Optimal TNFS-secure pairings on elliptic curves with even embedding degree

Optimal TNFS-secure pairings on elliptic curves with even embedding degree

Georgios Fotiadis[1] and Chloe Martindale[2] *

[1] University of the Aegean, Greece
gfotiadis@aegean.gr
[2] Technische Universiteit Eindhoven, the Netherlands
chloemartindale@gmail.com

Abstract. In this paper we give a comprehensive comparison between pairing-friendly elliptic curves in Jacobi Quartic and Edwards form with quadratic, quartic, and sextic twists. Our comparison looks at the best choices to date for pairings on elliptic curves with even embedding degree on both $\mathbb{G}_1 \times \mathbb{G}_2$ and $\mathbb{G}_2 \times \mathbb{G}_1$ (these are the twisted Ate pairing and the optimal Ate pairing respectively). We apply this comparison to each of the nine possible 128-bit TNFS-secure families of elliptic curves computed by Fotiadis and Konstantinou [14]; we compute the optimal choice for each family together with the fastest curve shape/pairing combination. Comparing the nine best choices from the nine families gives a optimal choice of elliptic curve, shape and pairing (given current knowledge of TNFS-secure families). We also present a proof-of-concept MAGMA implementation for each case. Additionally, we give the first analysis, to our knowledge, of the use of quadratic twists of both Jacobi Quartic and Edwards curves for pairings on $\mathbb{G}_2 \times \mathbb{G}_1$, and of the use of sextic twists on Jacobi Quartic curves on $\mathbb{G}_1 \times \mathbb{G}_2$.

Keywords: TNFS-secure, optimal pairing, twisted Ate pairing, twisted Edwards curves, Jacobi Quartic curves.

1 Introduction

Pairings in cryptography first appeared in 1940 when André Weil showed that there is a way to map points of order r on a supersingular elliptic curve to an element of order r in a finite field; his map became known as the *Weil pairing*. In 1986, Victor Miller [22] gave an algorithm that computes the Weil pairing efficiently, and in 1993, Menezes, Okamoto and Vanstone [23] applied Miller's method to the elliptic curve discrete logarithm problem (ECDLP) for supersingular elliptic curves. They reduced ECDLP for supersingular elliptic curves to the discrete logarithm problem in a finite field (DLP), giving a subexponential

* Author list in alphabetical order; see https://www.ams.org/profession/leaders/culture/CultureStatement04.pdf. This work was supported in part by the Netherlands Organisation for Scientific Research (NWO) under CHIST-ERA USEIT (grant number 651.002.004). Date of this document: October 11, 2018.

attack now known as the MOV-attack. This attack was followed by the more general FR-reduction [15], which can be applied to ordinary elliptic curves (and higher-dimensional abelian varieties) when using a variant of the Weil pairing called the *Tate pairing*. In the early 2000s however, several authors presented secure and efficient pairing-based protocols (see e.g. [6,7,17]) which are now the backbone of privacy-related cryptosystems.

1.1 Pairings on elliptic curves

Let $\mathbb{G}_1, \mathbb{G}_2$, and $\mathbb{G}_T$ be cyclic groups of prime order r and assume that the discrete logarithm problem is intractable in all three groups. An *abstract pairing* is a bilinear, non-degenerate, efficiently computable map of the form: $\widehat{e} : \mathbb{G}_1 \times \mathbb{G}_2 \to \mathbb{G}_T$. We call $\mathbb{G}_1$ and $\mathbb{G}_2$ the *source groups* and $\mathbb{G}_T$ the *target group*. When $\mathbb{G}_1 \neq \mathbb{G}_2$ the pairing is called *asymmetric*, otherwise it is called *symmetric*.

Let E be an ordinary elliptic curve defined over a prime field $\mathbb{F}_p$ and let r be largest prime such that $r|\#E(\mathbb{F}_p)$. The minimal integer k for which all the r-th roots of unity are contained in $\mathbb{F}_{p^k}$ is called the *embedding degree* of E. For all pairings on elliptic curves that are currently used in cryptography, the source groups $\mathbb{G}_1$ and $\mathbb{G}_2$ are r-order subgroups of $E(\mathbb{F}_{p^k})$ and the target group $\mathbb{G}_T$ is the subgroup $\mu_r \subseteq \mathbb{F}_{p^k}^*$ of r^{th} roots of unity. (Typically $\mathbb{G}_1$ is in fact contained in $E(\mathbb{F}_p)$). That is, a pairing of elliptic curves is a map: $\widehat{e} : E(\mathbb{F}_{p^k})[r] \times E(\mathbb{F}_{p^k})[r] \to \mu_r \subset \mathbb{F}_{p^k}^*$. The most widely used pairings on ordinary elliptic curves are the *Tate pairing* and its variants and the *Ate pairing* and its variants. All of these different types of pairings can be efficiently computed using variants of Miller's algorithm ([22], c.f. Algorithm 1).

1.2 Attacks on pairings

For a sufficiently generic elliptic curve $E/\mathbb{F}_p$, the complexity of ECDLP in any r-order subgroup of $E(\mathbb{F}_{p^k})$ is $O(\sqrt{r})$, due to Pollard's rho algorithm. The complexity of DLP in the multiplicative group $\mathbb{F}_{p^k}^*$, however, depends both on the factorisation of k and on how the prime p is constructed. In the case of pairing-friendly curves, we may assume that the prime p is large (at least 256-bits) and that it is derived from the evaluation of a polynomial with degree greater that 2. The asymptotic complexity of DLP in $\mathbb{F}_{p^k}^*$ is then

$$L_N[c, \ell] = \exp\left[\big(c + o(1)\big)\big(\ln N\big)^{\ell}\big(\ln \ln N\big)^{1-\ell}\right], \tag{1}$$

for some $c, \ell \in \mathbb{R}$, with $\ell \in [0, 1]$, $c > 0$ and $N = p^k$.

When k is prime, the asymptotic complexity of DLP in $\mathbb{F}_{p^k}^*$ is $L_N[1/3, 1.923]$; the best known attack is the number field sieve (NFS) method. For composite embedding degrees, Kim and Barbulescu's [18] improvements on the tower number field sieve (TNFS) method have reduced complexity of DLP in $\mathbb{F}_{p^k}^*$ from $L_N[1/3, 1.923]$ to $L_N[1/3, 1.529]$. These new improvements have immediate consequences on the selection of the extension fields $\mathbb{F}_{p^k}$. Fotiadis and Konstantinou [14] present a summary of (new) recommendations of pairing-friendly elliptic

curves that are resistant against the new TNFS attacks, for many different embedding degrees.

1.3 Our contributions

The main goal of this paper to present the best choice of pairing and of elliptic curve that gives 128-bit security according to the state-of-the-art. As families of TNFS-secure elliptic curves are already presented in [14], our main contribution is a comprehensive comparison of pairings and elliptic curve shapes and the consequent selection of a curve from the available families. To our knowledge, this is the first suggestion of a 128-bit secure pairing-friendly elliptic curve that takes into account the latest attacks described above.

Our comparison takes into account competing candidates for the most efficient pairings (Section 2) and competing curve shapes for the most efficient curve arithmetic (Section 3). In section 4 we combine the discussion in Sections 2 and 3 to compute the optimal elliptic curve and pairing choice for a 128-bit security level.

Additionally, we present a new analysis for efficient curve arithmetic in the case of quadratic twists of Edwards curves and Jacobi Quartic curves for the Ate pairing (Section 3.2 and Section 3.3), and of sextic twists of Jacobi Quartic curves (Section 3.2).

For every case that we consider we present an implementation in MAGMA, available at `www.martindale.info/research`.

2 Secure and efficient pairings on elliptic curves

In all that follows, we write $\mathbb{F}_p$ for a finite field of prime order, we write $E/\mathbb{F}_p$ for an elliptic curve defined over this field, we write r for a large prime dividing $\#E(\mathbb{F}_p)$, and k for the embedding degree of E with respect to r. For $\mathbb{G}_1$ and $\mathbb{G}_2$ distinct r-order subgroups of $E(\mathbb{F}_{p^k})$ and $\mathbb{G}_T$ an r-order subgroup of $\mathbb{F}^*_{p^k}$, there is an asymmetric pairing $\widehat{e} : \mathbb{G}_1 \times \mathbb{G}_2 \to \mathbb{G}_T$. Good choices of p, E, r, and k give rise to pairing-based cryptographic protocols, the security and efficiency of which depends on $\widehat{e}$ and is discussed in detail below.

2.1 Constructing explicit pairing-friendly elliptic curves

We make (severe) restrictions on the families of elliptic curves we consider for specifically pairing-based applications; these families are commonly referred to as *pairing-friendly* in the literature. In all that follows p, E, r, and k are as in the preamble to this section.

Definition 1. *We say that $E/\mathbb{F}_p$ is* pairing-friendly *(c.f. [13]) if:*

1. *There is a sufficiently large choice for r such that the discrete logarithm problem in $\mathbb{G}_1$ and $\mathbb{G}_2$ is computationally hard.*
2. *The ρ-*value *$\rho = \log(p)/\log(r)$ is close to 1.*

3. *The discrete logarithm problem in $\mathbb{G}_T$ is as computationally hard as the discrete logarithm problem in $\mathbb{G}_1$ and $\mathbb{G}_2$.*
4. *Computations in $\mathbb{G}_T \subset \mathbb{F}_{p^k}$ are efficient.*

The precise meanings of 'computationally hard', 'close to 1', and 'efficient' depend heavily on the desired security parameters.

Points (1) and (3) above are to ensure the security of a protocol based on a pairing using such an elliptic curve. The recent TNFS attacks [18] on the discrete logarithm problem for pairing-friendly elliptic curves changed the meaning of 'computationally hard' in this definition, so that the security analyses of papers pre-dating this attack are now too weak. Fotiadis and Konstantinou [14] gave examples of families of pairing-friendly elliptic curves that are resistant to the TNFS attacks.

We aim to present which choices can be made with regards to points (2) and (4). Point (2) gives control on the bandwidth of the pairing computation and point (4) ensures efficient arithmetic in $\mathbb{G}_T$, normally achieved by using small k. Each family of pairing-friendly elliptic curves presented in [14] is given together with the ρ-value and the embedding degree, so the remaining work is to assess the trade-off.

For the construction of pairing-friendly elliptic curves we use the notion of *complete polynomial families*, introduced in [8]. For a given embedding degree $k > 0$ the elliptic curve parameters p, t and r are described as polynomials $p(x), t(x), r(x) \in \mathbb{Q}[x]$ respectively, such that $4p(x) - t(x)^2 = Dy(x)^2$, where $D > 0$ is the square-free complex multiplication (CM) discriminant and $y(x) \in \mathbb{Q}[x]$. These polynomials must satisfy $\Phi_k(t(x) - 1) \equiv 0 \bmod r(x)$, where $\Phi(x)$ is the k^{th} cyclotomic polynomial. This implies that $t(x) - 1$ is a primitive k^{th} root of unity in the number field $\mathbb{Q}[x]/\langle r(x)\rangle$. An additional constraint for these polynomials is that $p(x) + 1 - t(x) \equiv 0 \bmod r(x)$. This ensures that the order of the curve has a polynomial representation as $\#E(\mathbb{F}_{p(x)}) = h(x)r(x)$.

We can generate elliptic curve parameters by evaluating the polynomial family at some integer x_0, such that $r = r(x_0)$ and $p = p(x_0)$ are both primes and $t = t(x_0) \leq 2\sqrt{p}$ (Hasse bound). We further need to verify that these choices respect the the recommendations of Definition 1. Particularly, for 128-bit security in $\mu_r \subset \mathbb{F}_{p^k}^*$ with composite k, Equation (1) implies that the extension field size $N = k \log p$ should be around 4400-bits. Given the embedding degree k and a desired security level S, we need to find a complete family of pairing-friendly elliptic curves with CM discriminant D and $\rho = N/(2kS)$.

2.2 Tate pairing

The reduced Tate pairing is a common example of a pairing that can be used in cryptographic algorithms. The (non-reduced) Tate pairing is something more general which we do not address here as our main focus is on applications. In the reduced Tate pairing we have $\mathbb{G}_1 = E(\mathbb{F}_p)[r]$, $\mathbb{G}_2 = E(\mathbb{F}_{p^k})[r]$, and $\mathbb{G}_T = \mu_r$, the group of r-th roots of unity in $\mathbb{F}_{p^k}^*$, with p, E, r, and k as defined at the beginning of Section 2. For a point $P \in E(\mathbb{F}_q)[r]$ define the function f_r, P to be

the unique function on E with divisor $r(P) - r(P_\infty)$ (existence and uniqueness follows from eg. [25, Corollary 3.5]). The *reduced Tate pairing* is the (p, E, r, k-dependent) non-degenerate bilinear map $\widehat{t} : E(\mathbb{F}_p)[r] \times E(\mathbb{F}_{p^k})[r] \to \mu_r$ given by $(P, Q) \mapsto f_{r,P}(Q)^{\frac{p^k-1}{r}}$. The value $f_{r,P}(Q)$ can be computed efficiently using *Miller's algorithm* (see [22], or Algorithm 1 below).

Definition 2. *Let R and S be points on the elliptic curve E. We denote by $h_{R,S}$ the rational function with divisor $(R) + (S) - (S + R) - (P_\infty)$.*

As explained in detail in [13,22], Miller's algorithm computes $f_{r,P}(Q)$ iteratively from $h_{R,S}(Q)$; see Algorithm 1.

