Pairing computation on elliptic curves with efficiently computable endomorphism and small embedding degree

Sorina Ionica[2] and Antoine Joux[1,2]

[1] Université de Versailles Saint-Quentin-en-Yvelines, 45 avenue des États-Unis,
78035 Versailles CEDEX, France
[2] DGA
sorina.ionica,antoine.joux@m4x.org

Abstract. Scott uses an efficiently computable isomorphism in order to optimize pairing computation on a particular class of curves with embedding degree 2. He points out that pairing implementation becomes thus faster on these curves than on their supersingular equivalent, originally recommended by Boneh and Franklin for Identity Based Encryption. We extend Scott's method to other classes of curves with small embedding degree and efficiently computable endomorphism.

1 Introduction

Pairings were first used in cryptography for attacking the discrete logarithm problem on the elliptic curve [21], but nowadays they are also used as bricks for building new cryptographic protocols such as the tripartite Diffie-Hellman protocol [15], identity-based encryption [5], short signatures [6], and others.

A cryptographic pairing is a bilinear map $e : \mathbb{G}_1 \times \mathbb{G}_2 \rightarrow \mathbb{G}_3$, where $\mathbb{G}_1$, $\mathbb{G}_2$ and $\mathbb{G}_3$ are groups of large prime order r. Known pairings on elliptic curves, i.e. the Weil, Tate pairings, map to the multiplicative group of the minimal extension of the ground field $\mathbb{F}_q$ containing the r-th roots of unity. The degree of this extension, denoted usually by k, is called the embedding degree with respect to r. The basic algorithm used in pairing computation was given by Miller and is an extension of the double-and-add method for finding a point multiple. The cost of the computation heavily depends on costs of operations in $\mathbb{F}_{q^k}$. Consequently, in practice we need curves with small embedding degree.

The reduction of the loop length in Miller's algorithm is one of the main directions taken by research in pairing computation during the past few years. These results concern only pairings on $\mathbb{G}_1 \times \mathbb{G}_2$ or $\mathbb{G}_2 \times \mathbb{G}_1$, where subgroups $\mathbb{G}_1$ and $\mathbb{G}_2$ are given by

$$\mathbb{G}_1 = E[r] \cap \mathrm{Ker}(\pi - [1]) \quad \text{and} \quad \mathbb{G}_2 = E[r] \cap \mathrm{Ker}(\pi - [q]),$$

where π is the Frobenius morphism of E, i.e. $\pi : E \rightarrow E,\ (x, y) \mapsto (x^q, y^q)$. The pairings computed by the new algorithms [2, 13] are actually powers of the

Tate pairing and are called in the literature the Eta ($\mathbb{G}_1 \times \mathbb{G}_2$, $\mathbb{G}_2 \times \mathbb{G}_1$), Ate ($\mathbb{G}_2 \times \mathbb{G}_1$) and twisted Ate pairing ($\mathbb{G}_1 \times \mathbb{G}_2$).

Furthermore, Hess and, independently, Vercauteren [12, 27] showed that on some families of curves with small Frobenius trace, the complexity of Miller's algorithm is $\mathcal{O}(\frac{1}{\varphi(k)} \log r)$, where φ is the Euler totient function.

In this paper, we propose the use of efficiently computable endomorphisms, other than the Frobenius map, to optimize pairing computation. Our method, which works on curves having a small embedding degree, improves pairing computation on curves constructed by the Cocks-Pinch method.

The remainder of this paper is organized as follows. Section 2 presents background on pairings and the Cocks-Pinch method for constructing curves with complex multiplication. Section 3 presents our results which make use of endomorphisms to compute pairings. Section 4 presents an evaluation of costs of an implementation of our algorithm and compares performances to those of the Tate pairing computation. In Appendix 7 we give examples of curves constructed using the Cocks-Pinch method, with small embedding degree and endomorphism of small degree.

2 Background on pairings

the definition of the Tate pairing and of Miller's algorithm [20] used in pairing computations. This algorithm heavily relies on the double-and-add method for finding a point multiple. Let E be an elliptic curve given by a Weierstrass equation:

$$y^2 = x^3 + a_4 x + a_6, \tag{1}$$

defined over a finite field $\mathbb{F}_q$, with $\mathrm{char}(\mathbb{F}_q) \neq 2, 3$. Let P_∞ denote the neutral element on the elliptic curve. Consider r a large prime dividing $\#E(\mathbb{F}_q)$ and k the embedding degree with respect to r.

Let P be an r-torsion point and for any integer i, denote by $f_{i,P}$ a function with divisor $\mathrm{div}\,(f_{i,P}) = i(P) - (iP) - (i-1)(P_\infty)$ (see [24] for an introduction to divisors). Note that $f_{r,P}$ is such that $\mathrm{div}\,(f_{r,P}) = r(P) - r(P_\infty)$.

In order to define the Tate pairing we take Q a point in $E(\mathbb{F}_{q^k})$ representing an element of $E(\mathbb{F}_{q^k})/rE(\mathbb{F}_{q^k})$. Let T be a point on the curve such that the support of the divisor $D = (Q+T) - (T)$ is disjoint from the one of $f_{r,P}$. We then define the Tate pairing as

$$t_r(P, Q) = f_{r,P}(D). \tag{2}$$

This value is a representative of an element of $\mathbb{F}^*_{q^k}/(\mathbb{F}^*_{q^k})^r$. However for cryptographic protocols it is essential to have a unique representative so we will raise it to the $((q^k-1)/r)$-th power, obtaining an r-th root of unity. We call the resulting value the *reduced* Tate pairing

$$T_r(P, Q) = t_r(P, Q)^{\frac{q^k-1}{r}}.$$

As stated in [10], if the function $f_{r,P}$ is normalized, i.e. $(u^r_{P_\infty} f_{r,P})(P_\infty) = 1$ for some $\mathbb{F}_q$-rational uniformizer u_{P_∞} at P_∞, then one can ignore the point T and compute the pairing as

$$T_r(P, Q) = f_{r,P}(Q)^{(q^k-1)/r}.$$

In the sequel of this paper we only consider normalized functions. Before going into the details of Miller's algorithm, we recall the standard addition law on an elliptic curve in Weierstrass form. Suppose we want to compute the sum of iP and jP for $i, j \geq 1$. Let l be the line through iP and jP. Then l intersects the cubic curve E at one further point R according to Bezout's theorem (see [11]). We take v the line between R and P_∞ (which is a vertical line when R is not P_∞). Then v intersects E at one more point and we define the sum of iP and jP to be this point.

