Optimal Eta pairing on supersingular genus-2 binary hyperelliptic curves

Diego Aranha, Jean-Luc Beuchat, Jérémie Detrey, Nicolas Estibals

To cite this version:

Diego Aranha, Jean-Luc Beuchat, Jérémie Detrey, Nicolas Estibals. Optimal Eta pairing on supersingular genus-2 binary hyperelliptic curves. Orr Dunkelman. Cryptographer's Track at the RSA Conference 2012 (CT-RSA 2012), Feb 2012, San Francisco, United States. Springer, pp.19, 2012. <inria-00540002>

HAL Id: inria-00540002

https://hal.inria.fr/inria-00540002

Submitted on 25 Nov 2010

Optimal Eta Pairing on Supersingular Genus-2 Binary Hyperelliptic Curves

Diego F. Aranha[1]*, Jean-Luc Beuchat[2], Jérémie Detrey[3], and Nicolas Estibals[3]

[1] Institute of Computing, University of Campinas
Av. Albert Einstein, 1251, CEP 13084-971, Campinas, Brazil
dfaranha@ic.unicamp.br

[2] Graduate School of Systems and Information Engineering, University of Tsukuba,
1-1-1 Tennodai, Tsukuba, Ibaraki, 305-8573, Japan
beuchat@risk.tsukuba.ac.jp

[3] CARAMEL project-team, LORIA, INRIA / CNRS / Nancy Université,
Campus Scientifique, BP 239, 54506 Vandœuvre-lès-Nancy Cedex, France
jeremie.detrey@loria.fr nicolas.estibals@loria.fr

Abstract. This article presents a novel optimal pairing over supersingular genus-2 binary hyperelliptic curves. Starting from Vercauteren's work on optimal pairings, we describe how to exploit the action of the 2^{3m}-th power Verschiebung in order to further reduce the loop length of Miller's algorithm compared to the genus-2 η_T approach.
As a proof of concept, we detail an optimized software implementation and an FPGA accelerator for computing the proposed optimal Eta pairing on a genus-2 hyperelliptic curve over $\mathbb{F}_{2^{367}}$, which satisfies the recommended security level of 128 bits.

Keywords: Optimal Eta pairing, supersingular genus-2 curve, software implementation, FPGA implementation.

1 Introduction

The Weil and Tate pairings were independently introduced in cryptography by Frey & Rück [17] and Menezes, Okamoto & Vanstone [32] as tools to attack the discrete-logarithm problem on some classes of elliptic curves defined over finite fields. The discovery of constructive properties by Joux [26], Mitsunari, Sakai & Kasahara [35], and Sakai, Oghishi & Kasahara [39] initiated the proposal of an ever-increasing number of protocols based on bilinear pairings: identity-based encryption [9], short signature [11], and efficient broadcast encryption [10], to mention but a few. However, such protocols rely critically on efficient implementations at high levels of security of pairing primitives on a wide range of targets.

Miller described the first iterative algorithm to compute the Weil and Tate pairings back in 1986 [33, 34]. The Tate pairing seems to be more suited to efficient implementations (see for instance [22,28]), and has therefore attracted a lot

* This work was performed while the author was visiting University of Waterloo.

of interest from the research community. A large number of articles, culminating in the η_T pairing algorithm [4], focused on shortening the loop of Miller's algorithm in the case of supersingular abelian varieties. The Ate pairing, introduced by Hess *et al.* [25] for elliptic curves and by Granger *et al.* [21] for hyperelliptic curves, generalises the η_T approach to ordinary curves. Eventually, several variants of the Ate pairing aiming at reducing the loop length of Miller's algorithm have been proposed in 2008 [24, 29, 40].

In this work, we target the AES-128 security level. When dealing with ordinary elliptic curves defined over a prime finite field $\mathbb{F}_p$, the family of curves introduced by Barreto & Naehrig (BN) [5] is a nearly optimal choice for the 128-bit security level. Their embedding degree $k = 12$ perfectly balances the security between the ℓ-torsion and the group of ℓ-th roots of unity, where ℓ is a prime number dividing the cardinal of the curve $\#E(\mathbb{F}_p)$. Naehrig *et al.* [37] have combined a new representation of the elements in the underlying field and SIMD floating-point instructions of the AMD64 architecture to design a fast software library. More recently, Beuchat *et al.* [7] have reported the computation in less than one millisecond on a single core of an Intel Core i7 processor of a bilinear pairing on a BN curve at a level of security roughly equivalent to that of AES-128. Kammler *et al.* [27] have proposed the design-space exploration for an ASIP for the computation of cryptographic pairings over BN curves. They have extended a RISC core for acceleration of arithmetic over $\mathbb{F}_p$. Their processor is very well-suited to embedded systems and needs 15.8 milliseconds to compute a pairing. A more surprising result is that the parallel hardware accelerators described in [16, 20] are slower than an optimized software implementation on a Core i7 processor [7]. The main difficulty is to implement an efficient arithmetic unit over a quite large prime field.

Supersingular curves over $\mathbb{F}_{2^m}$ and $\mathbb{F}_{3^m}$ are better suited to hardware implementation, and offer more efficient point doubling and tripling formulae than BN-curves. However, the embedding degree of a supersingular elliptic curve is always less than or equal to 6 [32]. As a consequence, the security on the curve is too high with respect to the security of the group of ℓ-th roots of unity, and one has to consider curves defined over very large finite fields. Therefore, most of the hardware accelerators are struggling to achieve the AES-128 level of security (see for instance [6] for a comprehensive bibliography). Software implementations at the 128-bit security level have for instance been reported in [3, 8]. However, the computation of a bilinear pairing is at least 6 times faster on a BN curve [7].

To mitigate the effect of the bounded embedded degree, Estibals has proposed to consider supersingular elliptic curves over field extensions of moderately-composite degree [15]. Curves are then vulnerable to Weil descent attacks [19], but a careful analysis has allowed him to maintain the security above the 128-bit threshold. As a proof of concept, he has designed a compact Field-Programmable Gate Array (FPGA) accelerator for computing the Tate pairing on a supersingular elliptic curve defined over $\mathbb{F}_{3^{5 \cdot 97}}$. Even though he targeted his architecture to low-resource hardware, his timings are very close to those of software implementations of BN curves.

Yet another way to better balance security on both the inputs and the output of the Tate pairing in the supersingular case is to consider a genus-2 binary hyperelliptic curve with embedding degree $k = 12$ [18]. To the best of our knowledge, Ronan *et al.* [38] have proposed the first hardware accelerator for the genus-2 η_T pairing devised by Barreto *et al.* in [4]. However, they assume both arguments of the Tate pairing to be degenerate divisors, and their coprocessor reaches only 75 bits of security.

In this work, we show that supersingular genus-2 binary hyperelliptic curves are very effective in the context of software implementations and hardware accelerators for embedded systems. After a general reminder on the hyperelliptic Tate pairing (Section 2) and on the Eta pairing on in the case of those particular curves (Section 3), we describe a novel optimal Eta pairing algorithm that further reduces the loop length of Miller's algorithm compared to the η_T approach [4] (Section 4). We then present an optimized software implementation (Section 5) and a low-area FPGA accelerator (Section 6) of the proposed pairing algorithm. We discuss our results and conclude in Section 7.

2 Background Material and Notations

In this section, we briefly recall a few definitions and results about hyperelliptic curves, and more precisely the Tate pairing on such curves. For more details, we refer the interested reader to [14, 21].

2.1 Reminder on Hyperelliptic Curves

Let C be an imaginary nonsingular hyperelliptic curve of genus g defined over the finite field $\mathbb{F}_q$, where $q = p^m$ and p is a prime, and whose affine part is given by the equation

$$y^2 + h(x)y = f(x),$$

where f, $h \in \mathbb{F}_q[x]$, $\deg f = 2g + 1$, and $\deg h \le g$.

For any algebraic extension $\mathbb{F}_{q^d}$ of $\mathbb{F}_q$, we define the set of $\mathbb{F}_{q^d}$-rational points of C as $C(\mathbb{F}_{q^d}) = \{(x, y) \in \mathbb{F}_{q^d} \times \mathbb{F}_{q^d} \mid y^2 + h(x)y = f(x)\} \cup \{P_\infty\}$, where P_∞ is the point at infinity of the curve. For simplicity's sake, we also write $C = C(\overline{\mathbb{F}}_q)$. Additionally, denoting by ϕ_q the q-th power Frobenius morphism $\phi_q : C \to C$, $(x, y) \mapsto (x^q, y^q)$, and $P_\infty \mapsto P_\infty$, note that a point $P \in C$ is $\mathbb{F}_{q^d}$-rational if and only if $\phi_q^d(P) = P$.

We then denote by Jac_C the Jacobian of C, which is an abelian variety of dimension g defined over $\mathbb{F}_q$, and whose elements are represented by the divisor class group of degree-0 divisors $\mathrm{Pic}^0_C = \mathrm{Div}^0_C \,/\, \mathrm{Princ}_C$. In other words, two degree-0 divisors D and D' belong to the same equivalence class $\overline{D} \in \mathrm{Jac}_C$ if and only if there exists a non-zero rational function $z \in \overline{\mathbb{F}}_q(C)^*$ such that $D' = D + \mathrm{div}(z)$. Naturally extending the Frobenius map to divisors as $\phi_q : \sum_{P \in C} n_P(P) \mapsto \sum_{P \in C} n_P(\phi_q(P))$, we say that D is $\mathbb{F}_{q^d}$-rational if and only if $\phi_q^d(D) = D$.

