Standard Lattices of Compatibly Embedded Finite Fields

Luca de Feo, Hugues Randriambololona, Édouard Rousseau

Standard Lattices of Compatibly Embedded Finite Fields

Luca De Feo
luca.de-feo@uvsq.fr
Université Paris Saclay – UVSQ, LMV

Hugues Randriam
randriam@enst.fr
LTCI, Télécom ParisTech

Édouard Rousseau
erousseau@enst.fr
LTCI, Télécom ParisTech
Université Paris Saclay – UVSQ, LMV

ABSTRACT

Lattices of compatibly embedded finite fields are useful in computer algebra systems for managing many extensions of a finite field $\mathbb{F}_p$ at once. They can also be used to represent the algebraic closure $\bar{\mathbb{F}}_p$, and to represent all finite fields in a standard manner.

The most well known constructions are Conway polynomials, and the Bosma–Cannon–Steel framework used in Magma. In this work, leveraging the theory of the Lenstra-Allombert isomorphism algorithm, we generalize both at the same time.

Compared to Conway polynomials, our construction defines a much larger set of field extensions from a small pre-computed table; however it is provably as inefficient as Conway polynomials if one wants to represent *all* field extensions, and thus yields no asymptotic improvement for representing $\bar{\mathbb{F}}_p$.

Compared to Bosma–Cannon–Steel lattices, it is considerably more efficient both in computation time and storage: all algorithms have at worst quadratic complexity, and storage is linear in the number of represented field extensions and their degrees.

Our implementation written in C/Flint/Julia/Nemo shows that our construction in indeed practical.

CCS CONCEPTS

• **Mathematics of computing** → *Mathematical software*; • **Computing methodologies** → **Algebraic algorithms**;

KEYWORDS

Finite fields; field extensions; Conway polynomials.

ACM Reference Format:
Luca De Feo, Hugues Randriam, and Édouard Rousseau. 2019. Standard Lattices of Compatibly Embedded Finite Fields. In *International Symposium on Symbolic and Algebraic Computation (ISSAC '19), July 15–18, 2019, Beijing, China.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3326229.3326251

1 INTRODUCTION

Computer algebra systems (CAS) are often faced with the problem of constructing several extensions of a finite field $\mathbb{F}_p$ in a *compatible* way, i.e., such that the (subfield) inclusion lattice of the given extensions can be computed and evaluated efficiently.

ISSAC '19, July 15–18, 2019, Beijing, China

ACM ISBN 978-1-4503-6084-5/19/07…$15.00
https://doi.org/10.1145/3326229.3326251

Concretely, what is sought is a data structure Λ to represent arbitrary collections of extensions of $\mathbb{F}_p$, in such a way that elements of $\mathbb{F}_{p^m}$ are represented in optimal space (i.e., $O(m)$ coefficients), and that arithmetic operations are performed efficiently (i.e., $O\left(\mathrm{lcm}(l,m)^d\right)$ arithmetic operations to combine an element of $\mathbb{F}_{p^l}$ and an element of $\mathbb{F}_{p^m}$, where $d \leq 3$ and, possibly, $d = 1+\varepsilon$). To this end, it is useful to set several sub-goals:

- ***Effective embeddings:*** For any pair of extensions $k \subseteq K$ in Λ, there exists an efficiently computable embedding $\phi : k \to K$, and algorithms to evaluate ϕ on k, and the section ϕ^{-1} on K.
- ***Compatibility:*** The embeddings are *compatible*, i.e., for any triple $k \subseteq K \subseteq L$ in Λ, and embeddings $\phi : k \to K$, $\psi : K \to L$, $\chi : k \to L$, one has $\chi = \psi \circ \phi$.
- ***Incrementality:*** The data associated with an extension (e.g., its irreducible polynomial, change-of-basis matrices, …) must be computable efficiently and *incrementally*, i.e., adding a new field extension to Λ does not require recomputing data for all extensions already in Λ.
- ***Uniqueness:*** Any extension of $\mathbb{F}_p$ is determined by an irreducible polynomial whose definition only depends on the characteristic p and the degree of the extension.
- ***Generality:*** Extensions of $\mathbb{F}_p$ can be represented by arbitrary irreducible polynomials.

Some goals, such as incrementality, uniqueness and generality are optional, and it is obvious that uniqueness and generality are even in conflict with each other. An incremental data structure can be used to effectively represent an algebraic closure $\bar{\mathbb{F}}_p$, with new finite extensions built on the fly as they are needed. Uniqueness is useful for defining field elements in a standard way, portable between different CAS, while generality is useful in a context where the user is left with the freedom of choosing the defining polynomials. Note that any solution can be made unique by replacing all random choices with pseudo-random ones, however one is usually interested in unique solutions that have a simple mathematical description. Also, any solution can be made general by means of an isomorphism algorithm [1, 8, 19, 21]. Other optional goals, such as computing normal bases or evaluating Frobenius morphisms, may be added to the list, however they are out of the scope of this work.

Previous work. The first and most well known solution is the family of Conway polynomials [17, 22], first adopted in GAP [26], and then also by Magma [2] and Sage [27]. Conway polynomials yield uniqueness, however computing them requires exponential time using the best known algorithm, thus incrementality is only available at a prohibitive cost; for this reason, they are usually pre-computed and tabulated up to some bound.

Lenstra [19] was the first to show the existence of a (incremental, general) data structure computable in deterministic polynomial

time. He proved that, besides the problem of finding irreducible polynomials, any other question is amenable to linear algebra. Subsequent work of Lenstra and de Smit [20] tackled the uniqueness problem, albeit only from a theoretical point of view.

In practice, randomized algorithms are good enough for a CAS, then polynomial factorization and basic linear algebra provide an easy (incremental, general) solution, that was first analyzed by Bosma, Cannon and Steel [3], and is currently used in Magma.

All solutions presented so far have superquadratic complexity, i.e., $d > 2$. Recent work on embedding algorithms [11, 12, 14] yields subquadratic (more precisely, $d \leq 1.5$) solutions for specially constructed (non-unique, non-general) families of irreducible polynomials, and even quasi-optimal ones (i.e., $d = 1+\varepsilon$) if a quasi-linear modular composition algorithm is available. However these constructions involve counting points of random elliptic curves over finite fields, and have thus a rather high polynomial dependency in $\log p$; for this reason, they are usually considered practical only for relatively small characteristic.

Our contribution. In this work we present an incremental, general and/or unique solution for lattices of compatibly embedded finite fields, where all embeddings can be computed and evaluated in quasi-quadratic time. Our starting point is Allombert's [1] and subsequent [8] improvements to Lenstra's isomorphism algorithm [19]. Plugging them in the Bosma–Cannon–Steel framework immediately produces an incremental general solution with quasi-quadratic complexity; however we go much further. Indeed, we show that the compatibility requirement can be taken a step further by constructing a lattice of $\mathbb{F}_p$-algebras with a distinguished element, which is a byproduct of the Lenstra-Allombert algorithm.

The advantages of our construction over a naive combination of the Lenstra-Allombert algorithm and the Bosma–Cannon–Steel framework are multiple. Storage drops from quadratic to linear in the number of extensions stored in Λ and in their degrees, and the cost of adding a new extension to Λ drops similarly.

Our $\mathbb{F}_p$-algebras are constructed by tensoring an arbitrary lattice of extensions of $\mathbb{F}_p$ with what we call a *cyclotomic lattice* (see next section). In this work we mostly abstract away from the concrete instantiation of the cyclotomic lattice, only fixing a choice in Section 6, where we use Conway polynomials to analyze the complexity and implement our algorithms. This choice allows us to uniquely represent finite fields of degree exponentially larger than with Conway polynomials alone; however it also has the serious drawback of being generically as hard to compute as Conway polynomials, and thus relatively unpractical. We leave the exploration of other, more practical, instantiations of cyclotomic lattices for future work.

Organization. The presentation is structured as follows. In Section 2 we review some basic algorithms and facts on roots of unity and Conway polynomials. In Section 3 we review the Lenstra-Allombert algorithm and we define and study *Kummer algebras*, the main ingredient to our construction. In Section 4 we introduce a notion of compatibility for solutions of Hilbert 90 in Kummer algebras, that provides standard defining polynomials for finite fields. Then in Section 5 we again use these compatible solutions to construct standard compatible embeddings between finite fields, from which a lattice can be incrementally constructed. Finally, in Section 6 we give the complexities of our algorithms, and present our implementation.

2 PRELIMINARIES

Fundamental algorithms and complexities. Throughout this paper we let $\mathbb{F}_p$ be a finite field. For simplicity, we shall assume that p is prime, which is arguably the most useful case, however our results could easily be extend to non-prime fields. We measure time complexities as a number of arithmetic operations $(+, \times, /)$ over $\mathbb{F}_p$, and storage as a number of elements of $\mathbb{F}_p$. We let $\mathsf{M}(m)$ denote the number of operations required to multiply two polynomials with coefficients in $\mathbb{F}_p$ of degree at most m, and adopt the usual super-linearity assumptions on the function M (see [28, Ch. 8.3]).