Algorithm 1 Miller's algorithm

Input: $P \in E(\mathbb{F}_p)[r]$, $Q \in E(\mathbb{F}_{p^k})[r]$, $r = (1, r_{n-2}, \ldots, r_1, r_0)_2$.

Output: The reduced Tate pairing $f_{r,P}(Q)^{\frac{p^k-1}{r}}$ of P and Q.

1: Set $f \leftarrow 1$ and $R \leftarrow P$.
2: **for** $i = n - 2$ to 0 **do**
3: $\quad f \leftarrow f^2 \cdot h_{R,R}(Q)$
4: $\quad R \leftarrow 2R$
5: $\quad$ **if** $r_i = 1$ **then**
6: $\quad\quad f \leftarrow f \cdot h_{R,P}(Q)$
7: $\quad\quad R \leftarrow R + P$
8: **return** $f^{\frac{p^k-1}{r}}$

Remark 1.
- The number of field operations needed to compute $h_{R,S}$ can be decreased by using elliptic curves written in a special form, such as Jacobi quartic form [16] or Edwards form [4]; see Section 3.
- The pairings that are presented in the following sections were partly introduced in order to reduce the number of iterations in Algorithm 1.
- The number of times that Steps 6-7 of Algorithm 1 are performed depends on the Hamming weight of r, which is minimized using Algorithm 2.
- Step 8 computes the power $(p^k - 1)/r$ of an element in $\mathbb{F}^*_{p^k}$. This is known as the *final exponentiation*; see e.g. [11], c.f. Section 2.4.

Algorithm 2 Finding suitable elliptic curve parameters for the Tate pairing

Input: A complete family of pairing-friendly elliptic curves: $[p(x), t(x), r(x)]$; security level: S; integers a, b, such that the family is integer-valued for every $x \equiv b \pmod a$.

Output: Optimal elliptic curve parameters p, t, r.

1: Set $n_{\min} \leftarrow (2S - \log \mathrm{lc}(r))/\deg r$ and $n_{\max} \leftarrow n_{\min} + (1/\deg r)$ and $w \leftarrow 2S$
2: **for** $i = 2^{n_{\min}}$ to $2^{n_{\max}}$ **do**
3: $\quad$ **if** $i \equiv b \pmod a$ **then**
4: $\quad\quad x_0 \leftarrow i$; $r \leftarrow r(x_0)$
5: $\quad\quad$ **if** r: ir prime and $\log r = 2S$ **then**
6: $\quad\quad\quad p \leftarrow p(x_0)$
7: $\quad\quad\quad$ **if** p: ir prime and $\mathrm{wt}(r) < w$ **then**
8: $\quad\quad\quad\quad t \leftarrow t(x_0)$; $w = \mathrm{wt}(r)$
9: **return** $[p, t, r]$

2.3 Optimal Ate pairing

As the name suggests, the optimal Ate pairing is a descendant of the (reduced) Ate pairing, which we introduce first for the benefit of the reader. In the Ate pairing, we take

$$\mathbb{G}_1 = E(\overline{\mathbb{F}_p})[r] \cap \ker(\pi_p - [1]) \quad \text{and} \quad \mathbb{G}_2 = E(\overline{\mathbb{F}_p})[r] \cap \ker(\pi_p - [p]),$$

where π_p denotes the p-power Frobenius endomorphism on E. Note that $\mathbb{G}_1 = E(\mathbb{F}_p)[r]$ and $\mathbb{G}_2 \subseteq E(\mathbb{F}_{p^k})[r]$. For a point $P \in \mathbb{G}_2$ and a positive integer T, we define the function $f_{T,P}$ to be the unique function on E with divisor $T(P) - ([T]P) - (T-1)P_\infty$ (as before, existence and uniqueness follows from e.g. [25, Corollary 3.5]). The *reduced Ate pairing* is the (p, E, r, k-dependent) non-degenerate bilinear map $\widehat{a} : \mathbb{G}_2 \times \mathbb{G}_1 \to \mu_r$ given by $(P, Q) \mapsto f_{T,P}(Q)^{\frac{p^k-1}{r}}$, where $T = t - 1$ and t is the trace of Frobenius.

In [26], Vercauteren presented the *optimal Ate pairing* which always gives the minimal number of iterations compared to other variants of the Ate pairing (e.g. [28], [21], [19]). Note that it may still be slower than the Tate pairing.

Let E, p, r, and k be as above. Recall that pairing-friendly curves satisfy $\Phi_k(p) \equiv 0 \bmod r$, where Φ_k is the k^{th} cyclotomic polynomial. We consider the $\varphi(k)$-dimensional lattice L (spanned by the rows):

$$L = \begin{pmatrix} r & 0\,0 \ldots 0 \\ -p & 1\,0 \ldots 0 \\ -p^2 & 0\,1 \ldots 0 \\ \vdots & \vdots\,\vdots\,\ddots\,\vdots \\ -p^{\varphi(k)-1} & 0\,0 \ldots 1 \end{pmatrix} \tag{2}$$

and let $V = [c_0, c_1, \ldots, c_{\varphi(k)-1}]$ be the shortest vector of this lattice. By [26, Theorem 7], the shortest vector V of the lattice L satisfies

$$\|V\|_2 \geq \frac{r^{1/\varphi(k)}}{\|\Phi_k\|_2} \qquad \text{and} \qquad \|V\|_\infty \leq \frac{r^{1/\varphi(k)}}{(\varphi(k)-1)\|\Phi_k\|_\infty},$$

where $\|\cdot\|_2$ and $\|\cdot\|_\infty$ are the square and infinite norms respectively. The *optimal Ate pairing* is defined as the bilinear, non-degenerate map $\widehat{a}_{\text{o}} : \mathbb{G}_2 \times \mathbb{G}_1 \to \mu_r \subset \mathbb{F}_{p^k}^*$ given by

$$(P, Q) \mapsto \left[\prod_{i=0}^{\varphi(k)-1} f_{c_i, P}^{p^i}(Q) \cdot \underbrace{\prod_{i=0}^{\varphi(k)-2} h_{[s_{i+1}]P, [c_i p^i]P}(Q)}_{H} \right]^{\frac{p^k-1}{r}},$$

where $h_{R,S}(Q)$ is as defined in Definition 2 and the values s_i are obtained by the relation: $s_i = \sum_{j=i}^{\varphi(k)-1} c_j p^j$. By [26] this choice of the coordinates c_i ensures the non-degeneracy property of the above pairing. Miller's algorithm can be adapted from Algorithm 1 to compute the optimal Ate pairing.

Algorithm 3 Miller's algorithm (optimal Ate pairing)

Input: $P \in \mathbb{G}_2$, $Q \in \mathbb{G}_1$, $V = [c_0, c_1, \ldots, c_{\varphi(k)-1}]$, H.
Output: The reduced optimal Ate pairing $\widehat{a}_o$ of P and Q.

for $j = 0$ to $\varphi(k) - 1$ **do**
  Set $n \leftarrow \lfloor \log_2 c_j \rfloor$, $f \leftarrow 1$, $R \leftarrow P$, $v_j \leftarrow (1, T_{n-2}, T_{n-1}, \ldots, T_1, T_0)_2$
  for $i = n - 2$ to 0 **do**
    $f \leftarrow f^2 \cdot h_{R,R}(Q)$
    $R \leftarrow 2R$
    if $T_i = 1$ **then**
      $f \leftarrow f \cdot h_{R,P}(Q)$
      $R \leftarrow R + P$
$f \leftarrow f \cdot H$
return $f^{\frac{p^k-1}{r}}$

Note also that the second product H in the above formula depends on the points P, Q which are fixed during the pairing computation so can be precomputed. Furthermore, we need the coordinates c_i to have the smallest possible Hamming weight.

Algorithm 4 Suitable elliptic curve parameters for the optimal Ate pairing

Input: A complete family of pairing-friendly elliptic curves: $[p(x), t(x), r(x)]$; security level: S; integers a, b, such that the family is integer-valued for every $x \equiv b \pmod{a}$; the shortest vector $V = [c_0, c_1, \ldots, c_{\varphi(k)-1}]$ of the lattice L.
Output: Optimal elliptic curve parameters p, t, r.

Set $n_{\min} \leftarrow (2S - \log \mathrm{lc}(r))/\deg r$ and $n_{\max} \leftarrow n_{\min} + 1/\deg r$ and $w \leftarrow 2S$
for $i = 2^{n_{\min}}$ to $2^{n_{\max}}$ **do**
  if $i \equiv b \pmod{a}$ **then**
    $x_0 \leftarrow i$; $r \leftarrow r(x_0)$; $p \leftarrow p(x_0)$; $t \leftarrow t(x_0)$;
    if r: ir prime and p: ir prime **then**
      $\mathrm{wt}(V) \leftarrow \mathrm{wt}(c_0) + \mathrm{wt}(c_1) + \ldots + \mathrm{wt}(c_{\varphi(k)-1})$
      if $\mathrm{wt}(V) < w$ **then**
        $w \leftarrow \mathrm{wt}(V)$
return $[p, t, r]$

The total number of iterations in this case is: $b_c = \sum'_{i=0}^{\varphi(k)-1} \log c_i = \log \prod'_{i=0}^{\varphi(k)-1} c_i$, where the sum and product run over all $i = 0, 1, \ldots, \varphi(k) - 1$ such that $c_i \neq 0$. Clearly, every c_i contributes in Miller's loop, as long as $c_i \notin \{0, \pm 1\}$ (otherwise, $f_{c_i,Q}(P) = 1$). According to [26], the total number of iterations in Miller's loop cannot be less than $\log r/\varphi(k)$. Given a complete family $[p(x), t(x), r(x)]$ of pairing-friendly elliptic curves with embedding degree k, the process for generating optimal elliptic curve parameters for the reduced optimal Ate pairing is described in Algorithm 4.

Twisted Ate Pairing. The downside to the (optimal) Ate pairing is that most of the operations have to occur in extension fields. However, as observed in [21], if an elliptic curve $E/\mathbb{F}_p$ has a twist (definition recalled below), we can essentially switch $\mathbb{G}_1$ and $\mathbb{G}_2$.

Definition 3. *Let $E/\mathbb{F}_p$ be an elliptic curve over a finite field $\mathbb{F}_p$. A* twist *of E is an elliptic curve E' that is $\overline{\mathbb{F}_p}$-isomorphic to E. Suppose that the isomorphism is defined over $\mathbb{F}_{p^k}$ (but not over any subfield) and that E' is defined over $\mathbb{F}_{p^e}$, where $e|k$, and not over any subfield. We say that E' is a* degree k/e twist of E.

Suppose now that E has a twist of degree $\delta|k$, let $e = k/\delta$, and let $T_e = T^e \pmod r$. Then there exists a non-degenerate bilinear pairing that is referred to as the *twisted Ate pairing* $\widehat{a}_e : \mathbb{G}_1 \times \mathbb{G}_2 \to \mu_r$ given by $(P, Q) \mapsto f_{T_e,P}(Q)^{\frac{p^k-1}{r}}$. In particular, Algorithm 1 for the reduced Tate pairing with r replaced by T_e computes $\widehat{a}_e$. In particular, if E has a (non-trivial) twist, then the twisted Ate pairing has lower cost than the Tate pairing. As we restrict our attention to elliptic curves with even embedding degree, every curve has at least a quadratic twist. The process of generating optimal pairing-friendly elliptic curve parameters is the same as Algorithm 2, except that we are looking for the smallest Hamming weight of T_e.

Remark 2. By construction the value $T = t - 1$ is a primitive k^{th}-root of unity modulo r. When k is even and $\delta = 2$, i.e. the case of quadratic twists, we have $e = k/2$. In this case: $T_e \equiv T^e \equiv T^{k/2} \equiv -1 \equiv (r - 1) \pmod r$. This implies that $\log T_{k/2} \approx \log r$ and in particular $\mathrm{wt}(T_e) = \mathrm{wt}(r) - 1$. Hence the complexity of the Tate and twisted Ate pairings should be roughly the same in this case. However, if there exist quartic or sextic twists, the complexity of the twisted Ate pairing will be strictly better than the Tate pairing. For this reason, from this point on we consider only the optimal Ate pairing (as the most efficient pairing on $\mathbb{G}_2 \times \mathbb{G}_1$) and the twisted Ate pairing (as the most efficient pairing on $\mathbb{G}_1 \times \mathbb{G}_2$).

2.4 Final exponentiation

As we have already discussed, the efficiency of pairing calculations heavily relies on the number of iterations in Miller's loop. Another costly part of the pairing calculations is the final step of Miller's algorithm, the final exponentiation. It comprises of raising an element f in an extension field $\mathbb{F}_{p^k}$ of $\mathbb{F}_p$ to the power $(p^k - 1)/r$. Experiments in MAGMA indicate that it is possible to reduce the cost of the final exponentiation by applying several tricks that will shorten the exponent $(p^k - 1)/r$ (see e.g. [11]).

