Explicit Computation of Isomorphisms Between Finite Fields

Bill Allombert

Université Bordeaux I, Laboratoire A2X, 351 cours de la Libération, 33 405 Talence, France.
E-mail: allomber@math.u-bordeaux.fr

Although it is easy to prove that two finite fields having the same cardinality are isomorphic, the proof uses embeddings into an algebraic closure (or at least into a common overfield), hence is not constructive, and so does not provide explicit isomorphisms. We give algorithms to solve this problem efficiently in practice, and as an application, we also give an algorithm for factoring a polynomial $P \in \mathbb{F}_p[X]$ over a finite extension of $\mathbb{F}_p$.

1. INTRODUCTION

Let p be a prime number, n an integer, T_1 and T_2 two irreducible polynomials of degree n with coefficients in the finite field $\mathbb{F}_p$, $K_1 = \mathbb{F}_p[X]/(T_1)$, and $K_2 = \mathbb{F}_p[X]/(T_2)$. The two finite fields K_1 and K_2 both have p^n elements, hence are isomorphic. However, the classical proof of this fact does not lead to an explicit isomorphism between them, that is to a polynomial S with coefficients in $\mathbb{F}_p$ such that $T_1 \circ S \equiv 0 \pmod{T_2}$.

In this paper, we describe an algorithm which solves this problem faster than factorization over finite fields (see for example [3]).

In practice, it is probably also faster than the theoretical polynomial-time algorithm given by H. W. Lenstra in [2] which to our knowledge has not been implemented.

We use Kummer theory and Artin–Shreier theory to compute an almost canonical defining polynomial for K_1 and K_2. From these polynomials, it is then easy to compute explicit isomorphisms. Furthermore, we also obtain from this a fast algorithm to compute the factorization of a univariate polynomial with coefficients in the base field $\mathbb{F}_p$ over a finite extension K of $\mathbb{F}_p$.

2. COMPUTING ISOMORPHISMS WHEN N DIVIDES $P-1$

In this section, we assume that the degree n divides $p-1$, or equivalently that there exists a primitive nth root of unity ζ in $\mathbb{F}_p$, giving us the possibility to use Kummer theory. Although this is a sub-case of the next section, it is much simpler to explain, and it uses simpler objects, and so it leads to a faster implementation.

2.1. Theoretical Aspects

The idea is as follows. The finite field K_1 being a cyclic extension of $\mathbb{F}_p$ containing a primitive nth root of unity ζ, it is a Kummer extension of $\mathbb{F}_p$. By Hilbert's Theorem 90, we can thus explicitly compute an element α_1 of K_1 such that $\sigma(\alpha_1)/\alpha_1 = \zeta$, where σ is a generator of the Galois group of $K_1/\mathbb{F}_p$. It follows that $a_1 = \alpha_1^n$ is in $\mathbb{F}_p$. In addition, since the smallest strictly positive integer k such that σ^k fixes α_1 is equal to n, it follows that $\mathbb{F}_p(\alpha_1)$ is of degree exactly equal to n over $\mathbb{F}_p$, hence is equal to K_1. Similarly, we can compute an element α_2 of K_2 such that $a_2 = \alpha_2^n$ is in $\mathbb{F}_p$ and $K_2 = \mathbb{F}_p(\alpha_2)$. By construction, it is clear that a_1/a_2 is an nth power in $\mathbb{F}_p$. We then compute an element c in $\mathbb{F}_p$ such that $c^n = a_1/a_2$, and so we can easily find explicitly the unique isomorphism sending α_1 to $c\alpha_2$.

In practice, α_1 and α_2 will be computed using linear algebra, and not by directly using the proof of Hilbert's Theorem 90.

2.2. Algorithmic Aspects

ALGORITHM 2.1. Under the above hypotheses and notation, denote by B_1 the power basis $(1, \overline{X}, \dots, \overline{X}^{n-1})$ of K_1 and by B_2 the power basis $(1, \overline{X}, \dots, \overline{X}^{n-1})$ of K_2. We compute an isomorphism S as follows.