It can also be shown that any divisor class $\overline{D} \in \mathrm{Jac}_C(\mathbb{F}_{q^d})$ can be uniquely represented by an $\mathbb{F}_{q^d}$-rational reduced divisor $\rho(\overline{D}) = \sum_{i=1}^{r}(P_i) - r(P_\infty)$, with $r \leq g$, $P_i \neq P_\infty$, and $P_i \neq -P_j$ for $i \neq j$, where the negative of a point $P = (x, y)$ is given via the hyperelliptic involution by $-P = (x, -y-h(x))$. In the following, we also denote by $\epsilon(\overline{D}) = \sum_{i=1}^{r}(P_i)$ the effective part of $\rho(\overline{D})$.

Using the Mumford representation, any non-zero $\mathbb{F}_{q^d}$-rational reduced divisor $D = \rho(\overline{D})$ (and therefore any non-zero element of the Jacobian $\mathrm{Jac}_C(\mathbb{F}_{q^d})$) can be associated with a unique pair of polynomials $[u(x), v(x)]$, with u, $v \in \mathbb{F}_{q^d}[x]$ and such that u is monic, $\deg(v) < \deg(u) = r \leq g$, and $u \mid v^2 + vh - f$. Furthermore, given two reduced divisors D_1 and D_2 in Mumford representation, Cantor's algorithm [12] can be used to compute the Mumford representation of $\rho(D_1 + D_2)$, the reduced divisor corresponding to their sum on the Jacobian.

2.2 Hyperelliptic Tate Pairing

Let ℓ be a prime dividing $\#\,\mathrm{Jac}_C(\mathbb{F}_q)$ and coprime to q. Let also k be the corresponding embedding degree, *i.e.*, the smallest integer such that $\ell \mid q^k - 1$. We denote by $\mathrm{Jac}_C(\mathbb{F}_{q^k})[\ell]$ the $\mathbb{F}_{q^k}$-rational ℓ-torsion subgroup of Jac_C. The Tate pairing on C is then the well-defined, non-degenerate, and bilinear map

$$\langle .,. \rangle_\ell : \mathrm{Jac}_C(\mathbb{F}_{q^k})[\ell] \times \mathrm{Jac}_C(\mathbb{F}_{q^k})/\ell\,\mathrm{Jac}_C(\mathbb{F}_{q^k}) \rightarrow \mathbb{F}_{q^k}^*/(\mathbb{F}_{q^k}^*)^\ell,$$

defined as $\langle \overline{D}_1, \overline{D}_2 \rangle_\ell \equiv f_{\ell,D_1}(D_2)$, where D_1 and D_2 represent the divisor classes $\overline{D}_1$ and $\overline{D}_2$, respectively, and such that they have disjoint supports: $\mathrm{supp}(D_1) \cap \mathrm{supp}(D_2) = \emptyset$. Moreover, for any integer n and any $\mathbb{F}_{q^k}$-rational divisor D, the notation $f_{n,D}$ denotes the Miller function in $\mathbb{F}_{q^k}(C)^*$ which is defined (up to a non-zero constant multiple) by its divisor such that $\mathrm{div}(f_{n,D}) = nD - [n]D$, where $[n]D = \rho(nD)$. In the case of the Tate pairing, since $\overline{D}_1 \in \mathrm{Jac}_C[\ell]$, we have $[\ell]D_1 = 0$ and $\mathrm{div}(f_{\ell,D_1}) = \ell D_1$.

So as to obtain a unique value for the Tate pairing, we also define the reduced Tate pairing as $e : (\overline{D}_1, \overline{D}_2) \mapsto \langle \overline{D}_1, \overline{D}_2 \rangle_\ell^{(q^k-1)/\ell} \in \mu_\ell$, with $\mu_\ell \subseteq \mathbb{F}_{q^k}^*$ the subgroup of ℓ-th roots of unity. Note that for any L such that $\ell \mid L \mid q^k - 1$, we also have $e(\overline{D}_1, \overline{D}_2) = \langle \overline{D}_1, \overline{D}_2 \rangle_L^{(q^k-1)/L}$.

Ensuring that there is no element of order ℓ^2 in $\mathrm{Jac}_C(\mathbb{F}_{q^k})$, we can also show that there is a natural isomorphism between the quotient $\mathrm{Jac}_C(\mathbb{F}_{q^k})/\ell\,\mathrm{Jac}_C(\mathbb{F}_{q^k})$ and $\mathrm{Jac}_C(\mathbb{F}_{q^k})[\ell]$. We can then identify these two groups, and define the Tate pairing on the domain $\mathrm{Jac}_C(\mathbb{F}_{q^k})[\ell] \times \mathrm{Jac}_C(\mathbb{F}_{q^k})[\ell]$.

The actual computation of the (reduced) Tate pairing is achieved thanks to Miller's algorithm [33, 34], which is based on the observation that, for any integer n, n', and for any $\mathbb{F}_{q^k}$-rational divisor D, one can take the function $f_{n+n',D} = f_{n,D} \cdot f_{n',D} \cdot g_{[n]D,[n']D}$, where $g_{[n]D,[n']D} \in \mathbb{F}_{q^k}(C)^*$ is such that $\mathrm{div}(g_{[n]D,[n']D}) = [n]D + [n']D - [n+n']D$. Note that the function $g_{[n]D,[n']D}$ can be explicitly obtained from the computation of $[n+n']D = \rho([n]D + [n']D)$ by Cantor's algorithm. See for instance [21, Algorithm 2] for more details. Therefore, computing $f_{\ell,D_1}(D_2)$ is tantamount to computing $[\ell]D_1$ on $\mathrm{Jac}_C(\mathbb{F}_{q^k})$ by means of any suitable scalar multiplication algorithm (*e.g.*, addition chain or double-and-add) while keeping track of the $g_{[n]D_1,[n']D_1}$ functions given by Cantor's algorithm and evaluating them at the divisor D_2. Miller's algorithm, based on the double-and-add approach, thus has a complexity of $\lfloor \log_2(\ell) \rfloor + \mathrm{wg}(\ell) - 1$ iterations (*i.e.*, evaluations of such $g_{[n]D_1,[n']D_1}$ functions), where $\mathrm{wg}(\ell)$ denotes the Hamming weight of ℓ.

Finally, let u_∞ be an $\mathbb{F}_q$-rational uniformizer at P_∞ (*i.e.*, $\mathrm{ord}_{P_\infty}(u_\infty) = 1$). For any function $z \in \overline{\mathbb{F}}_q(C)^*$, we denote by $\mathrm{lc}_\infty(z) = (u_\infty^{-\mathrm{ord}_{P_\infty}(z)} \cdot z)(P_\infty)$ the leading coefficient of z expressed as a Laurent series in u_∞. Restricting the domain of the Tate pairing to $\overline{D}_1 \in \mathrm{Jac}_C(\mathbb{F}_q)[\ell]$, one can easily check that $\mathrm{lc}_\infty(f_{\ell,D_1}) \in \mathbb{F}_q^*$ with $D_1 = \rho(\overline{D}_1)$. We can then apply [21, Lemma 1] to show that we can simply compute the Tate pairing as $\langle \overline{D}_1, \overline{D}_2 \rangle_\ell = f_{\ell,D_1}(\epsilon(\overline{D}_2))$, as long as $\mathrm{supp}(D_1) \cap \mathrm{supp}(\epsilon(\overline{D}_2)) = \emptyset$. This last condition is ensured by taking $\overline{D}_2 \in \mathrm{Jac}_C(\mathbb{F}_{q^k})[\ell] \setminus \mathrm{Jac}_C(\mathbb{F}_q)[\ell]$.

3 Eta Pairing on Supersingular Genus-2 Binary Curves

3.1 Curve Definition and Basic Properties

In this work, we consider the family of supersingular genus-2 hyperelliptic curves defined over $\mathbb{F}_2$ by the following equation:

$$C_d : y^2 + y = x^5 + x^3 + d,$$

where $d \in \mathbb{F}_2$. Because of their supersingularity, which provides them with a very efficient arithmetic, along with their embedding degree of 12, which is the highest among all supersingular genus-2 curves, these curves are a target of choice for implementing pairing-based cryptography. They have therefore already been studied in this context in several articles [4, 13, 18, 30, 38].

