340	256	12	$P(x^2)$	7	148	134			
12	(Cyclo) 6.4	1,1,1	8	510	6120	256	12	$P(x)$	8	$\emptyset$	138			
13	Cyclo (6.2)	1,4,7	30	329	4265	*218*	13	$P(x^2)$	15	325	143	$P(u^4)$	7	140
13	Cyclo (6.6)	3,3,9	28	309	4009	*218*	13	$P(x)$	28	$\emptyset$	288	$P(u^3)$	9	140
14	Cyclo (6.3)	1,2,1	18	394	5516	264	14	$P(x^2)$	9	148	132			
14	Cyclo (6.6)	3,3,5	16	351	4906	264	14	$P(x)$	16	175	**151**	$P(u^2)$	8	151
15	Cyclo (BLS)	3,1,1	12	383	5745	257	15	$P(x)$	12	286	**138**			
15	Cyclo (6.6)	3,1,11	12	383	5736	256	15	$P(x)$	12	175	**138**			
16	KSS16 (6.11)	1,–,–	10	331	5281	257	16	$P(x\!+\!1)$	10	154	**140**			
17	Cyclo (6.2)	1,4,9	38	304	5153	*135*	17	$P(x^2)$	19	254	189	$P(u^4)$	9	153
17	Cyclo (6.6)	3,3,6	36	348	5914	*249*	17	$P(x\!+\!1/x)$	18	$\emptyset$	186	$P((u\!+\!1/u)^2)$	9	168

that $r \cdot c = p + 1 - t$ is larger (not just 1 or 2 for example), and the curve has parameters of polynomial form, one can reduce the lattice spanned by the rows $(r(x), 0)$ and $(y(x), t(x) - 2)$ to obtain a short basis. The Magma language for example allows lattice reduction over polynomials.

For curves with $-D = -3$, the endomorphism is $(x, y) \mapsto (\omega x, y)$, where $\omega \in \mathbb{F}_p$ is a third root of unity, such that $\omega^2 + \omega + 1 = 0$. The endomorphism has characteristic polynomial $x^2 + x + 1$ and eigenvalue $\lambda \bmod r$ such that $\lambda^2 + \lambda + 1 = 0 \bmod r$. We can easily precompute ω and λ. Since $p = (t^2 + 3y^2)/4$, then $\sqrt{-3} \equiv t/y \bmod p$, and $\omega \equiv (-1 + \sqrt{-3})/2 \equiv (-y + t)/(2y) \bmod p$. We also have $\sqrt{-3} \equiv (t - 2)/y \bmod r$. The eigenvalue is $\lambda \equiv (-1 + \sqrt{-3})/2 \equiv (-y + t - 2)/(2y) \bmod r$. Since the square roots are given up to sign, in practice one obtains equality up to sign ($[\pm\lambda](x_p, y_P) = (\omega x_P, y_P)$ or $[\pm\lambda](x_P, y_P) = (\omega^2 x_P, y_P)$), that is, a practical adjustment is required.

We give a polynomial form of low degree for $\beta = \sqrt{-D} \bmod p$ and $\lambda = \sqrt{-D} \bmod r$ for the curves below.

Table 4. Seeds provided in [3,4,19]. No seed is given for $k = 9$, $k = 10$ with 6.6, $k = 17$. The seeds for $k = 12, 16$ are from [3].

k	curve	ref	seed u	$p(u)$ bits	$p^k(u)$ bits	$r(u)$ bits	$\eta,$ special poly	DL cost
9	BLS	[4, T.23]	$2^{74} + 2^{35} - 2^{22} + 2$	591	5314	443	9 $P(x)$	128
9	BLS	[19, §8.1]	$2^{70} + 2^{59} + 2^{46} + 2^{41} + 1$	559	5026	419	9 $P(x)$	**125**
9	6.2	[4, T.6]	$-1+2^3+2^4+2^5+2^9+2^{10}+2^{22}$	483	4339	265	9 $P(x^2)$	**116**
9	6.7	[4, T.19]	$-1-2^4+2^6+2^9+2^{11}$=0xa2f	520	4672	273	9 $P(u^4)$	140
10	6.3	[4, T.7]	$1+2^3-2^5+2^{10}+2^{13}+2^{31}$	433	4321	*249*	10 $P(x^2)$	**120**
11	6.2	[4, T.6]	$-1 + 2^8 + 2^{14}$	363	3993	281	11 $P(x^2)$	**122**
11	6.6	[4, T.16]	$2^4+2^6+2^7+2^9+2^{10}+2^{14}$	338	3718	283	11 $P(x+1/x)$	**118**
12	BN	[3]	$2^{114} + 2^{101} - 2^{14} - 1$	462	5535	462	6 $P(x)$	135
12	BLS	[3]	$-2^{77} + 2^{50} + 2^{33}$	461	5525	308	6 $P(x)$	134
12	6.7	[4, T.18]	$1 + 2^{14} + 2^{17} + 2^{32}$	446	5341	257	12 $P(x^2)$	134
12	6.4	[4, T.8]	$1+2+2^3+2^8+2^9+2^{11}+2^{64}$	511	6121	257	12 $P(x)$	138
13	6.2	[4, T.6]	$1+2+2^3+2^4+2^8+2^{10}+2^{14}+2^{20}$	599	7784	481	13 $P(x^2)$	162
13	6.6	[4, T.16]	$2^4+2^7+2^{10}+2^{11}+2^{13}$=0x2c90	376	4886	324	13 $P(u^3)$	152
14	6.3	[4, T.7]	$1-2^2+2^6+2^9-2^{12}-2^{15}-2^{19}+2^{22}$	391	5464	262	14 $P(x^2)$	131
14	6.6	[4, T.15]	$-1+2^6+2^7+2^9+2^{10}+2^{13}+2^{17}+2^{22}$	352	4917	265	14 $P(u^2)$	150
15	BLS	[19, §8.1]	$2^2 + 2^5 + 2^{19} + 2^{31}$	371	5557	*249*	15 $P(x)$	137
15	BLS	[4, T.23]	$2 + 2^{10} + 2^{16} + 2^{19} + 2^{32}$	383	5737	257	15 $P(x)$	138
15	6.6	[4, T.14]	$1 + 2^2 + 2^{12} + 2^{16} + 2^{32}$	383	5737	257	15 $P(x)$	138
16	KSS	[3]	$-2^{34}+2^{27}-2^{23}+2^{20}-2^{11}+1$	330	5280	257	16 $P(x+1)$	140

Curve 1. A pairing-friendly curve $y^2 = x^3 + ax + b$ with the Brezing–Weng method, $k = 10$, $D = 15$, $m = 3$, $e_0 = 1$, $\rho = 7/4 = 1.75$ ([17, Table 2]).
$r = \Phi_{30}(x) = x^8 + x^7 - x^5 - x^4 - x^3 + x + 1$
$p = (4x^{14} + 4x^{13} + x^{12} - 12x^{11} - 12x^{10} - 7x^9 + 11x^8$
$\qquad + 17x^7 + 15x^6 - 3x^5 - 11x^4 + x^3 - 2x^2 + 3x + 6)/15$
$t = x^3 + 1 \qquad\qquad ; y = (x-1)(4x^6 + 6x^5 + 6x^4 - 3x^2 - 5x - 3)/15$
$u = 1, 3, 6, 13 \bmod 15$; $c = (x-1)(2x^2 + x + 2)(2x^2 + 3x + 3)/15$
The Hilbert class polynomial is $H(-15) = x^2 + 191025x - 121287375$ of discriminant $5(3^3 \cdot 5 \cdot 7^2 \cdot 13)^2$. For a root $j_0 = 135(-1415 \pm 637\sqrt{5})/2$ of $H(-15)$ modulo p, one has $a = -3j_0/(j_0 - 1728)$, $b = 2j_0/(j_0 - 1728)$. A simplified pair is $(a, b) = (-3(245 \pm 416\sqrt{5}), 154(\pm 416 + 49\sqrt{5}))$. Moreover if $\omega = j_0/(j_0 - 1728) = 5^2/11^2 \pm 2^5 \cdot 5 \cdot 13\sqrt{5}/(7^2 11^2)$ is a square modulo p, one can have $a' = -3$, $b' = b/\omega^{3/2}$. If the curve $y^2 = x^3 + ax + b$ is the quadratic twist (of order $p + 1 + t$ instead of $p + 1 - t$), then $y^2 = x^3 + a\nu^2 x + b\nu^3$ is the curve we want, where ν is a non-square modulo p.