Any finite extension $\mathbb{F}_{p^m}$ can be represented as the quotient of $\mathbb{F}_p[X]$ by an irreducible polynomial of degree m. The algorithms we present in the next sections need not assume any particular representation for finite fields, however when analyzing their complexities we will assume this representation. Then, multiplications in $\mathbb{F}_{p^m}$ can be carried out using $O(\mathsf{M}(m))$ operations, and inversions using $O(\mathsf{M}(m)\log(m))$. In this work we will also need to perform computations in algebras $\mathbb{F}_{p^m} \otimes \mathbb{F}_{p^n}$: representing them as quotients of a bivariate polynomial ring, we can multiply elements using $O(\mathsf{M}(mn))$ operations.

We will extensively use a few standard routines, that we recall briefly. Brent and Kung's algorithm [6] computes the **modular composition** $f(g) \bmod h$ of three polynomials $f, g, h \in \mathbb{F}_p[X]$ of degree at most m using $O\big(m^{(\omega+1)/2}\big)$ operations, where ω is the *exponent of linear algebra* over $\mathbb{F}_p$. The Kedlaya–Umans algorithm [18] solves the same problem, and has better complexity in the binary RAM model, however it is widely considered impractical, we shall thus not consider it in our complexity estimates.

By applying *transposition techniques* [5, 9] to Brent and Kung's algorithm, Shoup [24, 25] derived an algorithm to compute **minimal polynomials** of arbitrary elements of $\mathbb{F}_{p^m}$, having the same complexity $O\big(m^{(\omega+1)/2}\big)$. Kedlaya and Umans' improvements also apply to Shoup's minimal polynomial algorithm, with the same practical limitations.

Shoup's techniques can also be applied to **evaluate embeddings of finite fields**. Let $\mathbb{F}_{p^l}, \mathbb{F}_{p^m}$ be a pair of finite fields related by an embedding $\phi : \mathbb{F}_{p^l} \hookrightarrow \mathbb{F}_{p^m}$, and let a generator α_l of $\mathbb{F}_{p^l}$ and its image $\phi(\alpha_l)$ in $\mathbb{F}_{p^m}$ be given. Given these data, for any element $x \in \mathbb{F}_{p^l}$ it is possible to compute its image $\phi(x)$ using $O\big(m^{(\omega+1)/2)}\big)$ operations; similarly, given an element $y = \phi(x)$ in $\mathbb{F}_{p^m}$, it is possible to recover x in the same asymptotic number of operations. The relevant algorithms are summarized in [7, Sec. 6]; note that, for specially constructed generators α_l, more efficient algorithms may exist [11–14].

The present work focuses on algorithms to **compute embeddings of finite fields**, i.e., algorithms that, given finite fields $\mathbb{F}_{p^l}$ and $\mathbb{F}_{p^m}$ with $l \mid m$, find an embedding $\phi : \mathbb{F}_{p^l} \hookrightarrow \mathbb{F}_{p^m}$, a generator α_l of $\mathbb{F}_{p^l}$, and its image $\phi(\alpha_l)$. An extensive review of known algorithms is given in [8]; here we shall only be interested in the Lenstra–Allombert isomorphism algorithm [1, 19], and its adaptation to compatible lattices of finite fields.

Conway polynomials and cyclotomic lattices. The algorithms of the next sections will be dependent on the availability of a *cyclotomic lattice*. By this we mean, formally, a collection

$$\mathcal{S}^I = \{(K_l, \zeta_l)\}_{l \in I}$$

over some support set $I \subseteq \mathbb{N} \setminus p\mathbb{N}$. Where K_l is an explicitly represented finite extension of $\mathbb{F}_p$, and $\zeta_l \in K_l$ a generating element that is also a primitive l-th root of unity, so

$$K_l = \mathbb{F}_p(\zeta_l), \quad (\zeta_l)^l = 1,$$

together with explicit embeddings

$$\begin{array}{rccc} \iota_{l,m} : & K_l & \hookrightarrow & K_m \\ & \zeta_l & \mapsto & (\zeta_m)^{\frac{m}{l}} \end{array}$$

whenever $l \mid m$.

If I is a finite set of indices, there is an easy randomized algorithm to construct a cyclotomic lattice: compute $n = \mathrm{lcm}_{l \in I}(l)$, construct the smallest field $\mathbb{F}_{p^a}$ such that n divides $p^a - 1$, take a random $x^{(p^a-1)/n} \in \mathbb{F}_{p^a}$ and test that it has multiplicative order n; then all roots ζ_l in the lattice are constructed as powers of this element, and we can set $K_l = \mathbb{F}_p(\zeta_l) \subseteq \mathbb{F}_{p^a}$ and let $\iota_{l,m}$ be natural inclusion.

Nevertheless, the most useful cyclotomic lattices are those where I is the whole set $\mathbb{N} \setminus p\mathbb{N}$. It may seem odd to ask for such data, which in the end provides a representation of $\bar{\mathbb{F}}_p$, as a prerequisite for a construction whose goal is precisely to represent $\bar{\mathbb{F}}_p$. However we shall see that a relatively small cyclotomic sub-lattice is enough to construct a much larger lattice of compatibly embedded finite fields; it thus makes sense to assume that a cyclotomic lattice is available, if it can be computed *incrementally*.

Conway polynomials [22] offer a classic example of cyclotomic lattice. The a-th Conway polynomial $C_a \in \mathbb{F}_p[X]$ is defined as the *lexicographically smallest* monic irreducible polynomial of degree a that is also *primitive* (i.e., its roots generate $\mathbb{F}_{p^a}^\times$) and *norm compatible* (i.e,

$$C_a\left(X^{\frac{p^b-1}{p^a-1}}\right) = 0 \mod C_b$$

whenever a divides b). The cyclotomic lattice is defined first by letting ζ_{p^a-1} be the image of X in $K_{p^a-1} = \mathbb{F}_{p^a} = \mathbb{F}_p[X]/C_a$ for any a; this is then extended to all $l \in I$ by setting $K_l = K_{p^a-1}$ and $\zeta_l = (\zeta_{p^a-1})^{\frac{p^a-1}{l}}$ where $\mathbb{F}_{p^a}$ is the smallest extension $\mathbb{F}_p$ containing l-th roots of unity.

The best known algorithm to compute Conway polynomials has exponential complexity [17], hence they are usually precomputed and tabulated up to a certain bound. Most computer algebra systems switch to other ways of representing finite fields when the tables of Conway polynomials are not enough. A notable exception is SageMath [27] (since version 5.13 [23]), that defines *pseudo-Conway polynomials* by relaxing the "lexicographically first" requirement; although easier to compute in practice, their computation still requires an exponential amount of work.

Other ways to construct cyclotomic lattices are possible. One may, for example, factor cyclotomic polynomials over $\mathbb{F}_p$, being careful to maintain compatibility. In the next sections we shall not suppose any cyclotomic lattice construction in particular, and simply assume that we are given a collection $\mathcal{S}^I$ that satisfies the properties given at the beginning of this paragraph.

3 THE LENSTRA-ALLOMBERT ALGORITHM

We now review the theory behind Allombert's adaptation [1] of Lenstra's isomorphism algorithm [19]. This will be our stepping stone towards the definition of some *standard* elements in field extensions of $\mathbb{F}_p$ with an effective compatibility condition.

The main ingredient of the algorithm is an extension of Kummer theory. Because of this, the algorithm is limited to field extensions $\mathbb{F}_{p^l}$ of degree l prime to p. The easier case of extensions of degree p^e is covered in a similar way using Artin-Schreier theory, and the generic case is solved by separately computing isomorphisms for the power-of-p and the prime-to-p parts, and then tensoring the results together. Due to space constraints, we will not give details for the general case here; see [1, 8, 19].

For any finite extension of $\mathbb{F}_p$, we denote by $\sigma : x \mapsto x^p$ the Frobenius automorphism. Let l be an integer not divisible by p. Then σ is an $\mathbb{F}_p$-linear endomorphism of $\mathbb{F}_{p^l}$ with minimal polynomial $T^l - 1$, separable but not necessarily split, i.e., $\mathbb{F}_{p^l}$ is not necessarily a Kummer extension of $\mathbb{F}_p$. We extend scalars and work in the *Kummer algebra of degree* l:

$$A_l = \mathbb{F}_{p^l} \otimes \mathbb{F}_p(\zeta_l),$$

where $\otimes$ is the tensor product over $\mathbb{F}_p$, and ζ_l is a primitive l-th root of unity, taken from the given cyclotomic lattice. We call $\mathbb{F}_p(\zeta_l)$ the *field of scalars* of A_l, and we define the *level* of A_l as

$$\nu(l) = \mathrm{ord}_{(\mathbb{Z}/l\mathbb{Z})^\times}(p) = [\mathbb{F}_p(\zeta_l) : \mathbb{F}_p],$$

that is, the degree of its field of scalars.