The lines l and v are functions on the curve E and the corresponding divisors are

$$\begin{aligned} \operatorname{div}(l) &= (iP) + (jP) + (R) - 3(P_\infty), \\ \operatorname{div}(v) &= (R) + ((i+j)P) - 2(P_\infty). \end{aligned}$$

One can then easily check the following relation:

$$f_{i+j,P} = f_{i,P} f_{j,P} \frac{l}{v}. \tag{3}$$

Turning back to Miller's algorithm, suppose we want to compute $f_{r,P}(Q)$. We compute at each step of the algorithm on one side mP, where m is the integer with binary expansion given by the i topmost bits of the binary expansion of r, and on the other side $f_{m,P}$ evaluated at Q, by exploiting the formula above. We call the set of operations executed for each bit i of r a *Miller operation*.

Algorithm 1 Miller's algorithm

INPUT: An elliptic curve E defined over a finite field $\mathbb{F}_q$, P an r-torsion point on the curve and $Q \in E(\mathbb{F}_{q^k})$.

OUTPUT: the Tate pairing $t_r(P, Q)$.

```
Let $i = [\log_2(r)]$, $K \leftarrow P$,$f \leftarrow 1$
while $i \geq 1$ do
    Compute the equation of $l$ arising in the doubling of $K$
    $K \leftarrow 2K$ and $f \leftarrow f^2 l(Q)$
    if the $i$-th bit of $r$ is 1 then
        Compute the equation of $l$ arising in the addition of $K$ and $P$
        $K \leftarrow P + K$ and $f \leftarrow f l(Q)$
    end if
    Let $i \leftarrow i - 1$.
end while
return $f$.
```

Implementing pairings. In implementations, we usually prefer curves with even embedding degree. On these curves, thanks to the existence of twists, most computations in a Miller operation are done in proper subfields of $\mathbb{F}_{q^k}$. Moreover, thanks to the final exponentiation, terms contained in proper subfields of $\mathbb{F}_{q^k}$ can be ignored (see [18]). Algorithm 1 gives the pseudocode of Miller's algorithm for curves with even embedding degree. In Table 1 we give costs for the doubling and the addition step in Algorithm 1 for an implementation on $\mathbb{G}_1 \times \mathbb{G}_2$, on curves with twists[3] of degree d. We denote by $\mathbf{m}, \mathbf{s}$ the costs of multiplication and squaring in $\mathbb{F}_q$ and $\mathbf{M}, \mathbf{S}$ the costs of multiplication and squaring in $\mathbb{F}_{q^k}$.

Table 1. Cost of one step in Miller's algorithm for even embedding degree

	Doubling	Mixed addition
$\mathcal{J}$ [1, 14]	$(1+k)\mathbf{m} + 11\mathbf{s} + 1\mathbf{M} + 1\mathbf{S}$	$(6+k)\mathbf{m} + 6\mathbf{s} + 1\mathbf{M}$
$\mathcal{J}$, $y^2 = x^3 + b$ $d = 2, 6$ [7]	$(2k/d+2)\mathbf{m} + 7\mathbf{s} + 1\mathbf{M} + 1\mathbf{S}$	$(2k/d+9)\mathbf{m} + 2\mathbf{s} + 1\mathbf{M}$
$\mathcal{J}$, $y^2 = x^3 + ax$ $d = 2, 4$ [7]	$(2k/d+2)\mathbf{m} + 8\mathbf{s} + 1\mathbf{M} + 1\mathbf{S}$	$(2k/d+12)\mathbf{m} + 4\mathbf{s} + 1\mathbf{M}$

Security issues. A secure pairing-based cryptosystem needs to be implemented on elliptic curve subgroups $\mathbb{G}_1$ and $\mathbb{G}_2$ such that the discrete logarithm problem is computationally difficult in $\mathbb{G}_1$, $\mathbb{G}_2$ and in $\mathbb{F}_{q^k}^*$. The best known algorithm for computing discrete logarithms on elliptic curves is the Pollard-rho method [25, 22], which has complexity $O(\sqrt{r})$, where r is the order of the groups $\mathbb{G}_1$ and $\mathbb{G}_2$. Meanwhile, the best known algorithm for solving the discrete logarithm problem in the multiplicative group of a finite field is the index calculus algorithm, which has sub-exponential running time [17, 16]. Consequently, in order to achieve the same level of security in both the elliptic curve subgroups and in the finite field subgroup, we need to choose a q^k which is significantly larger than r. It is therefore interesting to consider the ratio of these sizes

$$\frac{k \log q}{\log r}.$$

As the efficiency of the implementation will depend critically on the so-called ρ-value

$$\rho = \frac{\log q}{\log r},$$

it is preferable to keep this value as small as possible.

[3] We briefly recall that a twist E' of degree d of an elliptic curve E defined over $\mathbb{F}_q$ is a curve isomorphic to E, such that the isomorphism between the two curves is minimally defined over $\mathbb{F}_{q^d}$. The reader should check [24] for more details on twists.