We give a brief description of the final exponentiation we used in our experiments. Let $[p(x), t(x), r(x)]$ be a family of pairing-friendly elliptic curves with even embedding degree k and let $x_0 \in \mathbb{Z}$, such that $p = p(x_0), r = r(x_0)$ are primes and $t = t(x_0)$ is the trace of Frobenius. Write $(p^k - 1)/r$ as:

$$e = \left(p^{k/2} - 1\right)\left[\frac{p^{k/2}+1}{\Phi_k(p)}\right]\left[\frac{\Phi_k(p)}{r}\right], \quad \text{where} \quad \frac{\Phi_k(p)}{r} = \sum_{i=0}^{\varphi(k)-1} \lambda_i p^i,$$

for some $\lambda_i \in \mathbb{Q}$. Then f^e is equivalently written as:

$$f^{\frac{p^k-1}{r}} = f^{\left(p^{k/2}-1\right)\frac{p^{k/2}+1}{\Phi_k(p)}\left(\lambda_0+\lambda_1 p+\ldots+\lambda_{\varphi(k)-1}p^{\varphi(k)-1}\right)}.$$

The first two exponentiations can be computed using the MAGMA's `Frobenius` function. The final step is to raise a value to $\lambda_0 + \lambda_1 p + \ldots + \lambda_{\varphi(k)-1}p^{\varphi(k)-1}$. This can be done by using the `Frobenius` function each time we need to raise to p^i and simple arithmetic operations when raising to λ_i. We point out that our final exponentiation procedures are not necessarily optimal.

3 Efficient arithmetic on elliptic curves

Let E, p, r, and k be as given at the beginning of Section 2. As $p \neq 2, 3$, it is possible to write E in the form $E/\mathbb{F}_p : y^2 = x^3 + Ax + B$, where A, $B \in \mathbb{F}_p$. This is a *short Weierstrass equation* for E. For certain choices of curve families, it is possible to rewrite the Weierstrass equation in an equivalent representation where point operations such as addition and doubling can be performed with less operations in $\mathbb{F}_p$. To our knowledge, the most competitive elliptic curve forms with respect to efficient arithmetic (for even embedding degree) are the Jacobi quartic form [16] and (twisted) Edwards form [12]. In this section we recall the definition of these special curve forms and the basic arithmetic of curve points which are needed in pairing computations, namely addition and doubling. We now set some notation:

- $\mathbf{s}_m$: time required to square an $\mathbb{F}_{p^m}$-element.
- $\mathbf{m}_m$: time required to multiply together two $\mathbb{F}_{p^m}$-elements.
- $\mathbf{mc}_m$: time required to multiply by a (small) constant in $\mathbb{F}_{p^m}$.

3.1 Twists of elliptic curves

Many authors have given ways to improve the performance of pairing computations via twists. There is of course the twisted Ate pairing that we have already discussed, but there also are three further improvements. All of these improvements are using the fact (see e.g. [24]) that, for a pairing-friendly elliptic curve $E/\mathbb{F}_p$ and $Q \in E(\mathbb{F}_{p^k})$, if E has a degree δ twist $E'/\mathbb{F}_{p^{k/\delta}}$, then with no loss of security we can take for the point Q the image of a point $Q' \in E'(\mathbb{F}_{p^{k/\delta}})$ under the twist isomorphism $\phi : E' \to E$.

Speed-up (1). When computing the optimal Ate pairing via Miller's algorithm, the basic double and add operations are performed in $E(\mathbb{F}_{p^k})$. However, if the curve has a degree δ twist E' as above then the operations can be performed instead in $E'(\mathbb{F}_{p^{k/\delta}})$. If E and E' can be written in the same form (i.e. both in Jacobi Quartic form or both in Edwards form), then the map ϕ is (usually) simple so does not add to the operation count. The map differs between different curve shapes; details are given on a case-by-case basis below.

Speed-up (2). Let E and E' be as above. For both the optimal Ate pairing and the twisted Ate pairing, in each iteration of the Miller loop we are required to compute the Miller function $h_{P_1,P_2}(Q)$ on E at least once. For optimal Ate, we have $P_1, P_2 \in \mathbb{G}_2$ and $Q \in \mathbb{G}_1$, and for twisted Ate, we have $P_1, P_2 \in \mathbb{G}_1$ and $Q \in \mathbb{G}_2$. Suppose that ω generates $\mathbb{F}_{p^k}$ over $\mathbb{F}_{p^{k/\delta}}$. If we take for the point(s) in $\mathbb{G}_2$ the image under ϕ of (a) point(s) in $E'(\mathbb{F}_{p^{k/\delta}})$, we can write $h_{P_1,P_2}(Q)$ as $h_{P_1,P_2}(Q) = h_1\omega + \cdots + h_\delta\omega^\delta$, that is, as an element of a δ-dimensional vector space over $\mathbb{F}_{p^{k/\delta}}$. The computation of each h_i is then a computation in $\mathbb{F}_{p^{k/\delta}}$. More details on a case-by-case basis are given below.

Speed-up (3). Let E and E' be as above. In some cases, the Miller function h has a denominator that lies in a subfield of $\mathbb{F}_{p^k}$ and hence goes to 1 in the final exponentiation step of Miller's algorithm. In some cases, in the numerator some of the h_i are zero, where the h_i are as given in Speed-up (2). If h has n non-zero coefficients h_i, then the multiplication by h with a generic element of $\mathbb{F}_{p^k}$ (as occurs at least once in every iteration of the Miller loop) can be performed in $n\delta$ $\mathbb{F}_{p^{k/\delta}}$-multiplications. If $n < \delta$, then this costs less than a generic multiplication in $\mathbb{F}_{p^k}$. More details on a case-by-case are given below.

3.2 Jacobi Quartic Curves

A Jacobi quartic curve over a prime field $\mathbb{F}_p$ is described by the equation:

$$E_J/\mathbb{F}_p : Y^2Z^2 = dX^4 + 2\mu X^2Z^2 + Z^4, \tag{3}$$

where $d, \mu \in \mathbb{F}_p$ and $d \neq 0$. The neutral element of the group of rational points is $[0:1:0]$. By [5], if an elliptic curve $E/\mathbb{F}_p$ has a rational point of order 2, then it can be written in the form given in Equation (3). The isomorphism from the short Weierstrass equation to the Jacobi Quartic form is also given in [5].

For efficient arithmetic, in [16] it is recommended to use the extended projective representation of points, namely $[X:Y:T:Z]$, where $T = X^2/Z$. Given two points in extended projective representation, $P_1 = [X_1:Y_1:T_1:Z_1]$ and $P_2 = [X_2:Y_2:T_2:Z_2]$, we can calculate their sum $P_3 = [X_3:Y_3:T_3:Z_3]$ by the formulas:

$$\begin{aligned}
X_3 &= (X_1Y_2 - Y_1X_2)(T_1Z_2 - Z_1T_2),\\
Y_3 &= (T_1Z_2 + Z_1T_2 - 2X_1X_2)(Y_1Y_2 - 2\mu X_1X_2 + Z_1Z_2 + dT_1T_2) - Z_3,\\
T_3 &= (T_1Z_2 - Z_1T_2)^2,\\
Z_3 &= (X_1Y_2 - Y_1X_2)^2.
\end{aligned}$$

For the doubling process, given a point $P_1 = [X_1:Y_1:T_1:Z_1]$, we can calculate the point $2P = [X_3:Y_3:T_3:Z_3]$ via:

$$\begin{aligned}
X_3 &= 2X_1Y_1(2Z_1^2 + 2\mu X_1^2 - Y_1^2),\\
Y_3 &= 2Y_1^2(Y_1^2 - 2\mu X_1^2) - (2Z_1^2 + 2\mu X_1^2 - Y_1^2)^2,\\
T_3 &= (2X_1Y_1)^2,\\
Z_3 &= (2Z_1^2 + 2\mu X_1^2 - Y_1^2)^2.
\end{aligned}$$

Using Wang, Wang, Zhang, and Li's recommendations [27], point addition costs $16\mathbf{m} + 1\mathbf{s} + 4\mathbf{mc}$ and point doubling costs $4\mathbf{m} + 8\mathbf{s} + 1\mathbf{mc}$. Here, $\mathbf{m}$ denotes the cost of the multiplication of two elements, $\mathbf{s}$ denotes the cost of squaring an element, and $\mathbf{mc}$ denotes the cost of the multiplication of an element with a constant value; all elements are considered to lie in the field of definition of the points P_1 and P_2.

Quadratic twists of Jacobi Quartic Curves. All the curves that we consider in this paper have even embedding degree, and hence admit quadratic twists. Let $\omega \in \mathbb{F}_{p^k} \setminus \mathbb{F}_{p^{k/2}}$, and define $E_{\mathrm{J}}^{\omega}/\mathbb{F}_{p^{k/2}} : Y^2Z^2 = d\omega^4X^4 + 2\mu\omega^2X^2Z^2 + Z^4$. The curve E_{J}^{ω} is a quadratic twist of E_{J} via the isomorphism

$$\phi : [X : Y : Z] \to [\omega X : Y : Z]. \tag{4}$$

We can use this isomorphism in the three ways described in the Section 3.1, which we summarize following [27].

Pairings on $\mathbb{G}_1 \times \mathbb{G}_2$. For pairings on $\mathbb{G}_1 \times \mathbb{G}_2$ (such as the twisted Ate pairing), Wang, Wang, Zhang, and Li [27] show that the function $h_{P_1,P_2}(Q)$, where $P_1, P_2 \in \mathbb{G}_1$ and $Q \in \mathbb{G}_2$, can be computed in time $k\mathbf{m}_1$, and that the result is a general element of $\mathbb{F}_{p^k}$ (i.e. has no zero coefficients as a vector over $\mathbb{F}_{p^{k/2}}$). The total time for the doubling steps of Miller (i.e. steps 3 and 4 of Algorithm 1) is therefore

$$1\mathbf{m}_k + 1\mathbf{s}_k + (4+k)\mathbf{m}_1 + 8\mathbf{s}_1 + 1\mathbf{mc}_1,$$

and for the addition steps of Miller (i.e. steps 6 and 7 of Algorithm 1) is therefore

$$1\mathbf{m}_k + (16+k)\mathbf{m}_1 + 1\mathbf{s}_1 + 4\mathbf{mc}_1.$$

Pairings on $\mathbb{G}_2 \times \mathbb{G}_1$. For pairings on $\mathbb{G}_2 \times \mathbb{G}_1$ (such as the optimal Ate pairing), the necessary formulas do not to our knowledge appear in the literature, so for completeness we include them here.

Speed-up (1) of Section 3.1 clearly applies; suppose that we wish to compute the optimal Ate pairing of $Q \in \mathbb{G}_1$ and $P \in \mathbb{G}_2$ where $P = \phi(P')$ and $P' \in E_{\mathrm{J}}^{\omega}(\mathbb{F}_{p^{k/2}})$. Every $R \in \mathbb{G}_2$ appearing in the Miller function $h_{R,R}(Q)$ or $h_{R,P}(Q)$ is a multiple of P, and hence $R' = \phi^{-1}(R) \in E_{\mathrm{J}}^{\omega}(\mathbb{F}_{p^{k/2}})$. Therefore, every point doubling/addition $R + S$, where $S = R$ or P, can be computed in $\mathbb{F}_{p^{k/2}}$ via ϕ as $R + S = [\omega X_T, Y_T, Z_T]$ where $T = \phi^{-1}(R) + \phi^{-1}(S)$. Thus, using [27], point doubling takes $4\mathbf{m}_{k/2} + 8\mathbf{s}_{k/2} + 1\mathbf{mc}_{k/2} = k^2\mathbf{m}_1 + 2k^2\mathbf{s}_1 + \frac{k^2}{4}\mathbf{mc}_1$, and point addition takes $16\mathbf{m}_{k/2} + 1\mathbf{s}_{k/2} + 4\mathbf{mc}_{k/2} = 4k^2\mathbf{m}_1 + \frac{k^2}{4}\mathbf{s}_1 + k^2\mathbf{mc}_1$.

For speed-up (2) of Section 3.1 we start from the formula for the Miller function $h_{P_1,P_2}(Q)$ given in [27] and apply it to the case that $P_1 = \phi(P_1')$ and $P_2 = \phi(P_2')$. To this end, let (x_Q, y_Q) be the affine coordinates of Q, set $P_1 = [X_1 : Y_1 : Z_1 : T_1]$, and set $P_2 = [X_2 : Y_2 : Z_2 : T_2]$. By [27], the Miller function is given by

$$h_{P_1,P_2}(Q) = \frac{x_Q^2}{N(y_Q + 1 - cx_Q^2)}(\eta N + (\theta + a)M_1 + M_3), \tag{5}$$

where

$$\eta = \frac{y_Q + 1 + ax_Q^2}{x_Q^3} \qquad \text{and} \qquad \theta = \frac{1 + y_Q}{x_Q^2}$$

and N, M_1, M_3 are calculated by:

$$N = \begin{cases} X_1X_2((T_1Y_2 - Y_1T_2) + (T_1Z_2 - Z_1T_2)), & \text{if } P_1 \neq P_2 \\ 2X_1^3, & \text{if } P_1 = P_2, \end{cases}$$

$$M_1 = \begin{cases} (Y_1 + Z_1 + aT_1)X_2T_2Z_1 - (Y_2 + Z_2 + aT_2)X_1T_1Z_2, & \text{if } P_1 \neq P_2 \\ -(Y_1 + 2Z_1)X_1^2 & \text{if } P_1 = P_2, \end{cases}$$

$$M_3 = \begin{cases} (Y_1 + Z_1 + aT_1)(Y_2 + Z_2 + aT_2)(X_1Z_2 - Z_1X_2) & \text{if } P_1 \neq P_2 \\ (aX_1^2 + Z_1^2 + Y_1Z_1)Y_1 & \text{if } P_1 = P_2. \end{cases}$$

Here $\eta, \theta \in \mathbb{F}_p$ and can be precomputed, and N, M_1, and M_3 are computed during the point addition/doubling step.

