Pairings on hyperelliptic curves with a real model

Steven D. Galbraith[1], Xibin Lin[1,2], and David J. Mireles Morales[1]

[1]Mathematics Department
Royal Holloway, University of London
United Kingdom
{steven.galbraith, d.mireles-morales}@rhul.ac.uk
[2]School of Mathematics and Computational Science
Sun-Yat Sen University
P.R.China
linxibin@mail2.sysu.edu.cn

Abstract. We analyse the efficiency of pairing computations on hyperelliptic curves given by a real model using a balanced divisor at infinity. Several optimisations are proposed and analysed. Genus two curves given by a real model arise when considering pairing friendly groups of order dividing $p^2 - p + 1$. We compare the performance of pairings on such groups in both elliptic and hyperelliptic versions. We conclude that pairings can be efficiently computable in real models of hyperelliptic curves.

1 Introduction

The study of efficient pairing computation on hyperelliptic curves has focused exclusively on the analysis of hyperelliptic curves given by an imaginary model. With the development of new divisor addition algorithms on hyperelliptic curves given by a real model [5], it is natural to ask if pairings can be implemented on these curves competitively.

The authors of [6] construct a genus 2 curve C, defined over $\mathbf{F}_p$ for p a prime $p \equiv 5 \mod 6$. The jacobian $\mathrm{Jac}(C)$ of this curve has $p^2 - p + 1$ points, and embedding degree 6 with respect to any subgroup with prime order $r > 3$. The curve C is given by a real model (see [5]), which in particular means that it has 2 points at infinity.

In [15], Verheul presents the construction of an elliptic curve with embedding degree 3. This curve is defined over a field $\mathbf{F}_{p^2}$ for p a prime $p \equiv 5 \mod 6$, and has $p^2 - p + 1$ $\mathbf{F}_{p^2}$-rational points. Pairings on these elliptic curves have been studied by Hu et.al. in [10].

The similiarities between these curves make them natural candidates for a comparison between elliptic and hyperelliptic curve pairing implementations. In this article we explore several optimisation techniques on these curves, implemenent pairings and compare their performance. Among the optimisations used in the implementation is the recent R-ate pairing proposal presented by Lee, Lee and Park in [11], and the well-known *denominator elimination* technique, which is combined with the R-ate pairing thanks to Theorem 2.

A crucial step towards a competitive implementation of pairings on hyperelliptic curves given by a real model is having efficient divisor addition algorithms that result in simple Miller functions. The addition algorithms presented in [5] allow for a fast implementation not only because the operation count in the addition and doubling algorithms is smaller than that in previous proposals [14], but also because the Miller function, whose evaluation is the bottleneck in pairing computations on high genus curves, is simpler using the algorithms of [5]. We make a theoretical and practical comparison of the efficiency of our pairings compared with that of pairings on elliptic and hyperelliptic curves. We conclude that pairings can be efficiently implemented on hyperelliptic curves given by a real model.

This article is organized as follows: Section 2 describes the representation of divisors (and hence the addition algorithms) that we will use for genus 2 curves given by a real model. In this section we also present the embedding degree 6 construction of Galbraith, Pujolas, Ritzenthaler and Smith [6]. Section 3 presents a brief overview of pairing computation techniques, including the recently presented R-ate pairing. Section 4 describes our parameter generation algorithms and the optimisations used in the implementation. In Section 5 we report our implementation results and compare them with pairing computation results obtained for similar elliptic or hyperelliptic curves. Some conclusions are discussed in Section 6.

2 Curves

Given an algebraic curve C and two divisors D_0 and D_1 on C, we say that D_0 and D_1 are *linearly equivalent*, denoted $D_0 \sim D_1$, if there is a function f such that

$$\operatorname{div}(f) = D_1 - D_0$$

where $\operatorname{div}(f)$ is the divisor of f.

Definition 1. *The* divisor class group *of C is the group of divisor classes modulo linear equivalence. We will denote it as $\mathrm{Cl}(C)$. The class of a divisor D in $\mathrm{Cl}(C)$ will be denoted by $[D]$. We define $\mathrm{Cl}^0(C)$ as the degree zero subgroup of $\mathrm{Cl}(C)$.*

Notice that the degree of the divisor $\operatorname{div}(f)$ associated to a function f is always zero, and thus it makes sense to talk of the degree of a divisor class $[D]$ in $\mathrm{Cl}(C)$. In this article we will work exclusively with curves C which are elliptic or hyperelliptic curves of genus 2.

2.1 Arithmetic on hyperelliptic curves

Let C be a genus 2 hyperelliptic curve given by

$$C : y^2 = F(x),$$

where $\mathrm{char}(K) \neq 2,3$ and $F(x) \in K(x)$ is a square-free degree 6 polynomial. We say that this is a *real model* for C. The desingularization of C has 2 different points at infinity, which we will denote ∞^+ and ∞^-. Let $D_\infty = \infty^+ + \infty^-$, note that this divisor is K-rational even if the points ∞^+ and ∞^- are not independently so.

Proposition 1 (Proposition 1 in [5]). *Let D_∞ denote the divisor $D_\infty = \infty^+ + \infty^-$, and let $D \in \mathrm{Div}_0(C)$ be a K-rational divisor on the curve C. Then $[D]$ has a unique representative in $\mathrm{Cl}^0(C)$ of the form $[D_0 - D_\infty]$, where $D_0 = P_1 + P_2$ is an effective K-rational divisor of degree 2 such that $P_1 \neq \bar{P}_2$.*

If $D_0 = P_1 + P_2$, generically $P_1, P_2 \notin \{\infty^+, \infty^-\}$, so we will only discuss arithmetic for generic divisors. Further details can be found in [5].

We will use Mumford's representation to represent divisors of the form

$$D = P_1 + P_2 - D_\infty, \quad P_1, P_2 \notin \{\infty^+, \infty^-\}.$$

Let $P_i = (x_i, y_i)$ for $i \in \{1, 2\}$. Mumford's representation is a pair of polynomials $(u(x), v(x))$, where $u(x) = (x - x_1)(x - x_2)$ and where $v(x)$ satisfies $v(x)^2 - F(x) \equiv 0 \mod u(x)$. This last condition implies that $y_i = v(x_i)$. The polynomial v is only determined modulo u; if a canonical representative is needed, the unique representative with $\deg v < \deg u$ can be used.

