A Refined Algorithm for the η_T Pairing Calculation in Characteristic Three

Jean-Luc Beuchat[1], Masaaki Shirase[2], Tsuyoshi Takagi[2], and Eiji Okamoto[1]

[1] Graduate School of Systems and Information Engineering, University of Tsukuba, 1-1-1 Tennodai, Tsukuba, Ibaraki, 305-8573, Japan
[2] Future University-Hakodate, School of Systems Information Science, 116-2 Kamedanakano-cho, Hakodate, Hokkaido, 041-8655, Japan

Abstract. We describe further improvements of the η_T pairing algorithm in characteristic three. Our approach combines the loop unrolling technique introduced by Granger *et. al* for the Duursma-Lee algorithm, and a novel algorithm for multiplication over $\mathbb{F}_{3^{6m}}$ proposed by Gorla *et al.* at SAC 2007. For $m = 97$, the refined algorithm reduces the number of multiplications over $\mathbb{F}_{3^m}$ from 815 to 692.

Keywords: η_T pairing, finite field arithmetic, characteristic three.

1 Introduction

This short paper describes further improvements of the η_T pairing algorithm in characteristic three without inverse Frobenius maps proposed in [3] (Algorithm 1). We consider the supersingular elliptic curve $E : y^2 = x^3 - x + 1$ over $\mathbb{F}_{3^m}$ and denote by $E(\mathbb{F}_{3^m})[\ell]$ the ℓ-torsion subgroup of $E(\mathbb{F}_{3^m})$. The η_T pairing is the map $\eta_T : E(\mathbb{F}_{3^m})[\ell] \times E(\mathbb{F}_{3^m})[\ell] \to \mathbb{F}^*_{3^{6m}}$ defined by $\eta_T(P,Q) = f_{T,P}(\psi(Q))$, where $T \in \mathbb{Z}$ and $f_{T,P}$ is a rational function on the curve with divisor $[T](P) - (TP) - [T-1](\mathcal{O})$. The distortion map $\psi : E(\mathbb{F}_{3^m}) \to E(\mathbb{F}_{3^{6m}})$ is defined, for all $Q = (x_q, y_q) \in E(\mathbb{F}_{3^m})$, by $\psi(Q) = (-x_q + \rho, y_q\sigma)$, where σ and ρ belong to $\mathbb{F}_{3^{6m}}$ and satisfy $\sigma^2 = -1$ and $\rho^3 = \rho + 1$ respectively. We construct $\mathbb{F}_{3^{6m}}$ as an extension of $\mathbb{F}_{3^m}$ using the basis $(1, \sigma, \rho, \sigma\rho, \rho^2, \sigma\rho^2)$. Hence, arithmetic operations over $\mathbb{F}_{3^{6m}}$ are replaced by computations over $\mathbb{F}_{3^m}$. In order to get a well-defined, non-degenerate, bilinear pairing, a final exponentiation is mandatory: we have to compute $\eta_T(P,Q)^W$, where $W = (3^{3m} - 1)(3^m + 1)(3^m - 3^{\frac{m+1}{2}} + 1)$.

In the following, we take advantage of a novel algorithm for multiplication over $\mathbb{F}_{3^{6m}}$ [4] and apply the loop unrolling technique proposed by Granger *et al.* for the Duursma-Lee algorithm [5]. For $m = 97$, the refined algorithm reduces the number of multiplications over $\mathbb{F}_{3^m}$ from 815 to 692, thus improving software and hardware implementations of the η_T pairing.

2 Refined Algorithm

Granger *et al.* proposed a loop unrolling technique for the Duursma-Lee algorithm [5]. They exploit the sparsity of R_1 in order to reduce the number of

Algorithm 1 Computation of $\eta_T(P,Q)^W$ [3].

