Speeding up the Arithmetic on Koblitz Curves of Genus Two

Christian Günther[1], Tanja Lange[2], and Andreas Stein[3]

[1] Siemens AG, Corporate Technology, Munich, Germany
christian-c.guenther@mchp.siemens.de
[2] Institut für Geometrie, TU Braunschweig, Pockelsstr. 14,
38106 Braunschweig, Germany,
ta.lange@tu-bs.de
[3] Department of Combinatorics and Optimization,
Centre for Applied Cryptographic Research, University of Waterloo,
Waterloo, ON, Canada N2L 3G1
astein@cacr.math.uwaterloo.ca

Abstract. Koblitz, Solinas, and others investigated a family of elliptic curves which admit faster cryptosystem computations. In this paper, we generalize their ideas to hyperelliptic curves of genus 2. We consider the following two hyperelliptic curves $C_\alpha : v^2 + uv = u^5 + \alpha\, u^2 + 1$ defined over $\mathbb{F}_2$ with $\alpha = 0, 1$, and show how to speed up the arithmetic in the Jacobian $\mathbb{J}_{C_\alpha}(\mathbb{F}_{2^n})$ by making use of the Frobenius automorphism. With two precomputations, we are able to obtain a speed-up by a factor of 5.5 compared to the generic double-and-add-method in the Jacobian. If we allow 6 precomputations, we are even able to speed up by a factor of 7.

1 Introduction

Public-key cryptosystems based on the discrete logarithm problem on elliptic curves over finite fields have been invented by Neal Koblitz [9] and Victor Miller [17]. Elliptic curve cryptosystems became a popular choice for implementations. The most important operation in an elliptic curve based cryptosystem is the computation of m–folds with a positive integer m. That means computing mP for a point P on an elliptic curve. For example, the complexity of the ElGamal encryption scheme [3] and the Diffie-Hellmann key agreement protocol [2] on an elliptic curve both depend mostly on the complexity of computing m–folds.

The standard method for computing m–folds in a group G is the *double-and-add-method*. If P is an element of G and m a positive integer, doubles and additions are performed with respect to the binary representation of m requiring about $\log_2(m)$ doubles and $\log_2(m)/2$ additions on average. Assuming that doubles and additions have about the same complexity, this method requires $3\log_2(m)/2$ group operations. Allowing precomputations and using memory, various techniques apply to speed up the double-and-add-method (see [8]).

In [11,12,23,15,24], a family of elliptic curves was investigated which allows to speed up the scalar multiplication considerably with the help of the Frobenius

D.R. Stinson and S. Tavares (Eds.): SAC 2000, LNCS 2012, pp. 106–117, 2001.

automorphism. They considered the elliptic curves $E : u^2 + uv = v^3 + av^2 + 1$ defined over $\mathbb{F}_2$ with base field $\mathbb{F}_{2^n}$, which are called *elliptic Koblitz curves*.

The fastest known attack to the elliptic curve discrete logarithm problem is the parallelized Pollard's rho method [20,22,27]. As noticed in [5,28], the attack time to these curves can be reduced by a factor of $\sqrt{2n}$ which causes one to select slightly larger secure key parameters.

Hyperelliptic curve cryptosystems have been introduced by Neal Koblitz [10] in 1989. Cantor's algorithm [1] provides an effective algorithm for performing the group law in the Jacobian of a hyperelliptic curve.

In this paper, we generalize the ideas for elliptic Koblitz curves to hyperelliptic curves of genus 2. We concentrate on the following two hyperelliptic curves

$$C_\alpha : v^2 + uv = u^5 + \alpha\, u^2 + 1 \qquad (\alpha = 0, 1) \ ,$$

which are defined over $\mathbb{F}_2$ and have the base field $\mathbb{F}_{2^n}$ where n is prime. These curves are generalized Koblitz curves of genus 2 and are twists of each other. Furthermore, they are the only non-supersingular curves mentioned in [10] and thus resist the Frey-Rück-attack [4].

We want to point out that the curves C_α have two major advantages. Firstly, the cardinality of the Jacobian of C_α, $\#\mathbb{J}_{C_\alpha}(\mathbb{F}_{2^n})$, can be easily determined for any n, whereas in general computing $\#\mathbb{J}_C(\mathbb{F}_{2^n})$ for a random hyperelliptic curve over $\mathbb{F}_{2^n}$ of genus 2 appears to be difficult. Secondly, we can use the Frobenius automorphism to eliminate many cryptosystem operations. On the cost of two precomputations, we are able to speed up the computation of m-folds by a factor of 5.5. With 6 precomputations, we even obtain a speed-up by a factor of 7. On the other side, a generalization of the methods in [5,28] shows that one can speed up the attack to hyperelliptic cryptosystems by a factor of $\sqrt{2n}$, if the curve has an automorphism of order n (see [7]). Since the curves C_α have at least an automorphism of order n, namely the Frobenius automorphism, the attack to cryptosystems based on the discrete logarithm in $\mathbb{J}_{C_\alpha}(\mathbb{F}_{2^n})$ can be sped up by a factor of $\sqrt{2n}$. As in the case of an elliptic curve, one then has to adjust the size of the key space marginally. Most of the results can be easily generalized to all other genus 2 hyperelliptic curves and, more general, to curves of arbitrary genus [13]. However, for arbitrary curves, one has to be careful with the parameter choice for g and n. Results in [6,7] seem to suggest that hyperelliptic curves of genus $g \geq 4$ are not as secure as elliptic curves.