Now $\sigma \otimes 1$ is a $1 \otimes \mathbb{F}_p(\zeta_l)$-linear endomorphism of A_l with l distinct eigenvalues, namely the powers of $1 \otimes \zeta_l$. Thus, if $\eta = \zeta_l^i$ is any l-th root of unity in $\mathbb{F}_p(\zeta_l)$, the corresponding eigenspace is defined by the *Hilbert 90 equation for* η:

$$(\sigma \otimes 1)(x) = (1 \otimes \eta)x, \tag{H90}$$

which plays the role of $\sigma(x) = \eta x$ in classical Kummer theory. The solutions of (H90) in A_l form a $1 \otimes \mathbb{F}_p(\zeta_l)$-vector space of dimension 1, and if x is such a solution for η, then x^j is a solution for η^j.

In particular, let α_l be a nonzero solution of (H90) for ζ_l. Then $1, \alpha_l, \cdots, (\alpha_l)^{l-1}$ are eigenvectors for distinct eigenvalues and thus form a basis of A_l over $1 \otimes \mathbb{F}_p(\zeta_l)$. Likewise

$$(\alpha_l)^l = 1 \otimes c_l$$

for some scalar $c_l \in \mathbb{F}_p(\zeta_l)$, that we shall call the *Kummer constant* of α_l. This proves:

Proposition 1. *Any nonzero solution α_l of* (H90) *for ζ_l is a generating element for A_l as an algebra over $1 \otimes \mathbb{F}_p(\zeta_l)$, inducing an isomorphism*

$$A_l \simeq \mathbb{F}_p(\zeta_l)[T]/(T^l - c_l).$$

Since A_l is known to be an étale algebra, α_l being nonzero implies that c_l is nonzero, which in turn implies that α_l is *invertible* in A_l, indeed $(\alpha_l)^{-1} = (1 \otimes c_l^{-1})(\alpha_l)^{l-1}$.

We will make frequent use of the following:

Lemma 2. *Let K, L be two finite extensions of $\mathbb{F}_p$. Then, for any $\beta \in K \otimes L$, we have $(\sigma \otimes \sigma)(\beta) = \beta^p$.*

Proof. If $\beta = u \otimes v$ is an elementary tensor we have $(\sigma \otimes \sigma)(\beta) = u^p \otimes v^p = \beta^p$. This then extends by linearity since we're in characteristic p. □

In this generality we also introduce the following notation: if $\eta \in L$ has degree d over $\mathbb{F}_p$, then any $\beta \in K \otimes \mathbb{F}_p(\eta) \subseteq K \otimes L$ decomposes uniquely as $\beta = \sum_{i=0}^{d-1} y_i \otimes \eta^i$, and we set

$$\lfloor \beta \rfloor_\eta = y_0.$$

In particular, coming back to A_l, if we write

$$\alpha_l = \sum_{i=0}^{a-1} x_i \otimes \zeta_l^i$$

where $a = \nu(l)$, it is shown in [1] that $x_0 = \lfloor \alpha_l \rfloor_{\zeta_l}$ is a *generating element* for the extension $\mathbb{F}_{p^l}/\mathbb{F}_p$. Moreover, [1] (see also [8]) provides the following equations that allow, in the opposite direction, to recover α_l from x_0:

$$\begin{aligned} x_{a-1} &= \sigma(x_0)/b_0 \\ x_i &= \sigma(x_{i+1}) - b_{i+1}x_{a-1} \quad \text{for } i = a-2 \text{ down to } 1 \end{aligned} \tag{1}$$

where $\zeta_l^a = \sum_{i=0}^{a-1} b_i \zeta_l^i$ is the minimal equation for ζ_l.

Proposition 3. *With the notations above, there are precisely l elements $x \in A_l$ that are solutions of* (H90) *for ζ_l and satisfy $x^l = 1 \otimes c_l$, namely, these are the $(1 \otimes \zeta_l)^u \alpha_l = (\sigma^u \otimes 1)(\alpha_l)$ for $0 \le u < l$. The corresponding generating elements for $\mathbb{F}_{p^l}/\mathbb{F}_p$ are the $\lfloor (\sigma^u \otimes 1)(\alpha_l) \rfloor_{\zeta_l} = \sigma^u(x_0)$; they all have the same minimal polynomial, which is a generating polynomial for $\mathbb{F}_{p^l}/\mathbb{F}_p$ depending only on c_l.*

Proof. The solutions of (H90) for ζ_l form a $1 \otimes \mathbb{F}_p(\zeta_l)$-vector space of dimension 1, thus they all are of the form $x = (1 \otimes \xi)\alpha_l$. Adding the condition $x^l = 1 \otimes c_l$ then forces $\xi^l = 1$, from which all assertions follow. □

Now we consider Kummer algebras of various degrees. Since we assumed that the fields of scalars are defined from a cyclotomic lattice $\mathcal{S}^I$, they are compatibly embedded: for $l \mid m$ prime to p, we have the embedding

$$\begin{array}{rccc} \iota_{l,m}: & \mathbb{F}_p(\zeta_l) & \hookrightarrow & \mathbb{F}_p(\zeta_m) \\ & \zeta_l & \mapsto & (\zeta_m)^{\frac{m}{l}}. \end{array}$$

It is easily shown that, as an $\mathbb{F}_p$-algebra, A_l is isomorphic to a product of copies of $\mathbb{F}_{p^l}(\zeta_l)$, and A_m to a product of copies of $\mathbb{F}_{p^m}(\zeta_m)$. This allows us to describe all $\mathbb{F}_p$-algebra morphisms from A_l to A_m. However here we will focus only on a certain subclass of them:

Definition 4. *A* Kummer embedding *of A_l into A_m is an* injective $\mathbb{F}_p$*-algebra morphism $\Phi : A_l \hookrightarrow A_m$ such that:*

- *Φ extends the scalar embedding $1 \otimes \iota_{l,m}$*
- *Φ commutes with $\sigma \otimes 1$.*

Proposition 5. *Let $\alpha_l \in A_l$ be a nonzero solution of* (H90) *for ζ_l, with Kummer constant c_l. Then, there is a* 1*-to-*1 *correspondence between Kummer embeddings $\Phi : A_l \hookrightarrow A_m$ and solutions $\hat{\alpha} \in A_m$ of* (H90) *for $(\zeta_m)^{\frac{m}{l}}$ that satisfy $(\hat{\alpha})^l = 1 \otimes \iota_{l,m}(c_l)$, given by*

$$\Phi \quad \longleftrightarrow \quad \hat{\alpha} = \Phi(\alpha_l).$$

Proof. Direct consequence of Proposition 1 and Definition 4. □

Actually, Kummer embeddings are easily characterized:

Proposition 6. *There is a natural* 1*-to-*1 *correspondence between Kummer embeddings $\Phi : A_l \hookrightarrow A_m$ and embeddings of finite fields $\phi : \mathbb{F}_{p^l} \hookrightarrow \mathbb{F}_{p^m}$, given by:*

- *If Φ is a Kummer embedding, then Φ maps $\mathbb{F}_{p^l} \otimes 1$ into $\mathbb{F}_{p^m} \otimes 1$. Thus the restriction of Φ to $\mathbb{F}_{p^l} \otimes 1$ is of the form $\phi \otimes 1$ for some $\phi : \mathbb{F}_{p^l} \hookrightarrow \mathbb{F}_{p^m}$.*
- *Conversely, if $\phi : \mathbb{F}_{p^l} \hookrightarrow \mathbb{F}_{p^m}$ is an embedding of finite fields, then $\Phi = \phi \otimes \iota_{l,m}$ is a Kummer embedding.*

Moreover, this correspondence commutes with composition of embeddings.

Proof. Let Φ be a Kummer embedding. Being a $\mathbb{F}_p$-algebra morphism, it satisfies $\Phi(\beta^p) = \Phi(\beta)^p$ for all $\beta \in A_l$. By Lemma 2, this means that Φ commutes with $\sigma \otimes \sigma$, and thus, also with $(\sigma \otimes 1)^{-1} \circ (\sigma \otimes \sigma) = 1 \otimes \sigma$. This implies that Φ maps $\mathbb{F}_{p^l} \otimes 1$ into $\mathbb{F}_{p^m} \otimes 1$. The other assertions are clear. □

Corollary 7. *Let $\alpha_l \in A_l$ be a nonzero solution of* (H90) *for ζ_l, with Kummer constant c_l, and let $\hat{\alpha} \in A_m$ be a solution of* (H90) *for $(\zeta_m)^{\frac{m}{l}}$ that satisfies $(\hat{\alpha})^l = 1 \otimes \iota_{l,m}(c_l)$. Then:*

- *$\hat{\alpha} \in \mathbb{F}_{p^m} \otimes \mathbb{F}_p((\zeta_m)^{\frac{m}{l}}) \subseteq A_m$;*
- *the assignation $\lfloor \alpha_l \rfloor_{\zeta_l} \mapsto \lfloor \hat{\alpha} \rfloor_{(\zeta_m)^{\frac{m}{l}}}$ defines an embedding $\phi : \mathbb{F}_{p^l} \hookrightarrow \mathbb{F}_{p^m}$;*
- *$\Phi = \phi \otimes \iota_{l,m}$ is the unique Kummer embedding such that $\Phi(\alpha_l) = \hat{\alpha}$.*

Proof. By Proposition 5 there is a unique Kummer embedding Φ such that $\Phi(\alpha_l) = \hat{\alpha}$. By Proposition 6 we have that $\Phi = \phi \otimes \iota_{l,m}$ for some $\phi : \mathbb{F}_{p^l} \hookrightarrow \mathbb{F}_{p^m}$. Writing $\alpha_l = \sum_{i=0}^{a-1} x_i \otimes \zeta_l^i$, it follows that

$$\hat{\alpha} = \Phi(\alpha_l) = \sum_{i=0}^{a-1} \phi(x_i) \otimes (\zeta_m)^{\frac{mi}{l}}.$$

Thus $\lfloor \hat{\alpha} \rfloor_{(\zeta_m)^{\frac{m}{l}}} = \phi(x_0) = \phi(\lfloor \alpha_l \rfloor_{\zeta_l})$, and, since $\lfloor \alpha_l \rfloor_{\zeta_l}$ generates $\mathbb{F}_{p^l}$, this uniquely characterizes ϕ. □

We can now state Allombert's algorithm and prove its correctness; we give below a minor variation on the original algorithm, better adapted to our more general setting.

