Log-$\mathcal{S}$-unit Lattices Using Explicit Stickelberger Generators to Solve Approx Ideal-SVP*

Olivier Bernard[1,2], Andrea Lesavourey[1], Tuong-Huy Nguyen[1,3], and Adeline Roux-Langlois[1]

[1] Univ Rennes, CNRS, IRISA, France
olivier.bernard@normalesup.org,
{andrea.lesavourey, tuong-huy.nguyen, adeline.roux-langlois}@irisa.fr
[2] Thales, Gennevilliers, France
[3] DGA Maîtrise de l'Information, Bruz, France

Abstract. In 2020, Bernard and Roux-Langlois introduced the Twisted-PHS algorithm to solve Approx-Svp for ideal lattices on any number field, based on the PHS algorithm by Pellet-Mary, Hanrot and Stehlé. They performed experiments for prime conductors cyclotomic fields of degrees at most 70, one of the main bottlenecks being the computation of a log-$\mathcal{S}$-unit lattice which requires subexponential time.
Our main contribution is to extend these experiments to cyclotomic fields of degree up to 210 for most conductors m. Building upon new results from Bernard and Kučera on the Stickelberger ideal, we use explicit generators to construct full-rank log-$\mathcal{S}$-unit sublattices fulfilling the role of approximating the full Twisted-PHS lattice. In our best approximate regime, our results show that the Twisted-PHS algorithm outperforms, over our experimental range, the CDW algorithm by Cramer, Ducas and Wesolowski, and sometimes beats its asymptotic volumetric lower bound. Additionally, we use these explicit Stickelberger generators to remove almost all quantum steps in the CDW algorithm, under the mild restriction that the plus part of the class number verifies $h_m^+ \leq O(\sqrt{m})$.

Keywords: Ideal lattices, Approx-SVP, Stickelberger ideal, S-unit attacks, Twisted-PHS algorithm

1 Introduction

The ongoing NIST Post-Quantum Cryptography competition illustrates the importance of the *Learning With Errors* (Lwe) problem as an intermediate building block for a wide variety of cryptographic schemes. Most of these cryptographic schemes rely on a structured version of the Lwe problem allowing for much more satisfactory performance, compared to schemes based on the unstructured Lwe problem. The first structured variant of Lwe, later known as the Ring-Lwe

* © IACR 2022. This paper is an extended version of the article published in the Proceedings of ASIACRYPT 2022, Part xx, LNCS xxxxx, Springer, which is available at https://doi.org/xx.xxxx/xxx-x-xxx-xxxxx-x_xx.

problem, is shown to be at least as hard as the *Approximate Shortest Vector Problem* on ideal lattices (Approx-id-Svp) using quantum worst-case to average-case reductions [SSTX09,LPR10]. One important matter is to determine whether using this structured version of LWE could lower the hardness hypothesis of the scheme. Notably, an efficient solver for Approx-id-SVP would render the worst-case to average-case reduction to Ring-LWE meaningless as a security argument. Note however that even in this case, this would not directly imply an efficient solver for the Ring-LWE problem.

In the case of arbitrary lattices, Approx-SVP is a well-studied hard problem. It consists in finding relatively short vectors of a given lattice, within an approximation factor of the shortest vector. The best theoretical trade-off between run-time and approximation factor is known as Schnorr's hierarchy [Sch87]: one can reach, for any $\alpha \in (0,1)$, an approximation factor $2^{\widetilde{O}(n^{\alpha})}$ in time $2^{\widetilde{O}(n^{1-\alpha})}$. The closest known practical algorithm to this trade-off is the BKZ algorithm [SE94], a generalization of the well-known LLL algorithm [LLL82]. In the particular case of ideal lattices, i.e., lattices that correspond to ideals of the ring of integers $\mathcal{O}_K$ of a number field K, one could hope that the best reduction algorithms would remain those associated with arbitrary lattices. However, this simplifying assumption seems questionable, since the underlying number-theoretic structure is precisely what makes Ring-LWE a more efficient building block. Thus, going beyond the BKZ algorithm and estimating the hardness of Approx-id-SVP using algebraic ideas has gathered more attention, starting by works from [EHKS14,CGS14,BS16,CDPR16]. Earlier works aimed at the more restricted case of Approx-id-SVP for principal ideals. A strategy for this case is devised as a two parts algorithm. The first part requires solving the Principal Ideal Problem (PIP), i.e., finding any generator of the ideal; the second part aims at finding the shortest one, by solving a Closest Vector Problem (CVP) in the so-called *log-unit lattice*. This shortest generator is expected to solve Approx-SVP for a sufficiently small approximation factor. Ultimately, for the particular case of cyclotomic fields of prime power conductors, [CDPR16] proved that Approx-id-SVP on principal ideals is solvable in quantum polynomial time, but only reaching an approximation factor $2^{\widetilde{O}(\sqrt{n})}$.

Subsequent works in a more general case can be divided in two different paths. The first one [CDW17,CDW21] aimed at extending the results from [CDPR16] to arbitrary ideal lattices over any cyclotomic fields, while still reaching in quantum polynomial time an approximation factor $2^{\widetilde{O}(\sqrt{n})}$. One of their contributions is to reduce the arbitrary ideal case to the principal ideal case by solving the *Close Principal Multiple Problem* (CPMP): given an ideal $\mathfrak{b}$, one computes an ideal $\mathfrak{c}$ of small algebraic norm s.t. $\mathfrak{bc}$ is a principal ideal. In order to ensure that $\mathfrak{c}$ has a small norm, a new key technical ingredient, specific to cyclotomic fields, was the use of the Stickelberger lattice, which has good geometric properties. Then, the results from [CDPR16] are applied to $\mathfrak{bc}$ to obtain a candidate short element of $\mathfrak{b}$, using the fact that $\mathfrak{c}$ has a small norm. The concrete consequences of this method were experimented in [DPW19], under different regimes which mainly differ upon which CVP solver is used. The first regime (called "Naive") uses Babai's Nearest Plane algorithm, whereas the second regime uses a heuristic Cvp algorithm relatively to *ad hoc* pseudo-norms. From these experiments, the asymptotic performance of those decoding algorithms was estimated, which led to simulated approximation factors reached by the CDW algorithm. Finally, given experimentally verified constants, a volumetric lower bound was derived for the approximation factors that could be reached in the best scenario. According to this lower bound, the CDW algorithm is expected to beat the BKZ_{300} algorithm for cyclotomic fields of degrees at least larger than 7000. Since NIST submissions based on structured lattices rely on cyclotomic fields of degree at most 1024, this could be perceived as somewhat reassuring.

