

## ISOMORPHISMS BETWEEN ARTIN-SCHREIER TOWERS

JEAN-MARC COUVEIGNES

ABSTRACT. We give a method for efficiently computing isomorphisms between towers of Artin-Schreier extensions over a finite field. We find that isomorphisms between towers of degree  $p^n$  over a fixed field  $\mathbb{F}_q$  can be computed, composed, and inverted in time essentially linear in  $p^n$ . The method relies on an approximation process.

### 1. INTRODUCTION

Let  $\mathbb{F}_q$  be a finite field with  $q = p^d$  elements. Let  $L_n$  be an extension of degree  $p^n$  of  $\mathbb{F}_q$  given as a tower

$$(1) \quad L_n \supset L_{n-1} \supset \cdots \supset L_1 \supset L_0 = \mathbb{F}_q$$

of nontrivial Artin-Schreier extensions each defined by

$$L_{k+1} = L_k(x_{k+1}) \text{ with } x_{k+1}^p - x_{k+1} - a_k = 0 \text{ and } a_k \in L_k.$$

We call  $n$  the *length* of the tower.

Artin-Schreier towers naturally arise in computational algebraic geometry. In particular, let  $G = \text{Gal}(\overline{\mathbb{F}_q}/\mathbb{F}_q)$  be the absolute Galois group of  $\mathbb{F}_q$ . Morphisms between abelian varieties  $A$  and  $B$  defined over  $\mathbb{F}_q$  induce  $G$ -morphisms between the Tate modules  $\mathcal{T}_\ell(A)$  and  $\mathcal{T}_\ell(B)$ . If  $\ell \neq p$ , this correspondence is known to be bijective by a theorem of Tate [8]. If  $\ell = p$ ,  $A$  is simple, and  $\mathcal{T}_\ell(A)$  is nonzero, then the correspondence is injective. Assume the  $p$ -torsion of  $A$  and  $B$  is defined over  $\mathbb{F}_q$ . One can easily show that the definition field  $L_k$  of the  $p^{k+1}$ -torsion of  $A$  is an extension of  $L_0 = \mathbb{F}_q$  with degree dividing  $p^k$ . Similarly the definition field  $M_k$  of the  $p^{k+1}$ -torsion of  $B$  is an extension of  $M_0 = L_0 = \mathbb{F}_q$  with degree dividing  $p^k$ . Assuming the existence of an isogeny between  $A$  and  $B$  with prime to  $p$  degree, the fields  $L_k$  and  $M_k$  are isomorphic. These fields can be constructed by taking successive preimages of a  $p$ -torsion point by separable isogenies of degree  $p$ . Thus they naturally come as Artin-Schreier towers. In the case of nonsupersingular elliptic curves, such isogenies are described in terms of Hasse functions. If we are looking for an isogeny with a given prime to  $p$  degree between  $A$  and  $B$ , we can compute it by interpolation at enough  $p^k$ -torsion points. This reduces to computing an isomorphism between the Artin-Schreier towers we have on each side. This method is of special interest for computing the cardinality of ordinary elliptic curves with the Schoof-Elkies-Atkin algorithm. See [2] where the fastest known algorithm for this purpose is given, assuming the characteristic  $p$  is fixed. Surveys on these questions are in [6, 4, 3, 5].

---

Received by the editor February 5, 1997 and, in revised form, July 24, 1998.  
1991 *Mathematics Subject Classification*. Primary 11Y40; Secondary 12E20.

