Computing optimal pairings on abelian varieties with theta functions

2013/08/08 — Microsoft Research

David Lubicz, **Damien Robert**

Outline

1

Curves, pairings and cryptography

Pairing-based cryptography

Definition

A pairing is a non-degenerate bilinear application $e : G_1 \times G_1 \to G_2$ between finite abelian groups.

Example

- If the pairing e can be computed easily, the difficulty of the DLP in G_1 reduces to the difficulty of the DLP in G_2.

$\Rightarrow$ MOV attacks on supersingular elliptic curves.

- Identity-based cryptography [BF03].
- Short signature [BLS04].
- One way tripartite Diffie–Hellman [Jou04].
- Self-blindable credential certificates [Ver01].
- Attribute based cryptography [SW05].
- Broadcast encryption [Goy+06].

The Weil pairing on elliptic curves

- Let $E : y^2 = x^3 + ax + b$ be an elliptic curve over a field k ($\mathrm{char}\, k \neq 2, 3$, $4a^3 + 27b^2 \neq 0$.)
- Let $P, Q \in E[\ell]$ be points of ℓ-torsion.
- Let f_P be a function associated to the principal divisor $\ell(P) - \ell(0)$, and f_Q to $\ell(Q) - \ell(0)$. We define:

$$e_{W,\ell}(P, Q) = \frac{f_P((Q) - (0))}{f_Q((P) - (0))}.$$

- The application $e_{W,\ell} : E[\ell] \times E[\ell] \to \mu_\ell(\overline{k})$ is a non degenerate pairing: the Weil pairing.

Definition (Embedding degree)

The embedding degree d is the smallest number such that $\ell \mid q^d - 1$; $\mathbb{F}_{q^d}$ is then the smallest extension containing $\mu_\ell(\overline{k})$.

The Tate pairing on elliptic curves over $\mathbb{F}_q$

Definition

The Tate pairing is a non degenerate (on the right) bilinear application given by

$$\begin{array}{rcl} e_T : E_0[\ell] \times E(\mathbb{F}_q)/\ell E(\mathbb{F}_q) & \longrightarrow & \mathbb{F}_{q^d}^* / \mathbb{F}_{q^d}^{*\ \ell} \\ (P,Q) & \longmapsto & f_P((Q)-(0)) \end{array}.$$

where

$$E_0[\ell] = \{P \in E[\ell](\mathbb{F}_{q^d}) \mid \pi(P) = [q]P\}.$$

- On $\mathbb{F}_{q^d}$, the Tate pairing is a non degenerate pairing
$$e_T : E[\ell](\mathbb{F}_{q^d}) \times E(\mathbb{F}_{q^d})/\ell E(\mathbb{F}_{q^d}) \to \mathbb{F}_{q^d}^* / \mathbb{F}_{q^d}^{*\ \ell} \simeq \mu_\ell;$$
- If $\ell^2 \nmid E(\mathbb{F}_{q^d})$ then $E(\mathbb{F}_{q^d})/\ell E(\mathbb{F}_{q^d}) \simeq E[\ell](\mathbb{F}_{q^d})$;
- We normalise the Tate pairing by going to the power of $(q^d-1)/\ell$.
- This final exponentiation allows to save some computations.
  For instance if $d = 2d'$ is even, we can suppose that $Q = (x_2, y_2)$ with $x_2 \in E(\mathbb{F}_{q^{d'}})$. Then the denominators of $\mathfrak{f}_{\lambda,\mu,P}(Q)$ are ℓ-th powers and are killed by the final exponentiation.

Miller's functions

- We need to compute the functions f_P and f_Q. More generally, we define the Miller's functions:

Definition

Let $\lambda \in \mathbb{N}$ and $X \in E[\ell]$, we define $f_{\lambda,X} \in k(E)$ to be a function thus that:

$$(f_{\lambda,X}) = \lambda(X) - ([\lambda]X) - (\lambda - 1)(0).$$

- We want to compute (for instance) $f_{\ell,P}((Q) - (0))$.

Miller's algorithm

- The key idea in Miller's algorithm is that

$$f_{\lambda+\mu,X} = f_{\lambda,X} f_{\mu,X} \mathfrak{f}_{\lambda,\mu,X}$$

where $\mathfrak{f}_{\lambda,\mu,X}$ is a function associated to the divisor

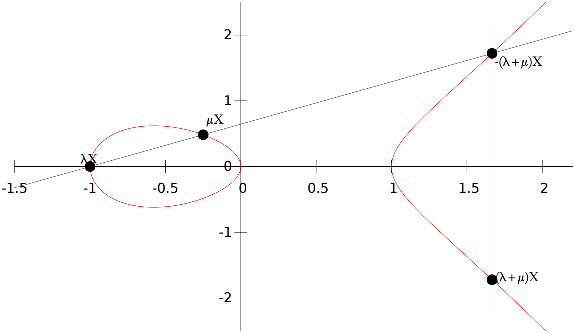
$$([\lambda]X) + ([\mu]X) - ([\lambda+\mu]X) - (0).$$

- We can compute $\mathfrak{f}_{\lambda,\mu,X}$ using the addition law in E: if $[\lambda]X = (x_1, y_1)$ and $[\mu]X = (x_2, y_2)$ and $\alpha = (y_1 - y_2)/(x_1 - x_2)$, we have

$$\mathfrak{f}_{\lambda,\mu,X} = \frac{y - \alpha(x - x_1) - y_1}{x + (x_1 + x_2) - \alpha^2}.$$

Miller's algorithm

$$[\lambda]X = (x_1, y_1) \quad [\mu]X = (x_2, y_2)$$



$$f_{\lambda,\mu,X} = \frac{y - \alpha(x - x_1) - y_1}{x + (x_1 + x_2) - \alpha^2}.$$

Miller's algorithm on elliptic curves

Algorithm (Computing the Tate pairing)

Input: $\ell \in \mathbb{N}$, $P = (x_1, y_1) \in E[\ell](\mathbb{F}_q), Q = (x_2, y_2) \in E(\mathbb{F}_{q^d})$.

