A note on efficient computation of cube roots in characteristic 3

Paulo S. L. M. Barreto[1]

Escola Politécnica, Universidade de São Paulo.
Av. Prof. Luciano Gualberto, tr. 3, 158.
BR 05508-900, São Paulo(SP), Brazil.
pbarreto@larc.usp.br

Abstract. The cost of the folklore algorithm for computing cube roots in $\mathbb{F}_{3^m}$ in standard polynomial basis is less that one multiplication, but still $O(m^2)$. Here we show that, if $\mathbb{F}_{3^m}$ is represented in trinomial basis as $\mathbb{F}_3[x]/(x^m + ax^k + b)$ with $a, b = \pm 1$, the actual cost of computing cube roots in $\mathbb{F}_{3^m}$ is only $O(m)$.

1 Introduction

The fastest method known for computing the Tate pairing on supersingular elliptic curves over $\mathbb{F}_{3^m}$ is the Duursma-Lee algorithm [1], which involves the computation of cube roots in that finite field. An efficient foklore algorithm for cube root extraction in standard polynomial basis (described e.g. in [3, section 5.2], whereby taking roots reduces to two multiplications by constant field elements. Since the degree of one of the involved factors in each multiplication is at most $m/3$, the overall cost of the "folklore" algorithm is actually about two thirds of one $\mathbb{F}_{3^m}$ multiplication.

While this algorithm improves upon other methods like solving a linear equation, it still takes $O(m^2)$ operations. It turns out, however, that further improvements are possible when $\mathbb{F}_{3^m}$ is represented as $\mathbb{F}_3[x]/(x^m + ax^k + b)$ with $k \equiv m \pmod 3$. Specifically, we will show that the *other* factors involved in the multiplications are a binomial and a trinomial, so that the incurred cost is only five shifts. Better yet, the degrees and the shift amounts are such that no reduction is needed (that is, all terms resulting from these operations have degrees smaller than m).

A similar technique has been devised to take square roots in characteristic 2 by Fong *et al.* [2]. Clearly, it also generalizes to higher characteristic p, although the number of relevant cases allowed by the condition $k \equiv m \pmod p$ becomes quite restricted.

2 The folklore algorithm

We assume throughout this note that m is prime and that $\mathbb{F}_{3^m}$ is represented as $\mathbb{F}_3[x]/(x^m + ax^k + b)$, with $a, b = \pm 1$. As we will see, the folklore algorithm involves multiplications by $x^{1/3}$ and $x^{2/3}$. The other factor in these products has degree not exceeding $m/3$.

For any $c \in \mathbb{F}_{3^m}$, we denote by $c^{\ll n}$ the value cx^n, which is c left-shifted by n positions and suitably reduced when necessary.

Let $r = \sum_{i=0}^{m-1} r_i x^i \in \mathbb{F}_{3^m}$. The goal is to compute $\sqrt[3]{r}$.

2.1 Case $m \equiv 1 \pmod 3$

Let $m = 3u + 1$. The folklore algorithm reads:

$$
\begin{aligned}
r &= \sum_{i=0}^{m-1} r_i x^i \\
&= \sum_{i=0}^{u} r_{3i} x^{3i} + x \cdot \sum_{i=0}^{u-1} r_{3i+1} x^{3i} + x^2 \cdot \sum_{i=0}^{u-1} r_{3i+2} x^{3i}. \\
\therefore \sqrt[3]{r} &= \sum_{i=0}^{u} r_{3i} x^{i} + x^{1/3} \cdot \sum_{i=0}^{u-1} r_{3i+1} x^{i} + x^{2/3} \cdot \sum_{i=0}^{u-1} r_{3i+2} x^{i}.
\end{aligned}
$$

Defining $c_0 = \sum_{i=0}^{u} r_{3i} x^i$, $c_1 = \sum_{i=0}^{u-1} r_{3i+1} x^i$, $c_2 = \sum_{i=0}^{u-1} r_{3i+2} x^i$, we have simply $\sqrt[3]{r} = c_0 + x^{1/3} \cdot c_1 + x^{2/3} \cdot c_2$.

2.2 Case $m \equiv 2 \pmod 3$

Let $m = 3u + 2$. The folklore algorithm reads:

$$
\begin{aligned}
r &= \sum_{i=0}^{m-1} r_i x^i \\
&= \sum_{i=0}^{u} r_{3i} x^{3i} + x \cdot \sum_{i=0}^{u} r_{3i+1} x^{3i} + x^2 \cdot \sum_{i=0}^{u} r_{3i+2} x^{3i}. \\
\therefore \sqrt[3]{r} &= \sum_{i=0}^{u} r_{3i} x^{i} + x^{1/3} \cdot \sum_{i=0}^{u} r_{3i+1} x^{i} + x^{2/3} \cdot \sum_{i=0}^{u} r_{3i+2} x^{i}.
\end{aligned}
$$

Defining $c_0 = \sum_{i=0}^{u} r_{3i} x^i$, $c_1 = \sum_{i=0}^{u} r_{3i+1} x^i$, $c_2 = \sum_{i=0}^{u} r_{3i+2} x^i$, we have simply $\sqrt[3]{r} = c_0 + x^{1/3} \cdot c_1 + x^{2/3} \cdot c_2$.