The short eigenvalue of the endomorphism for GLV (see [34]) is $\lambda = \sqrt{-15} \equiv 2x^7 - 2x^5 - 4x^4 - 2x^3 - 2x^2 + 4x + 3 \equiv (2x^4 + x^3 - 4x^2 + x + 2)/(x^3 - x) \bmod r(x)$. Note that the square root is defined up to sign. We also have $\sqrt{-15} \equiv (-64x^{13} - 24x^{12} + 8x^{11} + 250x^{10} + 92x^9 + 32x^8 - 448x^7 - 226x^6 - 146x^5 + 398x^4 + 222x^3 + 32x^2 - 42x - 159)/45 \bmod p(x)$. The endomorphism can be obtained from a 3-isogeny and a 5-isogeny. There are two 3-isogenies and two 5-isogenies, one

19

Table 5. Pairing-friendly Constructions for $9 \le k \le 17$ from Table 2 and our new security estimate. For $k = 17$, r is a prime divisor of $r(u)$ but $r(u)$ itself is not prime, there is a cofactor (mark *). For many families with $k = 11$ and $k = 13$, it was not possible to find a seed u such that r is 256-bit long ($^+$) because $r(x)$ has a high degree.

k	Construction, D,m,e_0		deg $p(x)$	p bits	p^k bits	r bits	η	§2.2	deg P	$\lVert P \rVert$	P	DL cost
9	Cyclo (BLS)	3,1,1	8	608	5472	456	9		8	1	$P(x)=3p(x)$	130
9	Cyclo (6.6)	3,1,7	8	608	5472	456	9		8	1	$P(x)=3p(x)$	130
9	Cyclo (6.2)	1,4,5	22	640	5752	350	9	(1)	11	1	$P(x^2)=4p(x)$	130
9	Cyclo (6.7)	2,8,1	46	520	4672	273^+	9	(1+4)	11	u^2	$P(u^4)=8p(u)$	140
9	Cyclo (FM10)	3,1,5	12	608	5472	304	9		12	1	$P(x)=3p(x)$	133
10	Cyclo (6.5)	1,2,9	12	480	4800	322	5	(1)	6	1	$P(x^2)=3p(x)$	128
10	Cyclo (6.3,FM13)	1,2,1	14	512	5120	294	10	(1)	7	1	$P(x^2)=4p(x)$	129
10	Cyclo (FM16)	2,4,9	30	488	4871	262^+	10	(1)	15	1	$P(x^2)=8p(x)$	141
10	(Cyclo) 6.6	3,3,1	16	511	5104	256	10	(4)	8	u	$P(u^2)=3p(u)$	145
10	Cyclo (FM14)	5,2,9	14	480	4800	276	10	(1)	7	1	$P(x^2)=20p(x)$	128
10	Cyclo (FM15)	15,3,1	14	446	4460	256	10		14	1	$P(x)=15p(x)$	133
11	Cyclo (6.2)	1,4,6	26	414	4554	320	11	(1)	13	1	$P(x^2)=4p(x)$	130
11	Cyclo	1,4,1	30	391	4297	262^+	11	(1+4)	7	u^2	$P(u^4)=4p(u)$	136
11	Cyclo	1,4,7	34	444	4876	262^+	11	(1+4)	8	u^2	$P(u^4)=4p(u)$	146
11	Cyclo (6.6)	3,3,4	24	446	4899	373	11	(2)	12	1	$x^{12}P(x+1/x)=3p(x)$	128
11	Cyclo	3,3,8	26	333	3663	258^+	11	(4)	8	u^2	$P(u^3)=3p(u)$	131
11	Cyclo	3,3,1	28	355	3901	255^+	11	(4)	9	u^2	$P(u^3)=3p(u)$	135
11	Cyclo	3,3,1	28	373	4101	268^+	11	(4)	9	u^2	$P(u^3)=3p(u)$	139
11	Cyclo	11,1,4	16	411	4521	256	11	(4)	8	u	$P(u^2)=11p(u)$	145
11	Cyclo	11,1,8	16	480	5280	298	11	(2)	8	1	$x^8P(x+1/x)=11p(x)$	130
12	BN (6.8)	3,–,–	4	446	5376	446	6		4	1	$P(x)=p(x)$	132
12	Cyclo (BLS)	3,1,1	6	446	5376	299	6		6	1	$P(x)=3p(x)$	132
12	FK12 (FM17)	3,–,–	6	446	5352	296	6		6	1	$P(6x+2)=108p(x)$	136
12	FM19	3,–,–	6	446	5352	295	6		6	1	$P(x)=225p(x)$	135
12	FM20	3,–,–	6	446	5352	292	6		6	1	$P(x+3)=1425p(x)$	137
12	Cyclo (6.7,FM18)	2,2,1	14	445	5329	256	12	(1)	7	1	$P(x^2)=8p(x)$	134
12	(Cyclo) 6.4	1,1,1	8	509	6097	256	12		8	1	$P(x)=4p(x)$	138
13	Cyclo (6.2)	1,4,7	30	339	4396	256^*	13	(1+4)	7	u^2	$P(u^4)=4p(u)$	142
13	Cyclo	1,4,1	34	380	4931	270^+	13	(1+4)	8	u^2	$P(u^4)=4p(u)$	141
13	Cyclo (6.6)	3,3,9	28	310	4027	267^+	13	(4)	9	u^2	$P(u^3)=3p(u)$	140
13	Cyclo	3,3,1	32	348	4512	262^+	13	(4)	10	u^2	$P(u^3)=3p(u)$	139
13	Cyclo	3,3,10	34	388	5037	275^+	13	(4)	8	u^2	$P(u^4)=3p(u)$	144
13	Cyclo	3,3,2	38	403	5233	256	13	(4)	6	u^2	$P(u^6)=3p(u)$	150
14	Cyclo (6.3)	1,2,1	18	382	5376	256	14	(1)	9	1	$P(x^2)=4p(x)$	130
14	Cyclo (6.6)	3,3,5	16	340	4755	256	14	(4)	8	u	$P(u^2)=3p(u)$	148
15	Cyclo (BLS)	3,1,1	12	381	5715	256	15		12	1	$P(x)=3p(x)$	137
15	Cyclo (6.6)	3,1,11	12	381	5715	256	15		12	1	$P(x)=3p(x)$	137
16	KSS16 (6.11)	1,–,–	10	330	5280	256	16		10	1	$P(x)=980p(x{-}1)$	140
17	Cyclo (6.2)	1,4,9	38	382	6494	262^*	17	(1+4)	9	u^2	$P(u^4)=4p(u)$	167
17	Cyclo (6.2)	1,4,9	38	359	6087	254^*	17	(1+4)	9	u^2	$P(u^4)=4p(u)$	164
17	Cyclo (6.6)	3,3,6	36	374	6358	281^*	17	(2+4)	9	u^2	$P((u+1/u)^2)u^{36}=3p(u)$	172
17	Cyclo	3,3,12	38	337	5718	255^*	17	(4)	9	u^3	$P(u^4)=3p(u)$	165

Table 6. Our short-list of pairing-friendly curves at the 128-bit security level.