2 Hyperelliptic Curves

2.1 Basic Definitions

For details on hyperelliptic curves we refer to [10,16,1,26]. Let $\mathbb{F}_q$ be a finite field and $\overline{\mathbb{F}}_q$ its algebraic closure. A non-singular hyperelliptic curve of genus g is defined by the equation

$$C : v^2 + h(u)v = f(u) \quad \text{in} \quad \mathbb{F}_q[u, v] \ , \tag{2.1}$$

where $h(u), f(u) \in \mathbb{F}_q[u]$, $\deg_u(h) \leq g$, $f(u)$ monic, $\deg_u(f) = 2g+1$, and if $y^2 + h(x)y = f(x)$ for $(x,y) \in \overline{\mathbb{F}}_q \times \overline{\mathbb{F}}_q$, then $2y + h(x) \neq 0$ or $h'(x)y - f'(x) \neq 0$.

Let $\mathbb{F}_{q^n}$ be a subfield of $\overline{\mathbb{F}}_q$ containing $\mathbb{F}_q$. The set of $\mathbb{F}_{q^n}$–points P on C is given by $C(\mathbb{F}_{q^n}) = \{(x,y) \in \mathbb{F}_{q^n}^2 \mid y^2 + h(x)y = f(x)\} \cup \{\infty\}$, where ∞ denotes the point at infinity. For a $\mathbb{F}_{q^n}$–point $P = (x,y) \in \mathbb{F}_{q^n}^2$, the *opposite* $\tilde{P}$ of P is immediately given by $\tilde{P} = (x, -y - h(x))$. For $P = \infty$ define $\tilde{P} = \infty$.

A *divisor* on C is a finite formal sum $D = \sum_P m_P P$, where m_P are integers that are 0 for almost all P. Then the *degree* of D is defined by $\deg D = \sum_P m_P$. D is said to be *defined over* $\mathbb{F}_{q^n}$, if[1] $D^\sigma = \sum_P m_P P^\sigma = D$ for any $\sigma \in$ Aut $(\overline{\mathbb{F}}_q/\mathbb{F}_{q^n})$. The set $\mathbb{D}_C(\mathbb{F}_{q^n})$ of divisors of C defined over $\mathbb{F}_{q^n}$ forms an additive group which contains the finite subgroup $\mathbb{D}_C^0(\mathbb{F}_{q^n})$ of all degree zero divisors of $\mathbb{D}$ defined over $\mathbb{F}_{q^n}$. The *greatest common divisor* of $D_1 = \sum_{P \in C} m_P P$ and $D_2 = \sum_{P \in C} n_P P$ in $\mathbb{D}_C^0(\mathbb{F}_{q^n})$ is defined by

$$\gcd(D_1, D_2) = \sum_{P \in C} \min(m_P, n_P)P - \Big(\sum_{P \in C} \min(m_P, n_P) \Big) \infty \ .$$

Furthermore, the divisor of a polynomial $G(u,v) \in \overline{\mathbb{F}}_q[u,v]$ is defined by $\operatorname{div}(G(u,v)) = \sum_P \operatorname{ord}_P(G)P - \sum_P \operatorname{ord}_P(G)\infty$, where $\operatorname{ord}_P(G)$ is the order of vanishing of $G(u,v)$ at P. Now, the divisor of a rational function $G(u,v)/H(u,v)$ is called a *principal divisor* and is defined by $\operatorname{div}(G(u,v)/H(u,v)) = \operatorname{div}(G(u,v)) - \operatorname{div}(H(u,v))$. We denote by $\mathbb{P}_C(\mathbb{F}_{q^n})$ the group of principal divisors. Since every principal divisor has degree 0, $\mathbb{P}_C(\mathbb{F}_{q^n})$ is a subgroup of $\mathbb{D}_C^0(\mathbb{F}_{q^n})$. Finally, the *Jacobian of C over $\mathbb{F}_{q^n}$* is given by

$$\mathbb{J}_C(\mathbb{F}_{q^n}) = \mathbb{D}_C^0(\mathbb{F}_{q^n}) / \mathbb{P}_C(\mathbb{F}_{q^n}) \ .$$

2.2 Reduced Divisors

Let C_f be the set $C - \{\infty\}$ of finite points on C. A degree zero divisor $D = \sum_{P \in C_f} m_P P - \big(\sum_{P \in C_f} m_P\big)\infty$ is called *semi-reduced*, if it satisfies the following conditions for each $P \in C_f$:

(i) $m_P \geq 0$.
(ii) If $P \neq \tilde{P}$ and $m_P > 0$, then $m_{\tilde{P}} = 0$.
(iii) If $P = \tilde{P}$ and $m_P > 0$, then $m_P = 1$.

A semi-reduced divisor is called *reduced*, if in addition

$$\sum_{P \in C_f} m_P \leq g \ .$$

Reduced divisors have the crucial property (see [19]) that for each degree zero divisor D there exists a unique reduced divisor D_r such that $D - D_r$ is a principal divisor. That means, the set of reduced divisors of C forms a complete system of

[1] P^σ denotes $(\sigma(x), \sigma(y))$, if $P = (x,y) \in \mathbb{F}_{q^n}^2$, and ∞, if $P = \infty$

representatives for the divisor classes of C. Semi-reduced divisors can be represented uniquely in an advantageous way (see [16,19]): Let $D = \sum_{P \in C_f} m_P P - \big(\sum_{P \in C_f} m_P\big)\infty$ be a semi-reduced divisor which is defined over $\mathbb{F}_{q^n}$. We put $a(u) = \prod_{P \in C_f}(u - x_P)^{m_P} \in \mathbb{F}_{q^n}[u]$, where $P = (x_P, y_P)$. Then there exists a unique polynomial $b(u) \in \mathbb{F}_{q^n}[u]$ such that $D = \gcd(\mathrm{div}(a(u)), (\mathrm{div}(b(u) - v))$ and

1. $\deg_u b < \deg_u a$,
2. $b(x_P) = y_P$ for each $P \in C_f$ with $m_P \neq 0$,
3. $a(u)$ divides $b(u)^2 + b(u)h(u) - f(u)$.

