Square Root in Even-Degree Extension Fields

Gora Adj

(Joint work with Francisco Rodriguez Henriquez)

Cinvestav

ECC 2012 Rump session - October 30th 2012

A New Algorithm

Few months ago, we came out with an algorithm which computes a square root of an quadratic residue in the field $\mathbb{F}_{q^2}$, where $q = p^n$ is a power of a prime number.

We saw that this algorithm was faster than every other algorithm that we knew for computing a square root in a even extension field.

Our Algorithm

This algorithm is based on the following equation:

Given a quadratic residue element a in $\mathbb{F}_{q^2}$, we have:

$$\sqrt{a} = \begin{cases} \pm \left(a^{\frac{q-1}{4}}\right)^{q} \sqrt{a^{\frac{q+1}{2}}} & \text{if } a^{\frac{q^2-1}{4}} = 1, \\ \pm \left(a^{\frac{q-1}{4}}\right)^{q} \left(c^{\frac{q+1}{2}}\right)^{-1} \sqrt{(ac^2)^{\frac{q+1}{2}}} & \text{otherwise,} \end{cases} \tag{1}$$

where c is a quadratic non-residue element in $\mathbb{F}_{q^2}$.

Our Desappointment

After months passed on implementing the algorithm for speed comparaisons and writing a paper for publishing our findings, we unfortunately discovered that there was another algorithm, rarely referenced in the open literature, which was in many cases (the most usual cases) faster than ours.

This algorithm, called complex algorithm, was given in:
Michael Scott. Implementing cryptographic pairings over barreto-naehrig curves. *Paring 2007*.

The Complex Algorithm

$\mathbb{F}_{q^2} \cong \mathbb{F}_p[y]/\left(y^2 - \beta\right)$, where $\beta \in \mathbb{F}_{p^2}$ is quadratic non-residue.

Let $a = a_0 + a_1 y \in \mathbb{F}_{q^2}^*$ be an arbitrary quadratic residue and

$x = x_0 + x_1 y \in \mathbb{F}_{q^2}$ such that $x^2 = a$, then we should have

$$a = {x_0}^2 + 2x_0 x_1 y + \beta {x_1}^2,$$

so that x_0 and x_1 must satisfy the following two equations

$$\begin{cases} {x_0}^2 + \beta {x_1}^2 = & a_0 \\ 2x_0 x_1 = & a_1 \end{cases}$$

The Complex Algorithm

Solving the system of these two equations for x_0 and x_1 gives

$$\begin{aligned} x_0 &= \left(\frac{a_0 \pm \left({a_0}^2 - \beta {a_1}^2\right)^{\frac{1}{2}}}{2}\right)^{\frac{1}{2}}, \qquad (2) \\ x_1 &= \frac{a_1}{2x_0}. \end{aligned}$$

We can see that the algorithm computes a square root of a quadratic residue in $\mathbb{F}_{q^2}$ with mostly two square roots and an inversion in $\mathbb{F}_q$, what is very efficient in this configuration.

SO, PLEASE USE
THE COMPLEX ALGORITHM!

MUCHAS GRACIAS!