The second path is explored in [PHS19,BR20]. Those works, applying to arbitrary number fields, replace the two reductions steps from [CDW21] with a single Cvp instance, so as to find a principal multiple ideal which is not only of small algebraic norm, but is also generated by a small element. A key ingredient achieving this is to use a generalization of the units of $\mathcal{O}_K$, called $\mathcal{S}$-units; this formalism was an underlying feature of [PHS19] and was later made explicit in [BR20]. The PHS algorithm splits into a preprocessing phase and a query phase. The preprocessing phase consists in preparing the decoding of a particular lattice depending only on the number field K, *via* the computation of a hint following Laarhoven's Cvp with preprocessing algorithm [Laa16], which takes exponential time. Then, any Approx-id-Svp instance in K can be interpreted as an Approx-Cvp instance in this lattice, efficiently solved thanks to the hint. Up to the preprocessing, the query phase yields new time/quality trade-offs: as in [CDW21] for cyclotomic fields, it reaches approximation factor $2^{\widetilde{O}(\sqrt{n})}$ in quantum polynomial time; however, the PHS algorithm also allows for better trade-offs than Schnorr's hierarchy, from polynomial to $2^{\widetilde{O}(\sqrt{n})}$ approximation factors. On the downside, the computation of the lattice itself takes classically subexponential time, which is a serious obstacle for studying their geometry and obtaining concrete asymptotic estimations as was done in [DPW19] for the CDW algorithm.

Then, [BR20] introduced Twisted-PHS, a "Twisted" version of the PHS algorithm whose main difference lies in a fundamental modification of the underlying lattice, thanks to a natural normalization coming from the *Product Formula*. The problem of finding a short vector is expected to be better encoded within this new lattice, ultimately leading to smaller outputs. Even though the proven trade-offs between runtime and approximation factor remain the same for the Twisted-PHS algorithm as for the PHS algorithm, very significant improvements have been experimentally illustrated in [BR20, Fig. 5.3], showing much better approximation factors compared to the PHS algorithm for number fields of degree up to 60, where Laarhoven's Cvp algorithm is replaced in practice by Babai's Nearest Plane algorithm [Bab86]. These were to our knowledge the first experimental evidence of the geometric peculiarity of normalized log-$\mathcal{S}$-unit lattices and of the practical potential of this type of attack. In this practical version, experiments are solely limited by the classical complexity of computing the lattice.

Unfortunately, the attained dimensions, up to 60, are not sufficient to assess the practical limits of the Twisted-PHS algorithm: its heuristic analysis [BR20]

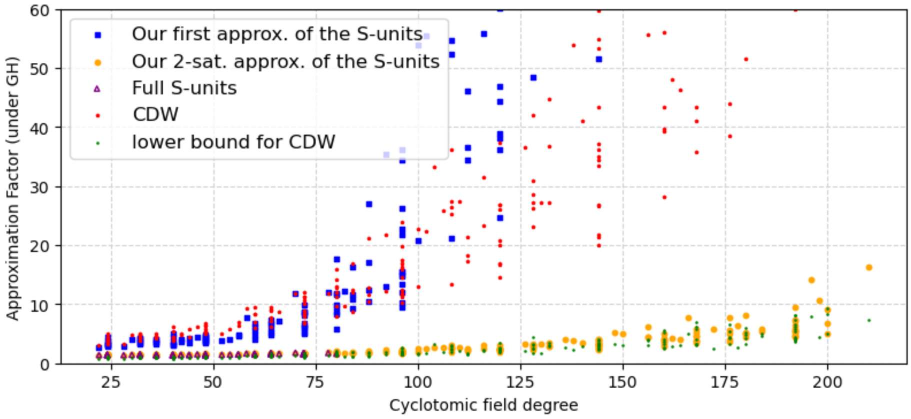

FIG. 1.1 – Average of approximation factors using (sublattices of) log-$\mathcal{S}$-unit lattices in cyclotomic fields, over random simulated instances, achieved by our implementation of Twisted-PHS (our work) as compared to those achieved by CDW [DPW19], assuming the Gaussian Heuristic throughout all instances.

could give only a loose upper bound, or miss unexpected performance in practical dimensions due to its asymptotic nature, even in the cryptographical range.

Our contributions. We develop theoretical and practical improvements regarding algorithms for solving Approx-id-SVP, in both lines of work following the CDW algorithm and the Twisted-PHS algorithm. Even though the hardness of the Approx-id-SVP does not concretely impact the security of cryptographic schemes, it is important to get a better understanding of both approaches, which are the only ones successfully exploiting the structure of a lattice.

Our core ingredient is the introduction of a full-rank family of independent $\mathcal{S}$-units, whose algebraic properties are proven in §3. In §4, we use this family to remove most quantum steps of the CDW algorithm, leaving only one step during a preprocessing phase done once for any given field, and one step for each query.

In §5, this family allows us to achieve experiments on algorithms in the (Twisted-)PHS family, for most cyclotomic fields of dimension up to 210. By comparison, previous experiments [DPW19,BR20] only considered cyclotomic fields of conductors $m = p > 2$ prime and $m = 2^e > 2$. Our work comes with an improved implementation of the initial Twisted-PHS algorithm, allowing us to extend the experiments conducted in [BR20] up to dimension 80 and for all cyclotomic fields. It also includes different regimes of approximation for this algorithm, using sublattices of the log-$\mathcal{S}$-unit lattice obtained thanks to our new construction beyond dimension 80 up to 210. These regimes yield concrete upper bounds for the approximation factors that could be reached by the full Twisted-PHS algorithm up to dimension 210, as illustrated in Fig. 1.1:

1. The depicted approximation factors were estimated using the *Gaussian Heuristic*, matching the *exact ones* obtained by [BR20] without this hypothesis.

2. Our best approximate regime yields approximation factors that are comparable (sometimes even smaller) to the asymptotical volumetric lower bound regime of the CDW algorithm.

In [DPW19], it was already noted that the PHS approach should outperform the lower bound, but at the cost of computing Laarhoven's hint in exponential time. Our work show that for medium dimensions, where asymptotical results should start to be meaningful, the Twisted-PHS algorithm is at least comparable to the CDW lower bound, though without this exponential hint precomputation.

As suggested in [BR20], and illustrated in small dimensions, the Twisted-PHS algorithm performance may be explained by the peculiar geometric nature of the log-$\mathcal{S}$-unit lattice. In our work, this is confirmed by the computations of several geometrical parameters on the basis obtained by our implementation, across all considered cyclotomic fields, sublattices and factor bases. This specificity, observed in a wide variety of regimes and even in medium dimensions, suggest a deeper explanation, a possibility recently explored by Bernstein and Lange [BL21]. We provide a full implementation of all our experiments at `https://github.com/ob3rnard/Tw-Sti`.

Technical overview. In [BR20], the log-$\mathcal{S}$-unit lattice needed for the preprocessing phase was built using generic number theory tools. Our main idea is to shortcut this generic computation by considering a maximal family $\mathfrak{F}$ of independent $\mathcal{S}$-units, where $\mathcal{S}$ verifies some conditions (detailed in §3), leading to sublattices of the log-$\mathcal{S}$-unit lattice. The family $\mathfrak{F}$ is composed of three parts:

1. Circular units, also known as cyclotomic units, e.g. in [Was97, §8];
2. Generators coming from the explicit proof of Stickelberger's theorem proof;
3. Real $\mathcal{S}$-units coming from the maximal real subfield K_m^+ of K_m, where K_m is the cyclotomic field of conductor m.