We will denote the divisor $D = P_1 + P_2 - D_\infty$ associated to the pair of polynomials (u, v) as $D = \mathrm{div}(u, v)$. Traditionally this notation has been used to denote the affine divisor $P_1 + P_2$ but we will extend it since there is no risk of confusion.

Let $D_1 = P_1 + P_2 - D_\infty$ and $D_2 = P_3 + P_4 - D_\infty$ be two divisors. An explicit interpretation of the results of [5] in the case of a genus 2 curve implies that if $p(x)$ denotes the unique polynomial of degree at most 3 passing through P_1, P_2, P_3 and P_4, and we let P_5, P_6 be the remaining intersection points of $y - p(x)$ with C, then

$$\mathrm{div}(y - p(x)) = \sum_{i=1}^{6} P_i - 3D_\infty. \tag{1}$$

If we write $D_3 = \bar{P}_5 + \bar{P}_6 - D_\infty$, equation (1) can be rewritten as

$$[D_1] + [D_2] = [D_3].$$

If u_3 is the first polynomial in the Mumford representation of D_3, the function

$$g_{D_1,D_2} = \frac{y - p(x)}{u_3} \tag{2}$$

has associated divisor $D_1 + D_2 - D_3$. This will be used later to compute pairings.

In our pairing implementation we will use the addition formulae presented in [3], which we include in an appendix for completeness. The polynomial $p(x)$ in equation (1) can be easily computed from the intermediate results in the addition formulae from [3] and presented in the Appendix.

When the divisor at infinity used is the traditional $D_\infty = 2\infty^+$, the function g_{D_1,D_2} with divisor $D_1 + D_2 - D_3$ has the form $g_{D_1,D_2} = (y - p_1(x))(y - p_2(x))/(u_3(x)u_4(x))$, where again $p_1(x)$ and $p_2(x)$ are cubic polynomials and $u_3(x), u_4(x)$ are quadratic polynomials. Since the bottleneck of pairing calculations is precisely the evaluation of this function, the speed-up obtained from using the representation of $\mathrm{Cl}^0(C)$ described in [5] goes beyond the operations saved in the addition algorithm.

2.2 Hyperelliptic curves with embedding degree 6

In this section we will substitute the notation $\mathrm{Cl}^0(C)$ we had been using for the more geometric (and equivalent) $\mathrm{Jac}(C)$, better suited when dealing with endomorphism rings.

In [6, Section 7], the authors present a family of genus 2 curves with embedding degree 6 and generators of a subring R of the endomorphism ring of $\mathrm{Jac}(C)$, such that R contains a distortion map for any non-trivial pair (D_1, D_2) of divisors.

The curves in this family will have 2 points at infinity and our addition algorithm is well-suited to perform efficient arithmetic on them. We now briefly describe the construction of the curves given in [6, Section 7].

Let $p \neq 2$ a prime such that $p \equiv 2 \pmod 3$. Denote by ζ_6 a root of $x^2 - x + 1$ and by $\zeta_3 = \zeta_6^2$, let $\gamma \in \mathbf{F}_{p^6}$ be such that $\gamma^{p^2-1} = \zeta_3$. An equation of C will then be

$$C : y^2 = (ax + b)^6 + (cx + d)^6,$$

where $a = \gamma^p, b = \zeta_3^2\gamma^p, c = \gamma$ and $d = \zeta_3\gamma$.

In this case, the coefficient of the x^6 term in the equation of C is $a^6 + c^c$, which is a non-zero $\mathbf{F}_p$-rational element. If it is not a square, we can take two rational points on C and move them to the line at infinity, and get a curve isomorphic to C given by a monic polynomial. This will let us use the addition formulae presented in [3], which only work on curves given by an equation of the form $y^2 = x^6 + f_4x^4 + f_3x^3 + f_2x^2 + f_1x + f_0$.

Lemma 1. *The model of the curve C defined above has 2 points at infinity.*

Proof. Let C be given by $y^2 = F(x)$ and denote the leading coefficient of F as F_6. Notice that

$$F_6 = a^6 + c^6 = \gamma^{6p} + \gamma^6.$$

To prove the lemma we only need to prove that $F_6 \neq 0$. Since $p^2 - 1$ is a multiple of 3 and $\gamma^{p^2-1} = \zeta_3$ the multiplicative order of γ is a multiple of 9. So $F_6 = \gamma^6(\gamma^{6p-6} + 1)$ cannot be zero as this would imply that $\gamma^{12p-12} = 1$, but $12p - 12$ is not a multiple of 9 as $p \equiv 2 \pmod 3$. □

The characteristic polynomial of Frobenius on C is $T^4 - pT^2 + p^2$, so $\mathrm{Jac}(C)$ will have $p^2 - p + 1$ elements. This implies that if r is a prime that divides $p^2 - p + 1$, the embedding degree of C with respect to r is 6. Note that if C'

is the curve $C' : y^2 = x^6 + 1$, then C is a twist of C' by the automorphism $u : (x, y) \mapsto (\frac{\zeta_3}{x}, \frac{y}{x^3})$. Furthermore, there is an isomorphism $\phi : C \longrightarrow C'$ given by

$$\phi(x, y) = \left(\frac{ax + b}{cx + d}, \frac{y}{(cx + d)^3}\right)$$

The authors of [6] then define the following endomorphisms of C':

$$\pi(x, y) = (x^p, y^p)$$

$$\chi(x, y) = \left(\frac{1}{x}, \frac{y}{x^3}\right)$$

$$\zeta_6(x, y) = (\zeta_6 x, y).$$

We will abuse notation and extend these endomorphisms to $\mathrm{Jac}(C')$. These endomorphisms are enough to find a distortion map on $\mathrm{Jac}(C)$ (see Definition 3), as the following result shows.