In this case, we write $D = [a(u), b(u)]$. It follows that every element D of $\mathbb{J}_C(\mathbb{F}_{q^n})$ can be uniquely represented by two polynomials $a(u), b(u) \in \mathbb{F}_{q^n}[u]$, where $a(u)$ is monic, $\deg b(u) < \deg a(u) \leq g$, and $a(u)$ divides $b(u)^2 + b(u)h(u) - f(u)$. We notice that operations in the Jacobian can be performed by using the arithmetic in $\mathbb{F}_{q^n}[u]$. Without explaining the algorithms here, we mention that there exists an effective method to add two elements of the Jacobian which is known as *Cantor's algorithm*. For details we refer to [1,10,16]. The generic operation needs $17g^2 + O(g)$ operations in $\mathbb{F}_{q^n}$ whereas doubling needs $16g^2 + O(g)$ operations in $\mathbb{F}_{q^n}$ (see [25])[2]. So, we can assume that both operations have roughly the same complexity. It is important to note that inversion is basically for free, since the negative of $D = [a(u), b(u)]$ is given by $-D = [a(u), -h(u) - b(u)]$.

2.3 Frobenius Automorphism

The Frobenius automorphism $\phi : \overline{\mathbb{F}}_q \longrightarrow \overline{\mathbb{F}}_q$, $x \longmapsto x^q$ extends to an automorphism on the Jacobian of C. Namely, we put $P^\phi = (x^q, y^q)$ for $P = (x, y) \in \overline{\mathbb{F}}_q \times \overline{\mathbb{F}}_q$, and $\infty^\phi = \infty$. For a divisor $D = \sum_{P \in C} m_P P$ of C define D^ϕ to be $\sum_{P \in C} m_P P^\phi$. A very important property of the Frobenius action on semi-reduced divisors is provided by the following

Theorem 2.1. *Let $C : v^2 + h(u)v = f(u)$ be a hyperelliptic curve of genus g, where $h(u), f(u) \in \mathbb{F}_q[u]$. If D is a semi-reduced divisor of C which is defined over $\mathbb{F}_{q^n}$, then the divisor D^ϕ is semi-reduced and defined over $\mathbb{F}_{q^n}$. In particular, if $D = [a(u), b(u)]$ with $a(u), b(u) \in \mathbb{F}_{q^n}[u]$, then we have*

$$D^\phi = [a(u)^\phi,\ b(u)^\phi]\ .$$

Proof. Let $D = \sum_{P \in C_f} m_P P - \big(\sum_{P \in C_f} m_P\big)\infty$ be a semi-reduced divisor which is defined over $\mathbb{F}_{q^n}$. Since ϕ is an automorphism, each property, that D^ϕ has to satisfy for being semi-reduced, follows from the condition that D is semi-reduced. ϕ commutes with any $\sigma \in \mathrm{Aut}(\overline{\mathbb{F}}_q/\mathbb{F}_{q^n})$. Therefore, we have $(D^\phi)^\sigma = (D^\sigma)^\phi = D^\phi$ for any $\sigma \in \mathrm{Aut}(\overline{\mathbb{F}}_q/\mathbb{F}_{q^n})$. That means D^ϕ is defined over $\mathbb{F}_{q^n}$. Now, let $a(u) = \prod_{P \in C_f}(u - x_P)^{m_P} \in \mathbb{F}_{q^n}[u]$ and let $b(u)$ be the unique polynomial satisfying a)-c) in Section 2.2 such that

[2] We remark that there exist even faster methods if the characteristic of $\mathbb{F}_{q^n}$ is 2 and if we use normal basis representation for elements in $\mathbb{F}_{q^n}$

$$D = \gcd(\mathrm{div}(a(u)), \mathrm{div}(b(u) - v)) = [a(u), b(u)] \ ,$$

Then

$$D^\phi = \sum_{P \in C_f} m_P P^\phi - \big(\sum_{P \in C_f} m_P\big)\infty = \gcd(\mathrm{div}(\overline{a}(u)), \mathrm{div}(\overline{b}(u) - v)) = [\overline{a}(u), \overline{b}(u)] \ ,$$

where $\overline{a}(u) = \prod_{P \in C_f} (u - x_P^q)^{m_P} \in \mathbb{F}_{q^n}[u]$, and $\overline{b}(u) \in \mathbb{F}_{q^n}[u]$ is the unique polynomial satisfying a)-c) in Section 2.2 for D^ϕ. Clearly, $a^\phi(u) = \prod_{P \in C_f} (u - x_P^q)^{m_P} = \overline{a}(u)$. It remains to show that $b^\phi(u) = \overline{b}(u)$. Firstly, we have $\deg_u b^\phi < \deg_u a^\phi = \deg \overline{a}$, since $\deg_u b < \deg_u a$. Secondly, $b^\phi(x_P^q) = (b(x_P))^q = y_P^q$ for each $P \in C_f$ with $m_P \neq 0$. Thirdly, $a^\phi(u)$ divides $(b(u)^2 + b(u)h(u) - f(u))^\phi = b^\phi(u)^2 + b^\phi(u)h(u) - f(u)$, since $a(u)$ divides $b(u)^2 + b(u)h(u) - f(u)$ and $h(u), f(u) \in \mathbb{F}_q[u]$. Since $\overline{b}(u)$ is unique with these three properties for D^ϕ and $\overline{a}(u) = a^\phi(u)$, we must have $b^\phi(u) = \overline{b}(u)$.