1. [Compute ζ] Compute a primitive nth root of unity ζ in $\mathbb{F}_p$.
2. [Compute Frobenius maps] Compute the matrix A_1 of the Frobenius automorphism of K_1 on the power basis B_1, and the matrix A_2 of the Frobenius automorphism of K_2 on the power basis B_2.
3. [Solve Hilbert 90] Compute an eigenvector V_1 of A_1 and an eigenvector V_2 of A_2, both for the eigenvalue ζ.
4. [Compute α_1 and α_2] Compute the element α_1 of K_1 whose representation on B_1 is V_1 and compute the element α_2 of K_2 whose representation on B_2 is V_2.
5. [Compute powers] Compute $a_1 = \alpha_1^n$ in K_1 and $a_2 = \alpha_2^n$ in K_2 and coerce them to elements of $\mathbb{F}_p$.
6. [Compute nth root] Compute an nth root $c \in \mathbb{F}_p$ of a_1/a_2.
7. [Compute base change] Compute the matrix M expressing the basis $(1, \alpha_1, \dots, \alpha_1^{n-1})$ of K_1 on the basis B_1.

8. [Express $\overline{X}$] Solve in V the linear system $MV = (0, 1, 0, \ldots, 0)$ and let P_1 be the polynomial whose representation in base $(1, X, \ldots, X^{n-1})$ is V and let P_2 be the polynomial whose representation is V_2.
9. [Substitute] Compute $S = P_1 \circ (cP_2) \pmod{T_2}$ and output S.

Proof. The validity of the algorithm follows immediately from Hilbert's Theorem 90. However in this particular setting, we have the following direct proof.

We know that the Frobenius automorphism is of order n, so that $A_1^n = I_n$, the $n \times n$ identity matrix. By a well-known theorem of linear algebra, it follows that we have

$$\mathbb{F}_p^n = \bigoplus_{i=0}^{n-1} \ker(A_1 - \zeta^i I_n) \ .$$

The subspace $\ker(A_1 - \zeta^i I_n)$ corresponds to the roots of $X^p - \zeta^i X$ in K_1 which is a polynomial of degree p, so it has at most p elements hence its dimension is at most equal to 1.

The vector space $\mathbb{F}_p^n$ is of dimension n and equal to a direct sum of n subspaces of dimension at most 1, hence all these subspaces must be of dimension equal to 1. Thus, A_1 (and similarly A_2) has indeed an eigenvector with associated eigenvalue ζ. As mentioned above, since α_1/α_2 is fixed by the Galois group of $\mathbb{F}_{p^n}/\mathbb{F}_p$ it follows that $c = a_1/a_2$ is an nth power in $\mathbb{F}_p$. ∎

There are several efficient ways to do the computations in Step 1 and Step 6. Note that Kummer theory asserts that, K_1 being a cyclic extension of degree exactly n, for every divisor d of n, a_1 is not a dth power. This can be used to solve Step 6 by Shanks's algorithm for nth roots in *deterministic* polynomial time, since the knowledge of the element a_1 makes the nondeterministic part of such an algorithm disappear.

3. COMPUTING ISOMORPHISMS WHEN *P* DOES NOT DIVIDE *N*

In this section, we generalize the preceding method to the case where p does not divide n.

3.1. Theoretical Aspects

Let $C = \mathbb{F}_p[\zeta]$ be the extension of $\mathbb{F}_p$ by a primitive nth root of unity in $\overline{\mathbb{F}}_p$. The classical use of Kummer theory relies on the tower of extensions shown below and splits the problem into three parts.

$$\begin{tikzcd}[column sep=small] & K_1C \arrow[dl, no head] \arrow[dr, no head] & \\ C \arrow[dr, no head] & & K_1 \arrow[dl, no head] \\ & K_1 \cap C \arrow[d, no head] & \\ & \mathbb{F}_p & \end{tikzcd}$$

1. Do the work in the Kummer extension K_1C/C.
2. Use Galois theory to transfer the result to the extension $K_1/K_1 \cap C$.
3. Get back to the initial extension $K_1/\mathbb{F}_p$.