If $P_1 \neq P_2$, then rewriting the formulas for N, M_1, and M_3 in terms of ω and the $\mathbb{F}_{p^{k/2}}$-coordinates of P_1' and P_2', we see that $M_1 =: \omega m_1 \in \omega\mathbb{F}_{p^{k/2}}, N \in \mathbb{F}_{p^{k/2}}$, and $M_3 =: \omega m_3 \in \omega\mathbb{F}_{p^{k/2}}$. As $N \in \mathbb{F}_{p^{k/2}}$, so is the coefficient $\frac{x_Q^2}{N(y_Q+1-cx_Q^2)}$, hence maps to 1 in the final exponentiation of Miller's algorithm (so can be ignored). Therefore, the computation of the Miller function $h_{P_1,P_2}(Q)$ when $P_1 \neq P_2$ amounts to the computation of $\eta N+((\theta+a)m_1+m_3)\omega$, where all the variables are known, the time-consuming part of which is 2 multiplications of an element in $\mathbb{F}_p$ by an element in $\mathbb{F}_{p^{k/2}}$, which using the schoolbook method takes $2 \cdot \frac{k}{2}\mathbf{m}_1 = k\mathbf{m}_1$.

If $P_1 = P_2$, then in the same way we see that $M_1 \in \mathbb{F}_{p^{k/2}}, N =: \omega n \in \omega\mathbb{F}_{p^{k/2}}$, and $M_3 \in \mathbb{F}_{p^{k/2}}$. As $N \in \omega\mathbb{F}_{p^{k/2}}$, so is N^{-1} and hence also $\frac{x_Q^2}{N(y_Q+1-cx_Q^2)}$, thus goes to ω in the final exponentiation of Miller's algorithm. Therefore, the computation of the Miller function $h_{P_1,P_1}(Q)$ amounts to the computation of $\eta\omega^2 n + ((\theta + a)M_1 + M_3)\omega$. As $\eta\omega^2$ can be precomputed, again all the variables are known, so the computation takes $k\mathbf{m}_1$ in exactly the same way as above.

Speed-up (3) of Section 3.1 does not apply as the Miller function h has no zero coordinates as a vector with coefficients in $\mathbb{F}_{p^{k/2}}$.

Combining the above, we see that the whole Miller doubling step (i.e. steps 4 and 5 of Algorithm 3) takes

$$\mathbf{m}_k + \mathbf{s}_k + \left(k^2 + k\right)\mathbf{m}_1 + 2k^2\mathbf{s}_1 + \frac{k^2}{4}\mathbf{mc}_1,$$

and the whole Miller addition step (i.e. steps 7 and 8 of Algorithm 3) takes

$$\mathbf{m}_k + \left(4k^2 + k\right)\mathbf{m}_1 + \frac{k^2}{4}\mathbf{s}_1 + k^2\mathbf{mc}_1.$$

Quartic twists of Jacobi Quartic curves. In the following section we summarize results of Duquesne, El Mrabet, and Fouotsa [9]. The only elliptic curves that admit quartic twists are those of j-invariant 1728, and with embedding

degree divisible by 4. For Jacobi Quartic curves this is equivalent to the coefficient μ in Equation (3) being zero, that is $E_{1728} : Y^2Z^2 = dX^4 + Z^4$. For E_{1728}, the formulas to add and double points are of course simpler than general curves in Jacobi Quartic form. Also, rather than the extended projective coordinates that we use for general Jacobi Quartic curves, we use the extended projective point representation proposed by Hisil, Koon-Ho Wong, Carter, and Dawson [16], namely $[X : Y : Z : U : V]$, where $U = X^2$ and $V = Z^2$.

Let $\omega \in \mathbb{F}_{p^k} \setminus \mathbb{F}_{p^{k/4}}$, and define $E'_{1728}/\mathbb{F}_{p^{k/4}} : Y^2Z^2 = d\omega^4X^4 + Z^4$. The curve E'_0 is a quartic twist of E_0 via the isomorphism $\phi : [X : Y : Z] \to [\omega X : Y : Z]$. We can use this isomorphism in the ways described in the Section 3.1 as is described in [9].

Pairings on $\mathbb{G}_1 \times \mathbb{G}_2$. As the point arithmetic in this case occurs in $\mathbb{G}_1$, clearly speed-up (1) of Section 3.1 does not apply.

In [9], the authors show that speed-ups (2) and (3) of Section 3.1 can be applied via the above isomorphism ϕ, together with the simpler arithmetic on this specific curve, to get an operation count of

$$\left(\frac{1}{k} + \frac{1}{2}\right)\mathbf{m}_k + 1\mathbf{s}_k + \left(\frac{k}{2} + 3\right)\mathbf{m}_1 + 7\mathbf{s}_1 + 1\mathbf{mc}_1$$

for the doubling steps of Miller (i.e. steps 3 and 4 of Algorithm 1), and of

$$\left(\frac{1}{k} + \frac{1}{2}\right)\mathbf{m}_k + \left(\frac{k}{2} + 12\right)\mathbf{m}_1 + 7\mathbf{s}_1 + 1\mathbf{mc}_1$$

for the addition steps of Miller (i.e. steps 6 and 7 of Algorithm 1). Note that mixed addition is always possible in this case ([9, Section 3.3]).

Pairings on $\mathbb{G}_2 \times \mathbb{G}_1$. Clearly speed-up (1) of Section 3.1 applies, so the point arithmetic can all be performed in $\mathbb{F}_{p^{k/4}}$. Following the recommendations of [9] this gives an operation count of

$$3\mathbf{m}_{k/4} + 7\mathbf{s}_{k/4} + 1\mathbf{mc}_{k/4} = \frac{3k^2}{16}\mathbf{m}_1 + \frac{7k^2}{16}\mathbf{s}_1 + \frac{k^2}{16}\mathbf{mc}_1$$

for point doubling, and

$$12\mathbf{m}_{k/4} + 7\mathbf{s}_{k/4} + 1\mathbf{mc}_{k/4} = \frac{3k^2}{4}\mathbf{m}_1 + \frac{7k^2}{16}\mathbf{s}_1 + \frac{k^2}{16}\mathbf{mc}_1$$

for point addition.

For speed-up (2), we refer to [9, p 16] in which it is shown that the Miller function can be computed in time $\frac{k}{2}\mathbf{m}_1$.

For speed-up (3), we refer to [9, Remark 8; Appendix B] in which it is shown that multiplication by the Miller function costs $\frac{3}{4}\mathbf{m}_k$.

Combining the results of [9] stated above, this gives an operation count of

$$\frac{3}{4}\mathbf{m}_k + \mathbf{s}_k + \left(\frac{k}{2} + \frac{3k^2}{16}\right)\mathbf{m}_1 + \frac{7k^2}{16}\mathbf{s}_1 + \frac{k^2}{16}\mathbf{mc}_1$$

for the doubling steps of Miller (i.e. steps 4 and 5 of Algorithm 3) and of

$$\frac{3}{4}\mathbf{m}_k + \left(\frac{k}{2} + \frac{3k^2}{4}\right)\mathbf{m}_1 + \frac{7k^2}{16}\mathbf{s}_1 + \frac{k^2}{16}\mathbf{mc}_1$$

for the addition steps of Miller (i.e. steps 7 and 8 of Algorithm 3).

Sextic twists of Jacobi quartic curves We include here the necessary formulas for sextic twists of Jacobi Quartic curves. These do not, to our knowledge, appear in the literature. As suggested in [20] for Edwards curves, one can use the Weierstrass sextic twist of a curve to save on arithmetic in extension fields of degree divisible by 6.

Define $E_{J,a}/\mathbb{F}_p : Y^2Z^2 = -\frac{3}{16}a^2X^4 - 3aX^2Z^2 + Z^4$. This curve has j-invariant 0, hence admits sextic twists; its embedding degree k is divisible by 6. Let ω be a generator of $\mathbb{F}_{p^k}$ as a $\mathbb{F}_{p^{k/6}}$-vector space, define $E_{W,\omega,a}/\mathbb{F}_{p^{k/6}} : \omega^6 y^2 = \omega^6 x^3 - a^3$, and define $E_{W,a}/\mathbb{F}_p : y^2 = x^3 - a^3$. As shown in [27], there is an isomorphism

$$\begin{aligned} E_{W,a} &\to E_{J,a} \\ (x,y) &\mapsto \left[2(x-a) : \frac{(2x+a)(x-a)^2 - y^2}{y} : y\right]. \end{aligned} \tag{6}$$

Clearly, there is a $\mathbb{F}_{p^k}$-isomorphism defined by $E_{W,\omega,a} \to E_{W,a}$ given by $(x,y) \mapsto (\omega^2 x, \omega^3 y)$, so by composition with (7) we get an $\mathbb{F}_{p^k}$-isomorphism

$$\begin{aligned} \varphi : E_{W,\omega,a} &\to E_{J,a} \\ (x,y) &\mapsto \left[2(\omega^2 x - a) : \frac{(2\omega^2 x + a)(\omega^2 x - a)^2 - \omega^6 y^2}{\omega^3 y} : \omega^3 y\right]. \end{aligned} \tag{7}$$

Pairings on $\mathbb{G}_1 \times \mathbb{G}_2$. We first give the formulas for pairings that are computed on $\mathbb{G}_1 \times \mathbb{G}_2$ (e.g. twisted Ate pairing). In this case, the point arithmetic is performed in $\mathbb{G}_1 \subseteq E(\mathbb{F}_p)$, so point (1) from Section 3.1 does not apply.

For point (2) of Section 3.1 we proceed by computing the Miller function h. Let $(x_Q, y_Q) = Q \in \mathbb{G}_2 = E_{J,a}(\mathbb{F}_{p^k})$ be the image of $(x'_Q, y'_Q) = Q' \in E_{W,\omega,a}(\mathbb{F}_{p^{k/6}})$ under the twist isomorphism φ of (7). Plugging these values into Equation (5) (i.e. the Miller function as given in [27]) gives

$$h_{P_1,P_2}(x_Q, y_Q) = \frac{4}{N(2\omega^2 x'_Q + a - 4c)}\left(M_3 - \frac{a}{2}M_1 + M_1\frac{x'_Q}{2}\omega^2 + N\frac{y'_Q}{4}\omega^3\right).$$

Observe that $\frac{4}{N(2\omega^2 x'_Q + a - 4c)} \in \mathbb{F}_{p^{k/2}}$, so in particular in the final exponentiation step of Miller's algorithm this becomes 1. Therefore, without loss of generality, in the pairing computation we can replace $h_{P_1,P_2}(x_Q, y_Q)$ by

$$h_{P_1,P_2}(x_Q, y_Q) = M_3 - \frac{a}{2}M_1 + M_1\frac{x'_Q}{2}\omega^2 + N\frac{y'_Q}{4}\omega^3.$$

Here $x'_Q/2$ and $y'_Q/2$ are in $\mathbb{F}_{p^{k/6}}$ and can be precomputed, so $h_{P_1,P_2}(Q)$ can be computed as an element of the $\mathbb{F}_{p^{k/6}}$-vector space $\mathbb{F}_{p^k}$ generated by ω in two

multiplications of an $\mathbb{F}_p$-element with an $\mathbb{F}_{p^{k/6}}$-element, namely $M_1 \cdot \frac{x'_Q}{2}$ and $N \cdot \frac{y'_Q}{4}$, and one multiplication by a constant in $\mathbb{F}_p$, namely $\frac{a}{2} \cdot M_1$. Hence the total cost for computing $h_{P_1,P_2}(Q)$ is given by $2 \cdot \frac{k}{6}\mathbf{m}_1 + \mathbf{mc}_1 = \frac{k}{3}\mathbf{m}_1 + \mathbf{mc}_1$.

Furthermore, this also shows us that speed-up (3) of Section 3.1 applies. The multiplication step in Miller's algorithm of a general element $f \in \mathbb{F}_{p^k}$ with h, is much more efficient than the general $k^2\mathbf{m}_1$ for two general elements

$$f = f_0 + f_1\omega + f_2\omega^2 + f_3\omega^3 + f_4\omega^4 + f_5\omega^5$$

and

$$h = h_0 + h_1\omega + h_2\omega^2 + h_3\omega^3 + h_4\omega^4 + h_5\omega^5,$$

as here $h_1 = h_4 = h_5 = 0$ and $h_0 \in \mathbb{F}_p$. Each of the 6 multiplications $h_0 \cdot f_i$ cost $\frac{k}{6}\mathbf{m}_1$, and each of the 12 multiplications $h_2 \cdot f_i$ and $h_3 \cdot f_i$ cost $\left(\frac{k}{6}\right)^2 \mathbf{m}_1$, giving a total time complexity of

$$6\frac{k}{6}\mathbf{m}_1 + 12\left(\frac{k}{6}\right)^2 \mathbf{m}_1 = \left(k + \frac{k^2}{3}\right)\mathbf{m}_1 = \left(\frac{1}{k} + \frac{1}{3}\right)\mathbf{m}_k$$

for the multiplication $f \cdot h$. This corresponds to the speed-up (3) of Section 3.1.