Theorem 1 (Theorem 7.2 in [6]). *Let r be a prime different from 2 and p. Then for all pairs of divisors D_1 and D_2 on C of order r, there exists a distortion map in the ring $\phi^{-1}\mathbb{Z}[\pi, \chi, \zeta_6]\phi$.*

It is well known that if the first coordinate of the Mumford representation of a divisor lies in a proper subfield of $\mathbf{F}_{p^6}$, then the function g_{D_1,D_2} in equation (2) can be substituted by $y - p(x)$ (p as in equation (2)) in the Miller loop of the pairing computation. The following Lemma shows that the automorphisms χ and ζ_6 can be used to this end.

Lemma 2. *Let $P \in C$ be a point with a $\mathbf{F}_p$-rational x-coordinate. Then:*

- *The x-coordinate of $(\phi^{-1} \circ \zeta_6 \circ \phi)(P)$ is $\mathbf{F}_p$-rational.*
- *The x-coordinate of $(\phi^{-1} \circ \chi \circ \phi)(P)$ is $\mathbf{F}_{p^3}$-rational.*

Proof. Let $P = (x, y)$ be the coordinates of P. A tedious but simple calculation shows that the x-coordinate of $(\phi^{-1} \circ \zeta_6 \circ \phi)(x, y)$ is given by

$$\frac{-x - 1}{x - 2},$$

which is $\mathbf{F}_p$-rational whenever x is an element of $\mathbf{F}_p$.

The x-coordinate of $(\phi^{-1} \circ \chi \circ \phi)(x, y)$ is given by

$$x_\chi = \frac{(\zeta_3\gamma^2 - \zeta_3^2\gamma^{2p})x + (\zeta_3^2\gamma^2 - \zeta_3\gamma^{2p})}{(\gamma^{2p} - \gamma^2)x + (\zeta_3^2\gamma^{2p} - \zeta_3\gamma^2)},$$

and again, it is straightforward to prove that $x_\chi^{p^3} = x_\chi$. □

The previous Lemma shows that using χ and ζ_6 as distortion maps (see Definition 3) makes it possible to use *denominator elimination.* We will now prove that the image of $\mathbf{F}_p$-rational divisors under the distortion map $(\phi^{-1} \circ \chi \circ \zeta_6 \circ \phi)$ lies in the p-eigenspace, thus allowing us to directly use loop-shortening techniques.

Theorem 2. *Let $D_1 \in \mathrm{Cl}^0(C)[r]$ be a $\mathbf{F}_p$-rational divisor. Then its image $D_2 = (\phi^{-1} \circ \chi \circ \zeta_6 \circ \phi)(D_1)$ under the distortion map lies in the p-eigenspace of* $\mathrm{Cl}^0(C)[r]$.

Proof. The r-torsion subgroup $\mathrm{Cl}^0(C)[r]$ can be decomposed as the direct sum of four 1-dimensional eigenspaces with respect to Frobenius π_p, with eigenvalues $1, -1, p$ and $-p$. The polynomial $T^2 - T + 1$ is divisible by $T - p \mod r$, hence the endomorphism $(\pi_p^2 - \pi_p + 1)$ annihilates the p-eigenspace, and is invertible when restricted to the other eigenspaces. It follows that D_2 lies in the p-eigenspace if and only if $(\pi_p^2 - \pi_p + 1)(D_2) = 0$.

To prove that this is the case, it suffices to show that the unique cubic polynomial passing through the four points in the affine support of D_2 and $\pi_p^2(D_2)$ also passes through the points in the affine support of $\pi_p(D_2)$. This can be proven symbolically simply by defining formal variables γ and γ^p over $\mathbb{Q}(\zeta_6)$, and formally defining the action of Frobenius as $\pi_p(\gamma) = \gamma^p$, $\pi_p(\gamma^p) = \zeta_6^2 \gamma$ and $\pi_p(\zeta^6) = \zeta_6^5$. The verification of our claim boils down to a trivial, albeit tedious calculation, which we performed using Magma [2]. □

2.3 Elliptic curves with embedding degree 3

In this subsection we describe the construction of elliptic curves with embedding degree $k = 3$ given in [15]. We will report our pairing implementation results on these curves in later sections.

Let p be a prime, $p \equiv 5 \bmod 6$, let E be an elliptic curve defined over $\mathbf{F}_{p^2}$ by $y^2 = x^3 + \rho^2$, where $\rho \in \mathbf{F}_{p^2}$ is an element such that ρ^2 is not a cube in $\mathbf{F}_{p^2}$. The number of $\mathbf{F}_{p^2}$ rational points of E is $p^2 - p + 1$ (see Lemma 7 of [7] for a proof). Let r be the largest prime dividing $p^2 - p + 1$, then E has embedding degree $k = 3$ with respect to r. Define the following map:

$$\begin{aligned} \phi_E : E(\mathbf{F}_{p^2}) &\to E(\mathbf{F}_{p^6}) \\ (x, y) &\to (a\beta x^p, by^p) \end{aligned}$$

where $a = \rho^{-(2p-1)/3}$, $b = \rho^{-(p-1)}$, and β is a cubic root of ρ in $\mathbf{F}_{p^6}$. If we let $(x', y') = \phi_E(x, y)$, it is not hard to see that $x' \in \mathbf{F}_{p^6}$ and $y' \in \mathbf{F}_{p^2}$. The endomorphism ϕ_E will be used as a distortion map in our pairing implementation (see Definition 3).

3 Pairings

3.1 Background on the Tate pairing

We will briefly recall the definition of the Tate pairing (see [4] for a more detailed description) and describe the applications of the results in [5] to the computation of pairings on hyperelliptic curves given by a real model. Let $\mathbf{F}_q$ be a finite field with $q = p^n$ elements and let C be a smooth, irreducible curve over $\mathbf{F}_q$. Denote the degree zero divisor class group of C by $\mathrm{Cl}^0_{\mathbf{F}_q}(C)$. Let r be an integer such that $r \mid \# \mathrm{Cl}^0_{\mathbf{F}_q}(C)$ and denote by $\mathrm{Cl}^0_{\mathbf{F}_q}(C)[r]$ the group of divisor classes of order dividing r.

Let k be the smallest integer such that $r \mid (q^k - 1)$. We say that k is the embedding degree of C.