Proposition 8. *Algorithm 1 is correct: it returns elements that define an embedding $\phi : \mathbb{F}_{p^l} \hookrightarrow \mathbb{F}_{p^m}$.*

Proof. By Propositions 5 and 6, there exists $\hat{\alpha} \in A_m$ solution of (H90) for $(\zeta_m)^{\frac{m}{l}}$ that satisfies $(\hat{\alpha})^l = 1 \otimes \iota_{l,m}(c_l)$. On the other hand, $(\alpha_m)^{\frac{m}{l}} \in A_m$ is also a solution of (H90) for $(\zeta_m)^{\frac{m}{l}}$, thus $\hat{\alpha} = (1 \otimes \lambda)(\alpha_m)^{\frac{m}{l}}$ for some $\lambda \in \mathbb{F}_p(\zeta_m)$. It follows that $\iota_{l,m}(c_l)/c_m = \lambda^l$ is a l-th power, and $\kappa = (\zeta_m)^{\frac{um}{l}} \lambda$ for some integer u. Now we can replace $\hat{\alpha}$ with $(1 \otimes (\zeta_m)^{\frac{um}{l}})(\hat{\alpha}) = (1 \otimes \kappa)(\alpha_m)^{\frac{m}{l}}$ and conclude with Corollary 7. □

From this proof and Proposition 3, it follows that another choice of the l-th root κ only changes ϕ by a power of σ.

Algorithm 1 (Allombert's algorithm)

Input: $\mathbb{F}_{p^l}, \mathbb{F}_{p^m}$, for $l \mid m$ integers prime to p, and a cyclotomic lattice $\mathcal{S}^{\{l,m\}}$.

Output: $s \in \mathbb{F}_{p^l}, t \in \mathbb{F}_{p^m}$, such that the assignation $s \mapsto t$ defines an embedding $\phi : \mathbb{F}_{p^l} \hookrightarrow \mathbb{F}_{p^m}$.

1: Prepare the Kummer algebras A_l and A_m.
2: Find $\alpha_l \in A_l$ and $\alpha_m \in A_m$, nonzero solutions of (H90) for ζ_l and ζ_m respectively.
3: Compute their Kummer constants: $(\alpha_l)^l = 1 \otimes c_l$ and $(\alpha_m)^m = 1 \otimes c_m$.
4: Compute κ, a l-th root of $\iota_{l,m}(c_l)/c_m$.
5: Return $\lfloor \alpha_l \rfloor_{\zeta_l}$ and $\left\lfloor (1 \otimes \kappa)(\alpha_m)^{\frac{m}{l}} \right\rfloor_{(\zeta_m)^{\frac{m}{l}}}$.

4 STANDARD SOLUTIONS OF (H90)

Plugging Algorithm 1 into the Bosma–Cannon–Steel framework immediately gives a way to compatibly embed arbitrary finite fields. However, there are two points in Allombert's algorithm on which we would like to improve:

- ***Uniqueness:*** As mentioned, the element $\lfloor \alpha_l \rfloor_{\zeta_l}$ is a generating element for $\mathbb{F}_{p^l}$, or equivalently, it provides a defining irreducible polynomial of degree l. However this polynomial depends on the choice of α_l (even though only through c_l, cf. Proposition 3).
- ***Compatibility:*** The embedding ϕ depends on a constant κ, which itself depends on the choice of α_l, α_m (and of a l-th root extraction). Thus, given a certain number of finite fields, in order to ensure compatibility of the various embeddings between them, one has to keep track of these constants κ for all pairs (l, m), which grow *quadratically* with the number of fields.

It would be useful if one could force $\kappa = 1$, that is, if α_l, α_m and $\Phi : A_l \hookrightarrow A_m$ could be chosen so that $\Phi(\alpha_l) = (\alpha_m)^{\frac{m}{l}}$. From the description of Algorithm 1, this requires $c_m = \iota_{l,m}(c_l)$. Thus, necessarily c_m lies in the subfield $\mathbb{F}_p((\zeta_m)^{\frac{m}{l}})$ of $\mathbb{F}_p(\zeta_m)$. Possibly this condition could fail if A_l and A_m do not have the same field of scalars. This motivates:

Definition 9. *A Kummer algebra is* complete *if it is of the largest degree for a given level.*

Thus, the complete Kummer algebra of level a is

$$A_{p^a-1} = \mathbb{F}_{p^{p^a-1}} \otimes \mathbb{F}_{p^a}$$

with field of scalars $\mathbb{F}_{p^a} = \mathbb{F}_p(\zeta_{p^a-1})$ given by the corresponding ζ_{p^a-1} in our cyclotomic lattice $\mathcal{S}^I$, e.g., defined by a (pseudo)-Conway polynomial of degree a.

Lemma 10. *All nonzero solutions $\alpha_{p^a-1} \in A_{p^a-1}$ of* (H90) *for ζ_{p^a-1} have the same Kummer constant $c_{p^a-1} = (\zeta_{p^a-1})^a$.*

Proof. From Lemma 2 and the fact that σ^a is trivial on $\mathbb{F}_{p^a} \simeq \mathbb{F}_p(\zeta_{p^a-1})$ we get that

$$\begin{aligned}(\alpha_{p^a-1})^{p^a} &= (\sigma^a \otimes \sigma^a)(\alpha_{p^a-1}) = (\sigma^a \otimes 1)(\alpha_{p^a-1}) \\ &= (1 \otimes \zeta_{p^a-1})^a \alpha_{p^a-1}.\end{aligned}$$

We conclude since α_{p^a-1} is invertible. □

Definition 11. *Let l be an integer prime to p. We define the* standard *Kummer constant of order l as*

$$c_l^{\mathrm{std}} = (\iota_{l,p^a-1})^{-1}((\zeta_{p^a-1})^a) \in \mathbb{F}_p(\zeta_l)$$

where $a = v(l)$ is the level of A_l.

We say a solution $\alpha_l \in A_l$ of (H90) *for ζ_l is* standard *if its Kummer constant is standard:*

$$(\alpha_l)^l = 1 \otimes c_l^{\mathrm{std}}.$$

Then, by a decorated *Kummer algebra we mean a pair*

$$(A_l, \alpha_l)$$

with such α_l standard.

Observe that ι_{l,p^a-1} is an isomorphism when $a = v(l)$, so c_l^{std} is well defined.

For complete algebras, Lemma 10 asserts that all nonzero α_{p^a-1} are standard.

Proposition 12. *Let l be an integer not divisible by p. Then A_l can be decorated, i.e., it admits a standard α_l. Moreover, this α_l is unique up to a l-th root of unity.*

Proof. Let α_l' be any nonzero solution of (H90) for ζ_l. Set $a = v(l)$, pick any $\alpha_{p^a-1} \in A_{p^a-1}$ standard (Lemma 10), and pick any Kummer embedding $\Phi : A_l \hookrightarrow A_{p^a-1}$ (Proposition 6). Then $\Phi(\alpha_l')$ and $(\alpha_{p^a-1})^{\frac{p^a-1}{l}}$ are two nonzero solutions of (H90) for $\iota_{l,p^a-1}(\zeta_l) = (\zeta_{p^a-1})^{\frac{p^a-1}{l}}$ in A_{p^a-1}, thus there is a scalar $\lambda \in \mathbb{F}_p(\zeta_{p^a-1})$ such that

$$(\alpha_{p^a-1})^{\frac{p^a-1}{l}} = (1 \otimes \lambda)\Phi(\alpha_l') = \Phi((1 \otimes \tilde{\lambda})\alpha_l'),$$

where $\tilde{\lambda} = (\iota_{l,p^a-1})^{-1}(\lambda) \in \mathbb{F}_p(\zeta_l)$.