Output: $e_T(P, Q)$.

1. *Compute the binary decomposition:* $\ell := \sum_{i=0}^{I} b_i 2^i$. *Let* $T = P, f_1 = 1, f_2 = 1$.
2. *For* i *in* $[I..0]$ *compute*
   - **2.1** α, *the slope of the tangent of* E *at* T.
   - **2.2** $T = 2T$. $T = (x_3, y_3)$.
   - **2.3** $f_1 = f_1^2(y_2 - \alpha(x_2 - x_3) - y_3)$, $f_2 = f_2^2(x_2 + (x_1 + x_3) - \alpha^2)$.
   - **2.4** *If* $b_i = 1$, *then compute*
     - **2.4.1** α, *the slope of the line going through* P *and* T.
     - **2.4.2** $T = T + Q$. $T = (x_3, y_3)$.
     - **2.4.3** $f_1 = f_1^2(y_2 - \alpha(x_2 - x_3) - y_3)$, $f_2 = f_2(x_2 + (x_1 + x_3) - \alpha^2)$.

Return

$$\left(\frac{f_1}{f_2}\right)^{\frac{q^d-1}{\ell}}.$$

Jacobian of curves

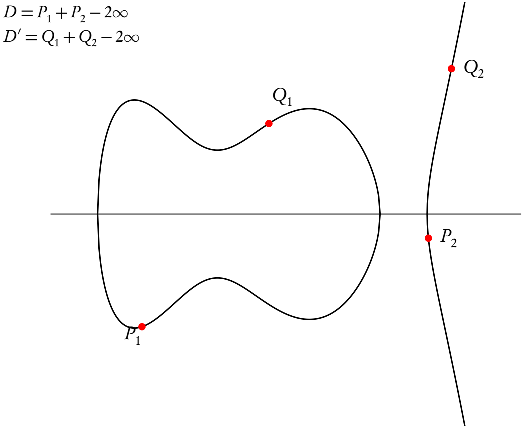
C a smooth irreducible projective curve of genus g.

- Divisor: formal sum $D = \sum n_i P_i$, $\quad P_i \in C(\overline{k})$.
  $\deg D = \sum n_i$.
- Principal divisor: $\sum_{P \in C(\overline{k})} v_P(f).P$; $\quad f \in \overline{k}(C)$.
- Jacobian of C = Divisors of degree 0 modulo principal divisors
  + Galois action
  = Abelian variety of dimension g.
- Divisor class of a divisor $D \in \mathrm{Jac}(C)$ is generically represented by a sum of g points.

Example of Jacobians

DIMENSION 2: Addition law on the Jacobian of an hyperelliptic curve of genus 2:

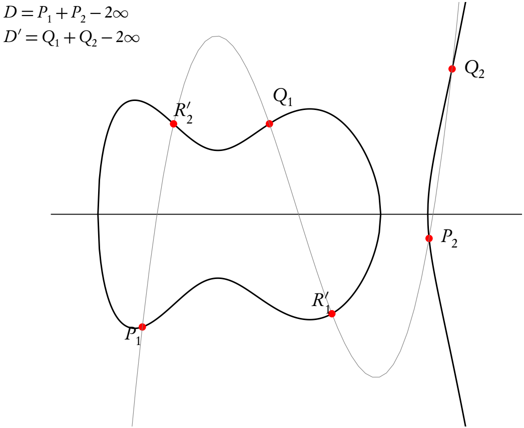
$$y^2 = f(x), \deg f = 5.$$



Example of Jacobians

DIMENSION 2: Addition law on the Jacobian of an hyperelliptic curve of genus 2:

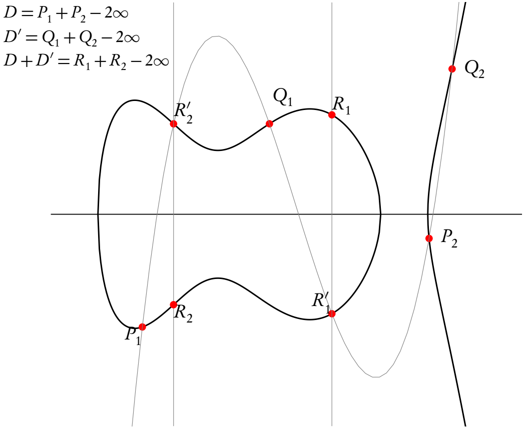
$$y^2 = f(x), \deg f = 5.$$



Example of Jacobians

DIMENSION 2: Addition law on the Jacobian of an hyperelliptic curve of genus 2:

$$y^2 = f(x), \deg f = 5.$$

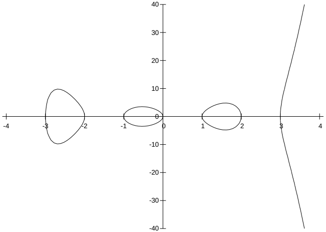


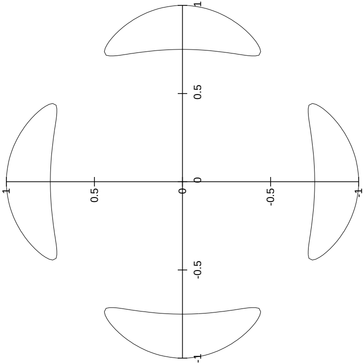
Example of Jacobians

Dimension 3

Jacobians of hyperelliptic curves of genus 3.

Jacobians of quartics.





Pairings on Jacobians

- Let $P \in \mathrm{Jac}(C)[\ell]$ and D_P a divisor on C representing P;
- By definition of $\mathrm{Jac}(C)$, ℓD_P corresponds to a principal divisor (f_P) on C;
- The same formulas as for elliptic curve define the Weil and Tate-Lichtenbaum pairings:

$$e_W(P,Q) = f_P(D_Q)/f_Q(D_P)$$
$$e_T(P,Q) = f_P(D_Q).$$

Pairings on Jacobians

- Let $P \in \mathrm{Jac}(C)[\ell]$ and D_P a divisor on C representing P;
- By definition of $\mathrm{Jac}(C)$, ℓD_P corresponds to a principal divisor (f_P) on C;
- The same formulas as for elliptic curve define the Weil and Tate-Lichtenbaum pairings:

$$e_W(P,Q) = f_P(D_Q)/f_Q(D_P)$$
$$e_T(P,Q) = f_P(D_Q).$$

- A key ingredient for evaluating $f_P(D_Q)$ comes from Weil reciprocity theorem.