Although it should be possible to design an algorithm using these three steps, we will not follow this strategy. Instead, we will extend Kummer theory to the $\mathbb{F}_p$-algebra $K_1 \otimes_{\mathbb{F}_p} C$ which is not necessarily a field. We present these results in a general setting in the next subsection.

3.2. Extension of Kummer Theory

Let k be a field, let K be a cyclic extension of k of degree n prime to the characteristic of k, and let C be a finite extension of k containing a primitive nth root of unity ζ. We denote by σ be a generator of the Galois group of the extension K/k.

Proposition 3.1. *The C-algebra $K \otimes_k C$ is an étale algebra, in other words, it has no nonzero nilpotent elements.*

Proof. Let T be an irreducible polynomial in $k[X]$ with splitting field K. The degree of T is not a multiple of the characteristic of k so T is separable. The field K is isomorphic to $k[X]/(T)$, hence $K \otimes_k C$ is isomorphic to $C[X]/(T)$ which has no nonzero nilpotent elements. ▮

We need to introduce the map

$$\tilde{\sigma} \left| \begin{array}{ccc} K \otimes_k C & \to & K \otimes_k C \\ x \otimes y & \mapsto & \sigma(x) \otimes y \end{array} \right.$$

which is clearly an automorphism of the k-algebra $K \otimes_k C$. It will play the role of the generator of the Galois group in the classical theory.

Proposition 3.2. *The set of elements of $K \otimes_k C$ fixed by $\tilde{\sigma}$ is a field isomorphic to C.*

Proof. Let $B = \{b_1, \ldots, b_c\}$ be a basis of C over k. An element β of $K \otimes_k C$ can be written uniquely in the form $\beta = \sum_{i=1}^{c} k_i \otimes b_i$, so $\tilde{\sigma}(\beta) =$

$\sum_{i=1}^{c} \sigma(k_i)\otimes b_i$. If $\tilde{\sigma}(\beta) = \beta$ then for all $1 \leqslant i \leqslant c$ we have $\sigma(k_i) = k_i$ so $k_i \in k$. We conclude that the set of elements of $K\otimes_k C$ fixed by $\tilde{\sigma}$ is $k\otimes_k C$ which is a field isomorphic to C. ▮

We will also need the following result.

PROPOSITION 3.3. *The characteristic polynomial of σ considered as an endomorphism of the k-vector space K is equal to $X^n - 1$.*

Proof. Since $\sigma^n - \mathrm{id} = 0$, the minimal polynomial M of σ divides $X^n - 1$. Dirichlet's character independence theorem asserts that the automorphisms $\mathrm{id}, \sigma, \sigma^2, \ldots, \sigma^{n-1}$ are linearly independent over k, so the degree of M must be at least n, hence $M = X^n - 1$. By using the Cayley-Hamilton theorem, we conclude that the characteristic polynomial of σ is equal to $X^n - 1$. ▮

We have the following generalization of a special case of Hilbert's Theorem 90.

PROPOSITION 3.4. *Under the above hypotheses,* $\ker(\tilde{\sigma} - \zeta id)$ *is a one-dimensional vector space over C.*

Proof. Let B be a basis of K over k. The matrix of the C-linear endomorphism $\tilde{\sigma}$ over the basis $B \otimes 1$ is the same as the matrix of the k-linear endomorphism σ over the basis B, so they both have the same characteristic polynomial $X^n - 1$, which is totally split over C. It follows that each eigenspace of $\tilde{\sigma}$ is one-dimensional. ▮

PROPOSITION 3.5. *Let α and β be two eigenvectors of $\tilde{\sigma}$ for the eigenvalue $1 \otimes \zeta$. Then:*

1. *both α^n and β^n are nonzero elements of $k \otimes_k C$;*
2. *the element α^n/β^n is an nth power in $k \otimes_k C$;*
3. *if $\alpha^n = \beta^n$ then there exists an integer j such that $\alpha = \tilde{\sigma}^j(\beta)$.*

Proof. As in classical Kummer theory, we have $\tilde{\sigma}(\alpha^n) = ((1 \otimes \zeta)\alpha)^n = \alpha^n$, so α^n belongs to the fixed field $k \otimes_k C$ of $\tilde{\sigma}$. The algebra $K \otimes_k C$ does not contain nonzero nilpotent elements, so $\alpha^n \neq 0$, and the same holds for β^n.

