Pairing Computation on Jacobi's Elliptic Curve

Sylvain Duquesne

University of Rennes 1

Symposium on Number Theory
Warwick, 2 July 2013

Joint work with E. Fouotsa and N. ElMrabet

Institut de recherche mathématique de Rennes

IRMAR - UMR 6625 du CNRS

Pairings in cryptography

Pairings are bilinear maps from $(G_1, +) \times (G_2, +)$ to $(G_3, \times)$

Destructive use (mid 90's)

- Transfer of discrete log from G_1 to G_3
- Decisional Diffie-Hellman is easy

Constructive use (since 2000)

- Short signatures
- ID-based cryptography
- Broadcast encryption
- ...

Such bilinear maps are available on elliptic curves

Realization of pairings

Context

- E elliptic curve defined over $\mathbb{F}_p$ (p prime) with neutral element P_∞.
- $P \in E(\mathbb{F}_p)$ of prime order r.
- k the embedding degree (smallest integer such that $r|p^k - 1$).
- $Q \in E(\mathbb{F}_{p^k})$ of order r

Let f_P be the function on the curve such that $\mathrm{Div}(f_P) = rP - rP_\infty$.

$$e(P, Q) = f_P(Q)^{\frac{p^k-1}{r}} \in \mathbb{F}_{p^k}$$

Examples

- Supersingular curves ($k \leq 2$ in large characteristic)
- MNT curves ($k = 6$), optimal for 80 bits security
- Barreto-Naherig curves ($k = 12$), optimal for 128 bits security
- Other ordinary curves with prescribed embedding degrees

Basic block for the computation of f_P

Let $f_{i,P}$ s.t. $\mathrm{Div}(f_{i,P}) = iP - [i]P - (i-1)P_\infty$. We have

$$f_{i+j,P} = f_{i,P} f_{j,P} h_{[i]P,[j]P}$$

where $h_{R,S}$ is the rational function involved in the sum U of R and S

$$\mathrm{Div}(h_{R,S}) = R + S - U - P_\infty$$

Example

In the case of Weierstrass elliptic curves, $h_{R,S} = \frac{\ell_{R,S}}{v_U}$ where $\ell_{R,S}$ is the line passing trough R and S and v_U is the vertical line passing by U

As a consequence, $f_P (= f_{r,P})$ can be computed via any addition chain

Tate pairing computation

The Miller loop (computation of $f_P(Q)$)

- $T \leftarrow P, f \leftarrow 1$
- for each bit of r do
  $f \leftarrow f^2.h_{T,T}(Q)$ and $T \leftarrow 2T$
  if the bit is 1 do $f \leftarrow f.h_{T,P}(Q)$ and $T \leftarrow T + P$

where $h_{R,S}$ is the function involved in the sum of R and S.

The final exponentiation (computation of $f^{\frac{p^k-1}{r}}$)

Split in an easy part (use of Frobenius) and a difficult part.
Difficult part is roughly f^s with $s \approx p$ and even $p^{\frac{1}{2}}$ (MNT) or $p^{\frac{3}{4}}$ (BN).

Using twists

A twist $\tilde{E}$ of degree d of a curve $E/\mathbb{F}_q$ is isomorphic to E over $\mathbb{F}_{q^d}$.

$\rightarrow$ variant of the Tate pairing with $G_2 = \tilde{E}(\mathbb{F}_{p^{k/d}})$.

In practice : isomorphism between E and $\tilde{E} \Rightarrow$ special form for Q in the classical Tate pairing definition.

Consequences

- Work on smaller fields ($\mathbb{F}_{p^{k/d}}$)
- Elimination of subfield factors thanks to the final exponentiation

Example of quadratic twist

If ν is not a square in $\mathbb{F}_{p^{k/2}}$, we have the twisted curves

$$E : y^2 = x^3 + ax + b \qquad \tilde{E} : \nu y^2 = x^3 + ax + b$$

The isomorphism from $\tilde{E}$ to E is $\varphi((x, y)) = (x, y\sqrt{\nu})$

$\rightarrow Q = (x, y\sqrt{\nu})$ with $x, y \in \mathbb{F}_{p^{k/2}}$

Remark : the degree d can only be 2, 3, 4 or 6.