For m a positive integer coprime to 6, the cardinality of the Jacobian of C_d over $\mathbb{F}_{2^m}$, denoted by L, is given by

$$L = \#\,\mathrm{Jac}_{C_d}(\mathbb{F}_{2^m}) = 2^{2m} + \delta 2^{(3m+1)/2} + 2^m + \delta 2^{(m+1)/2} + 1,$$

where the value of δ is

$$\delta = \begin{cases} (-1)^d & \text{when } m \equiv 1,\ \ 7,\ 17, \text{ or } 23 \pmod{24}, \text{ and} \\ -(-1)^d & \text{when } m \equiv 5,\ 11,\ 13, \text{ or } 19 \pmod{24}. \end{cases}$$

The embedding degree of C_d is $k = 12$, and $\#\,\mathrm{Jac}_{C_d}(\mathbb{F}_{2^m}) \mid 2^{12m} - 1$. The Tate pairing and its variants will then map into the degree-12 extension $\mathbb{F}_{2^{12m}}$, which we represent as the tower field $\mathbb{F}_{2^{12m}} \cong \mathbb{F}_{2^m}[w, s_0]$ where $w \in \mathbb{F}_{2^6}$ is such that $w^6 + w^5 + w^3 + w^2 + 1 = 0$, and $s_0 \in \mathbb{F}_{2^{12}}$ is such that $s_0^2 + s_0 + w^5 + w^3 = 0$.

Furthermore, since C_d is supersingular, it has non-trivial distortion maps embedding $\mathrm{Jac}_{C_d}(\mathbb{F}_{2^m})$ into distinct subgroups of $\mathrm{Jac}_{C_d}(\mathbb{F}_{2^{12m}})$. In this work, we consider the distortion map ψ which acts on the curve as

$$\begin{aligned} \psi : \quad & C_d \to C_d \\ & (x, y) \mapsto (x + w, y + s_2 x^2 + s_1 x + s_0), \end{aligned}$$

with $s_1 = w^4 + w^2$ and $s_2 = w^4 + 1$. The action of ψ is naturally extended to the Jacobian in the following way:

$$\begin{array}{rcl} \psi : & \mathrm{Jac}_{C_d} \rightarrow \mathrm{Jac}_{C_d} \\ & \sum_{i=1}^{r}(P_i) - r(P_\infty) \mapsto \sum_{i=1}^{r}(\psi(P_i)) - r(P_\infty). \end{array}$$

3.2 Modified Tate Pairing on C_d

Let ℓ be a large (odd) prime dividing $L = \#\,\mathrm{Jac}_{C_d}(\mathbb{F}_{2^m})$. After ensuring that there are no points of order ℓ^2 in $\mathrm{Jac}_{C_d}(\mathbb{F}_{2^{12m}})$, we can restrict the domain of the Tate pairing to $\mathrm{Jac}_{C_d}(\mathbb{F}_{2^m})[\ell] \times \mathrm{Jac}_{C_d}(\mathbb{F}_{2^{12m}})[\ell]$, as detailed in Section 2.2. Using the distortion map ψ which maps $\mathrm{Jac}_{C_d}(\mathbb{F}_{2^m})[\ell]$ to a subgroup $\psi(\mathrm{Jac}_{C_d}(\mathbb{F}_{2^m})[\ell]) \subset \mathrm{Jac}_{C_d}(\mathbb{F}_{2^{12m}})[\ell]$ such that $\mathrm{Jac}_{C_d}(\mathbb{F}_{2^m})[\ell] \cap \psi(\mathrm{Jac}_{C_d}(\mathbb{F}_{2^m})[\ell]) = \{0\}$, we can then define the reduced modified Tate pairing as the non-degenerate, bilinear map

$$\begin{array}{rcccl} \hat{e} & : & \mathrm{Jac}_{C_d}(\mathbb{F}_{2^m})[\ell] \times \mathrm{Jac}_{C_d}(\mathbb{F}_{2^m})[\ell] & \longrightarrow & \mu_\ell \subseteq \mathbb{F}_{2^{12m}}^* \\ & & (\quad \overline{D}_1 \quad , \quad \overline{D}_2 \quad) & \longmapsto & \langle \overline{D}_1, \psi(\overline{D}_2)\rangle_\ell^{(2^{12m}-1)/\ell} \\ & & & & = \langle \overline{D}_1, \psi(\overline{D}_2)\rangle_L^{(2^{12m}-1)/L}, \end{array}$$

where $\langle \overline{D}_1, \psi(\overline{D}_2)\rangle_L = f_{L,D_1}(\epsilon(\psi(D_2)))$, the divisor classes $\overline{D}_1$ and $\overline{D}_2$ being represented by the $\mathbb{F}_{2^m}$-rational reduced divisors $D_1 = \rho(\overline{D}_1)$ and $D_2 = \rho(\overline{D}_2)$. As long as $\overline{D}_1$ and $\overline{D}_2$ are not both trivial, the distortion map ψ ensures that the affine supports of D_1 and $\psi(D_2)$ are disjoint.

At this stage, we have to point out that, in this case, the $g_{[n]D_1,[n']D_1}$ functions required by Miller's algorithm in the computation of the Tate pairing can be simplified. Indeed, from Cantor's algorithm, most of these functions involve vertical lines, which all pass through multiples of the $\mathbb{F}_{2^m}$-rational reduced divisor D_1, meaning that their equations will also be $\mathbb{F}_{2^m}$-rational. Furthermore, noticing that the x-coordinate of $\psi(P)$ is in $\mathbb{F}_{2^{6m}}$ when P is $\mathbb{F}_{2^m}$- or $\mathbb{F}_{2^{2m}}$-rational, we can conclude that the evaluation of those vertical lines at $\epsilon(\psi(D_2))$ for any $\mathbb{F}_{2^m}$-rational reduced divisor D_2 will also be in $\mathbb{F}_{2^{6m}}^*$ and therefore annihilated by the final exponentiation to the $(2^{12m}-1)/L$-th power. We can then safely ignore the computation of those vertical lines.

3.3 Octupling and Action of $\hat{\phi}_{2^{3m}}$

Even though C_d is supersingular, it is not superspecial. Therefore, one cannot directly exploit the action of the Verschiebung $\hat{\phi}_{2^m}$, dual of the 2^m-th power Frobenius map, in order to benefit from the Miller's loop reduction of the Eta (or superspecial Ate) pairing [21]. However, as already noted by Barreto *et al.* in [4], the 2^{3m}-th power Verschiebung $\hat{\phi}_{2^{3m}}$, which appears in the octupling formulae of Jac_{C_d}, can be used instead. We detail this construction in the following paragraphs.

Let $P = (x_P, y_P)$ be a point of C_d distinct from P_∞, and $D = (P) - (P_\infty)$ be the corresponding degenerate divisor. Its Mumford representation is then $D = [x + x_P, y_P]$. Doubling and reducing D three times via Cantor's algorithm, we obtain $[8]D = \rho(8D) = [x + x_P^{64} + 1, x_P^{128} + y_P^{64} + 1]$. Note that the divisor $[8]D$ is also degenerate, as $[8]D = ([8]P) - (P_\infty)$, and corresponds to the point $[8]P = (x_P^{64} + 1, x_P^{128} + y_P^{64} + 1) \in C_d$.

Octupling therefore acts not only on Jac_{C_d} but also on the curve C_d itself, and is actually an automorphism of C_d defined over $\mathbb{F}_2$ as $[8] = \sigma \circ \phi_8^2$ with $\sigma : (x, y) \mapsto (x+1, x^2+y+1)$ and ϕ_8 the 8th power Frobenius map $(x, y) \mapsto (x^8, y^8)$.

Iterating this octupling m times, we obtain the $\mathbb{F}_2$-rational automorphism $[2^{3m}]$ on C_d defined as $[2^{3m}] = \gamma \circ \phi_{2^{3m}}^2$, with $\gamma = \sigma^m : (x, y) \mapsto (x+1, x^2+y+\nu)$ and $\nu = (m+1)/2 \bmod 2$. Note that γ, $\phi_{2^{3m}}$, and $[2^{3m}]$ can be naturally extended to Jac_{C_d}, where the latter corresponds to the multiplication by 2^{3m}.

Furthermore, since the Frobenius map $\phi_{2^{3m}}$ is a degree-2^{3m} isogeny of Jac_{C_d}, we know that $\phi_{2^{3m}} \circ \hat{\phi}_{2^{3m}} = \hat{\phi}_{2^{3m}} \circ \phi_{2^{3m}} = [2^{3m}]$, where $\hat{\phi}_{2^{3m}}$ denotes the Verschiebung, *i.e.* the dual isogeny of $\phi_{2^{3m}}$. Having written $[2^{3m}] = \gamma \circ \phi_{2^{3m}}^2$, we can then identify this Verschiebung with $\hat{\phi}_{2^{3m}} = \gamma \circ \phi_{2^{3m}}$ and thus verify that $\hat{\phi}_{2^{3m}}$ is also a degree-2^{3m} purely inseparable automorphism of the curve C_d. We are therefore in the conditions of [21, Lemma 5], from which we get that, for any reduced divisor D, $\hat{\phi}_{2^{3m}}(D)$ is also reduced and we have the equality of Miller functions (up to a non-zero constant multiple)

$$f_{n,\hat{\phi}_{2^{3m}}(D)} \circ \hat{\phi}_{2^{3m}} = f_{n,D}^{2^{3m}}. \tag{1}$$

We now focus on the action of $\hat{\phi}_{2^{3m}}$ on $\mathrm{Jac}_{C_d}(\mathbb{F}_{2^m})$ and its image through the distortion map $\psi(\mathrm{Jac}_{C_d}(\mathbb{F}_{2^m}))$. First of all, since $\phi_{2^{3m}}$ is the identity over $\mathrm{Jac}_{C_d}(\mathbb{F}_{2^m})$, we have that, for any $\mathbb{F}_{2^m}$-rational reduced divisor D, $\hat{\phi}_{2^{3m}}(D) = (\hat{\phi}_{2^{3m}} \circ \phi_{2^{3m}})(D) = [2^{3m}]D$.

