ON THE SHORT PRINCIPAL IDEAL PROBLEM OVER SOME REAL KUMMER FIELDS

ANDREA LESAVOUREY[1], THOMAS PLANTARD[2], AND WILLY SUSILO[2]

Abstract. Several cryptosystems using structured lattices have been believed to be quantum resistant. Their security can be linked to the hardness of solving the Shortest Vector Problem over module or ideal lattices. During the past few years it has been shown that the related problem of finding a short generator of a principal ideal can be solved in quantum polynomial time over cyclotomic fields, and classical polynomial time over a range of multiquadratic and multicubic fields. Hence, it is important to study as many as possible other number fields, to improve our knowledge of the aformentioned problems. In this paper we generalise the work done over multiquadratic and multicubic fields to a larger range of real Kummer extensions of prime exponent p. Moreover, we extend the analysis by studying the Log-unit lattice over these number fields, in comparison to already studied fields.

Key words and phrases: Post-quantum cryptography, Ideal lattices, Short Principal Ideal Problem, Kummer fields, Log-units

1. Introduction

One of the most important family of cryptosystems explored as a post-quantum solution is based on euclidean lattices. For efficiency reasons most of these systems use structured lattices, and their security can be linked to the problems of finding a short vector in a module lattice, namely the *Module Shortest Vector Problem* (MSVP), or in an ideal lattice which is the *Ideal Shortest Vector Problem* (ISVP). The oldest and simplest cryptosystems using ideal lattices such as in [21, 22, 34] are based on the related problem of finding a short generator of a principal ideal. They can be described as follows. Consider a number field K and $I = g\mathcal{O}_K$ a principal ideal with a short g when I is considered as a lattice. Then K and I are public and g is private. The private key security relies on the hardness of finding g. Finding a generator is called the *Principal Ideal Problem* (PIP) and is referred as one of the main tasks of Computational Number Theory by H. Cohen in [14]. Finding a short generator is referred as the *Short Principal Ideal Problem* (SPIP). A generic way of recovering g is done in two steps:

(1) recover a generator h of I;
(2) find a short generator given h.

As mentioned previously, the first step is considered to be a hard problem in classical computational number theory and the best known generic algorithm runs in sub-exponential time [14]. However, it can be computed in polynomial time with quantum computing as in [6], for any principal ideal. The second step is a reduction phase which is the kind of tasks that seem difficult even with quantum computing. In order to solve it, one may use the structure of the set of generators of I and the Log-unit lattice. This strategy was mentioned in [12] where it was claimed that in the case of cyclotomic fields the group of cyclotomic units has a good enough geometry in the Log-unit lattice to help recovering a short generator. A proper analysis over cyclotomic fields has been done by Cramer et al. in [17] where the authors gave a bound for the norm of the vectors of the dual basis. In [2] Bauch et al. studied

[1] Univ Rennes, CNRS, IRISA

[2] School of Computing and Information Technology, Faculty of Engineering and Information Sciences, University of Wollongong

E-mail addresses: andrea.lesavourey@irisa.fr, thomas.plantard@gmail.com, wsusilo@uow.edu.au.

Acknowledgments: Andrea Lesavourey is funded by the Direction Générale de l'Armement (Pôle de Recherche CYBER), with the support of Région Bretagne.

another family of fields, namely multiquadratic fields, and were able to recover a short generator of an ideal in classical polynomial time for a wide range of fields. This latest approach has been generalised to multicubic fields in [27].

Even though the actual propositions of lattice based cryptosystems rely on other problems such as the MSVP or the ISVP, it is important to study the SPIP. Indeed such works can help determining which fields or structures are weak. Moreover one could build upon such analysis a successful strategy for harder problems, or even draw a definitive line between these problems. Furthermore one has to remark that solving an instance of the SPIP is one step of the strategy to solve the ISVP introduced by Cramer et al. [16], and that the other line of work – namely the PHS algorithm – initiated by Hanrot et al. [29] and modified by Bernard and Roux-Langlois in [3] can be seen as an extension of the strategy described above to solve the SPIP. Indeed, it can be described as using a Log-S-unit lattice to shorten an element α of the ideal considered.

Finally from a post-quantum perspective the PIP can be solved in polynomial time. Indeed, all the number theoretical objects used can be computed efficiently following [6, 19]. Over general number fields the last unknown is therefore the possibility of retrieving a short generator using the Log-unit lattice. In order to study these problems without a quantum computer, it is important to obtain more efficient algorithms to be able to operate over number fields with large degrees.

Our contribution. As mentioned earlier, the SPIP is shown to be solvable with quantum computers over cyclotomic fields [17], and experimental data from [2, 27] indicate that it is also the case over multiquadratic and multicubic fields. However the methods are not similar. Over the two last families, the algorithms use the strong structure of the set of subfields. The authors of [2, 27] show that the unit group of high degree number fields can be computed in a reasonable amount of time (polynomial in the degree for a wide range of number field), as well as generators of principal ideals. We generalised these works to all real Kummer extensions of prime exponent p, i.e. generated by p-th roots of integers. We also considered real Kummer extensions of $\mathbb{Q}$ with two exponents – generated by p-th and q-th roots of integers where p and q are prime integers – in order to break the structure and see if one can still solve the SPIP with a good probability. Moreover, once implemented, our algorithms allowed us to study the Log-unit lattice of these fields and classify them with respect to their security level. Our implementation in Magma V2.24-9 is publicly available [1].

In this work we:

(1) describe algorithms to compute the unit group and solve the PIP of Kummer extensions;
(2) study the hardness of solving the SPIP over real Kummer fields using our implementation of these algorithms.

In particular we were able to evaluate the probability of success of shortening a generator with the Log-unit lattice, and study the quality of the basis obtained for this lattice. Our implementation allowed us to study high dimensional fields, and the data gathered highlights the need for considering such fields to draw conclusions on asymptotic behaviours. We therefore divided them in two categories: fields of degree less than 120 are called *low dimensional fields* and the others are called *high dimensional fields*. One can find in Table 1 a summary of the results obtained from our computations.

From the experimental data that we computed, general Kummer extensions of $\mathbb{Q}$ with only one exponent seem to show the same properties than multiquadratic fields. In particular we obtained high probabilities to retrieve private keys for a wide range of fields. However, within this family, we were able to identify a subcategory over which solving the SPIP is more difficult than over other fields. Indeed the probability of success of solving the SPIP is smaller for fields with degree p^2 and defined by small integers, especially (2,3). Moreover the data computed on the key and the basis of the Log-unit lattice show that the quality of the basis obtained is not as good as over cyclotomic fields, and cannot be used to solve the SPIP over high dimensional fields. This is highlighted by the

[1] https://github.com/AndLesav/spip-on-kummer

TABLE 1. Summary of the data obtained with: probability of shortening a generator, quality of the basis obtained, and for which category of fields the data is available

Field	Dimension achieved	Probability of shortening	Quality of the basis
Cyclotomics [17]	High	High	Good
Multiquadratics [2]	High	High	Good
Multicubics [27]	High	High	Good
Most of Kummer of exponent p	High	High	Good
Kummer with two exponents p, q	Low	High	Good
Kummer of degree p^3 defined by small integers (2,3,5)	High	High	Medium
Kummer of degree p^2 defined by small integers (2,3)	High	Low	Bad

fact that over these fields, the enumeration cost necessary to retrieve the Log-embedding of a short generator g grows exponentially with the rank of Log-unit lattice, which is not the case for cyclotomic fields. This is shown in Figure 1.

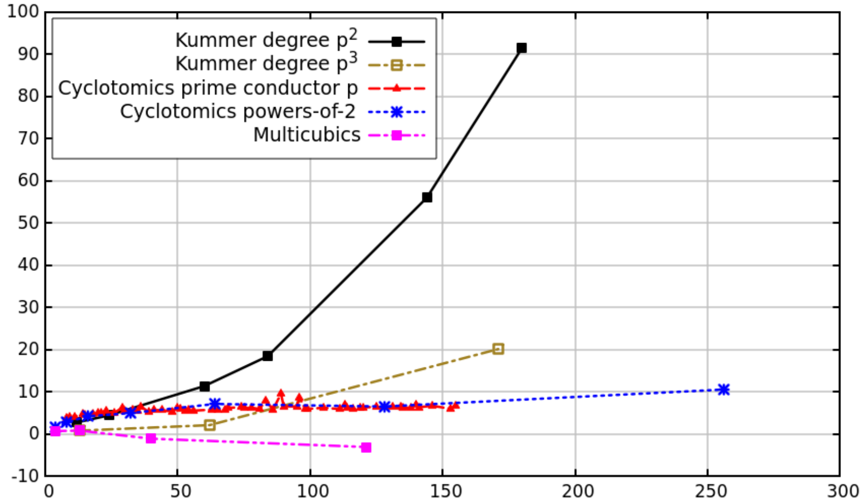


FIGURE 1. Median values of the bit-size of enumeration costs for retrieving a key $\mathrm{Log}_K(g)$, plotted against $r_1 + r_2 - 1$ for Kummer fields of degrees p^2 and p^3, and cyclotomic fields after BKZ$_{20}$ reductions

Moreover, the data gathered on Kummer fields with degree p^3 show that the basis obtained for the Log-unit lattice is already well reduced – since using BKZ instead of LLL does not have much impact – but with worse orthgonality parameters than for cyclotomic fields.

All of these observations can indicate than Kummer fields – especially with degree p^2 and defined by small integers – could be an alternative to cyclotomic fields for asymptotic cryptography. Obviously, more work is required to infirm or confirm this. We stress that these observations can be made only because we were able to compute the units of such fields for dimensions larger than 121, where significant differences between the type of fields truly appear. This leads us to think that one should always consider high dimensional fields (if the computational power at hand allows it) when studying problems such as the SPIP or the ISVP.

We were not able to compute as much data for Kummer extensions with two exponents, particularly for high dimensional fields. The data gathered seems to show that these extensions have the same global properties than Kummer fields with one exponent, despite behaviours which are less consistent. Further improvements could be necessary to confirm it.

Related work. Biasse et al. generalised in a concurrent work [9] the approach of [2, 8, 27] to compute the unit group, S-units and the class group to normal fields. They give necessary and sufficient conditions for the existence of what is called *norm relations*, which allow to design algorithms based on reduction to computations into subfields, in order to compute several number theoretical objects such as the maximal order, S-units and the class group. The algorithms we designed for general Kummer – Algorithms 1 and 2 – can be seen as a specialisation of their work, when considering normal Kummer extensions. The core of our study are *real* Kummer fields, which are not normal, so [9] cannot be applied directly. Finally, note that our implementation is a necessary part of our work, and that Biasse et al. did not provide an implementation regarding the computation of objects like S-units that could be applied or easily modified to suit our needs.

Future work. Further work can consist in studying other important tasks of computational number theory over these fields such as computing the class group and S-units. The authors of [8] provide a polynomial time algorithms for these over multiquadratic fields. It could be possible to implement the algorithms presented and studied in [29, 3] to solve the ISVP, and compare its performance over Kummer extensions and cyclotomic fields, especially since a recent work implemented the Tw-PHS algorithm from [3] over large degree cyclotomic fields, and using a sublattice of the Log-S-unit lattice [4]. Finally, the work of Biasse et al. [9] could be used to extend these considerations to a variety of other number fields.

Organisation of the paper. The rest of the article is organised as follows.

- In Section 2, we give some useful background on lattices, number fields and the SPIP.
- In Section 3, we describe the number field extensions L/K we are interested in, i.e. general Kummer extensions of degree p^2 with p a prime integer. We provide general recursive algorithms to compute $\mathcal{O}_L^\times$ and solve the PIP following the framework of the ones in [2, 27].
- In Section 4 we study number fields for which we implemented said procedures, i.e. real fields of the form $\mathbb{Q}(\sqrt[p]{m_1}, \ldots, \sqrt[p]{m_r}, \sqrt[q]{n_1}, \ldots, \sqrt[q]{n_s})$ with p and q primes. In particular, if we write P for the product of prime integers dividing $\prod_{i,j} m_i n_j$, we:
  - recall some results from a note of the first author [25] concerning the splitting of prime ideals in these number fields and their discriminants – Propositions 10 and 11 describe in what manner it depends on P in some cases – which leads to exhibiting a $\mathbb{Q}$-basis of L useful for implementation;
  - give details on some of the auxiliary procedures used in our implementation, such as p-th roots extraction;
  - describe heuristic algorithms to compute $\mathcal{O}_L^\times$ and solve the PIP running in time
  $$\mathrm{Poly}(\ln |D_L|) e^{\tilde{O}((\ln P)^{2/3})}$$
  and
  $$\mathrm{Poly}(\ln |D_L|, \ln \mathrm{N}(I)) e^{\tilde{O}((\ln P)^{2/3})}.$$
- We provide data gathered from our implementation in Section 5 and study the possibility of solving the SPIP over real Kummer extensions. In particular we are able to evaluate the probability that an attack is successful where Kannan's embedding technique [23] is used for step 2. of the strategy, i.e. the reduction using the Log-unit lattice, and compute several parameters linked to the basis of the lattice to evaluate its quality. We also compare these values to the ones obtained for a range of cyclotomic fields.
- We provide in Appendix A some timings regarding the computation of their unit group with our implementation.

2. Background

In this section we will quickly present the essential background regarding lattices, number theory and cryptology necessary to understand this article. However some parts might be left out for clarity.

Notations. The inner product is denoted by $(\cdot \mid \cdot)$. When we consider a tuple $(\lambda_1, \ldots, \lambda_n)$ we can write it λ. Algebraic closures of number field extensions considered will be designated by Ω. We will write $\delta_{(\mathscr{P}(n))}$ for the indicator function of proposition $\mathscr{P}(\cdot)$, i.e. $\delta_{(\mathscr{P}(n))}$ is equal to 1 if $\mathscr{P}(n)$ is true and 0 otherwise. Finally, given two ordered sets A and B we will denote by $A \otimes B$ the tensor product of A and B (when it makes sense), i.e. $A \otimes B = \{ab, b \in B \mid a \in A\}$. If A and B are not ordered $A \otimes B$ will be the collection of all the products ab such that $a \in A$ and $b \in B$. Finally, we also use this notation for the tensor product of vectors and matrices.

2.1. **Lattices.**

We refer the reader to the part dedicated to lattices in the book of S. Galbraith [20]. One can find a fine exposé on lattices and their use in public-key cryptography. We refer the reader interested in more in-depth presentations on Euclidean lattices to [28, 15].

An *euclidean lattice* is a discrete subgroup of $\mathbb{R}^n$ where n is a positive integer. A *basis* of a lattice $\mathcal{L}$ is a basis of $\mathcal{L}$ when considered as a $\mathbb{Z}$-module. One way of representing a lattice is then to consider the matrix of a basis of the lattice. Let us denote by $\lambda_1(\mathcal{L})$ the norm of the shortest non zero vector of $\mathcal{L}$. There is an approximation of $\lambda_1(\mathcal{L})$ called the Gaussian heuristic which tells that the expected value of $\lambda_1(\mathcal{L})$ is in $O(\sqrt{\frac{r}{2\pi e}} \times \sqrt[r]{\det(\mathcal{L})})$. This gives an expected value for the norm of what we call a *short* vector. The classical problems over lattices are :

1. the *Shortest Vector Problem (SVP)* : «Given a a lattice $\mathcal{L}$ of dimension n, find $u \in \mathcal{L} \setminus \{0\}$ such that $\|u\| = \lambda_1(\mathcal{L})$ »;
2. the *Closest Vector Problem (CVP)* : «Given a lattice $\mathcal{L}$ of dimension n and $t \in \mathbb{R}^n$, find $u \in \mathcal{L}$ such that $\forall v \in \mathcal{L}, \|t - u\| \leqslant \|t - v\|$; »;
3. the *Bounded Distance Decoding (BDD)* : «Given a basis B of a lattice $\mathcal{L}$, a target vector t such that $d(t, \mathcal{L}) < \lambda_1(\mathcal{L})/2$, find the lattice vector $v \in \mathcal{L}$ closest to t. ».

In practice we can consider relaxed versions of these problems with respect to an approximation factor. For general lattices these problems are NP-hard thus at least as hard as factorising for example. Moreover we do not have any result showing that quantum computers can solve these problems for general lattices. These problems are easier to solve if we have a good basis at our disposal, i.e. a basis built with relatively short vectors which are nearly orthogonal to each other.