We refer to the recommendations of [27] for the point arithmetic, giving a total cost of

$$\left(\frac{1}{k} + \frac{1}{3}\right)\mathbf{m}_k + \mathbf{s}_k + \left(4 + \frac{k}{3}\right)\mathbf{m}_1 + 8\mathbf{s}_1 + 2\mathbf{mc}_1$$

for the Miller doubling step (Steps 3 and 4 of Algorithm 1) and

$$\left(\frac{1}{k} + \frac{1}{3}\right)\mathbf{m}_k + \left(16 + \frac{k}{3}\right)\mathbf{m}_1 + 1\mathbf{s}_1 + 5\mathbf{mc}_1$$

for the Miller addition step (Steps 6 and 7 of Algorithm 1).

Pairings on $\mathbb{G}_2 \times \mathbb{G}_1$. In this case, the point arithmetic would naïvely be computed in $\mathbb{F}_{p^k}$, but point (1) of Section 3.1 applies (to some extent). We cannot do arithmetic only in $\mathbb{F}_{p^{k/6}}$ without paying for the conversion between Weierstrass and Jacobi Quartic form; also, doing point doubling and addition on a curve in Weierstrass form is more expensive than on a curve in Jacobi Quartic form. What we can do is perform our point arithmetic on a quadratic twist in Jacobi Quartic form, so in $\mathbb{F}_{p^{k/2}}$, via the isomorphism of Equation (4), as described in Section 3.2.

Both speed-up (2) and speed-up (3) of Section 3.1 use the formula for the Miller function h viewed as an element of a $\mathbb{F}_{p^{k/\delta}}$-vector space. The speed-up comes partly from the fact that the coefficients of h can be written relatively simply in terms of values that were computed during the point addition/doubling just before the computation of h. For this reason, as the point doubling/addition is performed now in $\mathbb{F}_{p^{k/2}}$, the best we can hope for in speed-ups (2) and (3) of Section 3.1 is the speed-up that we get for the quadratic twist.

In particular, for pairings on $\mathbb{G}_2 \times \mathbb{G}_1$, the operation count is the same for sextic twists as it is for quadratic twists.

3.3 Edwards Curves

A twisted Edwards curve over a prime field $\mathbb{F}_p$ is defined by the equation:

$$E_{\mathrm{Ed}}/\mathbb{F}_p : aX^2Z^2 + Y^2Z^2 = Z^4 + dX^2Y^2, \tag{8}$$

where $a, d \in \mathbb{F}_p^*$ and $a \neq d$. The base point (neutral group element) is $[0:1:0]$. The isomorphism to short Weierstrass form can be found in [1].

As stated in [1,20], it is common practice for fast addition and doubling on $E_{\mathrm{Ed}}(\mathbb{F}_p)$ to represent points on twisted Edwards curves in four coordinates $[X:Y:Z:T]$, where $T = XY/Z$. The sum $P_3 = [X_3 : Y_3 : Z_3 : T_3]$ of two points $P_1 = [X_1 : Y_1 : Z_1 : T_1]$ and $P_2 = [X_2 : Y_2 : Z_2 : T_2]$ is given by the formulas:

$$\begin{aligned} X_3 &= (X_1Y_2 - Y_1X_2)(T_1Z_2 + Z_1T_2), \\ Y_3 &= (aX_1X_2 + Y_1Y_2)(T_1Z_2 - Z_1T_2), \\ Z_3 &= (aX_1X_2 + Y_1Y_2)(X_1Y_2 - Y_1X_2), \\ T_3 &= T_1^2Z_2^2 - Z_1^2T_2^2. \end{aligned}$$

Using the recommendations in [1,20], addition can be performed using $14\mathbf{m} + 1\mathbf{mc}$, mixed addition using $12\mathbf{m}_1 + \mathbf{mc}$, and doubling using $4\mathbf{m} + 7\mathbf{s} + 1\mathbf{mc}$.

Quadratic Twists of Edwards Curves As mentioned previously, all the curves that we consider in this paper have even embedding degree and admit quadratic twists. Let $\omega \in \mathbb{F}_{p^k} \setminus \mathbb{F}_{p^{k/2}}$, and define

$$E'^{\omega}_{\mathrm{Ed}}/\mathbb{F}_{p^{k/2}} : a\omega^2X^2Z^2 + Y^2Z^2 = Z^4 + d\omega^2X^2Y^2.$$

The curve $E'^{\omega}_{\mathrm{Ed}}$ is a quadratic twist of E_{Ed} via the isomorphism

$$\phi : [X:Y:Z] \to [\omega X : Y : Z]. \tag{9}$$

We can use this isomorphism in the three ways described in Section 3.1.

Pairings on $\mathbb{G}_1 \times \mathbb{G}_2$. For pairings on $\mathbb{G}_1 \times \mathbb{G}_2$ (such as the twisted Ate pairing), we refer to [1] and the improvement mentioned in [20]. Clearly speed-up (1) of Section 3.1 does not apply as the point arithmetic is performed in $\mathbb{G}_1$. In [1,20] they apply speed-ups (2) and (3) of Section 3.1 to get a total operation count of

$$1\mathbf{m}_k + 1\mathbf{s}_k + (k+4)\mathbf{m}_1 + 7\mathbf{s}_1 + 2\mathbf{mc}_1$$

for the doubling steps of Miller (steps 3 and 4 of Algorithm 1) and a total operation count of

$$1\mathbf{m}_k + (k+14)\mathbf{m}_1 + 1\mathbf{mc}_1 \quad \text{or} \quad 1\mathbf{m}_k + (k+12)\mathbf{m}_1 + 1\mathbf{mc}_1$$

for the addition steps of Miller (steps 6 and 7 of Algorithm 1) using regular or mixed addition respectively.

Pairings on $\mathbb{G}_2 \times \mathbb{G}_1$. For pairings on $\mathbb{G}_2 \times \mathbb{G}_1$ (such as the optimal Ate pairing), the necessary formulas for quadratic twists of Edwards curves do not, to our knowledge, appear in the literature, so for completeness we include them here.

It turns out to be more convenient for this case to use the twist

$$E_{\mathrm{Ed}}^{\omega,Y}/\mathbb{F}_{p^{k/2}} : aX^2Z^2 + \omega^2Y^2Z^2 = Z^4 + d\omega^2X^2Y^2$$

of E_{Ed}. The curve $E_{\mathrm{Ed}}^{\omega,Y}$ is a quadratic twist of E_{Ed} via the isomorphism

$$\phi_Y : [X : Y : Z] \to [X : \omega Y : Z]. \tag{10}$$

Speed-up (1) of Section 3.1 clearly applies; all the point arithmetic can be performed in $\mathbb{F}_{p^{k/2}}$. Using Li, Wu, and Zhang's recommendations [20], point doubling takes

$$4\mathbf{m}_{k/2} + 7\mathbf{s}_{k/2} + 2\mathbf{mc}_{k/2} = k^2\mathbf{m}_1 + \frac{7k^2}{4}\mathbf{s}_1 + \frac{k^2}{2}\mathbf{mc}_1,$$

regular point addition takes

$$14\mathbf{m}_{k/2} + \mathbf{mc}_{k/2} = \frac{7k^2}{2}\mathbf{m}_1 + \frac{k^2}{4}\mathbf{mc}_1,$$

and mixed point addition takes

$$12\mathbf{m}_{k/2} + \mathbf{mc}_{k/2} = 3k^2\mathbf{m}_1 + \frac{k^2}{4}\mathbf{mc}_1.$$

For speed-up (2) of Section 3.1 we start from the formula for the Miller function $h_{P_1,P_2}(Q)$ given in [20] and apply it to the case that $P_1 = \phi_Y(P_1')$ and $P_2 = \phi_Y(P_2')$, where $P_1', P_2' \in E_{E_{\mathrm{Ed}}}^{\omega,Y}(\mathbb{F}_{p^{k/2}})$. To this end, let (x_Q, y_Q) be affine coordinates for Q, $P_1 = [X_1 : Y_1 : Z_1 : T_1]$, $P_2 = [X_2 : Y_2 : Z_2 : T_2]$, and $P_1 + P_2 = [X_3 : Y_3 : Z_3 : T_3]$. By [20], the Miller function is given by

$$h_{P_1,P_2}(Q) = \frac{C_X x_Q + C_Y(y_Q + x_Q y_Q) + C_Z}{Z_3 x_Q - X_3},$$

where

$$C_X = \begin{cases} Z_2(T_1 + Y_1) - Z_1(T_2 + Y_2), & \text{if } P_1 \neq P_2 \\ Y_1T_1 - aX_1^2, & \text{if } P_1 = P_2 \end{cases}$$

$$C_Y = \begin{cases} X_2Z_1 - X_1Z_2 & \text{if } P_1 \neq P_2 \\ X_1T_1 - X_1Y_1 & \text{if } P_1 = P_2 \end{cases}$$

$$C_Z = \begin{cases} X_1(Y_2 + T_2) - X_2(T_1 + Y_1) & \text{if } P_1 \neq P_2 \\ dX_1Z_1 - T_1^2 & \text{if } P_1 = P_2. \end{cases}$$

Here x_Q and $y_Q + x_Q y_Q \in \mathbb{F}_p$ and can be precomputed, and C_X, C_Y, and C_Z are computed during the point addition/doubling step.

Observe that $P_3 = \phi_Y(P_1' + P_2')$ is in the image of ϕ_Y and hence $X_3, Z_3 \in \mathbb{F}_{p^{k/2}}$. In particular, the denominator $Z_3 x_Q - X_3 \in \mathbb{F}_{p^{k/2}}$ and so goes to 1 in the final exponentiation of Miller's algorithm (and so can be ignored).

If $P_1 \neq P_2$, then rewriting the formulas for C_X, C_Y, and C_Z in terms of ω and the $\mathbb{F}_{p^{k/2}}$-coordinates of P_1' and P_2', we see that $C_Y \in \mathbb{F}_{p^{k/2}}$, $C_X = \omega c_X \in \omega\mathbb{F}_{p^{k/2}}$, and $C_Z = \omega c_Z \in \omega\mathbb{F}_{p^{k/2}}$. That is, computing $h_{P_1,P_2}(Q)$ amounts to computing

$$(y_Q + x_Q y_Q)C_Y + (x_Q c_X + c_Z)\omega,$$

in other words, performing 2 multiplications of an $\mathbb{F}_p$-element by an $\mathbb{F}_{p^{k/2}}$-element, which costs $k\mathbf{m}_1$ using the schoolbook method.

If $P_1 = P_2$, then rewriting the formulas for C_X, C_Y, and C_Z in terms of ω and the $\mathbb{F}_{p^{k/2}}$-coordinates of P_1', we see that C_X, $C_Z \in \mathbb{F}_{p^{k/2}}$ and $C_Y =: \omega c_Y \in \omega\mathbb{F}_{p^{k/2}}$. That is, computing $h_{P_1,P_1}(Q)$ amounts to computing

$$x_Q C_X + C_Z + (y_Q + x_Q y_Q)c_Y\omega,$$

in other words, performing 2 multiplications of an $\mathbb{F}_p$-element by an $\mathbb{F}_{p^{k/2}}$-element, which again costs $k\mathbf{m}_1$ using the schoolbook method.

Speed-up (3) of Section 3.1 does not apply as the Miller function h has no zero coordinates as a vector with coefficients in $\mathbb{F}_{p^{k/2}}$.

Combining the above, we see that the whole Miller doubling step (i.e. steps 4 and 5 of Algorithm 3) takes

$$\mathbf{m}_k + \mathbf{s}_k + (k^2 + k)\mathbf{m}_1 + \frac{7k^2}{4}\mathbf{s}_1 + \frac{k^2}{2}\mathbf{mc}_1$$

and the addition step (i.e. steps 7 and 8 of Algorithm 3) takes

$$\mathbf{m}_k + \left(\frac{7k^2}{2} + k\right)\mathbf{m}_1 + \frac{k^2}{4}\mathbf{mc}_1 \quad \text{or} \quad \mathbf{m}_k + (3k^2 + k)\mathbf{m}_1 + \frac{k^2}{4}\mathbf{mc}_1$$

without, or with mixed addition respectively.

Quartic twists of Edwards curves This section summarizes results of [20]. As mentioned previously, the only elliptic curves that admit quartic twists are those of j-invariant 1728, and with embedding degree $k = 2^i$ for some $i \geq 2$. For Edwards curves this is equivalent to setting $d = -a$ in Equation (8), that is

$$E_{\mathrm{Ed}}/\mathbb{F}_p : aX^2Z^2 + Y^2Z^2 = Z^4 - aX^2Y^2. \tag{11}$$

There is no quartic twist of E_{Ed} that can be written in Edwards form, but in [20, Lemma 2] it is shown that for $\omega \in \mathbb{F}_{p^k} \setminus \mathbb{F}_{p^{k/4}}$, the curve

$$W_a/\mathbb{F}_{p^{k/4}} : \frac{2}{a}v^2 = u^3 + \frac{1}{\omega^4}u$$

defines a degree 4 twist of E_{Ed}.

Pairings on $\mathbb{G}_1 \times \mathbb{G}_2$. In this case, the point arithmetic is performed in $\mathbb{G}_1$ so speed-up (1) of Section 3.1 does not apply.