The first two parts are classically easy to compute. In particular, the effectiveness of the second part comes from two recent results of [BK21]: the knowledge of an explicit *short* $\mathbb{Z}$*-basis* of the Stickelberger ideal for *any* conductor [BK21, Th. 3.6], and the effective computations of generators corresponding to these short relations using Jacobi sums [BK21, §5]. On the contrary, the last part still relies on generic number theory tools which are classically costly, but are now performed in a number field of half degree, which propels us to degree 210.

As an important theoretical contribution, we prove in Th. 3.11 that $\mathfrak{F}$ is indeed a full-rank family of multiplicatively independent $\mathcal{S}$-units, by computing explicitly its (finite) index in the full $\mathcal{S}$-unit group. This can be seen as a generalization of the strategy of [CDW17, Def. 2] to obtain a full-rank lattice of class relations, restricted to the relative class group. In particular, our result proves the experimentally conjectured value [DPW19, Rem. 3] of the index of their family.

Finally, the index of $\mathfrak{F}$ contains a large power of 2 that can be removed using classical 2-saturation techniques of §3.5, leading to a family $\mathfrak{F}_{\mathsf{sat}}$. We then use the explicit knowledge of these special $\mathcal{S}$-units in two different situations.

Theoretical improvements of the CDW algorithm. In §4, we remove almost all quantum steps of the CDW algorithm while still guaranteeing its approximation factor [CDW21, Th. 5.1], at the small price of restricting to cyclotomic fields s.t. $h_m^+ \leq O(\sqrt{m})$ (Hyp. B.1), whereas [CDW21, Ass. 2] uses $h_m^+ \leq \mathrm{poly}(m)$, where h^+ denotes the plus part of the class number (defined in §2.2).

For that purpose, we first propose an equivalent rewriting of [CDW21, Alg. 7], making explicit some hidden steps that are useful for subsequent modifications. Then, the explicit Stickelberger generators and real $\mathcal{S}$-units are used to remove the last call to the quantum Pip solver. Finally, considering the module of *all* real class group relations allows us to remove the quantum random walk mapping any ideal of K_m into the relative class group. This last part uses our Th. 3.11 and needs Hyp. B.1 to obtain the same bound on the approximation factor.

Only two quantum steps remain: the first is performed once to compute real $\mathcal{S}$-units in K_m^+, of degree only half, the second is for solving the ClDl for each query.

Experimenting the Twisted-PHS algorithm in medium dimensions. We apply Twisted-PHS [BR20] on our full-rank sublattices of the log-$\mathcal{S}$-unit lattice, yielding *approximated* regimes of the Twisted-PHS algorithm. Up to degree 210, for most conductors, the newly implemented algorithm is used to compute the sublattices associated with $\mathfrak{F}$ and $\mathfrak{F}_{\mathsf{sat}}$, for varying subsets $\mathcal{S}$ according to the number of Galois orbits of totally split primes used. In particular, we explicitly compute the Stickelberger generators and real generators of $\mathfrak{F}$ and effectively perform the 2-saturation of $\mathfrak{F}$ to get $\mathfrak{F}_{\mathsf{sat}}$. Up to degree 80, the whole log-$\mathcal{S}$-unit lattice is also computed, corresponding to a fundamental system $\mathfrak{F}_{\mathsf{su}}$ of $\mathcal{S}$-units. This last computation of $\mathfrak{F}_{\mathsf{su}}$ remains unfeasible at higher dimensions. We evaluate the geometry of all these lattices with standard indicators described in §2.5: the root-Hermite factor δ_0, the orthogonality defect δ and the logarithm of the Gram-Schmidt norms. We consistently observe the same phenomena already pointed out in [BR20, §5.1 and 5.2], that indicate close to orthogonal lattices.

Next, since computing ClDl solutions for random ideals quickly becomes intractable, we simulate this step by sampling random outputs similarly to what was done in [DPW19, Hyp. 8]. Given those targets and the preprocessed lattices associated with $\mathfrak{F}$, $\mathfrak{F}_{\mathsf{sat}}$ and $\mathfrak{F}_{\mathsf{su}}$, we evaluate the approximation factors reached by these different regimes, by assuming the Gaussian Heuristic. These two assumptions, i.e., using simulated targets and the Gaussian Heuristic, are validated by the fact that up to degree 80, where it is feasible to compute the full $\mathcal{S}$-unit group generated by $\mathfrak{F}_{\mathsf{su}}$, our approximation factors match the *exact* approximation factors obtained in [BR20, Fig. 1.1], where those heuristics were not used. Finally, we compare our results to the approximation factors obtained by the CDW algorithm [CDW21] in the “Naive” regime of [DPW19], under the same working assumptions as above. We observe that in our best approximate regime, using $\mathfrak{F}_{\mathsf{sat}}$, our estimated approximation factors are close, and sometimes smaller, than the theoretical lower bound derived in [DPW19]. This suggests that the crossover with BKZ_{300} could be lower than expected for the Twisted-PHS algorithm.

Relations to other works related to $\mathcal{S}$-units. Some recent mathematical results regarding the Stickelberger lattice were established in [BK21]. The authors described, for *any* conductor, an easily computable *short basis* for this lattice, and how to explicitly compute the associated principal ideal generators through Jacobi sums. In our work, this result is brought into fruition to solve Approx-id-Svp. The completion of this short basis into a full-rank lattice of class relations, the effective computation of the explicit generators and the 2-saturation of these elements, yielded the different approximated regimes of Twisted-PHS and allowed us to remove many quantum steps from the CDW algorithm.

In a talk on August 2021 at SIAM Conference,[4] Bernstein announced a joint work with Eisenträger, Rubin, Silverberg and van Vredendaal, by illustrating the construction of small $\mathcal{S}$-units using Jacobi sums that lead to an "$\mathcal{S}$-unit attack" in the power-of-2 conductor case up to degree 64, assuming $h^+_{2^e} = 1$. The talk also announced a paper that has yet to appear. In this light, we are not able to compare our use of explicit Stickelberger generators to their work. However, this talk does neither mention a short basis of the Stickelberger lattice, which is at the heart of our work, nor lift all obstructions to apply it to *any* conductor.

In December 2021, a "filtered-$\mathcal{S}$-unit software" was released by Bernstein, treating the prime $p \leq 43$ conductor case, on a webpage[5] describing the "simplest $\mathcal{S}$-unit attack" using a technique described in [BL21]. This work is not related to our construction. Finally, the authors of [BL21] argued that "spherical models" should not be applied to log-$\mathcal{S}$-unit lattices, which may have particular geometric properties. This phenomenon was experimentally observed already in [BR20], and is confirmed by all of our experiments in medium dimensions.