Despite the hardness of these problems over random lattices, high-dimensional lattices are large objects and slow to handle. A way of copping with that is to work with lattices with extra algebraic structure such as ideal lattices. However this can introduce a security weakness as it may be easier to find good basis related to such lattices or to use the algebraic structure to solve lattice problems.

2.2. **Number fields.**

We refer the reader to [14, 13, 31] for anything related to number fields and computational number theory.

A *number field* K is a field which is a finite extension of $\mathbb{Q}$. It can always be described as a polynomial quotient ring $\frac{\mathbb{Q}[X]}{(P(X))}$ where $P(X)$ is irreducible in $\mathbb{Q}[X]$. Equivalently if we choose θ to be any root of $P(X)$ we can see K as $\mathbb{Q}(\theta)$ the smallest field containing $\mathbb{Q}$ and θ. If we write n the degree of $P(X)$ then the dimension of K over $\mathbb{Q}$ – written $[K : \mathbb{Q}]$ – is n. We will consider two types of number fields in this paper. The fields of the first type are the most used in cryptography and well-studied in mathematics. They are called *cyclotomic fields*. They are generated by a root of unity ζ_m, with m being called the *conductor* of the field. The second type of number fields are Kummer extensions and are the subject of this paper. They are generated by p-th root of integers, and we will describe them more thoroughly in Section 3.

Complex embeddings and ring of integers. A number field K of dimension n over $\mathbb{Q}$ admits n distinct *complex field embeddings* $K \hookrightarrow \mathbb{C}$ usually denoted by $\sigma_1, \ldots, \sigma_n$. This set is denoted by $\mathrm{Hom}(K, \mathbb{C})$. There are r_1 real embeddings and r_2 pairs of complex embeddings. The two elements of a given pair are conjugates one from each other. It is the usage to write $\sigma_1, \ldots, \sigma_{r_1}$ the real embeddings and to consider that $\sigma_{j+r_2} = \overline{\sigma_j}$ for all $j \in [\![r_1 + 1, r_1 + r_2]\!]$.

Given a complex embedding $\sigma \in \mathrm{Hom}(K,\mathbb{C})$ the set $\{x \in K \mid \sigma(x) = x\}$ is a subfield of K. We will denote it by K^σ. One important map is the *Minkowski embedding* and is usually defined as

$$\begin{array}{rccl} \sigma_{\mathbf{K}}: & K & \longrightarrow & \mathbb{C}^{r_1+r_2} \\ & x & \longmapsto & (\sigma_i(x))_{i\in[\![1,r_1+r_2]\!]}. \end{array}$$

This allows to see K as embedded in $\mathbb{R}^n$. In this paper we relax the definition and consider σ_K to be defined as

$$\sigma_{\mathbf{K}} : x \longmapsto (\sigma_i(x))_{i\in[\![1,n]\!]},$$

where all complex embeddings are taken into account.

The *ring of integers of K denoted by $\mathcal{O}_K$* consists of the elements of K which are roots of a monic polynomial of $\mathbb{Z}[X]$. This ring as well as its ideals are full rank sub-$\mathbb{Z}$-module of K. In fact for a given ideal I one can find a basis $(b_1,\dots,b_n)$ of elements of $\mathcal{O}_K$ such that $K = \bigoplus_{i=1}^n \mathbb{Q}b_i$, $\mathcal{O}_K = \bigoplus_{i=1}^n \mathbb{Z}b_i$ and $I = \bigoplus_{i=1}^n \mathbb{Z}d_ib_i$ with $(d_1,\dots,d_n) \in \mathbb{Z}^n$. The images of $\mathcal{O}_K$ and of any ideal I of $\mathcal{O}_K$ under the action of any embedding of K into $\mathbb{R}^n$ are lattices. The usual embedding corresponds to view a number field K as a quotient $\frac{\mathbb{Q}[X]}{(f(X))}$. Then every element $g(X) = g_0 + \cdots + g_nX^n$ of K can be seen as the vector with coordinates $(g_0,\dots,g_n)$ in $\mathbb{R}^n$. The other fundamental example is the Minkowski embedding. Given a family $(x_1,\dots,x_n)$ of a number field K the *discriminant* $D(x_1,\dots,x_n)$ is the rational number $\det((\sigma_j(x_i))_{i,j})^2$. Given $\mathcal{O}$ an order of K the discriminant of $\mathcal{O}$ is $D(x_1,\dots,x_n)$ where $(x_1,\dots,x_n)$ is an integral basis of $\mathcal{O}$. The discriminant of K written D_K is the discriminant of its integer ring, and its absolute value can be seen as the squared volume of $\mathcal{O}_K$ when seen as a lattice through the action of the Minkowski embedding.

Embeddings in an algebraic closure and Galois group. As it is the case for complex embeddings, a number field K of dimension n over $\mathbb{Q}$ with algebraic closure Ω admits n distinct *field embeddings* $K \hookrightarrow \Omega$. This set is denoted by $\mathrm{Hom}(K,\Omega)$. Similarly, given an extension of number fields L/K we will denote by $\mathrm{Hom}(L/K,\Omega)$ the set of K-embeddings of L into Ω, i.e. elements of $\mathrm{Hom}(L,\Omega)$ congruent to Id_K when restricted to K.
The *Galois Group* of a field extension L/K denoted by $\mathrm{Gal}(L/K)$ is the group of field automorphisms of L which are congruent to the identity when restricted to K. It is a subset of $\mathrm{Hom}(L/K,\Omega)$. An extension L/K is called a *Galois extension* when the cardinality of $\mathrm{Gal}(L/K)$ equals the dimension $[L : K]$. Moreover we have the *Galois correspondence* which states that given a Galois extension K/L there is a one-to-one correspondence between the subgroups of $\mathrm{Gal}(L/K)$ and the subfields of K containing L. Given a subgroup H of $\mathrm{Gal}(L/K)$ we will write L^H the corresponding subfield of K. In the case of a number field K we say it is a *Galois field* if it is Galois as an extension of $\mathbb{Q}$. For example the cyclotomic fields are Galois number fields as well as the multiquadratic fields considered in [2]. However this property is not verified by a general number field K and we have to consider the Galois closure of K which is in fact the smallest extension containing all the roots of the irreducible polynomial $P(X)$. Given p a prime integer, we will denote by τ_p a generator of $\mathrm{Gal}(\mathbb{Q}(\zeta_p)/\mathbb{Q})$ the Galois group of the cyclotomic field generated by ζ_p. Therefore the action of τ_p can be described by $\tau_p(\zeta_p) = \zeta_p^{i_0}$ for a fixed $i_0 \in [\![1, p-1]\!]$.

It is usual for the two approaches of field embeddings described above – algebraic or complex – to be identified, as it the case in [13] for example. We will do the same, and the context will help determine which objects are considered. We might therefore talk about "complex embeddings" and use the notations $\mathrm{Hom}(L/K,\mathbb{C})$ and $\mathrm{Hom}(L,\mathbb{C})$, even when considering morphisms from a number field into an algebraic closure.

Unit group and Log embedding. The group of units of $\mathcal{O}_K$ written $\mathcal{O}_K^\times$ is the set $\left\{u \in \mathcal{O}_K \mid u^{-1} \in \mathcal{O}_K\right\}$. It has a specific structure that we can take advantage of. Given a number field K of degree n with $n = r_1 + 2r_2$ as before, we have

$$O_K^\times \cong \frac{\mathbb{Z}}{m\mathbb{Z}} \times \mathbb{Z}^{r_1+r_2-1}.$$

This isomorphism which allows to see the units of $\mathcal{O}_K^\times$ modulo its torsion group as a lattice is realised by an important embedding which is the *Log-embedding* of K. Consider $(c_j)_{j\in[\![1,r1+r2]\!]}$ such that $c_j = 1$ if $j \leqslant r_1$ and $c_j = 2$ otherwise. Then the Log-embedding is defined as

$$\begin{array}{rccl} \mathrm{Log}_K : & K^* & \longrightarrow & \mathbb{R}^{r_1+r_2} \\ & x & \longmapsto & (c_i \ln|\sigma_i(x)|)_{i\in[\![1,r_1+r_2]\!]} \,. \end{array}$$

The set $\mathrm{Log}_K(\mathcal{O}_K^\times)$ is a lattice of the hyperplane orthogonal to the all ones vector. It is called the *Log-unit lattice.* One can also define the Log-embedding by using all of the embeddings σ_i and forgetting the c_j:

$$\begin{array}{rccl} \mathrm{Log}_K : & K^* & \longrightarrow & \mathbb{R}^{n} \\ & x & \longmapsto & (\ln|\sigma_i(x)|)_{i\in[\![1,n]\!]} \,. \end{array} \tag{1}$$

By doing so the Log-unit lattice is a lattice of rank $r_1 + r_2 - 1$ in $\mathbb{R}^n$. The volume of $\mathrm{Log}_K(\mathcal{O}_K^\times)$ is $\sqrt{r_1 + r_2}R_K$ if we use the first definition and $\sqrt{\frac{n}{2^{r_2/2}}}R_K$ is we use the second, where R_K is the *regulator* of K. In the rest of the paper we will use the second form of the Log-embedding, and denote by V_K its volume.

Log-unit lattice and SPIP. For a general introduction on ideal lattices and their use in cryptography, one could refer to the survey of Ducas [18]. Recall that ideal based cryptosystems such as presented in [21, 22, 34] have in general a private key which is a short generator of a public ideal I. The security of such cryptosystems relies on the supposed hardness of finding such a generator given an ideal, problem called the *Short Principal Ideal Problem.* The *Principal Ideal Problem* consists in finding any generator of the principal ideal, i.e. given an ideal $I = g\mathcal{O}_K$, find some h such that $I = h\mathcal{O}_K$. As mentioned the process done to solve the SPIP relies essentially in two steps : solve the PIP and then shorten the retrieved generator. The set of generators of I is $\{gu \mid u \in \mathcal{O}_K^\times\}$. Therefore solving the PIP yields $h = gu$ with $u \in \mathcal{O}_K^\times$. It is then possible to retrieve g from h by finding u. This is where we can use the Log-unit lattice. If we transpose the situation with the Log-embedding, for every generator h we have $\mathrm{Log}_K(h) = \mathrm{Log}_K(g) + \mathrm{Log}_K(u)$. Using that remark and finding the element of the Log-unit lattice closest to h it is possible to retrieve g. This corresponds to solve the CVP with respect to the target h and the lattice $\mathrm{Log}_K(\mathcal{O}_K^\times)$, and even the BDD because we know the generator g is short. The success of such a method is therefore dependent on the length of $\mathrm{Log}_K(g)$ and the particular geometry of the Log-unit lattice meaning that we want to have access to a somehow good basis, i.e. orthogonal enough. This attack requires to

(1) solve the PIP : this is considered hard classically and can be done in quantum polynomial time;
(2) compute $\mathcal{O}_K^\times$: this is also considered hard classically and can be done in quantum polynomial time;
(3) shorten a generator h by solving the BDD with respect to $\mathrm{Log}_K(\mathcal{O}_K^\times)$: this will depend on the basis obtained.

One can remark that since the Log-unit lattice lies into H, the hyperplane orthogonal to $\mathbf{1} = (1, \dots, 1)$, the last step is to be carried over H. Thus the attack will require to retrieve $p_H(\mathrm{Log}_K(g))$ from $p_H(\mathrm{Log}_K(h))$. Because of this and to recall the cryptograhic setting, we will call $p_H(\mathrm{Log}_K(g))$ the *key (vector)* of the problem. As a matter of fact, step 3 will correspond to a BDD depending on the norm of the target. Solving a CVP or BDD can be done by several techniques. The main ones are Babaï's rounding technique and nearest-plane algorithm [1], as well as Kannan's embedding technique [23]. In [17, 2] the authors considered the rounding technique, and we follow [27] where the authors used an embedding technique to compute the output of the nearest plane algorithm.

Representation of elements. A fairly natural way of representing an element of K is by its coefficients in $\mathbb{Q}-$basis, typically $(\theta^i)_{i\in[\![1,n]\!]}$ or an integral basis of $\mathcal{O}_K$ if one is available. We will call this the *standard representation.* However during computations some objects can become very large and one can use the so-called *compact representation*, which expresses x as a product $x_0 x_1^{l} x_2^{l^2} \dots x_r^{l^r}$ with x_i having a bounded size.

Complexities. Because the discriminant of a number field measure the volume of its ring of integers, complexity of algorithms are often expressed in terms of $\log(|\Delta(K)|)$. The best algorithms to compute an integral basis of $\mathcal{O}_K$ are sub-exponential in general but one can find more efficient algorithms for specific number fields. This is the case for multiquadratic fields for instance as stated in [35]. The unit or the class groups can also be computed in sub-exponential time as in [7]. Finally in [7] the authors showed that the compact representation of an element can be computed polynomial time with respect to the size of the input. Complexities are often expressed by mean of the L-notation. Given a variable N and two constants α and c with $\alpha \in [0,1]$ and $c > 0$, $L_N(\alpha, c)$ is defined by

$$\exp\left((c + o(1))\ln(N)^{\alpha}(\ln\ln N)^{1-\alpha}\right).$$

3. Structure of Kummer extensions

Definition 1. A number field extension L/K is called a *Kummer extension of exponent* n if $\zeta_n \in K$ and there are elements $m_1, \ldots, m_r$ of K such that $L = K(\sqrt[n]{m_1}, \ldots, \sqrt[n]{m_r})$.

Remark 1. In our work we relax this definition to allow ζ_n to not belong to L. We will also only consider extensions of prime exponents p. First let us recall some facts and fix some notations about the structure of Kummer extensions, and $\mathrm{Hom}(L/K, \mathbb{C})$. We refer the reader interested in a more general and in-depth presentation of Kummer extensions to [13].

3.1. **Field embeddings and Galois closure.**

3.1.1. *Simple extensions.*

Definition 2. Consider L/K an extension of number fields, and prime number p. Then L/K is called a *simple Kummer extension of exponent* p if there is $m \in K$ such that $\sqrt[p]{m} \notin K$ and $L = K(\sqrt[p]{m})$.

Proposition 1. *Consider $L = K(\sqrt[p]{m})$ a simple Kummer extension. Then the following properties are true.*

(1) L/K is a field extension of degree p.

(2) The elements of the set $\mathrm{Hom}(L/K, \Omega)$ can be fully described by their action on $\sqrt[p]{m}$ as $\sigma^{(i)} : \sqrt[p]{m} \longmapsto \zeta_p^i \sqrt[p]{m}, i \in [\![0, p-1]\!]$.

(3) If $\zeta_p \in L$ then L/K is Galois. If $\zeta_p \notin K$ then the Galois closure of L/K is $\widetilde{L} = L(\zeta_p)$ and if p is odd then $\mathrm{Gal}(\widetilde{L}/K) = \langle \tau_p \rangle \ltimes \langle \sigma \rangle$ where σ is the extension of the complex embedding $\sigma^{(1)}$ which acts trivially on ζ_p. If p is 2 then L is Galois.

Proposition 2. *Let $L = K(\sqrt[p]{m})$ be a simple Kummer extension of exponent p, and $n \in K$. Then $L = K(\sqrt[p]{n})$ if, and only if, there is $a \in K$ such that $n = ma^p$.*

3.1.2. *General extensions.* The properties described for simple Kummer extensions can be extended to general extensions.

Proposition 3. *Consider $L = K(\sqrt[p]{m_1}, \ldots, \sqrt[p]{m_r})$ be a Kummer extension. Then the following assertions are equivalent.*

(1) $[L : K] = p^r$;

(2) $(\forall \alpha \in \mathbb{Z}^r)$, $m_1^{\alpha_1} m_2^{\alpha_2} \cdots m_r^{\alpha_r} \in (K^*)^p \iff \forall i \in [\![1, r]\!], p \mid \alpha_i$.

Definition 3. Given a prime p, an integer $r \in \mathbb{N}^*$ and a sequence m of rational numbers $m_1, \ldots, m_r$ we will say that m is *p-reduced for K* if it verifies the condition of Proposition 3.