Let us now consider the map $\hat{\phi}_{2^{3m}} \circ \psi$ over C_d. We have

$$\hat{\phi}_{2^{3m}} \circ \psi = \gamma \circ \phi_{2^{3m}} \circ \psi = \gamma \circ \psi^{(2^{3m})} \circ \phi_{2^{3m}},$$

where $\psi^{(2^{3m})}$ is obtained by raising the coefficients of ψ to the 2^{3m}-th power:

$$\begin{aligned} \psi^{(2^{3m})}(x, y) &= (x + w^{2^{3m}}, y + s_2^{2^{3m}} x^2 + s_1^{2^{3m}} x + s_0^{2^{3m}}) \\ &= (x + w + 1, y + (s_2 + 1)x^2 + s_1 x + (s_0 + w^2 + \nu + 1)). \end{aligned}$$

As shown in [4, Lemma 9], it then follows that

$$\left(\gamma \circ \psi^{(2^{3m})}\right)(x, y) = (x + w, y + s_2 x^2 + s_1 x + s_0) = \psi(x, y),$$

from which we conclude that $\hat{\phi}_{2^{3m}} \circ \psi = \psi \circ \phi_{2^{3m}}$ on C_d, and on Jac_{C_d} via natural extension. This in turn shows that $\hat{\phi}_{2^{3m}}(\psi(D)) = \psi(D)$ for any $\mathbb{F}_{2^m}$-rational reduced divisor D or, in other words, that $\hat{\phi}_{2^{3m}}$ acts as the identity over $\psi(\mathrm{Jac}_{C_d}(\mathbb{F}_{2^m}))$.

3.4 Eta Pairing on C_d

We now follow the construction of [4] in order to obtain the η_T pairing with $T = 2^{3m}$. Remarking indeed that $\ell \mid L \mid N$ for $N = 2^{12m} - 1 = T^4 - 1$, and taking $M = N/L$, we can write

$$\hat{e}(\overline{D}_1, \overline{D}_2)^M = f_{L,D_1}(\epsilon(\psi(D_2)))^{M(2^{12m}-1)/L} = f_{N,D_1}(\epsilon(\psi(D_2)))^{(2^{12m}-1)/L}.$$

As $\ell \mid N$, we can then take the Miller function

$$f_{N,D_1} = f_{N+1,D_1} = f_{T^4,D_1} = \prod_{i=0}^{3} f_{T,[T^i]D_1}^{T^{3-i}} = \prod_{i=0}^{3} f_{2^{3m},[2^{i\cdot 3m}]D_1}^{2^{(3-i)\cdot 3m}}.$$

Furthermore, since D_1 and D_2 are $\mathbb{F}_{2^m}$-rational reduced divisors, we also have that $[2^{i\cdot 3m}]D_1 = \hat{\phi}_{2^{3m}}^{i}(D_1)$ and $\epsilon(\psi(D_2)) = \hat{\phi}_{2^{3m}}^{i}(\epsilon(\psi(D_2)))$ for all i. Iterating (1) then yields

$$\begin{aligned} f_{2^{3m},[2^{i\cdot 3m}]D_1}(\epsilon(\psi(D_2))) &= \left(f_{2^{3m},\hat{\phi}_{2^{3m}}^{i}(D_1)} \circ \hat{\phi}_{2^{3m}}^{i}\right)(\epsilon(\psi(D_2))) \\ &= f_{2^{3m},D_1}(\epsilon(\psi(D_2)))^{2^{i\cdot 3m}}. \end{aligned}$$

Putting it all together, we finally obtain

$$\hat{e}(\overline{D}_1, \overline{D}_2)^M = f_{2^{3m},D_1}(\epsilon(\psi(D_2)))^{4\cdot 2^{3\cdot 3m}\cdot(2^{12m}-1)/L},$$

and, as $\ell \nmid 4 \cdot 2^{3\cdot 3m}$,

$$f_{2^{3m},D_1}(\epsilon(\psi(D_2)))^{(2^{12m}-1)/L} = \hat{e}(\overline{D}_1, \overline{D}_2)^{M\cdot(4\cdot 2^{3\cdot 3m})^{-1} \bmod L}.$$

From the bilinearity and the non-degeneracy of the Tate pairing, we can then conclude that the η_T pairing defined as follows is also bilinear and non-degenerate [4]:

$$\begin{array}{rccccccl} \eta_T & : & \mathrm{Jac}_{C_d}(\mathbb{F}_{2^m})[\ell] & \times & \mathrm{Jac}_{C_d}(\mathbb{F}_{2^m})[\ell] & & \longrightarrow & \mu_\ell \subseteq \mathbb{F}_{2^{12m}}^{*} \\ & & (\quad \overline{D}_1 & , & \overline{D}_2 \quad &) & \longmapsto & f_{2^{3m},D_1}(\epsilon(\psi(D_2)))^{(2^{12m}-1)/L}. \end{array}$$

4 Optimal Eta Pairing on C_d

4.1 Construction and Definition

In order to further decrease the loop length in Miller's algorithm, we adapt in this work the optimal pairing technique as introduced by Vercauteren [40] to the case of the action of the 2^{3m}-th power Verschiebung $\hat{\phi}_{2^{3m}}$ and the Eta pairing detailed in the previous section.

To that intent, let us consider the 2-dimensional lattice spanned by the rows of the matrix

$$\mathfrak{L} = \begin{pmatrix} L & 0 \\ -2^{3m} & 1 \end{pmatrix}.$$

Note that since $\ell \mid L \mid 2^{6m} + 1$, we know that $2^{6m} \equiv -1 \pmod{\ell}$, meaning that there is no need to look for 2^{3m}-ary expansions of multiples of L having more than two digits.

A shortest vector of $\mathfrak{L}$ is $[c_0, c_1] = [\delta 2^{(m-1)/2} + 1, 2^m + \delta 2^{(m-1)/2}]$, which corresponds to taking the multiple $N' = c_1 2^{3m} + c_0 = M'L$ with $M' = 2^{2m} - \delta 2^{(3m-1)/2} - \delta 2^{(m-1)/2} + 1$.

We then have the M'-th power of the reduced modified Tate pairing

$$\hat{e}(\overline{D}_1, \overline{D}_2)^{M'} = f_{N',D_1}(\epsilon(\psi(D_2)))^{(2^{12m}-1)/L},$$

for which we can take the Miller function

$$\begin{aligned} f_{N',D_1} &= f_{c_1 2^{3m},D_1} \cdot f_{c_0,D_1} \cdot g_{[c_0]D_1,[c_1 2^{3m}]D_1} \\ &= f^{c_1}_{2^{3m},D_1} \cdot f_{c_1,[2^{3m}]D_1} \cdot f_{c_0,D_1} \cdot g_{[c_0]D_1,[c_1 2^{3m}]D_1}. \end{aligned}$$

Remarking that $c_1 2^{3m} \equiv -c_0 \pmod{\ell}$, $g_{[c_0]D_1,[c_1 2^{3m}]D_1}$ actually corresponds to the vertical lines passing through $[c_0]D_1$ and $[-c_0]D_1$, which can simply be ignored. Furthermore, exploiting the action of the Verschiebung $\hat{\phi}_{2^{3m}}$, we can rewrite $f_{c_1,[2^{3m}]D_1}(\epsilon(\psi(D_2)))$ as $f^{2^{3m}}_{c_1,D_1}(\epsilon(\psi(D_2)))$. Finally, also note that $f_{2^{3m},D_1}(\epsilon(\psi(D_2)))^{c_1 \cdot (2^{12m}-1)/L}$ is actually a power of the Eta pairing $\eta_T(\overline{D}_1, \overline{D}_2)$ defined in the previous section.

Consequently, let $\eta_{[c_0,c_1]} : \mathrm{Jac}_{C_d}(\mathbb{F}_{2^m})[\ell] \times \mathrm{Jac}_{C_d}(\mathbb{F}_{2^m})[\ell] \to \mu_\ell$ be the optimal Eta pairing defined as

$$\eta_{[c_0,c_1]} \quad : \quad (\overline{D}_1, \overline{D}_2) \quad \longmapsto \quad \left(f^{2^{3m}}_{c_1,D_1} \cdot f_{c_0,D_1}\right)(\epsilon(\psi(D_2)))^{(2^{12m}-1)/L}.$$

From the previous considerations, we thus have that

$$\hat{e}(\overline{D}_1, \overline{D}_2)^{M'} = \eta_{[c_0,c_1]}(\overline{D}_1, \overline{D}_2) \cdot \eta_T(\overline{D}_1, \overline{D}_2)^{c_1},$$

whence $\eta_{[c_0,c_1]}(\overline{D}_1, \overline{D}_2) = \hat{e}(\overline{D}_1, \overline{D}_2)^W$ with

$$\begin{aligned} W &= M' - c_1 M \cdot (4 \cdot 2^{3 \cdot 3m})^{-1} \bmod L \\ &= 2^{2m} + \delta 2^{(3m-1)/2} + 2^m + \delta 2^{(m-1)/2} + 1. \end{aligned}$$

Finally, as $\ell \nmid W$, we show that the optimal Eta pairing $\eta_{[c_0,c_1]}$ is also bilinear and non-degenerate.