2 Preliminaries

Notations. For any $i, j \in \mathbb{Z}$ with $i \leq j$, the set of all integers between i and j is denoted by $[\![i, j]\!]$. For any $x \in \mathbb{Q}$, let $\{x\}$ denote its fractional part, i.e., such that $0 \leq \{x\} < 1$ and $x - \{x\} \in \mathbb{Z}$. A vector is represented by a bold letter $\mathbf{v}$, and for any $p \in \mathbb{N}^* \cup \{\infty\}$, its ℓ_p-norm is written $\|\mathbf{v}\|_p$. The n-dimensional vector with all 1's is denoted by $\mathbf{1}_n$. All matrices are given using *row* vectors.

2.1 Cyclotomic fields

We denote the cyclotomic field of conductor m, $m \not\equiv 2 \mod 4$, by $K_m = \mathbb{Q}[\zeta_m]$, where ζ_m is a primitive m-th root of unity. It has degree $n = \varphi(m)$, its maximal order is $\mathcal{O}_{K_m} = \mathbb{Z}[\zeta_m]$ ([Was97, Th. 2.6]), and its discriminant is given precisely by $\Delta_{K_m} = (-1)^{\varphi(m)/2} \frac{m^{\varphi(m)}}{\prod_{p|m} p^{\varphi(m)/(p-1)}}$ ([Was97, Pr. 2.7]), which is of order n^n.

In this paper, we consider *any* conductor $m > 1$ of the general prime factorization $m = p_1^{e_1} p_2^{e_2} \cdots p_t^{e_t}$, $m \not\equiv 2 \mod 4$, and let $q_i = p_i^{e_i}$ for all $i \in [\![1, t]\!]$. In particular, m has exactly t distinct prime divisors. Let G_m denote the Galois group of K_m, which can be made explicit by ([Was97, Th. 2.5]):

$$G_m = \{\sigma_s : \zeta_m \longmapsto \zeta_m^s;\ 0 < s < m, (s, m) = 1\} \simeq (\mathbb{Z}/m\mathbb{Z})^\times.$$

[4] The slides are available at https://cr.yp.to/talks.html#2021.08.20.

[5] This is hosted by https://s-unit.attacks.cr.yp.to/filtered.html.

In particular, we denote by $\sigma_s \in G_m$ the automorphism sending any m-th root of unity to its s-th power. For convenience, the automorphism induced by complex conjugation is written $\tau = \sigma_{-1}$.

The algebraic norm of $\alpha \in K_m$ is defined by $\mathcal{N}(\alpha) = \prod_{\sigma\in G_m} \sigma(\alpha)$, hence the absolute norm element in the integral group ring $\mathbb{Z}[G_m]$ is $N_m = \sum_{\sigma\in G_m} \sigma$.

Maximal real subfield. The maximal real subfield of K_m, written K_m^+, is the fixed subfield of K_m under complex conjugation, i.e., $K_m^+ := K_m^{\langle\tau\rangle} = \mathbb{Q}\big(\zeta_m + \zeta_m^{-1}\big)$. Its maximal order is $\mathcal{O}_{K_m^+} = \mathbb{Z}\big[\zeta_m + \zeta_m^{-1}\big]$ (see e.g. [Was97, Pr. 2.16]).

By Galois theory, since $\langle\tau\rangle$ is a normal subgroup of G_m, the maximal real subfield of K_m is a Galois extension of $\mathbb{Q}$ with Galois group $G_m^+ := \mathrm{Gal}\big(K_m^+/\mathbb{Q}\big)$ isomorphic to $G_m/\langle\tau\rangle$. We identify G_m^+ with the following system of representatives modulo τ restricted to K_m^+:

$$G_m^+ = \big\{\sigma_s|_{K_m^+};\ 0 < s < \tfrac{m}{2}, (s,m) = 1\big\}.$$

Technically, each $\sigma_s|_{K_m^+} \in G_m^+$ extends in G_m to either σ_s or $\tau\sigma_s = \sigma_{-s}$. For simplicity, we always choose to lift $\sigma_s|_{K_m^+} \in G_m^+$ to $\sigma_s \in G_m$ and drop the restriction to K_m^+ which should be clear from the context. This slight abuse of notation appears to be very practical. For example, the corestriction $\mathrm{Cor}_{K_m/K_m^+}\big(\sigma_s|_{K_m^+}\big)$, defined as the sum of all elements of G_m that restricts to $\sigma_s|_{K_m^+}$, namely $\sigma_s + \tau\sigma_s$, is written using the much simpler expression $(1+\tau)\cdot\sigma_s$.

2.2 Real and relative class groups

Fractional ideals of K_m form a multiplicative group $\mathcal{I}_m$ containing the normal subgroup $\mathcal{P}_m := \big\{\langle\alpha\rangle;\ \alpha \in K_m\big\}$ of principal ideals. The quotient group $\mathcal{I}_m/\mathcal{P}_m$ is called the *class group* of K_m and denoted by Cl_m. It is finite and its cardinal h_m is the *class number* of K_m. For any $\mathfrak{b} \in \mathcal{I}_m$, the class of $\mathfrak{b}$ in Cl_m is written $\big[\mathfrak{b}\big]$.

The integral group ring $\mathbb{Z}[G_m]$ acts naturally on $\mathcal{I}_m$; more precisely, for any element $\alpha = \sum_{\sigma\in G_m} a_\sigma\sigma \in \mathbb{Z}[G_m]$, and any $\mathfrak{b} \in \mathcal{I}_m$, $\mathfrak{b}^\alpha := \prod_{\sigma\in G_m} \sigma\big(\mathfrak{b}\big)^{a_\sigma}$. The class group and class number of the maximal real subfield K_m^+ are denoted respectively by Cl_m^+ and h_m^+. The relative norm map $\mathcal{N}_{K_m/K_m^+}$ induces a homomorphism from Cl_m to Cl_m^+, whose kernel is the so-called *relative class group*, written Cl_m^- and of cardinal the *relative class number* h_m^-. Hence, by construction, for any $\mathfrak{b}$ s.t. $\big[\mathfrak{b}\big] \in \mathrm{Cl}_m^-$, $\mathfrak{b}^{1+\tau} \cap K_m^+$ is principal. One important specificity of cyclotomic fields is that the real class group Cl_m^+ embeds into Cl_m *via* the natural inclusion map, which to each ideal class $\big[\mathfrak{b}\big] \in \mathrm{Cl}_m^+$ associates the ideal class $\big[\mathfrak{b}\cdot\mathcal{O}_{K_m}\big] \in \mathrm{Cl}_m$ [Was97, Th. 4.14]. Concretely, it implies that $h_m = h_m^+ \cdot h_m^-$ is the product of the plus part and the relative part of the class number.