Proposition 4. *Consider p a prime number and $L = K(\sqrt[p]{m_1}, \ldots, \sqrt[p]{m_r})$ a Kummer extension of exponent p. Then L can be described as $K(\sqrt[p]{n_1}, \ldots, \sqrt[p]{n_s})$ with $n = (n_1, \ldots, n_s)$ being a p-reduced sequence.*

From now on, all Kummer extensions are considered to be generated by reduced sequences.

Notation. Consider $m = (m_1, \ldots, m_r) \in K^r$ such that $L = K(\sqrt[p]{m_1}, \ldots, \sqrt[p]{m_r})$ is an extension of degree p^r. For $i \in \llbracket 1, r \rrbracket$ the field $L_{m_i} = K(\sqrt[p]{m_i})$ is a simple Kummer extension of K of exponent p. Given any $j \in \llbracket 0, p-1 \rrbracket$, write $\sigma_{m_i}^{(j)}$ the field embeddings of $L_{m_i} \hookrightarrow \Omega$ following the notation described previously and $\sigma_{m_i}^{j}$ the corresponding element of $\mathrm{Gal}(\widetilde{L_{m_i}}/K)$.

The simple extensions of a Kummer extension L/K are important as they allow to fully describe L/K, as we will see later.

Proposition 5. *Consider L/K which verifies the equivalent assertions of Proposition 3. Then the following assertions are true.*

(1) L/K has exactly $\frac{p^r-1}{2}$ simple subextensions of degree p over K and they are of the form $L_\alpha := L(\prod_{i=1}^{r} \sqrt[p]{m_i}^{\alpha_i})$ with $\alpha \in \llbracket 0, p-1 \rrbracket^r$. Moreover L_α and L_β are equal if, and only if, there is an integer λ such that $\alpha = \lambda \cdot \beta \pmod{p}$.

(2) Any subextension of L/K can be written as $K(\sqrt[p]{M_1}, \ldots, \sqrt[p]{M_{r'}})$ where $0 \leqslant r' \leqslant r$ and $M_j = \prod_{i=1}^{r} \sqrt[p]{m_i}^{\alpha_i^{(j)}}$ with $\alpha^{(j)} \in \llbracket 0, p-1 \rrbracket^r$ for any $j \in \llbracket 1, r' \rrbracket$.

The set complex embeddings of L and the Galois group of $\widetilde{L}/\mathbb{Q}$ can also be fully described with the ones of the subfields L_{m_i}.

Proposition 6. *Consider L/K which verifies the equivalent assertions of Proposition 3. Then the following assertions are true.*

- $\mathrm{Hom}(L/K, \Omega) \cong \bigotimes_{i=1}^{r} \mathrm{Hom}(L_{m_i}/K, \Omega) = \{\otimes_{i=1}^{r} \sigma_{m_i}^{(\beta_i)} \mid \beta \in \llbracket 0, p-1 \rrbracket^r\}$.
- *$L(\zeta_p)/K(\zeta_p)$ is abelian with Galois group isomorphic to $\langle \sigma_{m_1} \rangle \times \cdots \times \langle \sigma_{m_r} \rangle$; if $\zeta_p \in K$ then the previous extension is L/K.*
- *If $\zeta_p \notin K$ then $L(\zeta_p)/K$ is Galois with Galois group isomorphic to $\langle \tau_p \rangle \ltimes \langle \sigma_{m_1} \rangle \times \cdots \times \langle \sigma_{m_r} \rangle$.*

Notation. Given a tuple β we will write $\sigma^{(\beta)}$ the complex embedding $\otimes_{i=1}^{r} \sigma_{m_i}^{(\beta_i)}$ and σ^{β} its extension in $\mathrm{Gal}(\widetilde{K}/\mathbb{Q})$. Given a subset S of $\mathrm{Hom}(K, \Omega)$ we will denote by $\widetilde{S}$ the subset of $\mathrm{Gal}(\widetilde{K}/\mathbb{Q})$ whose elements are the direct extension of elements of S.

3.2. Structural result.

The main brick of the efficient algorithms in [2, 8, 27] are structural results which allow to express a power of any field element as a product of relative norms over several subfields. As the fields studied here are the generalisation of multiquadratic and multicubic fields, the same structural result appears.

Notation. Given an integer k and a subset S of a field F we will denote by S^k the set $\{x^k \mid x \in S\}$.

Proposition 7. *Let p be an odd prime number. Consider $L = K(\sqrt[p]{m_1}, \sqrt[p]{m_2})$ a Kummer extension such that $[L : K] = p^2$. Let u and v be two elements of $\mathrm{Hom}(L/K, \Omega)$ such that their extensions $\tilde{u}$ and $\tilde{v}$ are independent. Then the following properties are true.*

(1) $L^p \subset L^u L^{uv} \ldots L^{uv^{p-1}} L^v$;

(2) $(\mathcal{O}_L^\times)^p \subset \mathcal{O}_{L^u}^\times \mathcal{O}_{L^{uv}}^\times \ldots \mathcal{O}_{L^{uv^{p-1}}}^\times \mathcal{O}_{L^v}^\times$.

Proof. The proof is similar to the ones of the corresponding results in [2]. Let $x \in L^*$ and u, v be two elements of $\mathrm{Hom}(L/K, \Omega)$ such that $\tilde{u}$ and $\tilde{v}$ are independent. Then we have:

$$x^p = \frac{\prod_{i=0}^{p-1} \prod_{j=0}^{p-1} (\tilde{u}\tilde{v}^i)^j(x)}{\prod_{i=0}^{p-1} \prod_{j=1}^{p-1} (\tilde{u}\tilde{v}^i)^j(x)} = \frac{\prod_{i=0}^{p-1} \mathrm{N}_{\tilde{L}/\tilde{L}^{\tilde{u}\tilde{v}^i}}(x)}{\prod_{j=1}^{p-1} \tilde{u}^j \left(\prod_{i=0}^{p-1} \tilde{v}^{ij}(x)\right)}. \tag{2}$$

For any $j \in \llbracket 1, p-1 \rrbracket$ the sets $\{i \mid i \in \llbracket 0, p-1 \rrbracket\}$ and $\{ij \mid i \in \llbracket 0, p-1 \rrbracket\}$ are the same, therefore:

$$x^p = \frac{\prod_{i=0}^{p-1} \mathrm{N}_{\tilde{L}/\tilde{L}^{\tilde{u}\tilde{v}^i}}(x)}{\prod_{j=1}^{p-1} \tilde{u}^j \left(\mathrm{N}_{\tilde{K}/\tilde{L}^{\tilde{v}}}(x)\right)} = \frac{\prod_{i=0}^{p-1} \mathrm{N}_{\tilde{L}/\tilde{L}^{\tilde{u}\tilde{v}^i}}(x)}{\mathrm{N}_{\tilde{L}/\tilde{L}^{\tilde{v}}} \left(\prod_{j=1}^{p-1} \tilde{u}^j(x)\right)}. \tag{3}$$

Now let us assume first that $\zeta_p \in K$. Then L/K is Galois, $u = \tilde{u}$, $v = \tilde{v}$ and Equation (3) can be written as

$$x^p = \frac{\prod_{i=0}^{p-1} \mathrm{N}_{L/L^{uv^i}}(x)}{\mathrm{N}_{L/L^v}\left(\prod_{j=1}^{p-1} u^j(x)\right)}.$$

For any morphism w the relative norm $\mathrm{N}_{L/L^w}(x)$ is an element of L^w and if x is an integer (resp. a unit) then its relative norms are also integers (resp. units). Therefore on has

$$x^p \in L^u L^{uv} \dots L^{uv^{p-1}} L^v \tag{4}$$

and if $x \in \mathcal{O}_L^\times$ we can replace the fields by their unit groups. Finally 4 is true for any x different from 0, but it is obviously correct for 0 as well, which proves that the claimed results are true if $\zeta_p \in K$. Now assume that $\zeta_p \notin K$. Then for all $i, j \in [\![0, p-1]\!]$ the action of $(\tilde{u}^i \tilde{v}^j)$ on x is the same as the action of $u^{(i)} v^{(j)}$. Therefore for all $i \in [\![0, p-1]\!]$ the relative norm $\mathrm{N}_{\tilde{L}/\tilde{L}^{\tilde{u}\tilde{v}^i}}(x)$ is equal to $\mathrm{N}_{L/L^{uv^{(i)}}}(x)$ which is an element of $K^{uv^{(i)}}$. The statements about integers and units are again true. In Equation (3) we know that x^p belongs to L as well as the numerator, so the denominator belongs to $\tilde{L}^{\tilde{v}} \cap L = L^v$. Finally the claimed results are also true if $\zeta_p \notin K$. □

If one remove zero from all of the sets, then the set inclusions in Proposition 7 become group inclusions. In fact remark that $U = \mathcal{O}_{L^u}^\times \mathcal{O}_{L^{uv}}^\times \dots \mathcal{O}_{L^{u^{p-1}v}}^\times \mathcal{O}_{L^v}^\times$ is a full-rank subgroup of $O_L^\times$ such that $(O_L^\times)^p < U < O_L^\times$.

Corollary 1. *Let p be an odd prime number. Consider $L = K(\sqrt[p]{m_1}, \dots, \sqrt[p]{m_r})$ a Kummer extension such that $[L : K] = p^r$. Then the following holds:*

(1) $L^{p^{r-1}} \subset \prod_\alpha L_\alpha$;
(2) $(\mathcal{O}_L^\times)^{p^{r-1}} < \prod_\alpha \mathcal{O}_{L_\alpha}^\times$.

Definition 4. Given a Kummer extension $L = K(\sqrt[p]{m_1}, \dots, \sqrt[p]{m_r})$ we will call *simple units of L/K* and denote by $\mathrm{SU}(L/K)$ the subgroup of $\mathcal{O}_L^\times$ defined by the following equation.

$$\mathrm{SU}(L/K) = \prod_\alpha \mathcal{O}_{L_\alpha}^\times$$

3.3. General algorithms.

The general procedures follow the same shape than the ones in [2, 27]. The algorithms rely on two tasks:

(1) detecting non trivial p-powers in the subgroup of L^* generated by a given finitely generated subgroup $S = \langle s_1, \dots, s_n \rangle < L^*$;
(2) computing the roots of the detected powers.

We will write `DetectPowers` the first procedure and `ElementsFromPower` the second. The procedure which finds a basis of a subgroup of $\mathcal{O}_L^\times$ given a generating family by reducing through the Log-embedding will be denoted by `BasisFromGeneratingSet`.

3.3.1. *Detecting powers.* One can use the *Saturation technique* mentioned in [5, 8]. For any prime ideal $\mathfrak{Q}$ such that $p \mid \mathrm{N}(\mathfrak{Q}) - 1$ one can construct a "character" $\chi_{\mathfrak{Q}} : S \to F_{\mathfrak{Q}}^* / (F_{\mathfrak{Q}}^*)^p$ where $F_{\mathfrak{Q}}$ is the residue class field. If $u \in S$ is a $p-$power then $\chi_{\mathfrak{Q}}(u)$ is trivial but the inverse is not true in general. In order to detect proper powers, one only has to intersect $\ker \chi_{\mathfrak{Q}}$ for sufficiently many $\mathfrak{Q}$. If n is the cardinal of a minimal generating family of S, then the rank $S/(S \cap (L^*)^p)$ is $n' \leqslant n$. If we consider the $\chi_{\mathfrak{Q}}$ to be uniformly distributed in the dual then one can adapt Lemma 8.2 of [11] to show that $n' + s$ characters generate the dual – so the intersection of their kernels is $S \cap (L^*)^p$ – with probability at least $1 - p^{-s}$. If B is a bound on the size of the basis elements generating S then `DetectPowers` can be computed in $\mathrm{Poly}(B, \max_{\mathfrak{Q}} \ln(\mathrm{N}(\mathfrak{Q})), n' + s)$. For the case of multiquadratic fields, the authors of [2] give a practical way of computing these characters and a precise analysis of the cost of the overall procedure, that we refer to. It can be generalised to the real Kummer extensions that we will study below.

3.3.2. *Computing units.* Algorithm 1 describes the recursive algorithm which can be used to compute the unit group of a Kummer extension L/K. It is the generalisation of the ones for multiquadratic fields or multicubic fields presented in [2, 27]. We denote by `UnitGroup` the general procedure computing the unit group of a number field as input. Depending on the number field, different algorithms can be used.

Algorithm 1 Compute the unit group of a Kummer extension L/K of exponents p. – `KE_Units`

```
Require: A Kummer extension $L = K(\sqrt[p]{m_1}, \dots, \sqrt[p]{m_r})$.
Ensure: A basis of the unit group $\mathcal{O}_L^\times$
 1: if $(r = 1)$ then
 2:     return UnitGroup($L$).
 3: else
 4:     Choose $u, v$ two independent elements of $\mathrm{Hom}\widetilde{(L/K}, \Omega)$.
 5:     Recursively compute a basis of $U = \mathcal{O}^\times_{L^u}\mathcal{O}^\times_{L^{uv}} \dots \mathcal{O}^\times_{L^{uv^{p-1}}}\mathcal{O}^\times_{L^v}$
 6:     $V \leftarrow$ DetectPowers($U, p$)
 7:     $V \leftarrow$ ElementsFromPower($V, p$)
 8:     $U \leftarrow$ BasisFromGeneratingSet($\langle U, V\rangle$)
 9:     return $U$
10: end if
```

Proposition 8. *Given a Kummer extension L/K, Algorithm 1 is correct, and returns a basis with probability at least $1 - p^{-[L:\mathbb{Q}]}$ provided that one computes $\mathrm{Poly}([L:\mathbb{Q}])$ characters for each subfield L' reached during the algorithm, and that the characters are uniformly distributed.*

Proof. By Proposition 7 the subgroup U of step 5 is such that $(\mathcal{O}_L^\times)^p < U < \mathcal{O}_L^\times$. Therefore $\mathcal{O}_L^\times$ is isomorphic to $U \times \frac{U \cap (\mathcal{O}_L^\times)^p}{U^p}$. The only part left to verify is the the validity of the recursion. Clearly each of the fields L_i is a Kummer extension of K but such that $[L_i : K] = p^{r-1}$ so the algorithm can be applied to it. Since the dimension is strictly decreasing, after $r - 1$ recursion steps the algorithm reaches simple extensions of L, i.e. the case $r = 1$. Then following the analysis done during the proof of Theorem 4.6 in [8], the probability of success is at least $(1 - p^{-(s)})^{[L:K]} \geqslant 1 - 2^{[L:K]}/p^s$ where s characters are computed for each field. Therefore if $s \in \mathrm{Poly}([L:\mathbb{Q}])$ one can reach the desired probability of success. □

3.3.3. *Solving the Principal Ideal Problem.* In order to solve the Principal Ideal Problem, i.e. retrieve a generator of a principal ideal I, we do as follow. First compute the relative ideal norm of I over subfields of K. Then recursively compute a generator of these ideals. By using Proposition 7 it is easy to see that a combination of these elements is a generator h of I^p (see [2, 27]). The final steps are finding a unit u such that hu is a p-power and compute its p-th root. This is summarised in Algorithm 2. The relative norm computations are polynomial with respect to the dimension and the size of the ideal. Moreover Algorithms 1 and 2 are very similar in shape. One can easily deduce that the validity and complexity analysis are also similar. One can see that Algorithm 2 will go through subextensions of L/K down to simple subextensions, where it will call the procedure `Generator` to solve the PIP. As for `UnitGroup`, different algorithms depending on the field can be used. In the general case one might need to compute the class group of the field so one cannot hope better than a sub-exponential complexity.