An important consequence of this theorem is that if $D = [a(u), b(u)]$ is a reduced divisor representing an element of $\mathbb{J}_C(\mathbb{F}_{q^n})$, then the action of the Frobenius on D is given by $D^\phi = [a(u)^\phi,\ b(u)^\phi]$. Notice that this is only true since C is defined over the subfield $\mathbb{F}_q$ of $\mathbb{F}_{q^n}$. The interpretation is that if $a(u) = \sum_{i=0}^{k} a_i u^i \in \mathbb{F}_{q^n}[u]$ and $b(u) = \sum_{i=0}^{k'} b_i u^i \in \mathbb{F}_{q^n}[u]$ are the explicit representations of $a(u)$ and $b(u)$, then $a^\phi(u) = \sum_{i=0}^{k} a_i^q u^i$ and $b^\phi(u) = \sum_{i=0}^{k'} b_i^q u^i$. The practical meaning of this observation is that if we use normal basis representation for elements in $\mathbb{F}_{q^n}$, then $a^\phi(u)$ and $b^\phi(u)$ can be determined by simply shifting the normal basis representation of each coefficient a_i and b_i in order to compute D^ϕ. The complexity is therefore at most $2g$ cyclic shifts. These shift operations are basically "for free" when compared to the more expensive group operation in the Jacobian.

3 Algorithms for $v^2 + uv = u^5 + \alpha\, u^2 + 1$

For the remainder of the paper, we consider the curves $C_\alpha : v^2 + uv = u^5 + \alpha\, u^2 + 1$ with $\alpha = 0, 1$ which are defined over $\mathbb{F}_2$. From [10], we know that the characteristic equation of the Frobenius of the curve C_α is given by

$$T^4 + (-1)^\alpha T^3 + (-1)^\alpha 2T + 4 = 0 \ . \tag{3.2}$$

It follows that

$$-4D = \phi^4(D) + (-1)^\alpha \phi^3(D) + (-1)^\alpha 2\phi(D) \tag{3.3}$$

for any $D \in \mathbb{J}_{C_\alpha}(\overline{\mathbb{F}}_2)$. Here, $\phi(D) := D^\phi$. The equation (3.2) has four solutions

$$\tau_{1/2} = (-1)^{\alpha+1}(\mu_1 \pm i\sqrt{4 - \mu_1})/2 \quad , \quad \tau_{3/4} = (-1)^{\alpha+1}(\mu_2 \pm i\sqrt{4 - \mu_2})/2 \ ,$$

where $\mu_{1/2} = (1 \pm \sqrt{17})/2$. We put $\tau = \tau_1$ and can regard τ as the element ϕ in the automorphism ring of $\mathbb{J}_{C_1}(\overline{\mathbb{F}}_2)$. As the roots for both curves are equal up to signs, it suffices to consider C_1. Analogous results hold true for the curve C_0 with some slight modifications. In particular, $\#\mathbb{J}_{C_0}(\mathbb{F}_{2^n})$ differs from $\#\mathbb{J}_{C_1}(\mathbb{F}_{2^n})$ only for odd n.

3.1 Computing τ-Adic Expansions

We are interested in expansions like $11 = -\tau^7 + \tau^4 - 2\tau^2 + 3$, which enable us to compute $11D$ by $11D = -\phi^7(D) + \phi^4(D) - 2\phi^2(D) + 3D$. More generally, for a given integer m, we are interested in its τ-adic expansion $m = \sum_{i=0}^{l-1} c_i\tau^i$ with coefficients $c_i \in R$, where R is a suitable subset of the integers. First, we consider $R = \{0, \pm 1, \pm 2, \pm 3\}$. In Sect. 5, we will vary the set R.

Let $z = a + b\tau + c\tau^2 + d\tau^3$ be any element of $\mathbb{Z}[\tau]$. Since τ is a root of (3.2), z is divisible by τ if and only if $4 \mid a$ as can be shown by direct computation. Therefore, we must have $\tau \mid z - u$ for some $u \in \{0, 1, 2, 3\}$. It follows that

$$z - u = \tau\Big(\Big(\frac{a-u}{2} + b\Big) + c\tau + \Big(\frac{a-u}{4} + d\Big)\tau^2 - \frac{a-u}{4}\tau^3\Big). \tag{3.4}$$

With $R = \{0, \pm 1, \pm 2, \pm 3\}$ we are able to realize the strategy "*at least one of four consecutive coefficients is zero*" when determining the c_i's. The basic algorithm for computing τ-adic expansions of $z = a + b\tau + c\tau^2 + d\tau^2 \in \mathbb{Z}[\tau]$ is to choose an $u \in R$ such that $4 \mid z - u$, to divide $z - u$ by τ and then to repeat these two steps with the new, replaced $z' = ((a-u)/2+b)+c\tau+((a-u)/4+d)\tau^2-((a-u)/4)\tau^3$, see (3.4), until the resulting z' will be zero. Then the sequence of those u's will be the sequence of the coefficients $c_0, \ldots, c_{l-1} \in R$ we have searched for. We proceed as follows:

1. If $4 \mid a$, then $\tau \mid z$ and we clearly use $u = 0$.
2. If $4 \nmid a$, then since $R = \{0, \pm 1, \pm 2, \pm 3\}$ we have exactly two choices for u and we can try to make one of the subsequent a's divisible by 4:
   (a) If $2 \mid b$, then there is exactly one $u \in R$ such that $4 \mid a - u$ and $4 \mid ((a - u)/2 + b)$, namely

$b \bmod 4 \backslash a \bmod 8$	1	2	3	5	6	7
0	1	2	3	−3	−2	−1
2	−3	−2	−1	1	2	3

   Using these values for u, the actual u is non zero but the next one will be zero.
   (b) If $2 \nmid b$, then we are only able to make the third successor of the actual a at the latest be divisible by 4 by using.

$d \bmod 2 \backslash a \bmod 8$	1	2	3	5	6	7
0	1	2	3	−3	−2	−1
1	−3	−2	−1	1	2	3

This strategy produces expansions $m = \sum_{i=0}^{l-1} c_i\tau^i$ with coefficients c_i in $R = \{0, \pm 1, \pm 2, \pm 3\}$, where $c_i c_{i+1} c_{i+2} c_{i+3} = 0 \quad (i \in \{0, \ldots, l-4\})$.

For an integer m, the expected length l of such an expansion is $2\log_2 |m|$. Note that this is about twice as long as the binary expansion $m = \sum b_i 2^i$, where $b_i \in \{0, 1\}$. We will show in the following section how to reduce the length of the τ-adic representation.

3.2 Reducing the Length of the Representation

Since the order of the Frobenius is n, two automorphisms are the same, if $\sum_{i=0}^{l_1-1} c_i\phi^i - \sum_{i=0}^{l_2-1} d_i\phi^i \in (\phi^n - 1)\mathbb{Z}[\phi]$. Thus the corresponding τ-adic expansions are equivalent. Therefore before expanding m, we reduce it modulo $\tau^n - 1$ in $\mathbb{Z}[\tau]$ to obtain a shorter representation $[m] = \sum_{i=0}^{l-1} c_i\phi^i$ of the multiplication-by-m-map. We look for an element $M \in \mathbb{Z}[\tau]$ such that $M \equiv m \mod \tau^n - 1$ and the τ-adic expansion of M is as short as possible, i. e. $|M|$ is as small as possible.

Theorem 3.1. *For any positive integers m and n, there exists an element $M \in \mathbb{Z}[\tau]$ such that*

1. $m \equiv M \mod \tau^n - 1$,
2. $2\log_2 |M| < n + 5$.

Proof. Let $r = m/(\tau^n - 1) \in \mathbb{Q}(\tau)$. Then there exist r_0, r_1,r_2, r_3 in $\mathbb{Q}$ such that $r = \sum_{i=0}^{3} r_i\tau^i$. Let v_i be the nearest integer to r_i for $i = 0, \ldots, 3$. We put $v = \sum_{i=0}^{3} v_i\tau^i$ and $M = m - v(\tau^n - 1)$. Then $m \equiv M \mod \tau^n - 1$. By using the identity

$$|M|^2 = |m - v(\tau^n - 1)|^2 = \left|\frac{m}{\tau^n - 1} - v\right|^2 |\tau^n - 1|^2$$

we derive that $|M/(\tau^n - 1)|^2 < 14$. Since $|\tau^n - 1|^2 < (2^{n/2} + 1)^2$, we derive that $2\log_2 |M| < n + 5$.

For any positive integers m and n, we are easily able to determine an element $M \in \mathbb{Z}[\tau]$, satisfying $m \equiv M \mod \tau^n - 1$ and having a τ-adic expansion of length $l \sim n$. We call this representation the *reduced τ-adic expansion* of m. In the automorphism ring of the Jacobian, we obtain for the multiplication-by-m map that $[m] = \sum_{i=0}^{l-1} c_i\phi^i$. The algorithm to compute M from m is along the lines of the proof of Theorem 3.1. We therefore omit it. We remark here that we need to be able to find a representation of $\tau^n - 1$ as $\tau^n - 1 = a + b\tau + c\tau^2 + d\tau^3$ with integers a, b, c, d. The next section will solve this problem. Furthermore, we need to be able to compute multiplicative inverses in $\mathbb{Q}[\tau]$. This can be done by the usual extended gcd for polynomials.

3.3 Representing $\tau^n - 1$ by $a + b\tau + c\tau^2 + d\tau^3$

To compute a, b, c, $d \in \mathbb{Z}$ such that $\tau^n - 1 = a + b\tau + c\tau^2 + d\tau^3$ is no difficult task. Let n be a positive integer. Suppose that $\tau^{n-1} = a_{n-1} + b_{n-1}\tau + c_{n-1}\tau^2 + d_{n-1}\tau^3$ for unique integers a_{n-1}, b_{n-1}, c_{n-1}, d_{n-1}, then

$$\begin{aligned}\tau^n &= a_{n-1}\tau + b_{n-1}\tau^2 + c_{n-1}\tau^3 + d_{n-1}\tau^4 \\ &= -4d_{n-1} + (a_{n-1} + 2d_{n-1})\tau + b_{n-1}\tau^2 + (c_{n-1} + d_{n-1})\tau^3 \ ,\end{aligned}$$

since $\tau^4 = -4 + 2\tau + \tau^3$, and hence

$$\tau^n - 1 = -(4d_{n-1} + 1) + (a_{n-1} + 2d_{n-1})\tau + b_{n-1}\tau^2 + (c_{n-1} + d_{n-1})\tau^3.$$

Starting with $\tau^0 = 1$, we can compute the integers a, b, c, d iteratively.