The eigenspace of $\tilde{\sigma}$ for the eigenvalue $1 \otimes \zeta$ is one-dimensional, so there exists an element c of $k \otimes C$ such that $\alpha = c\beta$, hence $\alpha^n/\beta^n = c^n$. Furthermore, if $c^n = 1$ then c is an nth root of unity, hence there exists an integer j such that $c = 1 \otimes \zeta^j$, and so $\alpha = (1 \otimes \zeta^j)\beta = \tilde{\sigma}^j(\beta)$. ▮

Note that, if the tensor product is a field, these results are well-known. In the sequel, we will assume that ζ generates C over k, so that $C = k(\zeta)$, and we will set $r = [C : k]$.

PROPOSITION 3.6. *Denote by α an eigenvector of $\tilde{\sigma}$ for the eigenvalue $1 \otimes \zeta$. If a_0 is the first component of α on the basis $(1 \otimes \zeta^i)_{i=0}^{r-1}$ of $K \otimes_k C$ over C, then a_0 is a primitive element for the extension K/k.*

Proof. Note first that $k(a_0)/k$ is a subextension of a cyclic extension, hence is Galois, so every conjugate of a_0 lies in $k(a_0)$.

Let us write $\alpha = \sum_{i=0}^{r-1}(a_i \otimes \zeta^i)$. Then $\tilde{\sigma}(\alpha) = \sum_{i=0}^{r-1}(\sigma(a_i) \otimes \zeta^i)$. If M is the companion matrix of the minimal polynomial $T = X^r - \sum_{i=0}^{r-1} b_i X^i$ of ζ over K, then $(\sigma(a_i)) = M(a_i)$ and

$$\begin{pmatrix} \sigma(a_0) \\ \sigma(a_1) \\ \sigma(a_2) \\ \vdots \\ \sigma(a_{r-1}) \end{pmatrix} = \begin{pmatrix} 0 & & & & b_0 \\ 1 & & \text{\Large 0} & & b_1 \\ & 1 & & & b_2 \\ & \text{\Large 0} & & \ddots & \vdots \\ & & & 1 & b_{r-1} \end{pmatrix} \begin{pmatrix} a_0 \\ a_1 \\ a_2 \\ \vdots \\ a_{r-1} \end{pmatrix}$$

so that

$$\begin{aligned} \sigma(a_0) &= b_0 a_{r-1} && (1) \\ \sigma(a_i) &= a_{i-1} + b_i a_{r-1} \quad , \ 1 \leqslant i \leqslant r-1 && (2) \end{aligned}$$

Since b_0 is nonzero, we can rewrite Equation (1) as $a_{r-1} = \sigma(a_0)/b_0$, hence $a_{r-1} \in k(a_0)$. Rewriting Equation (2) as $a_{i-1} = \sigma(a_i) - b_i a_{r-1}$ for $i = r-1$ down to 1, we see that all the a_i belong to $k(a_0)$. If e is an integer such that $\sigma^e(x) = x$ for all elements $x \in k(a_0)$, then $\tilde{\sigma}^e(\alpha) = \alpha$ and so $(1 \otimes \zeta^e)\alpha = \alpha$, hence $\zeta^e = 1$ so e is a multiple of n. By Galois theory, we thus have $K = k(a_0)$. ∎

Note that Proposition 3.5 asserts that the first coordinates of two eigenvectors are conjugate.