k	Construction D,m,e_0	deg $p(x)$	seed u	p bits	p^k bits	r bits	DL cost in $\mathbb{F}_{p^k}$
6	Cocks–Pinch 3,–,–	4	$2^{128} - 2^{124} - 2^{69}$[22]	672	4028	256	128 [22]
8	Cocks–Pinch 1,–,–	8	$2^{64} - 2^{54} + 2^{37} + 2^{32} - 4$[22]	544	4349	256	131 [22]
10	Cyclo(FM15)15,3,1	14	$2^{32} - 2^{26} - 2^{17} + 2^{10} - 1,\, a = -3$	446	4460	256	133
11	Cyclo 3,3,8	26	$-2^{13}+2^{10}-2^8-2^5-2^3-2=-\mathtt{0x1d2a}, b=13$	333	3663	258^+	131
11	Cyclo 11,1,4	16	$-2^{26} + 2^{21} + 2^{19} - 2^{11} - 2^9 - 1,\, a = 2$	412	4522	256	145
12	BN (6.8) 3,–,–	4	$2^{110} + 2^{36} + 1,\, b = 257$ [31]	446	5376	446	132 [23]
12	Cyclo (BLS) 3,1,1	6	$-(2^{74}+2^{73}+2^{63}+2^{57}+2^{50}+2^{17}+1), b{=}1$ [22,23]	446	5376	299	132[23]
12	FK12 (FM17)3,–,–	6	$-2^{72} - 2^{71} - 2^{36},\, b = -2$ [18, §4(b)]	446	5352	296	136
13	Cyclo (6.6) 3,3,9	28	$2^{11} + 2^8 - 2^6 - 2^4 = \mathtt{0x8b0},\, b = -17$	310	4027	267^+	140
14	Cyclo (6.6) 3,3,5	16	$2^{21} + 2^{19} + 2^{10} - 2^6,\, b = -4$	340	4755	256	148
16	KSS16 (6.11) 1,–,–	10	$-2^{34}+2^{27}-2^{23}+2^{20}-2^{11}+1, a{=}1$ [3]	330	5280	257	140 [23]
16	KSS16 (6.11) 1,–,–	10	$2^{34}-2^{30}+2^{26}+2^{23}+2^{14}-2^5+1, a{=}1$	330	5268	256	140

combination gives an endomorphism (we were able to check it on a numerical example in Magma, and obtained the eigenvalue $-\lambda(u)$).

Curve 2. A pairing-friendly curve $y^2 = x^3 + b$ with the Brezing–Weng method, $k = 11$, $D = 3$, $m = 3$, $e_0 = 8$, $\rho = 13/10 = 1.30$. Since $D = 3$, $a = 0$.
$r = \Phi_{33}(x) = x^{20} - x^{19} + x^{17} - x^{16} + x^{14} - x^{13} + x^{11} - x^{10} + x^9 - x^7 + x^6$
$\qquad - x^4 + x^3 - x + 1$
$p = (x^{26} + x^{24} + x^{22} + x^{15} - 2x^{13} + x^{11} + x^4 - 2x^2 + 1)/3$
$t = x^{3 \times 8} + 1 \bmod r = -x^{13} - x^2 + 1$; $y = (x^{13} + 2x^{11} - x^2 + 1)/3$
$u = 1, 2 \bmod 3 \qquad\qquad\qquad ; c = (x^2 - x + 1)(x^2 + x + 1)^2/3$
The eigenvalue of the endomorphism $(x, y) \mapsto (\omega x, y)$ is $\lambda \equiv (-1 + \sqrt{-3})/2 \equiv x^{11}$
$\equiv (x^{10} - x^9 + x^7 - x^6 + x^4 - x^3 + x - 1)/(x^9 - x^8 + x^6 - x^5 + x^3 - x^2 + 1) \bmod r(x)$, and
$\omega \equiv (-1 + \sqrt{-3})/2 \equiv (2x^{25} - x^{24} + 5x^{23} + 7x^{21} - x^{20} + 8x^{19} + x^{18} + 7x^{17} - x^{16} + 8x^{15} + 3x^{14} + 6x^{13} - 2x^{12} + 6x^{11} - x^{10} - 2x^9 + x^8 + 2x^7 - x^6 - 2x^5 + x^4 + 4x^3 - 2x^2 - 3x - 1)/5$.

Curve 3. A pairing-friendly curve $y^2 = x^3 + ax + b$ with the Brezing–Weng method, $k = 11$, $D = 11$, $m = 1$, $e_0 = 4$, $\rho = 8/5 = 1.6$.
$r = \Phi_{11}(x) = x^{10} + x^9 + x^8 + x^7 + x^6 + x^5 + x^4 + x^3 + x^2 + x + 1$
$p = (x^{16} + 2x^{15} + x^{14} - x^{12} - 3x^{11} - x^5 + 9x^4 - x^3 + x + 3)/11$
$t = x^4 + 1 \qquad ; y = (2x^8 + 2x^7 - x^4 - 2x^3 + 2x^2 - 2x - 1)/11$
$u = 1 \bmod 11$; $c = (x - 1)^2(x^4 + 3x^3 + 4x^2 + 4x + 3)$
The j-invariant of a curve of discriminant -11 is $-32768 = -2^{15}$, and $(a, b) = (-264, 1694)$. Moreover if 22 is a square modulo p, one can define $(a', b') = (-3, 7\sqrt{22}/2^4)$. The endomorphism can be obtained from a 11-isogeny. The eigenvalue is $\lambda \equiv \sqrt{-11} \equiv 2x^9 + 2x^5 + 2x^4 + 2x^3 + 2x + 1 \equiv (2x^5 + x^4 - 2x^3 + 2x^2 - x - 2)/(x^4 + x) \bmod r(x)$.

Curve 4. A pairing-friendly curve $y^2 = x^3 + b$ with the Brezing–Weng method, $k = 13$, $D = 3$, $m = 3$, $e_0 = 9$ (this is (6.6)), $\rho = 7/6 = 1.17$. Since $D = 3$, $a = 0$.

$$r = \Phi_{39}(x) = x^{24} - x^{23} + x^{21} - x^{20} + x^{18} - x^{17} + x^{15} - x^{14} + x^{12} - x^{10} + x^9$$
$$- x^7 + x^6 - x^4 + x^3 - x + 1$$
$$p = (x^{28} + x^{27} + x^{26} + x^{15} - 2x^{14} + x^{13} + x^2 - 2x + 1)/3$$
$$t = (x^{3 \times 9} + 1) \bmod r = -x^{14} - x + 1 \ ; \ y = (x^{14} + 2x^{13} - x + 1)/3$$
$$u = 1 \bmod 3 \qquad\qquad\qquad ; \ c = (x^2 + x + 1)^2/3$$

The endomorphism is $(x, y) \mapsto (\omega x, y)$ where $\omega \equiv (-1 + \sqrt{-3})/2 \equiv x^{26} + x^{25} + 2x^{24} + x^{23} + 2x^{22} + x^{21} + 2x^{20} + x^{19} + 2x^{18} + x^{17} + 2x^{16} + x^{15} + 2x^{14} + x^{13} - x^{12} + x^{11} - x^{10} + x^9 - x^8 + x^7 - x^6 + x^5 - x^4 + x^3 - x^2 + x - 1 \bmod p(x)$. The eigenvalue is $\lambda \equiv (-1 + \sqrt{-3})/2 \equiv x^{13} \equiv (x^{11} - x^{10} + x^8 - x^7 + x^5 - x^4 + x^2 - x)/(x^{12} - x^{11} + x^9 - x^8 + x^6 - x^5 + x^3 - x^2 + 1) \bmod r(x)$.