Plus part and relative part of the class number. Generally, not much is known about the class number of a number field, and the analytic class number formula [Neu99, Cor. 5.11(ii)] allows obtaining a rough upper bound $h_m \le \widetilde{O}\big(\sqrt{|\Delta_{K_m}|}\big)$.

m	$\varphi(m)$	h_m^+	m	$\varphi(m)$	h_m^+	m	$\varphi(m)$	h_m^+	m	$\varphi(m)$	h_m^+	m	$\varphi(m)$	h_m^+	m	$\varphi(m)$	h_m^+
225	120	1	213	140	1	205	160	2	203	168	1	460	176	1	416	192	1
231	120	1	219	144	1	352	160	1	215	168	1	552	176	1	448	192	1
244	120	1	285	144	1	400	160	1	245	168	1	209	180	1	576	192	1
248	120	4	296	144	1	440	160	5	261	168	1	217	180	1	612	192	1
308	120	1	304	144	1	492	160	1	392	168	1	279	180	1	672	192	1
372	120	1	380	144	1	528	160	1	516	168	1	297	180	1	275	200	1
396	120	1	432	144	1	600	160	1	588	168	1	235	184	1	375	200	1
384	128	1	444	144	1	660	160	1	267	176	1	564	184	1	500	200	1
201	132	1	540	144	1	243	162	1	345	176	1	291	192	1			
207	132	1	237	156	1	249	164	1	368	176	1	357	192	1			

TABLE 2.1 – Additional values of h_m^+ for some m with $\varphi(m) \leq 200$.

In the case of cyclotomic fields though, the structure of the relative class group is better understood. Using analytic means, the relative class number has the following explicit expression [Was97, Th. 4.17]:

$$h_m^- = Qw \cdot \prod_{\chi \text{ odd}} \left(-\tfrac{1}{2} B_{1,\chi}\right),$$

where $w = 2m$ if m is odd and $w = m$ if m is even, $Q = 1$ if m is a prime power and $Q = 2$ otherwise, and $B_{1,\chi}$ is defined by $\frac{1}{f}\sum_{a=1}^{f} a\cdot\chi(a)$ for any odd primitive character χ modulo m of conductor f dividing m. Computing this value is in practice very efficient, using adequate representations of Dirichlet characters.

The hard part of cyclotomic class numbers computations is to obtain the plus part h_m^+, and relatively few of them are known. We use the values from [Was97, Tab. §4], [Mil14, Th. 1.1 and 1.2] and [BFHP21, Tab. 1], consistently assuming the *Generalized Riemann Hypothesis* (GRH). We also provide 58 additional values of h_m^+ in Tab. 2.1 for completeness.

The fact that the plus part of the class number seems much smaller than the relative part is striking. Weber's conjecture claims that $h_{2^e}^+ = 1$ for any $e > 1$, and Buhler, Pomerance and Robertson [BPR04] argue, based on Cohen-Lenstra heuristics, that for all but finitely many pairs (p, e), where p is a prime and e is a positive integer, $h_{p^{e+1}}^+ = h_{p^e}^+$. For prime power conductors, this conjecture claims that the plus part is asymptotically constant. These conjectures are backed up by Schoof's extensive calculations [Sch03] in the prime conductor case, and by the above explicit values. In particular, under GRH, Miller proved Weber's conjecture up to $m = 512$, and we note that according to Schoof's table, $h_m^+ \leq \sqrt{m}$ holds for more than 96.6% of all prime conductors $m = p < 10000$.

Prime ideal classes generators. When picking a set of prime ideals in the algorithms of this paper, an important feature is that they generate the class group. In general, even assuming GRH, only a large bound on the norm of generators is known, indeed Bach proved [Bac90, Th. 4] that $\mathcal{N}(\mathfrak{L}_{\max}) \leq 12 \ln^2 |\Delta_{K_m}|$, where $\mathfrak{L}_{\max}$ is the biggest ideal inside a generating set of Cl_m of minimum norm. In practice though, this bound seems very pessimistic [BDF08, §6].

On the other hand, as prime ideals belong to Cl_m^- only with probability roughly $1/h_m^+$, searching for generators of the *subgroup* Cl_m^- mechanically increases the provable upper bound on generators. More precisely, writing as $\mathfrak{L}^-_{\max}$ the biggest ideal of a generating set of Cl^-_m, Wesolowski proved [Wes18, Rem. 2] that $\mathcal{N}(\mathfrak{L}^-_{\max}) \leq \big(2.71 h^+_m \cdot \ln|\Delta_{K_m}| + 4.13\big)^2$.

Finally, we use the notation $h_{m,(\mathfrak{L}_1,\ldots,\mathfrak{L}_k)}$ to denote the cardinal of the subgroup of Cl_m generated by the k classes $\big[\mathfrak{L}_i\big]$, i.e., the determinant of the kernel of

$$\mathfrak{f}_{\mathfrak{L}_1,\ldots,\mathfrak{L}_k} : \big(e_1,\ldots,e_k\big) \in \mathbb{Z}^k \longmapsto \prod_{1\leq i\leq k} \big[\mathfrak{L}_i\big]^{e_i} \in \mathrm{Cl}_m\,.$$

2.3 Logarithmic $\mathcal{S}$-embeddings

We briefly introduce log-$\mathcal{S}$-unit lattices and discuss proper normalization by the Product Formula that was at the heart of the practical improvements of [BR20] compared to [PHS19].

Places of the cyclotomic field K_m are usually split into two parts: the set $\mathcal{S}_\infty$ of *infinite* places can be identified with the (complex) embeddings of K_m into $\mathbb{C}$, up to conjugation; the set $\mathcal{S}_0$ of *finite* places is specified by the infinite set of prime ideals of K_m, each prime ideal $\mathfrak{p}$ inducing an embedding of K_m into its $\mathfrak{p}$-adic completion $K_{m,\mathfrak{p}}$. Hence, any place $v \in \mathcal{S}_\infty \cup \mathcal{S}_0$ induces an absolute value $|\cdot|_v$ on K_m, and Ostrowski's theorem for number fields [Nar04, Th. 3.3] shows that all possible absolute values on K_m are obtained in this way. Concretely, for $\alpha \in K_m$: $\forall\sigma \in \mathcal{S}_\infty$, $|\alpha|_\sigma = \big|\sigma(\alpha)\big|$ and $\forall\mathfrak{p} \in \mathcal{S}_0$, $|\alpha|_\mathfrak{p} = p^{-v_\mathfrak{p}(\alpha)}$, where $v_\mathfrak{p}(\cdot)$ is the valuation of α at $\mathfrak{p}$ and $\langle p\rangle = \mathfrak{p}\cap\mathbb{Z}$. A remarkable fact is that all these absolute values are tied by the *Product Formula* [Nar04, Th. 3.5]:

$$\forall\alpha \in K_m, \qquad \prod_{v\in\mathcal{S}_\infty\cup\mathcal{S}_0} |\alpha|_v^{[K_{m,v}:\mathbb{Q}_v]} = 1. \tag{2.1}$$

The $\mathcal{S}_\infty$-part of this product is $\big|\mathcal{N}(\alpha)\big|$, as for $\sigma \in \mathcal{S}_\infty$, $K_{m,\sigma} = \mathbb{C}$ and $\mathbb{Q}_\sigma = \mathbb{R}$, so that $[K_{m,\sigma} : \mathbb{Q}_\sigma] = 2$. Similarly, for $\mathfrak{p} \in \mathcal{S}_0$, we have $|\alpha|_\mathfrak{p}^{[K_{m,\mathfrak{p}}:\mathbb{Q}_p]} = \mathcal{N}(\mathfrak{p})^{-v_\mathfrak{p}(\alpha)}$.