The Cocks-Pinch method for constructing pairing friendly curves. Let E be an ordinary curve defined over a finite field $\mathbb{F}_q$. We denote by π the Frobenius morphism and by t its trace. Given the fact that curve must have a subgroup of large order r and that the number of points on the curve is $\#E(\mathbb{F}_q) = q + 1 - t$ we write

$$q + 1 - t = hr.$$

Furthermore, the fact that the Frobenius is an element of an order in a quadratic imaginary field $\mathbb{Q}(\sqrt{-D})$ gives

$$Dy^2 = 4q - t^2 = 4hr - (t-2)^2.$$

To sum up, in order to generate a pairing friendly curve, we are looking for q, r, k, D, t and y satisfying the following conditions

$$\begin{aligned} r &\mid Dy^2 + (t-2)^2, \\ r &\mid q^k - 1, \\ t^2 + Dy^2 &= 4q. \end{aligned}$$

Cocks and Pinch gave an algorithm (which is presented in [4]) which finds, given r and a small k, parameters q prime and t satisfying the equations above.

Algorithm 2 The Cocks-Pinch algorithm

```
INPUT: $k$,$r$ a prime number, $D$ and $k|(r-1)$.
OUTPUT: $q$, $t$ such that there is a curve with CM by $\sqrt{-D}$ over $\mathbb{F}_q$ with $q+1-t$
    points where $r|(q+1-t)$ and $r|(q^k-1)$.
Choose a primitive $k$-th root of unity $g$ in $\mathbb{F}_r$.
Choose an integer $t \leftarrow g + 1 (\text{mod } r)$.
if $\gcd(t, D) \neq 1$ then
    exit (or choose another $g$).
end if
Choose an integer $y_0 = \pm(t-2)/\sqrt{-D} (\text{mod } r)$.
$j \rightarrow 0$
repeat
    $q \leftarrow (t^2 + D(y_0 + jr)^2)/4$
    $j \leftarrow j + 1$
until $q$ is prime
return $q$ and $t$
```

This method produces ordinary curves with a ρ-value approximatively 2, which is less preferred in practice. However, Vercauteren [8, Prop. 7.1] showed that for certain embedding degrees and certain values of the CM discriminant $-D$, there are no ordinary curves with smaller ρ-value.

Proposition 1. *Let E be an elliptic curve over $\mathbb{F}_q$ with a subgroup of prime order $r > 3$ and embedding degree $k > 1$ with respect to r. If E has a twist $E'/\mathbb{F}_q$ of degree k and $r \geq 4\sqrt{q}$, then E is supersingular.*

It follows that in some cases, like $k = 2$ or $k = 4$ and [4]discriminant -4, the curves produced by the Cocks-Pinch algorithm have optimal ρ-value. Moreover, with this method we may choose the value of r from the very beginning. This is an advantage, because we may choose r with low Hamming weight. On curves with such r, in Algorithm 1, we perform mostly doublings and very few additions.

The Eta pairing and its variants. To our knowledge, isogenies were proposed to speed up pairing computation for the first time by Barreto et al. [2], who introduced the Eta pairing. This idea was extended by Hess et al. [13]. We present here the main result in [13], without giving the proof.

Theorem 1. *Let E be an elliptic curve defined over $\mathbb{F}_q$, r a large prime with $r|\#E(\mathbb{F}_q)$ and k the embedding degree with respect to r. Assume that $k > 1$ and denote by t the trace of the Frobenius.*

(a) For $T = t - 1$, $Q \in \mathbb{G}_2 = E[r] \cap Ker(\pi - [q])$, $P \in \mathbb{G}_1 = E[r] \cap Ker(\pi - [1])$ we have

(i) $f_{T,Q}(P)$ defines a bilinear pairing, which we call the Ate pairing;

(ii) Let $N = gcd(T^k - 1, q^k - 1)$ and $T^k - 1 = LN$, with k the embedding degree, then

$$t_r(Q, P)^L = f_{T,Q}(P)^{c(q^k-1)/N}$$

where $c = \sum_{i=0}^{k-1} T^{k-1-i}q^i \equiv kq^{k-1}$ mod r. For $r \nmid L$, the Ate pairing is non-degenerate.

(b) Assume E has a twist of degree d and set $m = gcd(k, d)$ and $e = k/m$. We denote by $c = \sum_{i=0}^{m-1} T^{e(m-1-i)}q^{ei} \equiv mq^{e(m-1)}$ mod r. We have

(i) $f_{T^e,P}(Q)$ defines a bilinear pairing, which we call the twisted Eta pairing;

(ii) $t_r(P, Q)^L = f_{T^e,P}(Q)^{c(q^k-1)/N}$ and the twisted Eta pairing is non-degenerate if $r \nmid L$.

The Ate and twisted Eta pairing can be computed using Miller's algorithm with a loop length of $\log T$ and $\log T^e$, respectively. Consequently, if the trace t is smaller than r, these pairings may be significantly faster than the Tate pairing.

3 Speeding up pairing computation using endomorphisms of small degree

The following result was given by Verheul [28], whose purpose was to investigate the existence of distortion maps for points of order r, i.e. maps ϕ such that for a point P, $\phi(P) \notin \langle P \rangle$. We will make use of this result to improve pairing computation.

Theorem 2. *Let E be an ordinary elliptic curve defined over $\mathbb{F}_q$ and let P be a point over $\mathbb{F}_q$ of E, whose order is a prime integer $r \neq q$. Suppose the embedding degree k is greater than 1 and denote by Q a point defined over $\mathbb{F}_{q^k}$, such that $\pi(Q) = qQ$. Then P and Q are eigenvectors of any other endomorphism of E.*

[4] Only curves with discriminant -4 have twists of degree 4.