Theorem (Weil)

Let D_1 and D_2 be two divisors with disjoint support linearly equivalent to (0) on a smooth curve C. Then

$$f_{D_1}(D_2) = f_{D_2}(D_1).$$

Pairings on Jacobians

- Let $P \in \mathrm{Jac}(C)[\ell]$ and D_P a divisor on C representing P;
- By definition of $\mathrm{Jac}(C)$, ℓD_P corresponds to a principal divisor (f_P) on C;
- The same formulas as for elliptic curve define the Weil and Tate-Lichtenbaum pairings:

$$e_W(P,Q) = f_P(D_Q)/f_Q(D_P)$$
$$e_T(P,Q) = f_P(D_Q).$$

- The extension of Miller's algorithm to Jacobians is “straightforward”;
- For instance if $g = 2$, the function $f_{\lambda,\mu,P}$ is of the form

$$\frac{y - l(x)}{(x - x_1)(x - x_2)}$$

where l is of degree 3.

2

Abelian varieties

Abelian varieties

Definition

An Abelian variety is a complete connected group variety over a base field k.

- Abelian variety = points on a projective space (locus of homogeneous polynomials) + an abelian group law given by rational functions.

Example

- Elliptic curves= Abelian varieties of dimension 1;
- If C is a (smooth) curve of genus g, its Jacobian is an abelian variety of dimension g;
- In dimension $g \geqslant 4$, not every abelian variety is a Jacobian.

Isogenies and pairings

Let $f : A \to B$ be a separable isogeny with kernel K between two abelian varieties defined over k:

$$
\begin{tikzcd}
0 \arrow[r] & K \arrow[r] & A \arrow[r, "f"] & B \arrow[r] & 0 & \\
 & 0 & \hat{A} \arrow[l] & \hat{B} \arrow[l, "\hat{f}"'] & \hat{K} \arrow[l] & 0 \arrow[l]
\end{tikzcd}
$$

- $\hat{K}$ is the Cartier dual of K, and we have a non degenerate pairing $e_f : K \times \hat{K} \to \overline{k}^*$:
    1. If $Q \in \hat{K}(\overline{k})$, Q defines a divisor D_Q on B;
    2. $\hat{f}(Q) = 0$ means that $f^* D_Q$ is equal to a principal divisor (g_Q) on A;
    3. $e_f(P,Q) = g_Q(x)/g_Q(x+P)$. (This last function being constant in its definition domain).
- The Weil pairing $e_{W,\ell}$ is the pairing associated to the isogeny $[\ell] : A \to A$:

$$e_{W,\ell} : A[\ell] \times \hat{A}[\ell] \to \mu_\ell(\overline{k}).$$

Polarisations

If $\mathscr{L}$ is an ample line bundle, the polarisation $\varphi_{\mathscr{L}}$ is a morphism $A \to \widehat{A}, x \mapsto t_x^* \mathscr{L} \otimes \mathscr{L}^{-1}$.

Definition

Let $\mathscr{L}$ be a principal polarization on A. The (polarized) Weil pairing $e_{W,\mathscr{L},\ell}$ is the pairing

$$\begin{array}{rcl} e_{W,\mathscr{L},\ell} : A[\ell] \times A[\ell] & \longrightarrow & \mu_\ell(\overline{k}) \\ (P,Q) & \longmapsto & e_{W,\ell}(P, \varphi_{\mathscr{L}}(Q)) \end{array}.$$

associated to the polarization $\mathscr{L}^\ell$:

$$A \xrightarrow{[\ell]} A \xrightarrow{\mathscr{L}} \hat{A}$$

The Tate pairings on abelian varieties over finite fields

- From the exact sequence

$$0 \to A[\ell](\overline{\mathbb{F}}_{q^d}) \to A(\overline{\mathbb{F}}_{q^d}) \to^{[\ell]} A(\overline{\mathbb{F}}_{q^d}) \to 0$$

  we get from Galois cohomology a connecting morphism

$$\delta : A(\mathbb{F}_{q^d})/\ell A(\mathbb{F}_{q^d}) \to H^1(\mathrm{Gal}(\overline{\mathbb{F}}_{q^d}/\mathbb{F}_{q^d}), A[\ell]);$$

- Composing with the Weil pairing, we get a bilinear application

$$A[\ell](\mathbb{F}_{q^d}) \times A(\mathbb{F}_{q^d})/\ell A(\mathbb{F}_{q^d}) \to H^1(\mathrm{Gal}(\overline{\mathbb{F}}_{q^d}/\mathbb{F}_{q^d}), \mu_\ell) \simeq \mathbb{F}_{q^d}^*/\mathbb{F}_{q^d}^{*\,\ell} \simeq \mu_\ell$$

  where the last isomorphism comes from the Kummer sequence

$$1 \to \mu_\ell \to \overline{\mathbb{F}}_{q^d}^* \to \overline{\mathbb{F}}_{q^d}^* \to 1$$

  and Hilbert 90;
- Explicitely, if $P \in A[\ell](\mathbb{F}_{q^d})$ and $Q \in A(\mathbb{F}_{q^d})$ then the (reduced) Tate pairing is given by

$$e_T(P,Q) = e_W(P, \pi(Q_0) - Q_0)$$

  where Q_0 is any point such that $Q = [\ell]Q_0$ and π is the Frobenius of $\mathbb{F}_{q^d}$.