Curve 5. A pairing-friendly curve $y^2 = x^3 + b$ with the Brezing–Weng method, $k = 14$, $D = 3$, $m = 3$, $e_0 = 5$ (this is (6.6)), $\rho = 4/3 = 1.33$. Since $D = 3$, $a = 0$.
$$r = \Phi_{42}(x) = x^{12} + x^{11} - x^9 - x^8 + x^6 - x^4 - x^3 + x + 1$$
$$p = (x^{16} + x^{15} + x^{14} - x^9 + 2x^8 - x^7 + x^2 - 2x + 1)/3$$
$$t = (x^{3 \times 5} + 1) \bmod r = x^8 - x + 1 \ ; \ y = (x^8 + 2x^7 + x - 1)/3$$
$$u = 1 \bmod 3 \qquad\qquad\qquad ; \ c = (x^2 - x + 1)(x^2 + x + 1)/3$$

The endomorphism is $(x, y) \mapsto (\omega x, y)$ where $\omega \equiv (-1 + \sqrt{-3})/2 \equiv (2x^{15} + 3x^{14} + 5x^{13} + 4x^{12} + 5x^{11} + 4x^{10} + 5x^9 + 2x^8 + 5x^7 - x^6 + x^5 - x^4 + x^3 - x^2 + 3x - 4)/3 \bmod p(x)$. The eigenvalue is $\lambda \equiv (-1 + \sqrt{-3})/2 \equiv x^{13} \equiv (x^5 + x^4 - x^2 - x)/(x^6 - x^4 - x^3 + x + 1) \bmod r(x)$.

4   Optimal Ate Pairing Computation: Miller Loop

We leave to future work the final exponentiation and we focus on the Miller loop, sketched in Alg. 4.1. We compare the curves of Table 6 to [22, Table 10] and summarise the costs in Table 8. Let $\mathbf{m}_k$ denotes a multiplication in $\mathbb{F}_{p^k}$, $\mathbf{m}$ a multiplication in $\mathbb{F}_p$, $\mathbf{s}_k$ a square in $\mathbb{F}_{p^k}$ and $\mathbf{s}$ a square in $\mathbb{F}_p$. For curves $y^2 = x^3 + b$ with j-invariant 0 ($a = 0$), we give the counts from [13]. For prime embedding degrees ($k = 11, 13$), we apply the formulas from [22, Table 7].

Algorithm 4.1: MILLERFUNCTION(u, P, Q)
Input: $E, \mathbb{F}_p, \mathbb{F}_{p^k}, P \in E(\mathbb{F}_p)[r], Q \in E(\mathbb{F}_{p^k})[r]$ in affine coord., $\pi_p(Q) = [p]Q$, $u \in \mathbb{N}$.
Result: $f = f_{u,Q}(P)$

1 $f \leftarrow 1; R \leftarrow Q;$
2 **for** b *from the second most significant bit of u to the least* **do**
3      $\ell \leftarrow \ell_{R,R}(P); R \leftarrow [2]R$;   DOUBLELINE
4      $v \leftarrow v_{[2]R}(P)$;   VERTICALLINE
5      $f \leftarrow f^2 \cdot \ell/v;$ UPDATE1
6   **if** $b = 1$ **then**
7          $\ell \leftarrow \ell_{R,Q}(P); R \leftarrow R + Q$;   ADDLINE
8          $v \leftarrow v_{R+Q}(P)$;   VERTICALLINE
9          $f \leftarrow f \cdot \ell/v$;   UPDATE2
10 **return** $f;$

Table 7. Miller loop cost in Weierstrass model from [11,13,22].

k	curve	DoubleLine AddLine	Vertical Line	Update1 Update2	reference
any k	$y^2 = x^3 + ax + b$	$5\mathbf{m}_k + 6\mathbf{s}_k + 2k\mathbf{m}$ $10\mathbf{m}_k + 3\mathbf{s}_k$	$k\mathbf{m}$	$4\mathbf{m}_k + 2\mathbf{s}_k$ $4\mathbf{m}_k$	[22, Alg. 3,4,5]
any k	$y^2 = x^3 + b$	$5\mathbf{m}_k + 5\mathbf{s}_k + 2k\mathbf{m}$ $10\mathbf{m}_k + 3\mathbf{s}_k$	$k\mathbf{m}$	$4\mathbf{m}_k + 2\mathbf{s}_k$ $4\mathbf{m}_k$	[22, Alg. 3,4,5]
$2 \mid k$	$y^2 = x^3 + b$ quadratic twist	$2\mathbf{m}_{k/2} + 7\mathbf{s}_{k/2} + k\mathbf{m}$ $10\mathbf{m}_{k/2} + 2\mathbf{s}_{k/2} + k\mathbf{m}$	0	$\mathbf{m}_k + \mathbf{s}_k$ $\mathbf{m}_k$	[13, §5,Tab.3]
$2 \mid k$	$y^2 = x^3 - 3x + b$ quadratic twist	$6\mathbf{m}_{k/2} + 4\mathbf{s}_{k/2} + k\mathbf{m}$ $10\mathbf{m}_{k/2} + 3\mathbf{s}_{k/2} + k\mathbf{m}$	0	$\mathbf{m}_k + \mathbf{s}_k$ $\mathbf{m}_k$	[11]

The cost of a Miller function $f_{u,Q}(P)$ for optimal ate pairing computation is given by Eq. (6), where nbits is the bitlength and $\mathrm{HW}_{\text{2-NAF}}$ is the Hamming weight in 2-non-adjacent form, and $\mathbf{i}_k$ an inversion in $\mathbb{F}_{p^k}$.

$$
\begin{aligned}
\mathrm{Cost}_{\text{MillerFunction}} = \; &(\text{nbits(2-NAF}(u)) - 1)\,(\mathrm{Cost}_{\text{DoubleLine}} + \mathrm{Cost}_{\text{VerticalLine}}) \\
&+ (\text{nbits(2-NAF}(u)) - 2)\mathrm{Cost}_{\text{Update1}} \\
&+ (\mathrm{HW}_{\text{2-NAF}}(u) - 1)(\mathrm{Cost}_{\text{AddLine}} + \mathrm{Cost}_{\text{VerticalLine}} + \mathrm{Cost}_{\text{Update2}}) \\
&+ (\text{if there is no twist})\mathbf{i}_k. \qquad (6)
\end{aligned}
$$

The Miller loop is a product of Miller functions $f_{m,Q}(P)$, lines $\ell_{R,S}(P)$ and verticals $v_R(P)$. In all our cases the Miller loop has length of the form $up^i + u^2p^j + p^l$ with i, j, l positive integers and is computed as

$$
f_{u,Q}^{p^i}(P)f_{u^2,Q}^{p^j}(P)\ell_{\pi_p^i(uQ),\pi_p^j(u^2Q)}(P)/v_{\pi_p^i(uQ)+\pi_p^j(u^2Q)}(P)\ell_{\pi_p^i(uQ)+\pi_p^j(u^2Q),\pi_p^l(Q)}(P) \, .
$$

The vertical $v_{\pi_p^i(uQ)+\pi_p^j(u^2Q)+\pi_p^l(Q)}(P)$ can be removed as the point is at infinity. We can optimise with $f_{u^2,Q}(P) = f_{u,Q}^u(P)f_{u,uQ}(P)$. We first compute $f_{u,Q}(P)$, then we start a second Miller iteration over u from $f = f_{u,Q}(P)$ instead of $f = 1$ and with the point uQ instead of Q. Computing uQ in affine coordinates from Jacobian coordinates costs $\mathbf{i}_k + \mathbf{s}_k + 3\mathbf{m}_k$. The exponentiation $f_{u,Q}^u$ is almost free: the squares are shared with the computation of $f_{u,uQ}(P)$. It costs one more multiplication $\mathbf{m}_k$ in addition steps. The computations $\pi_p^i(uQ), \pi_p^l(Q)$ cost $2\mathbf{f}_k$ because uQ is in affine coordinates, while $\pi_p^j(u^2Q)$ costs $3\mathbf{f}_k$, with u^2Q in