Proof. Let ϕ be an endomorphism of E. For the point P we have

$$\phi(\pi(P)) = \pi(\phi(P)) \text{ and } \phi(\pi(P)) = \phi(P). \tag{4}$$

The first equality comes from the fact that the ring $\mathrm{End}(E)$ is commutative, the second one is due to the fact that $P \in E(\mathbb{F}_q)$. It follows that $\pi(\phi(P)) = \phi(P)$, so $\phi(P)$ is an eigenvector for the eigenvalue 1 of π. This means that $\phi(P) \in \langle P \rangle$. The proof for Q is similar.

Notation 1 *In the sequel we denote the correction of two points R_1 and R_2 as follows:*

$$corr_{R_1,R_2} = \frac{l_{R_1,R_2}}{v_{R_1+R_2}},$$

where l_{R_1,R_2} is the line passing through R_1 and R_2 and $v_{R_1+R_2}$ is the vertical line through $R_1 + R_2$.

Our starting idea is a method to exploit efficiently computable endomorphisms in pairing computation suggested by Scott [23], for a family of curves called NSS. These curves are defined over $\mathbb{F}_q$ with $q \equiv 1 \bmod 3$ and given by an equation of the form $y^2 = x^3 + B$. Since they have $k = 2$ and $\rho \sim 2$, the Eta and Ate pairings will not bring any improvement to pairing computation. However, these curves admit an endomorphism $\phi : (x, y) \rightarrow (\beta x, y)$, where β is a non-trivial cube root of unity. Its characteristic equation is $\phi^2 + \phi + 1 = 0$. If P is an eigenvalue of ϕ such that $\phi(P) = \lambda P$, then λ verifies the equation

$$\lambda^2 + \lambda + 1 = cr.$$

We obtain

$$f_{r,P}^c(Q) = f_{\lambda^2+\lambda,P}(Q) = f_{\lambda(\lambda+1),P}(Q) = f_{\lambda,P}^{\lambda+1}(Q) \cdot f_{\lambda+1,[\lambda]P}(Q) \cdot \frac{l_{[\lambda]P,P}}{v_{[\lambda+1]P}}.$$

Since for $P = (x, y)$, λP is given by $(\beta x, y)$, we can easily compute $f_{\lambda,\lambda P}(Q)$ and $f_{\lambda,P}(Q)$ at the same time when running Miller's algorithm, by replacing x with βx when computing doublings, additions and line equations. Note that pairing computation on these curves has been recently improved by Zhao and al. [29].

We apply similar techniques to curves with endomorphisms that verify a characteristic equation $x^2 + ax + b = 0$, with a, b small. In all cases, we use the Cocks-Pinch method to construct curves such that there is a $\lambda \sim \sqrt{r}$ which verifies $\lambda^2 + a\lambda + b = cr$. This can be done by exhaustive search on λ. Thanks to the density of prime numbers, we are able to produce couples (λ, r) within seconds with MAGMA.

We obtain a new algorithm for pairing computation, whose loop is shorter than that of the algorithm computing the Tate pairing.

Lemma 1. *Let E be an elliptic curve defined over a finite field $\mathbb{F}_q$ and ϕ an endomorphism of E whose degree is b. Let P, Q be two points on the curve E. Then for any integer λ the following equality is true up to a constant:*

$$f_{\lambda,\phi(P)}(\phi(Q)) = f^b_{\lambda,P}(Q)\left(\prod_{K\in Ker\,\phi\backslash\{P_\infty\}} corr_{P,K}(Q)\right)^{\lambda}\left(\prod_{K\in Ker\,\phi\backslash\{P_\infty\}} corr_{\lambda P,K}(Q)\right)^{-1}$$

Proof. We have

$$\begin{aligned}
&\phi^*(f_{\lambda,\phi(P)}) = \lambda\sum_{K\in\mathrm{Ker}\phi}(P+K) - \sum_{K\in\mathrm{Ker}\phi}(\lambda P+K) - (\lambda-1)\sum_{K\in\mathrm{Ker}\phi}(K)\\
&= \lambda\sum_{K\in\mathrm{Ker}\phi}((P+K)-(K)) - \sum_{K\in\mathrm{Ker}\phi}((\lambda P+K)-(K))\\
&= \lambda\sum_{K\in\mathrm{Ker}\phi}((P)-(O)) - \sum_{K\in\mathrm{Ker}\phi}(\lambda P)-(O) + \mathrm{div}\left(\left(\prod_{K\in\mathrm{Ker}\,\phi}\frac{l_{K,P}}{v_{K+P}}\right)^{\lambda}\right)\\
&-\mathrm{div}\left(\prod_{K\in\mathrm{Ker}\,\phi}\frac{l_{K,\lambda P}}{v_{K+\lambda P}}\right) = \mathrm{div}(f_{\lambda,P}) + \mathrm{div}\left(\prod_{K\in\mathrm{Ker}\,\phi\backslash\{P_\infty\}} corr_{\lambda P,K}\right)\\
&-\mathrm{div}\left(\prod_{K\in\mathrm{Ker}\,\phi\backslash\{P_\infty\}} corr_{\lambda P,K}\right).
\end{aligned}$$

Using the fact that $\phi^*(f_{\lambda,\phi(P)}) = f_{\lambda,\phi(P)}\circ\phi$, we obtain the equality we have announced.

In the sequel, we make use of the following relation which holds for all $m, n \in \mathbb{Z}$ and any point P on the elliptic curve

$$f_{mn,P} = f^n_{m,P}\cdot f_{n,mP}. \tag{5}$$

This equality can be easily checked using divisors. In the sequel, we denote by $\hat{\phi}$ the dual of an isogeny ϕ. The reader is referred to [24] for the definition of the dual.