The authors of [20] show that applying speed-ups (2) and (3) via the isomorphism to W_a gives a total operation count of

$$\left(\frac{1}{2}+\frac{1}{k}\right)\mathbf{m}_k + 1\mathbf{s}_k + \left(\frac{k}{2}+4\right)\mathbf{m}_1 + 7\mathbf{s}_1 + 2\mathbf{mc}_1$$

for the doubling steps of Miller's algorithm (i.e. steps 3 and 4 of Algorithm 1 and of

$$\left(\frac{1}{2}+\frac{1}{k}\right)\mathbf{m}_k + \left(\frac{k}{2}+14\right)\mathbf{m}_1 + 1\mathbf{mc}_1 \quad \text{or} \quad \left(\frac{1}{2}+\frac{1}{k}\right)\mathbf{m}_k + \left(\frac{k}{2}+12\right)\mathbf{m}_1 + 1\mathbf{mc}_1$$

for the addition steps of Miller's algorithm (i.e. steps 5 and 6 of Algorithm 1 using regular or mixed addition respectively.

Pairings on $\mathbb{G}_2 \times \mathbb{G}_1$. As there is no quartic twist of E_{Ed} that can be written in Edwards form, the best we can do (as explained in Section 3.2) is use the methods for quadratic twists described in 3.3.

Sextic twists of Edwards curves The following section summarizes results of [20]. As mentioned previously, the only elliptic curves that admit sextic twists are those of j-invariant 0 and with embedding degree divisible by 6. For Edwards curves this is equivalent to setting $a = (-7 \pm 4\sqrt{3})d$ in Equation (8); this gives rational a when $p \equiv 1 \pmod{12}$. In this case the curve is given by

$$E_{\mathrm{Ed}}/\mathbb{F}_p : (-7 \pm 4\sqrt{3})dX^2Z^2 + Y^2Z^2 = Z^4 + dX^2Y^2.$$

There is no sextic twist of $E_{E_{\mathrm{Ed}}}$ that can be written Edwards form, but in [20] (Lemma 3), it is shown that for $\omega \in \mathbb{F}_{p^k} \setminus \mathbb{F}_{p^{k/6}}$, the curve

$$W_{M,N}/\mathbb{F}_{p^{k/6}} : v^2 = u^3 - \frac{M^3N^3\omega^6}{27},$$

with $M =$ and $N =$ is a degree 6 twist of E_{Ed}.

Pairings on $\mathbb{G}_1 \times \mathbb{G}_2$. In this case, the point arithmetic is perform in $\mathbb{G}_1$ so speed-up (1) of Section 3.1 does not apply.

The authors of [20] show that applying speed-ups (2) and (3) via the isomorphism to $W_{M,N}$ gives a total operation count of

$$\left(\frac{1}{3}+\frac{1}{k}\right)\mathbf{m}_k + 1\mathbf{s}_k + \left(\frac{k}{3}+4\right)\mathbf{m}_1 + 7\mathbf{s}_1 + 3\mathbf{mc}_1$$

for the doubling steps of Miller's algorithm (i.e. steps 3 and 4 of Algorithm 1 and of

$$\left(\frac{1}{3}+\frac{1}{k}\right)\mathbf{m}_k + \left(\frac{k}{3}+14\right)\mathbf{m}_1 + 2\mathbf{mc}_1 \quad \text{or} \quad \left(\frac{1}{3}+\frac{1}{k}\right)\mathbf{m}_k + \left(\frac{k}{3}+12\right)\mathbf{m}_1 + 2\mathbf{mc}_1$$

for the addition steps of Miller's algorithm (i.e. steps 5 and 6 of Algorithm 1 using regular or mixed addition respectively.

Pairings on $\mathbb{G}_2 \times \mathbb{G}_1$. As there is no sextic twist of E_{Ed} that can be written in Edwards form, the best we can do (as explained in Section 3.2) is use the quadratic twist methods described in Section 3.3.

4 Computational results

In this section we first summarize the operation counts for each pairing and curve type addressed in this survey, and then use this review to choose the optimal curve in each known TNFS-secure compact family for 128-bit security level (of which there are 9 to which our methods may be applied), and give the best pairing and curve shape for this curve. We then present the best choice(s) of curve, pairing, and curve shape from these 9 choices, giving the optimal known TNFS-secure pairing-friendly elliptic curve for 128-bit security level. This method can easily be applied also to 192- and 256-bit security level.

Notation

- **s**: time required to square an $\mathbb{F}_p$-element.
- **m**: time required to multiply an $\mathbb{F}_p$-element.
- **mc**: time required to multiply by a (small) constant in $\mathbb{F}_p$.
- hDBL: steps 3 and 4 of Algorithm 1 or steps 4 and 5 of Algorithm 3 (computation of and multiplication by the line function $h_{R,R}(Q)$).
- hADD: steps 6 and 7 in Miller's algorithm of Algorithm 1 or steps 7 and 8 of Algorithm 3 (computation of and multiplication by the line function $h_{R,P}(Q)$).
- **e**: final exponentiation in Miller's algorithm (c.f. Algorithms 1, 3).
- b_x: the bit length of x.
- w_x: the Hamming weight of x.

In Tables 1 and 2, we compare operation counts for hDBL and hADD in each of the cases studied in Section 3. For simplicity, where relevant the operation counts are for mixed addition (not general addition). Observe that the total cost of the twisted Ate pairing $\widehat{a}_e$ is

$$(b_{T_e} - 1)\mathrm{hDBL} + (w_{T_e} - 1)\mathrm{hADD} + e$$

and the total cost of the optimal Ate pairing $\widehat{a}_0$ with parameter s is

$$(b_s - 1)\mathrm{hDBL} + (w_s - 1)\mathrm{hADD} + e.$$

For easier comparison, we now replace each instance of $\mathbf{m}_k$, $\mathbf{s}_k$, and $\mathbf{mc}_k$ with $k^2\mathbf{m}_1 = k^2\mathbf{m}$, $k^2\mathbf{s}_1 = k^2\mathbf{s}$, and $k^2\mathbf{mc}_1 = k^2\mathbf{mc}$. (Consider $\mathbb{F}_{p^k}$ as a k-dimensional $\mathbb{F}_p$-vector space, then this is clearly true in general). Besides the comparison in terms of operation count, we also give the timing of our MAGMA implementation for each of the examples that follow. These timings are definitely not optimal (but serve as a basic comparison between families) as we have not yet considered optimising finite field arithmetic and the implementation is not yet in C. We leave this for future work.

Table 1. Operation counts for hDBL

hDBL	JQ on $\mathbb{G}_1 \times \mathbb{G}_2$	JQ on $\mathbb{G}_2 \times \mathbb{G}_1$	Ed on $\mathbb{G}_1 \times \mathbb{G}_2$	Ed on $\mathbb{G}_2 \times \mathbb{G}_1$
$2\|k$	$(k^2+k+4)\mathbf{m}$	$\left(2k^2+k\right)\mathbf{m}$	$(k^2+k+4)\mathbf{m}$	$(2k^2+k)\mathbf{m}$
$j \neq 0, 1728$	$+(k^2+8)\mathbf{s}+1\mathbf{mc}$	$+3k^2\mathbf{s}+\frac{k^2}{4}\mathbf{mc}$	$+(k^2+7)\mathbf{s}+2\mathbf{mc}$	$+\frac{11k^2}{4}\mathbf{s}+\frac{k^2}{2}\mathbf{mc}$
$4\|k$	$(\frac{k^2}{2}+\frac{3k}{2}+3)\mathbf{m}$	$(\frac{15k^2}{16}+\frac{k}{2})\mathbf{m}$	$(\frac{k^2}{2}+\frac{3k}{2}+4)\mathbf{m}$	$(2k^2+k)\mathbf{m}$
$j = 1728$	$+(k^2+7)\mathbf{s}+1\mathbf{mc}$	$+\frac{23k^2}{16}\mathbf{s}+\frac{k^2}{16}\mathbf{mc}$	$+(k^2+7)\mathbf{s}+2\mathbf{mc}$	$+\frac{11k^2}{4}\mathbf{s}+\frac{k^2}{2}\mathbf{mc}$
$6\|k$	$(\frac{k^2}{3}+\frac{4k}{3}+4)\mathbf{m}$	$\left(2k^2+k\right)\mathbf{m}$	$(\frac{k^2}{3}+\frac{4k}{3}+4)\mathbf{m}$	$(2k^2+k)\mathbf{m}$
$j = 0$	$+(k^2+8)\mathbf{s}+2\mathbf{mc}$	$+3k^2\mathbf{s}+\frac{k^2}{4}\mathbf{mc}$	$+(k^2+7)\mathbf{s}+3\mathbf{mc}$	$+\frac{11k^2}{4}\mathbf{s}+\frac{k^2}{2}\mathbf{mc}$

Table 2. Operation counts for hADD

hADD	JQ on $\mathbb{G}_1 \times \mathbb{G}_2$	JQ on $\mathbb{G}_2 \times \mathbb{G}_1$	Ed on $\mathbb{G}_1 \times \mathbb{G}_2$	Ed on $\mathbb{G}_2 \times \mathbb{G}_1$
$2\|k$	$(k^2+k+16)\mathbf{m}$	$\left(5k^2+k\right)\mathbf{m}$	$(k^2+k+12)\mathbf{m}$	$(4k^2+k)\mathbf{m}$
$j \neq 0, 1728$	$+1\mathbf{s}+4\mathbf{mc}$	$+\frac{k^2}{4}\mathbf{s}+k^2\mathbf{mc}$	$+1\mathbf{mc}$	$+\frac{k^2}{4}\mathbf{mc}$
$4\|k$	$(\frac{k^2}{2}+\frac{3k}{2}+12)\mathbf{m}$	$(\frac{3k^2}{2}+\frac{k}{2})\mathbf{m}$	$(\frac{k^2}{2}+\frac{3k}{2}+12)\mathbf{m}$	$(4k^2+k)\mathbf{m}$
$j = 1728$	$+7\mathbf{s}+1\mathbf{mc}$	$+\frac{7k^2}{16}\mathbf{s}+\frac{k^2}{16}\mathbf{mc}$	$+1\mathbf{mc}$	$+\frac{k^2}{4}\mathbf{mc}$
$6\|k$	$(\frac{k^2}{3}+\frac{4k}{3}+16)\mathbf{m}$	$\left(5k^2+k\right)\mathbf{m}$	$(\frac{k^2}{3}+\frac{4k}{3}+12)\mathbf{m}$	$(4k^2+k)\mathbf{m}$
$j = 0$	$+1\mathbf{s}+5\mathbf{mc}$	$+\frac{k^2}{4}\mathbf{s}+k^2\mathbf{mc}$	$+2\mathbf{mc}$	$+\frac{k^2}{4}\mathbf{mc}$

Family 1 (embedding degree 8, CM discriminant D = 1). By [14] there is a complete polynomial family $[p(x), t(x), r(x)]$ of pairing-friendly elliptic curves $E/\mathbb{F}_p : y^2 = x^3 + x$ with embedding degree $k = 8$, CM-discriminant $D = 1$ and $\rho = 2$ given by the polynomials:

$$p(x) = \frac{1}{4}\left(x^8 + x^6 + 5x^4 + x^2 + 4x + 4\right), \quad t(x) = x^4 + x + 2, \quad r(x) = \Phi_8(x) = x^4 + 1.$$

All three polynomials produce integer values for every $x \equiv 0 \pmod 2$ and give 128-bit level security when x is in the range $[2^{64}, 2^{64.25})$. That is, for this range, the extension field $\mathbb{F}_{p^8}$ has a size of 4096-bits, which by Equation (1) offers a security level of approximately 124-bits.

Twisted Ate pairing: As $T = t - 1$ is a primitive 8th root of unity for this family, we get that $T_4 = r - 1$, so we set $e = 2$ in order to minimize b_{T_e}. We then apply Algorithm 2 (with T_2 in place of r) to minimize w_{T_2} for our range of choices for x_0, giving $x_0 = 18446744073710252032$ and

$$T_2 = 340282366920964304252768870620960129024,$$

with $w_{T_2} = 14$ and $b_{T_2} = 129$. The elliptic curve in Jacobi Quartic form is $E_J/\mathbb{F}_p : y^2 = dx^4 + 1$, where

$$d = 837987995621666863459334712113181651507008401733766474117388087210727033399020037991767863321863117443242734491985609643880597836109560681909227642596352$$

and in Edwards form is $E_{\mathrm{Ed}}/\mathbb{F}_p : 2x^2 + y^2 = 1 - 2x^2y^2$.

Optimal Ate pairing: The shortest vector of the corresponding lattice L is $V = [x, -1, 0, 0]$. Thus, the optimal Ate pairing is calculated by the formula:

$$\widehat{a}_o(Q, P) = f_{x,Q}(P)^{\frac{p^k-1}{r}}.$$

We apply Algorithm 4 to choose the best value of x_0, which in this case is (also) $x_0 = 18446744073710252032$ with $s = x_0$, giving $w_s = 6$ and $b_s = 65$.

Comparison: Following e.g. [10], we give two comparisons by plugging in these values to Tables 1 and 2 corresponding to the two options $\mathbf{s} = 0.8\mathbf{m}$ and $\mathbf{s} = \mathbf{m}$. The table below indicates time required to compute the pairing in each case as a multiple of $\mathbf{m}$.