projective coordinates. A general formula is

$$
\begin{aligned}
\text{Cost}_{\text{MillerLoop}} = {} & (\text{nbits}(2\text{-NAF}(u)) - 1)(\text{Cost}_{\text{DoubleLine}} + \text{Cost}_{\text{VerticalLine}}) \\
& + (\text{nbits}(2\text{-NAF}(u)) - 2)\text{Cost}_{\text{Update1}} \\
& + (\text{HW}_{2\text{-NAF}}(u) - 1)(\text{Cost}_{\text{AddLine}} + \text{Cost}_{\text{VerticalLine}} + \text{Cost}_{\text{Update2}}) \\
& + \mathbf{i}_k + \mathbf{s}_k + 3\mathbf{m}_k \ (\text{we computed } f_{u,Q}(P) \text{ and } [u]Q \text{ in affine coordinates}) \\
& + (\text{nbits}(u) - 1)(\text{Cost}_{\text{DoubleLine}} + \text{Cost}_{\text{VerticalLine}} + \text{Cost}_{\text{Update1}}) \\
& + (\text{HW}(u) - 1)(\text{Cost}_{\text{AddLine}} + \text{Cost}_{\text{VerticalLine}} + \text{Cost}_{\text{Update2}} + \mathbf{m}_k) \\
& + (1 \text{ or } 2)\mathbf{f}_k + \mathbf{m}_k \ (\text{we computed } f_{u,Q}^{p^i}(P) f_{u^2,Q}^{p^j}(P) \text{ and } [u^2]Q) \\
& + (4 \text{ or } 5)\mathbf{f}_k + 2(\text{Cost}_{\text{AddLine}} + \text{Cost}_{\text{Update2}}) + \text{Cost}_{\text{VerticalLine}} + \mathbf{i}_k \qquad (7)
\end{aligned}
$$

If $\text{HW}_{2\text{-NAF}}(u) \ll \text{HW}(u)$ then one can replace u by the form $2\text{-NAF}(u)$ and $\text{HW}(u)$ by $\text{HW}_{2\text{-NAF}}(u)$ in the second Miller loop to save addition steps, at a cost of one extra inversion $\mathbf{i}_k$.

A Tate pairing has Miller loop $f_{r,P}(Q)$. The curve arithmetic is in the base field $\mathbb{F}_p$ instead of $\mathbb{F}_{p^k}$, but the Miller loop has length $\log_2 r$ bits, this is at least 256 bits. The estimated cost has the same formula given in eq. (6) but with $2\text{-NAF}(r)$ instead of u.

4.1 Prime Embedding Degrees 11 and 13.

Curve 2 ($k = 11$, $D = 3$, $a = 0$, p of 333 bits, $u = $ `-0x1d2a`). The optimal ate Miller loop has length $u + u^2 p^5 + p^6$. The formula is

$$
f_{u,Q}(P) f_{u^2,Q}^{p^5}(P) \ell_{uQ, \pi_p^5(u^2 Q)} / v_{uQ + \pi_p^5(u^2 Q)} \ell_{uQ + \pi_p^5(u^2 Q), \pi_p^6(Q)}
$$

(we omit $v_{uQ + \pi_p^5(u^2 Q) + \pi_p^6(Q)}(P)$ because $uQ + \pi_p^5(u^2 Q) + \pi_p^6(Q) = \mathcal{O}$). The seed $u = $ `-0x1d2a` is 13-bit long and has Hamming weight 7, and $2\text{-NAF}(u)$ is 14-bit long and has Hamming weight 6. Since $D = 3$, we have $a = 0$. No twist is available. From eq. (7) with $(1 + 5)\mathbf{f}_k$ for Frobenius, we obtain $87k\mathbf{m} + 413\mathbf{m}_k + 213\mathbf{s}_k + 6\mathbf{f}_k + 2\mathbf{i}_k$. A schoolbook implementation of multiplication and squaring would give $\mathbf{m}_k = k^2 \mathbf{m} = 121\mathbf{m}$ and $\mathbf{s}_k = k(k-1)\mathbf{m} = 110\mathbf{m}$. A Frobenius power in $\mathbb{F}_{p^{11}}$ costs $10\mathbf{m}$, assuming $p = 1 \bmod 11$ and some precomputations. We obtain the upper bound $74420\mathbf{m} + 2\mathbf{i}_k$. An optimised Karatsuba multiplication in $\mathbb{F}_{p^k}$ would require at least $k^{\log_2 3}\mathbf{m}$, that is, $45\mathbf{m}$. Assuming that $\mathbf{s}_k \approx \mathbf{m}_k \approx 45\mathbf{m}$, we obtain the lower bound $29187\mathbf{m} + 2\mathbf{i}_k$.

For the Tate pairing, from Eq. (6) with $r = \Phi_{33}(u)$, $2\text{-NAF}(r)$ of 258 bits and Hamming weight 86, we obtain $1026k\mathbf{m} + 1364\mathbf{m}_k + 512\mathbf{s}_k + 2477\mathbf{m} + 1540\mathbf{s} + \mathbf{i}_k$, it is roughly $235127\mathbf{m} + 1540\mathbf{s} + \mathbf{i}_k$ with schoolbook $\mathbf{m}_k, \mathbf{s}_k$, and $98183\mathbf{m} + 1540\mathbf{s} + \mathbf{i}_k$ with optimised Karatsuba-like $\mathbf{m}_k, \mathbf{s}_k$.

Curve 3 ($k = 11$, $D = 11$, $a = 2$, p of 412 bits, $u = $ `-0x3d80a01`). The optimal ate Miller loop has length $u - p^3$. The formula is

$$
f_{u,Q}(P) \ell_{uQ, -\pi_p^3(Q)}(P)
$$

24

(we omit $v_{uQ-\pi_p^3(Q)}(P)$ because $uQ - \pi_p^3(Q) = \mathcal{O}$). We have u of 26 bits and Hamming weight 9, 2-NAF(u) of 27 bits and HW$_{\text{2-NAF}}(u) = 6$, and $a = 2$. No twist is available. We obtain from Eq. (6) with in addition $2\mathbf{f}_k + \text{Cost}_{\text{ADDLINE}} + \text{Cost}_{\text{UPDATE2}}$, the cost $83k\mathbf{m} + 314\mathbf{m}_k + 224\mathbf{s}_k + 3\mathbf{f}_k + \mathbf{i}_k$. With the upper bound $\mathbf{m}_k = k^2\mathbf{m}$ and $\mathbf{s}_k = k(k-1)\mathbf{m}$, the count is $63577\mathbf{m} + \mathbf{i}_k$. With the lower bound $\mathbf{m}_{11} = \mathbf{s}_{11} = 45\mathbf{m}$, the count is $25153\mathbf{m} + \mathbf{i}_k$.

A Tate pairing would cost more than the previous curve since $a = 2$ instead of $a = 0$, with this time $r = \Phi_{11}(u)/11$ of 256 bits, 2-NAF(r) of 257 bits, and HW$_{\text{2-NAF}}(r) = 87$. The estimated cost is $1026k\mathbf{m} + 1364\mathbf{m}_k + 510\mathbf{s}_k + 2738\mathbf{m} + 1794\mathbf{s} + \mathbf{i}_k$, upper bound $235168\mathbf{m} + 1794\mathbf{s} + \mathbf{i}_k$, lower bound $98354\mathbf{m} + 1794\mathbf{s} + \mathbf{i}_k$.

Curve 4 ($k = 13$, $D = 3$, (6.6), $a = 0$, p of 310 bits, $u = $ `0x8b0`). The optimal ate Miller loop has length $u^2 + up + p^2$, giving

$$f_{u^2,Q}(P)f_{u,Q}^p(P)\ell_{u^2Q,\pi(uQ)}(P)/v_{u^2Q+\pi(uQ)}\ell_{u^2Q+\pi_p(uQ),\pi_p^2(Q)}$$

(we omit $v_{u^2Q+\pi_p(uQ)+\pi_p^2(Q)}(P)$). We have u and 2-NAF(u) of 12 bits and Hamming weight 4, and $a = 0$, but no twist is available. From eq. 7 with $(1+4)\mathbf{f}_k$, we obtain $73k\mathbf{m} + 313\mathbf{m}_k + 177\mathbf{s}_k + 5\mathbf{f}_k + 2\mathbf{i}_k$. With the schoolbook upper bound $\mathbf{m}_{13} = k^2\mathbf{m} = 169\mathbf{m}$ and $\mathbf{s}_{13} = k(k-1)\mathbf{m} = 156\mathbf{m}$, the count is $81518\mathbf{m} + 2\mathbf{i}_k$. With the Karatsuba-like lower bound $\mathbf{m}_{13} = \mathbf{s}_{13} = 13^{\log_2 3}\mathbf{m} = 59\mathbf{m}$, the count is $29919\mathbf{m} + \mathbf{i}_k$. For Tate from Eq. 6 with 2-NAF(r) of 268 bits and Hamming weight 90, the cost is $1068k\mathbf{m} + 1420\mathbf{m}_k + 532\mathbf{s}_k + 2581\mathbf{m} + 1602\mathbf{s} + \mathbf{i}_k$, upper bound $339437\mathbf{m} + 1602\mathbf{s} + \mathbf{i}_k$ and lower bound $131633\mathbf{m} + 1602\mathbf{s} + \mathbf{i}_k$.