3 Exploiting trinomial representation

Using a trinomial representation is advantageous for implementing modular reduction. We now show that a careful trinomial choice is also advantageous for taking cube roots when $k \equiv m \pmod 3$. We consider separately the cases $m \equiv 1 \pmod 3$ and $m \equiv 2 \pmod 3$. Notice that in either case no modular reduction is needed, since all terms are smaller than m.

3.1 Case $m \equiv 1 \pmod 3$, $k \equiv 1 \pmod 3$

Let $m = 3u + 1$ and $k = 3v + 1$. We observe that $x^{3u+1} + ax^{3v+1} + b = 0 \implies x^{2/3} = -bx^{u+1} - abx^{v+1} \implies x^{1/3} = x^{2u+1} - ax^{u+v+1} + x^{2v+1}$.

Thus $\sqrt[3]{r} = c_0 + x^{1/3} \cdot c_1 + x^{2/3} \cdot c_2 = c_0 + c_1(x^{2u+1} - ax^{u+v+1} + x^{2v+1}) - c_2(bx^{u+1} + abx^{v+1}) = c_0 + c_1^{\ll 2u+1} - ac_1^{\ll u+v+1} + c_1^{\ll 2v+1} - bc_2^{\ll u+1} - abc_2^{\ll v+1}$.

3.2 Case $m \equiv 2 \pmod 3$, $k \equiv 2 \pmod 3$

Let $m = 3u + 2$ and $k = 3v + 2$. We observe that $x^{3u+2} + ax^{3v+2} + b = 0 \implies x^{1/3} = -bx^{u+1} - abx^{v+1} \implies x^{2/3} = x^{2u+2} - ax^{u+v+2} + x^{2v+2}$.

Thus $\sqrt[3]{r} = c_0 + x^{1/3} \cdot c_1 + x^{2/3} \cdot c_2 = c_0 - c_1(bx^{u+1} + abx^{v+1}) + c_2(x^{2u+2} - ax^{u+v+2} + x^{2v+2}) = c_0 - bc_1^{\ll u+1} - abc_1^{\ll v+1} + c_2^{\ll 2u+2} - ac_2^{\ll u+v+2} + c_2^{\ll 2v+2}$.

4 Examples

Several fields of interest for pairing-based cryptosystems admit an irreducible trinomial $x^m + ax^k + b$ with the optimal property $k \equiv m \pmod 3$; in table 1 we list a few with $a = 1$ and $b = -1$.

finite field	irreducible trinomial
$\mathbb{F}_{3^{97}}$	$x^{97} + x^{16} - 1$
$\mathbb{F}_{3^{167}}$	$x^{167} + x^{92} - 1$
$\mathbb{F}_{3^{193}}$	$x^{193} + x^{64} - 1$
$\mathbb{F}_{3^{239}}$	$x^{239} + x^{26} - 1$

Table 1. Trinomials $x^m + x^k - 1$ such that $k \equiv m \pmod 3$.

Timings for the Duursma-Lee algorithm are given in table 2. These timings refer to a C++ implementation running on an Athlon XP 2GHz; the pairing algorithm itself was implemented according to the guidelines in [4]. We see that the overall pairing time decreases by about 10% with the use of trinomials.

finite field	plain	trinomial
$\mathbb{F}_{3^{97}}$	4.828	4.328
$\mathbb{F}_{3^{167}}$	27.047	24.625
$\mathbb{F}_{3^{193}}$	33.922	31.062
$\mathbb{F}_{3^{239}}$	63.562	57.344

Table 2. Timings of the Duursma-Lee (in ms).

5 Conclusion

We described an improvement for the folklore cube root algorithm in characteristic 3, an important operation for the efficient computation of the Tate pairing on supersingular elliptic curves. The complexity of our technique is only $O(m)$ rather than $O(m^2)$.

6 Acknowledgements

We are grateful to Mike Scott for bringing [2] to our attention, and to Fré Vercauteren for pointing out the folklore status of the basic cube root algorithm.

References

1. I. Duursma and H.-S. Lee. Tate pairing implementation for hyperelliptic curves $y^2 = x^p - x + d$. In *Advances in Cryptology – Asiacrypt'2003*, volume 2894 of *Lecture Notes in Computer Science*, pages 111–123, Taipei, Taiwan, 2003. Springer-Verlag.
2. K. Fong, D. Hankerson, J. López, and A. Menezes. Field inversion and point halving revisited. Technical report CORR 2003-18, University of Waterloo, 2002.
3. R. Granger, D. Page, and M. Stam. Hardware and software normal basis arithmetic for pairing based cryptography in characteristic three. Cryptology ePrint Archive, Report 2004/157, 2004.
4. M. Scott and Paulo S. L. M. Barreto. Compressed pairings. In *Proceedings*, Lecture Notes in Computer Science, Santa Barbara, USA, 2004. Advances in Cryptology – Crypto'2004, Springer-Verlag. to appear.