	JQ, $\widehat{a}_2$	JQ, $\widehat{a}_o$	Ed, $\widehat{a}_2$	Ed, $\widehat{ao}$
$\mathbf{s} = \mathbf{m}$	16064	10900	15960	21288
$\mathbf{s} = 0.8\mathbf{m}$	14228.2	9694.4	14142.4	19035.2

As each pairing includes exactly $1\mathbf{e}$, we do not include this in the count, as it does not change the comparison. As the curve constants for E_J are in the order of p, we set $\mathbf{mc} = \mathbf{m}$ for the Jacobi Quartic case. As the curve constants for E_{Ed} are very small, we set $\mathbf{mc} = 0$ for the Edwards case. The above table shows clearly that the best choice for this family is the optimal Ate pairing applied to the above curve in Jacobi Quartic form. Our MAGMA implementation runs this example in 16ms.

Family 2 (embedding degree 8, CM discriminant D = 2). The second family of TNFS-secure pairing-friendly elliptic curves in [14] is also of embedding degree 8, it has CM discriminant $D = 2$ and it is parametrized by:

$$p(x) = \frac{1}{8}\left(2x^8 + 4x^7 + 3x^6 + 2x^5 + 11x^4 + 12x^3 + 3x^2 + 2x + 9\right),$$
$$t(x) = x^4 + x^3 + 2, \quad r(x) = x^4 + 1,$$

where $x \equiv 1 \pmod 2$ in the range $[2^{64}, 2^{64.25})$.

Twisted Ate pairing: The curves generated by this family admit only quadratic twists, thus the only possibility for the twisted Ate pairing is T_4. The value $x = 18446744073709584935$ gives the lowest Hamming weight for T_4, particularly $b_{T_4} = 256$ and $w_{T_4} = 68$. The coefficients of the elliptic curve in Weierstrass form are:

$A = 19339957016022543733528152272955779880013915750752149052244325776723696340238896700311167681216806999831537071596907112291500601551238672847753729638793 18$

$B = 32499257242445616807374079943906148935513841319714498271680824047654448427002208648788265190103673429884742123474891938446749436326415737101406066666223982$

Optimal Ate pairing: For this family, we take the shortest vector of the lattice L as $V = [x, 0, 0, -1]$, so that the optimal Ate pairing is:

$$\widehat{a}_o(Q, P) = f_{x,Q}(P)^{\frac{p^k-1}{r}}.$$

The best value of x is (also) $x = 18446744073709584935$ with $s = x$, giving $w_s = 6$ and $b_s = 65$.

	JQ, $\widehat{a}_4$	JQ, $\widehat{a}_o$	Ed, $\widehat{a}_4$	Ed, $\widehat{a}o$
$\mathbf{s = m}$	44226	24464	43690	23696
$\mathbf{s} = 0.8\mathbf{m}$	40540.6	21987.2	40069	21443.2

For this family the best choice is the optimal Ate pairing with Edwards curve. Our MAGMA implementation runs this example in 21ms.

Family 3 (embedding degree 8, CM discriminant D = 3). The third family with embedding degree 8 has $D = 3$ and it is parametrized by:

$$p(x) = \frac{1}{3}\left(3x^{16} - 9x^{12} + x^{10} - 2x^9 + 16x^8 - x^6 + 5x^5 - 13x^4 + x^2 - 5x + 7\right)$$
$$t(x) = (x^8 + x^5 - x^4 - x + 2), \quad r(x) = x^8 - x^4 + 1,$$

for every $x \equiv 1 \pmod 3$ and $x \in [2^{32}, 2^{32.125})$.

Twisted Ate pairing: We only have only choice of T_e for this family, that is $T_4 = r - 1$. Applying Algorithm 2, we see that the best choice in this family for the twisted Ate pairing is $x = 4295331013$, giving $b_{T_4} = 256$ and $w_{T_4} = 107$.

Optimal Ate pairing: The shortest vector of the corresponding lattice that we choose is $V = [x^2, x, 1, 0]$, so that in this case the optimal pairing is

$$\widehat{a}_o(Q, P) = \left(f_{x^2,Q}(P) f_{x,Q}^p(P) h_{s_2Q,xpQ}(P) h_{s_3Q,p^2Q}(P)\right)^{\frac{p^k-1}{r}}.$$

Observe that $s_3Q = \infty$, so h_{s_3Q,p^2Q} is the vertical line passing through $p^2Q = (a, b)$ with equation $X - a$. Recall that in every case outlined above $Q \in E(\mathbb{F}_{p^k})$ is chosen to be in the image of the quadratic twist isomorphism $(X, Y) \mapsto (X, \omega Y)$, where $X, Y \in \mathbb{F}_{p^{k/2}}$ and ω generates $\mathbb{F}_{p^k}$ as a $\mathbb{F}_{p^{k/2}}$-vector space. In particular, the multiple p^2Q of Q is also in the image of this isomorphism, hence $a \in \mathbb{F}_{p^{k/2}}$ and $h_{s_3Q,p^2Q}(P) = x_P - a \in \mathbb{F}_{p^{k/2}}$, so maps to 1 under the final exponentiation. This leaves $h_{s_2Q,xpQ}(P)$, which does give a nontrivial contribution, but can be precomputed so adds negligable computation to the pairing.

We also recall from [26] that $f_{x^2,Q} = f_{x,Q}^x f_{x,[x]Q}$ and thus the formula for the optimal Ate pairing simplifies to:

$$\widehat{a}_o(Q, P) = \left(f_{x,Q}^{p+x}(P) f_{x,[x]Q}(P) h_{s_2Q,xpQ}(P)\right)^{\frac{p^k-1}{r}}.$$

Then total time to compute the optimal Ate pairing $\widehat{a}_o$ is

$$2(b_s - 1)\text{hDBL} + 2(w_s - 1)\text{hADD} + \mathbf{e} + \mathbf{E},$$

where $\mathbf{E}$ denotes exponentiaion by $p + x$. We apply Algorithm 4 to choose the best value of x, which in this case is (also) $x = 4295331013$ with $s = x$, giving $w_s = 10$ and $b_s = 32$. We exclude the cost of $\mathbf{E}$ from the following table as our implementation uses a combination of MAGMA's Frobenius function and a basic exponentiation algorithm, meaning that the cost is hard to give precisely in terms of $\mathbf{m}$. This means that the operation count for $\widehat{a}_o$ is an underestimation.

	JQ, $\widehat{a}_4$	JQ, $\widehat{a}_o$	Ed, $\widehat{a}_4$	Ed, $\widehat{a}o$
$\mathbf{s} = \mathbf{m}$	48002	28672	47154	26368
$\mathbf{s} = 0.8\mathbf{m}$	44294.4	26233.6	43518.8	24185.6

Our MAGMA implementation runs this example in 31ms for the optimal Ate pairing on twisted Edwards curves (this is also the fastest implementation).

Family 4 (embedding degree 10, CM discriminant D = 1). This family is parametrized by:

$$p(x) = \frac{1}{4}\left(x^{14} - 2x^{12} + x^{10} + x^4 + 2x^2 + 1\right), \quad t(x) = x^2 + 1, \quad r(x) = \Phi_{20}(x)$$

It produces pairing-friendly elliptic curves $E/\mathbb{F}_p : y^2 = x^3 + x$, with $\rho = 1.7422$ whenever $x \equiv 1 \pmod 2$. For a 128-bit security level, the input x must be in the range $[2^{32}, 2^{32.125})$.

Twisted Ate pairing: Elliptic curves derived by this family admit quadratic twists. For $x = 4295075489$ we get $T_5 = (t(x)-1)^5 \pmod{r(x)}$, with $\log T_5 = 256$ and $\text{wt}(T_5) = 98$.

Optimal Ate pairing: We choose the shortest vector of the lattice L to be $V = [x^2, -1, 0, 0]$ and the formula for the optimal Ate pairing becomes:

$$\widehat{a}_o(Q, P) = f_{x^2,Q}(P)^{\frac{p^k-1}{r}}.$$

For $x = 3963617801$ we set $s = x^2$, where we have $b_s = 64$ and $w_s = 24$.

Comparison: Since $\mu = 0$ in the Jacobi quartic form, there is only one multiplication with the constant d in the doubling step, for which we set $\mathbf{mc} = \mathbf{m}$. In addition for the twisted Edwards curve we set $\mathbf{mc} = 0$.

	JQ, $\widehat{a}_5$	JQ, $\widehat{a}_o$	Ed, $\widehat{a}_5$	Ed, $\widehat{a}o$
$\mathbf{s} = \mathbf{m}$	69026	45010	68189	39985
$\mathbf{s} = 0.8\mathbf{m}$	63498.6	41115	62732	36520

Our MAGMA implementation runs this example in 31ms, using the optimal Ate pairing for the elliptic curve in twisted Edwards form.

Family 5 (embedding degree 10, CM discriminant D = 5). The next polynomial family is parametrized by

$$p(x) = \frac{1}{20}\left(4x^{14} - 7x^{12} + 11x^{10} - 11x^8 - 9x^6 + 13x^4 - 16x^2 + 20\right),$$
$$t(x) = -x^6 + x^4 - x^2 + 2, \quad r(x) = x^8 - x^6 + x^4 - x^2 + 1,$$

and produces pairing-friendly curves when $x \equiv 0, 4$ or $6 \pmod{10}$.

Twisted Ate pairing: The elliptic curves produced by this family have quadratic twists; for $x = 4299680754$ we get $\log T_5 = 256$ and $\mathrm{wt}(T_5) = 94$. The coefficients of the Weierstrass curve are:

$A = 14341425580724829927177679876052249268220716545811818876608068839790853315749422796657197785750997862118879094705015729540019504334875 4$
$B = 14481523556552710411671660006638957469451766030785792115734473329284654238896443045732697805525577471996853952070706350931660209629801 9$

Optimal Ate pairing: We take the shortest vector $V = [x^2 - 1, 1, -1, 1]$ and the formula for the optimal Ate pairing becomes:

$$\widehat{a}_o(Q, P) = \left[f_{x^2-1,Q}(P) h_{[s_2]Q,[p]Q}(P) h_{[s_3]Q,[-p^2]Q}(P)\right]^{\frac{p^k-1}{r}},$$

where both values $h_{[s_2]Q,[p]Q}(P)$ and $h_{[s_3]Q,[-p^2]Q}(P)$ contribute in the pairing computation but they can be precomputed. We take $x = 4295426686$; the coefficients of the Weierstrass curve are:

$A = 4983541974485518942640702841041487964808494612567841938054317835752967 99138988465904070638152576106659286268257938382130349688955316 05$
$B = 8175213845678406654230954570916316289077807358823318928079745531213973 2354874453453192312647819009597646943271112111073990129871455013$

	JQ, $\widehat{a}_5$	JQ, $\widehat{a}_o$	Ed, $\widehat{a}_5$	Ed, $\widehat{a}o$
$\mathbf{s = m}$	69048	47675	68304	43275
$\mathbf{s = 0.8m}$	63521.4	43785	62847	39810

In the above table we can see that the best choice is to use the optimal Ate pairing for twisted Edwards curves. The running time using our implementation is 40ms. Note that there is also an additional multiplication of f and H, before the final exponentiation.

Family 6 (embedding degree 10, CM discriminant D = 15). The final complete family with embedding degree 10 is parametrized by:

$$p(x) = \frac{1}{15}(4x^{14} + 4x^{13} + x^{12} - 12x^{11} - 12x^{10} - 7x^9 + 11x^8 + 17x^7 + 15x^6 - 3x^5 - 11x^4 + x^3 - 2x^2 + 3x + 6), \quad (x) = x^3 + 1, \quad r(x) = \Phi_{30}(x),$$

with $x \equiv \{1, 3, 6, 13\} \pmod{15}$.

Twisted Ate pairing: These curves have quadratic twists and hence the twisted Ate pairing requires $\log T_5$ iterations in Miller's loop. We take $x = 4295609701$; then $b_{T_5} = 256$ and $w_{T_5} = 109$. The coefficients of the Weierstrass curve are:

$$A = 753054987561999597390417147687415963459724732555441318909950220887612731195958560020932597122322062072527883129330204233724407099964453$$
$$B = 189107879463636387914108825446373730621474814382984597653988441512830268496147067961907300153652307761692277431976821323488452819833486$$

Optimal Ate pairing: We choose the shortest vector $V = [x, 0, -1, x^2]$ for the corresponding lattice L and the formula for the optimal Ate pairing is:

$$\widehat{a}_o(Q,P) = \left(f_{x,Q}(P) f_{x^2,Q}^{p^3}(P) h_{[s_2]Q,\infty}(P) h_{[s_3]Q,[-p^2]Q}(P)\right)^{\frac{p^k-1}{r}}.$$

Note that for $h_{[s_2]Q,\infty}(P)$, we have $h_{[s_2]Q,\infty}(P) = x_P - x_{[s_2]Q} \in \mathbb{F}_{p^5}$ and hence maps to 1 in the final exponentiation. Therefore, only the value $h_{[s_3]Q,[-p^2]Q}(P)$ contributes in the pairing computation. Furthermore, we can use the relation $f_{x^2,Q}(P) = f_{x,Q}^x(P) f_{x,[x]Q}(P)$, and thus the formula for the optimal Ate pairing transforms to:

$$\widehat{a}_o(Q,P) = \left(f_{x,Q}^{1+xp^3}(P) f_{x,[x]Q}^{p^3}(P) h_{[s_3]Q,[-p^2]Q}(P)\right)^{\frac{p^k-1}{r}}.$$

We take $s = x = 4295609701$ as in the case of the twisted Ate pairing, for which $b_s = 32$ and $w_s = 12$.