4. Real Kummer extensions

In this section we will focus on real Kummer extensions. More precisely we are interested in fields of the form $L = K(\sqrt[p]{m_1}, \dots, \sqrt[p]{m_r})$ with $K = \mathbb{Q}(\sqrt[q]{n_1}, \dots, \sqrt[q]{n_s})$ with p and q prime integers. We always consider $\sqrt[p]{m_i}$ and $\sqrt[q]{n_j}$ to be the real roots of the polynomials $X^p - m_i$ and $X^q - n_j$ respectively. Then K is a real Kummer extension of $\mathbb{Q}$ of exponent q and L is a real Kummer extension of K of exponent p. We will call such fields *real Kummer*

Algorithm 2 Solve the PIP in a Kummer extension of exponent p – `KE_PIP`

```
Require: A principal ideal $I$ of a Kummer extension $L = K(\sqrt[p]{m_1}, \dots, \sqrt[p]{m_r})$, the unit group $\mathcal{O}_L^\times$
Ensure: A generator $g$ of $I$.
if $(r = 1)$ then
    return Generator(I).
else
    Choose $u, v$ two independent elements of $\widetilde{\mathrm{Hom}(L/K, \Omega)}$.
    Recursively compute generators of $\mathrm{N}_{L^u}(I), \mathrm{N}_{L^{uv}}(I), \dots, \mathrm{N}_{L^{u^{p-1}v}}(I), \mathrm{N}_{L^v}(I)$ and use Equation 3 to have $h$ a generator of $I^p$.
    $h \leftarrow$ DetectPowers($\mathcal{O}_L^\times \cup \{h\}, p$).
    return ElementsFromPower($h, p$).
end if
```

extension of exponents p, q. For the particular case of $s = 0$ the field K is $\mathbb{Q}$. Multiquadratic and multicubic fields studied in [2, 27] fall in this category. We will call these fields *real Kummer extensions with exponent* p.

4.1. **Field structure.** First let us describe the structure of considered extensions.

Proposition 9. *Consider a Kummer extension* L/K *as before. Then the following assertions are true :*

(1) $\mathrm{Hom}(L, \Omega) \cong \bigotimes_{i=1}^r \mathrm{Hom}(L_{m_i}, \Omega) \bigotimes_{j=1}^s \mathrm{Hom}(K_{n_j}, \Omega) = \{\otimes_{i=1}^r \sigma_{m_i}^{(\beta_i)} \otimes_{j=1}^s \sigma_{n_j}^{(\gamma_j)} \mid \beta \in [\![0, p-1]\!]^r, \gamma \in [\![0, q-1]\!]^s\}$.
(2) $L(\zeta_p)/K(\zeta_p)$ *is abelian with Galois group isomorphic to* $\langle \sigma_{m_1} \rangle \times \dots \times \langle \sigma_{m_r} \rangle$
(3) $L(\zeta_p)/\mathbb{Q}(\zeta_p)$ *is abelian with Galois group isomorphic to* $\langle \sigma_{m_1} \rangle \times \dots \times \langle \sigma_{m_r} \rangle \times \langle \sigma_{n_1} \rangle \times \dots \times \langle \sigma_{n_s} \rangle$.
(4) $L(\zeta_p)/K$ *is Galois with Galois group isomorphic to a subgroup of* $\langle \tau_p \rangle \ltimes \langle \sigma_{m_1} \rangle \times \dots \times \langle \sigma_{m_r} \rangle$.

Notation. Given tuples β and γ we will denote by $\sigma^{(\gamma, \beta)}$ the morphism $\otimes_{i=1}^r \sigma_{m_i}^{(\beta_i)} \otimes_{j=1}^s \sigma_{n_j}^{(\gamma_j)}$ its extension in $\mathrm{Gal}(\widetilde{L}/\mathbb{Q})$. Given a subset S of $\mathrm{Hom}(L, \Omega)$ we will write $\widetilde{S}$ for the subset of $\mathrm{Gal}(\widetilde{L}/\mathbb{Q})$ whose elements are the direct extension of elements of S.

4.2. **Basis and discriminant.** We will state some facts about $\mathbb{Q}$bases of real Kummer extensions considered, and about their discriminants. These results and their proofs appear in a note from the first author in [25].

Knowing the discriminant of a number field is important as it is a measure of the size of the ring of integers, and one usually express complexities of algorithms in term of the discriminant. It can be difficult to find a formula for it. However it can be done over multiquadratic fields and multicubic fields [32, 27]. Moreover we wish to exhibit a simple $\mathbb{Q}$-basis of real Kummer extensions L and $d_L \in L$ such that $d_L \mathcal{O}_L$ is included in the order generated by this basis. Knowing such basis and a coefficient d_L allows us to represent an element x of $\mathcal{O}_L$ in this basis and guarantee that we can reduce computations on x to computations to an element with integral coefficients, namely $d_L x$. This is notably important for root extraction, since the method we use in our implementation handle only integral coefficients.

Regarding the discriminant and other mathematical objects linked to the problems studied in what follows, we refer the reader to [14, 13, 31].

4.2.1. *Extensions with one exponent.* First we will study fields of the form $K = \mathbb{Q}(\sqrt[p]{m_1}, \dots, \sqrt[p]{m_r})$.

Notations. Given a tuple $m = (m_1, \dots, m_r)$, we will write $\mathcal{P}(m)$ the set $\{p \in \mathcal{P}, p \mid \prod_{i=1}^r m_i\}$. For $m \in \mathbb{Q}$ and $n \in \mathbb{N}$ we will denote by $PF(m, n)$ the rational number $\prod_{p \in \mathcal{P}(m)} m^{v_p(m) \pmod n}$. Similarly if $m \in \mathbb{Q}^r$ then $PF(m, n) = (PF(m_1, n), \dots, PF(m_r, n))$. Finally we extend $PF(\cdot, p)$ to elements in $\mathbb{Q}^{1/p}$ and sequences in $\mathbb{Q}^{1/p}$ with $PF(x, p) = PF(x^p, p)^{1/p}$.

A canonical $\mathbb{Q}-$*basis of* K. One can define two fairly natural bases of K. One has already be mentioned earlier.

Definition 5. Let $K = \mathbb{Q}(\sqrt[p]{m_1}, \dots, \sqrt[p]{m_r})$ be a real Kummer field. Then the *naive basis of* K *relative to* m is $(\prod_{i=1}^{r} m_i^{\alpha_i/p})_{\alpha \in [\![0,p-1]\!]^r}$. It will be denoted by $\mathfrak{B}(p, m)$. The *power-free basis of* K *relative to* m is $PF(\mathfrak{B}(p, m), p)$. It will be denoted by $\mathfrak{PB}(p, m)$.

Remark 2. Both bases were considered in several work on Kummer fields such as [8, 36].

The first property that can be proven is that $\mathfrak{PB}(p, m)$ is somehow independent on the choice of m.

Lemma 1. *Let* K *be a real Kummer field. Consider* m *and* n *be two sequences defining* K*. Then* $\mathfrak{PB}(p, m)$ *and* $\mathfrak{PB}(p, n)$ *are equal as sets.*

The equality given by Lemma 1 shows that the set of power-free basis of a real Kummer field is a canonical choice of a $\mathbb{Q}$-basis of K.

Definition 6. Let K be a real Kummer field with one exponent p defined by a sequence m. *The power-free basis* of K is the unordered sequence set $\mathfrak{PB}(p, m)$. It will be denoted $\mathfrak{PB}(K)$.

Theorem 1. *Let* $K = \mathbb{Q}(\sqrt[p]{m_1}, \dots, \sqrt[p]{m_r})$ *be a real Kummer extension, and denote by* $\mathcal{O}$ *the order* $\mathbb{Z}[\mathfrak{PB}(K)]$*. Then the following propositions are true.*

- $\forall q \in \mathcal{P}(m) \setminus \{p\}$, $\mathcal{O}$ *is* $q-$*maximal.*
- $[K : \mathbb{Q}]\mathcal{O}_K < \mathcal{O}$.

Theorem 1 is proven by studying the discriminant of both $\mathcal{O}_L$ and $\mathcal{O}$. In particular, the following result concerning primes ramifying in L different from the exponent p is achieved.

Proposition 10. *Consider* $K = \mathbb{Q}(\sqrt[p]{m_1}, \dots, \sqrt[p]{m_r})$ *a real Kummer extension with one exponent, and* $q \in \mathcal{P}(m) \setminus \{p\}$*. Then* $v_q(D_K) = (p-1)p^{r-1}$.

4.2.2. *Extensions with two exponents.* For general extensions with two exponents, the "denominator" d_L is not proven to be $[L : \mathbb{Q}]$, but is expressed is terms of $\mathcal{P}(m) \cap \mathcal{P}(n)$.

Definition 7. Let L/K be a real Kummer extension with two exponents p, q. We will call *power-free basis of* L/K and denote by $\mathfrak{PB}(L/K)$ the basis $\mathfrak{PB}(M) \otimes \mathfrak{PB}(K)$.

Theorem 2. *Let* $L = K(\sqrt[p]{m_1}, \dots, \sqrt[p]{m_r})$ *with* $K = \mathbb{Q}(\sqrt[q]{n_1}, \dots, \sqrt[q]{n_s})$ *be a real Kummer extension with two exponents. Denote by* $\mathcal{O}$ *the order* $\mathbb{Z}[\mathfrak{PB}(L)]$*, and* $A = (\mathcal{P}(m) \cap \mathcal{P}(n)) \setminus \{p, q\}$ *and* $P_A = \prod_{a \in A} a$*. Then the following properties are true.*

- $\forall a \in \mathcal{P}(m) \cup \mathcal{P}(n) \setminus (A \cup \{p, q\})$, $\mathcal{O}$ *is* $a-$*maximal.*
- $P_A[L : \mathbb{Q}]\mathcal{O}_L < \mathcal{O}$.

Again, one can express how the discriminant of L depends on the primes dividing the defining coefficients of L.

Proposition 11. *Let* $L = K(\sqrt[p]{m_1}, \dots, \sqrt[p]{m_r})$ *with* $K = \mathbb{Q}(\sqrt[q]{n_1}, \dots, \sqrt[q]{n_s})$ *be a real Kummer extension with two exponents. Let* $a \in \mathcal{P}(m) \cup \mathcal{P}(n) \setminus \{p, q\}$*. Then the splitting of* a *in* $L/\mathbb{Q}$ *and* $v_a(D_L)$ *verify the following.*

(1) $a \in \mathcal{P}(m) \setminus \mathcal{P}(n) \implies v_a(D_L) = [L : \mathbb{Q}]\frac{p-1}{p}$.
(2) $a \in \mathcal{P}(n) \setminus \mathcal{P}(m) \implies v_a(D_L) = [L : \mathbb{Q}]\frac{q-1}{q}$.
(3) $a \in \mathcal{P}(m) \cap \mathcal{P}(n) \implies v_a(D_L) = [L : \mathbb{Q}]\frac{pq-1}{pq}$.

4.3. Geometry under Log_L.

Lemma 2. *Consider K_1 and K_2 two number fields, and $K = K_1K_2$ their compositum. Assume that* $\mathrm{Hom}(K, \mathbb{C}) \cong \mathrm{Hom}(K_1, \mathbb{C}) \otimes \mathrm{Hom}(K_2, \mathbb{C})$. *Then one has the following.*

$$\forall (x_1, x_2) \in K_1 \times K_2, (\mathrm{Log}_K(x_1) \mid \mathrm{Log}_K(x_2)) = \ln|\mathrm{N}_{K_1/\mathbb{Q}}(x_1)| \ln|\mathrm{N}_{K_2/\mathbb{Q}}(x_2)|.$$

In particular $\mathrm{Log}_K(\mathcal{O}_{K_1}^\times)$ *is orthogonal to* $\mathrm{Log}_K(x_2)$ *for any* $x_2 \in K_2$.

Proof. Let us denote by H, H_1 and H_2 the sets $\mathrm{Hom}(K, \mathbb{C})$, $\mathrm{Hom}(K_1, \mathbb{C})$ and $\mathrm{Hom}(K_2, \mathbb{C})$ respectively. Moreover we will write S for $(\mathrm{Log}_K(x_1) \mid \mathrm{Log}_K(x_2))$. Then we have

$$S = \sum_{\sigma \in H} \ln|\sigma(x_1)| \ln|\sigma(x_2)| = \sum_{\sigma_1 \in H_1} \sum_{\sigma_2 \in H_2} \ln|\sigma_1 \otimes \sigma_2(x_1)| \ln|\sigma_1 \otimes \sigma_2(x_2)|.$$

Then for $i \in \{1, 2\}$ we get $\sigma_1 \otimes \sigma_2(x_i) = \sigma_i(x_i)$. Thus we obtain

$$S = \sum_{\sigma_1 \in H_1} \sum_{\sigma_2 \in H_2} \ln|\sigma_1(x_1)| \ln|\sigma_2(x_2)| = \sum_{\sigma_1 \in H_1} \ln|\sigma_1(x_1)| \sum_{\sigma_2 \in H_2} \ln|\sigma_2(x_2)|$$

which gives the first result. The statement about the orthogonality of the units follows from the fact that their algebraic norm is ±1. □

Corollary 2. *Let $K = \mathbb{Q}(\sqrt[p]{m_1}, \ldots, \sqrt[p]{m_r})$ be a real Kummer field with one exponent. Then we have the following.*

$$\mathrm{Log}_K(\mathrm{SU}(K)) = \overset{\perp}{\bigoplus_{\alpha \in \frac{\mathbb{F}_p^r \setminus \{0\}}{\sim}}} \mathrm{Log}_K(\mathcal{O}_{K_\alpha}^\times) \tag{5}$$

Proof. Just remark that for any pair $(\alpha, \beta) \in \mathbb{F}_p^r \setminus \{0\}$ such that $\alpha \not\sim \beta$ we can apply Lemma 2 to K_α and K_β. □

We know that $\mathrm{SU}(K)$ is a full-rank subgroup of $\mathcal{O}_K^\times$ following Corollary 1, and equivalently $\mathrm{Log}_K(\mathrm{SU}(K))$ is a full-rank sublattice of $\mathrm{Log}_K(\mathcal{O}_K^\times)$. In the case of multiquadratic and multicubic fields, one can see from Corollary 2 that each set of fundamental units $\{\epsilon_\alpha \mid \alpha \in \frac{\mathbb{F}_p^r \setminus \{0\}}{\sim}\}$ is sent by Log_K to an orthogonal basis of this sublattice. This is the best situation possible when it comes to solving lattices problems. In particular one could hope to decode respectively to $\mathrm{Log}_K(\mathrm{SU}(K))$, and use enumerations like over cyclotomic fields in [17]. However as mentioned in [2] the index $[\mathcal{O}_K^\times : \mathrm{SU}(K)]$ is too large for this strategy to be efficient. On the other hand, Algorithm 1 shows that one can obtain $\mathrm{Log}_K(\mathcal{O}_K^\times)$ from $\mathrm{Log}_K(\mathrm{SU}(K))$ by doing simple operations on vectors: additions and division by a scalar (2 or 3 depending on the case).

For Kummer extensions with one exponent $p > 3$, we obtain blocks of size $\frac{p-1}{2}$ orthogonal one to each other, i.e. if we consider a matrix basis M of $\mathrm{Log}_K(\mathrm{SU}(K))$ then its Gram matrix $MM^\intercal$ is a block diagonal matrix

$$\begin{bmatrix} G_\alpha & 0 & \ldots & 0 \\ 0 & G_\beta & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \ldots & 0 & G_\gamma \end{bmatrix}$$

with the diagonal blocks being of the form $M_\alpha M_\alpha^\intercal$, with $M_\alpha = \mathrm{Log}_K(\mathcal{O}_{K_\alpha}^\times)$. The basis from which we construct the unit group is therefore not orthogonal anymore. One can wonder whether it has an impact on the quality of the basis obtained for $\mathrm{Log}_K(\mathcal{O}_K^\times)$ and on performance of the SPIP procedure.

For Kummer extensions with two exponents, we cannot apply Lemma 2 to the minimal subfields reached by the recursion of the version of Algorithm 1 adapted to these type of extensions, i.e. Algorithm 6. Indeed, we will see that they are of the form $\mathbb{Q}(\sqrt[p]{M_\alpha}\sqrt[q]{N_\beta})$, which do not satisfy the required properties of Lemma 2. The reunion of their unit groups will still generate a full-rank sublattice, but not as a direct sum anymore. Thus we obtain a situation more entangled than with real Kummer extensions with one exponent. Again one may ask how it impacts the possibility of recovering a short generator through the Log-unit lattice.