Alternatives to the Weierstrass model

Introduced in cryptography for

- efficiency reasons
- security reasons

Alternatives models

- Montgomery form $by^2 = x^3 + ax^2 + x$
- Hessian form $x^3 + y^3 + 1 = cxy$
- Jacobi form $y^2 = dx^4 + 2\mu x^2 + 1$
- Edwards form $u^2 + v^2 = c^2(1 + du^2v^2)$
- Huff form $ax(y^2 - 1) = by(x^2 - 1)$

Drawbacks

- 2 or 3 rational torsion
- only twists of degree 2 in certain cases

Jacobi quartic curves

Defined by equation of the form

$$E_{d,\mu} : y^2 = dx^4 + 2\mu x^2 + 1$$

$E_{d,\mu}$ has a rational point of order 2

Group law

- The neutral element is $O = (0, 1)$
- The opposite of (x_1, y_1) is $(-x_1, y_1)$
- The sum of (x_1, y_1) and (x_2, y_2) is given by

$$\left(\frac{x_1^2 - x_2^2}{x_1 y_2 - y_1 x_2}, \frac{(x_1 - x_2)^2}{(x_1 y_2 - y1 x_2)^2} (y_1 y_2 + 1 + dx_1^2 x_2^2) - 1 \right)$$

- The doubling of (x_1, y_1) is given by

$$\left(\frac{2y_1}{2 - y_1^2} x_1, \frac{2y_1}{2 - y_1^2} \left(\frac{2y_1}{2 - y_1^2} - y1 \right) - 1 \right)$$

functions involved in the group law

Opposite

$-P_1 = (-x_1, y_1)$ but the function h_{-P_1,P_1} involved is not $y - y_1$.

$$\mathrm{Div}\left(\frac{c}{l_0^2}\right) = P + (-P) - 2O$$

where c is the conic passing through P and $O' = (0, -1)$ (2 times) and l_0 is the line passing through O and O' ($l_0 = x$)

Addition

The function h_{P_1,P_2} involved in $P_1 + P_2 = P_3$ is given by

$$\mathrm{Div}\left(\frac{C_{P_1,P_2}}{h_{-P_3,P_3} l_0^3}\right) = P_1 + P_2 - P_3 - O$$

where C_{P_1,P_2} is the cubic passing through P_1, P_2 and O' (3 times).
Same idea for doubling.

Formulas for these functions are obtained by solving systems

Twist of Jacobi quartic curves

Assuming k is divisible by 4, $E_{d,\mu}$ has a twist of order 4 iff $\mu = 0$.
It is defined over $\mathbb{F}_{p^{k/4}}$ by

$$\tilde{E_{d,0}} : y^2 = d\omega^4 x^4 + 1$$

where $\{1, \omega, \omega^2, \omega^3\}$ is a basis of $\mathbb{F}_{p^k}/\mathbb{F}_{p^{k/4}}$.
The isomorphism between $\tilde{E_{d,0}}$ and $E_{d,0}$ is $\varphi(x, y) = (x\omega, y)$

Consequence

The second input of the Tate pairing can be chosen in the form $(x_Q\omega, y_Q)$ with $x_Q, y_Q \in \mathbb{F}_{p^{k/4}}$

$\Rightarrow$ All the factors involving only x_Q, y_Q, P, ω^2 are cancelled by the final exponentiation

This is the case for $l_0^2, h_{P,-P}(Q)$ and other terms involved in the cubic equation defining the group law

Doubling step of Miller algorithm

$$h'_{T,T}(Q) = B\left(\frac{y_Q+1}{x_Q^2\omega^4}\right)\omega^2 + D\left(\frac{y_Q+1}{x_Q^3\omega^4}\right)\omega + A$$

- h' is h up to subfield factors
- $\left(\frac{y_Q+1}{x_Q^2\omega^4}\right)$ and $\left(\frac{y_Q+1}{x_Q^3\omega^4}\right)$ precomputed in $\mathbb{F}_{p^4}$
- A, B and $D \in \mathbb{F}_p$ are quantities involved in the classical doubling of T

$$\begin{aligned} A &= Y(Y+Z^2) \\ B &= -X^2(Y+2Z^2) \\ D &= 2X^3Z \end{aligned}$$

Remarks

- We use the coordinates (X, Y, Z, X^2, Z^2) with $x = X/Z, y = Y/Z^2$
- No term in ω^3 and constant term in $\mathbb{F}_p \Rightarrow f^2.h'_{T,T}(Q)$ is faster
- Only A, B and D are different for the addition step