Theorem 3. *Let E be an elliptic curve defined over a finite field $\mathbb{F}_q$, r a prime number such that $r|\#E(\mathbb{F}_q)$ and k the embedding degree with respect to r. Let ϕ be an efficiently computable separable endomorphism of E, whose characteristic equation is $X^2 + aX + b = 0$. Let $\mathbb{G}_1$ and $\mathbb{G}_2$ be the the subgroups of order r whose elements are eigenvectors of ϕ defined over $\mathbb{F}_q$ and $\mathbb{F}_{q^k}$, respectively. Let λ be the eigenvalue of ϕ on $\mathbb{G}_1$, verifying $\lambda^2 + a\lambda + b = cr$, with $r \nmid bc$. Then the* *map* $a_\phi(\cdot,\cdot) : \mathbb{G}_1 \times \mathbb{G}_2 \to \mathbb{F}_{q^k}^*/(\mathbb{F}_{q^k}^*)^r$ *given by*

$$a_\phi(P,Q) = f_{\lambda,P}^{\lambda+a}(bQ) f_{\lambda,P}^{b}(\hat{\phi}(Q)) f_{a,\lambda P}(bQ) f_{b,P}(bQ) \left(\prod_{K\in Ker\phi\backslash\{P_\infty\}} corr_{P,K}(\hat{\phi}(Q))\right)^{\lambda}$$
$$\cdot \left(\prod_{K\in Ker\phi\backslash\{P_\infty\}} corr_{\lambda P,K}(\hat{\phi}(Q))\right)^{-1} corr_{\lambda^2 P,a\lambda P}(bQ)\ l_{\lambda^2 P+a\lambda P,bP}(bQ)$$

is a bilinear non-degenerate pairing.

Proof. The following equality is obtained by repeatedly applying the equality at (5)

$$\begin{aligned} f_{\lambda^2+a\lambda+b,P} = {} & (f_{\lambda,P}^{\lambda}) \cdot (f_{\lambda,\lambda P}) \cdot (f_{\lambda,P}^{a}) \cdot (f_{a,\lambda P}) \cdot (f_{b,P}) \\ & \cdot corr_{\lambda^2 P,a\lambda P} \cdot l_{\lambda^2 P+a\lambda P,bP}. \end{aligned} \tag{6}$$

By applying Lemma 1, we obtain

$$f_{\lambda,\lambda P}(bQ) = f_{\lambda,P}^{b}(\hat{\phi}(Q)) \left(\prod_{K\in \mathrm{Ker}\phi\backslash\{P_\infty\}} corr_{P,K}(\hat{\phi}(Q))\right)^{\lambda}$$
$$\cdot \left(\prod_{K\in \mathrm{Ker}\phi\backslash\{P_\infty\}} corr_{\lambda P,K}(\hat{\phi}(Q))\right)^{-1}$$

By replacing this term in equation (6), we derive that $a_\phi(P,Q)$ is a power of $t_r(P,Q)$. Since $(bc,r)=1$, we conclude that a_ϕ defines a non-degenerate pairing on $\mathbb{G}_1 \times \mathbb{G}_2$.

If the value of λ is close to $\sqrt{r}$ and a and b are small, Theorem 3 gives an efficient algorithm to compute the Tate pairing (actually a small power of the Tate pairing). This is Algorithm 3. The complexity of the new algorithm is $O(\log ab\lambda)$.

4 Computational costs

Suppose we use an endomorphism ϕ whose characteristic equation is

$$\phi^2 + a\phi + b = 0,$$

with a and b small. We also neglect the cost of computing the dual of ϕ at Q, $\hat{\phi}(Q)$, because $\hat{\phi}$ can be precomputed by Vélu's formulae [26] and is given by polynomials of small degree. Note that in some protocols Q is a fixed point, so all the precomputations on this point may be done before the computation of the pairing.

Algorithm 3 Our algorithm for pairing computation for curves with an efficiently computable endomorphism

```
INPUT: An elliptic curve $E$, $P, Q$ points on $E$ and $\phi$ such that $\phi(P) = \lambda P$, $Q' = \hat{\phi}(Q)$.
OUTPUT: A power of the Tate pairing $T_r(P, Q)$.
 1: Let $i = [\log_2(\lambda)]$, $K \leftarrow P$, $f \leftarrow 1$, $g \leftarrow 1$
 2: while $i \geq 1$ do
 3:     Compute equation of $l$ arising in the doubling of $K$
 4:     $K \leftarrow 2K$ and $f \leftarrow f^2 l(bQ)$ and $g \leftarrow g^2 l(Q')$
 5:     if the $i$-th bit of $\lambda$ is 1 then
 6:         Compute equation of $l$ arising in the addition of $K$ and $P$
 7:         $K \leftarrow P + K$ and $f \leftarrow fl(bQ)$ and $g \leftarrow gl(Q')$
 8:     end if
 9:     Let $i \leftarrow i - 1$
10: end while
11: Compute $A \leftarrow f^{\lambda+a}$
12: Compute $B \leftarrow g^b$
13: Compute $C \leftarrow \left(\prod_{K \in \mathrm{Ker}\phi \setminus \{P_\infty\}} corr_{P,K}(Q')\right)^{\lambda} \left(\prod_{K \in \mathrm{Ker}\phi \setminus \{P_\infty\}} corr_{\lambda P,K}(Q')\right)^{-1}$
14: Compute $D \leftarrow f_{a,\lambda P}(bQ) f_{b,P}(bQ)$
15: $F \leftarrow corr_{\lambda^2 P, a\lambda P}(bQ) l_{\lambda^2 P + a\lambda P, bP}(bQ)$
16: Return $A \cdot B \cdot C \cdot D \cdot F$
```