Input: $P = (x_p, y_p)$ and $Q = (x_q, y_q) \in E(\mathbb{F}_{3^m})[l]$. The algorithm requires R_0 and $R_1 \in \mathbb{F}_{3^{6m}}$, as well as $r_0 \in \mathbb{F}_{3^m}$ and $d \in \mathbb{F}_3$ for intermediate computations.
Output: $\eta_T(P,Q)^{(3^{3m}-1)(3^m+1)(3^m+1-3^{(m+1)/2})}$.

```
for i = 0 to (m-1)/2 - 1 do
    x_p ← x_p^9 - 1; y_p ← -y_p^9;
end for
y_p ← -y_p; d ← 1;
r_0 ← x_p + x_q + d;
R_0 ← -y_p r_0 + y_q σ + y_p ρ;
R_1 ← -r_0^2 + y_p y_q σ - r_0 ρ - ρ^2;
R_0 ← (R_0 R_1)^3;
for i = 0 to (m-1)/2 - 1 do
    y_p ← -y_p; x_q ← x_q^9; y_q ← y_q^9; d ← (d - 1) mod 3;
    r_0 ← x_p + x_q + d;
    R_1 ← -r_0^2 + y_p y_q σ - r_0 ρ - ρ^2;
    R_0 ← (R_0 R_1)^3;
end for
R_0 ← R_0^((3^{3m}-1)(3^m+1)(3^m+1-3^{(m+1)/2}));
R_0 ← (3^m)-th root of R_0;
return R_0;
```

multiplications over $\mathbb{F}_{3^m}$. Let $R_1[i]$ and $R_1[i+1]$ denote the value of R_1 at steps i and $i+1$ respectively. By noting that $R_1[i]^3$ is as sparse as $R_1[i]$, we can apply the same approach to Algorithm 1. Let $A = a_0 + a_1\sigma + a_2\rho + a_3\sigma\rho + a_4\rho^2 + a_5\sigma\rho^2$ and recall that the cubing formula is given by:

$$\begin{aligned} A^3 = {} & (a_0^3 + a_2^3 + a_4^3) + (-a_1^3 - a_3^3 - a_5^3)\sigma + (a_2^3 - a_4^3)\rho + \\ & (-a_3^3 + a_5^3)\sigma\rho + a_4^3\rho^2 + (-a_5^3)\sigma\rho^2. \end{aligned}$$

By substituting $a_0 = -r_0[i]^2$, $a_1 = y_p[i]y_q[i]$, $a_2 = -r_0[i]$, $a_3 = a_5 = 0$, and $a_4 = -1$ in the above equation, we obtain:

$$R_1[i]^3 = (-r_0[i]^6 - r_0[i]^3 - 1) - (y_p[i]y_q[i])^3\sigma + (-r_0[i]^3 + 1)\rho - \rho^2.$$

By unrolling the main loop of Algorithm 1, we get:

$$\begin{aligned} R_0[i+1] &= (R_0[i] \cdot R_1[i+1])^3 \\ &= ((R_0[i-1] \cdot R_1[i])^3 \cdot R_1[i+1])^3 \\ &= (R_0[i-1]^3 \cdot R_1[i]^3 \cdot R_1[i+1])^3. \end{aligned}$$

The product $R_1[i]^3 \cdot R_1[i+1]^3$ can be computed by means of six multiplications over $\mathbb{F}_{3^m}$ (Algorithm 2). Note that neither $R_0[i+1]$ nor $R_1[i]^3 \cdot R_1[i+1]^3$ are sparse in general. Their multiplication can be performed according to a novel algorithm introduced by Gorla *et al.* [4]. This approach is based on the fast Fourier transform and reduces the number of multiplications over $\mathbb{F}_{3^m}$ from 18 (see for instance [6]) to 15 (Algorithm 3). Note that we rewrote the algorithm in order to save additions. Therefore, $R_0[i+1]$ can be computed by means of 25 multiplications over $\mathbb{F}_{3^m}$ (Table 1). Algorithm 4 summarizes the η_T pairing calculation with loop unrolling. The first multiplication over $\mathbb{F}_{3^{6m}}$ (lines 7 and 8) involves 8 multiplications over $\mathbb{F}_{3^m}$ [1]. The final exponentiation features a single multiplication over $\mathbb{F}_{3^{6m}}$ [2]. Thus, only three multiplications over $\mathbb{F}_{3^m}$ can be saved here. Table 2 summarizes the number of multiplications over $\mathbb{F}_{3^m}$ requested for the full pairing. When $m = 97$, we have to carry out $8+25\cdot(m-1)/4+84 = 692$ multiplications over $\mathbb{F}_{3^m}$ instead of 815 as in [1].