Cycles and Lang reciprocity

- Let $(A, \mathscr{L})$ be a principally polarized abelian variety;
- To a degree 0 cycle $\sum n_i(P_i)$ on A, we can associate the divisor $\sum t_{P_i}^* \mathscr{L}^{n_i}$ on A;
- The cycle $\sum n_i(P_i)$ corresponds to a trivial divisor iff $\sum n_i P_i = 0$ in A;
- If f is a function on A and $D = \sum (P_i)$ a cycle whose support does not contain a zero or pole of f, we let

$$f(D) = \prod f(P_i)^{n_i}.$$

(In the following, when we write $f(D)$ we will always assume that we are in this situation.)

Theorem ([Lan58])

Let D_1 and D_2 be two cycles equivalent to 0, and f_{D_1} and f_{D_2} be the corresponding functions on A. Then

$$f_{D_1}(D_2) = f_{D_2}(D_1)$$

The Weil and Tate pairings on abelian varieties

Theorem

Let $P, Q \in A[\ell]$. Let D_P and D_Q be two cycles equivalent to $(P) - (0)$ and $(Q) - (0)$. The Weil pairing is given by

$$e_W(P,Q) = \frac{f_{\ell D_P}(D_Q)}{f_{\ell D_Q}(D_P)}.$$

Theorem

Let $P \in A[\ell](\mathbb{F}_{q^d})$ and $Q \in A(\mathbb{F}_{q^d})$, and let D_P and D_Q be two cycles equivalent to $(P) - (0)$ and $(Q) - (0)$. The (non reduced) Tate pairing is given by

$$e_T(P,Q) = f_{\ell D_P}(D_Q).$$

Cryptographic usage of pairings on abelian varieties

- The moduli space of abelian varieties of dimension g is a space of dimension $g(g+1)/2$. We have more liberty to find optimal abelian varieties in function of the security parameters.
- Supersingular abelian varieties can have larger embedding degree than supersingular elliptic curves.
- Over a Jacobian, we can use twists even if they are not coming from twists of the underlying curve.
- If A is an abelian variety of dimension g, $A[\ell]$ is a $(\mathbb{Z}/\ell\mathbb{Z})$-module of dimension $2g \Rightarrow$ the structure of pairings on abelian varieties is richer.

3

Theta functions

Complex abelian varieties

- A complex abelian variety is of the form $A = V/\Lambda$ where $V \simeq \mathbb{C}^g$ is a $\mathbb{C}$-vector space and Λ a lattice, with a polarization (actually an ample line bundle) $\mathscr{L}$ on it;
- The Chern class of $\mathscr{L}$ corresponds to a symplectic real form E on V such that $E(ix, iy) = E(x, y)$ and $E(\Lambda, \Lambda) \subset \mathbb{Z}$;
- The commutator pairing $e_{\mathscr{L}}$ is then given by $\exp(2i\pi E(\cdot, \cdot))$;
- A principal polarization on A corresponds to a decomposition $\Lambda = \Omega\mathbb{Z}^g + \mathbb{Z}^g$ with $\Omega \in \mathfrak{H}_g$ the Siegel space;
- The associated Riemann form on A is then given by $E(\Omega x_1 + x_2, \Omega y_1 + y_2) = {}^t x_1 \cdot y_2 - {}^t y_1 \cdot x_2$.

Theta coordinates

- The theta functions of level n give a system of projective coordinates:

$$\vartheta\left[\begin{smallmatrix} a \\ b \end{smallmatrix}\right](z,\Omega) = \sum_{n\in\mathbb{Z}^g} e^{\pi i\, {}^t(n+a)\Omega(n+a)+2\pi i\, {}^t(n+a)(z+b)} \qquad a,b\in\mathbb{Q}^g$$

- If $n=2$, we get (in the generic case) an embedding of the Kummer variety $A/\pm 1$.

Remark

Working on level n mean we take a n-th power of the principal polarisation. So in the following we will compute the n-th power of the usual Weil and Tate pairings.

The differential addition law ($k = \mathbb{C}$)

$$\Big(\sum_{t \in Z(\overline{2})} \chi(t)\vartheta_{i+t}(x+y)\vartheta_{j+t}(x-y)\Big).\Big(\sum_{t \in Z(\overline{2})} \chi(t)\vartheta_{k+t}(0)\vartheta_{l+t}(0)\Big) =$$

$$\Big(\sum_{t \in Z(\overline{2})} \chi(t)\vartheta_{-i'+t}(y)\vartheta_{j'+t}(y)\Big).\Big(\sum_{t \in Z(\overline{2})} \chi(t)\vartheta_{k'+t}(x)\vartheta_{l'+t}(x)\Big).$$

where $\chi \in \hat{Z}(\overline{2}), i, j, k, l \in Z(\overline{n})$

$$(i', j', k', l') = A(i, j, k, l)$$

$$A = \frac{1}{2}\begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & 1 & -1 \\ 1 & -1 & -1 & 1 \end{pmatrix}$$

Example: differential addition in dimension 1 and in level 2

Algorithm

Input $z_P = (x_0, x_1)$, $z_Q = (y_0, y_1)$ *and* $z_{P-Q} = (z_0, z_1)$ *with* $z_0 z_1 \neq 0$;
$z_0 = (a, b)$ *and* $A = 2(a^2 + b^2)$, $B = 2(a^2 - b^2)$.

Output $z_{P+Q} = (t_0, t_1)$.

1. $t'_0 = (x_0^2 + x_1^2)(y_0^2 + y_2^2)/A$
2. $t'_1 = (x_0^2 - x_1^2)(y_0^2 - y_1^2)/B$
3. $t_0 = (t'_0 + t'_1)/z_0$
4. $t_1 = (t'_0 - t'_1)/z_1$

Return (t_0, t_1)

Cost of the arithmetic with low level theta functions ($\mathrm{char}\, k \neq 2$)

	Montgomery	Level 2	Jacobians coordinates
Doubling	$5M + 4S + 1m_0$	$3M + 6S + 3m_0$	$3M + 5S$
Mixed Addition			$7M + 6S + 1m_0$

Multiplication cost in genus 1 (one step).