Let $D_1 \in \mathrm{Cl}^0_{\mathbf{F}_q}(C)[r]$ and $D_2 \in \mathrm{Cl}^0_{\mathbf{F}_{q^k}}(C)$ be two divisors. Since rD_1 is principal, there is a function f_{r,D_1} defined over $\mathbf{F}_q$ such that $\mathrm{div}(f_{r,D_1}) = rD_1$. The Tate pairing is defined as

$$\langle D_1, D_2 \rangle_r = f_{r,D_1}(D_2),$$

and one can prove that it is a non-degenerate, bilinear pairing:

$$\mathrm{Cl}^0_{\mathbf{F}_{q^k}}(C)[r] \times \mathrm{Cl}^0_{\mathbf{F}_{q^k}}(C)/r\,\mathrm{Cl}^0_{\mathbf{F}_{q^k}}(C) \longrightarrow \mathbf{F}^*_{q^k}/(\mathbf{F}^*_{q^k})^r.$$

The result is only defined up to an r-th power, hence to obtain a unique representative, one defines the reduced Tate pairing as

$$e(D_1, D_2) = \langle D_1, D_2 \rangle_r^{(q^k-1)/r} = f_{r,D_1}^{(q^k-1)/r}(D_2).$$

In practice to compute the Tate pairing one uses Miller's algorithm, which we now describe.

Definition 2. *Let C be a curve for which there exists a way to select a canonical representative for every element of $\mathrm{Cl}^0_{\mathbf{F}_{q^k}}(C)$. Given a degree 0 divisor D on C and an integer n, let D_n be the canonical representative of the class $[nD]$. We will denote the unique function (up to scalar multiples) with associated divisor $nD - D_n$ as $f_{n,D}$.*

By definition, given two degree 0 divisors D_1, D_2 on C, if D_3 is the canonical representative of $[D_1 + D_2]$, there is a function whose associated divisor is $D_1 + D_2 - D_3$. Denote this function as g_{D_1,D_2}. Miller's fundamental observation is that

$$f_{n_1+n_2,D} = f_{n_1,D} \cdot f_{n_2,D} \cdot g_{n_1D,n_2D}, \tag{3}$$

which allows us to compute $f_{r,D}$ (and hence the Tate-pairing) using a square and multiply calculation with $O(\log r)$ steps. Note that in the case of a genus 2 hyperelliptic curve, g_{n_1D,n_2D} is given by equation (2). We will refer to the process of calculating $f_{n_1+n_2,D}$ from $f_{n_1,D}$ and $f_{n_2,D}$ as a Miller step.

Definition 3. *Let $e : \mathbf{G}_1 \times \mathbf{G}_2 \longrightarrow \mathbf{G}_T$ be a non-degenerate bilinear pairing. A morphism $\psi : \mathbf{G}_1 \longrightarrow \mathbf{G}_2$ is called a* distortion map *for $D_1 \in \mathbf{G}_1$ if $e(D_1, \psi(D_1)) \neq 1$.*

Several techniques have been developed to reduce the length of the Miller loop in pairing computations. We will now describe these techniques for elliptic and hyperelliptic curves.

3.2 Elliptic twisted Ate pairing

Let E be an elliptic curve defined over the finite field $\mathbf{F}_q$, with $\#E(\mathbf{F}_q) = q-t+1$, where t is the trace of Frobenius. Let $T = t-1$, and

$$\mathbf{G}_1 = E[r] \cap \mathrm{Ker}(\pi_q - \mathrm{id}), \text{ and } \mathbf{G}_2 = E[r] \cap \mathrm{Ker}(\pi_q - q).$$

In [9, Section 6], Hess, Smart and Vercauteren prove that if E has a twist of degree d, embedding degree k, and we set $m = \gcd(d,k)$ and $e = k/m$, then the function

$$e(P,Q) = f_{T^e,P}(Q)^{(q^k-1)/r}, \tag{4}$$

defines a bilinear function on $\mathbf{G}_1 \times \mathbf{G}_2$, called the *twisted Ate pairing.*

We know that the elliptic curve E constructed in Subsection 2.3 accepts a twist of degree 3, has embedding degree $k=3$ and $T = p-1$. Using equation (4), the function

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

defines a bilinear function on $\mathbf{G}_1 \times \mathbf{G}_2$.

Since $k=3$, the denominator elimination method of [1] does not apply. We now describe a way to replace the denominator with a few multiplications.

When executing Miller's algorithm to compute pairings on an elliptic curve, the denominator of the function $g_{n_1,n_2,D}$ in equation (3) has the form $(x_R - x_Q)$, where R and Q are points on the elliptic curve. Note that $x_R \in \mathbf{F}_{p^2}$ and $x_Q \in \mathbf{F}_{p^6}$. We replace

$$\frac{1}{x_R - x_Q} = \frac{x_R(x_R + x_Q) + x_Q^2}{y_R^2 - y_Q^2},$$

and since $y_R^2 - y_Q^2$ lies in the proper subfield $\mathbf{F}_{p^2}$ of $\mathbf{F}_{p^6}$, we can discard its value as it will become 1 after the final exponentiation.

So the function $g_{n_1,n_2,D}$ in equation (3) can be substituted by

$$l_{R,P}(Q) \cdot (x_R(x_R + x_Q) + x_Q^2), \tag{5}$$

where $l_{R,P}$ denotes the line passing through the points P and R. If x_Q^2 is precomputed then the saving compared with the standard method (i.e., writing the Miller variable f as a numerator and a denominator) is to replace a squaring in $\mathbf{F}_{p^6}$ by a multiplication of an element in $\mathbf{F}_{p^2}$ with an element in $\mathbf{F}_{p^6}$.

3.3 Hyperelliptic Ate pairings

We have seen that in some cases it is possible to compute pairings using a function $f_{n,D}$ where n is much smaller than required for the Tate-pairing. We will revisit some of these techniques in the case of hyperelliptic curves.