Note that the η_T pairing introduced in [4] with $T = -\delta 2^{(3m+1)/2} - 1$ corresponds to the lattice vector $[-\delta 2^{(3m+1)/2} - 1, -1] \in \mathfrak{L}$.

4.2 Computing $\boldsymbol{\eta_{[c_0,c_1]}}$

The computation of the optimal Eta pairing $\eta_{[c_0,c_1]}$ defined in the previous section relies on the evaluation of the two Miller functions f_{c_0,D_1} and f_{c_1,D_1} at $\epsilon(\psi(D_2))$. With $[c_0, c_1] = [\delta 2^{(m-1)/2} + 1, 2^m + \delta 2^{(m-1)/2}]$, we can take the following functions

$$\begin{cases} f_{c_0,D_1} = f_{\delta 2^{(m-1)/2},D_1} \cdot g_{[\delta 2^{(m-1)/2}]D_1,D_1} \quad \text{and} \\ f_{c_1,D_1} = f_{2^m,D_1} \cdot f_{\delta 2^{(m-1)/2},D_1} \cdot g_{[2^m]D_1,[\delta 2^{(m-1)/2}]D_1}. \end{cases}$$

Since we are ignoring the vertical lines, we can further rewrite

$$\begin{aligned} f_{\delta 2^{(m-1)/2},D_1} &= f_{2^{(m-1)/2},[\delta]D_1} \quad \text{and} \\ f_{2^m,D_1} &= f_{\delta 2^{(m-1)/2}\cdot\delta 2^{(m+1)/2},D_1} = f_{2^{(m-1)/2},[\delta]D_1}^{\delta 2^{(m+1)/2}} \cdot f_{2^{(m+1)/2},[2^{(m-1)/2}]D_1}, \end{aligned}$$

which finally gives

$$\begin{cases} f_{c_0,D_1} = f_{2^{(m-1)/2},[\delta]D_1} \cdot g_{[\delta 2^{(m-1)/2}]D_1,D_1} \quad \text{and} \\ f_{c_1,D_1} = f_{2^{(m-1)/2},[\delta]D_1}^{\delta 2^{(m+1)/2}+1} \cdot f_{2^{(m+1)/2},[2^{(m-1)/2}]D_1} \cdot g_{[2^m]D_1,[\delta 2^{(m-1)/2}]D_1}. \end{cases}$$

The computation of $\eta_{[c_0,c_1]}$ therefore chiefly involves the evaluation of the two Miller functions $f_{2^{(m-1)/2},[\delta]D_1}$ and $f_{2^{(m+1)/2},[2^{(m-1)/2}]D_1}$ of loop length $(m-1)/2$ and $(m+1)/2$, respectively. This represents a saving of 33% with respect to the η_T pairing presented in [4] whose Miller's loop length is $(3m+1)/2$.

Note that in order to exploit the octupling formula, we have to consider two cases, depending on the value of $m \bmod 6$.

- When $m \equiv 1 \pmod 6$, then $(m-1)/2$ is a multiple of 3, and $f_{2^{(m-1)/2},[\delta]D_1}$ can be computed via $(m-1)/6$ octuplings, whereas $f_{2^{(m+1)/2},[2^{(m-1)/2}]D_1}$ can be computed by means of another $(m-1)/6$ octuplings and one extra doubling, as per Algorithm 1 in Appendix A.
- When $m \equiv 5 \pmod 6$, $(m-1)/2$ is not a multiple of 3, but $(m+1)/2$ is. We then compute $\eta_{[c_0,c_1]}^2 = \eta_{[2c_0,2c_1]}$ instead, with the Miller functions

  $$\begin{cases} f_{2c_0,D_1} = f_{2^{(m+1)/2},[\delta]D_1} \cdot f_{2,D_1} \cdot g_{[\delta 2^{(m+1)/2}]D_1,[2]D_1} \quad \text{and} \\ f_{2c_1,D_1} = f_{2^{(m+1)/2},[\delta]D_1}^{\delta 2^{(m+1)/2}+1} \cdot f_{2^{(m+1)/2},[2^{(m+1)/2}]D_1} \cdot g_{[2^{m+1}]D_1,[\delta 2^{(m+1)/2}]D_1}. \end{cases}$$

  The two $f_{2^{(m+1)/2},D}$ functions are then evaluated using $(m+1)/6$ octuplings each, whereas f_{2,D_1} only require one doubling, as per Algorithm 2 in Appendix A.

4.3 Evaluation of the Complexity

From the above description of the optimal Eta pairing $\eta_{[c_0,c_1]}$, we can see that most of its computational cost lies in the iterated octuplings of D_1 and the evaluation of the corresponding Miller functions of the form $f_{8,[\pm 8^i]D_1}$ at the effective divisor $\epsilon(\psi(D_2))$. Here, we denote by $[\pm 8^i]D_1$ a reduced divisor representing one of the iterated octuples of D_1 or of $[\delta]D_1$ as required in the evaluation of $\eta_{[c_0,c_1]}$.

In that sense, since D_1 is defined over $\mathbb{F}_{2^m}$, then $[\pm 8^i]D_1$ is also $\mathbb{F}_{2^m}$-rational. Moreover, as octupling directly acts on the curve C_d, if D_1 is degenerate (*i.e.*, of the form $D_1 = (P) - (P_\infty)$), then so is $[\pm 8^i]D_1$. Finally, note that if D_2 is degenerate, then so is $\psi(D_2)$, meaning that $\epsilon(\psi(D_2))$ is of degree 1 and has only one point in its support.

Considering the Miller function for octupling, we rewrite $f_{8,D} = f_{4,D}^2 \cdot f_{2,[4]D}$. Each iteration of Miller's algorithm is then just a matter of evaluating $f_{4,[\pm 8^i]D_1}$ and $f_{2,[\pm 4\cdot 8^i]D_1}$ at $\epsilon(\psi(D_2))$, squaring the former, and accumulating both into the running product via two successive multiplications over $\mathbb{F}_{2^{12m}}$. The respective costs of these operations are given in terms of basic operations over the base field $\mathbb{F}_{2^m}$ in Table 1. Where relevant, several costs are given, depending on whether D_1 and D_2 are general (Gen.) or degenerate (Deg.) divisors.

Note that to obtain these costs, we have constructed $\mathbb{F}_{2^{12m}}$ as the tower field $\mathbb{F}_{2^m}[i, w, s_0]$, where $i \in \mathbb{F}_{2^2}$ is such that $i^2 + i + 1 = 0$, $w \in \mathbb{F}_{2^6}$ is such that $w^3 + iw^2 + iw + i = 0$ (one can then check that we still have $w^6 + w^5 + w^3 + w^2 + 1 = 0$), and s_0 is defined as before. Using the Karatsuba method for the two quadratic extensions and the Toom–Cook method for the cubic one, we obtain the expected complexity of 45 multiplications over $\mathbb{F}_{2^m}$ for computing one product over $\mathbb{F}_{2^{12m}}$ [28].

Table 1. Costs of various operations involved in the computation of the optimal Eta pairing in terms of basic operations (multiplication, addition, squaring, and inversion) over the base field $\mathbb{F}_{2^m}$.

Operation	$\boldsymbol{D_1}$	$\boldsymbol{D_2}$	**Operations over $\mathbb{F}_{2^m}$**			
			Mult.	**Add.**	**Sq.**	**Inv.**
Addition over $\mathbb{F}_{2^{12m}}$	—	—	0	12	0	0
Squaring over $\mathbb{F}_{2^{12m}}$	—	—	0	21	12	0
Multiplication over $\mathbb{F}_{2^{12m}}$	—	—	45	199	0	0
$[\pm 8^i]D_1 \mapsto [\pm 8^{i+1}]D_1$	Deg.	—	0	2	13	0
	Gen.	—	0	5	24	0
$f_{4,[\pm 8^i]D_1}(\epsilon(\psi(D_2)))$	Deg.	Deg.	3	11	1	0
	Gen.	Deg.	19	40	2	0
	Gen.	Gen.	83	247	17	0
$f_{2,[\pm 4\cdot 8^i]D_1}(\epsilon(\psi(D_2)))$	Deg.	Deg.	2	9	1	0
	Gen.	Deg.	16	34	2	0
	Gen.	Gen.	81	236	17	0
Optimal Eta pairing $\eta_{[c_0,c_1]}(\overline{D}_1, \overline{D}_2)$ over $C_0(\mathbb{F}_{2^{367}})$	Deg.	Deg.	7 894	40 356	11 571	1
	Gen.	Deg.	15 293	64 644	15 472	1
	Gen.	Gen.	31 644	118 382	19 161	1

In the two following sections, as a proof of concept, we detail the software and hardware implementation results of the proposed optimal Eta pairing $\eta_{[c_0,c_1]}$. The selected curve is C_0 (*i.e.*, $d = 0$) over the field $\mathbb{F}_{2^{367}}$. One can check that $\#\,\mathrm{Jac}_{C_0}(\mathbb{F}_{2^{367}}) = 13 \cdot 7170258097 \cdot \ell$, where ℓ is a 698-bit prime, whereas the finite field $\mathbb{F}_{2^{12\cdot 367}}$ ensures a security of 128 bits for the computation of discrete logarithms via the function field sieve. The costs of the optimal Eta pairing on $C_0(\mathbb{F}_{2^{367}})$ are also given in Table 1.