4.4. **Auxiliary algorithms.** First we will describe the procedures used in Algorithm 1 when applied to real Kummer extensions, as well as how we compute the final reduction step to solve the SPIP. Most of the procedures are very similar to the ones in [2, 27]. We present them for completness purposes. In the following we will denote by N the absolute dimension of L. As in [2, 27] we will always assume that an element x is represented together with an approximation of $\mathrm{Log}_L(x)$, that we will denote by $\mathrm{ApproxLog}_L(x)$. Moreover, we used the *power-free basis* defined and studied in Subsection 4.2 to represent elements x. This way we know that there is a coefficient d_L such that the coefficients of $d_L x$ are integers.

4.4.1. *Finding Good Primes.* As in [2] we will need to be able to find primes verifying fixed conditions with respect to the m_i's.

Definition 8. Consider $m = (m_1, \dots, m_r)$, $C = (c_1, \dots, c_r) \in \{0,1\}^n$ and a prime number p. A *good prime relatively to* (m, C, p) is a prime Q such that:

$$\forall i \in [\![1, r]\!], \exists a_i \mid m_i \equiv a_i^p \bmod Q \iff c_i = 1.$$

In particular we need to find good primes Q for the condition sequence $(1, \dots, 1)$ in order to construct morphisms from K^* into finite fields $\mathbb{F}_Q$. Remark that the primes should not divide any of the integers m_i. Now if we fix a prime $Q > 3$ we have the following situation:

- if $Q \equiv 1 \bmod p$ then $\mathbb{F}_Q$ contains a primitive p-th root of unity and $\frac{\mathbb{F}_Q^*}{(\mathbb{F}_Q^*)^p} \simeq \mathbb{F}_p$;
- if $Q \not\equiv 1 \bmod p$ then $\mathbb{F}_Q$ does not contain a primitive p-th root of unity and $\frac{\mathbb{F}_Q^*}{(\mathbb{F}_Q^*)^p} \simeq \{1\}$.

Therefore we can have different strategies depending on our goal. If we want the condition $(1, \dots, 1)$ to be verified we might consider primes which are not congruent to 1 modulo p as long as we do not need a non-trivial p-th root of 1 to be in the field $\mathbb{F}_Q$.

Let us now describe how the algorithm operates to find a good prime $Q \equiv 1 \bmod p$. First we have to draw a prime Q and verify that it is congruent to 1 modulo p. This happens with probability $\frac{1}{p-1}$. Then we have to check whether the sequence of conditions C is verified by $(m_1, \dots, m_r)$ and Q. We know that $m_i^{\frac{Q-1}{p}} \bmod Q$ has order 1 or p which is equivalent to m_i being a power or not. We have therefore Algorithm 3 where we make use of two functions: `CheckPowerCondition` which has been explained, and `DrawPrime` which corresponds to the way we select the candidates for the prime numbers. One can follow [2] and generate a random prime number in a range given as argument. We could also generate a random prime first and then draw the next prime each time we need a new one.

For a random prime $Q \equiv 1 \bmod p$ the probability that a condition is true is equal to $\frac{p-1}{p}$ if $c_i = 0$ and $\frac{1}{p}$ if $c_i = 1$. Therefore if $\mathrm{Hw}(C)$ designates the Hamming weight of C we have

$$\mathbb{P}\left(\prod_{i=1}^{r} \texttt{CheckPowerCondition}(m_i, c_1, Q, p) = 1\right) = (\frac{1}{p})^{\mathrm{Hw}(C)} \times (\frac{p-1}{p})^{r-\mathrm{Hw}(C)}.$$

In average the algorithm will try $\frac{p^r}{(p-1)^{r-\mathrm{Hw}(C)}}$ primes before finding one verifying the condition sequence C. In particular the probability that each m_i is equal to a p-th power in $\mathbb{F}_Q$ is $\dfrac{1}{p^r}$ and the algorithm will try $O(p^r)$ primes before finding one verifying the condition sequence $C = (1, \dots, 1)$. Moreover we check if a m_i is a power or not modulo Q by doing a modular exponentiation. Therefore if Q is polynomial in N as it is expected, the complexity of `CheckCubeCondition` will be polynomial in $\log(N)$.

Algorithm 3 Finding a good prime for a sequence $\underline{d}$ and a condition sequence C - `OneGoodPrime`

Require: A reduced sequence $(m_1, \dots, m_r)$, $C = (c_1, \dots, c_r) \in \{0,1\}^r$ and a prime p

Ensure: A good prime Q relatively to (m, C, p) which does not divide any of the m_i'.

```
b ← 0
while $b = 0$ do
    $Q \leftarrow$ DrawPrime
    while $Q \not\equiv 1 \bmod p$ do
        $Q \leftarrow$ DrawPrime
    end while
    $b \leftarrow \prod_{i=1}^{r}$ CheckPowerCondition$(m_i, c_i, Q, p)$
end while
return $Q$
```

If we need to find good primes for a given sequence C – as it will be the case to detect non trivial cubes of units – we repeat Algorithm 3 until obtaining enough primes. The only thing to be careful with is the function `DrawPrime` in the case we generate random primes in a given range. It needs to be large enough so that the time taken before generating the desired number of "good" primes is low enough. If `DrawPrime` generate primes by finding the next one then we repeat this process.

4.4.2. *Detecting powers.* As mentioned earlier the authors of [2, 27] showed how to realise the characters in the case of multiquadratic and multicubic fields. It can be adapted to general real Kummer extension. Consider L/K a Kummer extension of exponents p, q and $S = \langle s_1, \dots, s_n \rangle$ a subgroup of L^*. In order to obtain a non trivial character $\chi_Q : S \to \mathbb{F}_p$ one can do as follow. First select a prime Q such that one can construct a ring morphism from $\mathbb{Z}[\sqrt[p]{m_1}, \dots, \sqrt[p]{m_r}, \sqrt[q]{n_1}, \dots, \sqrt[q]{n_s}]$ to $\mathbb{F}_Q$. The prime Q must be such that for all $i \in [\![1, r]\!]$ the rational m_i has a p-th root in $\mathbb{F}_Q$, and that for all $j \in [\![1, s]\!]$ the rational n_i has a q-th root in $\mathbb{F}_Q$. Moreover since the character needs to be non trivial, $\mathbb{F}_Q$ has to contain a primitive p-th root of unity, i.e. $Q = 1 \pmod p$. After the reduction modulo Q, one can verify if $\phi_Q(s_i)$ is a p-power by computing an exponentiation with exponent $\frac{p-1}{Q}$. The composition of this and ϕ_Q will be the character χ_Q. Following the analysis of [2], finding such a prime Q can be done in time $\mathrm{Poly}(N)$ so finding R good primes can be done in $\mathrm{Poly}(NR)$ with the maximum of the Q to be also in $\mathrm{Poly}(NR)$ Finally if B is an upper bound for the size of the coefficients of $s_1, \dots, s_n$ then one can construct and apply the characters in time $\mathrm{Poly}(BNRn)$. Then detecting the powers can be done using Algorithm 4 in polynomial time with respect to the entries.

Algorithm 4 Compute non trivial p-powers of a subgroup of K^* – `DetectPowers`

Require: A real Kummer extension L/K of exponents p, q, $S = \langle s_1, \dots, s_n \rangle$ a subgroup of K^*

Ensure: $\lambda_1, \dots, \lambda_{n'} \in [\![0, p-1]\!]^n$ such that $\prod_{i=1}^{n} s_i^{\lambda_{j,i}}$ is a p-power in K, for all $j \in [\![1, n']\!]$.

```
Generate sufficiently enough characters $\chi_{Q_1}, \dots, \chi_{Q_R}$.        ▷ Use OneGoodPrime
$M \leftarrow [\chi_{Q_j}(s_i)]_{i,j} \in M_{n,R}(\mathbb{F}_p)$
$N \leftarrow \ker(M)$                                                        ▷ Left Kernel in $\mathbb{F}_p$
return $N$ as a matrix in $\mathbb{Z}$
```

Remark that Algorithm 4 returns exponents corresponding to true $p-$powers with probability at least $1-p^{-(R-n)}$ under assumption that the characters constructed are uniformly distributed in the dual of $S/(S \cap K^p)$. We never encountered failure during our computations.

Heuristic 1. *Let $S < K^*$ with L/K a real Kummer extension of exponents p, q. Then the characters χ_Q described previously are uniformly distributed in* $\mathrm{Hom}(S/(S \cap K^p), \mathbb{F}_p)$.

4.4.3. *Reducing a basis subgroup.* In order to find a basis of a subgroup $U < \mathcal{O}_K^\times$ one can use Pohst's modified LLL [30] algorithm on the matrix $\mathrm{ApproxLog}_L(U)$. In order to find a transformation matrix with small coefficients, one can follow [2] and compute a LLL on a matrix of the form $\left[\mathrm{Id}|C \cdot \mathrm{ApproxLog}_L(U)\right]$. This leads to a reduction in $\mathrm{Poly}(NB)$ if B is a bound on the size of the elements of $\mathrm{ApproxLog}_L(U)$, as we take C with size polynomial in N. The use of a reducing algorithm allows also to find a basis of better quality. One can choose to use another reducing algorithm such as BKZ [33].

4.4.4. *Reducing an element with respect to a lattice.* In order to retrieve a short generator g of a principal ideal from another generator h, we mentioned that one can try to solve a CVP with respect to the Log-unit lattice. In order to do so, we followed [27] and computed the result of Babaï's nearest plane algorithm using Kannan's embedding technique. This technique can be used more generally to reduce an element $[h, \mathrm{ApproxLog}_L(h)]$ with respect to a sublattice $\mathrm{ApproxLog}_L(U)$ of $\mathrm{ApproxLog}_L(\mathcal{O}_L^\times)$, in order to control the size of the elements which are handled. Recall that if B is an upper bound of the norm of the vectors of the basis of $\mathrm{ApproxLog}_L(U)$ then one can consider the matrix

$$\left[\begin{array}{c|c} \mathrm{ApproxLog}_L(U) & \mathbf{0} \\ \hline \mathrm{ApproxLog}_L(h) & B \end{array}\right] = \left[\begin{array}{c|c} \mathrm{ApproxLog}_L(u_1) & 0 \\ \mathrm{ApproxLog}_L(u_2) & 0 \\ \vdots & \vdots \\ \mathrm{ApproxLog}_L(u_m) & 0 \\ \hline \mathrm{ApproxLog}_L(h) & B \end{array}\right].$$

Reducing it with a LLL algorithm is expected to reduce the last row to the Log-embedding of a shorter element in the same coset. In order to obtain again a transformation matrix with small coefficients, we consider a matrix of the form

$$\left[\mathrm{Id} \mid \begin{array}{c|c} C \times \mathrm{ApproxLog}_K(U) & \mathbf{0} \\ \hline C \times \mathrm{ApproxLog}_K(h) & B \end{array}\right].$$

We will denote by `RKEBabai`(U, h) this procedure.

4.4.5. *Computing p-th roots.* The authors of [2] were able to exhibit a recursive algorithm in order to compute square roots in a multiquadratic field. The method cannot be adapted to Kummer extension of exponents larger than 3. We then implemented a classical method using approximations of complex embeddings – which can be traced back to the seminal paper introducing LLL algorithm [24] – and developped a version allowing to decode approximations of elements in a subfield (if one exists) instead of the largest field. Let us briefly describe these algorithms. We refer the interested reader to the PhD thesis of the first author [26], which study them in details and generalise them to the computation of roots of general polynomials.

If $y = x^p$, since L/K is a real extension, one can compute x_l a rational approximation of x with precision l together with $\mathbf{b}_l$ the vector of approximations of the $\mathbb{Q}-$basis elements of L. Store $\mathrm{LLL}\left(\left[\mathbf{b}_l|C \cdot \mathrm{Id}\right]\right)$ in the matrix L_l and the transformation matrix in U_l. Then create the row vector $\mathbf{x}_l = [d_L x_l \mid \mathbf{0} \mid B]$ with B being a coefficient larger than the maximum euclidean norm of the rows of L_l. Then apply a LLL reduction on $\left[\begin{array}{c} L_l \mid \mathbf{0} \\ \hline \mathbf{x}_l \end{array}\right]$ which is essentially a reduction on $\left[\begin{array}{c|c|c} \mathbf{b}_l & C \times \mathrm{Id} & \mathbf{0} \\ \hline d_L x_l & \mathbf{0} & B \end{array}\right]$. After the reduction, the central part of the last row vector is expected to be the vector of coefficients of $d_L x$ in K. Theorems 1 and 2 ensure that $d_L x$ as integral coefficients in the basis $\mathfrak{IB}(L)$. Then one obtains a candidate x_{test} for x. If $x_{test}^p \neq y$ then increase the precision. We will denote by `TestDecode` the procedure which takes L_l and x_l as input and outputs x. As explained in [27] an advantage of this method is that one can save the unitary matrix U_l. Therefore if the precision needs to be increased from l to l', one can apply U_l to $\left[\mathbf{b}_l|d_L \cdot \mathrm{Id}\right]$ before applying LLL. This is expected to save some time. A last advantage is that one can used L_l for several p-th root extractions, such as in the computation of $\mathcal{O}_K^\times$. Finally the complexity of this method is in $\mathrm{Poly}(N, B)$ if B is an upper bound on the bit size of the coefficients of x [26].

Implementing these ideas, we obtain Algorithm 5 which computes roots of powers such as outputted by `DetectPowers`. In this context, we will write `InitBasisLatt` and `UpdateBasisLatt` the procedures which respectively initialise and update to a larger precision the basis lattice matrix of L/K.

Algorithm 5 Compute the p-th roots in L/K – `ElementsFromPower`

Require: A Kummer extension $L = K(\sqrt[p]{m_1}, \dots, \sqrt[p]{m_r})$ with $K = \mathbb{Q}(\sqrt[q]{n_1}, \dots, \sqrt[q]{n_s})$, a subgroup $S = \langle s_1, \dots, s_n \rangle$ of K^* and $V = \langle y_1, \dots, y_t \rangle < S^p$ non-trivial powers of p

Ensure: A basis $\langle x_1, \dots, x_t \rangle$ of $V^{1/p}$

```
$Y \leftarrow$ RKEBabai($S^p, V$)                    ▷ Reduce in the Log-representation
$Y \leftarrow$ Sort($X$)
$X \leftarrow \emptyset$
$[L, U, l] \leftarrow$ InitBasisLatt($L/K$)
for $i = 1$ to $n$ do
    $[L, U, l] \leftarrow$ UpdateBasisLatt($L/K$, PrecisionEvaluation($y_i$), $U$)
    $x \leftarrow (x_i)_l$
    $x \leftarrow$ TestDecode($L, x$)
    $X \leftarrow X \cup \{x\}$
end for
return $X$
```

Improvement. One can use the relative twisted Fourier transform to do LLL algorithms in subfields. Consider $L = K(\sqrt[p]{m_1}, \dots, \sqrt[p]{m_r})$ a real Kummer extension. Given a complex embeddings σ of y up to precision l there are p possibilities for a p-th root of $\sigma(y)$, i.e. for the good approximation of $\sigma(x)$. This leads to p^{p^r} possibilities (in fact $p^{(p^r-1)/2}$ if p is odd because there are $(p^r - 1)/2$ pairs of conjugate embeddings). Fix one of these possibilities. It gives a set of possible approximations of coefficients of x over L by the twisted Fourier transform. Then it is possible to decode these coefficients in L using the LLL method and test if the good element has been computed. If not, test another possibility. Because the number of possibilities is exponential, this technique becomes quickly impractical. If one assumes that the complexity of LLL is N^4 where $N = [K : \mathbb{Q}]$ then this technique requires to compute $p^{(p^r-1)/2} \times p$ LLL algorithms on matrices of dimensions N/p^r, which leads to a complexity in $p^{(p^r-1)/2-4r+1}$ times the complexity of the original technique with LLL. Experimentally it is useful to descend down to $r = 2$ for $p < 5$ and down to $r = 1$ for $p < 11$.

Finally, in order to reduce the running time, one can try to bound the norm of the powers. Let y be one of the powers outputted by `DetectPowers`, and $S = \langle s_1, \dots, s_n \rangle$ the subgroup of K^* given as input. Then one can reduce y with respect to $\mathrm{ApproxLog}(S^p)$ using `RKEBabai` as explained above. Experimentally, it allows to hasten the computations.