Let C be a hyperelliptic curve defined over a finite field $\mathbf{F}_q$. Denote the Frobenius automorphism of C as π_q, and extend this notation to $\mathrm{Cl}^0_{\mathbf{F}_{q^k}}(C)$. Let

$$\mathbf{G}_1 = \mathrm{Cl}^0_{\mathbf{F}_{q^k}}(C)[r] \cap \mathrm{Ker}(\pi_q - \mathrm{id}) \text{ and } \mathbf{G}_2 = \mathrm{Cl}^0_{\mathbf{F}_{q^k}}(C)[r] \cap \mathrm{Ker}(\pi_q - q),$$

denote the 1- and q-eigenspaces of π_q in the r-torsion subgroup of $\mathrm{Cl}^0_{\mathbf{F}_{q^k}}(C)$. If $D_1 \in \mathbf{G}_1$ and $D_2 \in \mathbf{G}_2$ are divisors on C, the authors of [8] proved:

Theorem 3. *The function $e_q : \mathbf{G}_1 \times \mathbf{G}_2 \longrightarrow \mu_r$, given by*

$$e_q(D_1, D_2) = f_{q,D_1}(D_2)^{(q^k-1)/r},$$

defines a non-degenerate bilinear pairing on $\mathbf{G}_1 \times \mathbf{G}_2$.

3.4 *R*-ate pairings

Let $\mathbf{G}_1$ and $\mathbf{G}_2$ be subgroups of the class group of a curve C. If $D_1 \in \mathbf{G}_1$ and $D_2 \in \mathbf{G}_2$, Lee, Lee and Park prove in [11] the following:

Theorem 4. *[Theorem 3.2 in [11]] Let A, B, a, b be integers such that $A = aB + b$, where the functions $f_{A,D}$ and $f_{B,D}$ define bilinear maps in $\mathbf{G}_1 \times \mathbf{G}_2$. Then the function*

$$f_{a,BD}(E) \cdot f_{b,D}(E) \cdot g_{aBD,bD}(E),$$

defines a bilinear map in $\mathbf{G}_1 \times \mathbf{G}_2$.

Note that if B is the order of D, then the functions $f_{a,BD}$ and $g_{aBD,bD}$ are constant, so the function $f_{b,D}(E)$ will define a bilinear map.

For the elliptic curve E constructed in Subsection 2.3, if P is a $\mathbf{F}_{p^2}$-rational point, both $f_{p-1,P}$ and $f_{r,P}$ define bilinear maps on appropriate subgroups of $E[r]$. Similarly, if D_1 is a r-torsion, $\mathbf{F}_p$-rational divisor on the curve C defined in Subsection 2.2, then the functions f_{p,D_1} and f_{r,D_1} define bilinear maps on appropriate subgroups of $\mathrm{Cl}^0(C)[r]$.

Remark 1. Letting $B = r$, $A_E = p - 1$ for $E, A_C = p$ for C, and choosing a, b such that $p = a \cdot r + b$, using Theorem 4, it follows that the function $f_{b-1,P}$ defines a bilinear map on (subgroups of) $E[r]$ and f_{b,D_1} defines a bilinear map on (subgroups of) $\mathrm{Cl}^0(C)[r]$. Choosing an appropriate b could greatly improve the pairing computations, we show how to do this in the following section.

4 Pairing implementation and efficiency analysis

In this section, we describe some optimizations of the pairing implementation on the hyperelliptic curves given above, including the generation of parameters to shorten the Miller loop, denominator elimination, and the finite field construction.

4.1 Efficient generation of parameters

In this subsection, we describe a method to generate parameters for the curves constructed in Subsections 2.2 and 2.3, which will shorten the Miller loop to half the bit-length of the subgroup order r.

Using Remark 1, if $b \equiv p \mod r$, then the functions f_{b,D_1} and $f_{b-1,P}$ give bilinear functions on the appropriate subgroups of $\mathrm{Cl}^0(C)[r]$ and $E[r]$ respectively.

In both cases, considering the current security level (AES 80), r is about the same size of p. Algorithm 1 shows how to choose p, r and b efficiently. As can be seen from the algorithm, b can be chosen to have very low hamming weight and half the bit-length of r.

Algorithm 1 Parameter Generation

```
INPUT: Integers $n, k_{max}$.
OUTPUT: Integers $b, r$ and a prime $p$ such that $r | p^2 - p + 1, p \equiv b \mod r$ .
repeat
    Choose $b$ of size $n$ bits and low hamming weight.
    Let $r = b^2 - b + 1$.
until $r$ is prime or nearly prime.
for $k$ from 1 to $k_{max}$. do
    let $p = k \cdot r + b$.
    if $p$ is a prime and $p \equiv 11 \mod 12$. then
        Break.
    end if
end for
if $k = k_{max}$, goto step 1.
return $p$, $r$, $b$.
```

The following is a set of parameters generated by Algorithm 1, using $n = 80$. These are the parameters used in our implementation, which will be described in the following section.

Example 1. **A set of parameters for AES 80 security**

- p =B00000000000000011260000000000000006AEFB
- r =100000000000000000018F0000000000000009B79
- b =100000000000000000C8

Remark 2. Algorithm 1 can be generalized to find parameters for many other types of curves. For example, a similar algorithm can be used to generate parameters for supersingular genus 2 curves given by an equation of the form $y^2 = x^5 + a$, where $a \in \mathbf{F}_p^*$, $p \equiv 2, 3 \mod 5$. Ó hÉigeartaigh and M. Scott efficiently implemented pairings on these curves in [13], achieving some of the fastest pairing computations on genus 2 curves. Using a parameter selection algorithm similar to Algorithm 1 could further improve their results.

4.2 Finite field construction and arithmetic

The following field construction was presented by Hu et al. in [10].

We restrict to $p \equiv 3 \mod 4$ so that -1 is not a quadratic residue modulo p. In other words, we require $p \equiv 11 \mod 12$. The finite fields are represented as follows:

$$\begin{aligned}\mathbf{F}_{p^2} &= \mathbf{F}_p[x]/(\alpha^2+1) = \{u\alpha + v | u, v \in \mathbf{F}_p\} = \{a_1 + a_2\beta^3 | a_1, a_2 \in \mathbf{F}_p\}.\\ \mathbf{F}_{p^6} &= \mathbf{F}_{p^2}[y]/(\beta^3 - \rho) = \{b_0 + b_1\beta + b_2\beta^2 + b_3\beta^3 + b_4\beta^4 + b_5\beta^5 | b_i \in \mathbf{F}_p\}\\ &= \{c_0 + c_1\beta + c_2\beta^2 | c_i \in \mathbf{F}_{p^2}\}\end{aligned}$$

where $\rho = \alpha + u_0$ and u_0 is a small integer such that $x^3 - \rho$ is irreducible over $\mathbf{F}_{p^2}$.