Algorithm 2 Computation of $R_1[i]^3 \cdot R_1[i+1]$.

Input: $r_0[i]$, $r_0[i+1]$, $y_p[i]$, $y_p[i+1]$, $y_q[i]$, and $y_q[i+1] \in \mathbb{F}_{3^m}$.
Output: $c_0 + c_1\sigma + c_2\rho + c_3\sigma\rho + c_4\rho^2 + c_5\sigma\rho^2 = R_1[i]^3 \cdot R_1[i+1]$.

$a_0 \leftarrow -r_0[i]^6 - r_0[i]^3 - 1$; $a_1 \leftarrow -(y_p[i]y_q[i])^3 = (y_p[i+1]y_q[i])^3$; $a_2 \leftarrow -r_0[i]^3 + 1$;
$b_0 \leftarrow r_0[i+1]^2$; $b_1 \leftarrow y_p[i+1]y_q[i+1]$; $b_2 \leftarrow r_0[i+1]$;
$e_0 \leftarrow a_0 + a_1$; $e_1 \leftarrow a_0 + a_2$; $e_2 \leftarrow a_1 + a_2$;
$e_3 \leftarrow -b_0 + b_1$; $e_4 \leftarrow -b_0 - b_2$; $e_5 \leftarrow b_1 - b_2$;
$e_6 \leftarrow a_0 \cdot b_0$; $e_7 \leftarrow a_1 \cdot b_1$; $e_8 \leftarrow a_2 \cdot b_2$;
$e_9 \leftarrow e_0 \cdot e_3$; $e_{10} \leftarrow e_1 \cdot e_4$; $e_{11} \leftarrow e_2 \cdot e_5$;
$c_0 \leftarrow -e_6 - e_7 + b_2 - a_2$;
$c_1 \leftarrow e_9 + e_6 - e_7$;
$c_2 \leftarrow e_{10} + e_6 + e_8 - a_2 + b_2 + 1$;
$c_3 \leftarrow e_{11} + e_8 - e_7$;
$c_4 \leftarrow -e_8 - a_0 + b_0 + 1$;
$c_5 \leftarrow -a_1 - b_1$;

Table 1. Number of multiplications over $\mathbb{F}_{3^m}$ to compute $R_0[i+1]$.

Operation	**# multiplications**
$r_0[i]^2$, $r_0[i+1]^2$, $y_p[i]y_q[i]$, and $y_p[i+1]y_q[i+1]$	4
$S = R_1[i]^3 \cdot R_1[i+1]$	6 (Algorithm 2)
$R_0[i+1] = R_0[i-1]^3 \cdot S$	15 [4]

Acknowledgments

This work was supported by the New Energy and Industrial Technology Development Organization (NEDO), Japan.

Table 2. Number of multiplications over $\mathbb{F}_{3^m}$ to compute the full η_T pairing.

Operation	**# multiplications**
$\eta_T(P, Q)$	$25 \cdot \frac{m-1}{4} + 8$
Final exponentiation	84 [2, 4]