We also note that the endomorphism is defined over $\mathbb{F}_q$, because the curve E is ordinary. Moreover, the points in $\mathrm{Ker}\,\phi$ are eigenvectors for the Frobenius endomorphism. Indeed, since $\mathrm{End}(E)$ is a commutative ring, we have $\phi(\pi(K)) = \pi(\phi(K)) = O$, for all $K \in \mathrm{Ker}\,\phi$. It follows that $\pi(K) \in \mathrm{Ker}\,\phi$. Thus the points of $\mathrm{Ker}\,\phi$ are defined over an extension field of $\mathbb{F}_q$ of degree smaller than b. Furthermore, if $\mathrm{Ker}\,\phi$ is cyclic, we have

$$\left(\prod_{K \in \mathrm{Ker}\,\phi \setminus \{P_\infty\}} corr_{P,K}(\hat{\phi}(Q))\right) \in \mathbb{F}_{q^k}.$$

Consequently, given that the degree of ϕ is small, we assume that the number of operations needed to compute the correction $\prod_{K \in \mathrm{Ker}\,\phi} corr_{P,K}(\hat{\phi}(Q))$ is negligible. Since a and b are small, we also assume that the costs of the exponentiation at line 12 and that of the computation of functions at line 14 of Algorithm 3 are negligible.

Since in practice we usually consider curves with even embedding degree, we present only results for these curves. We assume that the curves have an efficiently computable endomorphism and eigenvalues of size $\sqrt{r}$. In our evaluation, we only counted the number of operations performed in the doubling part of Miller's algorithm, because we suppose that λ and r have low Hamming weight (which is possible if the curve is constructed with the Cocks-Pinch method).
For operations in extension fields of degree 2, we use tower fields. For example, to construct an extension field of degree 4 we have

$$\mathbb{F}_q \subset \mathbb{F}_{q^2} \subset \mathbb{F}_{q^4}.$$

With Karatsuba's method the cost of an operation in the extension field of degree 2 is three times the cost of the same operation in the base field, while with Toom-Cook a multiplication in an extension field of degree 3 costs 5 multiplications in the base field. Using the formulas in Table 1 the total cost of the doubling step in Algorithm 3 and of the exponentiation at line 11 is

$$(11\mathbf{s} + (1+2k)\mathbf{m} + 2\mathbf{M} + 2\mathbf{S}) \log \lambda + \log \lambda \mathbf{M} \text{ if } D \neq 3, 4.$$

Our computations showed that our method gives better performances than the Tate pairing for some families of ordinary curves with embedding degree 2, 3 and 4. Indeed, using the complexity estimations above and making the assumption that $\mathbf{s} \approx \mathbf{m}$, our algorithm is faster than the Tate pairing if and only if

$$(12+2k)\mathbf{m} + 5\mathbf{M} > 2((12+k)\mathbf{m} + 2\mathbf{M}).$$

A simple computation shows that this is true if and only if $k \leq 4$. In Table 2, we compare the performances of our method to those of Miller's algorithm, for curves with embedding degree 2 and 4 constructed via the Cocks-Pinch method. Note that for $k = 2$, the Eta pairing algorithm (and its variants) is not faster than the Tate pairing algorithm, because $t \approx r$. We assume that for curves with embedding degree 4 the CM discriminant is not -4, because for such curves the Tate and the twisted Ate pairing have comparable costs. Note that for $D = -4$ the twisted Ate pairing computation has complexity $O(\log t)$ and is thus faster than the Tate pairing and also faster than our method if t is careffully chosen of small size.

Table 2. Our method versus the Tate pairing

bit length of r	$k = 2$		$k \geq 4$ and $D \neq 4$	
	Tate pairing	This work	Tate pairing	This work
160 bits	3040	2400	5120	4880

As explained in [8], curves with small embedding degrees are preferred in implementations at low security levels (80 bits). Thus, if we want to set up a pairing-based cryptosystem with a 160-bit elliptic curve subgroup, we may choose a MNT curve with embedding degree 6 and ρ-value close to 1 or we may take a curve with embedding degree $k \in \{2, 3, 4\}$ and ρ-value close to 2. Table 3 presents a list of families of curves that have been proposed for pairing-based cryptography at 80 bits security level. Note that we may use an ordinary curve with embedding degree 2 and ρ-value approximatively 3 constructed by the Cocks-Pinch method or a supersingular curve with $k = 2$ and $\rho \sim 3$ (see Algorithm 3.3 in [8]). Table 6 presents the operation count in $\mathbb{F}_q$ for pairing computation on curves with different embedding degrees. We assume that ordinary curves with embedding degrees 2 are constructed via the Cocks-Pinch method and we evaluate the cost of the computation performed in Algorithm 3 for these curves, as explained above. For supersingular curves and MNT curves with embedding degree 6 we estimate the cost of the doubling part in Algorithm 1 computing the Tate pairing using formulae in Table 1. Since on these curves, the parameter r does not necessarily have low Hamming weight, we also count the number of operations performed in the mixed addition part of the Miller operation, if a NAF representation of r is used. For $k = 4$, the Cocks-Pinch method allows constructing curves with small trace of the Frobenius (by choosing a small g and then taking r a divisor of $\Phi_4(g)$ in Algorithm 2). In this case, if the curve has a twist o degree 4, i.e. $D = 4$, then the twisted Ate pairing is optimal. Otherwise, the Tate pairing is optimal. Finally, we also give a family with $k = 4$ and $D = 3$ from [8]. On curves from this family, the optimal pairing can be computed using lattice reduction in $\frac{\log r}{2}$ Miller iterations (see [12] for details).