4.5. **Computing the unit group and solving the PIP.** In order to compute the unit group of a real Kummer extension of exponents p, q we will able to use Algorithm 1 several times. Indeed if L/K is a Kummer extension of exponents p, q then each of the minimal subextensions $L(\sqrt[p]{M_\alpha})$ can be written as $\mathbb{Q}(\sqrt[p]{M_\alpha})(\sqrt[q]{n_1}, \dots, \sqrt[q]{n_s})$, i.e. a Kummer extension of $\mathbb{Q}(\sqrt[p]{M_\alpha})$ of exponent q. Therefore if one applies Algorithm 1 to L/K, when it reaches the simple subextensions $L(\sqrt[p]{M_\alpha})$ in step 2, one can again apply `KE_Units` instead of `UnitGroup`. This leads to Algorithm 6. We do not show its complexity, and instead refer to the analysis done in [2, 8, 9, 27].

Theorem 3. *Consider $K = L(\sqrt[p]{m_1}, \dots, \sqrt[p]{m_r})$ with $L = \mathbb{Q}(\sqrt[q]{n_1}, \dots, \sqrt[q]{n_s})$ a real Kummer extension with p and q prime integers such that $[L : \mathbb{Q}] = p^r q^s$. Under the assumption of Heuristic 1 and GRH Algorithm 6 heuristically*

Algorithm 6 Compute the unit group of a Kummer extension L/K of exponents p, q. – `RKE_Units`

```
Require: A Kummer extension $L = K(\sqrt[p]{m_1}, \dots, \sqrt[p]{m_r})$ with $K = \mathbb{Q}(\sqrt[q]{n_1}, \dots, \sqrt[q]{n_s})$.
Ensure: A basis of the torsion-free part of the unit group $\mathcal{O}_K^\times$.
if ($r = 1$ and $s \leqslant 1$) then
    return UnitGroup($L$).
end if
if ($r = 1$ and $s > 1$) then
    return KE_Units($L/\mathbb{Q}(\sqrt[p]{m_1})$). ▷ Compute a basis of $U = O_L^\times$ by considering $L$ as a Kummer extension of $\mathbb{Q}(\sqrt[p]{m_1})$.
else
    Choose $u, v$ two independent elements of $\widetilde{\mathrm{Hom}(L/K)}$.
    Recursively compute a basis of $U = \mathcal{O}_{L^u}^\times \mathcal{O}_{L^{uv}}^\times \dots \mathcal{O}_{L^{u^{p-1}v}}^\times \mathcal{O}_{L^v}^\times$
    $V \leftarrow$ DetectPowers($U, p$)
    $V \leftarrow$ ElementsFromPower($V, p$)
    $U \leftarrow$ BasisFromGeneratingSet($\langle U, V \rangle$)
    return $U$
end if
```

computes $\mathcal{O}_L^\times$ in $\mathrm{Poly}(\ln(|D_L|))L_P(2/3+\epsilon, c)$ for some $c > 0$ and $\epsilon > 0$ as small as desired, with probability at least $1 - (pq)^{-N}$, where P is the product of all primes dividing the m_i and n_j.

As we saw the procedure to find a generator of a principal ideal is very similar to the one to compute the unit group. Therefore we obtain easily Algorithm 7. The analysis of the running time is similar to the one of Algorithm 6 which gives the same complexity since solving the PIP on the subfields of dimension pq is also sub-exponential.

Theorem 4. *Consider $L = K(\sqrt[p]{m_1}, \dots, \sqrt[p]{m_r})$ with $K = \mathbb{Q}(\sqrt[q]{n_1}, \dots, \sqrt[q]{n_s})$ a real Kummer extension with p and q prime integers such that $[L : \mathbb{Q}] = p^r q^s$ and a principal ideal I. Under the assumption of Heuristic 1 and GRH Algorithm 7 heuristically computes a generator of I in $\mathrm{Poly}(\ln(\mathrm{N}_{L/\mathbb{Q}}(I)), \ln(|D_L|))L_P(2/3+\epsilon, c)$ for some $c > 0$ and $\epsilon > 0$ as small as desired, with probability at least $1 - (pq)^{-N}$, where P is the product of all primes dividing the m_i and n_j.*

Algorithm 7 Solve the PIP in a Kummer extension of exponents p, q – `RKE_PIP`

```
Require: A principal ideal $I$ of a Kummer extension $L = K(\sqrt[p]{m_1}, \dots, \sqrt[p]{m_r})$ with $K = \mathbb{Q}(\sqrt[q]{n_1}, \dots, \sqrt[q]{n_s})$, the unit group $\mathcal{O}_L^\times$.
Ensure: A generator $g$ of $I$.
if ($r = 1$ and $s \leqslant 1$) then
    return Generator(I).
end if
if ($r = 1$ and $s > 1$) then
    return KE_PIP($L/\mathbb{Q}(\sqrt[p]{m_1})$).    ▷ Compute a generator of $I$ by considering $L$ as a Kummer extension of $\mathbb{Q}(\sqrt[p]{m_1})$.
else
    Choose $u, v$ two independent elements of $\widetilde{\mathrm{Hom}(L/K)}$.
    Recursively compute generators of $\mathrm{N}_{L^u}(I)\mathrm{N}_{L^{uv}}(I), \dots, \mathrm{N}_{L^{uv^{p-1}}}(I), \mathrm{N}_{L^v}(I)$ and use Equation 3 to have $h$ a generator of $I^p$.
    return ElementsFromPower($[\mathcal{O}_L^\times, h], p$).
end if
```

Again, for the proof of Theorem 4 we refer to the analysis done in [2, 8, 9, 27].

Solving the SPIP. In order to solve the SPIP, one only has to reduce the ouput h of Algorithm 7 modulo the Log-unit lattice. For this we apply `RKEBabai` to $\mathcal{O}_L^\times$ and h.

5. Experimental results

We implemented the algorithms for real Kummer extensions using Magma V2.24-9 [10], with the procedures described but without the compact representation of elements, which leads to exponential algorithms. We recall that ourimplementation is publicly available [2]. The section is organised as follows.

- We study in Subsection 5.1 the probability to retrieve a short generator of a principal ideal through an attack using the algorithms presented in Section 4 ; we computed data for Kummer extensions with one and two exponents, and compare the results to the ones of [2, 27]. This allows us to identify Kummer fields with degree p^2 and defined by small integers to be fields over which the SPIP is more difficult to solve.
- Then in Subsection 5.2 we study further the geometrical situation. In particular we compute the size of the key vector normalised by the volume of the Log-unit lattice and the quality of the basis obtained for the Log-unit lattice through Algorithm 6. We focus on Kummer extensions with one exponent with degree p^2 and compare them with other number fields.

Some timings concerning the running time of Algorithm 6 can be found in Appendix A.

5.1. Probability of solving the SPIP.

The first way we studied the possibility of solving the SPIP over real Kummer extensions was to launch attacks with Algorithm 7 and `RKEBabai`. As a matter of fact, we did not do proper attacks because computing ideal norms can be quite long even though the theoretical complexity is polynomial. However the knowledge of the secret key allows us to compute the HNF of the norms efficiently, and the rest of the attack is unchanged. We tried to retrieve generator of principal ideals (g) such that the coefficients of the generators g are drawn uniformly in $\{-1, 0, 1\}$. The previous observations in [2, 27] seemed to show two phenomena. The probability of retrieving a generator increased when:

- the length of the sequence defining the field was increasing;
- the size of the coefficients of the sequence was increasing.

Part of our work has been to verify that it happens on all Kummer extensions.

5.1.1. *Kummer extensions with one exponent.* First let us consider fields of the form $K = \mathbb{Q}(\sqrt[p]{m_1}, \dots, \sqrt[p]{m_r})$. We present the results obtained in Tables 2 and 3. There is one table for each exponent p defining the field, except for Table 3 which presents the results for the three exponents (11, 13, 17). For each exponent we computed attacks for fields defined by sequences of increasing length and increasing coefficients ; moreover the coefficients are consecutive prime numbers. For each field we provide the probability of retrieving a generator when LLL or BKZ$_{20}$ is used to reduce the different basis during the algorithms.

Table 3. Experimental results for Kummer extension of $\mathbb{Q}$ with degree p^2 and exponents 11, 13 and 17.

Field exponent	11					13					17
Rank of lattice	60					84					144
First coefficient	2	3	5	7	11	2	3	5	7	11	2
Success LLL (%)	52	100	100	100	100	20	99.6	100	100	100	0
Success BKZ (%)	82	100	100	100	100	78	100	100	100	100	18

[2] `https://github.com/AndLesav/spip-on-kummer`

Table 2. Experimental results for Kummer extension of $\mathbb{Q}$ with exponents 3, 5, and 7

(a) $p = 3$, and $r = 2$ or 3

Sequence length r	2					3				
Rank of the lattice $r_1 + r_2 - 1$	4					13				
First coefficient	2	3	5	7	11	2	3	5	7	11
Success LLL (%)	38	86	98	98	99.99	47	100	100	100	100
Success BKZ (%)	41	89	100	99	100	48	100	100	100	100

(b) $p = 3$, and $r = 4$ or 5

Sequence length r	4					5				
Rank of lattice $r_1 + r_2 - 1$	40					121				
First coefficient	2	3	5	7	11	2	3	5	7	11
Success LLL (%)	56	100	100	100	100	77.6	100	100	100	100
Success BKZ (%)	61	100	100	100	100	74.3	100	100	100	100

(c) $p = 5$, and $r = 2$ or 3

Sequence length r	2					3				
Rank of lattice $r_1 + r_2 - 1$	12					62				
First coefficient	2	3	5	7	11	2	3	5	7	11
Success LLL (%)	58	75	100	100	100	65	99	100	100	100
Success BKZ (%)	64	79	99	100	100	73	97	100	100	100

(d) $p = 7$ and $r \in \{2, 3\}$

Sequence length r	2					3				
Rank of lattice $r_1 + r_2 - 1$	24					171				
First coefficient	2	3	5	7	11	2	3	5	7	11
Success LLL (%)	86.6	100	100	100	100	80.6	100	100	100	–
Success BKZ (%)	84.9	100	100	100	100	98.7	100	100	100	–

We can remark that the two phenomena described before seem to be true for all exponents p. Moreover the probability of success seems to converge quickly to one. For similar degrees and rank of $\mathrm{Log}_K(\mathcal{O}_K^{\times})$ we can remark that we obtain a better probability of success with fields defined by longer sequences and smaller exponents. Compare for instance fields of degree 7^3 in Table 2.c and fields of degree 13^2 or 17^2 in Table 3.

Fields with degree p^2 :. Let us now focus our attention on the subclass of fields of the form $K = \mathbb{Q}(\sqrt[p]{m_1}, \sqrt[p]{m_2})$. First we see that again the probability of success converges quickly to 1 when m_1 increases. Now fix $(m_1, m_2) = (2, 3)$ and let p vary. One can find the percentages of success plotted in Figure 2.

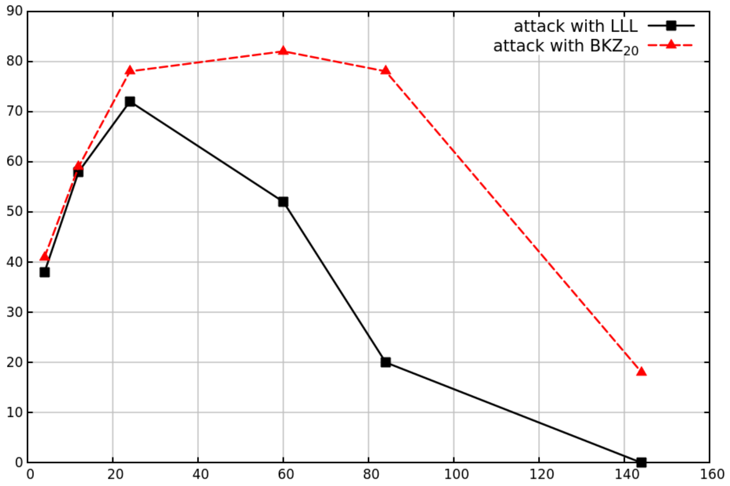


FIGURE 2. Percentage of success of an attack with LLL or BKZ$_{20}$ for fields $K = \mathbb{Q}(\sqrt[p]{2}, \sqrt[p]{3})$ plotted against the rank $r_1 + r_2 - 1$ of $\mathrm{Log}_K(\mathcal{O}_K^\times)$

We can notice that for one or the other method used as a reduction algorithm throughout the procedures, the probability of retrieving a short generator starts to increase but decreases when p is larger than 11. It converges to 0 when using LLL and is bigger when using BKZ$_{20}$ but is still quickly decreasing.

Remark 3 (Importance of studying high degree number fields)*.* One important observation is that computations on high degree number fields were required to observe meaningful data. Indeed when restricted to fields with degree less than 121, i.e. to primes strictly smaller than 11, the probability of success of an attack is quickly increasing and there is no difference between using LLL or BKZ$_{20}$. *This highlights the need to work over high degree number fields.*

Finally one could consider Kummer fields of degree p^2 defined by small integers as an alternative to number fields already used in cryptography such as cyclotomic fields. Indeed, in addition to the data gathered here, their structure could be used to build an efficient arithmetic as done over multiquadratic fields in [2]. One could also consider Kummer fields of degree p if the pattern concerning the probability of success (decreasing with the length of the sequence) is still valid. However we cannot confirm or invalidate it. We only have access to the classical algorithms to do computations on these fields, thus preventing examining fields with high degree.

5.1.2. *Kummer extensions with two exponents.* Consider real Kummer extensions of the form $L = K(\sqrt[p]{m_1}, \ldots, \sqrt[p]{m_r})$ with $K = \mathbb{Q}(\sqrt[q]{n_1}, \ldots, \sqrt[q]{n_s})$. We tried to verify whether the phenomena mentioned earlier were still true over such fields or not. To do so, we computed data for several fixed ground field K and varying parameters for the extension L. Because of efficiency reasons, we were restricted in our choice of parameters. Indeed, our implementation seems to be slower over Kummer extensions with two exponents than extensions with one exponent. We only present the probabilities with LLL because the ones with BKZ$_{20}$ are very similar, due to the fact that the ranks of the Log-unit lattices manipulated are small.

Simple Kummer field as ground field and square relative degree. First let us consider fields such that K is a simple Kummer field $\mathbb{Q}(\sqrt[q]{n})$ and $L = K(\sqrt[p]{p_1}, \sqrt[p]{p_2})$ with p_1, p_2 being consecutive prime numbers. The data gathered can be found in Tables 4, 5 and 6.

Table 4. Success of an attack (in %) with over Kummer extensions of the form $L = K(\sqrt[p]{p_1}, \sqrt[p]{p_2})$ with $K = \mathbb{Q}(\sqrt[2]{n})$

Exponent p	3					5					7				
$r_1 + r_2 - 1$	9					25					49				
Coefficient p_1	2	3	5	7	11	2	3	5	7	11	2	3	5	7	11
$n = 2$	–	62	71	63	69	–	64	77	59	70	–	82	78	73	65
$n = 5$	12	50	–	44	42	11	61	–	64	54	66	61	–	55	58
$n = 13$	24	86	86	90	79	60	67	89	89	88	79	90	92	81	92

Table 5. Success of an attack (in %) over Kummer extensions of the form $L = K(\sqrt[p]{p_1}, \sqrt[p]{p_2})$ with $K = \mathbb{Q}(\sqrt[3]{n})$

Exponent p	5				
$r_1 + r_2 - 1$	37				
Coefficient p_1	2	3	5	7	11
$n = 2$	–	76	87	86	77
$n = 5$	31	92	–	100	97
$n = 13$	67	86	97	97	–

Table 6. Success of an attack (in %) with over Kummer extensions of the form $L = K(\sqrt[p]{p_1}, \sqrt[p]{p_2})$ with $K = \mathbb{Q}(\sqrt[5]{n})$

Exponent p	3				
$r_1 + r_2 - 1$	22				
Coefficient p_1	2	3	5	7	11
$n = 2$	–	73	85	79	72
$n = 5$	54	97	–	96	97
$n = 13$	47	92	96	98	97

We can see that the results are different for these fields than for Kummer extensions with one exponent. For each pair (p, q) it seems that the probability of success does not converge to 1 when the coefficients (p_1, p_2) increase ; for some pairs the probability is even decreasing. We are still able to retrieve a high percentage of generators, but one should remark that the dimensions are all relatively low. We mentioned in Remark 3 the importance of studying high dimensional number fields i.e. with dimension at least greater than 100, and we stress that the data we were able to produce regarding Kummer extensions with two exponents do not meet this requirement. Thus the observations made from these data might not be representative of the asymptotic behaviours.