COROLLARY 3.1. *The matrix $T(M)$ is of rank exactly $n - r$. Moreover let $a_0 \neq 0$ be in the kernel of $T(M)$, and $(a_i)_{i=1}^{r-1}$ such that*

$$\begin{aligned} a_{r-1} &= \sigma(a_0)/b_0 && (3) \\ a_{i-1} &= \sigma(a_i) - b_i a_{r-1} \quad \text{for } i = r-1 \text{ down to } 1 && (4) \end{aligned}$$

Then the element $\alpha = \sum_{i=0}^{r-1}(a_i \otimes \zeta^i)$ is an eigenvector of M for the eigenvalue ζ.

The significance of this corollary is that we can compute α with linear algebra over k instead of K.

3.3. Algorithmic Aspects

ALGORITHM 3.1. Under the above hypotheses and notation, denote by B_1 the power basis $(1, \overline{X}, \ldots, \overline{X}^{n-1})$ of K_1 and by B_2 the power basis $(1, \overline{X}, \ldots, \overline{X}^{n-1})$ of K_2. We compute an isomorphism S as follows.

1. [Compute ζ] Find an irreducible factor F of the nth cyclotomic polynomial as a polynomial over $\mathbb{F}_p$ and set $C = \mathbb{F}_p[X]/(F)$ and $\zeta = \overline{X}$.
2. [Compute Frobenius maps] Compute the matrix A_1 of the Frobenius automorphism of K_1 on the power basis B_1, and the matrix A_2 of the Frobenius automorphism of K_2 on the power basis B_2.
3. [Find kernel] Compute a nonnil vector $a_{0,1} \in \ker T(A_1)$ and a nonnil vector $a_{0,2} \in \ker T(A_2)$
4. [Solve Hilbert 90] Apply Corollary 3.1 to obtain an eigenvector V_1 of A_1 and an eigenvector V_2 of A_2, both for the eigenvalue ζ over the field C.
5. [Compute α_1 and α_2] Compute the element α_1 of $K_1 \otimes C$ whose representation on $B_1 \otimes 1$ is V_1 and compute the element α_2 of $K_2 \otimes C$ whose representation on $B_2 \otimes 1$ is V_2.
6. [Compute powers] Compute $a_1 = \alpha_1^n$ in $K_1 \otimes C$ and $a_2 = \alpha_2^n$ in $K_2 \otimes C$ and coerce them to elements of C.
7. [Compute nth root] Compute an nth root c of a_1/a_2 in the field C.
8. [Compute β_1 and β_2] Compute β_1 as the first coordinate of α_1 in the basis $\{1, \zeta, \ldots, \zeta^{\deg F - 1}\}$ and β_2 as the first coordinate of $c\alpha_2$ in the same basis.
9. [Compute base change] Compute the matrix M expressing the basis $(1, \beta_1, \ldots, \beta_1^{n-1})$ of K_1 on the basis B_1.
10. [Express $\overline{X}$] Solve in V the linear system $MV = (0, 1, 0, \ldots, 0)$ and let P_1 be the polynomial whose representation in base $(1, X, \ldots, X^{n-1})$ is V and let P_2 be the polynomial such that $P_2(\overline{X}) = \beta_2$ in K_2.
11. [Substitute] Compute $S = P_1 \circ P_2 \pmod{T_2}$ and output S.

Proof. This simply follows from the results of Subsection 3.2 applied to the extensions $K_1/\mathbb{F}_p$ and $K_2/\mathbb{F}_p$. ∎

The two nonelementary steps of the algorithm are the following.

1. The factorization over $\mathbb{F}_p$ of cyclotomic polynomials. This is solved efficiently by [5, Theorem 9].
2. The computation of an nth root in a finite field. This is solved by applying the following modification of Shanks's algorithm for nth roots in a finite prime field, which we give below for completeness.

Note that the other steps are deterministic. In practice, most of the time is spent in Step 4.

The computation of an nth root in a finite field is easily reducible to the extraction of an ℓth root, where ℓ is a prime number dividing $q-1$.