Let $e_{ij} \in \mathbf{F}_p$ be defined by $\beta^{ip} = e_{i0} + e_{i1}\beta + \cdots + e_{i5}\beta^5$. We have that $\beta^{ip} = \beta^{2i}\rho^{i(p-2)/3}$. Since $\beta^3 = \rho$ and $\rho \in \mathbf{F}_{p^2}$, there are at most two non-zero terms in the coefficients vector $(e_{i0}, e_{i1}, \cdots e_{i5})$. Specifically, we have $(e_{30}, e_{31}, \cdots e_{35}) = (2u_0, 0, 0, -1, 0, 0)$.

So that raising a random element to the pth power is given by

$$(b_0 + b_1\beta + b_2\beta^2 + b_3\beta^3 + b_4\beta^4 + b_5\beta^5)^p = b_0 + \sum_{i=1}^{5} b_i(e_{i0} + e_{i1}\beta + \cdots + e_{i5}\beta^5).$$

This computation costs only $8\mathbf{F}_p-$multiplications (remember u_0 is a small integer).

The final exponentiation is often computed through base p expansion. In the cases $k = 6$, the final exponentiation can be represented as

$$\frac{p^6-1}{r} = (p^3-1)(p+1)\frac{p^2-p+1}{r} = (p^3-1)(p+1)(k_1 p + k_0)$$

where k_1 is small. Thus, the construction above allows for very fast exponentiation.

4.3 Optimized pairing computation

The cost of Miller's algorithm to compute pairings is determined by the length of the Miller loop, the cost of the calculations inside the loop, and the final exponentiation. To compute pairings on hyperelliptic genus 2 curves given by a real model, we used the techniques described above to speed up the computation, that is:

- Algorithm 1 generates suitable parameters to get a short, low Hamming weight Miller loop.
- Use $D_\infty = \infty^+ + \infty^-$ to represent elements of $\mathrm{Cl}^0(C)$ to get fast addition and a simple Miller funciton.
- The distortion map $(\phi^{-1} \circ \chi \circ \zeta_6 \circ \phi)$ described in Theorem 2 allows for denominator elimination while using the R-ate pairing [11] technique.
- The field construction in Subsection 4.2 provides the arithmetic for a very efficient final exponentiation.

5 Efficiency analysis and implementation results

The optimization techniques described above make the computation of pairings on hyperelliptic genus 2 curves practical and efficient. In this section we analyse the efficiency, and compare it with pairing implementations on elliptic curves with similar characteristics.

5.1 Comparision with elliptic curves with $k = 3$

As mentioned in the introduction, the curves constructed in Subsections 2.2 and 2.3 have very similiar characteristics, so implementation results on the embedding degree 3 elliptic curve provide a useful benchmark to analyse our pairing implementation on hyperelliptic curves given by a real model.

As mentioned before, (the class groups of) both curves have the same number of $\mathbf{F}_p$-rational points, and the *embedding field* for both curves is the same, as is the bandwith requirement. A point $P = (x, y) \in E(\mathbf{F}_{p^2})$ is represented by 4 elements of $\mathbf{F}_p$, which is the same number of coefficients required to represent a divisor $D = (x + u_1 x + u_0, x^3 + v_1 x + v_0)$. Since the target field is the same, both pairing values can be compressed at the same rate by using the technique of the XTR public key cryptosystem [12].

In the notation of Theorem 4, we need to calculate $f_{b,D}$. Since b is an integer calculated using Algorithm 1, it will have very low Hamming weight and we will only analyse the cost of the doubling steps in the Miller loop.

In our implementation, the second argument of the pairing in hyperelliptic curves is a divisor $D_2 = (R_1) + (R_2) - D_\infty$, with R_1 and R_2 known points with $\mathbf{F}_{q^3}$-rational x-coordinates. The divisor D_2 is calculated as the image under the distortion map of a divisor $P_1 + P_2 - D_\infty$, where P_1 and P_2 are $\mathbf{F}_p$-rational points. Theorem 2 proves that D_2 lies in the p-eigenspace, and hence R-ate pairings can be used at no extra cost. The Miller functions are evaluated on each point in the affine support of D_2.

To compare the efficiency of our pairing implementations on elliptic and hyperelliptic curves, we first estimate the cost of each doubling step. We will let f denote the intermediate value in the Miller loop. The update of f is similar to that used in other standard implementations of Miller's algorithm, such as Algorithm 1 in Section 2 of [8], except that the denominator of $g_{n_1 D, n_2 D}$ in equation (3) can be removed as described by equation (5) in the elliptic curve case, and by Lemma 2 in the hyperelliptic curve case.

elliptic : $f \leftarrow f^2 \cdot l_{R,P}(Q) \cdot (x_R(x_R + x_Q) + x_Q^2)$ and $R \leftarrow 2R$
hyperelliptic: $f \leftarrow f^2 \cdot (y_1 - p(x_1)) \cdot (y_2 - p(x_2))$ and $D_1 \leftarrow 2D_1$.

Here $l_{R,P}$ is the line through R and P, and $y - p(x)$ is as in equation (1). Note that $p(x)$ will be a cubic polynomial with coefficients in $\mathbf{F}_p$.

We will consider the relative cost of arithmetic operations as described in Section 7 of [9]. Let M_k and I_k denote the cost of multiplication and inversion in $\mathbf{F}_{p^k}$. We will assume that $M_6 = 15M_1$, $M_3 = 5M_1$, $1I_1 = 10M_1$, and $M_1 = S_1$ [9].

We will assume that $1I_1 = 10M_1$, $M_1 = S_1$.

In the elliptic case, the total cost of each doubling Miller step is $83M_1$. In the hyperelliptic case, doubling a divisor costs about $1I_1 + 32M_1 = 42M_1$ [5], which makes the cost of each Miller step $105M_1$. There are a total of 84 doubling steps using the parameters given in Example 1. So the costs of the Miller loops are $6972M_1$ and $8820M_1$ respectively.