©2000 American Mathematical Society

We shall prove the following

**Theorem 1.** *An isomorphism between two Artin-Schreier towers  $L_n$  and  $M_n$  of degree  $p^n$  over  $\mathbb{F}_q = L_0 = M_0$  can be computed in time  $O(n^6 p^n)$  multiplications in  $\mathbb{F}_q$  for fixed  $q$  and  $n \rightarrow \infty$ .*

Computational aspects of Artin-Schreier towers have already been studied by D. G. Cantor in [1]. For any integer  $u$  in  $[0, p^n[$  with  $p$ -adic expansion  $u = u_1 + u_2 p + \dots + u_n p^{n-1}$  he sets  $\chi_u = x_1^{u_1} x_2^{u_2} \dots x_n^{u_n}$ . The monomials  $(\chi_u)_{0 \leq u < p^n}$  form a basis  $\mathcal{X}$  of the  $L_0$ -vector space  $L_k$ . If  $a_0 = 1$  and  $a_k = \chi_{p^k-1} + \sum_{u=0}^{p^k-2} c_u \chi_u$  with all the  $c_u \in \mathbb{F}_q$ , we say that the tower in formula (1) is a *Cantor tower*. One of the results in [1] is that for any prime  $p$  there exists a constant  $K_p$  such that two elements in a Cantor tower of length  $n$  over  $\mathbb{F}_p$  can be multiplied at the expense of  $K_p n^2 p^n$  operations in  $\mathbb{F}_p$ . The same holds for Cantor towers over a nonnecessarily prime field  $\mathbb{F}_q$ . We shall need this result and the corresponding algorithm. In order to compute an isomorphism between two Artin-Schreier towers, we shall first compute isomorphisms between each of the two towers and a given Cantor tower. The expected isomorphism will then be obtained as a composition of these two isomorphisms. It is the purpose of Lemma 1 to state how efficiently isomorphisms between Artin-Schreier towers can be dealt with.

If  $\alpha, \beta \in L_n$ , we define the *écart*  $\mathbf{d}(\alpha, \beta)$  to be the logarithm (with base  $p$ ) of the degree of the extension  $\mathbb{F}_q(\alpha - \beta)/\mathbb{F}_q$ . The triangle inequality is easily checked. Note that  $\mathbf{d}$  is not a distance since  $\mathbf{d}(\alpha, \beta) = 0$  if and only if  $\alpha - \beta$  is in  $\mathbb{F}_q$ . On the other hand,  $\mathbf{d}$  is invariant under translation.

For any two positive integers  $i$  and  $j$  we define the following polynomials in  $\mathbb{F}_p[X]$   $\Phi_i(X) = X^{p^i}$  and  $\wp_i(X) = X^{p^i} - X$  and  $T_{i,j} = X + X^{p^j} + X^{p^{2j}} + \dots + X^{p^{(i-1)j}}$ . The polynomial  $\wp_i$  is usually called an isogeny [7]. To simplify we set  $T_i = T_{i,1}$ . We have the trivial relations

$$\wp_i \circ \wp_j = \wp_j \circ \wp_i \text{ and } \wp_j \circ T_{i,j} = T_{i,j} \circ \wp_j = \wp_{ij} \text{ and } T_{j,k} \circ T_{i,jk} = T_{ij,k}.$$

If  $\mathcal{K} \subset \mathcal{L}$  is an extension of finite fields with cardinalities  $p^j$  and  $p^{ij}$ , respectively, we have the following exact sequence of  $\mathcal{K}$ -vector spaces:

$$0 \rightarrow \mathcal{K} \rightarrow \mathcal{L} \xrightarrow{\wp_j} \mathcal{L} \xrightarrow{T_{i,j}} \mathcal{K} \rightarrow 0.$$

Assume we are looking for an isomorphism

$$\iota : M_n \rightarrow L_n$$

between two Artin-Schreier towers  $L_n$  and  $M_n$ , with  $M_n$  defined by

$$M_n \supset M_{n-1} \supset \dots \supset M_1 \supset M_0 = \mathbb{F}_q$$

and

$$M_{k+1} = M_k(y_{k+1}) \text{ and } y_{k+1}^p - y_{k+1} - b_k = 0 \text{ with } b_k \in M_k.$$

We define  $\zeta_u = y_1^{u_1} y_2^{u_2} \dots y_n^{u_n}$  similarly to  $\chi_u$ . We may assume that an isomorphism has already been constructed between  $L_{n-1}$  and  $M_{n-1}$ . In order to extend it, we have to solve in  $L_n$  an Artin-Schreier equation.