4.2 Even embedding degrees 10 and 14.

The vertical lines can be removed, thanks to the quadratic twist, because the x-coordinates are in a proper subfield $\mathbb{F}_{p^{k/2}}$. The optimisation of line and tangent computation focused on curves with twists of degrees 3, 4 and 6 in [13]. We refer to the former papers [11,2] for pairing formulas on curves with quadratic twists only. The count is

$$\begin{aligned}
\text{Cost}_{\text{MILLERLOOP}} = {} & (\text{nbits}(u) - 1)\text{Cost}_{\text{DOUBLELINE}} + (\text{nbits}(u) - 2)\text{Cost}_{\text{UPDATE1}} \\
& + (\text{HW}(u) - 1)(\text{Cost}_{\text{ADDLINE}} + \text{Cost}_{\text{UPDATE2}}) \\
& + \mathbf{i}_{k/2} + \mathbf{s}_{k/2} + 3\mathbf{m}_{k/2} \text{ (we computed } f_{u,Q}(P) \text{ and } [u]Q \text{ in affine coord.)} \\
& + (\text{nbits}(u) - 1)(\text{Cost}_{\text{DOUBLELINE}} + \text{Cost}_{\text{UPDATE1}}) \\
& + (\text{HW}(u) - 1)(\text{Cost}_{\text{ADDLINE}} + \text{Cost}_{\text{UPDATE2}} + \mathbf{m}_k) \\
& + (1 \text{ or } 2)\mathbf{f}_k + \mathbf{m}_k \text{ (we computed } (f_{u,Q}^{p^i}(P)f_{u^2,Q}^{p^j}(P)) \text{ and } [u^2]Q) \\
& + (2 \text{ or } 3)\mathbf{f}_{k/2} + \text{Cost}_{\text{ADDLINE}} + \text{Cost}_{\text{UPDATE2}} \quad\quad (8)
\end{aligned}$$

Curve 1 ($k = 10$, $D = 15$, $a = -3$, p of 446 bits, $u = $ `0xfbfe03ff`). The optimal ate Miller loop has length $u - p^2 + u^2p^3$. The formula is

$$f_{u,Q}(P)f_{u^2,Q}^{p^3}(P)\ell_{uQ,\pi_p^3(u^2Q)}(P)$$

and we removed the line $\ell_{uQ+\pi_p^3(u^2Q),\pi_p^2(Q)}(P)$ as it is a vertical. Computing $\pi_p^3(u^2Q)$ costs $3\mathbf{f}_{k/2}$. We have u of 32 bits, $\mathrm{HW}(u) = 24$, 2-NAF(u) of 33 bits, $\mathrm{HW}_{\text{2-NAF}}(u) = 5$, $a = -3$, and a quadratic twist is available. We write u in 2-NAF form for both Miller functions, it costs one extra inversion $\mathbf{i}_k$ and allows to save 27 addition steps in the second Miller function, and obtain $72k\mathbf{m} + 76\mathbf{m}_k + 62\mathbf{s}_k + \mathbf{f}_k + 525\mathbf{m}_{k/2} + 280\mathbf{s}_{k/2} + 3\mathbf{f}_{k/2} + \mathbf{i}_k + \mathbf{i}_{k/2}$. We have $\mathbf{m}_{k/2} = \mathbf{m}_5$, a schoolbook implementation of a multiplication in $\mathbb{F}_{p^5}$ would need $\mathbf{m}_5 = k^2\mathbf{m} = 25\mathbf{m}$, and a square $\mathbf{s}_5 = k(k-1)\mathbf{m} = 20\mathbf{m}$, then with a quadratic extension, $\mathbb{F}_{p^{10}}$ would have $\mathbf{m}_{10} = 3\mathbf{m}_5 = 75\mathbf{m}$ (with Karatsuba) and $\mathbf{s}_{10} = 2\mathbf{m}_5 = 50\mathbf{m}$. The total count would be $32648\mathbf{m} + \mathbf{i}_k + \mathbf{i}_{k/2}$. With optimised Karatsuba-like formulas [30], we would have the lower bound $\mathbf{m}_5 = \mathbf{s}_5 = 13\mathbf{m}$, and $\mathbf{m}_{10} = 39\mathbf{m}$, $\mathbf{s}_{10} = 26\mathbf{m}$, and the final count would be $15784\mathbf{m} + \mathbf{i}_k + \mathbf{i}_{k/2}$.

Curve 5 ($k = 14$, $D = 3$, (6.6), $a = 0$, p of 340 bits, $u = $ 0x2803c0). The optimal ate Miller loop has length $u^2 + up + p^2$. The Miller loop formula is

$$f_{u^2,Q}(P)f_{u,Q}^p(P)\ell_{u^2Q,\pi_p(uQ)}(P)$$

We removed the line $\ell_{u^2Q+\pi_p(uQ),\pi_p^2(Q)}$ as it is a vertical. We have u of 22 bits, $\mathrm{HW}(u) = 6$, 2-NAF(u) of 22 bits, $\mathrm{HW}_{\text{2-NAF}}(u) = 4$, $a = 0$, and a quadratic twist is available. Computing $\pi_p(uQ)$ costs $2\mathbf{f}_{k/2}$. We obtain from eq. (8) $51k\mathbf{m} + 56\mathbf{m}_k + 41\mathbf{s}_k + \mathbf{f}_k + 177\mathbf{m}_{k/2} + 313\mathbf{s}_{k/2} + 2\mathbf{f}_{k/2} + \mathbf{i}_{k/2}$. From [30], we consider the lower (Karatsuba) bound $\mathbf{m}_7 = \mathbf{s}_7 = 22\mathbf{m}$, and $\mathbf{m}_{14} = 3\mathbf{m}_7 = 66\mathbf{m}$, $\mathbf{s}_{14} = 2\mathbf{m}_7 = 44\mathbf{m}$. We obtain $17020\mathbf{m} + \mathbf{i}_7$, where $\mathbf{m}$ is a multiplication in $\mathbb{F}_p$ of 340 bits. With 2-NAF(u) for the second Miller function, the cost is $49k\mathbf{m} + 52\mathbf{m}_k + 41\mathbf{s}_k + \mathbf{f}_k + 157\mathbf{m}_{k/2} + 309\mathbf{s}_{k/2} + 2\mathbf{f}_{k/2} + \mathbf{i}_k + \mathbf{i}_{k/2}$, upper bound $39037\mathbf{m} + \mathbf{i}_k + \mathbf{i}_{k/2}$, lower bound $16200\mathbf{m} + \mathbf{i}_k + \mathbf{i}_{k/2}$.

4.3 Comparison

The five curves of Sec. 4.1 and 4.2 are compared to BN, BLS12, FK12 and KSS16 curves, and modified Cocks-Pinch curves in Table 8. The curves of even embedding degrees $k = 10, 14$ are not competitive by a factor two compared to KSS16 curves (over a 339-bit field, Miller loop in $7691\mathbf{m}$) and BLS12, FK12 curves (over a 446-bit field, Miller loop in $7805\mathbf{m}$ and $7853\mathbf{m}$ resp.), because they have only a quadratic twist, whereas KSS16 curves have a quartic twist and BLS12, FK12 curves have a sextic twist. The curves of prime embedding degrees $k = 11, 13$ are not competitive by a factor four compared to the same curves, because no twist is available.