$\mathcal{S}$-unit group structure. Fix a finite set $\mathcal{S}$ of places; in this paper we shall consider that $\mathcal{S}$ *always* contains $\mathcal{S}_\infty$. The so-called $\mathcal{S}$-unit group of K_m, denoted by $\mathcal{O}^\times_{K_m,\mathcal{S}}$, is the multiplicative subgroup of K_m generated by all elements whose valuations are non zero only at the finite places of $\mathcal{S}$. Formally:

$$\mathcal{O}^\times_{K_m,\mathcal{S}} = \Big\{\alpha \in K_m;\ \langle\alpha\rangle = \prod_{\mathfrak{p}\in\mathcal{S}\cap\mathcal{S}_0} \mathfrak{p}^{v_\mathfrak{p}(\alpha)}\Big\}.$$

Theorem 2.1 (Dirichlet-Chevalley-Hasse [Nar04, Th. III.3.12, Cor. 1]). *The $\mathcal{S}$-unit group is the direct product of the group of roots of unity $\mu\big(\mathcal{O}^\times_{K_m}\big)$ and a free abelian group with $|\mathcal{S}| - 1$ generators. There exists a fundamental system of $\mathcal{S}$-units $\varepsilon_1,\ldots,\varepsilon_{|\mathcal{S}|-1}$ such that any $\varepsilon \in \mathcal{O}^\times_{K_m,\mathcal{S}}$ is uniquely written as: $\varepsilon = \mu\cdot\prod_{i=1}^{|\mathcal{S}|-1}\varepsilon_i^{k_i}$, where $\mu \in \big\langle\pm\zeta_m\big\rangle$ is a root of unity and $k_i \in \mathbb{Z}$.*

Log-$\mathcal{S}$-unit lattice. A fundamental ingredient of the proof of this theorem is to build an embedding of $\mathcal{O}_{K_m,\mathcal{S}}^{\times}$ into the real space of dimension $|\mathcal{S}|$, whose kernel is $\mu\big(\mathcal{O}_{K_m}^{\times}\big)$ and whose image is a lattice of dimension $\big(|\mathcal{S}|-1\big)$. This embedding is called the *logarithmic $\mathcal{S}$-embedding*, and its image is called the log-$\mathcal{S}$-unit lattice.

Several equivalent definitions of this logarithmic $\mathcal{S}$-embedding are acceptable for the proof. However, for cryptanalytic purposes, experimental evidence [BR20] suggests that it is crucial to use a properly normalized embedding for the decodability of the log-$\mathcal{S}$-unit lattice. Thus, we define [Nar04, §3, p.98]:

$$\operatorname{Log}_{\mathcal{S}} \alpha = \big([K_{m,v}:\mathbb{Q}_v]\cdot\ln|\alpha|_v\big)_{v\in\mathcal{S}} = \Big(\big\{2\ln|\sigma(\alpha)|\big\}_{\sigma\in\mathcal{S}_\infty}, \big\{-v_{\mathfrak{p}}(\alpha)\ln\mathcal{N}(\mathfrak{p})\big\}_{\mathfrak{p}\in\mathcal{S}\cap\mathcal{S}_0}\Big).$$

From the definition of $\mathcal{O}_{K_m,\mathcal{S}}^{\times}$, it is easy to see that $\mathbb{R}\otimes\operatorname{Log}_{\mathcal{S}}\mathcal{O}_{K_m,\mathcal{S}}^{\times}$ is included in the hyperplane orthogonal to $\mathbf{1}_{|\mathcal{S}|}$. Showing that its dimension is at least $|\mathcal{S}|-1$ is more involved.

A basis of the log-$\mathcal{S}$-unit lattice is given by the images $\operatorname{Log}_{\mathcal{S}}\varepsilon_i$ of the fundamental system of $\mathcal{S}$-units of Th. 2.1, as in [BR20, Eq. (2.7)]. Actually, we shall use later that for any maximal set of independent $\mathcal{S}$-units, their images under any logarithmic $\mathcal{S}$-embedding form a full rank sublattice of the corresponding log-$\mathcal{S}$-unit lattice. Its volume is given by [BR20, Pr. 2.2 and Eq. (2.8)].

As mentioned in [PHS19,BDPW20,BR20], a convenient trick in the context of the cryptanalysis of id-Svp is to consider an *expanded* version of the logarithmic $\mathcal{S}$-embedding, halving and repeating twice $\mathcal{S}_\infty$-coordinates, namely:

$$\overline{\operatorname{Log}}_{\mathcal{S}}\alpha = \Big(\big\{\ln|\sigma(\alpha)|, \ln|\sigma(\alpha)|\big\}_{\sigma\in\mathcal{S}_\infty}, \big\{[K_{m,\mathfrak{p}}:\mathbb{Q}_p]\cdot\ln|\alpha|_{\mathfrak{p}}\big\}_{\mathfrak{p}\in\mathcal{S}\cap\mathcal{S}_0}\Big).$$

In particular, this reduces the volume of the log-$\mathcal{S}$-unit lattice, as shown by [BR20, Pr. 2.3]. In practice though, we did not observe any fundamental difference between the approximation factors obtained using $\operatorname{Log}_{\mathcal{S}}$ or $\overline{\operatorname{Log}}_{\mathcal{S}}$.

2.4 Hard problems in Number Theory

One of the most difficult classical steps of the Approx-id-Svp algorithms proposed in [CDW17,PHS19,BR20,CDW21] is to find a solution to the ClDl defined as:

Problem 2.2 (Class Group Discrete Logarithm (ClDL)). Given a basis of prime ideals $\big\{\mathfrak{L}_1,\ldots,\mathfrak{L}_k\big\}$, and a challenge ideal $\mathfrak{b}$, find $\alpha\in K_m$ and integers $e_1,\ldots,e_k$ such that $\langle\alpha\rangle = \mathfrak{b}\cdot\prod_i\mathfrak{L}_i^{e_i}$, if this decomposition exists.

In this definition, we also ask for an *explicit* element α of the field, contrary to the definition of, e.g., [CDW17, Pr. 2]. Nevertheless, we note that in both quantum and classical worlds, the standard way to solve this problem boils down to computing $\mathcal{S}$-units, for $\mathcal{S}$ containing $\mathfrak{b}$ and the $\mathfrak{L}_i$'s, so that this explicit element is a byproduct of the resolution. Furthermore, put in this form it encompasses the well-known *Principal Ideal Problem* (Pip), using an empty set of ideals.