ALGORITHM 3.2. Let K be a finite field with q elements and let ℓ be a prime number dividing $q-1$. Let r and e be such that $q-1=\ell^e r$ and r prime to ℓ. We compute an ℓth root x of the element $a \in K$, if it exists, as follows.

1. [Find nonresidue] Choose elements $y \in K$ at random until $y^{(q-1)/\ell} \neq 1$ and set $z := y^{(q-1)/\ell}$.
2. [Extended Euclid Algorithm] Compute u_1 and u_2 such that $ru_1 + \ell \times u_2 = 1$.
3. [Initialize] Set $x := a^{u_2}$, $b := a^{-ru_1}$.
4. [Test] If $b = 1$ then output x.
5. [Get exponent] Find the smallest integer m such that $b^{\ell^m} = 1$. If $m = e$ output "Element a is not an ℓth power in K".
6. [Discrete logarithm] Compute the smallest integer n such that $z^n = b^{-\ell^{m-1}}$.
7. [Reduce exponent] Set $w := y^{n\ell^{e-m-1}}$, $z := z^n$, $e := k$, $x := wx$, $y := w^\ell$ and $b := yb$.
8. [Loop] Go to Step 4.

4. COMPUTING ISOMORPHISMS WHEN $N = P^K$

4.1. Theoretical Aspects

The main idea of this section is to use Artin–Shreier theory to construct almost canonical irreducible polynomials of degree p^k.

We will use the following lemma (see [5, Lemma 2.3]).

LEMMA 4.1. *The polynomial $X^p - X - 1$ is irreducible in $\mathbb{F}_p[X]$. Furthermore, let K be an extension of $\mathbb{F}_p$, let $a \in K$, and assume that the polynomial $X^p - X - a$ is irreducible in $K[X]$. Then, if $E = K(\alpha)$ for a root α of $X^p - X - a$, the polynomial $X^p - X - a\alpha^{p-1}$ is irreducible in $E[X]$.*

Note that the coefficients of the polynomial $X^p - X - a\alpha^{p-1}$ are themselves *polynomials in α* so they commute with finite field isomorphisms. Thus this lemma allows us to compute two conjugate towers of subfields of K_1 and K_2 as follows:

PROPOSITION 4.1.

Let $K_{1,0} = K_{2,0} = \mathbb{F}_p$, $a_{1,0} = a_{2,0} = 1$, $\alpha_{1,0} = \alpha_{2,0} = 1$. We define inductively the subfields $K_{i,j}$ of K_i, and the elements $a_{i,j}$ and $\alpha_{i,j}$ of $K_{i,j}$ for $i \in \{1,2\}$ and $1 \leqslant j \leqslant k$ by

- $a_{i,j} = a_{i,j-1}\alpha_{i,j-1}$
- $\alpha_{i,j}$ *is a root of the polynomial* $X^p - X - a_{i,j}$ *in the field* K_i
- $K_{i,j}$ *is the subfield of* K_i *generated by* $\alpha_{i,j}$.

For each $1 \leqslant j \leqslant k$ there exists an isomorphism between $K_{1,j}$ and $K_{2,j}$ sending $\alpha_{1,j}$ to $\alpha_{2,j}$. In particular, for $j = k$, we can compute explicitely an isomorphism between K_1 and K_2.

4.2. Algorithmic Aspects

ALGORITHM 4.1. Under the above hypotheses and notation, denote by B_1 the power basis $(1, \overline{X}, \ldots, \overline{X}^{n-1})$ of K_1 and by B_2 the power basis $(1, \overline{X}, \ldots, \overline{X}^{n-1})$ of K_2. We compute an isomorphism S as follows.