5 Overview of the 192-bit security level

At the 192-bit security level, we would like to set the constrain

$$7168 \le 384\rho k \le 14336 \ . \tag{9}$$

Table 8. Count for Miller loop cost from [22] for Cocks–Pinch and BN, BLS12, KSS16, and from [18] for FK12 (a.k.a. FM17).

Curve	bits p	Miller loop	final exp.	total
Cocks–Pinch $k = 6$	672	4601**m**	3871**m**	8472**m**
Cocks–Pinch $k = 8$	544	4502**m**	7056**m**	11558**m**
BN	446	11620**m**	5349**m**	16969**m**
BLS12	446	7805**m**	7723**m**	15528**m**
FK12 (FM17)	446	7853**m**	8002**m**	15855**m**
KSS16	339	7691**m**	18235**m**	25926**m**
$k = 11, D = 3, a = 0$	333	29187**m** $+2\mathbf{i}_{11}$		
$k = 11, D = 11, a = 2$	412	25153**m** $+\mathbf{i}_{11}$		
$k = 13, D = 3, a = 0$	310	29919**m** $+2\mathbf{i}_{13}$		
$k = 10, D = 15, a = -3$	446	15784**m** $+\mathbf{i}_{10} + \mathbf{i}_5$		
$k = 14, D = 3, a = 0$	340	16200**m** $+\mathbf{i}_{14} + \mathbf{i}_7$		

With $\rho = 1$ we obtain $k \leq 37$, and with $\rho = 2$ we obtain $k \geq 10$. Curves like Fotiadis–Konstantinou with exactly $\rho = 2$ satisfy (9) for $10 \leq k \leq 18$. No cyclotomic family of embedding degree above 32 satisfying (9) was found. For BN, BLS12, BLS24, KSS16, KSS18, we reproduce in Table 9 the results of Guillevic and Singh [23]: BN with a 1022-bit p, BLS12 with a 1150-bit p, KSS16 with a 766-bit prime p, KSS18 with a 638-bit prime p, BLS24 with a 509-bit prime p. We list in Table 10 seed ranges for $k \in \{14, 15, 20, 21, 27, 28\}$. We also refer to [18] for alternative curves with $\rho = 2$. We leave to future work a complete study of pairing-friendly curves at the 192-bit security level.

Table 9. Seeds at the 192-bit security level from [23].

k	curve	r bits	p bits	p^k bits	seed u	DL cost
12	BN	1022	1022	12255	$-2^{254} + 2^{33} + 2^6$	191
12	BLS12	768	1150	13799	$-2^{192} + 2^{188} - 2^{115} - 2^{110} - 2^{44} - 1$	193
16	KSS16	605	766	12255	$2^{78} - 2^{76} - 2^{28} + 2^{14} + 2^7 + 1$	194
18	KSS18	474	638	11477	$2^{80} + 2^{77} + 2^{76} - 2^{61} - 2^{53} - 2^{14}$	193
24	BLS24	409	509	12202	$-2^{51} - 2^{28} + 2^{11} - 1$ [14]	193

6 Conclusion

Because of the Special Tower Number Field Sieve algorithm, the security of pairing-friendly curves should be reconsidered. We presented a new variant of STNFS for pairing-friendly curves constructed with the Brezing–Weng method, where the characteristic has a polynomial form. It does not apply to the modified Cocks–Pinch curves of [22]. We refine the analysis of Barbulescu, El Mrabet and Ghammam and present an updated short-list of secure pairing-friendly curves at

Table 10. Seeds at the 192-bit security level for $k \in \{14, 15, 20, 21, 27, 28\}$. For $k = 14, 15$ the range of u is such that p is 928-bit long (a smaller p of 920 to 928 bits is possible). For $k = 20$, u is s.t. r is 448-bit long. For $k = 21, 27, 28$, u is s.t. r is 384-bit long.

k	curve D, m, e_0	r bits	p bits	p^k bits	seed u	DL cost
14	Cyclo 1,2,1 (6.3)	620	928	12979– 12992	$u \geq$ 0xc382fe8f05eaf $u \leq$ 0xcb2ff529e85b5	194
15	Cyclo 3,1,1 (BLS-15)	620	928	13906– 13920	$u \leq$ -0x29b3f997f573d609c26f $u \geq$ -0x2c2ecd2df12c9d54ec07 $u \geq$ 0x29b3f997f573d6097e04 $u \leq$ 0x2c2ecd2df12c9d52b8c9	193
20	Cyclo 1,1,1 (6.4)	448	669–670	13371– 13400	$u \geq$ 0xeac0c6e7dd29e3 $u \leq$ 0xfffffffffffd1ed	192
21	Cyclo 3,1,1 (BLS-21)	384	510–511	10691– 10719	-0xf1a1c083 $\geq u \geq$ -0xffff6fd1 0xf1a1ddd7 $\leq u \leq$ 0xfffffccc1	195
27	Cyclo 3,1,1 (BLS-27)	384	426–427	11496– 11524	-0x29487b $\geq u \geq$ -0x2ac5ea 0x2955f1 $\leq u \leq$ 0x2ac66d	212
28	Cyclo 1,1,1 (6.2)	384	510	14243– 14280	0xf1a202f1 $\leq u \leq$ 0xffffd341	208

the 128-bit security level. For embedding degrees from 10 to 16, we obtain curves so that the size of p^k is at least 3663 bits ($k = 11$) and at most 5376 bits (for BLS12 curves). The estimated cost of a DL computation with STNFS for these finite fields is between 2^{128} and 2^{148}. The fastest pairings are obtained with a BLS12 curve or a Fotiadis–Konstantinou–Martindale curve of embedding degree 12, discriminant 3 and twist of degree 6 over a 446-bit prime. The additional curves of this paper have embedding degrees 10, 11, 13 and 14 and a twist of degree 2 for even embedding degrees. It was not sure by how much a prime embedding degree k allows to reduce the total size of p^k: for $k = 11$ the smallest possible p is 333 bit long, and for $k = 13$ p is 310 bit long. Although p is smaller than 446 bits, no twist is available with a prime embedding degree. For this reason, the efficiency of pairings on prime embedding degree curves is not competitive compared to BLS12 and FK12 curves.

References

1. Aranha, D.F., Gouvêa, C.P.L.: RELIC is an Efficient LIbrary for Cryptography. https://github.com/relic-toolkit/relic
2. Arène, C., Lange, T., Naehrig, M., Ritzenthaler, C.: Faster computation of the Tate pairing. Journal of Number Theory **131**(5, Elliptic Curve Cryptography), 842–857 (2011). https://doi.org/10.1016/j.jnt.2010.05.013, http://cryptojedi.org/papers/#edpair
3. Barbulescu, R., Duquesne, S.: Updating key size estimations for pairings. Journal of Cryptology **32**(4), 1298–1336 (Oct 2019). https://doi.org/10.1007/s00145-018-9280-5, https://ia.cr/2017/334
4. Barbulescu, R., El Mrabet, N., Ghammam, L.: A taxonomy of pairings, their security, their complexity. ePrint 2019/485, Sept. 24 (2019), https://ia.cr/2019/485

5. Barbulescu, R., Gaudry, P., Kleinjung, T.: The tower number field sieve. In: Iwata, T., Cheon, J.H. (eds.) ASIACRYPT 2015, Part II. LNCS, vol. 9453, pp. 31–55. Springer, Heidelberg (Nov / Dec 2015). https://doi.org/10.1007/978-3-662-48800-3_2, `https://ia.cr/2015/505`

6. Barreto, P.S.L.M., Lynn, B., Scott, M.: Constructing elliptic curves with prescribed embedding degrees. In: Cimato, S., Galdi, C., Persiano, G. (eds.) SCN 02. LNCS, vol. 2576, pp. 257–267. Springer, Heidelberg (Sep 2003). https://doi.org/10.1007/3-540-36413-7_19