Consider such an equation

$$(2) \quad \wp_1(Y) = Y^p - Y = \beta.$$

with  $\beta \in L_n$  and  $\text{Tr}_{L_n/\mathbb{F}_p}(\beta) = 0$ .

This is a linear equation over  $\mathbb{F}_p$ . The corresponding linear system of dimension  $dp^n$  over  $\mathbb{F}_p$  can be solved with Gauss's algorithm at the expense of  $O(d^3p^{3n})$  operations in  $\mathbb{F}_p$ . We notice, however, that equation (2) implies

$$(3) \quad \wp_i(Y) = Y^{p^i} - Y = \beta + \beta^p + \dots + \beta^{p^{i-1}} = T_i(\beta)$$

which is linear over the intermediate field  $\mathbb{F}_{p^i}$ . The corresponding linear system of dimension  $dp^n/i$  over  $\mathbb{F}_{p^i}$  can be solved with Gauss's algorithm at the expense of  $O(d^3p^{3n}/i^3)$  operations in  $\mathbb{F}_{p^i}$ . This is better when multiplication is fast in  $L_n$  (e.g., when  $L_n$  is a Cantor tower).

Equation (3), of course, does not imply equation (2) but if we know a solution  $\gamma$  to equation (3) and set  $Y = Z + \gamma$  in equation (2) we get

$$\wp_1(Z) = Z^p - Z = \beta - \gamma^p + \gamma.$$

Let  $\delta = \beta - \gamma^p + \gamma$ . We have  $\wp_i(\delta) = \wp_i(\beta) - \wp_i(\wp_1(\gamma)) = \wp_i(\beta) - \wp_1(\wp_i(\gamma)) = \wp_i(\beta) - \wp_1(T_i(\beta)) = 0$  so  $\delta \in \mathbb{F}_{p^i}$ . We also check easily that  $T_i(\delta) = T_i(\beta) - \wp_1(T_i(\gamma)) = T_i(\beta) - \wp_i(\gamma) = 0$ . We conclude that the *écart* between  $\gamma$  and any solution of (2) is at most  $\log_p(i/\text{pgcd}(d, i))$ . We say that  $\delta$  is an approximate solution to equation (2) with accuracy  $\log_p(i/\text{pgcd}(i, d))$ .

Since our strategy is to deal with the smallest possible matrices, we shall take  $i = dp^{n-1}$ . This way, for  $\beta \in L_n$  and  $\text{Tr}_{L_n/\mathbb{F}_p}(\beta) = 0$ , a solution to  $Y^p - Y = \beta$  can be found in three steps:

1. Compute  $B = T_{dp^{n-1}}(\beta)$ .
2. Find a solution  $\gamma$  to  $Y^{p^{dp^{n-1}}} - Y = B$  which amounts to solving a linear system of dimension  $p$  over  $L_{n-1}$ .
3. Solve  $Z^p - Z = \delta$ , where  $\delta = \beta - \gamma^p + \gamma$  is in  $L_{n-1}$  and  $\text{Tr}_{L_{n-1}/\mathbb{F}_p}(\delta) = 0$ .

And the same method is applied recursively to the equation in step 3. After  $k$  steps, we obtain an approximate solution to equation (2) with accuracy  $n - k$ . After  $n$  steps, we reduce to an Artin-Schreier equation over the base field  $\mathbb{F}_q$ .

In the rest of this paper, we provide details and a complexity analysis for the algorithm sketched above.

## 2. ARTIN-SCHREIER TOWERS

We recall a few elementary facts about Artin-Schreier extensions. Let  $\mathcal{K}$  be a field of characteristic  $p$ , not necessarily finite, and  $\mathcal{L} = \mathcal{K}[X]/(X^p - X - \alpha)$  an Artin-Schreier extension. Set  $x = X \bmod X^p - X - \alpha$ . Its conjugates are the  $x + c$  with  $c \in \mathbb{F}_p$ . The trace is given by

$$\text{Tr}_{\mathcal{L}/\mathcal{K}}\left(\sum_{0 \leq i \leq p-1} u_i x^i\right) = -u_{p-1} \text{ when } u_i \in \mathcal{K}$$

and the dual basis of  $(1, x, x^2, \dots, x^{p-1})$  is  $(-x^{p-1} + 1, -x^{p-2}, -x^{p-3}, \dots, -x, -1)$ .