	Mumford	Level 2	Level 4
Doubling	$34M + 7S$	$7M + 12S + 9m_0$	$49M + 36S + 27m_0$
Mixed Addition	$37M + 6S$		

Multiplication cost in genus 2 (one step).

4

Pairings with theta functions

Miller functions with theta coordinates

Proposition (Lubicz-R. [LR13])

- *For $P \in A$ we note z_P a lift to $\mathbb{C}^g$. We call P a projective point and z_P an affine point (because we describe them via their projective, resp affine, theta coordinates);*
- *We have (up to a constant)*

$$f_{\lambda,P}(z) = \frac{\vartheta(z)}{\vartheta(z+\lambda z_P)} \left(\frac{\vartheta(z+z_P)}{\vartheta(z)}\right)^{\lambda};$$

- *So (up to a constant)*

$$\mathfrak{f}_{\lambda,\mu,P}(z) = \frac{\vartheta(z+\lambda z_P)\vartheta(z+\mu z_P)}{\vartheta(z)\vartheta(z+(\lambda+\mu)z_P)}.$$

Three way addition

Proposition (Lubicz-R. [LR13])

From the affine points $z_P, z_Q, z_R, z_{P+Q}, z_{P+R}$ and z_{Q+R} one can compute the affine point z_{P+Q+R}.
(In level 2, the proposition is only valid for "generic" points).

Proof.

We can compute the three way addition using a generalised version of Riemann's relations:

$$\Big(\sum_{t\in Z(\overline{2})}\chi(t)\vartheta_{i+t}(z_{P+Q+R})\vartheta_{j+t}(z_P)\Big).\Big(\sum_{t\in Z(\overline{2})}\chi(t)\vartheta_{k+t}(z_Q)\vartheta_{l+t}(z_R)\Big) =$$

$$\Big(\sum_{t\in Z(\overline{2})}\chi(t)\vartheta_{-i'+t}(z_0)\vartheta_{j'+t}(z_{Q+R})\Big).\Big(\sum_{t\in Z(\overline{2})}\chi(t)\vartheta_{k'+t}(z_{P+R})\vartheta_{l'+t}(z_{P+Q})\Big).$$

□

Three way addition in dimension 1 level 2

Algorithm

Input *The points* $x, y, z, X = y + z, Y = x + z, Z = x + y$;

Output $T = x + y + z$.

Return

$$T_0 = \frac{(aX_0 + bX_1)(Y_0Z_0 + Y_1Z_1)}{x_0(y_0z_0 + y_1z_1)} + \frac{(aX_0 - bX_1)(Y_0Z_0 - Y_1Z_1)}{x_0(y_0z_0 - y_1z_1)}$$

$$T_1 = \frac{(aX_0 + bX_1)(Y_0Z_0 + Y_1Z_1)}{x_1(y_0z_0 + y_1z_1)} - \frac{(aX_0 - bX_1)(Y_0Z_0 - Y_1Z_1)}{x_1(y_0z_0 - y_1z_1)}$$

Computing the Miller function $f_{\lambda,\mu,P}((Q)-(0))$

Algorithm

Input *λP, μP and Q;*

Output $f_{\lambda,\mu,P}((Q)-(0))$

1. *Compute $(\lambda+\mu)P$, $Q+\lambda P$, $Q+\mu P$ using normal additions and take any affine lifts $z_{(\lambda+\mu)P}$, $z_{Q+\lambda P}$ and $z_{Q+\mu P}$;*
2. *Use a three way addition to compute $z_{Q+(\lambda+\mu)P}$;*

Return

$$f_{\lambda,\mu,P}((Q)-(0)) = \frac{\vartheta(z_Q+\lambda z_P)\vartheta(z_Q+\mu z_P)}{\vartheta(z_Q)\vartheta(z_Q+(\lambda+\mu)z_P)} \cdot \frac{\vartheta((\lambda+\mu)z_P)\vartheta(z_P)}{\vartheta(\lambda z_P)\vartheta(\mu z_P)}.$$

Lemma

The result does not depend on the choice of affine lifts in Step 2.

- ☺ This allows us to evaluate the Weil and Tate pairings and derived pairings;
- ☹ Not possible *a priori* to apply this algorithm in level 2.

The Tate pairing with Miller's functions and theta coordinates

- Let $P \in A[\ell](\mathbb{F}_{q^d})$ and $Q \in A(\mathbb{F}_{q^d})$; choose any lift z_P, z_Q and z_{P+Q}.
- The algorithm loop over the binary expansion of ℓ, and at each step does a doubling step, and if necessary an addition step.

Given $z_{\lambda P}, z_{\lambda P+Q}$;

Doubling Compute $z_{2\lambda P}, z_{2\lambda P+Q}$ using two differential additions;

Addition Compute $(2\lambda+1)P$ and take an arbitrary lift $z_{(2\lambda+1)P}$. Use a three way addition to compute $z_{(2\lambda+1)P+Q}$.

- At the end we have computed affine points $z_{\ell P}$ and $z_{\ell P+Q}$. Evaluating the Miller function then gives exactly the quotient of the projective factors between $z_{\ell P}$, z_0 and $z_{\ell P+Q}$, z_Q.

☺ Described this way can be extended to level 2 by using compatible additions;

☹ Three way additions and normal (or compatible) additions are quite cumbersome, is there a way to only use differential additions?