5 Software Implementation

A software implementation was realized to illustrate the performance of the proposed pairing. The C programming language was used in conjunction with compiler intrinsics for accessing vector instructions. The chosen compiler was GCC version 4.5.1 so the new instruction for carry-less multiplication [23] was properly supported. Compiler flags included optimization level `-O3`, together with loop unrolling and platform-dependent tuning with `-march=native`. For evaluation, we considered as target platforms the Core 2 Duo 45 nm (Penryn microarchitecture) and Core i5 32 nm (Nehalem microarchitecture), represented by an Intel Xeon X3320 2.5 GHz and a mobile Core i5 540 2.53 GHz, respectively. Field arithmetic was implemented following the vectorization-friendly formulation presented in [2], with the exception of the Core i5 architecture, where multiplication in $\mathbb{F}_{2^{367}}$ was implemented with the help of the native binary field multiplier inside a 6-way Karatsuba formula [36]. Table 2 presents our timings in cycles for finite field arithmetic. Note the significant performance improvement of 60% in multiplication when there is support for the carry-less multiplier.

Table 2. Timings for our software implementations of finite field arithmetic in $\mathbb{F}_{2^{367}}$.

Platform	**Operation cost (cycles)**			
	Add.	**Sq.**	**Mult.**	**Inv.**
Intel Core 2 Duo	7	44	511	19109
Intel Core i5	7	37	208	16547

Table 3 presents our timings in millions of cycles for the pairing computation at the 128-bit security level. Timings from several related works are also collected for direct comparison with our software implementation. Our implementation considers all the three possible choices of divisors (general × general, general × degenerate and degenerate × degenerate). For a fair comparison and completeness, the standard genus-1 η_T pairing was also implemented over $E(\mathbb{F}_{2^{1223}})$ using the native multiplier in the Nehalem microarchitecture. Our implementation of the proposed genus-2 Optimal Eta pairing presents itself as a very efficient candidate among the Type-1 pairings defined on supersingular curves over small-characteristic fields. In particular, the proposed pairing is more efficient than all other Type-1 pairings when at least one of the arguments is a degenerate divisor. Considering the latest Nehalem microarchitecture as an example of a future trend for 64-bit computing platforms, the proposed pairing computed with degenerate divisors is also the closest in terms of performance to the current speed record for Type-3 pairing computation [1].

6 FPGA Implementation

We detail here an FPGA accelerator for the propose hyperelliptic optimal Eta pairing on the curve $C_0(\mathbb{F}_{2^{367}})$ when both inputs are general divisors. Beuchat *et al.* [6] have proposed a coprocessor architecture for computing the final exponentiation of the η_T pairing over supersingular curves in characteristics two and three. The core of their arithmetic and logic unit is a parallel–serial multiplier

Table 3. Software implementations of pairing at the 128-bit security level.

Implementation	**Curve**	**Pairing**	**Intel Core 2 ($\times 10^6$ cycles)**	**Intel Core Nehalem ($\times 10^6$ cycles)**
Beuchat *et al.* [8]	$E(\mathbb{F}_{2^{1223}})$	η_T	23.03	—
	$E(\mathbb{F}_{3^{359}})$		15.13	—
Aranha *et al.* [3]	$E(\mathbb{F}_{2^{1223}})$	η_T	18.76	—
Chatterjee *et al.* [13]	$E(\mathbb{F}_{2^{1223}})$	η_T	19.0	—
	$E(\mathbb{F}_{3^{359}})$		15.8	—
	$C_0(\mathbb{F}_{2^{439}})$		16.4	—
Naehrig *et al.* [37]	$E(\mathbb{F}_p)$	Opt. Ate	4.38	—
Beuchat *et al.* [7]	$E(\mathbb{F}_p)$	Opt. Ate	2.95	2.33
Aranha *et al.* [1]	$E(\mathbb{F}_p)$	Opt. Ate	2.21	1.70
This work	$\boldsymbol{E(\mathbb{F}_{2^{1223}})}$	$\boldsymbol{\eta_T}$	—	**7.50**
This work – Degenerate	$\boldsymbol{C_0(\mathbb{F}_{2^{367}})}$	**Opt. Eta**	**4.96**	**2.50**
This work – Mixed			**9.25**	**4.47**
This work – General			**18.4**	**8.63**

processing D coefficients of an operand at each clock cycle, along with a unified operator supporting addition, Frobenius map, and n-fold Frobenius map. Intermediate results are stored in a register file implemented by means of dual-ported RAM (*cf.* Appendix B for the details of the architecture). As illustrated by Estibals [15], this streamlined design also allows one to design a low-area yet performant FPGA accelerator for the Tate pairing over supersingular elliptic curves. For these reasons, we decided to adapt such a finite field coprocessor for implementing our optimal Eta pairing. In the case of the finite field $\mathbb{F}_{2^{367}}$, we selected the parameters $D = 16$ and $n = 3$ for this coprocessor. We captured our architectures in the VHDL language and prototyped our design on Xilinx Virtex-II Pro, Virtex-4, and Spartan-3 FPGAs with average speedgrade (Table 4). Place-and-route results show for instance that our pairing accelerator uses 4518 slices and 20 RAM blocks of a Virtex-4 device clocked at 220 MHz.

Table 4. FPGA implementations of pairings at medium- and high-security levels.

	Curve	**Sec. (bits)**	**FPGA**	**Area (slices)**	**Freq. (MHz)**	**Time (μs)**	**Area×time (slices.s)**
Ronan *et al.* [38]	$C_0(\mathbb{F}_{2^{103}})$	75	xc2vp100-6	30464	41	132	4.02
Beuchat *et al.* [6]	$E(\mathbb{F}_{2^{691}})$	105	xc4vlx200-11	78874	130	19	1.48
	$E(\mathbb{F}_{3^{313}})$	109	xc4vlx200-11	97105	159	17	1.64
Ghosh *et al.* [20]	$E(\mathbb{F}_{p_{256}})$	128	xc4vlx200-12	52000	50	16400	852.8
Estibals [15]	$E(\mathbb{F}_{3^{5\cdot 97}})$	128	xc4vlx25-11	4755	192	2227	10.59
			xc3s1000-5	4713	104	4113	19.38
This work	$\boldsymbol{C_0(\mathbb{F}_{2^{367}})}$	**128**	**xc2vp30-6**	**4646**	**176**	**4405**	**20.5**
			xc4vlx25-11	**4518**	**220**	**3518**	**15.9**
			xc3s1500-5	**4713**	**114**	**6800**	**32.0**

Ronan *et al.* [38] have proposed a hardware accelerator for the genus-2 η_T pairing on the curve C_0. However, they selected a much lower level of security and assumed that both arguments were degenerate divisors, which does not seem to be applicable to known protocols [13, Remark 2].

Most of the literature about pairing computation on FPGA is devoted to supersingular elliptic curves, and only focuses on low- or medium-security levels. We summarized the most relevant results in Table 4 and refer the reader to [6] for a comprehensive bibliography. Since the datapath handles the field of definition of the curve, one has to increase the area of the circuit in order to improve the level of security. It is therefore difficult to achieve the AES-128 level of security without risking to exhaust the FPGA resources. One way to avoid this pitfall is to use higher genus curves. Another way is to consider supersingular elliptic curves over field extensions of moderately-composite degree [15]. Thanks to this method, Estibals designed an accelerator for computing the Tate pairing on a supersingular curve over $\mathbb{F}_{3^{5 \cdot 97}}$, which satisfies the 128-bit security level. Our results are very comparable with Estibals's work when considering general divisors. Chatterjee *et al.* [13] have for instance proposed a variant of the BLS signature scheme [11] in which one argument of each pairing function is a degenerate divisor. In such settings, the number of multiplications over the underlying field becomes significantly smaller (Table 1) and the computation time of our coprocessor should be roughly divided by 2.

Ghosh *et al.* [20] have reported the first FPGA implementation of the R-ate pairing on a BN curve defined over a 256-bit prime field $\mathbb{F}_p$. According to the authors, their parallel coprocessor can be programmed for any curve parameters, however this flexibility comes at the cost of increased hardware resources. Therefore, it should be possible to reduce the gap between this accelerator and our solution by optimising the arithmetic and logic unit for a carefully selected prime p.