Increasing $[L : K]$ *with constant exponent.* Now let us consider extensions $L = K(\sqrt[p]{p_1}, \ldots, \sqrt[p]{p_r})$ with fixed L and p, with increasing length sequence r of consecutive prime numbers.

TABLE 7. Success of an attack over Kummer extensions of the form $L = K(\sqrt[3]{p_1}, \dots, \sqrt[3]{p_r})$ with $K = \mathbb{Q}(\sqrt[q]{11})$

Exponent q	2			5		
Length r	2	3	4	2	3	4
$r_1 + r_2 - 1$	9	27	81	22	67	202
Success with LLL (%)	28	37	47	52	52	–

The data in Table 7 seems to show that again, the phenomena observed over Kummer fields with one exponent cannot be seen as clearly over Kummer extensions with two exponents, at least for $q = 2$.

Conclusion. The probabilities of successfully retrieving the private key seem to be smaller and to differ much more than for the previous type of fields. It could be an indication that breaking the regularity of the field structure makes the attack more difficult. However one has to remark that we lack of data, and that they are essentially over fields with relatively low degrees.

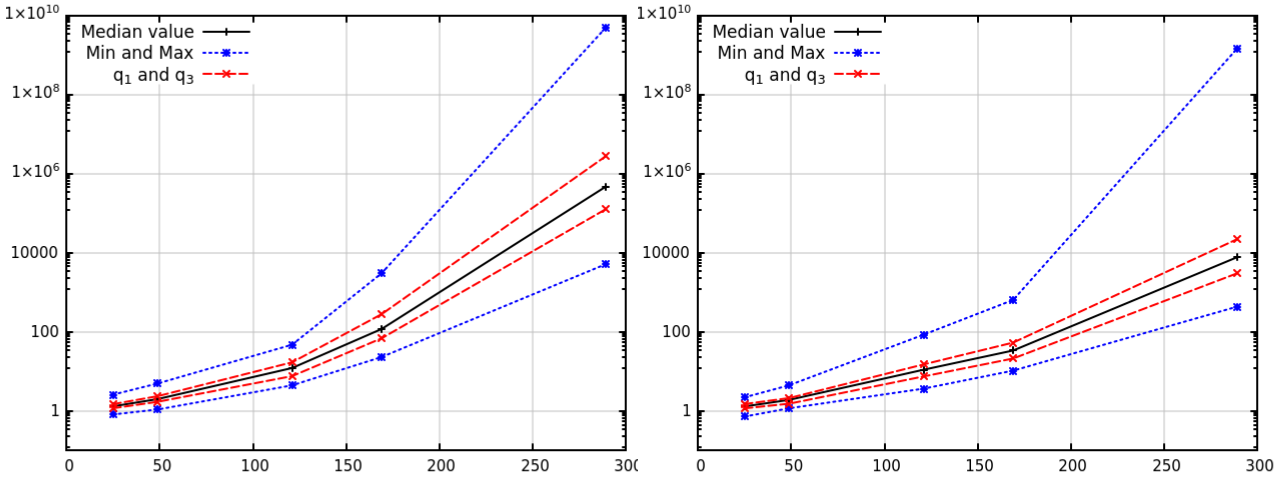
5.2. **Kummer field with square degrees.** Let us now focus on Kummer extensions with one exponent, since we are able to compute data for high dimensional fields. Moreover recall that we identified Kummer extensions of degree p^2 defined by the sequence $(2, 3)$ as fields for which recovering a short generator through the Log-unit lattice could be more difficult than over other number fields. Thus all Kummer extensions considered further are defined by sequences of the first prime integers. In order to study further the situation we looked into the possibility of recovering a short generator through an enumeration process. In order to evaluate the cost of enumerations, we used the function $\texttt{EnumerationCost}(L, m^2)$ of MAGMA. It computes an estimation of the number of nodes to visit during an enumeration process of short vectors of lattice L within the ball $B(0, m)$. Moreover we studied the quality of the basis obtained by computing several parameters. Given a basis B (whose vectors are sorted by increasing norms), evaluating its orthogonality can be difficult. Let us denote by r and V respectively the rank and the volume of the lattice generated by B. We chose to compute:

(1) the Hermite factor $\delta_0 = \frac{\|b_1\|}{\sqrt[r]{V}}$ which is used to evaluate the quality of basis reduction on random lattices;
(2) the orthogonality defect $\delta = \sqrt[r]{\frac{\prod_{i=1}^{r} \|b_i\|}{V}}$ which expresses the overall orthogonality of the basis.

We gathered data of cyclotomic fieldsand Kummer fields. In order to obtain data on cyclotomic fields of larger degree we used the subgroup C of cyclotomic units, which has a very small index [17]. For some fields they are even equal, for example for power-of-2 cyclotomics (under GRH). Even if C is not $\mathcal{O}_K^\times$ one can argue that it is close to it and is used by the authors of [17] to solve the SPIP over cyclotomic fields.

5.2.1. *Comparison with naïve attacks.* We compared the results exhibited in Subsubsection 5.1.1 with naïve attacks consisting in reducing the basis given as input and looking at the small vectors obtained. We again considered LLL and BKZ$_{20}$ as reduction algorithms. Using only LLL, we could not retrieve any generator starting from $p = 11$. All keys were retrieved with BKZ$_{20}$ up to $p = 13$, but no generator were found for $p = 17$. This tends to indicate that the ideal lattices considered react as generic lattices under reduction algorithms. Hence, one can expect reduction algorithms to be ineffective asymptotically, and that using Algorithm 7 together with a reduction modulo the Log-unit lattice will produce better results.

5.2.2. *Norm of the retrieved vector.* We also compared the norm of retrieved generators with the norm of the key (in coefficient representation), to verify that the attack does not retrieve short enough solutions. We will call *approximation factor* the quotient $\|h\|_2 \,/\, \|g\|_2$, where h is the generator obtained after Algorithm 7 together with a reduction modulo the Log-unit lattice and g is the key generator. One can find data about the approximation factors that we got in Figure 3.



(A) With LLL

(B) With BKZ$_{20}$

FIGURE 3. Approximation factors plotted against the dimension of the field (in logarithmic scale), after LLL or BKZ reduction.

One can remark that after both reduction algorithms, the approximation factors seem to have an exponential behaviour, even if using BKZ$_{20}$ gives significantly better results. Thus, the attack through the Log-unit lattice coupled with the level of lattice reduction that we considered does not allow us to solve the SPIP of Kummer fields of the form $\mathbb{Q}(\sqrt[p]{2}, \sqrt[p]{3})$, even approximately. This is clearly different from the situation over cyclotomic fields [17].

5.2.3. *Norm of the key vector.* One important geometrical parameter is the size of the key when compared to the volume of the Log-unit lattice, in order to know if retrieving it through a CVP computation or an enumeration process is conceivable. In addition to the size of the key vector we studied the cost one would obtain for an enumeration.

Let us recall a quick result which can be found in [3, 17].

Lemma 3. *Let K be a number field, H be the subspace of $\mathbb{R}^n$ orthogonal to $\mathbf{1} = (1, \ldots, 1)$ and p_H be the orthogonal projection on H. Then for any $g \in K$ one has $\mathrm{Log}_{\mathbf{K}}(\mathbf{g}) = p_H(\mathrm{Log}_{\mathbf{K}}(\mathbf{g})) + \frac{\ln |\mathrm{N}_{K/\mathbb{Q}}(g)|}{n}\mathbf{1}$.*

One can conclude from Lemma 3 that if g is the secret key, then the norm of the key is

$$\sqrt{\sum_{i=1}^{n} \left(\ln |\sigma_i(g)| - \frac{\|\mathrm{Log}_{\mathbf{K}}(\mathbf{g})\|_1}{n} \right)^2}.$$

For each field we computed the ratio of the norm of $p_H(\mathrm{Log}_K(g))$ by the scaled volume of the Log-unit lattice $\sqrt[r]{V_K}$ where $r = r_1 + r_2 - 1$. We computed the median value of this ratio for each set of keys, and the corresponding enumeration cost. Let us denote by M_K said median value, and EC_K the bit-size of the corresponding enumeration cost.

The attacks showed that the SPIP seems to be more resistant over fields of the form $\mathbb{Q}(\sqrt[p]{2}, \sqrt[p]{3})$, so we will focus on them. In order to have a better idea of the situation, let us compare them with:

- cyclotomic fields of prime conductor p;
- cyclotomic fields of the form $\mathbb{Q}(\zeta_{2^n})$;
- Kummer fields of degree p^3 and Kummer fields of exponent 3 and defined by successive primes i.e. of the form $\mathbb{Q}(\sqrt[3]{2}, \sqrt[3]{3}, \ldots, \sqrt[3]{p_r})$.

Remember that in order to compute data for high degree cyclotomic fields, we considered C the subgroup of cyclotomic units. Again we computed the median values of the quotients $\|p_H(\mathrm{Log}_{\mathbf{K}}(\mathbf{g}))\|_2 / V_K^{1/r}$ and the corresponding enumeration costs. One can find the values corresponding to the first parameter plotted in Figure 4.

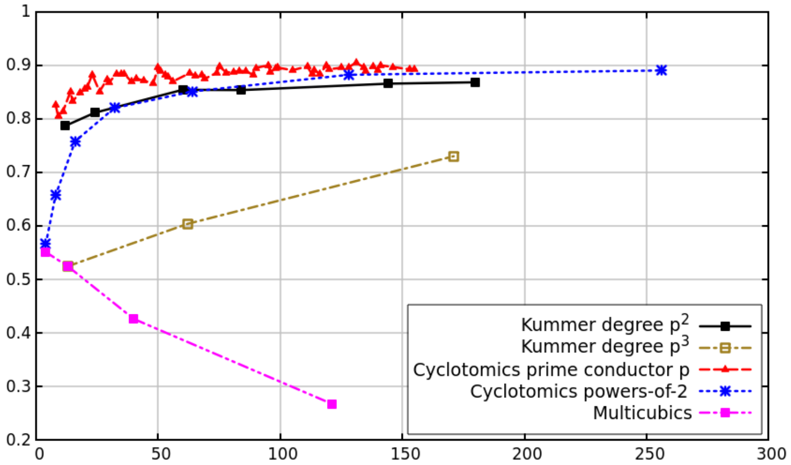


Figure 4. Median values M_K plotted against $r_1 + r_2 - 1$ for Kummer fields of degrees p^2 and p^3, and different types of cyclotomic fields

We can remark that the values for Kummer extensions of square degrees are close to the ones for cyclotomic fields, in particular the ones of the form $\mathbb{Q}(\zeta_{2^n})$. Moreover the values for cyclotomic with conductor of the form p^k with $k \geqslant 2$ are also similar, even if we did not plot them for clarity purposes. For Kummer fields of degree p^3, the plot suggests that the values could asymptotically be close to the ones over the previous fields. However we cannot confirm this because the state of our implementation does not allow us to compute the units for the following prime $p = 11$, which corresponds to a field of degree 1331. We can see that the size of keys over multicubic fields is decreasing quickly, which is coherent with the probability of success already observed. This also confirms the differences between fields with increasing exponents such that the defining sequence has a constant length, and fields with a constant exponent such that the length of the defining sequence is increasing.

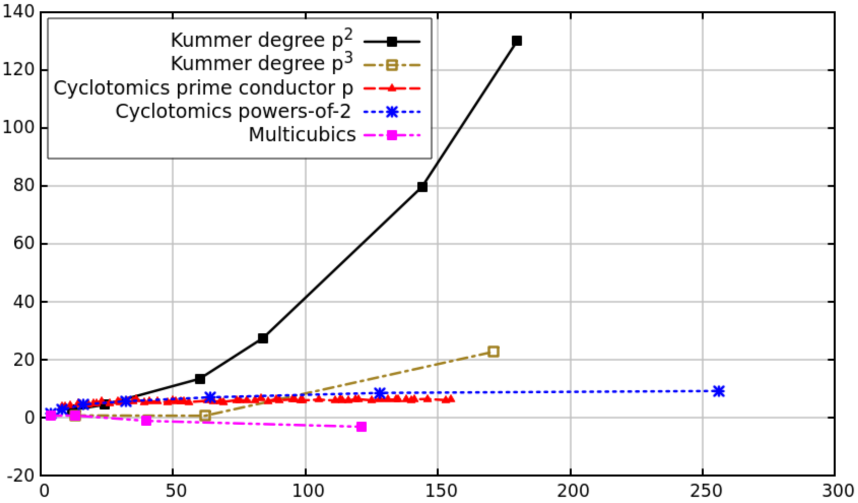


Figure 5. Median values of enumeration cost EC_K plotted against $r_1 + r_2 - 1$ over Kummer fields of degrees p^2 and p^3, and cyclotomic fields, after LLL reductions

Enumeration costs. With the previous observations, one could expect to obtain similar enumeration costs for cyclotomic and Kummer fields. However we can see on Figures 5 and 6 – which show the corresponding enumeration costs with the use of LLL and BKZ$_{20}$ respectively – that the costs are low over cyclotomic fields (and close one to each other) but asymptotically bigger over Kummer fields of degree p^2 and p^3. Again the situation is worse for

Kummer fields of degree p^2 than p^3. Regarding the influence of BKZ$_{20}$, it has again a positive and noticeable impact for ranks greater than 80 i.e. degrees greater 160, and only over Kummer fields of degree p^2. These observations coupled with the values of the enumeration cost obtained seem to indicate that Kummer extensions of degree p^2 could be better options than cyclotomic fields when it comes to building a cryptosystem which security relies on the hardness of solving the SPIP or the ISVP. Indeed for the field $\mathbb{Q}(\sqrt[19]{2}, \sqrt[19]{3})$, the enumeration cost after BKZ$_{20}$ is still large enough to prevent an enumeration process.

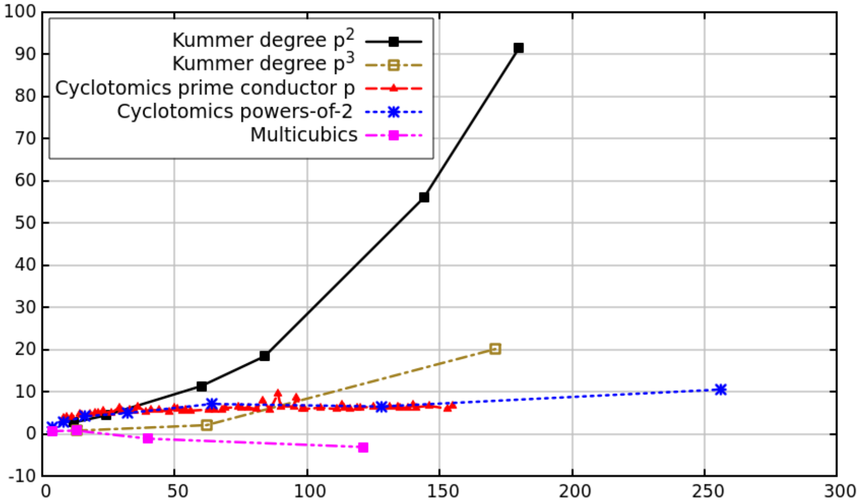


FIGURE 6. Median values of enumeration cost EC_K plotted against $r_1 + r_2 - 1$ for Kummer fields of degrees p^2 and p^3, and cyclotomic fields after BKZ$_{20}$ reductions

5.2.4. *Basis of Log-unit lattice.* As mentioned before, we studied further the situation by computing several parameters to evaluate the quality of the basis of $\operatorname{Log}_K(\mathcal{O}_K^\times)$ for the fields K considered. Results of these computations are gathered in Figures 7, 8 and 9 for the same type of fields considered in the previous analysis.

Hermite factor. One can see that for all types of fields considered, the shortest vector of the basis of the Log-unit lattice is relatively short, as shown in Figure 7 where the δ_0 is plotted. There is only one plot for each type of field because the values are not widely modified by BKZ$_{20}$.