The Weil and Tate pairing with theta coordinates (Lubicz-R. [LR10])

Using directly the formula for $f_{\ell,P}(z)$ we get that the Weil and Tate pairings are given by

$$e_{W,\ell}(P,Q) = \frac{\vartheta(z_Q + \ell z_P)\vartheta(0)}{\vartheta(z_Q)\vartheta(\ell z_P)} \frac{\vartheta(z_P)\vartheta(\ell z_Q)}{\vartheta(z_P + \ell z_Q)\vartheta(0)}$$

$$e_{T,\ell}(P,Q) = \frac{\vartheta(z_Q + \ell z_P)\vartheta(0)}{\vartheta(z_Q)\vartheta(\ell z_P)}$$

The Weil and Tate pairing with theta coordinates (Lubicz-R. [LR10])

P and Q points of ℓ-torsion.

$$\begin{array}{ccccc}
z_0 & z_P & 2z_P & \ldots & \ell z_P = \lambda_P^0 z_0 \\
z_Q & z_P \oplus z_Q & 2z_P + z_Q & \ldots & \ell z_P + z_Q = \lambda_P^1 z_Q \\
2z_Q & z_P + 2z_Q & & & \\
\ldots & \ldots & & & \\
\ell Q = \lambda_Q^0 0_A & z_P + \ell z_Q = \lambda_Q^1 z_P & & &
\end{array}$$

- $e_{W,\ell}(P,Q) = \frac{\lambda_P^1 \lambda_Q^0}{\lambda_P^0 \lambda_Q^1}$.
- $e_{T,\ell}(P,Q) = \frac{\lambda_P^1}{\lambda_P^0}$.

Why does it work?

$$\begin{array}{ccccc}
z_0 & \alpha z_P & \alpha^4(2z_P) & \dots & \alpha^{\ell^2}(\ell z_P) = \lambda'^0_P z_0 \\
\beta z_Q & \gamma(z_P \oplus z_Q) & \frac{\gamma^2\alpha^2}{\beta}(2z_P + z_Q) & \dots & \frac{\gamma^\ell \alpha^{\ell(\ell-1)}}{\beta^{\ell-1}}(\ell z_P + z_Q) = \lambda'^1_P \beta z_Q \\
\beta^4(2z_Q) & \frac{\gamma^2\beta^2}{\alpha}(z_P + 2z_Q) & & & \\
\dots & \dots & & & \\
\beta^{\ell^2}(\ell z_Q) = \lambda'^0_Q z_0 & \frac{\gamma^\ell \beta^{\ell(\ell-1)}}{\alpha^{\ell-1}}(z_P + \ell z_Q) = \lambda'^1_Q \alpha z_P & & &
\end{array}$$

We then have

$$\lambda'^0_P = \alpha^{\ell^2}\lambda^0_P, \quad \lambda'^0_Q = \beta^{\ell^2}\lambda^0_Q, \quad \lambda'^1_P = \frac{\gamma^\ell \alpha^{(\ell(\ell-1)}}{\beta^\ell}\lambda^1_P, \quad \lambda'^1_Q = \frac{\gamma^\ell \beta^{(\ell(\ell-1)}}{\alpha^\ell}\lambda^1_Q,$$

$$e'_{W,\ell}(P,Q) = \frac{\lambda'^1_P \lambda'^0_Q}{\lambda'^0_P \lambda'^1_Q} = \frac{\lambda^1_P \lambda^0_Q}{\lambda^0_P \lambda^1_Q} = e_{W,\ell}(P,Q),$$

$$e'_{T,\ell}(P,Q) = \frac{\lambda'^1_P}{\lambda'^0_P} = \frac{\gamma^\ell}{\alpha^\ell \beta^\ell}\frac{\lambda^1_P}{\lambda^0_P} = \frac{\gamma^\ell}{\alpha^\ell \beta^\ell} e_{T,\ell}(P,Q).$$

The case $n = 2$

- If $n = 2$ we work over the Kummer variety K over k, so $e(P,Q) \in \overline{k}^{*,\pm 1}$.
- We represent a class $x \in \overline{k}^{*,\pm 1}$ by $x + 1/x \in \overline{k}^{*}$. We want to compute the symmetric pairing
$$e_s(P,Q) = e(P,Q) + e(-P,Q).$$
- From $\pm P$ and $\pm Q$ we can compute $\{\pm(P+Q), \pm(P-Q)\}$ (need a square root), and from these points the symmetric pairing.
- e_s is compatible with the $\mathbb{Z}$-structure on K and $\overline{k}^{*,\pm 1}$.
- The $\mathbb{Z}$-structure on $\overline{k}^{*,\pm}$ can be computed as follow:

$$(x^{\ell_1+\ell_2} + \frac{1}{x^{\ell_1+\ell_2}}) + (x^{\ell_1-\ell_2} + \frac{1}{x^{\ell_1-\ell_2}}) = (x^{\ell_1} + \frac{1}{x^{\ell_1}})(x^{\ell_2} + \frac{1}{x^{\ell_2}})$$

Ate pairing

- Let $P \in G_2 = A[\ell] \bigcap \mathrm{Ker}(\pi_q - [q])$ and $Q \in G_1 = A[\ell] \bigcap \mathrm{Ker}(\pi_q - 1)$; $\lambda \equiv q \mod \ell$.
- In projective coordinates, we have $\pi_q^d(P+Q) = \lambda^d P + Q = P + Q$;
- Of course, in affine coordinates, $\pi_q^d(z_{P+Q}) \neq \lambda^d z_P + z_Q$.
- But if $\pi_q(z_{P+Q}) = C * (\lambda z_P + z_Q)$, then C is exactly the (non reduced) ate pairing (up to a renormalisation)!

Algorithm (Computing the ate pairing)

Input $P \in G_2, Q \in G_1$;

1. *Compute* $z_Q + \lambda z_P$, λz_P *using differential additions;*
2. *Find the projective factors* C_1 *and* C_0 *such that* $z_Q + \lambda z_P = C_1 * \pi(z_{P+Q})$ *and* $\lambda z_P = C_0 * \pi(z_P)$ *respectively;*

Return $(C_1/C_0)^{\frac{q^d-1}{\ell}}$.

Optimal ate pairing

- Let $\lambda = m\ell = \sum c_i q^i$ be a multiple of ℓ with small coefficients c_i. ($\ell \nmid m$)
- The pairing

$$\begin{array}{rcl} a_\lambda : G_2 \times G_1 & \longrightarrow & \mu_\ell \\ (P, Q) & \longmapsto & \left(\prod_i f_{c_i,P}(Q)^{q^i} \prod_i f_{\sum_{j>i} c_j q^j, c_i q^i, P}(Q) \right)^{(q^d-1)/\ell} \end{array}$$

is non degenerate when $mdq^{d-1} \not\equiv (q^d - 1)/r \sum_i i c_i q^{i-1} \mod \ell$.