In such an Artin-Schreier extension,  $p$ -powers are easy to compute. Indeed

$$(4) \quad x^{ip^h} = (x + T_h(\alpha))^i.$$

In particular if  $\mathcal{K}$  is the field  $\mathbb{F}_q$  with  $q = p^d$  elements then

$$x^{iq} = (x + \text{Tr}_{\mathbb{F}_q/\mathbb{F}_p}(\alpha))^i,$$

and  $\text{Tr}_{\mathbb{F}_q/\mathbb{F}_p}(\alpha)$  is in  $\mathbb{F}_p$ . Thus the  $p \times p$  matrix of the Frobenius automorphism  $x \mapsto x^q$  has coefficients in  $\mathbb{F}_p$ .

We shall first prove a few complexity estimates concerning basic computations with isomorphisms between Artin-Schreier towers over finite fields.

We consider an isomorphism  $\iota$  between two towers  $L_n$  and  $M_n$ :

$$\iota : M_n \rightarrow L_n.$$

The computer representation of  $\iota$  will consist of the images of the  $y_k^i$  by  $\iota$  for  $0 \leq i \leq p-1$  and  $1 \leq k \leq n$ .

We shall see that this representation is very efficient. For  $0 \leq k \leq n$ , we denote by  $\mathcal{C}_\times^L(k)$  the complexity of multiplication in  $L_k$ . This complexity is given as a number of multiplications in the base field  $\mathbb{F}_q$ , disregarding additions. We denote by  $\mathcal{C}_\times^M(k)$  the complexity of multiplication in  $M_k$ . Let  $\mathcal{C}_i(n)$  be the cost of evaluating  $\iota$  at some  $\mu$  in  $M_n$ . Let  $\mathcal{C}_i^\bullet(n)$  be the complexity of computing  $\iota^{-1}(\nu)$  for  $\nu$  in  $L_n$ .

We shall first prove the following

**Lemma 1.** *Given an isomorphism  $\iota : M_n \rightarrow L_n$  between two Artin-Schreier towers, we have, with the notation given above*

$$\begin{aligned} (5) \quad & \mathcal{C}_i(n) \leq pn \mathcal{C}_\times^L(n), \\ (6) \quad & \mathcal{C}_i^\bullet(n) \leq 2np^3 \mathcal{C}_\times^L(n), \\ (7) \quad & \mathcal{C}_\times^M(n) \leq 4np^3 \mathcal{C}_\times^L(n). \end{aligned}$$

We first prove inequality (5). For  $\mu \in M_n$ , let us write  $\mu = \sum_{0 \leq i \leq p-1} \mu_i y_n^i$  with  $\mu_i \in M_{n-1}$ . Then  $\iota(\mu) = \sum_i \iota(\mu_i) \iota(y_n^i)$  and since we have stored the  $\iota(y_n^i)$ , we reduce to computing  $p$  multiplications in  $L_n$  and the images  $\iota(\mu_i)$ . Therefore

$$\mathcal{C}_i(n) \leq p(\mathcal{C}_i(n-1) + \mathcal{C}_\times^L(n))$$

and the result follows iterating the above inequality and using the easy inequality

$$\mathcal{C}_\times^L(n) \geq p \mathcal{C}_\times^L(n-1).$$

In order to compute the inverse image of  $\nu \in L_n$ , we first express  $\nu$  as a linear combination

$$(8) \quad \nu = \sum_{0 \leq i \leq p-1} \nu_i \iota(y_n^i)$$

with  $\nu_i \in L_{n-1}$  for all  $i$ . This is achieved at the expense of  $2p^3$  multiplications in  $L_n$  using Gauss's algorithm. From equation (8) we deduce

$$\iota^{-1}(\nu) = \sum_{0 \leq i \leq p-1} \iota^{-1}(\nu_i) y_n^i.$$

We thus reduce to computing the  $p$  preimages of the  $\nu_i \in L_{n-1}$ . Therefore

$$\mathcal{C}_i^\bullet(n) \leq 2p^3 \mathcal{C}_\times^L(n) + p \mathcal{C}_i^\bullet(n-1)$$

and inequality (6) follows.