References

1. J.-L. Beuchat, N. Brisebarre, J. Detrey, and E. Okamoto. Arithmetic operators for pairing-based cryptography. Cryptology ePrint Archive, Report 2007/091, 2007.
2. J.-L. Beuchat, N. Brisebarre, M. Shirase, T. Takagi, and E. Okamoto. A coprocessor for the final exponentiation of the η_T pairing in characteristic three. In C. Carlet and B. Sunar, editors, *Proceedings of Waifi 2007*, number 4547 in Lecture Notes in Computer Science, pages 25–39. Springer, 2007.
3. J.-L. Beuchat, M. Shirase, T. Takagi, and E. Okamoto. An algorithm for the η_T pairing calculation in characteristic three and its hardware implementation. In P. Kornerup and J.-M. Muller, editors, *Proceedings of the 18th IEEE Symposium on Computer Arithmetic*, pages 97–104. IEEE Computer Society, 2007.
4. E. Gorla, C. Puttmann, and J. Shokrollahi. Explicit formulas for efficient multiplication in $\mathbb{F}_{3^{6m}}$. In *Proceedings of SAC 2007*, Lecture Notes in Computer Science. Springer, 2007.
5. R. Granger, D. Page, and M. Stam. On small characteristic algebraic tori in pairing-based cryptography. Cryptology ePrint Archive, Report 2004/132, 2004.
6. T. Kerins, W. P. Marnane, E. M. Popovici, and P.S.L.M. Barreto. Efficient hardware for the Tate Pairing calculation in characteristic three. In J. R. Rao and B. Sunar, editors, *Cryptographic Hardware and Embedded Systems – CHES 2005*, number 3659 in Lecture Notes in Computer Science, pages 412–426. Springer, 2005.

Algorithm 3 Multiplication over $\mathbb{F}_{3^{6m}}$ [4].

Input: A, $B \in \mathbb{F}_{3^{6m}}$ with $A = a_0 + a_1\sigma + a_2\rho + a_3\sigma\rho + a_4\rho^2 + a_5\sigma\rho^2$ and $B = b_0 + b_1\sigma + b_2\rho + b_3\sigma\rho + b_4\rho^2 + b_5\sigma\rho^2$.

Output: $C = AB$. The algorithm requires 15 multiplications and 67 additions over $\mathbb{F}_{3^m}$.

$r_0 \leftarrow a_0 + a_4$; $e_0 \leftarrow r_0 + a_2$; $e_{12} \leftarrow r_0 - a_2$;
$r_0 \leftarrow b_0 + b_4$; $e_3 \leftarrow r_0 + b_2$; $e_{15} \leftarrow r_0 - b_2$;
$r_0 \leftarrow a_0 - a_4$; $e_6 \leftarrow r_0 - a_3$; $e_{18} \leftarrow r_0 + a_3$;
$r_0 \leftarrow b_0 - b_4$; $e_9 \leftarrow r_0 - b_3$; $e_{21} \leftarrow r_0 + b_3$;
$r_0 \leftarrow a_1 + a_5$; $e_1 \leftarrow r_0 + a_3$; $e_{13} \leftarrow r_0 - a_3$;
$r_0 \leftarrow b_1 + b_5$; $e_4 \leftarrow r_0 + b_3$; $e_{16} \leftarrow r_0 - b_3$;
$r_0 \leftarrow a_1 - a_5$; $e_7 \leftarrow r_0 + a_2$; $e_{19} \leftarrow r_0 - a_2$;
$r_0 \leftarrow b_1 - b_5$; $e_{10} \leftarrow r_0 + b_2$; $e_{22} \leftarrow r_0 - b_2$;
$e_2 \leftarrow e_0 + e_1$; $e_5 \leftarrow e_3 + e_4$; $e_8 \leftarrow e_6 + e_7$; $e_{11} \leftarrow e_9 + e_{10}$;
$e_{14} \leftarrow e_{12} + e_{13}$; $e_{17} \leftarrow e_{15} + e_{16}$; $e_{20} \leftarrow e_{18} + e_{19}$; $e_{23} \leftarrow e_{21} + e_{22}$;
$e_{24} \leftarrow a_4 + a_5$; $e_{25} \leftarrow b_4 + b_5$;
$m_0 \leftarrow e_0 \cdot e_3$; $m_1 \leftarrow e_2 \cdot e_5$; $m_2 \leftarrow e_1 \cdot e_4$;
$m_3 \leftarrow e_6 \cdot e_9$; $m_4 \leftarrow e_8 \cdot e_{11}$; $m_5 \leftarrow e_7 \cdot e_{10}$;
$m_6 \leftarrow e_{12} \cdot e_{15}$; $m_7 \leftarrow e_{14} \cdot e_{17}$; $m_8 \leftarrow e_{13} \cdot e_{16}$;
$m_9 \leftarrow e_{18} \cdot e_{21}$; $m_{10} \leftarrow e_{20} \cdot e_{23}$; $m_{11} \leftarrow e_{19} \cdot e_{22}$;
$m_{12} \leftarrow a_4 \cdot b_4$; $m_{13} \leftarrow e_{24} \cdot e_{25}$; $m_{14} \leftarrow a_5 \cdot b_5$;
$e_0 \leftarrow m_0 + m_4 + m_{12}$; $e_1 \leftarrow m_2 + m_{10} + m_{14}$;
$e_2 \leftarrow m_6 + m_{12}$; $e_3 \leftarrow -m_8 - m_{14}$; $e_4 \leftarrow m_7 + m_{13}$;
$e_5 \leftarrow e_3 + m_2$; $e_6 \leftarrow e_2 - m_0$;
$e_7 \leftarrow e_3 - m_2 + m_5 + m_{11}$; $e_8 \leftarrow e_2 + m_0 - m_3 - m_9$;
$c_0 \leftarrow -e_0 + e_1 - m_3 + m_{11}$;
$c_1 \leftarrow e_0 + e_1 - m_1 + m_5 + m_9 - m_{13}$;
$c_2 \leftarrow e_5 + e_6$;
$c_3 \leftarrow e_5 - e_6 + e_4 - m_1$;
$c_4 \leftarrow e_7 + e_8$;
$c_5 \leftarrow e_7 - e_8 + e_4 + m_1 - m_4 - m_{10}$;