Comparison with previous results for the doubling step

$k = 8$

- With schoolbook arithmetic for $\mathbb{F}_{p^8}$

Method	Weierstrass 2010	Jacobi 2011	This work
Mult in $\mathbb{F}_p$	79	79	59

- With Karatsuba arithmetic for $\mathbb{F}_{p^8}$

Method	Weierstrass 2010	Jacobi 2011	This work
Mult in $\mathbb{F}_p$	42	42	37

$k = 16$

- With schoolbook arithmetic for $\mathbb{F}_{p^{16}}$

Method	Weierstrass 2010	Jacobi 2011	This work
Mult in $\mathbb{F}_p$	271	275	163

- With Karatsuba arithmetic for $\mathbb{F}_{p^{16}}$

Method	Weierstrass 2010	Jacobi 2011	This work
Mult in $\mathbb{F}_p$	100	100	81

The Ate pairing

Let π_p be the Frobenius map on the curve : $\pi_p(x,y) = (x^p, y^p)$.
π_p has trace t and its eigenvalues are 1 and p.
$\rightarrow$ choose the proper spaces as G_1 and G_2

The Ate pairing and its variants

$$e_A(P,Q) = f_{t-1,Q}(P)^{\frac{p^k-1}{r}}$$

is a pairing (in fact a power of the Tate pairing)

- The trace t is twice shorter than r
- The role of P and Q are swapped : arithmetic on the elliptic curve is performed over extension field
- Using twists allows Q to have a special form and then to work on subfields (less expensive, discard subfield factors)
- Can be generalized to obtain smaller loop length (optimal pairing)

Computing the (optimal-)Ate pairing for Jacobi curves

The formulas must be rewriting assumming

- The point T is in $\mathbb{F}_{p^k}$ but has the form $(X\omega, Y, Z)$ with $X, Y, Z \in \mathbb{F}_{p^{k/4}}$
- The function are evaluated in $P = (x_P, y_P) \in E(\mathbb{F}_p)$
- All the factors lying in a proper subfield of $\mathbb{F}_{p^k}$ can be discarded

We obtain $h'_{T,T}(P) = B\left(\frac{y_P + 1}{x_P^2}\right)\omega^3 + A\omega + D\omega^4\left(\frac{y_P + 1}{x_P^3}\right)$

Remarks

- A, B and D are the same as for the Tate pairing (but $\in \mathbb{F}_{p^{k/4}}$)
- No term in ω^2
- Same for addition

The situation is very similar to the Tate pairing

Comparison with Weierstrass form for the doubling step

$k = 8$

- With schoolbook arithmetic for $\mathbb{F}_{p^8}$

Method	Weierstrass 2010	Jacobi 2011	This work
Mult in $\mathbb{F}_p$	101	-	85

- With Karatsuba arithmetic for $\mathbb{F}_{p^8}$

Method	Weierstrass 2010	Jacobi 2011	This work
Mult in $\mathbb{F}_p$	62	-	59

$k = 16$

- With schoolbook arithmetic for $\mathbb{F}_{p^{16}}$

Method	Weierstrass 2010	Jacobi 2011	This work
Mult in $\mathbb{F}_p$	377	-	313

- With Karatsuba arithmetic for $\mathbb{F}_{p^{16}}$

Method	Weierstrass 2010	Jacobi 2011	This work
Mult in $\mathbb{F}_p$	180	-	171

Working example

- A curve with embedding degree 8 can be obtained via Brezing-Weng like method.

$$\begin{aligned} x &= 240000000000010394 \\ r &= 82x^4 + 108x^3 + 54x^2 + 12x + 1 \\ p &= 379906x^6 + \ldots \end{aligned}$$

- An optimal pairing is obtained using Vercauteren lattice based method

$$e_o(Q, P) = \left(f_{x,Q}^{3p^3+1}(P).h\right)^{\frac{p^8-1}{r}}$$

where h is the product of 3 functions of the form $h_{R,S}$

- No timing but the result is bilinear ;-)

Conclusion

- We obtained the best complexities to date for curves with twists of order 4
- A careful implementation is missing to provide timings
- To have more interest for reasonable security levels (say 96-110 bits), it would be very useful to find prime curves with $k = 8$ (at least $\log(r) \equiv \log(p)$)
- Adapt other improvements known for BN curves (clever factorisation of $\frac{p^8-1}{r}$, fast formulas for squaring during the final exponentiation, ...)

Conclusion

- We obtained the best complexities to date for curves with twists of order 4
- A careful implementation is missing to provide timings
- To have more interest for reasonable security levels (say 96-110 bits), it would be very useful to find prime curves with $k = 8$ (at least $\log(r) \equiv \log(p)$)
- Adapt other improvements known for BN curves (clever factorisation of $\frac{p^8-1}{r}$, fast formulas for squaring during the final exponentiation, ...)

Thank you for your attention