Inequality (7) follows easily from inequalities (5) and (6). This shows that if we can multiply efficiently in  $L_n$ , the knowledge of  $\iota$  allows fast multiplication in  $M_n$  as well.

The crucial step in our isomorphism computations will be the evaluation of polynomials  $T_{i,j}$  at numbers  $\mu$  that are not necessarily in  $\mathbb{F}_{p^{ij}}$ . Lemma 2 states how efficiently one can compute  $\Phi_{dp^l}(\mu) = \mu^{p^{dpl}}$  and  $T_{dp^l}(\mu)$  for  $\mu \in L_k$  and  $0 \leq l \leq k$ .

We denote by  $\mathcal{C}_\Phi^L(l, k)$  the complexity of computing  $\Phi_{dp^l}(\mu)$  for  $\mu \in L_k$ . We denote by  $\mathcal{C}_T^L(l, k)$  the complexity of computing  $T_{dp^l}(\mu)$  for  $\mu \in L_k$ . In order to compute  $T_{dp^l}(\mu)$  we notice that

$$(9) \quad T_{dp^l} = T_d \circ T_{p,d} \circ \cdots \circ T_{p,dp^{l-2}} \circ T_{p,dp^{l-1}}.$$

Using this formula we obtain

$$(10) \quad \mathcal{C}_T^L(l, k) \leq p(\mathcal{C}_\Phi^L(l-1, k) + \mathcal{C}_\Phi^L(l-2, k) + \cdots + \mathcal{C}_\Phi^L(1, k) + \mathcal{C}_\Phi^L(0, k)) + pd\mathcal{C}_\times^L(k).$$

If we now want to compute  $\Phi_{dp^l}(\mu)$  we use formula (4). Writing  $\mu = \sum_{0 \leq i \leq p-1} \mu_i x_k^i$  we have

$$(11) \quad \Phi_{dp^l}(\mu) = \sum_{0 \leq i \leq p-1} \Phi_{dp^l}(\mu_i) \Phi_{dp^l}(x_k^i) = \sum_{0 \leq i \leq p-1} \Phi_{dp^l}(\mu_i) (x_k + T_{dp^l}(a_{k-1}))^i$$

since  $x_k^p - x_k = a_{k-1}$ .

We first assume that we already computed and stored the  $T_{dp^l}(a_\kappa)$  and their first  $p$  powers for all  $l$  and  $\kappa$  such that  $0 \leq l \leq \kappa < k$ , which is the same as computing the expansions of polynomials  $(x + T_{dp^l}(a_\kappa))^i$  for  $0 \leq i \leq p-1$ .

We call  $\tilde{\mathcal{C}}_\Phi^L(l, k)$  the complexity of computing  $\Phi_{dp^l}(\mu)$  for  $\mu \in L_k$  under this assumption. We define  $\tilde{\mathcal{C}}_T^L(l, k)$  to be the complexity of computing  $T_{dp^l}(\mu)$  for  $\mu \in L_k$  in the same situation.

From equation (11) we deduce

$$\tilde{\mathcal{C}}_\Phi^L(l, k) \leq p\tilde{\mathcal{C}}_\Phi^L(l, k-1) + p^2\mathcal{C}_\times^L(k-1).$$

Since  $\mathcal{C}_\Phi^L(l, k) = 0$  as soon as  $l \geq k$ , we obtain

$$\tilde{\mathcal{C}}_\Phi^L(l, k) \leq p(k-l)\mathcal{C}_\times^L(k),$$

and from equation (10) and the definition of  $T_{dp^l}$

$$(12) \quad \tilde{\mathcal{C}}_T^L(l, k) \leq (p^2kl + pd)\mathcal{C}_\times^L(k) \leq 2p^2kld\mathcal{C}_\times^L(k).$$

We now bound the cost  $\mathcal{C}_{\text{init}}^L(k)$  of precomputing all the  $T_{dp^l}(a_\kappa)$  and their first  $p$  powers for all  $l$  and  $\kappa$  such that  $0 \leq l \leq \kappa < k$ .