Setting $\alpha_l = (1 \otimes \eta\tilde{\lambda})\alpha_l' \in A_l$ for $\eta \in \mathbb{F}_p(\zeta_l)$, we get $c_l = \eta^l(\iota_{l,p^a-1})^{-1}(c_{p^a-1}^{\mathrm{std}}) = \eta^l c_l^{\mathrm{std}}$, and thus the standard $\alpha_l \in A_l$ are the $(1 \otimes \zeta_l^u\tilde{\lambda})\alpha_l'$, for $0 \le u < l$. □

Definition 13. *A generating element $s \in \mathbb{F}_{p^l}$ is called* standard *if it is of the form $s = \lfloor \alpha_l \rfloor_{\zeta_l}$ for $\alpha_l \in A_l$ a standard solution of* (H90).

The standard defining polynomial *P_l for $\mathbb{F}_{p^l}$ is then the minimal polynomial over $\mathbb{F}_p$ of such a standard s.*

By Proposition 3, we note that P_l is entirely determined by c_l^{std}, and thus, by the given cyclotomic lattice $\mathcal{S}^I$, possibly up to order $p^{v(l)} - 1$. As an example, we give in Table 1 the first ten standard polynomials induced by the system of Conway polynomials for $p = 2$ (thus, in this example, P_l only depends on the Conway polynomial of degree $v(l)$).

We remark that it is easy to extend the definitions of decorated algebras and standard elements to any extension degree, similarly to the way this is done for the basic Lenstra–Allombert algorithm. Use any (standard) construction for Artin-Schreier towers over finite fields (e.g., [13]), define decorated algebras by tensoring together Kummer algebras and Artin-Schreier extensions of $\mathbb{F}_p$, and define standard elements as, e.g., the product of a solution of multiplicative H90 and one of additive H90. While this solution is simple and

effective, it is rather orthogonal to our work, hence we omit the details here.

The decoration of an algebra A_l, and the associated standard generating element and polynomial for $\mathbb{F}_{p^l}$, can be computed by the simple adaptation of Allombert's algorithm presented below.

Algorithm 2 (Decoration – Standardization)

Input: $\mathbb{F}_{p^l}$, for l prime to p, and $\mathcal{S}^I$ a cyclotomic lattice.

Output: (A_l, α_l) decorated, P_l standard irreducible polynomial of degree l, and $s \in \mathbb{F}_{p^l}$ standard generating element inducing $\mathbb{F}_{p^l} \simeq \mathbb{F}_p[T]/(P_l)$.

1: Prepare the Kummer algebra A_l.
2: Prepare $c_l^{\mathrm{std}} = (\iota_{l,p^a-1})^{-1}((\zeta_{p^a-1})^a) \in \mathbb{F}_p(\zeta_l)$.
3: Find $\alpha'_l \in A_l$ nonzero solution of (H90) for ζ_l.
4: Compute its Kummer constant: $(\alpha'_l)^l = 1 \otimes c'_l$.
5: Compute κ a l-th root of c_l^{std}/c'_l.
6: Set $\alpha_l = (1 \otimes \kappa)\alpha'_l$.
7: Compute P_l the minimal polynomial of $\lfloor \alpha_l \rfloor_{\zeta_l}$ over $\mathbb{F}_p$.
8: Return (A_l, α_l), P_l, and $\lfloor \alpha_l \rfloor_{\zeta_l}$.

Algorithm 2 is correct, indeed Proposition 12 ensures that a standard α_l exists, and thus $\alpha'_l = (1 \otimes \kappa^{-1})\alpha_l$ for some $\kappa \in \mathbb{F}_p(\zeta_l)$, so $c_l^{\mathrm{std}}/c'_l = \kappa^l$ is a l-th power.

By design, decorated Kummer algebras of the same level admit standard Kummer embeddings, under which the corresponding standard solutions of (H90) are power-compatible:

Proposition 14. *Let $l \mid m$ be integers prime to p and such that $v(l) = v(m) = a$. Let (A_l, α_l), (A_m, α_m) be decorated Kummer algebras of degree l, m respectively (and of the same level a). Then, there is a unique Kummer embedding*

$$\Phi_{l,m}^{\mathrm{std}} : A_l \hookrightarrow A_m$$

such that $\Phi_{l,m}^{\mathrm{std}}(\alpha_l) = (\alpha_m)^{\frac{m}{l}}$.

Proof. Proposition 5 with $\hat{\alpha} = (\alpha_m)^{\frac{m}{l}}$. □

Most often we will apply Proposition 14 with $m = p^a - 1$.

On the other hand, since power-compatibility implies $c_m = \iota_{l,m}(c_l)$, it cannot be satisfied for a Kummer embedding between decorated Kummer algebras of different levels. However, at least between *complete* decorated algebras, we can request some *norm*-compatibility instead.

$x + 1$	$x^9 + x^7 + x^4 + x^2 + 1$
$x^3 + x + 1$	$x^{11} + x^8 + x^7 + x^6 + x^2 + x + 1$
$x^5 + x^3 + 1$	$x^{13} + x^{10} + x^5 + x^3 + 1$
$x^7 + x + 1$	$x^{15} + x + 1$
$x^{17} + x^{11} + x^{10} + x^8 + x^7 + x^6 + x^4 + x^3 + x^2 + x + 1$	
$x^{19} + x^{17} + x^{16} + x^{15} + x^{14} + x^{13} + x^{12} + x^8 + x^7 + x^6 + x^5 + x^3 + 1$	

Table 1: The first ten standard polynomials derived from Conway polynomials for $p = 2$.

Let A_m be a Kummer algebra of level $b = v(m)$, so

$$A_m = \mathbb{F}_{p^m} \otimes \mathbb{F}_p(\zeta_m) \xrightarrow{\sim} \mathbb{F}_{p^m} \otimes \mathbb{F}_p(\zeta_{p^b-1}),$$

where the isomorphism is given by $1 \otimes \iota_{m,p^b-1}$. Then, for an integer $a \mid b$, the subalgebra of A_m invariant under $1 \otimes \sigma^a$ is identified by this isomorphism with

$$(A_m)^{1\otimes\sigma^a} \simeq \mathbb{F}_{p^m} \otimes \mathbb{F}_p((\zeta_{p^b-1})^{\frac{p^b-1}{p^a-1}}),$$

where $(\zeta_{p^b-1})^{\frac{p^b-1}{p^a-1}} = \mathrm{N}_{\mathbb{F}_{p^b}/\mathbb{F}_{p^a}}(\zeta_{p^b-1}) = \iota_{p^a-1,p^b-1}(\zeta_{p^a-1})$, and with $\mathrm{N}_{\mathbb{F}_{p^b}/\mathbb{F}_{p^a}}$ the norm of the field extension $\mathbb{F}_{p^b}/\mathbb{F}_{p^a}$.

Definition 15. *Given a Kummer algebra A_n, and some integers $a \mid b \mid v(n)$, we define the* scalar norm *operator*

$$\begin{array}{rccc} \mathcal{N}_{b/a,A_n} : & (A_n)^{1\otimes\sigma^b} & \to & (A_n)^{1\otimes\sigma^a} \\ & \gamma & \mapsto & \prod_{0\le j<\frac{b}{a}} (1 \otimes \sigma^{ja})(\gamma). \end{array}$$

This is well-defined, i.e., the image of $\mathcal{N}_{b/a,A_n}$ is invariant under $1 \otimes \sigma^a$ as specified. Often the ambient algebra A_n will be implicit, and we will write $\mathcal{N}_{b/a}$ instead of $\mathcal{N}_{b/a,A_n}$.

By construction, $\mathcal{N}_{b/a}$ acts on $1 \otimes \mathbb{F}_{p^b}^\times$ as $1 \otimes \mathrm{N}_{\mathbb{F}_{p^b}/\mathbb{F}_{p^a}}$. Scalar norms are multiplicative:

$$\mathcal{N}_{b/a}(\gamma\gamma') = \mathcal{N}_{b/a}(\gamma)\mathcal{N}_{b/a}(\gamma'),$$

transitive:

$$\mathcal{N}_{c/a} = \mathcal{N}_{b/a} \circ \mathcal{N}_{c/b},$$

and they commute with $\sigma \otimes 1$.

Proposition 16. *Let $a \mid b$ be integers, and let $(A_{p^a-1}, \alpha_{p^a-1})$, $(A_{p^b-1}, \alpha_{p^b-1})$ be decorated complete Kummer algebras of level a, b respectively. Then there is a unique Kummer embedding*

$$\Phi_{p^a-1,p^b-1}^{\mathrm{std}} : A_{p^a-1} \hookrightarrow A_{p^b-1}$$

such that $\Phi_{p^a-1,p^b-1}^{\mathrm{std}}(\alpha_{p^a-1}) = \mathcal{N}_{b/a}(\alpha_{p^b-1})$.