- Since $\varphi_d(q) = 0 \mod \ell$ we look at powers $q, q^2, \ldots, q^{\varphi(d)-1}$.
- We can expect to find λ such that $c_i \approx \ell^{1/\varphi(d)}$.

Optimal ate pairing with theta functions

Algorithm (Computing the optimal ate pairing)

Input $\pi_q(P) = [q]P$, $\pi_q(Q) = Q$, $\lambda = m\ell = \sum c_i q^i$;

1. *Compute the* $z_Q + c_i z_P$ *and* $c_i z_P$;
2. *Apply Frobeniuses to obtain the* $z_Q + c_i q^i z_P$, $c_i q^i z_P$;
3. *Compute* $c_i q^i z_P \oplus \sum_j c_j q^j z_P$ *(up to a constant) and then do a three way addition to compute* $z_Q + c_i q^i z_P + \sum_j c_j q^j z_P$ *(up to the same constant);*
4. *Recurse until we get* $\lambda z_P = C_0 * z_P$ *and* $z_Q + \lambda z_P = C_1 * z_Q$;

Return $(C_1/C_0)^{\frac{q^d-1}{\ell}}$.

The case $n = 2$

- Computing $c_i q^i z_P \pm \sum_j c_j q^j z_P$ requires a square root (very costly);
- And we need to recognize $c_i q^i z_P + \sum_j c_j q^j z_P$ from $c_i q^i z_P - \sum_j c_j q^j z_P$.
- We will use compatible additions: if we know x, y, z and $x+z$, $y+z$, we can compute $x+y$ without a square root;
- We apply the compatible additions with $x = c_i q^i z_P$, $y = \sum_j c_j q^j z_P$ and $z = z_Q$.

Compatible additions

- Recall that we know x, y, z and $x+z$, $y+z$;
- From it we can compute $(x+z)\pm(y+z)=\{x+y+2z, x-y\}$ and of course $\{x+y, x-y\}$;
- Then $x+y$ is the element in $\{x+y, x-y\}$ not appearing in the preceding set;
- Since $x-y$ is a common point, we can recover it without computing a square root.

The compatible addition algorithm in dimension 1

Algorithm

Input $x, y, Y = x + z, X = y + z$;

1. *Computing* $x \pm y$:

$$\alpha = (x_0^2 + x_1^2)(y_0^2 + y_1^2)/A$$
$$\beta = (x_0^2 - x_1^2)(y_0^2 - y_1^2)/B$$
$$\kappa_{00} = (\alpha + \beta), \kappa_{11} = (\alpha - \beta)$$
$$\kappa_{10} := x_0 x_1 y_0 y_1 / ab$$

2. *Computing* $(x + z) \pm (y + z)$:

$$\alpha' = (Y_0^2 + Y_1^2)(X_0^2 + X_1^2)/A$$
$$\beta' = (Y_0^2 - Y_1^2)(X_0^2 - X_1^2)/B$$
$$\kappa'_{00} = \alpha' + \beta', \kappa'_{11} = \alpha' - \beta'$$
$$\kappa'_{10} = Y_1 Y_2 X_1 X_2 / ab$$

Return $x + y = [\kappa_{00}(\kappa_{10}\kappa'_{00} - \kappa'_{10}\kappa_{00}), \kappa_{10}(\kappa_{10}\kappa'_{00} - \kappa'_{10}\kappa_{00}) + \kappa_{00}(\kappa_{11}\kappa'_{00} - \kappa'_{11}\kappa_{00})]$

5

Performance

One step of the pairing computation

Algorithm (A step of the Miller loop with differential additions)

Input $nP = (x_n, z_n)$; $(n+1)P = (x_{n+1}, z_{n+1})$, $(n+1)P + Q = (x'_{n+1}, z'_{n+1})$.

Output $2nP = (x_{2n}, z_{2n})$; $(2n+1)P = (x_{2n+1}, z_{2n+1})$; $(2n+1)P + Q = (x'_{2n+1}, z'_{2n+1})$.

1. $\alpha = (x_n^2 + z_n^2)$; $\beta = \frac{A}{B}(x_n^2 - z_n^2)$.
2. $X_n = \alpha^2$; $X_{n+1} = \alpha(x_{n+1}^2 + z_{n+1}^2)$; $X'_{n+1} = \alpha({x'}^2_{n+1} + {z'}^2_{n+1})$;
3. $Z_n = \beta(x_n^2 - z_n^2)$; $Z_{n+1} = \beta(x_{n+1}^2 - z_{n+1}^2)$; $Z'_{n+1} = \beta({x'}^2_{n+1} + {z'}^2_{n+1})$;
4. $x_{2n} = X_n + Z_n$; $x_{2n+1} = (X_{n+1} + Z_{n+1})/x_P$; $x'_{2n+1} = (X'_{n+1} + Z'_{n+1})/x_Q$;
5. $z_{2n} = \frac{a}{b}(X_n - Z_n)$; $z_{2n+1} = (X_{n+1} - Z_{n+1})/z_p$; $z'_{2n+1} = (X'_{n+1} - Z'_{n+1})/z_Q$;

Return (x_{2n}, z_{2n}); (x_{2n+1}, z_{2n+1}); (x'_{2n+1}, z'_{2n+1}).

Weil and Tate pairing over $\mathbb{F}_{q^d}$

$g = 1$	$4\mathbf{M} + 2\mathbf{m} + 8\mathbf{S} + 3\mathrm{m}_0$
$g = 2$	$8\mathbf{M} + 6\mathbf{m} + 16\mathbf{S} + 9\mathrm{m}_0$

Tate pairing with theta coordinates, $P, Q \in A[\ell](\mathbb{F}_{q^d})$ (one step)

Operations in $\mathbb{F}_q$: M: multiplication, S: square, m multiplication by a coordinate of P or Q, m_0 multiplication by a theta constant;
Mixed operations in $\mathbb{F}_q$ and $\mathbb{F}_{q^d}$: M, m and m_0;
Operations in $\mathbb{F}_{q^d}$: $\mathbf{M}$, $\mathbf{m}$ and $\mathbf{S}$.