We first bound  $\mathcal{C}_{\text{init}}^L(k+1) - \mathcal{C}_{\text{init}}^L(k)$ . Indeed if we already know the  $T_{dp^l}(a_\kappa)$  and their first  $p$  powers for all  $0 \leq l \leq \kappa < k$ , then computing the  $T_{dp^l}(a_k)$  for all  $0 \leq l \leq k$  will require less than  $2(k+1)p^2k^2d\mathcal{C}_\times^L(k)$  multiplications (using formula (12)) and computing the powers will take time  $p(k+1)\mathcal{C}_\times^L(k)$ . Therefore

$$\mathcal{C}_{\text{init}}^L(k+1) \leq \mathcal{C}_{\text{init}}^L(k) + (k+1)(p + 2p^2k^2d)\mathcal{C}_\times^L(k).$$

We obtain

$$\mathcal{C}_{\text{init}}^L(k) \leq 6p^2k^3d\mathcal{C}_\times^L(k).$$

**Lemma 2.** For  $0 \leq l \leq k$  and for any  $\mu$  in  $L_k$ , one can compute  $\Phi_{dp^l}(\mu)$  (resp.  $T_{dp^l}(\mu)$ ) in time  $\tilde{\mathcal{C}}_\Phi^L(l, k)$  (resp.  $\tilde{\mathcal{C}}_T^L(l, k)$ ) with

$$(13) \quad \tilde{\mathcal{C}}_\Phi^L(l, k) \leq p(k-l)\mathcal{C}_\times^L(k),$$

$$(14) \quad \tilde{\mathcal{C}}_T^L(l, k) \leq 2p^2kld\mathcal{C}_\times^L(k),$$

using data that only depend on  $L_k$  and can be computed once and for all in time  $\mathcal{C}_{\text{init}}^L(k)$  with

$$(15) \quad \mathcal{C}_{\text{init}}^L(k) \leq 6p^2k^3d\mathcal{C}_{\times}^L(k).$$

We call  $\mathcal{C}_{AS}^L(n)$  the complexity of solving equation (2) in  $L_n$  for  $\beta \in L_n$  and  $\text{Tr}_{L_n/\mathbb{F}_p}(\beta) = T_{dp^n}(\beta) = 0$ . We shall adopt the three steps strategy described in the introduction.

We first compute and store the  $T_{dp^l}(a_\kappa)$  for all  $0 \leq l \leq \kappa < n$ . This takes time  $\mathcal{C}_{\text{init}}^L(n)$ . We call  $\tilde{\mathcal{C}}_{AS}^L(n)$  the complexity of solving equation (2) once all this precomputation has been done.

In these conditions, step 1 (the computation of  $B = T_{dp^{n-1}}(\beta)$ ) will take time  $\tilde{\mathcal{C}}_T^L(n-1, n)$ .

The second step reduces to computing the  $p \times p$  matrix representing the  $L_{n-1}$ -linear map  $\wp_{dp^{n-1}} : L_n \rightarrow L_n$  in the basis  $(1, x_n, x_n^2, \dots, x_n^{p-1})$ . Using Gauss's algorithm, we then find a solution  $\gamma$  to the equation  $\wp_{dp^{n-1}}(\gamma) = B$ .

All this is achieved at the expense of  $p\tilde{\mathcal{C}}_{\Phi}^L(n-1, n) + 2p^3\mathcal{C}_{\times}^L(n-1)$  multiplications.