The *Shortest Generator Problem* (Sgp) asks, from a generator α of a principal ideal, for the shortest generator α' such that $\langle\alpha\rangle = \langle\alpha'\rangle$. Similarly, we define:

Problem 2.3 (Shortest Class Group Discrete Logarithm (S-ClDL)). Given a solution $\langle\alpha\rangle = \mathfrak{b} \cdot \prod_i \mathfrak{L}_i^{e_i}$ to the ClDL problem, find $w_1, \ldots, w_k \in \mathbb{Z}_{\geq 0}$ and $\alpha' \in K_m$ such that $\langle\alpha'\rangle = \mathfrak{b} \cdot \prod_i \mathfrak{L}_i^{w_i}$ and α' is the smallest possible one.

The condition for the w_i's to be positive is crucial. Note that all recent algorithms for Approx-id-SVP that are not bound to principal ideals eventually output an approximate solution of the S-ClDL [CDW21,PHS19,BR20]. If the set of prime ideals is sufficiently large compared to $\mathfrak{b}$, then S-ClDL is exactly id-SVP.

We also mention the Close Principal Multiple (CPM) problem which, given an ideal $\mathfrak{b}$, asks to find $\mathfrak{c}$ such that $\mathfrak{bc}$ is principal and $\mathcal{N}(\mathfrak{c})$ is small. This specific problem is used in [CDW21], and the authors prove that under GRH and using a factor base containing all prime ideals of norm up to $m^{4+o(1)}$, there exists a solution $\mathfrak{c}$ with $\mathcal{N}(\mathfrak{c}) \leq \exp\bigl(\widetilde{O}(m^{1+o(1)})\bigr)$ [CDW21, §1.3.4].

Complexities. As shown in [BS16], class groups, unit groups, class group discrete logarithms and principal ideal generator computations can be reduced to $\mathcal{S}$-unit groups computations for appropriate sets of places $\mathcal{S}$. Denote by $\mathrm{T}_{\mathcal{S}}(K_m)$ the running time of the computation of the $\mathcal{S}$-unit group in K_m. Under GRH, in a quantum setting, $\mathrm{T}_{\mathcal{S}}(K_m) = \mathrm{poly}\bigl(\ln|\Delta_{K_m}|, \bigl|\mathcal{S}\bigr|, \max_{\mathfrak{p}\in\mathcal{S}} \ln \mathcal{N}(\mathfrak{p})\bigr)$ by [EHKS14,BS16]. In a classical setting, $\mathrm{T}_{\mathcal{S}}(K_m) = \mathrm{poly}\bigl(\bigl|\mathcal{S}\bigr|, \max_{\mathfrak{p}\in\mathcal{S}} \ln \mathcal{N}(\mathfrak{p})\bigr) \cdot \exp \widetilde{O}\bigl(\ln^{2/3}(|\Delta_K|\bigr)$ is mainly subexponential in the degree of the cyclotomic field K_m [BF14,PHS19]. The exponent can be lowered to $1/2$ when m is a prime power [BEF$^+$17].

2.5 Lattices

Let L be a Euclidean lattice of full rank n. The first minimum $\lambda_1(L)$ of L is defined as the ℓ_2-norm of the smallest vector $\mathbf{v} \in L^*$, and the ℓ_2-distance from $\mathbf{t}$ to L, for any $\mathbf{t}$ in the span $L\otimes\mathbb{R}$ of L, is defined by $\mathrm{dist}_2(L, \mathbf{t}) = \min_{\mathbf{v}\in L} \|\mathbf{t} - \mathbf{v}\|_2$.

The *Approximate Shortest Vector Problem* (Approx-SVP) is, given a lattice L and an approximation factor af, to find $\mathbf{v} \in L$ such that $\|\mathbf{v}\|_2 \leq \mathrm{af} \cdot \lambda_1(L)$. Similarly, the *Approximate Closest Vector Problem* (Approx-CVP) asks, given a lattice L, an approximation factor af and a target $\mathbf{t}$ in the span $L \otimes \mathbb{R}$ of L, for a vector $\mathbf{v} \in L$ such that $\|\mathbf{t} - \mathbf{v}\|_2 \leq \mathrm{af} \cdot \mathrm{dist}_2(L, \mathbf{t})$. A practical Approx-CVP oracle is given by Babai's Nearest Plane algorithm [Bab86].

Bounding approximation factors. An ideal lattice of K_m is the full-rank image under the Minkowski embedding in $\mathbb{R}^{\varphi(m)}$ of a fractional ideal $\mathfrak{b}$ of K_m. Unlike generic lattices, a lower bound of the first minimum is implied by the arithmetic-geometric mean inequality, using that for any $b \in \mathfrak{b}$, $\mathcal{N}(\mathfrak{b})$ divides $|\mathcal{N}(b)|$. Thus:

$$\sqrt{n} \cdot \mathcal{N}(\mathfrak{b})^{1/n} \leq \lambda_1(\mathfrak{b}) \leq \sqrt{n} \cdot \mathcal{N}(\mathfrak{b})^{1/n} \sqrt{|\Delta_{K_m}|}^{1/n}, \qquad (2.2)$$

where $n = \varphi(m) = \deg K_m$ and the right inequality is Minkowski's inequality. Actually, applying the Gaussian Heuristic to ideal lattices would give that on average, $\lambda_1(\mathfrak{b}) \approx \sqrt{\frac{n}{2\pi e}} \cdot \mathrm{Vol}^{1/n}(\mathfrak{b})$, where $\mathrm{Vol}(\mathfrak{b}) = \mathcal{N}(\mathfrak{b})\sqrt{|\Delta_{K_m}|}$. This hypothesis is commonly used for the analysis of cryptosystems based on structured lattices, and we note that the *exact* approximation factors reached by the Twisted-PHS algorithm in [BR20] match this heuristic.