Table 3. Bit sizes of curve parameters for pairing-based cryptography at 80 bits security level

bit length of r	bit length of q	bit length of q^k	value of k and ρ
160	480	960	supersingular curves $k = 2$ and $\rho \sim 3$
160	480	960	ordinary curves $k = 2$ and $\rho \sim 3$
160	160	960	MNT curves $k = 6$ and $\rho \sim 1$
160	320	1280	ordinary curves $k = 4$ and $\rho \sim 2$
160	220	960	ordinary curves $k = 4$ and $\rho \sim 1.5$

Table 4. Pairing computation at 80 bits security level

value of k and ρ	doubling step (operations in $\mathbb{F}_q$)	mixed addition (operations in $\mathbb{F}_q$)
supersingular curves $k = 2$ and $\rho \sim 3$	3040	-
ordinary curves $k = 2$ and $\rho \sim 3$	2400	-
MNT curves $k = 6$ and $\rho \sim 1$	7680	1760
ordinary curves $k = 4$ and $\rho \sim 2$ and $D = 4$	2400	-
ordinary curves $k = 4$ and $\rho \sim 1.5$ and $D = 3$	2480	-

Table 5 presents total costs for the Miller loop and for the final exponentiation, in terms of number of operations in $\mathbb{F}_q$, for different types of curves and embedding degrees. In the final exponentiation, we assume that applying the Frobenius operator can be done for free and we estimate only the cost of $\Phi_r(q)/r$. The last column presents global costs of pairing computation. Note that the size of q is different for these families of curves. We have therefore taken into account costs of integer multiplication for different bit lengths (see [3] for GMP benchmarks).

We have realised a simple implementation under MAGMA 2.15-1.5. We give the time in seconds for the computation of 100 pairings on a 2.6 GHz Intel Core

Table 5. Pairing computation at 80 bits security level

value of k and ρ	Miller loop	final exponentiation	total cost
supersingular curves $k = 2$ and $\rho \sim 3$	13680	5760	19350
ordinary curves $k = 2$ and $\rho \sim 3$	10800	5760	16470
MNT curves $k = 6$ and $\rho \sim 1$	9440	6400	15840
$k = 4$ and $\rho \sim 2$ and $D = 4$	5880	14138	20018
$k = 4$ and $\rho \sim 1.5$ and $D = 3$	3100	4980	8080

2 Duo processor. Note that in theory an implementation of pairings on MNT curves is expected to be faster.

Table 6. Execution time on a 2.6 GHz Intel Core 2 Duo processor

value of k and ρ	bit length of r	This work	Eta/Tate
ordinary curves $k = 2$ and $\rho \sim 3$	160	1.02	1.42
MNT curves $k = 6$ and $\rho \sim 1$	172	-	1.96
ordinary curves $k = 4$ and $\rho \sim 1.5$ and $D = 3$	160	-	0.41

We conclude by giving in Appendix 7 examples of curves constructed with the Cocks-Pinch method, with endomorphism verifying an equation of the form $X^2 + aX + b = 0$ and roots $\lambda \sim \sqrt{r}$. We also note that on these curves, the GLV method [9] can be used to speed up scalar multiplication in the implementation of a pairing-based protocol. We therefore believe that these curves offer a good choice for pairing-based cryptography at 80 bits security level.

5 Conclusion

We have given a new algorithm for pairing computation on curves with endomorphisms of small degree. Our pairing on curves constructed with the Cocks-Pinch method is more efficient than than known pairings for some curves with embedding degree 2 and 4.

6 Acknowledgements

The authors thank anonymous reviewers for helpful comments.

References

1. C. Arène, T. Lange, M. Naehrig, and C. Ritzenthaler. Faster computation of the Tate pairing. http://eprint.iacr.org/2009/155.

2. P. Barreto, S. Galbraith, C. Héigeartaigh, and M. Scott. Efficient Pairing Computation on Supersingular Abelian Varieties. *Des. Codes Cryptography*, 42(3):239–271, 2007.
3. D. Bernstein. Integer multiplication benchmarks. `http://cr.yp.to/speed/mult/gmp.html`.
4. I. F. Blake, G. Seroussi, and N. P. Smart. *Advances in Elliptic Curve Cryptography*, volume 317 of *London Mathematical Society Lecture Note Series*. Cambridge University Press, 2005.
5. D. Boneh and M. K. Franklin. Identity-based encryption from the Weil pairing. In Joe Kilian, editor, *Advances in Cryptology – CRYPTO 2001*, volume 2139 of *Lecture Notes in Computer Science*, pages 213–229. Springer Verlag, 2001.
6. D. Boneh, B. Lynn, and H. Shacham. Short signatures from the Weil pairing. In Colin Boyd, editor, *Advances in Cryptology – ASIACRYPT 2001*, volume 2248 of *Lecture Notes in Computer Science*, pages 514–532. Springer Verlag, 2001.
7. C. Costello, T. Lange, and M. Naehrig. Faster Pairing Computations on Curves with High-Degree Twists. In P.Q. Nquyen and D. Pointcheval, editors, *Public Key Cryptography-PKC 2010*, volume 6056, pages 224–242. Springer, 2010.
8. D. Freeman, M. Scott, and E. Teske. A taxonomy of pairing-friendly elliptic curves. *Journal of Cryptology*, 23:224–280, 2010.
9. R. P. Gallant, R. J. Lambert, and S. A. Vanstone. Faster point multiplication on elliptic curves with efficient endomorphisms. In Joe Kilian, editor, *CRYPTO01*, volume 2139 of *Lecture Notes in Computer Science*, pages 190–200. Springer, 2001.
10. R. Granger, F. Hess, R. Oyono, N. Thériault, and F. Vercauteren. Ate pairing on hyperelliptic curves. In Moni Naor, editor, *EUROCRYPT07*, Lecture Notes in Computer Science, pages 430–447. Springer, 2007.
11. R. Hartshorne. *Algebraic geometry*, volume 52 of *Graduate Texts in Mathematics*. Springer, 1977.
12. F. Hess. A note on the Tate pairing of curves over finite fields. *Arch. Math*, 82:28–32, 2004.
13. F. Hess, N. P. Smart, and F. Vercauteren. The Eta pairing revisited. *IEEE Transactions on Information Theory*, 52:4595–4602, 2006.
14. S. Ionica and A. Joux. Another approach to pairing computation in Edwards coordinates. In Dipanwita Roy Chowdhury, Vincent Rijmen, and Abhijit Das, editors, *Progress in Cryptography- Indocrypt 2008*, volume 5365 of *Lecture Notes in Computer Science*, pages 400–413. Springer, 2008.
15. A. Joux. A one round protocol for tripartite Diffie-Hellman. *Journal of Cryptology*, 17(4):263–276, September 2004.
16. A. Joux and R. Lercier. The function field sieve in the medium prime case. In Serge Vaudenay, editor, *Advances in Cryptology: Eurocrypt 2006*, volume 4004 of *Lecture Notes in Computer Science*, pages 254–270, 2006.
17. A. Joux, R. Lercier, N. Smart, and F. Vercauteren. The number field sieve in the medium prime case. In Cynthia Dwork, editor, *Advances in Cryptology- CRYPTO 2006*, volume 4117 of *Lecture Notes in Computer Science*, pages 326–344, 2006.
18. N. Koblitz and A. Menezes. Pairing-based cryptography at high security levels. In Nigel P. Smart, editor, *IMA Int. Conf.*, volume 3796 of *Lecture Notes in Computer Science*, pages 13–36, 2005.
19. MAGMA Computational Algebra System. *MAGMA version V2.16-5*, 2010. http://magma.maths.usyd.edu.au/magma.
20. V. Miller. The Weil pairing, and its efficient calculation. *Journal of Cryptology*, 17(4):235–261, September 2004.