3.4 Computing m-Folds Using τ-Adic Expansions

We now present our main algorithm for computing m-folds in the Jacobian of the genus 2 curve $C_1 : v^2 + uv = u^5 + u^2 + 1$ with base field F_{2^n}. Let $D = [(a(u), b(u))] \in \mathbb{J}_{C_1}(\mathbb{F}_{2^n})$, where $\deg_u b < \deg_u a \leq 2$, and $a(u)$ is monic. For instance, if $\deg_u a = 2$, then $a(u) = a_0 + a_1 u + u^2$ and $b(u) = b_0 + b_1 u$ with coefficients $a_0, a_1, b_0, b_1 \in \mathbb{F}_{2^n}$. If $\deg_u a < 2$, then we even need less coefficients for $a(u)$ and $b(u)$. We assume that the coefficients of $a(u)$ and $b(u)$ are represented with respect to a normal basis of $\mathbb{F}_{2^n}$ over $\mathbb{F}_2$.

Algorithm 3.2. *(Computing Scalar Multiples)*
`INPUT:` $a(u), b(u) \in \mathbb{F}_{2^n}[u]$ *such that* $D = [a(u), b(u)] \in \mathbb{J}_{C_1}(\mathbb{F}_{2^n})$*;* $c_0, \ldots, c_{l-1} \in \{0, \pm 1, \pm 2, \pm 3\}$ *with*

$$m \equiv \sum_{i=0}^{l-1} c_i \tau^i \pmod{\tau^n - 1} \ .$$

`OUTPUT:` $s(u), t(u) \in \mathbb{F}_{2^n}[u]$ *such that* $mD = [s(u), t(u)] \in \mathbb{J}_{C_1}(\mathbb{F}_{2^n})$.

1. *Precompute* $2D$ *and* $3D$.
2. $H \leftarrow c_{l-1} D = [s(u), t(u)]$;
3. `For` i `from` $l-2$ `downto` 0 `do:`
   (a) $H \leftarrow H^\phi = [s(u)^\phi, t(u)^\phi]$;
   (b) `If`$(c_i \neq 0)$ $H \leftarrow H + c_i D$;
4. `Output`$(s(u), t(u))$.

Note that the operation $H = H^\phi$ is nothing else than cyclic shifting of at most 4 coefficients of $s(u)$ and $t(u)$, if $s(u)$ and $t(u)$ are represented with respect to a normal basis. In general, $s(u) = s_0 + s_1 u + u^2$, $t(u) = t_0 + t_1 u$ with $s_0, s_1, t_0, t_1 \in \mathbb{F}_{2^n}$. Thus, $H^\phi = [s_0^2 + s_1^2 u + u^2, t_0^2 + t_1^2 u]$, and the computation of H^ϕ needs four cyclic shifts of elements in $\mathbb{F}_{2^n}$.

3.5 Computing the Cardinality of the Jacobian

The main step in computing the cardinality of $\mathbb{J}_{C_1}(\mathbb{F}_{2^n})$ is to determine the characteristic polynomial of the Frobenius. Since C_1 is defined over $\mathbb{F}_2$, this can be accomplished very easily. The Theorem of Weil then gives us immediately that

$$\#\mathbb{J}_{C_1}(\mathbb{F}_{2^n}) = \prod_{i=1}^{4} (1 - \tau_i^n) = \big((1 + 2^n) - (\tau_1^n + \tau_2^n)\big)\big((1 + 2^n) - (\tau_3^n + \tau_4^n)\big) \ .$$

It appears that an explicit formula for the cardinality of Jacobians over $\mathbb{F}_{2^n}$ can only be developed for supersingular curves (see [10]). For our curve, we have to proceed differently. A way of evaluating products of the form $\prod_{i=1}^{r}(1 - \alpha_i^n)$, where $\alpha_i,\ i = 1, \ldots, r$ are the roots of an arbitrary polynomial, was suggested by Pierce [21] and Lehmer [14]. Our case is very special. We can exploit the

Table 1. Average length and density

n	average length	average density	n	average length	average density
61	62.38	0.5460	97	98.34	0.5437
67	68.36	0.5458	101	102.36	0.5433
71	72.38	0.5455	103	104.31	0.5429
73	74.35	0.5449	107	108.33	0.5434
79	80.33	0.5445	109	110.34	0.5424
83	84.35	0.5440	113	114.35	0.5427
89	90.32	0.5441			

additional structure of $P(T)$ to derive recursions of lower order and using integer arithmetic only. If we assume that $\tau_1^n + \tau_2^n = A_n + \mu_1 B_n$, then we get for $n \geq 2$ that $\tau_1^n + \tau_2^n = (4B_{n-1} - 2A_{n-2}) + \mu_1(A_{n-1} + B_{n-1} - 2B_{n-2})$. Equating coefficients yields the following recursions. Put $A_0 = 2$, $A_1 = 0$, $B_0 = 0$, and $B_1 = 1$. For $n \geq 2$ we define $A_n = 4B_{n-1} - 2A_{n-2}$ and $B_n = A_{n-1} + B_{n-1} - 2B_{n-2}$. Then we have

$$\#\mathbb{J}_{C_1}(\mathbb{F}_{2^n}) = (1 + 2^n)^2 - (2A_n + B_n)(1 + 2^n) + (A_n^2 + A_n B_n - 4B_n^2) \ .$$

A similar approach leads to formulas for $\#\mathbb{J}_{C_0}(\mathbb{F}_{2^n})$.