The final exponentiation step is identical in both cases, and costs about $1621M_1$.

This shows that pairings on real hyperelliptic genus 2 curves with $k = 6$ are competitive to parings on elliptic curves with $k = 3$.

5.2 Theoretical comparision with imaginary hyperelliptic curves with $k = 4$

To complement our efficiency analysis, we will also make an abstract comparison of our implementation results with those reported in [13], using genus 2 hyperelliptic curves with embedding degree $k = 4$. The implementation results in [13] are amongst the best reported in the literature.

In curves with embedding degree $k = 4$, the underlying prime field needs to be 96 bits larger than our implementation to achieve an equivalent level of security. The representation of each divisor will then need 384 more bits.

The estimated cost of a pairing computation on a degenerate divisor reported in Section 4.9 of [13] is of about $162I_1 + 10375M_1 + 645S_1$ (excluding the cost of the final exponentiation). This estimate is a bit slower than the estimate for hyperelliptic pairings considered in this paper. Although, as mentioned in Remark 2, the use of an algorithm similar to Algorithm 1 to find curve parameters could improve the results of [13]. However, we expect that a R-ate pairing on this curve for general divisors will not be faster than our case.

We can see that pairings on hyperelliptic curves given by a real model are competitive with pairings on curves given by an imaginary model, in terms of bandwidth and computation requirements.

5.3 Implementation results

This section reports some implementation results. The implementation uses the parameters given in Example 1. The timings are obtained using the Magma Online Platform [2].

The following table summarizes the results. The first row shows our implementation results for hyperelliptic curves, and the second row shows our implementation results for elliptic curves.

6 Conclusion

In this article we presented several techniques to speed-up the calculation of pairings on hyperelliptic curves given by a real model. We showed that computing

Table 1. Efficiency Comparision with an AES 80 Security Level

Curve	size of p	Operation Count	time(ms)
$C(\mathbf{F}_p)\ k = 6$	160	$10441M_1$	21.6
$E(\mathbf{F}_{p^2})\ k = 3$	160	$8593M_1$	15.3

pairings on real genus 2 curves is practical. The implementation results are comparable to existing results in the literature for similar settings. We compared the efficiency of two similar elliptic and hyperelliptic curves, and conclude that pairings on elliptic curves with $k = 3$ require 21% less field multiplications than pairings on real hyperelliptic genus 2 curves with $k = 6$. The timing difference in our implementation was that elliptic curves are 28% faster than genus 2 curves.

A Appendix: Addition Formulae

We now present the formulae from [3], which are explicit formulae for the sub-algorithms used in [5] to build an efficient algorithm for divisor arithmetic on hyperelliptic curves with two points at infinity. These formulae require that the curve have model of the form

$$y^2 = x^6 + f_4x^4 + f_3x^3 + f_2x^2 + f_1x + f_0.$$

To make the polynomial monic one takes a random pair of $\mathbf{F}_p$-rational points $(x, \pm y)$ on the curve, moves them to infinity, and absorbs the square root of the leading coefficient into y. Since we are working in large characteristic there is no problem setting $f_5 = 0$.

To be compatible with the divisor representation used in [3] the second polynomial in the Mumford representation is the unique polynomial $v' \equiv v \mod u$ of the form $v' = x^3 + v_1x + v_0$. Notice that v' can be represented only by 2 coefficients even though it has degree 3.

When adding divisors D_1 and D_2, the cubic polynomial $p(x)$ given by equation (1) can be calculated as $p(x) = v_2(x) + u_2(x)s(x)$, where $s(x) = s_1x + s_0$ in Algorithm 2.

The cubic polynomial from equation (1) used in Miller's algorithm when doubling a divisor D is given by $p(x) = v(x) + u(x)s(x)$, where $s(x) = s_1x + s_0$ in Algorithm 3.

Acknowledgments

X. Lin thanks the Chinese Scholarship Council. D. Mireles thanks CONACyT for its financial support.

Algorithm 2 Addition Formulae

```
INPUT: Divisors $D_1 = \mathrm{div}(u_1, v_1)$ and $D_2 = \mathrm{div}(u_2, v_2)$ .
$z_0 = u_{10} - u_{20}, z_1 = u_{11} - u_{21}$.
$z_2 = u_{11} \cdot z_1 - z_0, z_3 = u_{10} \cdot z_1$.
$r = z_1 \cdot z_3 - z_0 \cdot z_2$.
$w_0 = v_{10} - v_{20}, w_1 = v_{11} - v_{21}$.
$s'_1 = w_0 \cdot z_1 - w_1 \cdot z_0, s'_0 = w_0 \cdot z_2 - w_1 \cdot z_3$.
$k_2 = f_4 - 2v_{21}$.
$r_2 = r^2, \hat{w}_0 = r_2 - (s'_1 + r)^2, \hat{w}_1 = (r \cdot \hat{w}^{-1})$.
$\hat{w}_2 = \hat{w}_0 \cdot \hat{w}_1, \hat{w}_3 = r \cdot r_2 \cdot w_1$.
$s_1 = s'_1 \cdot \hat{w}_2, s_0 = s'_0 \hat{w}_2$.
$\tilde{w}_0 s_0 \cdot u_{20}, \tilde{w}_1 = s_1 \cdot u_{21}, l_2 = s_0 + \tilde{w}_1$.
$l_1 = (s_0 + s_1)(u_{21} + u_{20}) - \tilde{w}_1 - \tilde{w}_0, l_0 = \tilde{w}_0$.
$m'_3 = \hat{w}_3 \cdot (-s_1 \cdot (s_0 + l_2) - 2s_0)$.
$m'_2 = \hat{w}_3 \cdot (k_2 - s_1 \cdot (l_1 + 2v_{21}) - s_0 l_2)$.
$u'_1 = m'_3 - u_{11}, u'_0 = m'_2 - u_{10} - u_{11} \cdot u'_1$.
$\underline{w}_1 = u'_1 \cdot (s_1 + 2), \underline{w}_0 = u'_0 \cdot (l_2 - \underline{w}_1)$.
$v'_1 = (u'_0 + u'_1) \cdot (s_1 + -\underline{w}_1 + l_2) - v_{21} - l_1 - \underline{w}_0 - \underline{w}_1$.
$v'_0 = \underline{w}_0 - v_{20} - l_0$.
```