Proof. By Lemma 2 and the properties of the norm,

$$\begin{aligned} (\mathcal{N}_{b/a}(\alpha_{p^b-1}))^{p^a} &= (\sigma^a \otimes \sigma^a)(\mathcal{N}_{b/a}(\alpha_{p^b-1})) \\ &= (\sigma^a \otimes 1)(\mathcal{N}_{b/a}(\alpha_{p^b-1})) \\ &= \mathcal{N}_{b/a}((\sigma^a \otimes 1)(\alpha_{p^b-1})) \\ &= \mathcal{N}_{b/a}((1 \otimes (\zeta_{p^b-1})^a)\alpha_{p^b-1}) \\ &= (1 \otimes \iota_{p^a-1,p^b-1}(\zeta_{p^a-1})^a)\mathcal{N}_{b/a}(\alpha_{p^b-1}). \end{aligned}$$

So $\hat{\alpha} = \mathcal{N}_{b/a}(\alpha_{p^b-1})$ satisfies $(\hat{\alpha})^{p^a-1} = 1 \otimes \iota_{p^a-1,p^b-1}(c_{p^a-1})$ and we conclude with Proposition 5. □

5 STANDARD EMBEDDINGS

In Proposition 14, we saw how to construct a standard power-compatible embedding of a decorated Kummer algebra into its decorated complete algebra, and in Proposition 16, a standard norm-compatible embedding between decorated complete algebras of dividing levels.

Now, consider general $l \mid m$ not divisible by p, set $a = v(l)$, $b = v(m)$, and consider the diagram

$$\begin{array}{ccc} (A_{p^a-1}, \alpha_{p^a-1}) & \xrightarrow{\Phi^{\mathrm{std}}_{p^a-1,p^b-1}} & (A_{p^b-1}, \alpha_{p^b-1}) \\ \Phi^{\mathrm{std}}_{l,p^a-1}\uparrow & & \uparrow\Phi^{\mathrm{std}}_{m,p^b-1} \\ (A_l, \alpha_l) & & (A_m, \alpha_m) \end{array}$$

of standard embeddings of decorated algebras.

Lemma 17. *In this setting, there exists a unique Kummer embedding*

$$\Phi^{\mathrm{std}}_{l,m} : A_l \hookrightarrow A_m$$

that makes the diagram commute.

Proof. Consider $\hat{\hat{\alpha}} = \Phi^{\mathrm{std}}_{p^a-1,p^b-1}(\Phi^{\mathrm{std}}_{l,p^a-1}(\alpha_l)) \in A_{p^b-1}$. Then $\hat{\hat{\alpha}}$ is invariant under $\sigma^l \otimes 1$ and under $1 \otimes \sigma^a$ (because α_l is), thus, *a fortiori*, invariant under $\sigma^m \otimes 1$ and under $1 \otimes \sigma^b$, which means it lies in the image of $\Phi^{\mathrm{std}}_{m,p^b-1}$. We can then set $\hat{\alpha} = (\Phi^{\mathrm{std}}_{m,p^b-1})^{-1}(\hat{\hat{\alpha}})$. If $\Phi^{\mathrm{std}}_{l,m}$ exists, then it necessarily maps α_l to $\hat{\alpha}$. However, chasing in the diagram, it is easily seen that $\hat{\alpha}$ is a solution of (H90) for $(\zeta_m)^{\frac{m}{l}}$ that satisfies $(\hat{\alpha})^l = 1 \otimes \iota_{l,m}(c_l)$, and we conclude with Proposition 5. □

This existence result is "constructive", but impractical, since it requires computations in the possibly very large algebra A_{p^b-1}. However, as in Algorithm 1, one should be able to write $\hat{\alpha} = (1 \otimes \kappa)(\alpha_m)^{\frac{m}{l}}$ for some $\kappa \in \mathbb{F}_p(\zeta_m)$. Moreover $\hat{\alpha}$ is uniquely determined by our data, thus, so should κ. Now our aim is to give an explicit expression for this $\kappa = \kappa_{l,m}$. We start with the case of complete algebras.

Proposition 18. *In the complete algebra A_{p^b-1} we have*

$$(\alpha_{p^b-1})^{\frac{p^b-1}{p^a-1}} = (1 \otimes \zeta_{p^b-1})^{\frac{(b-a)p^{b+a}-bp^b+ap^a}{(p^a-1)^2}} \mathcal{N}_{b/a}(\alpha_{p^b-1}).$$

Proof. Using first Lemma 2, and then (H90), we get:

$$\begin{aligned} \frac{(\alpha_{p^b-1})^{\frac{p^b-1}{p^a-1}}}{\mathcal{N}_{b/a}(\alpha_{p^b-1})} &= \prod_{0 \le j < \frac{b}{a}} \frac{(\sigma^{ja} \otimes \sigma^{ja})(\alpha_{p^b-1})}{(1 \otimes \sigma^{ja})(\alpha_{p^b-1})} \\ &= \prod_{0 \le j < \frac{b}{a}} (1 \otimes \sigma^{ja}) \left(\frac{(\sigma^{ja} \otimes 1)(\alpha_{p^b-1})}{\alpha_{p^b-1}} \right) \\ &= \prod_{0 \le j < \frac{b}{a}} (1 \otimes \sigma^{ja})(1 \otimes \zeta_{p^b-1})^{ja} \\ &= (1 \otimes \zeta_{p^b-1})^{\sum_{0 \le j < \frac{b}{a}} jap^{ja}}. \end{aligned}$$

We conclude thanks to the identity

$$\sum_{0 \le j < n} jT^j = T\frac{d}{dT}\left(\frac{T^n - 1}{T - 1}\right) = \frac{(n-1)T^{n+1} - nT^n + T}{(T-1)^2}.$$

□

Corollary 19. *Let (A_l, α_l) and (A_m, α_m) be decorated Kummer algebras, of respective degrees $l \mid m$ prime to p. Then the standard Kummer embedding $\Phi^{\mathrm{std}}_{l,m} : A_l \hookrightarrow A_m$ is defined by the assignation $\alpha_l \mapsto (1 \otimes \kappa_{l,m})(\alpha_m)^{\frac{m}{l}}$, where*

$$\kappa_{l,m} = (\iota_{m,p^b-1})^{-1}((\zeta_{p^b-1})^{-\frac{(b-a)p^{b+a}-bp^b+ap^a}{(p^a-1)l}}).$$

Proof. It suffices to check that $\Phi^{\mathrm{std}}_{p^a-1,p^b-1}(\Phi^{\mathrm{std}}_{l,p^a-1}(\alpha_l))$ and the image of the right-hand-side under $\Phi^{\mathrm{std}}_{m,p^b-1}$ coincide in A_{p^b-1}. However, we have $\Phi^{\mathrm{std}}_{p^a-1,p^b-1}(\Phi^{\mathrm{std}}_{l,p^a-1}(\alpha_l)) = \mathcal{N}_{b/a}(\alpha_{p^b-1})^{\frac{p^a-1}{l}}$, while $\Phi^{\mathrm{std}}_{m,p^b-1}((\alpha_m)^{\frac{m}{l}}) = (\alpha_{p^b-1})^{\frac{p^b-1}{l}}$, and we conclude with Proposition 18 □

Proposition 20. *Standard Kummer embeddings are compatible with composition: if (A_l, α_l), (A_m, α_m), and (A_n, α_n) are decorated Kummer algebras with $l \mid m \mid n$, the corresponding standard embeddings satisfy $\Phi^{\mathrm{std}}_{l,n} = \Phi^{\mathrm{std}}_{m,n} \circ \Phi^{\mathrm{std}}_{l,m}$.*

Proof. We have to show that $\Phi^{\mathrm{std}}_{l,n}(\alpha_l) = \Phi^{\mathrm{std}}_{m,n}(\Phi^{\mathrm{std}}_{l,m}(\alpha_l))$; nothing but a pleasant computation with the explicit constants given by Corollary 19.

Alternatively, set $a = v(l), b = v(m), c = v(n)$, and decorate $A_{p^a-1}, A_{p^b-1}, A_{p^c-1}$. It suffices to show that the elements $\Phi^{\mathrm{std}}_{l,n}(\alpha_l)$ and $\Phi^{\mathrm{std}}_{m,n}(\Phi^{\mathrm{std}}_{l,m}(\alpha_l))$ have the same image under $\Phi^{\mathrm{std}}_{n,p^c-1}$ in A_{p^c-1}. Chasing in the diagram

$$\begin{array}{ccccc} A_{p^a-1} & \longrightarrow & A_{p^b-1} & \longrightarrow & A_{p^c-1} \\ \uparrow & & \uparrow & & \uparrow \\ A_l & \longrightarrow & A_m & \longrightarrow & A_n \end{array}$$

we see that this common image is $\mathcal{N}_{c/a}(\alpha_{p^c-1})^{\frac{p^a-1}{l}}$. □

By a *decorated finite field* (of degree l, an integer prime to p, and relative to a given cyclotomic lattice $\mathcal{S}^I$), we mean a pair $(\mathbb{F}_{p^l}, s_l)$, where $\mathbb{F}_{p^l}$ is a finite field, and $s_l \in \mathbb{F}_{p^l}$ a standard generating element in the sense of Definition 13.

We can finally state:

Algorithm 3 (Standard compatible embeddings)

Input: $\mathcal{S}^I$ a cyclotomic lattice, and $(\mathbb{F}_{p^l}, s_l)$, $(\mathbb{F}_{p^m}, s_m)$, decorated finite fields, for $l \mid m$ integers prime to p.