Algorithm 4 Computation of $\eta_T(P,Q)^W$.

Input: $P = (x_p, y_p)$ and $Q = (x_q, y_q) \in E(\mathbb{F}_{3^m})[l]$. The algorithm requires R_0 and $R_1 \in \mathbb{F}_{3^{6m}}$, as well as $r_0 \in \mathbb{F}_{3^m}$ and $d \in \mathbb{F}_3$ for intermediate computations.

Output: $\eta_T(P,Q)^{(3^{3m}-1)(3^m+1)(3^m+1-3^{(m+1)/2})}$.

for $i = 0$ to $\frac{m-1}{2} - 1$ **do**
    $x_p \leftarrow x_p^9 - 1$; $y_p \leftarrow -y_p^9$;
end for
$y_p \leftarrow -y_p$; $d \leftarrow 1$;
$r_0 \leftarrow x_p + x_q + d$;
$R_0 \leftarrow -y_p r_0 + y_q \sigma + y_p \rho$;
$R_1 \leftarrow -r_0^2 + y_p y_q \sigma - r_0 \rho - \rho^2$;
$R_0 \leftarrow (R_0 R_1)^3$;
for $i = 0$ to $\frac{m-1}{4} - 1$ **do**
    $x_q \leftarrow x_q^9$; $y_q \leftarrow y_q^9$; $d \leftarrow (d-1) \bmod 3$;
    $r_0 \leftarrow x_p + x_q + d$;
    $R_1 \leftarrow (-r_0^6 - r_0^3 - 1) + (y_p y_q)^3 \sigma + (-r_0^3 + 1)\rho - \rho^2$;
    $R_0 \leftarrow R_0^3$;
    $x_q \leftarrow x_q^9$; $y_q \leftarrow y_q^9$; $d \leftarrow (d-1) \bmod 3$;
    $r_0 \leftarrow x_p + x_q + d$;
    $R_1 \leftarrow R_1 \cdot (-r_0^2 + y_p y_q \sigma - r_0 \rho - \rho^2)$;
    $R_0 \leftarrow (R_0 R_1)^3$;
end for
$R_0 \leftarrow R_0^{(3^{3m}-1)(3^m+1)(3^m+1-3^{(m+1)/2})}$;
$R_0 \leftarrow \sqrt[3^m]{R_0}$;
return R_0;