7. Bowe, S.: BLS12-381: New zk-SNARK elliptic curve construction. Zcash blog (March 11 2017), `https://blog.z.cash/new-snark-curve/`

8. Bowe, S., Chiesa, A., Green, M., Miers, I., Mishra, P., Wu, H.: Zexe: Enabling decentralized private computation. ePrint 2018/962 (2018), `https://ia.cr/2018/962`

9. Brezing, F., Weng, A.: Elliptic curves suitable for pairing based cryptography. Des. Codes Cryptography **37**(1), 133–141 (2005). https://doi.org/10.1007/s10623-004-3808-4, `https://ia.cr/2003/143`

10. Chatterjee, S., Menezes, A., Rodríguez-Henríquez, F.: On instantiating pairing-based protocols with elliptic curves of embedding degree one. IEEE Transactions on Computer **66**(6), 1061–1070 (2017). https://doi.org/10.1109/TC.2016.2633340, `https://ia.cr/2016/403`

11. Chatterjee, S., Sarkar, P., Barua, R.: Efficient computation of Tate pairing in projective coordinate over general characteristic fields. In: Park, C., Chee, S. (eds.) ICISC 04. LNCS, vol. 3506, pp. 168–181. Springer, Heidelberg (Dec 2005). https://doi.org/10.1007/11496618_13

12. Chiesa, A., Chua, L., Weidner, M.: On cycles of pairing-friendly elliptic curves. SIAM Journal on Applied Algebra and Geometry **3**(2), 175–192 (2019). https://doi.org/10.1137/18M1173708

13. Costello, C., Lange, T., Naehrig, M.: Faster pairing computations on curves with high-degree twists. In: Nguyen, P.Q., Pointcheval, D. (eds.) PKC 2010. LNCS, vol. 6056, pp. 224–242. Springer, Heidelberg (May 2010). https://doi.org/10.1007/978-3-642-13013-7_14, `https://ia.cr/2009/615`

14. Costello, C., Lauter, K., Naehrig, M.: Attractive subfamilies of BLS curves for implementing high-security pairings. In: Bernstein, D.J., Chatterjee, S. (eds.) IN-DOCRYPT 2011. LNCS, vol. 7107, pp. 320–342. Springer, Heidelberg (Dec 2011), `https://ia.cr/2011/465`

15. Euthereum, Go implementation. `https://github.com/ethereum/go-ethereum/tree/master/crypto/bn256`

16. Foster, K.: HT90 and "simplest" number fields. Illinois J. Math. **55**(4), 1621–1655 (2011), `http://arxiv.org/abs/1207.6099`

17. Fotiadis, G., Konstantinou, E.: TNFS resistant families of pairing-friendly elliptic curves. Theoretical Computer Science **800**, 73–89 (31 December 2019). https://doi.org/10.1016/j.tcs.2019.10.017, `https://ia.cr/2018/1017`

18. Fotiadis, G., Martindale, C.: Optimal TNFS-secure pairings on elliptic curves with composite embedding degree. ePrint 2019/555 (2019), `https://ia.cr/2019/555`

19. Fouotsa, E., El Mrabet, N., Pecha, A.: Computing optimal ate pairings on elliptic curves with embedding degree 9, 15 and 27. ePrint 2016/1187 (2016), `https://ia.cr/2016/1187`

20. Freeman, D., Scott, M., Teske, E.: A taxonomy of pairing-friendly elliptic curves. Journal of Cryptology **23**(2), 224–280 (Apr 2010). https://doi.org/10.1007/s00145-009-9048-z, `https://ia.cr/2006/372`

21. Galbraith, S.: Pairings. In: Blake, I.F., Seroussi, G., Smart, N.P. (eds.) Advances in Elliptic Curve Cryptography, p. 183–214. London Mathematical Society Lecture Note Series, Cambridge University Press (2005). https://doi.org/10.1017/CBO9780511546570.011

22. Guillevic, A., Masson, S., Thomé, E.: Cocks-Pinch curves of embedding degrees five to eight and optimal ate pairing computation. ePrint 2019/431 (2019), https://ia.cr/2019/431

23. Guillevic, A., Singh, S.: On the alpha value of polynomials in the tower number field sieve algorithm. ePrint 2019/885 (2019), https://ia.cr/2019/885

24. ISO: ISO/IEC 15946-5:2017 Information technology – Security techniques – Cryptographic techniques based on elliptic curves – Part 5: Elliptic curve generation, 2 edn. (August 2017), https://www.iso.org/standard/69726.html

25. Joux, A., Pierrot, C.: The special number field sieve in $\mathbb{F}_{p^n}$ - application to pairing-friendly constructions. In: Cao, Z., Zhang, F. (eds.) PAIRING 2013. LNCS, vol. 8365, pp. 45–61. Springer, Heidelberg (Nov 2014). https://doi.org/10.1007/978-3-319-04873-4_3, https://ia.cr/2013/582

26. Kim, T., Barbulescu, R.: Extended tower number field sieve: A new complexity for the medium prime case. In: Robshaw, M., Katz, J. (eds.) CRYPTO 2016, Part I. LNCS, vol. 9814, pp. 543–571. Springer, Heidelberg (Aug 2016). https://doi.org/10.1007/978-3-662-53018-4_20, https://ia.cr/2015/1027

27. Kim, T., Jeong, J.: Extended tower number field sieve with application to finite fields of arbitrary composite extension degree. In: Fehr, S. (ed.) PKC 2017, Part I. LNCS, vol. 10174, pp. 388–408. Springer, Heidelberg (Mar 2017). https://doi.org/10.1007/978-3-662-54365-8_16, https://ia.cr/2016/526

28. Lenstra, A.K., Verheul, E.R.: Selecting cryptographic key sizes. Journal of Cryptology **14**(4), 255–293 (Sep 2001). https://doi.org/10.1007/s00145-001-0009-4

29. Menezes, A., Sarkar, P., Singh, S.: Challenges with assessing the impact of NFS advances on the security of pairing-based cryptography. In: Phan, R.C., Yung, M. (eds.) Mycrypt Conference. LNCS, vol. 10311, pp. 83–108. Springer, Kuala Lumpur, Malaysia (December 1-2 2016). https://doi.org/10.1007/978-3-319-61273-7_5, https://ia.cr/2016/1102

30. Montgomery, P.L.: Five, six, and seven-term Karatsuba-like formulae. IEEE Transactions on Computer **54**, 362–369 (March 2005). https://doi.org/10.1109/TC.2005.49

31. Pereira, G.C., Simplício, M.A., Naehrig, M., Barreto, P.S.: A family of implementation-friendly BN elliptic curves. Journal of Systems and Software **84**(8), 1319 – 1326 (2011). https://doi.org/10.1016/j.jss.2011.03.083, https://ia.cr/2010/429

32. Sakemi, Y., Kobayashi, T., Saito, T.: Pairing-friendly curves. https://tools.ietf.org/html/draft-irtf-cfrg-pairing-friendly-curves-00 (Nov 2019), IETF draft

33. Scott, M., Guillevic, A.: A new family of pairing-friendly elliptic curves. In: Budaghyan, L., Rodríguez-Henríquez, F. (eds.) Arithmetic of Finite Fields. pp. 43–57. Springer, Cham (2018). https://doi.org/10.1007/978-3-030-05153-2_2, https://ia.cr/2018/193

34. Smith, B.: Easy scalar decompositions for efficient scalar multiplication on elliptic curves and genus 2 Jacobians. Contemporary mathematics **637**, 15 (May 2015), https://hal.inria.fr/hal-00874925

35. Wahby, R.S., Boneh, D.: Fast and simple constant-time hashing to the BLS12-381 elliptic curve. IACR TCHES **2019**(4), 154–179 (2019). https://doi.org/10.13154/tches.v2019.i4.154-179