Algorithm 3 Doubling Formulae

```
INPUT: $D = \mathrm{div}(u, v)$ .
$w_1 = u_1^2, \tilde{v}_1 = 2(v_1 + w_1 - u_0), \tilde{v}_0 = 2(v_0 + u_0 \cdot u_1)$.
$w_2 = u_0 \cdot \tilde{v}_1, w_3 = u_1 \cdot \tilde{v}_1$.
$\mathrm{inv}_1 = \tilde{v}_1, \mathrm{inv}_0 = w_3 - \tilde{v}_0$.
$r = \tilde{v}_0 \cdot \mathrm{inv}_0 - w_2 \cdot \tilde{v}_1$.
$k'_2 = f_4 - 2v_1$
$k'_1 = f_3 - 2v_0 - 2k'_2 \cdot u_1$.
$k'_0 = f_2 - v_1^2 - k'_1 \cdot u_1 - k'_2(w_1 + 2u_0)$.
$s'_1 = \mathrm{inv}_1 \cdot k'_0 - \tilde{v}_0 \cdot k'_1, s'_0 = \mathrm{inv}_0 \cdot k'_0 - w_2 \cdot k'_1$.
$r_2 = r^2, \hat{w}_0 = (s'_1 + r)^2 - r_2, \hat{w}_1 = (r \cdot \hat{w}_0)^{-1}$.
$\hat{w}_2 = \hat{w}_0 \cdot \hat{w}_1, \hat{w}_3 = r \cdot r_2 \cdot \hat{w}_1$.
$s_1 = \hat{w}_2 \cdot s'_1, s_0 = \hat{w}_2 \cdot s'_0$.
$u'_1 = 2\hat{w}_3 \cdot ((s_0 - u_1) \cdot s_1 + s_0)$.
$u'_0 = \hat{w}_3 \cdot ((s_0 2u_1) \cdot s_0 + \tilde{v}_1 \cdot s_1 - k'_2)$.
$z_0 = u'_0 - u_0, z_1 = u'_1 - u_1$.
$\underline{w}_0 = z_0 \cdot s_0, \underline{w}_1 = z_1 \cdot s_1$.
$v'_1 = 2u'_0 - v_1 + (s_0 + s_1) \cdot (z_0 + z_1) - \underline{w}_0 - \underline{w}_1 - u'_1 \cdot (2u'_1 + \underline{w}_1)$.
$v'_0 = \underline{w}_0 - v_0 - u'_0 \cdot (2u'_1 + \underline{w}_1)$.
```

References

1. Barreto, P. S. L. M., Kim, H. Y., Lynn, B., and Scott, M. Efficient algorithms for pairing-based cryptosystems. In *CRYPTO* (2002), M. Yung, Ed., vol. 2442 of *Lecture Notes in Computer Science*, Springer, pp. 354–368.
2. Bosma, W., Cannon, J., and Playoust, C. The Magma algebra system. I. The user language. *J. Symbolic Comput. 24*, 3-4 (1997), 235–265. Computational algebra and number theory (London, 1993).
3. Erickson, S., Jacobson, M. J., Shang, N., Shen, S., and Stein, A. Explicit formulas for real hyperelliptic curves of genus 2 in affine representation. In *WAIFI* (2007), C. Carlet and B. Sunar, Eds., vol. 4547 of *Lecture Notes in Computer Science*, Springer, pp. 202–218.
4. Frey, G., and Rück, H.-G. A remark concerning m-divisibility and the discrete logarithm in the divisor class group of curves. *Math. Comp. 62*, 206 (1994), 865–874.
5. Galbraith, S. D., Harrison, M., and Mireles Morales, D. J. Efficient hyperelliptic arithmetic using balanced representation for divisors. *ANTS 2008 procceedings* (2008). to appear.
6. Galbraith, S. D., Pujolas, J., Ritzenthaler, C., and Smith, B. Distortion maps for genus two curves, 2006.
7. Galbraith, S. D., and Verheul, E. R. An analysis of the vector decomposition problem. In *Public Key Cryptography* (2008), R. Cramer, Ed., vol. 4939 of *Lecture Notes in Computer Science*, Springer, pp. 308–327.
8. Granger, R., Hess, F., Oyono, R., Thériault, N., and Vercauteren, F. Ate pairing on hyperelliptic curves. In *EUROCRYPT* (2007), M. Naor, Ed., vol. 4515 of *Lecture Notes in Computer Science*, Springer, pp. 430–447.
9. Hess, F., Smart, N. P., and Vercauteren, F. The eta pairing revisited. *IEEE Transactions on Information Theory 52*, 10 (2006), 4595–4602.
10. Hu, L., Dong, J.-W., and Pei, D. Implementation of cryptosystems based on tate pairing. *J. Comput. Sci. Technol. 20*, 2 (2005), 264–269.
11. Lee, E., Lee, H.-S., and Park, C.-M. Efficient and generalized pairing computation on abelian varieties. Cryptology ePrint Archive, Report 2008/040, 2008. http://eprint.iacr.org/.
12. Lenstra, A. K., and Verheul, E. R. The xtr public key system. In *CRYPTO* (2000), M. Bellare, Ed., vol. 1880 of *Lecture Notes in Computer Science*, Springer, pp. 1–19.
13. O'Eigeartaigh, C., and Scott, M. Pairing calculation on supersingular genus 2 curves. In *Selected Areas in Cryptography* (2006), E. Biham and A. M. Youssef, Eds., vol. 4356 of *Lecture Notes in Computer Science*, Springer, pp. 302–316.
14. Paulus, S., and Rück, H.-G. Real and imaginary quadratic representations of hyperelliptic function fields. *Math. Comp. 68*, 227 (1999), 1233–1241.
15. Verheul, E. R. Evidence that xtr is more secure than supersingular elliptic curve cryptosystems. *J. Cryptology 17*, 4 (2004), 277–296.