The first ASIC implementations of pairings on BN curves with 128 bits of security have been proposed by Fan *et al.* [16] and Kammler *et al.* [27], and compute a pairing in 2.91 ms and 15.8 ms, respectively. It is however difficult to make a fair comparison between our respective works since the curves and the target technologies are not the same.

7 Conclusion and Perspectives

We presented a novel optimal Eta pairing algorithm on supersingular genus-2 binary hyperelliptic curves. Starting from Vercauteren's work on optimal pairings [40], we described how to exploit the action of the 2^{3m}-th power Verschiebung in order to further reduce the loop length of Miller's algorithm with respect to the genus-2 η_T approach [4], thus resulting in a 33% improvement.

In order to demonstrate the efficiency of our approach, we implemented the optimal Eta pairing at the 128-bit security level in both software and hardware. As far as pairings are concerned, our results show that genus-2 curves are a very effective alternative to both ordinary and supersingular elliptic curves.

Furthermore, Lubicz & Robert have recently presented a novel technique for computing the Weil and Tate pairings over abelian varieties based on an efficient representation of their elements by means of theta functions [31]. We are planning to investigate the application of this method to the case of our proposed genus-2 optimal Eta pairing, as both software and hardware implementations might benefit from the faster arithmetic of theta functions.

Acknowledgments

First of all, the authors would like to express their deepest thanks to Guillaume Hanrot who advised us to have a go at genus-2 pairings. He shall receive here our utmost gratitude.

The authors would also like to thank Pierrick Gaudry for the careful proofreading of the technical sections of this paper, along with Gaëtan Bisson, Romain Cosset, and Emmanuel Thomé who were always available to provide some clear answers to our many questions.

References

1. Aranha, D.F., Karabina, K., Longa, P., Gebotys, C.H., López, J.: Faster explicit formulas for computing pairings over ordinary curves. Cryptology ePrint Archive, Report 2010/526 (2010), `http://eprint.iacr.org/`
2. Aranha, D.F., López, J., Hankerson, D.: Efficient software implementation of binary field arithmetic using vector instruction sets. In: Abdalla, M., Barreto, P. (eds.) Progress in Cryptology–LATINCRYPT 2010. Lecture Notes in Computer Science, vol. 6212, pp. 144–161. Springer (2010)
3. Aranha, D.F., López, J., Hankerson, D.: High-speed parallel software implementation of the η_T pairing. In: Pieprzyk, J. (ed.) Topics in Cryptology–CT-RSA 2010. Lecture Notes in Computer Science, vol. 5985, pp. 89–105. Springer (2010)
4. Barreto, P., Galbraith, S., Ó Éigeartaigh, C., Scott, M.: Efficient pairing computation on supersingular Abelian varieties. Designs, Codes and Cryptography 42, 239–271 (2007)
5. Barreto, P., Naehrig, M.: Pairing-friendly elliptic curves of prime order. In: Preneel, B., Tavares, S. (eds.) Selected Areas in Cryptography–SAC 2005. Lecture Notes in Computer Science, vol. 3897, pp. 319–331. Springer (2006)
6. Beuchat, J.L., Detrey, J., Estibals, N., Okamoto, E., Rodríguez-Henríquez, F.: Fast architectures for the η_T pairing over small-characteristic supersingular elliptic curves. Cryptology ePrint Archive, Report 2009/398 (2009), available at `http://eprint.iacr.org/2009/398.pdf`
7. Beuchat, J.L., Díaz, J.G., Mitsunari, S., Okamoto, E., Rodríguez-Henríquez, F., Teruya, T.: High-speed software implementation of the optimal ate pairing over Barreto–Naehrig curves. In: Joye, M., Miyaji, A., Otsuka, A. (eds.) Pairing-Based Cryptography–Pairing 2010. Lecture Notes in Computer Science, Springer (2010), to appear
8. Beuchat, J.L., López-Trejo, E., Martínez-Ramos, L., Mitsunari, S., Rodríguez-Henríquez, F.: Multi-core implementation of the Tate pairing over supersingular elliptic curves. In: Garay, J., Miyaji, A., Otsuka, A. (eds.) Cryptology and Network Security–CANS 2009. pp. 413–432. No. 5888 in Lecture Notes in Computer Science, Springer (2009)

9. Boneh, D., Franklin, M.: Identity-based encryption from the Weil pairing. In: Kilian, J. (ed.) Advances in Cryptology–CRYPTO 2001. pp. 213–229. No. 2139 in Lecture Notes in Computer Science, Springer (2001)
10. Boneh, D., Gentry, C., Waters, B.: Collusion resistant broadcast encryption with short ciphertexts and private keys. In: Shoup, V. (ed.) Advances in Cryptology–CRYPTO 2005. pp. 258–275. No. 3621 in Lecture Notes in Computer Science, Springer (2005)
11. Boneh, D., Lynn, B., Shacham, H.: Short signatures from the Weil pairing. In: Boyd, C. (ed.) Advances in Cryptology–ASIACRYPT 2001. pp. 514–532. No. 2248 in Lecture Notes in Computer Science, Springer (2001)
12. Cantor, D.: Computing in the Jacobian of a hyperelliptic curve. Mathematics of Computation 48(177), 95–101 (Jan 1987)
13. Chatterjee, S., Hankerson, D., Menezes, A.: On the efficiency and security of pairing-based protocols in the type 1 and type 4 settings. In: Hasan, M., Helleseth, T. (eds.) Arithmetic of Finite Fields–WAIFI 2010. Lecture Notes in Computer Science, vol. 6087, pp. 114–134. Springer (2010)
14. Cohen, H., Frey, G. (eds.): Handbook of Elliptic and Hyperelliptic Curve Cryptography. Discrete Mathematics and its Applications, Chapman & Hall/CRC (2006)
15. Estibals, N.: Compact hardware for computing the Tate pairing over 128-bit-security supersingular curves. In: Joye, M., Miyaji, A., Otsuka, A. (eds.) Pairing-Based Cryptography–Pairing 2010. Lecture Notes in Computer Science, Springer (2010), to appear
16. Fan, J., Vercauteren, F., Verbauwhede, I.: Faster $\mathbb{F}_p$-arithmetic for cryptographic pairings on Barreto–Naehrig curves. In: Clavier, C., Gaj, K. (eds.) Cryptographic Hardware and Embedded Systems–CHES 2009. pp. 240–253. No. 5747 in Lecture Notes in Computer Science, Springer (2009)
17. Frey, G., Rück, H.G.: A remark concerning m-divisibility and the discrete logarithm in the divisor class group of curves. Mathematics of Computation 62(206), 865–874 (Apr 1994)
18. Galbraith, S.: Supersingular curves in cryptography. In: Boyd, C. (ed.) Advances in Cryptology–ASIACRYPT 2001. pp. 495–513. No. 2248 in Lecture Notes in Computer Science, Springer (2001)
19. Gaudry, P., Hess, F., Smart, N.: Constructive and destructive facets of Weil descent on elliptic curves. Journal of Cryptology 15(1), 19–46 (2001)
20. Ghosh, S., Mukhopadhyay, D., Chowdhury, D.: High speed flexible pairing cryptoprocessor on FPGA platform. In: Joye, M., Miyaji, A., Otsuka, A. (eds.) Pairing-Based Cryptography–Pairing 2010. Lecture Notes in Computer Science, Springer (2010), to appear
21. Granger, R., Hess, F., Oyono, R., Thériault, N., Vercauteren, F.: Ate pairing on hyperelliptic curves. In: Naor, M. (ed.) Advances in Cryptology–EUROCRYPT 2007. pp. 430–447. No. 4515 in Lecture Notes in Computer Science, Springer (2007)
22. Granger, R., Page, D., Smart, N.: High security pairing-based cryptography revisited. In: Hess, F., Pauli, S., Pohst, M. (eds.) Algorithmic Number Theory–ANTS VII. pp. 480–494. No. 4076 in Lecture Notes in Computer Science, Springer (2006)
23. Gueron, S., Kounavis, M.E.: Carry-less multiplication and its usage for computing the GCM mode. White paper, available at `http://software.intel.com/`
24. Hess, F.: Pairing lattices. In: Galbraith, S., Paterson, K. (eds.) Pairing-Based Cryptography–Pairing 2008. pp. 18–38. No. 5209 in Lecture Notes in Computer Science, Springer (2008)
25. Hess, F., Smart, N., Vercauteren, F.: The Eta pairing revisited. IEEE Transactions on Information Theory 52(10), 4595–4602 (Oct 2006)