4 Experimental Results

This section contains three tables. Table 1 describes the length and the density of reduced τ-adic expansions For each prime $n \in \{61, \ldots, 113\}$, we generated 10000 random integers m in the range $0 < m < \#\mathbb{J}_{C_1}(\mathbb{F}_{2^n})$ and computed the reduced τ-adic representation. If d denotes the number of the nonzero coefficients c_i, and l the length of the representation, the quotient d/l is its density.

The value $n + \frac{4}{3}$ seems to be a good approximation for the expected length l of a reduced τ-adic expansion. The asymptotic density (obtained by combinatorial means) is $\frac{489}{910} \sim 0.537$. The experiments provide evidence that the density is approaching the expected bound, so that the number of nonzero coefficients c_i is approximately $\frac{489}{910}(n + \frac{4}{3})$. Therefore, Algorithm 3.2 for computing multiples mD for $D \in \mathbb{J}_{C_1}(\mathbb{F}_{2^n})$ needs about $\frac{5}{9}n$ additions of reduced divisors, while the shift operations are essentially for free. The double-and-add-method for $\mathbb{J}_{C_1}(\mathbb{F}_{2^n})$ needs about $2n$ doubles and n additions of reduced divisors, so that the τ-adic method leads to a speed-up by a factor of

$$\frac{3n}{\frac{489n}{910}} \sim 5.5 \ .$$

In Table 2 and 3, respectively, we list examples for factorizations of $\#J_{C_1}(\mathbb{F}_{2^n})$ and $\#J_{C_0}(\mathbb{F}_{2^n})$. We only considered the cases where n takes on prime values in the range 61 to 113 and where the cardinalities of Jacobians contain a large prime factor.

Table 2. Computing the cardinality of the Jacobian $\mathbb{J}_{C_1}(\mathbb{F}_{2^n})$

n	$\#\mathbb{J}_{C_1}(\mathbb{F}_{2^n})$
61	5316911976894487061973100640561324954
	$= 2 \cdot 2658455988447243530986550320280662477$
67	217780714811051400238322367953881227296 42
	$= 2 \cdot 3217 \cdot 33848416974052129350065646247106190 13$
97	251084069415467379963903548856251249433764395706842274777 54
	$= 2 \cdot 389 \cdot 1747 \cdot 184733924638688269103187946767540719407169099070196 19$
103	102844034832575383397207943835010553634640254575820398436691978
	$= 2 \cdot 47381 \cdot 108528771904957032773905092584591453994892736092337011076 9$
109	42124916667422880025110033012494514026132187984275004118977699228 2
	$= 2 \cdot 2617 \cdot 620764811 \cdot 1296517091071062805290214064753207111492717872789885 43$
113	10783978666860255743164659534768246152128560543003808709952838673676 2
	$= 2 \cdot 5391989333430127871582329767384123076064280271501904354976419336838 1$

Table 3. Computing the cardinality of the Jacobian $\mathbb{J}_{C_0}(\mathbb{F}_{2^n})$

n	$\#\mathbb{J}_{C_0}(\mathbb{F}_{2^n})$
67	217780714847749832994997151829687427694 96
	$= 2^3 \cdot 27222589355968729124374643978710928461 87$
89	38312388521645115721969038261434081449988961294626400 8
	$= 2^3 \cdot 179 \cdot 1069 \cdot 83091469 \cdot 30120492445235537115154202849824591399 79$

5 Improvements

Following the idea of Koblitz [12], we modified our set of possible coefficients and used the set

$$R' = \{0, \pm 1, \pm 2, \pm(1+\tau), \pm(1-\tau), \pm(1-2\tau), \pm 2+\tau\}$$

as the domain of coefficients. Accepting the cost of 6 precomputations and storing these elements (instead of only 2 for set R), this choice enables us to realize a sparse τ-adic expansion in the sense that no two consecutive coefficients are nonzero (cf. [24]). Using u as in the following table we force $a+b\tau+c\tau^2+d\tau^3-u$ to be divisible by τ^2, i. e. the next coefficient will be zero. If $4|a$ then $u = 0$, else take

$b \bmod 4 \backslash a \bmod 8$	1	2	3	5	6	7
0	1	2	$-(1-2\tau)$	$1-2\tau$	-2	-1
1	$1+\tau$	$2+\tau$	$-(1+\tau)$	$1-\tau$	$-2+\tau$	$-(1-\tau)$
2	$1-2\tau$	-2	-1	1	2	$-(1-2\tau)$
3	$1-\tau$	$-2+\tau$	$-(1-\tau)$	$1+\tau$	$2+\tau$	$-(1+\tau)$

By using this modified version of the τ-adic expansion, the average length of the reduced τ-adic representations was $< n+2$ for an extension of degree n. The expected density for this set of coefficients is $\frac{3}{7} \sim 0.42857$. In Table 4, we present our experimental results. The generation of the integers m was identical to the one in Table 1. The difference lies in the choice of the set R' which yields new τ-adic expansions.

Table 4. Average length and density

n	average length	average density	n	average length	average density
61	63.02	0.4284	97	99.67	0.4177
67	69.00	0.4275	101	102.95	0.4287
71	72.98	0.4288	103	104.93	0.4289
73	32.15	0.4287	107	109.05	0.4288
79	81.01	0.4287	109	111.01	0.4287
83	84.99	0.4286	113	114.96	0.4285
89	91.00	0.4288			

Therefore with this set R' we obtain a speed-up by a factor of

$$\frac{3n}{\frac{3n}{7}} = 7$$

with respect to the binary expansion on the cost of more storing and precomputations.