Remark

- *Doubling step for a Miller loop with Edwards coordinates:* $9\mathbf{M} + 7\mathbf{S} + 2\mathrm{m}_0$*;*
- *Just doubling a point in Mumford projective coordinates using the fastest algorithm [Lan05]:* $33\mathbf{M} + 7\mathbf{S} + 1\mathrm{m}_0$*;*
- *Asymptotically the final exponentiation is more expensive than Miller's loop, so the Weil's pairing is faster than the Tate's pairing!*

Tate pairing

$g=1$	$1\mathbf{m}+2\mathbf{S}+2\mathrm{M}+2M+1m+6S+3m_0$
$g=2$	$3\mathbf{m}+4\mathbf{S}+4\mathrm{M}+4M+3m+12S+9m_0$

Tate pairing with theta coordinates, $P \in A[\ell](\mathbb{F}_q), Q \in A[\ell](\mathbb{F}_{q^d})$ (one step)

		Miller		Theta coordinates
		Doubling	Addition	One step
$g=1$	d even	$1\mathbf{M}+1\mathbf{S}+1\mathrm{M}$	$1\mathbf{M}+1\mathrm{M}$	$1\mathbf{M}+2\mathbf{S}+2\mathrm{M}$
	d odd	$2\mathbf{M}+2\mathbf{S}+1\mathrm{M}$	$2\mathbf{M}+1\mathrm{M}$	
$g=2$	Q degenerate + d even	$1\mathbf{M}+1\mathbf{S}+3\mathrm{M}$	$1\mathbf{M}+3\mathrm{M}$	$3\mathbf{M}+4\mathbf{S}+4\mathrm{M}$
	General case	$2\mathbf{M}+2\mathbf{S}+18\mathrm{M}$	$2\mathbf{M}+18\mathrm{M}$	

$P \in A[\ell](\mathbb{F}_q)$, $Q \in A[\ell](\mathbb{F}_{q^d})$ (counting only operations in $\mathbb{F}_{q^d}$).

Ate and optimal ate pairings

$g=1$	$4\mathbf{M}+1\mathbf{m}+8\mathbf{S}+1\mathrm{m}+3\mathrm{m}_0$
$g=2$	$8\mathbf{M}+3\mathbf{m}+16\mathbf{S}+3\mathrm{m}+9\mathrm{m}_0$

Ate pairing with theta coordinates, $P \in G_2, Q \in G_1$ (one step)

Remark

Using affine Mumford coordinates in dimension 2, the hyperelliptic ate pairing costs [Gra+07]:

Doubling $1\mathbf{I}+29\mathbf{M}+9\mathbf{S}+7\mathrm{M}$

Addition $1\mathbf{I}+29\mathbf{M}+5\mathbf{S}+7\mathrm{M}$

(where $\mathbf{I}$ denotes the cost of an affine inversion in $\mathbb{F}_{q^d}$).

D. Boneh and M. Franklin. “Identity-based encryption from the Weil pairing”. In: *SIAM Journal on Computing* 32.3 (2003), pp. 586–615 (cit. on p. 4).

D. Boneh, B. Lynn, and H. Shacham. “Short signatures from the Weil pairing”. In: *Journal of Cryptology* 17.4 (2004), pp. 297–319 (cit. on p. 4).

V. Goyal, O. Pandey, A. Sahai, and B. Waters. “Attribute-based encryption for fine-grained access control of encrypted data”. In: *Proceedings of the 13th ACM conference on Computer and communications security*. ACM. 2006, p. 98 (cit. on p. 4).

R. Granger, F. Hess, R. Oyono, N. Thériault, and F. Vercauteren. “Ate pairing on hyperelliptic curves”. In: *Advances in cryptology—EUROCRYPT 2007*. Vol. 4515. Lecture Notes in Comput. Sci. Berlin: Springer, 2007, pp. 430–447 (cit. on p. 53).

A. Joux. “A one round protocol for tripartite Diffie–Hellman”. In: *Journal of Cryptology* 17.4 (2004), pp. 263–276 (cit. on p. 4).

S. Lang. “Reciprocity and Correspondences”. In: *American Journal of Mathematics* 80.2 (1958), pp. 431–440 (cit. on p. 24).

T. Lange. “Formulae for arithmetic on genus 2 hyperelliptic curves”. In: *Applicable Algebra in Engineering, Communication and Computing* 15.5 (2005), pp. 295–328 (cit. on p. 51).

D. Lubicz and D. Robert. “Efficient pairing computation with theta functions”. In: *Algorithmic Number Theory*. Lecture Notes in Comput. Sci. 6197 (July 2010). Ed. by G. Hanrot, F. Morain, and E. Thomé. 9th International Symposium, Nancy, France, ANTS-IX, July 19-23, 2010, Proceedings. DOI: 10.1007/978-3-642-14518-6_21. URL: http://www.normalesup.org/~robert/pro/publications/articles/pairings.pdf. Slides http://www.normalesup.org/~robert/publications/slides/2010-07-ants.pdf (cit. on pp. 39, 40).

D. Lubicz and D. Robert. “A generalisation of Miller’s algorithm and applications to pairing computations on abelian varieties”. Mar. 2013. URL: http://www.normalesup.org/~robert/pro/publications/articles/optimal.pdf. HAL: hal-00806923, eprint: 2013/192 (cit. on pp. 34, 35).

A. Sahai and B. Waters. “Fuzzy identity-based encryption”. In: *Advances in Cryptology–EUROCRYPT 2005* (2005), pp. 457–473 (cit. on p. 4).

E. Verheul. “Self-blindable credential certificates from the Weil pairing”. In: *Advances in Cryptology—ASIACRYPT 2001* (2001), pp. 533–551 (cit. on p. 4).