1. [Compute Frobenius maps] Compute the matrix A_1 of the Frobenius automorphism of K_1 on the power basis B_1, and the matrix A_2 of the Frobenius automorphism of K_2 on the power basis B_2.
2. [Initialize loop] Set $e := 1$ $a_1 := 1$, $a_2 := 1$, $\alpha_1 := 1$ and $\alpha_2 := 1$.
3. [Apply lemma 4.1] Set $a_1 := a_1\alpha_1^{p-1}$ and $a_2 := a_2\alpha_2^{p-1}$.
4. [Compute coordinates] Compute the vector W_1 expressing a_1 on the power basis B_1 and compute the vector W_2 expressing a_2 on the power basis B_2.
5. [Solve additive Hilbert 90] Solve the linear systems $A_1V_1 = V_1 + W_1$ and $A_2V_2 = V_2 + W_2$.
6. [Compute α_1 and α_2] Compute the element α_1 of K_1 whose representation on B_1 is V_1 and compute the element α_2 of K_2 whose representation on B_2 is V_2.
7. [Loop] If $e < k$, set $e := e + 1$ and go to Step 3.
8. [Compute base change] Compute the matrix M expressing the basis $(1, \alpha_1, \ldots, \alpha_1^{n-1})$ of K_1 on the basis B_1.
9. [Express $\overline{X}$] Solve in V the linear system $MV = (0, 1, \ldots, 0)$ and let P_1 be the polynomial whose representation in base $(1, X, \ldots, X^{n-1})$ is V and let P_2 be the polynomial whose representation is V_2.
10. [Substitute] Compute $S = P_1 \circ P_2 \pmod{T_2}$ and output S.

Note that this algorithm is deterministic and that all computations are done with elements of $\mathbb{F}_p$.

5. COMPUTING ISOMORPHISMS IN THE GENERAL CASE AND FACTORIZATION OVER AN EXTENSION

We present these two topics in the same section because they both make use of the same idea: the computation of isomorphisms between subfields.

Note that Algorithms 3.1 and 4.1 can be applied to polynomials of degree multiple of n. In this case they output an isomorphism between the unique subfields of order n of K_1 and K_2. But if n is small with respect to the degree of the fields, it will generally be faster to compute first the subfields of order n before computing the isomorphisms.

5.1. Computing Isomorphisms in the General Case

Let us write $n = qp^k$ where q and p are coprime. Algorithms 3.1 and 4.1 allow us to compute $\alpha_1, \alpha'_1 \in K_1$ and $\alpha_2, \alpha'_2 \in K_2$ such that

- The fields $\mathbb{F}_p(\alpha_1)$ and $\mathbb{F}_p(\alpha_2)$ are isomorphic of order q and there exists an isomorphism sending α_1 to α_2.
- The fields $\mathbb{F}_p(\alpha'_1)$ and $\mathbb{F}_p(\alpha'_2)$ are isomorphic of order p^k and there exists an isomorphism sending α'_1 to α'_2.

Then there exists an isomorphism between K_1 and K_2 sending $\alpha_1 + \alpha'_1$ to $\alpha_2 + \alpha'_2$. The proof comes directly from the fact that $\mathbb{F}_p(\alpha_1)/\mathbb{F}_p$ and $\mathbb{F}_p(\alpha'_1)/\mathbb{F}_p$ are linearly disjoint Galois extensions. Note that this is essentially the content of [6, Lemma 2.4].

5.2. Factorization Over an Extension

Let P be a polynomial with coefficients in $\mathbb{F}_p$ and let T_2 be an irreducible polynomial defining an extension K_2 of $\mathbb{F}_p$ of degree n. We want to compute the factorization of P over K_2. The algorithm is based on the following easy lemma from Galois theory:

Lemma 5.1 (Galois factorization). *Let T be an irreducible polynomial over a field F and let α be a root of T in an algebraic closure of F such that the splitting field $L = F(\alpha)$ is Galois over F. Then the irreducible factors of T over a subextension K/F of L/F are*

$$\prod_{\sigma \in C} (X - \sigma(\alpha)) \quad \textit{for each coset } C \in \mathcal{G}a\ell(L/F)/\mathcal{G}a\ell(L/K)$$

Algorithm 5.1. Under the above hypotheses and notation, we compute the factorization of P over K_2 as follows. Factor P over $\mathbb{F}_p$ and apply the following steps to factor all the irreducible factors T_1 of P, and output the factorization.