Output: $t \in \mathbb{F}_{p^m}$, such that the assignation $s_l \mapsto t$ defines a standard embedding $\phi^{\mathrm{std}}_{l,m} : \mathbb{F}_{p^l} \hookrightarrow \mathbb{F}_{p^m}$, compatible with composition.

1: Prepare the Kummer algebras A_l and A_m.
2: Recover α_l from s_l and α_m from s_m using equations (1).
3: Compute $\kappa_{l,m} = (\iota_{m,p^b-1})^{-1}((\zeta_{p^b-1})^{-\frac{(b-a)p^{b+a}-bp^b+ap^a}{(p^a-1)l}})$ where $a = v(l)$, $b = v(m)$.
4: Return $\left[(1 \otimes \kappa)(\alpha_m)^{\frac{m}{l}}\right]_{(\zeta_m)^{\frac{m}{l}}}$.

Proposition 21. *Standard finite field embeddings are compatible with composition: if $(\mathbb{F}_{p^l}, \alpha_l)$, $(\mathbb{F}_{p^m}, \alpha_m)$, and $(\mathbb{F}_{p^n}, \alpha_n)$ are decorated finite fields with $l \mid m \mid n$, the corresponding standard embeddings satisfy $\phi^{\mathrm{std}}_{l,n} = \phi^{\mathrm{std}}_{m,n} \circ \phi^{\mathrm{std}}_{l,m}$.*

Proof. Corollary 7 and Proposition 20. □

6 IMPLEMENTATION

In the previous sections we kept the description of Kummer embeddings abstract, leaving many computational details unspecified. There are various ways in which our algorithms can possibly be implemented, depending on how one chooses to represent finite fields and the cyclotomic lattice $\mathcal{S}^I$. A reasonable option is to use (pseudo)-Conway polynomials to represent the fields $\mathbb{F}_p(\zeta_{p^a-1})$, and deduce from them the smallest possible representation for any other field $\mathbb{F}_p(\zeta_l)$. Assuming this technique, we can prove a bound on the complexity of our algorithms.

Proposition 22. *Given a collection of (pseudo)-Conway polynomials for $\mathbb{F}_p$, of degree up to d, standard solutions α_l of* (H90) *can be computed for any $l \mid (p^i - 1)$ for any $i \le d$ using $O\big(\mathsf{M}(l^2)\log(l) + \mathsf{M}(l)\log(l)\log(p)\big)$ operations. After that, Kummer embeddings $\mathbb{F}_{p^l} \subseteq \mathbb{F}_{p^m}$ can be computed using $O(\mathsf{M}(m^2)\log(m))$ operations.*

Proof. Let $a = \nu(l)$ be the level of A_l. We take the a-th polynomial from the collection of (pseudo)-Conway polynomials, and use it to define ζ_{p^a-1}. Because $a \in O(l)$, the cost of multiplications in $\mathbb{F}_p(\zeta_{p^a-1})$ will be bounded by $O(\mathsf{M}(l))$.

From ζ_{p^a-1}, we compute ζ_l using $O(l\mathsf{M}(l))$ operations, and its minimal polynomial in $O(l^{(\omega+1)/2})$. Then, the Kummer constant $c^{\mathrm{std}}_l = (\zeta_{p^a-1})^a$ is computed in negligible time, and its expression in the power basis of ζ_l is computed in $O(l^{(\omega+1)/2})$ using the algorithms for evaluating embeddings mentioned in Section 2.

To construct the Kummer algebra $A_l = \mathbb{F}_{p^l} \otimes \mathbb{F}_p(\zeta_l)$ we need an irreducible polynomial of degree l, not necessarily related to the (pseudo)-Conway polynomials used to represent the fields of scalars. Very efficient, quasi-optimal algorithms for finding such a polynomial are given in [4, 10, 11], we can thus neglect this cost.

The cost of computing a solution α'_l to (H90) was extensively studied in [8], where it was found to be bounded by $O\big(\mathsf{M}(l^2)\log(l) + \mathsf{M}(l)\log(p)\big)$. Then, the constant $c'_l = (\alpha'_l)^l$ is computed using $O\left(\mathsf{M}(l^2)\log(l)\right)$ operations, and the l-th root κ is computed in $O(\mathsf{M}(l)\log(l)\log(p))$ according to [8]. $\alpha_l = (1 \otimes \kappa)\alpha'_l$ is then computed in a negligible number of operations.

Finally, the projection $\lfloor \alpha_l \rfloor_{\zeta_l}$ comes for free, and its minimal polynomial P_l is computed again in $O(l^{(\omega+1)/2})$ operations.

Now, in order to compute a Kummer embedding of $(\mathbb{F}_{p^l}, \alpha_l)$ into $(\mathbb{F}_{p^m}, \alpha_m)$, we compute the scalar $\kappa_{l,m}$ in $O(m\mathsf{M}(m))$ operations, and $(\alpha_m)^{\frac{m}{l}}$ in $O(\mathsf{M}(m^2)\log(m))$.

We then need to convert $(\alpha_m)^{\frac{m}{l}}$ in the power basis of ζ_l. Applying a generic change of basis algorithm as before would be too expensive: indeed we have to convert m coefficients from the field of scalars $\mathbb{F}_p(\zeta_m)$ to $\mathbb{F}_p(\zeta_l)$, which would cost $O(m^{(\omega+3)/2})$. Instead we notice that we are only interested in the value $\left\lfloor (1 \otimes \kappa)(\alpha_m)^{\frac{m}{l}} \right\rfloor_{\zeta_l}$, therefore we proceed as follows.

Let Tr denote the trace map from $\mathbb{F}_p(\zeta_m)$ to $\mathbb{F}_p(\zeta_l)$, and let $\eta \in \mathbb{F}_p(\zeta_m)$ be such that $\mathrm{Tr}(\eta) = 1$. Then the map $x \mapsto \mathrm{Tr}(x\eta)$ sends $(\zeta_m)^{\frac{m}{l}}$ to ζ_l, and is $\mathbb{F}_p(\zeta_l)$-linear, it thus agrees with the inverse map of $\zeta_l \mapsto (\zeta_m)^{\frac{m}{l}}$ on the image of $\mathbb{F}_p(\zeta_l)$.

We thus need to evaluate $x \mapsto \lfloor \mathrm{Tr}(x\eta) \rfloor_{\zeta_l}$ for many values in $\mathbb{F}_p(\zeta_m)$, but this is a $\mathbb{F}_p$-linear form, hence we can precompute its vector on the power basis of ζ_m. Let h_m, h_l be the minimal polynomials of ζ_m, ζ_l, and let b, a be their degrees. Let h_0 be the constant coefficient of h_l, and let

$$\tau = -\frac{h_0}{(\zeta_m)^{\frac{m}{l}}} \frac{h'_m(\zeta_m)}{h'_l((\zeta_m)^{\frac{m}{l}})} \in \mathbb{F}_p(\zeta_m),$$

direct calculation shows that

$$\sum_{i=0}^{b-1} \left\lfloor \mathrm{Tr}(\zeta_m^i) \right\rfloor_{\zeta_l} Z^i = \frac{\tau(Z^{-1})}{Z h_m(Z^{-1})} \mod Z^b,$$

where by $\tau(Z)$ we mean $\tau \in \mathbb{F}_p(\zeta_m)$ seen as a polynomial in ζ_m. Hence, we can compute the vector of the linear form $x \mapsto \lfloor \mathrm{Tr}(x) \rfloor_{\zeta_l}$ using only basic polynomial arithmetic and modular composition, i.e., in $O(m^{(\omega+1)/2})$ operations.

Finally, we compute $(1 \otimes \kappa\eta)(\alpha_m)^{\frac{m}{l}}$, we see it as a polynomial with coefficients in $\mathbb{F}_p(\zeta_m)$, and we apply the map $\lfloor \mathrm{Tr}(x) \rfloor_{\zeta_l}$ to each coefficient. This costs $O(m\mathsf{M}(m))$ operations. □

We remark that storing the decorated fields $(\mathbb{F}_{p^l}, \alpha_l)$ requires $O(l^2)$ field elements, however, using the formulas in [1, 8], it is possible to only store $\lfloor \alpha_l \rfloor_{\zeta_l}$, and recover all other coefficients of α_l in $O(l\mathsf{M}(l)\log(p))$ operations.

To demonstrate the feasibility of this approach, we implemented it in the Julia-based CAS Nemo [15], with performance critical routines written in C/Flint [16]. Our code is available as a Julia package at https://github.com/erou/LatticeGFH90.jl.

We tested Algorithms 2 and 3 for various small primes, using precomputed Conway polynomials available in Nemo. We do not see major differences between different primes. In Figure 1 we report timings obtained for the case $p = 3$, on an Intel Core i7-7500U CPU clocked at 2.70GHz, using Nemo 0.11.1 running on Julia 1.1.0, and Nemo's current version of Flint. The plot on the left shows timings for Algorithm 2, for degrees l growing from 1 to 200, for every l coprime to p and such that the $\nu(l)$-th Conway polynomial is available in Nemo; the color scale shows the level of the associated algebra A_l. The bottleneck of this algorithm appears to be the l-th root extraction routine.