The third step is done in time  $p\mathcal{C}_{\times}^L(n) + \tilde{\mathcal{C}}_{AS}^L(n-1)$ . We thus have

$$\tilde{\mathcal{C}}_{AS}^L(n) \leq \tilde{\mathcal{C}}_{AS}^L(n-1) + \tilde{\mathcal{C}}_T^L(n-1, n) + p\tilde{\mathcal{C}}_{\Phi}^L(n-1, n) + 2p^3\mathcal{C}_{\times}^L(n-1) + p\mathcal{C}_{\times}^L(n),$$

and using Lemma 2,

$$\tilde{\mathcal{C}}_{AS}^L(n) \leq \tilde{\mathcal{C}}_{AS}^L(n-1) + 6p^2n^2d\mathcal{C}_{\times}^L(n).$$

Thus

$$(16) \quad \tilde{\mathcal{C}}_{AS}^L(n) \leq 12n^2p^2d\mathcal{C}_{\times}^L(n) + \mathcal{C}_{AS}.$$

where  $\mathcal{C}_{AS} = \mathcal{C}_{AS}^L(0)$  is the complexity of solving an Artin-Schreier equation in the base field  $\mathbb{F}_q$ .

We now want to compute an isomorphism between two Artin-Schreier towers of length  $n$  over  $\mathbb{F}_q$ :

$$L_n \supset L_{n-1} \supset \dots \supset L_1 \supset L_0 = \mathbb{F}_q$$

and

$$M_n \supset M_{n-1} \supset \dots \supset M_1 \supset M_0 = \mathbb{F}_q.$$

We look for an isomorphism  $\iota : M_n \rightarrow L_n$  given by  $\iota(y_k^i)$  for  $0 \leq i < p$  and  $0 \leq k \leq n$ .

We let the length  $k$  increase from 0 to  $n$ . We call  $\mathcal{C}_M^L(k)$  the complexity of computing an isomorphism from  $M_k$  to  $L_k$ . We call  $\tilde{\mathcal{C}}_M^L(k)$  the complexity of computing an isomorphism from  $M_k$  to  $L_k$  assuming the  $T_{dp^l}(a_\kappa)$  have been computed for all  $0 \leq l \leq \kappa < k$ . We want to bound  $\tilde{\mathcal{C}}_M^L(n) - \tilde{\mathcal{C}}_M^L(n-1)$ . Thus assume we have computed the isomorphism up to length  $n-1$ . In order to go further we have to solve the Artin-Schreier extension

$$(17) \quad Y^p - Y = \iota(b_{n-1})$$

over  $L_n$ . We first apply  $\iota$  to  $b_{n-1}$  in time  $\mathcal{C}_\iota(n-1)$ . Solving equation (17) takes time  $\tilde{\mathcal{C}}_{AS}^L(n)$ . We take  $\iota(y_n)$  to be one of the solutions we found. We then compute the powers  $\iota(y_n)^i$  for  $0 \leq i \leq p-1$ , which takes time  $p\mathcal{C}_{\times}^L(n)$ . We thus have

$$\tilde{\mathcal{C}}_M^L(n) \leq \tilde{\mathcal{C}}_M^L(n-1) + \mathcal{C}_\iota(n-1) + \tilde{\mathcal{C}}_{AS}^L(n) + p\mathcal{C}_{\times}^L(n),$$

and using Lemma 1 and inequality (16),

$$\tilde{\mathcal{C}}_M^L(n) \leq \tilde{\mathcal{C}}_M^L(n-1) + 14n^2p^2d\mathcal{C}_\times^L(n) + \mathcal{C}_{AS}.$$

Summing up we have

$$\tilde{\mathcal{C}}_M^L(n) \leq 28n^2p^2d\mathcal{C}_\times^L(n) + n\mathcal{C}_{AS},$$

and using (15),

$$(18) \quad \mathcal{C}_M^L(n) \leq 34n^3p^2d\mathcal{C}_\times^L(n) + n\mathcal{C}_{AS}.$$

Assume now we have a third Artin-Schreier tower  $N_n$  over  $\mathbb{F}_q$ . We shall relate the complexity  $\mathcal{C}_\times^L(n)$  of multiplication in  $L_n$  and the complexity  $\mathcal{C}_N^M(n)$  of computing an isomorphism from  $N_n$  to  $M_n$ . This makes sense in case  $L_n$  has been designed to allow fast multiplication (e.g.,  $L_n$  is a Cantor tower).