1. [Compute intersection] Compute the greatest common divisor d of n and the degree of T_1.

2. [Compute isomorphism] Compute an isomorphism S between the unique subfields of order d of $K_1 = \mathbb{F}_p[X]/(T)$ and K_2, given by the image $\alpha_2 \in K_2$ of an element α_1 of K_1.
3. [Compute factorization] Compute $Q_1(Y) = \prod_{0 \leqslant k < n/d}(Y - \overline{X}^{p^{dk}})$ in K_1, where $\overline{X}$ denotes the class of X in $\mathbb{F}_p[X]/(T)$.
4. [Apply isomorphism] Coerce all the coefficients of Q_1 to elements of K_1 and apply the isomorphism S to each of them, thus obtaining a factor $Q_2 \in K_2[X]$.
5. [Compute conjugate factors] Compute the conjugate factors of Q_2 by applying successive powers of the Frobenius automorphism of the field K_2 to all the coefficients of Q_2.

Note that, apart from the use of Algorithm 3.1 and the initial factorization over $\mathbb{F}_p$, all the steps are deterministic. Any algorithm which computes isomorphisms can be used in Step 2.

6. TIMINGS

The above algorithm has been implemented using the Pari library (see [4]), and the computations have been made on a 248Mhz UltraSparc-II.

degree	p	order	val.	algo.	fact.
10	131	1	0	0.01 s	0.230 s
16	1009	1	0	0.08 s	3.31 s
20	1009	2	0	0.32	14,32
30	67108879	2	0	1.260 s	238.1 s
20	23	4	0	0.39	2.7 s
30	10007	4	0	1.910 s	79.2 s
17	1009	16	0	2.26 s	2.16 s
29	300007	29	0	40.07 s	120.35 s
14	7	1	1	0.02 s	0.28 s
30	5	2	1	0.29 s	9.84 s
35	7	4	1	1.06 s	14.06 s
18	3	1	2	0.02 s	0.71 s
25	5	1	2	0.06 s	4.07 s
20	2	4	2	0.11 s	0.45 s
27	3	1	3	0.06 s	4.26 s
40	2	4	3	0.58 s	8.17 s
16	2	1	4	0.02 s	0.28 s
32	2	1	5	0.07 s	3.01 s
64	2	1	6	0.39 s	37.42 s
128	2	1	7	4.22 s	422.12 s
48	2	2	4	0.46 s	15.48 s

Column *degree* denotes the degree n of the polynomials and column p denotes the prime number such that the polynomials have theirs coefficients in $\mathbb{F}_p$. Column *order* gives the degree of the extension $\mathbb{F}_p[\zeta_n]$ over $\mathbb{F}_p$ and column *val.* gives the p–adic valuation of n. Column *algo.* corresponds to the timing of our implementation of the algorithm. This column is to compare with column *fact.* who corresponds to the timing of the root-finding function for polynomials over finite fields of the NTL library of Victor Shoup (see [7]).

REFERENCES

1. Henri COHEN, *A Course in Computational Algebraic Number Theory*, Graduate Texts in Mathematics. **138**, Springer, 1993, corr. 3^{rd} printing 1996
2. H.W. LENSTRA, *Finding isomorphism between finite fields.*, Math. Comp. **56** (1991) 329–347.
3. Erich KALTOFEN and Victor SHOUP, *Subquadratic-time factorization of polynomials over finite fields*, Math. Comp. **67** (1998) 1179–1197.
4. PARI, C. BATUT, K. BELABAS, D. BERNARDI, H. COHEN and M. OLIVIER, *User's Guide to PARI-GP*, version 2.0.19
5. Victor SHOUP, *Fast construction of irreducible polynomials over finite fields*, J.Symbolic Comp. **17** (1994) 371–391.
6. Victor SHOUP, *New algorithms for finding irreducible polynomials over finite fields*, Math. Comp. **54** (1990) 435–447.
7. Victor SHOUP, *NTL: A Library for doing Number Theory (version 4.0a)* , http://www.shoup.net/ntl/.