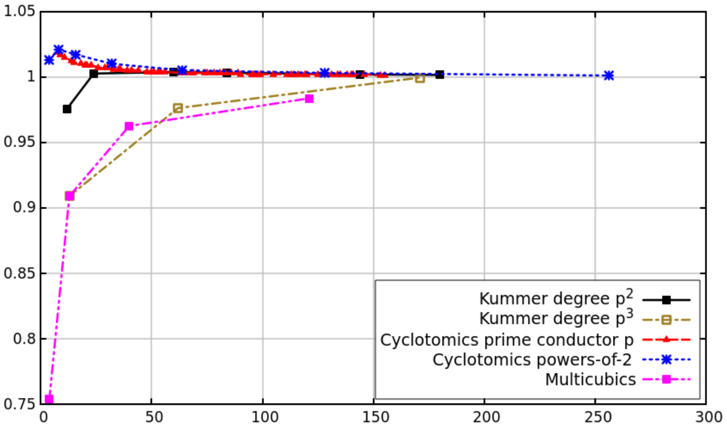


FIGURE 7. Values of δ_0 over Kummer fields of degrees p^2 and p^3, and over cyclotomic fields after LLL reductions

Orthogonality defect. Now let us consider the orthogonality defect δ for high degree number fields. We plotted the values obtained after LLL in Figure 8 and after BKZ_{20} in Figure 9. One can notice that the only fields for which BKZ_{20} has a significant impact are Kummer fields with degree p^2, as it was the case for the enumeration cost shown in Figures 5 and 6. This indicates that for these fields, the basis of the Log-unit lattice obtained by our procedures is not well reduced, and better reduction algorithms modify the basis. This is completely different than over cyclotomic fields where the basis formed by cyclotomic units are massively orthogonal and are not modified by reduction algorithms. We can also conclude from the values for Kummer extensions of degree p^3 that it is possible to obtain reduced basis of Log-unit lattices which are not as orthogonal as over cyclotomic units, but are not reduced further by BKZ_{20}.

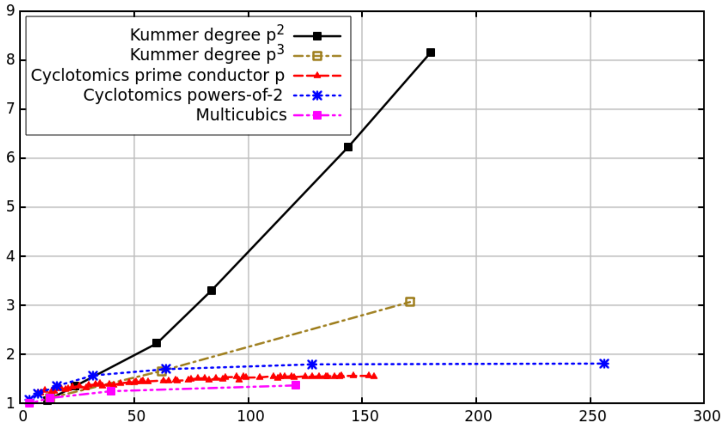


FIGURE 8. Values of δ over Kummer fields of degrees p^2 and p^3, and over cyclotomic fields after LLL reductions

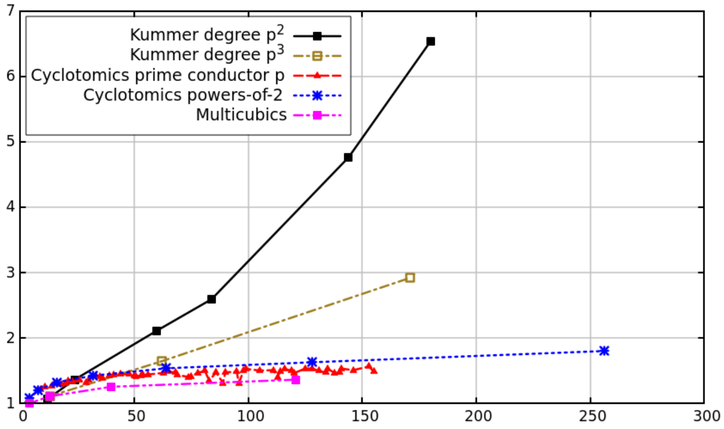


FIGURE 9. Values of δ over Kummer fields of degrees p^2 and p^3, and over cyclotomic fields after BKZ reductions

References

[1] L. Babai. "On Lovasz lattice reduction and the nearest lattice point problem". In: *Combinatorica* 6 (Mar. 1986), pp. 1–13. DOI: 10.1007/BF02579403.

[2] J. Bauch et al. "Short Generators Without Quantum Computers: The Case of Multiquadratics". In: *Advances in Cryptology – EUROCRYPT 2017*. Ed. by J.-S. Coron and J. B. Nielsen. Cham: Springer International Publishing, 2017, pp. 27–59. ISBN: 978-3-319-56620-7.

[3] O. Bernard and A. Roux-Langlois. "Twisted-PHS: Using the Product Formula to Solve Approx-SVP in Ideal Lattices". In: *Advances in Cryptology – ASIACRYPT 2020*. Ed. by S. Moriai and H. Wang. Cham: Springer International Publishing, 2020, pp. 349–380. ISBN: 978-3-030-64834-3.

[4] O. Bernard et al. *Log-S-unit lattices using Explicit Stickelberger Generators to solve Approx Ideal-SVP*. Cryptology ePrint Archive, Report 2021/1384. https://ia.cr/2021/1384. 2021.

[5] J.-F. Biasse and C. Fieker. "Improved techniques for computing the ideal class group and a system of fundamental units in number fields." In: *Algorithmic Number Theory, 10th International Symposium, ANTS-IX, San Diego CA, USA, July 9-13, 2012. Proceedings*. Vol. 1. Open Book Series. Mathematical Science Publishers, 2012, pp. 113–133.

[6] J.-F. Biasse and F. Song. "Efficient quantum algorithms for computing class groups and solving the principal ideal problem in arbitrary degree number fields". In: *Proceedings of the Twenty-Seventh Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2016, Arlington, VA, USA, January 10-12, 2016*. 2016, pp. 893–902. DOI: 10.1137/1.9781611974331.ch64. URL: http://dx.doi.org/10.1137/1.9781611974331.ch64.

[7] J.-F. Biasse and C. Fieker. "Subexponential class group and unit group computation in large degree number fields". In: *LMS Journal of Computation and Mathematics* 17.A (2014). DOI: 10.1112/S1461157014000345.

[8] J.-F. Biasse and C. Vredendaal. "Fast multiquadratic S-unit computation and application to the calculation of class groups". In: *The Open Book Series* 2 (Jan. 2019), pp. 103–118. DOI: 10.2140/obs.2019.2.103.

[9] J.-F. Biasse et al. *Norm relations and computational problems in number fields*. 2020. arXiv: 2002.12332 [math.NT].

[10] W. Bosma, J. Cannon, and C. Playoust. "The Magma algebra system. I. The user language". In: *J. Symbolic Comput.* 24.3-4 (1997). Computational algebra and number theory (London, 1993), pp. 235–265. ISSN: 0747-7171. DOI: 10.1006/jsco.1996.0125. URL: http://dx.doi.org/10.1006/jsco.1996.0125.

[11] J. P. Buhler, H. W. Lenstra, and C. Pomerance. "Factoring integers with the number field sieve". In: *The development of the number field sieve*. Ed. by A. K. Lenstra and H. W. Lenstra. Berlin, Heidelberg: Springer Berlin Heidelberg, 1993, pp. 50–94. ISBN: 978-3-540-47892-8.

[12] P. Campbell, M. Groves, and D. Shepherd. *Soliloquy : a cautionary tale*. 2014.

[13] H. Cohen. *Advanced Topics in Computational Number Theory*. Graduate Texts in Mathematics. Springer New York, 2012. ISBN: 9781441984890. URL: https://books.google.cz/books?id=OFjdBwAAQBAJ.

[14] H. Cohen. *A Course in Computational Algebraic Number Theory*. Berlin, Heidelberg: Springer-Verlag, 1993. ISBN: 0-387-55640-0.

[15] J. Conway and N. Sloane. *Sphere Packings, Lattices and Groups*. Vol. 290. Jan. 1988. ISBN: 978-1-4757-2018-1. DOI: 10.1007/978-1-4757-2016-7.

[16] R. Cramer, L. Ducas, and B. Wesolowski. "Short Stickelberger Class Relations and Application to Ideal-SVP". In: *EUROCRYPT*. 2017.

[17] R. Cramer et al. "Recovering Short Generators of Principal Ideals in Cyclotomic Rings". In: *Advances in Cryptology – EUROCRYPT 2016*. Ed. by M. Fischlin and J.-S. Coron. Berlin, Heidelberg: Springer Berlin Heidelberg, 2016, pp. 559–585. ISBN: 978-3-662-49896-5.

[18] L. Ducas. "Advances on quantum cryptanalysis of ideal lattices". In: 2017.

[19] K. Eisenträger et al. “A quantum algorithm for computing the unit group of an arbitrary degree number field”. In: *Proceedings of the Annual ACM Symposium on Theory of Computing* (May 2014), pp. 293–302. DOI: 10.1145/2591796.2591860.

[20] S. Galbraith. *Mathematics of public key cryptography*. Apr. 2012. ISBN: 9781107013926. DOI: 10.1017/CBO9781139012843.

[21] C. Gentry. “A Fully Homomorphic Encryption Scheme”. AAI3382729. PhD thesis. Stanford, CA, USA: Stanford University, 2009. ISBN: 978-1-109-44450-6.

[22] C. Gentry and S. Halevi. “Implementing Gentry’s Fully-Homomorphic Encryption Scheme”. In: vol. 6632. May 2011, pp. 129–148. DOI: 10.1007/978-3-642-20465-4_9.

[23] R. Kannan. “Minkowski’s Convex Body Theorem and Integer Programming”. In: *Mathematics of Operations Research* 12.3 (1987), pp. 415–440. ISSN: 0364765X, 15265471. URL: http://www.jstor.org/stable/3689974.

[24] A. Lenstra, H. Lenstra, and L. Lovász. “Factoring Polynomials with Rational Coefficients”. In: *Mathematische Annalen* 261 (Dec. 1982). DOI: 10.1007/BF01457454.

[25] A. Lesavourey. “A note on the discriminant and prime ramification of some real Kummer extensions”. working paper or preprint. Nov. 2021. URL: https://hal.archives-ouvertes.fr/hal-03456622.

[26] A. Lesavourey. “Usability of structured lattices for a post-quantum cryptography: practical computations, and a study of some real Kummer extensions”. PhD thesis. University of Wollongong, 2021.

[27] A. Lesavourey, T. Plantard, and W. Susilo. “Short Principal Ideal Problem in multicubic fields”. In: *Journal of Mathematical Cryptology* 14.1 (1Jan. 2020), pp. 359–392. DOI: https://doi.org/10.1515/jmc-2019-0028. URL: https://www.degruyter.com/view/journals/jmc/14/1/article-p359.xml.

[28] D. Micciancio and S. Goldwasser. *Complexity of Lattice Problems*. USA: Kluwer Academic Publishers, 2002. ISBN: 0792376889.

[29] A. Pellet-Mary, G. Hanrot, and D. Stehlé. “Approx-SVP in Ideal Lattices with Pre-processing”. In: *Advances in Cryptology – EUROCRYPT 2019*. Ed. by Y. Ishai and V. Rijmen. Cham: Springer International Publishing, 2019, pp. 685–716. ISBN: 978-3-030-17656-3.

[30] M. Pohst. “A Modification of the LLL Reduction Algorithm.” In: *Journal of Symbolic Computation* 4 (Aug. 1987), pp. 123–127. DOI: 10.1016/S0747-7171(87)80061-5.

[31] P. Samuel. *Algebraic theory of numbers*. Hermann, 1970.

[32] B. Schmal. “Diskriminanten,$\mathbb{Z}$-Ganzheitsbasen und relative Ganzheitsbasen bei multiquadratischen Zahlkörpern”. In: *Archiv der Mathematik* 52 (1989), pp. 245–257.

[33] C. Schnorr. “A hierarchy of polynomial time lattice basis reduction algorithms”. In: *Theoretical Computer Science* 53.2 (1987), pp. 201–224. ISSN: 0304-3975. DOI: https://doi.org/10.1016/0304-3975(87)90064-8. URL: http://www.sciencedirect.com/science/article/pii/0304397587900648.

[34] N. Smart and F. Vercauteren. “Fully homomorphic encryption with relatively small key and ciphertext sizes”. In: *Fully Homomorphic Encryption with Relatively Small Key and Ciphertext Sizes* (Jan. 2010), pp. 420–443.

[35] C. van Vredendaal. “Exploiting mathematical structures in cryptography”. English. Proefschrift. PhD thesis. Department of Mathematics and Computer Science, June 2018. ISBN: 978-90-386-4508-7.

[36] J. Westlund. “On the Fundamental Number of the Algebraic Number-Field $k(\sqrt[p]{m})$”. In: *Transactions of the American Mathematical Society* 11.4 (1910), pp. 388–392.

Appendix Appendix A Timings for unit group computation

Kummer extensions with one exponent. We first report some timings for fields with one exponent, i.e. of the form $K = \mathbb{Q}(\sqrt[p]{m_1}, \ldots, \sqrt[p]{m_r})$. We considered fields such that $m_1, \ldots, m_r$ are consecutive prime numbers. One can see from Table 8 that the timings increase with respect to the size of the coefficients of the defining sequence $m = (m_1, \ldots, m_r)$ similarly to what has been observed for multicubic and multiquadratic fields [2, 27]. This is not surprising as we saw in Proposition 10 that the discriminant of these number fields is deeply connected to $\prod_{q \in \mathcal{P}(m)} q$.

Table 8. Timings for some Kummer fields with one exponent

Coefficients $m_1, \ldots, m_r$	Exponent p	Dimension	Timings for $\mathcal{O}_K^\times(s)$
$(2, 3, 5, 7)$	3	81	9.6801
$(5, 7, 11, 13)$	3	81	10.140
$(11, 13, 17, 19)$	3	81	24.880
$(2, 3, 5, 7, 11)$	3	243	233.57
$(5, 7, 11, 13)$	3	243	1181.8
$(11, 13, 17, 19, 23)$	3	243	16520.0
$(2, 3, 5$	5	125	36.690
$(3, 5, 7)$	5	125	73.720
$(5, 7, 11)$	5	125	549.37
$(2, 3, 5)$	7	343	3628.5
$(3, 5, 7)$	7	343	18700.
$(5, 7, 11)$	7	343	98449.
$(2, 3)$	11	121	167.03
$(5, 7)$	11	121	1007.9
$(11, 13)$	11	121	6908.8
$(2, 3)$	13	169	1297.2
$(2, 3)$	17	289	32230.
$(2, 3)$	19	289	1.2026E5

Kummer extensions with two exponents. We then give timings for fields with two exponents, i.e. of the form $L = K(\sqrt[p]{m_1}, \ldots, \sqrt[p]{m_r})$ with $K = \mathbb{Q}(\sqrt[q]{n_1}, \ldots, \sqrt[q]{n_s})$. We considered fields such that K is a simple field – meaning of the form $\mathbb{Q}(\sqrt[q]{n})$ – and such that $m_1, \ldots, m_r, n$ are prime integers.

Table 9. Timings for some Kummer extension L/K with two exponents

n	$(m_1, \ldots, m_r)$	q	p	Dimension	$\mathcal{O}_L^\times(s)$
2	(3, 5)	2	5	50	9.02
2	(11, 13)	2	5	50	38.56
13	(2, 3)	2	5	50	7.99
13	(11, 17)	2	5	50	58.53
2	(3, 5)	2	7	98	131.36
2	(11, 13)	2	7	98	2489.4
13	(2, 3)	2	7	98	94.91
13	(11, 17)	2	7	98	11170.0
2	(3, 5)	3	5	75	29.90
2	(11, 13)	3	5	75	436.38
13	(2, 3)	3	5	75	33.30
13	(11, 17)	5	5	75	4191.0
11	(2,3,5,7)	2	3	162	208.40
11	(2,3,5)	5	3	135	305.93
11	(2,3,5,7)	5	3	405	25371.0

Again, we see that the running times are quite sensible to the increase of the primes defining the field.