21. T. Okamoto, A. Menezes, and S.A. Vanstone. Reducing elliptic curve logarithms to logarithms in the finite field. In *Proceedings 23rd Annual ACM Symposium on Theory of Computing (STOC)*, pages 80–89. ACM Press, 1991.
22. J. Pollard. Monte Carlo methods for index computation (mod p). *Mathematics of Computation*, (32):918–924, 1978.
23. M. Scott. Faster pairings using an elliptic curve with an efficient endomorphism. In Subhamoy Maitra, C. E. Veni Madhavan, and Ramarathnam Venkatesan, editors, *INDOCRYPT 2005*, volume 3797 of *Lecture Notes in Computer Science*, pages 258–269. Springer, 2005.
24. J. H. Silverman. *The Arithmetic of Elliptic Curves*, volume 106 of *Graduate Texts in Mathematics*. Springer, 1986.
25. P.C. van Oorschot and M.J. Wiener. Parallel collision search with cryptanalytic applications. *Journal of Cryptology*, (12):1–18, 1999.
26. J. Vélu. Isogenies entre courbes elliptiques. *Comptes Rendus De Academie Des Sciences Paris, Serie I-Mathematique, Serie A.*, 273:238–241, 1971.
27. F. Vercauteren. Optimal pairings. *IEEE Transactions on Information Theory*, 2009. to appear.
28. E.R. Verheul. Evidence that XTR is more secure that supersingular elliptic curve cryptosystems. In Birgit Pfitzmann, editor, *Advances in Cryptography: EUROCRYPT 2001*, volume 2045 of *Lecture Notes in Computer Science*, pages 195–201. Springer, 2001.
29. C. Zhao, D. Xie, F. Zhang, J. Zhang, and B. Chen. Computing the Bilinear Pairings on Elliptic Curves with Automorphisms. *Designes, Codes and Cryptography.* to appear.

7 Appendix 1

In order to display the equations of the endomorphism easily, we give first a small example.

Example 1. A toy example

We take $D = -4 \cdot 2$ and we want a curve with an endomorphism whose characteristic equation will be

$$X^2 + 2 = 0.$$

We choose $\lambda = 66543$ verifying the equation $\lambda^2 + 2 = r$, with

$$r = 4427970851.$$

Our implementation of the Cocks-Pinch method in MAGMA [19] found the following curve

$$y^2 = x^3 + 49768875163241226696283x + 22119500072551656427966$$

over the prime field $\mathbb{F}_q$, with

$$q = 149306625489723680888559.$$

This curve has $k = 2$ with respect to r. The endomorphism corresponding to $\alpha = \sqrt{-2}$ in $\mathbb{Z}[\sqrt{-2}]$ is

$$[\alpha](x,y) = \left(7465331274486184044429 \frac{x^2 + 4976887516324122696285x + 2}{x + 4976887516324122696285} \right. ,$$
$$\left. 1119794081769030040965 9\frac{x^2 + 9953775032648245392570x + 8294812527206871160477}{(x + 4976887516324122696285)^2} y \right) .$$

As observed in [9], computing this endomorphism is slightly harder than doubling. The equations of the dual of α are similar.

Example 2. Consider $D = 3$ and an endomorphism with characteristic equation given by $X^2 + 2X + 4 = 0$. We found $\lambda = 2^{40} + 2^{29} + 1$ verifying $\lambda^2 + 2\lambda + 4 = r$ where r is given by $r = 1210106699470122931716103$. We have

$$q = 1264229266808611574080347733550955195230739769633 57.$$

The curve E given by the equation $y^2 = x^3 + 1$ has embedding 2 with respect to r.

Example 3. Consider $D = 4$ and an endomorphism with characteristic equation given by $X^2 + 2X + 2 = 0$. We found $\lambda = 2^{40} + 2^{25} + 1$ verifying $\lambda^2 + 2\lambda + 2 = r$ where r is given by $r = 1208999607721222100484101$. We have

$$q = 1983865249857766464311821377127778049381340635654955 7.$$

The curve E given by the equation $y^2 = x^3 + x$ has embedding 4 with respect to r.