	JQ, $\widehat{a}_5$	JQ, $\widehat{a}_o$	Ed, $\widehat{a}_5$	Ed, $\widehat{ao}$
$\mathbf{s} = \mathbf{m}$	71013	47140	70149	42740
$\mathbf{s} = 0.8\mathbf{m}$	65483.4	43310	64692	39330

The best choice is to use the optimal Ate pairing for twisted Edwards curves. Our MAGMA implementation runs this example in 37ms.

Family 7 (embedding degree 12, CM discriminant D = 3). We study the following complete polynomial family with $k = 12$:

$$p(x) = \frac{1}{3}\left(x^6 - 2x^5 + 2x^3 + x + 1\right), \quad t(x) = x + 1, \quad r(x) = x^4 - x^2 + 1,$$

where $x \equiv 1 \pmod 3$. This is a family of elliptic curves in Weierstrass form given by $E/\mathbb{F}_p : y^2 = x^3 + 1$.

Twisted Ate pairing: Elliptic curves produced by this family admit sextic twists, giving a choice of $\widehat{a}_2$, $\widehat{a}_3$, and $\widehat{a}_6$ for the twisted Ate pairing. The optimal choice for the twisted Ate pairing is $\widehat{a}_2$ with $x = 18446744073709611553$, for which $b_{T_2}(r) = 129$ and $\mathrm{wt}(T_2) = 19$.

Optimal Ate pairing: The shortest vector of the corresponding lattice for this family is $V = [x, -1, 0, 0]$ and so the formula for the optimal Ate pairing is:

$$\widehat{a}_o(Q, P) = f_{x,Q}(P)^{\frac{p^k-1}{r}}.$$

We choose $s = x = 18446744073709818889$, with $b_s = 64$ and $w_s = 6$.

	JQ, $\widehat{a}_2$	JQ, $\widehat{a}_o$	Ed, $\widehat{a}_2$	Ed, $\widehat{a}o$
$\mathbf{s = m}$	29742	52944	29598	51504
$\mathbf{s = 0.8m}$	25877.6	47464.8	25762.6	46514.4

The best choice for this family is the twisted Ate pairing with twisted Edwards curves. Our MAGMA implementation runs this example in 23ms.

Family 8 (embedding degree 12, CM discriminant D = 2). The next complete polynomial family has embedding degree 12 and is parameterized by:

$$p(x) = \frac{1}{8}(x^{14} - 4x^{10} + 2x^8 + 4x^6 - 2x^4 + 5x^2 + 2), \quad t(x) = x^2 + 1, \quad r(x) = \Phi_{24}(x),$$

with $x \equiv 1 \pmod 2$.

Twisted Ate pairing: This family produces elliptic curves which admit quadratic twists and thus for the twisted Ate pairing the number of iterations in Miller's loop derives from the size of $T_6 = (t-1)^6 \bmod r$. For $x = 4295114753$ we get that $\log T_6 = 256$, with $\mathrm{wt}(T_6) = 97$. The coefficients of the elliptic curve in Weierstrass form are:

$A = 10353841140864156393850331915848774030682220612200405154255017002541059163488608120808766441757343152957214120074465669038166665814946$

$B = 39473289363775963104102109451594297948691152843912940945273433085571876583864616864154630093167743249959592989117035943541619831594138$

Optimal Ate pairing: The shortest vector for this family is $V = [x^2, -1, 0, 0]$ and the formula for the optimal Ate pairing is

$$\widehat{a}_o(Q, P) = f_{x^2,Q}(P)^{\frac{p^k-1}{r}}.$$

We choose $s = x^2$, for x as in the case of the twisted Ate pairing. Here $b_s = 64$ and $w_s = 10$.

	JQ, $\widehat{a}_6$	JQ, $\widehat{a}_o$	Ed, $\widehat{a}_6$	Ed, $\widehat{a}o$
$\mathbf{s = m}$	96807	56592	96039	54000
$\mathbf{s = 0.8m}$	89035.8	51084	88338	49010.4

The best choice for this family is to use the optimal Ate pairing with twisted Edwards curves. Our MAGMA implementation runs this example in 41ms.

Family 9 (embedding degree 14, CM discriminant D = 1). The next polynomial family is a *complete family with variable discriminant (CVD)*. The construction of such families is similar to complete families, however the CM discriminant is not a fixed value, but varies depending on the input of the polynomial family (see [14] for more details). In this case, a CVD polynomial family $[p(x), t(x), r(x)]$ satisfies $4p(x) - t(x)^2 = xy(x)^2$ and thus we need to find an input of the form $x = Dy^2$, for some square-free $D > 0$, such that $p(x)$ and $r(x)$ are both primes of a desired size. For a desired security level of S-bits, this can be easily done by fixing a square-free value $D > 0$ and searching for a $y \in \mathbb{Z}$, such that

$$\log y = \frac{1}{2}\left(\frac{2S - \log \mathrm{lc}(r)}{\deg r} - \log D\right),$$

where $\mathrm{lc}(r)$ is the leading coefficient of the polynomial $r(x)$.

The following is an example of a CVD family with embedding degree $k = 14$, which is parametrized by the polynomials:

$$p(x) = \frac{1}{4}\left(x^9 - 2x^8 + x^7 + x^2 + 2x + 1\right), \quad t(x) = x + 1, \quad r(x) = \Phi_{14}(x)$$

with $x \equiv 1 \bmod 2$. This family has $\rho = 1.5$, so for 128-bit security level we get an extension field $\mathbb{F}_{p^{14}}$ of 5376-bits. We fix $D = 1$ and choose $x = D \cdot 1901697^2$ Note that with this choice the prime r is 250-bits and the base field prime p 373-bits, corresponding to an extension field of 5222-bits.

Twisted Ate pairing: These curves admit only quadratic twists, so the only choice is to set $T_7 \equiv (t-1)^7 \pmod r$. Then $b_{T_7} = 250$ and $w_{T_7} = 124$.

Optimal Ate pairing: We choose the shortest vector $V = [x_0, -1, 0, 0, 0, 0]$, then the formula for the optimal Ate pairing is:

$$\widehat{a}_o(Q, P) = f_{x,Q}(P)^{\frac{p^k - 1}{r}}.$$

For x as above we get $b_x = 41$ and $w_x = 15$.

	JQ, $\widehat{a}_7$	JQ, $\widehat{a}_o$	Ed, $\widehat{a}_7$	Ed, $\widehat{a}o$
$\mathbf{s} = \mathbf{m}$	132744	59066	131760	53578
$\mathbf{s} = 0.8\mathbf{m}$	122560.2	54224.8	121650.6	49266

The best choice for this family is the optimal Ate pairing applied to twisted Edwards curves. Our MAGMA implementation runs this example in 41ms.

5 Conclusion

We give a comprehensive comparison of the competing proposals put forward in the literature for curve shapes and pairing choices for elliptic curves with even embedding degree, for each known TNFS-secure complete pairing-friendly family for 128-bit security level. We additionally provide the formulas for the

'gaps' in the literature: utilizing quadratic twists for pairings on $\mathbb{G}_2 \times \mathbb{G}_1$ with Jacobi Quartic and Edwards curves, and utilizing sextic twists for pairings on $\mathbb{G}_1 \times \mathbb{G}_2$ with Jacobi Quartic curves.

Our comparisons show that, from the currently known TNFS-secure families, the best pairing implementation choice for 128-bit security is the optimal Ate pairing applied to the Jacobi Quartic elliptic curve $E/\mathbb{F}_p : y^2 = dx^4 + 1$ (utilizing quartic twists), where

$$p = 335195198248666745383733884845272660602803360693506589646955234884290813359608015196707145328745246977297093796794243857552239134443824272763691057038540 9$$

and

$$d = 837987995621666863459334712113181651507008401733766474117388087210727033399020037991767863321863117443242734491985609643880597836109560681909227642596352.$$

This choice comes from Family 1, for which our MAGMA implementation runs in 16ms. We leave an optimised implementation of this example to future work.

References

1. C. Aréne, T. Lange, M. Naehrig, C. Ritzenthaler, *Faster computation of the Tate pairing*. Journal of Number Theory, Vol. 131, No. 5, pp. 842–857, Elsevier, 2011.
2. P.S.L.M. Barreto, M. Naehrig. *Pairing-Friendly Elliptic Curves of Prime Order*. SAC 2005, LNCS Vol. 3897, pp. 319–331, Springer, 2005.
3. D. Bernstein, P. Birkner, M. Joye, T. Lange, C. Peters. *Twisted Edwards Curves*. AFRICACRYPT 2008, LNCS Vol. 5023, pp. 389–405, Springer, 2008.
4. D. Bernstein, T. Lange. *Faster Addition and Doubling on Elliptic Curves*. ASIACRYPT 2007, LNCS Vol. 4833, pp. 29–50, Springer, 2007.
5. O. Billet, M. Joye. *The Jacobi model of an elliptic curve and side-channel analysis*. AAECC 2003, LNCS Vol. 2643, pp. 34–42, Springer, 2003.
6. D. Boneh, M. Franklin. *Identity-Based Encryption from the Weil Pairing*. SIAM Journal on Computing: 32(3), pp. 586–615, 2003.
7. D. Boneh, B. Lynn, H. Shacham. *Short Signatures From the Weil Pairing*. Journal of Cryptology: 17(4), pp. 297–319, 2004.
8. F. Brezing, A. Weng. *Elliptic Curves Suitable for Pairing Based Cryptography*. Designs, Codes and Cryptography: 37(1), pp. 133–141, 2005.
9. S. Duquesne, N. El Mrabet, E. Fouotsa. *Efficient computation of pairings on Jacobi quartic elliptic curves*. Journal of Mathematical Cryptology, Vol. 8, No. 4, pp. 331–362, De Gruyter, 2014.
10. S. Duquesne, N. El Mrabet, S. Haloui, F. Rondepierre. *Choosing and generating parameters for low level pairing implementation on BN curves*. Cryptology ePrint Archive, Report 2015/1212, https://eprint.iacr.org/2015/1212, 2015.
11. S. Duquesne, L. Ghammam. *Memory-saving computation of the pairing final exponentiation on BN curves*. Groups Complexity Cryptology: 8 (1), pp. 75–90, De Gruyter, 2016.
12. H. Edwards. *A normal form for elliptic curves*. Bulletin of the American Mathematical Society: 44(3), pp. 393–422, 2007.

13. D. Freeman, M. Scott, E. Teske. *A Taxonomy of Pairing-Friendly Elliptic Curves*. Journal of Cryptology, Vol. 23, No. 2, pp. 224–280, Springer, 2010.
14. G. Fotiadis, E. Konstantinou. *TNFS Resistant Families of Pairing-Friendly Elliptic Curves*. Journal of Theoretical Computer Science, Elsevier, 2018 (to appear).
15. G. Frey, H.G. Rück. *A Remark Concerning m-divisibility and the Discrete Logarithm in the Divisor Class Group of Curves*. Mathematics of Computation, Vol. 62, No. 206, pp. 865–874, 1994.
16. H. Hisil, K. Koon-Ho Wong, G. Carter, E. Dawson. *Jacobi quartic curves revisited*. ACISP 2009, LNCS Vol. 5594, pp. 452–468, Springer, 2009.
17. A. Joux. *One Round Protocol for Tripartite DiffieHellman*. Journal of Cryptology: 17 (4), pp. 263–276, 2004.
18. T. Kim, R. Barbulescu. *Extended Tower Number Field Sieve: A New Complexity for the Medium Prime Case*. CRYPTO 2016, LNCS Vol. 9814, pp. 543–571, Springer, 2016.
19. E. Lee, H-S. Lee, C-M. Park. *Efficient and generalized pairing computation on abelian varieties*. IEEE Transactions on Information Theory: 55 (4), pp. 1793–1803, IEEE, 2009.
20. L. Li, H. Wu, F Zhang. *Pairing Computation on Edwards Curves with High-Degree Twists*. Inscrypt 2013, LNCS Vol. 8567, pp. 185–200, Springer, 2014.
21. S. Matsuda, K. Naoki, F. Hess, E. Okamoto. *Optimised versions of the Ate and twisted Ate pairings*. IMACC 2007, LNCS Vol. 4887, pp. 302–312, Springer, 2007.
22. V.S. Miller. *The Weil Pairing, and Its Efficient Calculation*. Journal of Cryptology, Vol. 17, pp. 235–261, 2004.
23. A.J. Menezes, T. Okamoto, S.A. Vanstone. *Reducing Elliptic Curve Logarithms to Logarithms in a Finite Field*. IEEE Transactions on Information Theory, Vol. 39, No. 5, pp. 1639–1646, 1993.
24. M. Naehrig, P.S.L.M. Barreto, P. Schwabe. (2008) *On Compressible Pairings and Their Computation*. Progress in Cryptology AFRICACRYPT 2008. Lecture Notes in Computer Science, Vol. 5023. Springer, 2008.
25. J. Silverman. *The Arithmetic of Elliptic Curves*. Vol. 106 of Graduate Texts in Mathematics, 1986.
26. F. Vercauteren. *Optimal Pairings*. IEEE Transactions on Information Theory: 56 (1), pp. 455–461, IEEE, 2010.
27. H. Wang, K. Wang, L. Zhang, B. Li. *Pairing Computation on Elliptic Curves of Jacobi Quartic Form*. Chinese Journal of Electronics, Vol. 20, No. 4, 2011.
28. C-A. Zhao, F. Zhang, J. Huang. *A note on the Ate pairing*. International Journal of Information Security: 7 (6), pp. 379–382, Springer 2008.