The plot on the right shows timings for Algorithm 3, measured by computing the standard embedding of $\mathbb{F}_{p^2}$ in $\mathbb{F}_{p^l}$. As expected, computing the embeddings takes negligible time in comparison to the decoration of the finite fields. We also tested embedding fields larger than $\mathbb{F}_{p^2}$, and noticed that the running time mostly depends on the size of the larger field.

7 CONCLUSION AND FUTURE WORK

We presented a new family of standard compatible polynomials for defining finite fields. Its construction being dependent on the availability of Conway polynomials, it has, at the present moment, very little practical impact; its existence is nevertheless remarkable in itself.

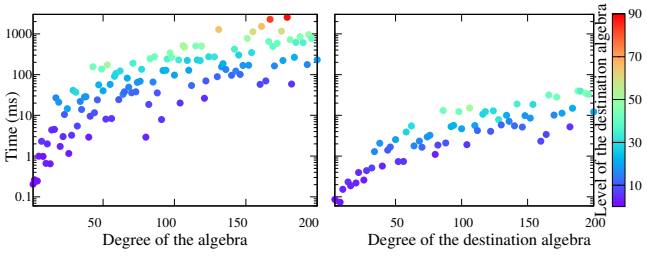

Figure 1: Timings for computing decorated fields $(\mathbb{F}_{p^l}, \alpha_l)$ (left, log scale), and for computing the standard Kummer embedding from $\mathbb{F}_{p^2}$ to $\mathbb{F}_{p^l}$ (right) for $p = 3$.

It is even evident that computing our standard polynomials is essentially equivalent to computing Conway polynomials; indeed from α_l one can immediately deduce $(\zeta_{p^a-1})^a$, and by taking an a-th root (doable in polynomial time in l), deduce ζ_{p^a-1} and the associated Conway polynomial. Hence, an efficient algorithm for computing our polynomials (for arbitrary degrees) would imply an efficient algorithm to compute Conway polynomials, which would be unexpected.

However, our proposed implementation is not the only possible way to exploit our definitions. It would be interesting, indeed, to find some middle ground between the flexibility of the Bosma–Steel–Cannon framework and the rigidity of Conway polynomials, for example by lazily enforcing the conditions required to have a standard solution of (H90), while incrementally constructing the lattice of roots of unity.

Another line of work would be to give a complete implementation of a lattice of finite fields, not limited to extensions of degree coprime to p. We leave these questions for future work.

ACKNOWLEDGMENTS

We would like to thank the anonymous reviewers for their useful comments. We thank Éric Schost for fruitful discussions and for helping bootstrap this work during a visit by two of the authors to the University of Waterloo. We acknowledge financial support from the French ANR-15-CE39-0013 project *Manta*, the *OpenDreamKit* Horizon 2020 European Research Infrastructures project (#676541), and from the French Domaine d'Intérêt Majeur *Math'Innov*.

REFERENCES

[1] Bill Allombert. 2002. Explicit Computation of Isomorphisms between Finite Fields. *Finite Fields and Their Applications* 8, 3 (2002), 332 – 342.
[2] Wieb Bosma, John Cannon, and Catherine Playoust. 1997. The MAGMA algebra system I: the user language. *Journal of Symbolic Computation* 24, 3-4 (1997), 235–265. https://doi.org/10.1006/jsco.1996.0125
[3] Wieb Bosma, John Cannon, and Allan Steel. 1997. Lattices of compatibly embedded finite fields. *Journal of Symbolic Computation* 24, 3-4 (1997), 351–369. https://doi.org/10.1006/jsco.1997.0138
[4] Alin Bostan, Philippe Flajolet, Bruno Salvy, and Éric Schost. 2006. Fast computation of special resultants. *Journal of Symbolic Computation* 41, 1 (2006), 1–29.
[5] Alin Bostan, Grégoire Lecerf, and Éric Schost. 2003. Tellegen's principle into practice. In *ISSAC'03*. ACM, 37–44. https://doi.org/10.1145/860854.860870
[6] Richard P. Brent and H.-T. Kung. 1978. Fast Algorithms for Manipulating Formal Power Series. *J. ACM* 25, 4 (1978), 581–595. https://doi.org/10.1145/322092.322099
[7] Ludovic Brieulle, Luca De Feo, Javad Doliskani, Jean-Pierre Flori, and Éric Schost. 2017. Computing isomorphisms and embeddings of finite fields (extended version). *arXiv preprint arXiv:1705.01221* (2017). https://arxiv.org/abs/1705.01221
[8] Ludovic Brieulle, Luca De Feo, Javad Doliskani, Jean-Pierre Flori, and Éric Schost. 2019. Computing isomorphisms and embeddings of finite fields. *Math. Comp.* 88 (2019), 1391–1426. https://doi.org/10.1090/mcom/3363
[9] P. Bürgisser, M. Clausen, and M. A. Shokrollahi. 1997. *Algebraic Complexity Theory*. Springer.
[10] Jean-Marc Couveignes and Reynald Lercier. 2013. Fast construction of irreducible polynomials over finite fields. *Israel Journal of Mathematics* 194, 1 (01 Mar 2013), 77–105. https://doi.org/10.1007/s11856-012-0070-8
[11] Luca De Feo, Javad Doliskani, and Éric Schost. 2013. Fast algorithms for ℓ-adic towers over finite fields. In *ISSAC'13*. ACM, 165–172.
[12] Luca De Feo, Javad Doliskani, and Éric Schost. 2014. Fast Arithmetic for the Algebraic Closure of Finite Fields. In *ISSAC '14*. ACM, 122–129. https://doi.org/10.1145/2608628.2608672
[13] Luca De Feo and Éric Schost. 2012. Fast arithmetics in Artin-Schreier towers over finite fields. *Journal of Symbolic Computation* 47, 7 (2012), 771–792. https://doi.org/10.1016/j.jsc.2011.12.008
[14] Javad Doliskani and Éric Schost. 2015. Computing in degree 2^k-extensions of finite fields of odd characteristic. *Designs, Codes and Cryptography* 74, 3 (01 Mar 2015), 559–569. https://doi.org/10.1007/s10623-013-9875-7
[15] Claus Fieker, William Hart, Tommy Hofmann, and Fredrik Johansson. 2017. Nemo/Hecke: Computer Algebra and Number Theory Packages for the Julia Programming Language. In *ISSAC '17*. ACM, 157–164. https://doi.org/10.1145/3087604.3087611
[16] William Hart, Fredrik Johansson, and Sebastian Pancratz. 2013. *FLINT: Fast Library for Number Theory*. http://flintlib.org Version 2.4.0.
[17] Lenwood S. Heath and Nicholas A. Loehr. 1999. New algorithms for generating Conway polynomials over finite fields. In *SODA '99*. SIAM, 429–437.
[18] Kiran S. Kedlaya and Christopher Umans. 2011. Fast Polynomial Factorization and Modular Composition. *SIAM J. Comput.* 40, 6 (2011), 1767–1802. https://doi.org/10.1137/08073408X
[19] Hendrik W. Lenstra. 1991. Finding isomorphisms between finite fields. *Math. Comp.* 56, 193 (1991), 329–347.
[20] Hendrick W. Lenstra Jr. and Bart de Smit. 2013. *Standard models for finite fields*. Chapman and Hall/CRC, Chapter 11.7 in Handbook of Finite Fields, 401–404. https://doi.org/10.1201/b15006
[21] Anand Kumar Narayanan. 2018. Fast Computation of Isomorphisms Between Finite Fields Using Elliptic Curves. In *WAIFI 2018 (LNCS)*, Vol. 11321. Springer.
[22] Werner Nickel. 1988. Endliche Körper in dem gruppentheoretischen Programmsystem GAP. (1988). https://www2.mathematik.tu-darmstadt.de/~nickel/
[23] David Roe, Jean-Pierre Flori, and Peter Bruin. 2013. Implement pseudo-Conway polynomials. Trac ticket #14958. (Oct. 2013). https://trac.sagemath.org/ticket/14958
[24] Victor Shoup. 1994. Fast construction of irreducible polynomials over finite fields. *Journal of Symbolic Computation* 17, 5 (1994), 371–391. https://doi.org/10.1006/jsco.1994.1025
[25] Victor Shoup. 1999. Efficient computation of minimal polynomials in algebraic extensions of finite fields. In *ISSAC'99*. ACM, 53–58. https://doi.org/10.1145/309831.309859
[26] The GAP Group 2018. *GAP – Groups, Algorithms, and Programming, Version 4.9.2*. The GAP Group. https://www.gap-system.org
[27] The Sage Developers 2019. *SageMath, the Sage Mathematics Software System (Version 8.7)*. The Sage Developers. https://www.sagemath.org
[28] Joachim von zur Gathen and Jurgen Gerhard. 1999. *Modern Computer Algebra*. Cambridge University Press, New York, NY, USA.