26. Joux, A.: A one round protocol for tripartite Diffie–Hellman. In: Bosma, W. (ed.) Algorithmic Number Theory–ANTS IV. pp. 385–394. No. 1838 in Lecture Notes in Computer Science, Springer (2000)
27. Kammler, D., Zhang, D., Schwabe, P., Scharwaechter, H., Langenberg, M., Auras, D., Ascheid, G., Mathar, R.: Designing an ASIP for cryptographic pairings over Barreto–Naehrig curves. In: Clavier, C., Gaj, K. (eds.) Cryptographic Hardware and Embedded Systems–CHES 2009. pp. 254–271. No. 5747 in Lecture Notes in Computer Science, Springer (2009)
28. Koblitz, N., Menezes, A.: Pairing-based cryptography at high security levels. In: Smart, N. (ed.) Cryptography and Coding. pp. 13–36. No. 3796 in Lecture Notes in Computer Science, Springer (2005)
29. Lee, E., Lee, H.S., Park, C.M.: Efficient and generalized pairing computation on abelian varieties. Cryptology ePrint Archive, Report 2008/040 (2008), available at `http://eprint.iacr.org/2008/040.pdf`
30. Lee, E., Lee, Y.: Tate pairing computation on the divisors of hyperelliptic curves of genus 2. Journal of the Korean Mathematical Society 45(4), 1057–1073 (Jul 2008)
31. Lubicz, D., Robert, D.: Efficient pairing computation with Theta functions. In: Hanrot, G., Morain, F., Thomé, E. (eds.) Algorithmic Number Theory–ANTS IX. Lecture Notes in Comput. Sci., vol. 6197, pp. 251–269. Springer (2010)
32. Menezes, A., Okamoto, T., Vanstone, S.: Reducing elliptic curves logarithms to logarithms in a finite field. IEEE Transactions on Information Theory 39(5), 1639–1646 (Sep 1993)
33. Miller, V.: Short programs for functions on curves (1986), available at `http://crypto.stanford.edu/miller`
34. Miller, V.: The Weil pairing, and its efficient calculation. Journal of Cryptology 17(4), 235–261 (2004)
35. Mitsunari, S., Sakai, R., Kasahara, M.: A new traitor tracing. IEICE Trans. Fundamentals E85–A(2), 481–484 (Feb 2002)
36. Montgomery, P.: Five, six, and seven-term Karatsuba-like formulae. IEEE Transactions on Computers 54(3), 362–369 (Mar 2005)
37. Naehrig, M., Niederhagen, R., Schwabe, P.: New software speed records for cryptographic pairings. In: Abdalla, M., Barreto, P. (eds.) Progress in Cryptology–LATINCRYPT 2010. pp. 109–123. No. 6212 in Lecture Notes in Computer Science, Springer (2010)
38. Ronan, R., Ó hÉigeartaigh, C., Murphy, C., Scott, M., Kerins, T.: Hardware acceleration of the Tate pairing on a genus 2 hyperelliptic curve. Journal of Systems Architecture 53, 85–98 (2007)
39. Sakai, R., Ohgishi, K., Kasahara, M.: Cryptosystems based on pairing. In: 2000 Symposium on Cryptography and Information Security–SCIS 2000. pp. 26–28 (2000)
40. Vercauteren, F.: Optimal pairings. IEEE Transactions on Information Theory 56(1), 455–461 (Jan 2010)

A Algorithms for Computing the Optimal Eta Pairing

We detail here the two algorithms used to compute the optimal Eta pairing described in Section 4. The choice of the algorithm to use depends on the value of m mod 6.

Algorithm 1 Computation of the optimal Eta pairing when $m \equiv 1 \pmod 6$.

```
Input: $\overline{D}_1$ and $\overline{D}_2 \in \mathrm{Jac}_{C_d}(\mathbb{F}_{2^m})[\ell]$ represented by the reduced divisors $D_1$ and $D_2$.
Output: $\eta_{[c_0,c_1]}(\overline{D}_1, \overline{D}_2) \in \mu_\ell \subseteq \mathbb{F}^*_{2^{12m}}$.
$G_1 \leftarrow 1$ ; $R_1 \leftarrow [\delta]D_1$ ; $E_2 \leftarrow \epsilon(\psi(D_2))$
for $i \leftarrow 1$ to $(m-1)/6$ do
    $G_1 \leftarrow G_1^8 \cdot f_{8,R_1}(E_2)$
    $R_1 \leftarrow [8]R_1$
end for                     // $G_1 = f_{2^{(m-1)/2},[\delta]D_1}(E_2)$ and $R_1 = [\delta 2^{(m-1)/2}]D_1$.
$G_2 \leftarrow g_1^\delta$ ; $R_2 \leftarrow [\delta]R_1$
for $i \leftarrow 1$ to $(m-1)/6$ do
    $G_2 \leftarrow G_2^8 \cdot f_{8,R_2}(E_2)$
    $R_2 \leftarrow [8]R_2$
end for                     // $G_2 = f_{2^{m-1},D_1}(E_2)$ and $R_2 = [2^{m-1}]D_1$.
$G_2 \leftarrow G_2^2 \cdot f_{2,R_2}(E_2)$                     // $G_2 = f_{2^m,D_1}(E_2)$.
$F_0 \leftarrow G_1 \cdot g_{R_1,D_1}(E_2)$                     // $F_0 = f_{c_0,D_1}(E_2)$.
$F_1 \leftarrow G_1 \cdot G_2 \cdot g_{[2]R_2,R_1}(E_2)$                     // $F_1 = f_{c_1,D_1}(E_2)$.
return $\left(F_1^{2^{3m}} \cdot F_0\right)^{(2^{12m}-1)/L}$
```

Algorithm 2 Computation of the optimal Eta pairing when $m \equiv 5 \pmod 6$.

```
Input: $\overline{D}_1$ and $\overline{D}_2 \in \mathrm{Jac}_{C_d}(\mathbb{F}_{2^m})[\ell]$ represented by the reduced divisors $D_1$ and $D_2$.
Output: $\eta_{[2c_0,2c_1]}(\overline{D}_1, \overline{D}_2) = \eta_{[c_0,c_1]}(\overline{D}_1, \overline{D}_2)^2 \in \mu_\ell \subseteq \mathbb{F}^*_{2^{12m}}$.
$G_1 \leftarrow 1$ ; $R_1 \leftarrow [\delta]D_1$ ; $E_2 \leftarrow \epsilon(\psi(D_2))$
for $i \leftarrow 1$ to $(m+1)/6$ do
    $G_1 \leftarrow G_1^8 \cdot f_{8,R_1}(E_2)$
    $R_1 \leftarrow [8]R_1$
end for                     // $G_1 = f_{2^{(m+1)/2},[\delta]D_1}(E_2)$ and $R_1 = [\delta 2^{(m+1)/2}]D_1$.
$G_2 \leftarrow g_1^\delta$ ; $R_2 \leftarrow [\delta]R_1$
for $i \leftarrow 1$ to $(m+1)/6$ do
    $G_2 \leftarrow G_2^8 \cdot f_{8,R_2}(E_2)$
    $R_2 \leftarrow [8]R_2$
end for                     // $G_2 = f_{2^{m+1},D_1}(E_2)$ and $R_2 = [2^{m+1}]D_1$.
$F_0 \leftarrow G_1 \cdot f_{2,D_1}(E_2) \cdot g_{R_1,[2]D_1}(E_2)$                     // $F_0 = f_{2c_0,D_1}(E_2)$.
$F_1 \leftarrow G_1 \cdot G_2 \cdot g_{R_2,R_1}(E_2)$                     // $F_1 = f_{2c_1,D_1}(E_2)$.
return $\left(F_1^{2^{3m}} \cdot F_0\right)^{(2^{12m}-1)/L}$
```

B Architecture of the hardware accelerator

We present in this section the design of the coprocessor by Beuchat *et al.* that we used for the computation of our optimal Eta pairing [6]. In order to best fit the arithmetic of $\mathbb{F}_{2^{367}}$, we parametrised their architecture as follows:

- The multiplier processes $D = 16$ coefficients and thus performs a multiplication over $\mathbb{F}_{2^{367}}$ in 23 clock cycles.
- We chose to support the 3-fold Frobenius map (*i.e.* raising to the eighth power) in the unified operator.
- The register file can store up to 127 intermediate variables belonging to $\mathbb{F}_{2^{367}}$ (46 kbit of RAM), along with the constant 1.

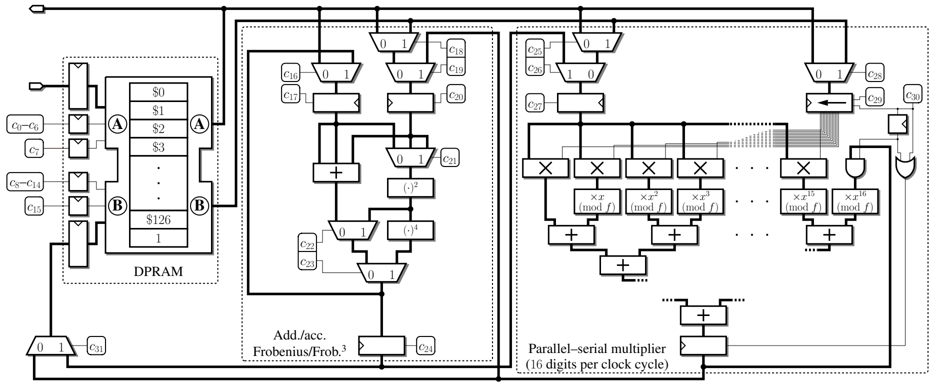

Fig. 1. A finite field coprocessor for $\mathbb{F}_{2^{367}}$.