References

1. Cantor, D. G.: Computing in the Jacobian of a Hyperelliptic Curve. Math. Comp. **48** (1987) 95–101
2. Diffie, W., Hellman, M. E.: New Directions in Cryptography. IEEE Trans. Inform. Theory **22** (1976) 644–654
3. ElGamal, T.: A Public-Key Cryptosystem and a Signature Scheme Based on Discrete Logarithms. IEEE Trans. Inform. Theory **31** (1985) 469–472
4. Frey, G., Rück, H.-G.: A Remark Concerning m-Divisibility and the Discrete Logarithm Problem in the Divisor Class Group of Curves. Math. Comp. **62** (1994) 865–874
5. Gallant, R., Lambert, R., Vanstone, S.: Improving the Parallelized Pollard Lambda Search on Binary Anomalous Curves. To Appear in Math. Comp. http://www.certicom.com/chal/download/paper.ps
6. Gaudry, P., Hess, F., Smart, N.: Constructive and Destructive Facets of Weil Descent on Elliptic Curves, preprint, 1999.
7. Gaudry, P., Morain, F., Duursma, I.: Speeding Up the Discrete Log Computation on Curves with Automorphisms In: Proc. of The Mathematics of Public Key Cryptography. Fields-Institute Toronto (1999)
8. Gordon, D.: A Survey of Fast Exponentiation Methods. J. Algorithms **27** (1998) 129–146
9. Koblitz, N.: Elliptic Curve Cryptosystems. Math. Comp. **48** (1987) 203–209
10. Koblitz, N.: Hyperelliptic Cryptosystems. J. Cryptology **1** (1989) 139–150
11. Koblitz, N.: CM Curves with Good Cryptographic Properties. In: Advances in Cryptology – Crypto '91. LNCS, Vol. 576. Springer-Verlag, Berlin Heidelberg New York (1992) 279–287
12. Koblitz, N.: An Elliptic Curve Implementation of the Finite Field Digital Signature Algorithm. In: Advances in Cryptology – Crypto '98. Lecture Notes in Computer Science, Vol. 1462. Springer-Verlag, Berlin Heidelberg New York (1998) 327–337

13. Lange, T.: Efficient Arithmetic on Hyperelliptic Koblitz Curves. preprint, 2000.
14. Lehmer, D.H.: Factorization of Certain Cyclotomic Functions. Ann. Math. **34**(1933) 461-479
15. Meier, W., Staffelbach, O.: Efficient Multiplication on Certain Nonsupersingular Elliptic Curves. In: Advances in Cryptology – Crypto '92. LNCS, Vol. 740. Springer-Verlag, Berlin Heidelberg New York (1993) 333–344
16. Menezes, A., Wu, Y., Zuccherato, R.: An Elementary Introduction to Hyperelliptic Curves. In: Koblitz, N.: Algebraic Aspects of Cryptography. Springer-Verlag, Berlin Heidelberg New York (1998)
17. Miller, V.: Use of Elliptic Curves in Cryptography. In: Advances in Cryptology – Crypto '85. LNCS, Vol. 218. Springer-Verlag, Berlin Heidelberg New York (1986) 417–426
18. Müller, V., Stein, A., Thiel, C.: Computing Discrete Logarithms in Real Quadratic Congruence Function Fields of Large Genus. Math. Comp. **68** (1999) 807–822
19. Mumford, D.: Tata Lectures on Theta I, II. Birkhäuser-Verlag, Boston (1983/84)
20. van Oorschot, P., Wiener, M. J.: Parallel Collision Search with Cryptanalytic Applications. J. Cryptology **12** (1999) 1–28
21. Pierce, T.A.: The Numerical Factors of the Arithmetic Forms $\prod_{i=1}^{n}(1 \pm \alpha_i^m)$. Ann. Math. **18**(1916), 53-64.
22. Pollard, J. M.: Kangaroos, Monopoly and Discrete Logarithms. To appear in J. Cryptology.
23. Solinas, J.: An Improved Algorithm for Arithmetic on a Family of Elliptic Curves. In: Advances in Cryptology – Crypto '97. LNCS, Vol. 1294. Springer-Verlag, Berlin Heidelberg New York (1997) 357–371
24. Solinas, J.: Efficient Arithmetic on Koblitz Curves. Techn. Report CORR 99-09, University of Waterloo (1999), 61 pages. http://www.cacr.math.uwaterloo.ca
25. Stein, A.: Sharp Upper Bounds for Arithmetics in Hyperelliptic Function Fields. Techn. Report CORR 99-23, University of Waterloo (1999), 68 pages. Available at http://www.cacr.math.uwaterloo.ca
26. Stichtenoth, H.: Algebraic Function Fields and Codes. Springer-Verlag, Berlin Heidelberg New York (1993)
27. Teske, E.: Speeding up Pollard's rho method for computing discrete logarithms. In: Algorithmic Number Theory Seminar ANTS-III. LNCS, Vol. 1423. Springer-Verlag, Berlin Heidelberg New York (1998) 541–554
28. Wiener, M., Zuccerato, R.: Faster Attacks on Elliptic Curve Cryptosystems. In: Proceedings of SAC, Workshop on Selected Areas in Cryptography. LNCS, Springer-Verlag, Berlin Heidelberg New York (1998).