We first compute an isomorphism  $\iota_1$  from  $M_n$  to  $L_n$  at the expense of  $\mathcal{C}_M^L(n)$  multiplications in  $\mathbb{F}_q$ . We then compute an isomorphism  $\iota_2$  from  $N_n$  to  $M_n$  at the expense of

$$\mathcal{C}_N^M(n) \leq 34n^3p^2d\mathcal{C}_\times^M(n) + n\mathcal{C}_{AS}$$

multiplications in  $\mathbb{F}_q$ . Using inequality (18) and inequality (7) we find

**Lemma 3.** *Let  $L_n, M_n, N_n$  be three Artin-Schreier towers of length  $n$  over  $\mathbb{F}_q$  the field with  $q = p^d$  elements and let  $\mathcal{C}_\times^L(n)$  be the complexity of multiplication in  $L_n$ . Let  $\mathcal{C}_{AS}$  be the complexity of solving an Artin-Schreier equation in  $\mathbb{F}_q$ . An isomorphism between  $M_n$  and  $N_n$  can be found at the expense of  $\mathcal{C}_N^M(n)$  multiplications in  $\mathbb{F}_q$  with*

$$\mathcal{C}_N^M(n) \leq 170p^5n^4d\mathcal{C}_\times^L(n) + 2n\mathcal{C}_{AS}.$$

If we take  $L_n$  to be a Cantor tower we have  $\mathcal{C}_\times^L(n) \leq K_q n^2 p^n$ , where  $K_q$  only depends on  $q$ . Using the Berlekamp factorization algorithm we have  $\mathcal{C}_{AS} = O(p^3d)$ , and Theorem 1 follows.

#### REFERENCES

- [1] David G. Cantor, *On arithmetical algorithms over finite fields*, Journal of Combinatorics, series A **50** (1989), 285–300. MR **90f**:11100
- [2] Jean-Marc Couveignes, *Computing  $l$ -isogenies with the  $p$ -torsion*, Algorithmic Number Theory, A.N.T.S. II (H. Cohen, ed.), vol. 1122, Springer, 1996, pp. 59–65. MR **98j**:11046
- [3] Noam D. Elkies, *Elliptic and modular curves over finite fields and related computational issues*, Computational perspectives on number theory, in honor of A.O.L. Atkin, AMS/IP Studies in Advanced Mathematics, vol. 7, AMS/IP, 1998, pp. 21–76. MR **99a**:11078
- [4] Reynald Lercier and François Morain, *Counting the number of points on elliptic curves over finite fields: strategies and performances*, Advances in Cryptology, EUROCRYPT 95 (L.C. Guillou and J.-J. Quisquater, eds.), Lecture Notes in Computer Science, vol. 921, Springer, 1995, pp. 79–94.
- [5] ———, *Algorithms for computing isogenies between elliptic curves*, Computational perspectives on number theory, in honor of A.O.L. Atkin, AMS/IP Studies in Advanced Mathematics, vol. 7, AMS/IP, 1998, pp. 77–94. MR **96h**:11060
- [6] René Schoof, *Counting points on elliptic curves over finite fields*, Journal de Théorie des Nombres de Bordeaux **7** (1995), no. 1. MR **97i**:11070
- [7] Jean-Pierre Serre, *Groupes algébriques et corps de classes*, Hermann, 1959. MR **21**:1973
- [8] John Tate, *Endomorphisms of abelian varieties over finite fields*, Inventiones Math. **2** (1966), 134–144. MR **34**:5829

GROUPE DE RECHERCHE EN MATHÉMATIQUES ET INFORMATIQUE DU MIRAIL, UNIVERSITÉ DE TOULOUSE II, LE MIRAIL, FRANCE

*E-mail address:* couveign@math.u-bordeaux.fr