For any $\mathbf{x} \in \mathfrak{b}$, let $\mathrm{af}(\mathbf{x}) = \|\mathbf{x}\|_2 / \lambda_1(\mathfrak{b})$ denote the approximation factor reached by $\mathbf{x}$ for the Svp in the ideal lattice $\mathfrak{b}$. In general, $\lambda_1(\mathfrak{b})$ is not known, but Eq. (2.2) imply the bounds $\mathrm{af}_{\mathsf{inf}}(\mathbf{x}) \leq \mathrm{af}(\mathbf{x}) \approx \mathrm{af}_{\mathsf{gh}}(\mathbf{x}) \leq \mathrm{af}_{\mathsf{sup}}(\mathbf{x})$, where:

$$\begin{gathered}\mathrm{af}_{\mathsf{inf}}(\mathbf{x}) := \frac{\|\mathbf{x}\|_2}{\sqrt{n} \cdot \mathrm{Vol}^{1/n}(\mathfrak{b})}, \qquad \mathrm{af}_{\mathsf{sup}}(\mathbf{x}) := \frac{\|\mathbf{x}\|_2}{\sqrt{n} \cdot \mathcal{N}(\mathfrak{b})^{1/n}}, \\ \mathrm{af}_{\mathsf{gh}}(\mathbf{x}) := \sqrt{2\pi e} \cdot \mathrm{af}_{\mathsf{inf}}(\mathbf{x}).\end{gathered} \tag{2.3}$$

Quality of a lattice basis. Several indicators have been used in the literature to attempt to measure the quality of a lattice basis $B = (\mathbf{b}_1, \ldots, \mathbf{b}_n)$ relatively to the Svp or the Cvp. We will focus on the following three standard quantities:

1. the root-Hermite Factor $\delta_0(B)$, defined by $\delta_0^n(B) = \|\mathbf{b}_1\|_2 / \mathrm{Vol}^{1/n} B$, is commonly used to compare lattice reduction algorithms like LLL [LLL82] or BKZ [CN11]. On average, LLL reaches $\delta_0 \approx 1.022$ [GN08] whereas BKZ with blocksize $b \geq 50$ heuristically yields $\delta_0 \approx \left(\frac{b}{2\pi e}(\pi b)^{1/b}\right)^{1/(2b-2)}$ [Che13].
2. the (normalized) orthogonality defect $\delta(B)$, given by $\delta^n(B) = \prod_i \left(\frac{\|\mathbf{b}_i\|_2}{\mathrm{Vol}^{1/n} B}\right)$ [MG02, Def. 7.5] involves all vectors of the basis. By Minkowski's second theorem, its smallest possible value is upper bounded by $\sqrt{1 + \frac{n}{4}}$.
3. the logarithms of the norms of *Gram-Schmidt Orthogonalization* (GSO) vectors $\mathbf{b}_i^\star$ give also valuable information. For example, a rapid decrease in the sequence $\ln\|\mathbf{b}_i^\star\|_2$ at $i \geq 2$ indicates that $\mathbf{b}_i$ is rather not orthogonal to the previously generated subspace $\langle \mathbf{b}_1, \ldots, \mathbf{b}_{i-1} \rangle$.

3 An explicit full-rank family of independent $\mathcal{S}$-units

In this section, we exhibit a full rank family of *independent* $\mathcal{S}$-units, where the finite places $\mathcal{S}$ correspond to a collection of full Galois orbits of split prime ideals. As mentioned in introduction, this family is composed of three parts:

1. Circular units are recalled in §3.1 using the material from [Kuč92, Th. 6.1];
2. Stickelberger generators are in §3.2, sticking to the exposition of [BK21];
3. Real $\mathcal{S}^+$-units (apart from real units), where $\mathcal{S}^+$ is the set $\mathcal{S} \cap K_m^+$ of places of $\mathcal{S}$ restricted to K_m^+, are in §3.3.

Considering real $\mathcal{S}^+$-units and proving in §3.4 the multiplicative index of our family in the full $\mathcal{S}$-unit group constitute our main theoretical contributions. Finally, the saturation process used to mitigate this index is described in §3.5.

Remark 3.1. Recall that m has prime factorization $m = q_1 q_2 \cdots q_t \not\equiv 2 \mod 4$, where $q_i = p_i^{e_i} > 2$ for $i \in [\![1, t]\!]$. In the rest of the section, we will use subsets M_m^+ and M_m' of $[\![1, m]\!]$ that are useful to describe resp. a fundamental family of circular units and a short $\mathbb{Z}$-basis of the Stickelberger ideal of K_m. Their precise definitions from resp. [Kuč92, p.293] and [BK21, Eq. (11)] can be found in §A.1.

3.1 Circular units

Circular units are sometimes called *cyclotomic units* in the litterature, as in [Was97, §8]. We prefer to use the historical terminology from algebraic number theory, see e.g. Sinnott [Sin78, §4] and Kučera [Kuč92, §2], in order to avoid any confusion with the whole unit group $\mathcal{O}_{K_m}^\times$ of the m-th cyclotomic field.

Definition 3.2 (Circular units [Was97, §8.1]). *Let V_m be the multiplicative subgroup of $K_m^\times$ generated by $\left\{1-\zeta_m^a;\ 1 \leq a \leq m\right\}$. The group of* circular units *is the intersection $C_m := V_m \cap \mathcal{O}_{K_m}^\times$.*

Note that V_m contains the torsion of K_m, since $-\zeta_m = \left(1-\zeta_m\right)/\left(1-\zeta_m^{-1}\right)$. The circular units form a subgroup of $\mathcal{O}_{K_m}^\times$ of finite index, more precisely:

Proposition 3.3 ([Sin78, Th. p.107]). *The index of C_m in $\mathcal{O}_{K_m}^\times$ is finite:*

$$\left[\mathcal{O}_{K_m}^\times : C_m\right] = 2^b \cdot h_m^+, \qquad \text{with } b = \begin{cases} 0 & \text{if } t = 1, \\ 2^{t-2}+1-t & \text{otherwise,} \end{cases}$$

where t is the number of distinct prime divisors of m.

Hence, circular units provide a very large subgroup of $\mathcal{O}_{K_m}^\times$: indeed, the real part of the class number is expected to be small (§2.2), and the other factor *generically* grows linearly in m (see [HW38, Th. 430 and 431] for a precise statement).

An explicit system of fundamental circular units for any m has been given in [GK89] and independently in [Kuč92, Th. 6.1]. More precisely, for $0 < a < m$, define the following special circular units, where $m_i = m/p_i^{e_i}$ [Kuč92, p.176]:

$$v_a = \begin{cases} 1-\zeta_m^a & \text{if } \forall i \in [\![1,t]\!],\ m_i \nmid a, \\ \dfrac{1-\zeta_m^a}{1-\zeta_m^{m_i}} & \text{otherwise, for the unique } m_i \mid a. \end{cases} \tag{3.1}$$

Theorem 3.4 ([Kuč92, Th. 6.1]). *Recall $M_m^+ \subsetneq [\![1,m]\!]$ is defined in §A.1. The set $\left\{v_a;\ a \in M_m^+\right\}$ is a system of fundamental circular units of K_m: for any circular unit $\eta \in C_m$, there exist a uniquely determined map $k : M_m^+ \to \mathbb{Z}$ and root of unity $\mu \in \left\langle \pm\zeta_m \right\rangle$ s.t. $\eta = \mu \cdot \prod_{a \in M_m^+} v_a^{k(a)}$.*

A crucial point for the cryptanalysis of id-Svp in [CDW21] is that the logarithmic embedding of these elements is short. Namely, explicitly writing the constants that appear in the proof of [CDW21, Lem. 3.5], we have, for any $0 < a < m$, that $\|\mathrm{Log}_{\mathcal{S}_\infty}(1-\zeta_m^a)\|_2 \leq 1.32 \cdot \sqrt{m}$.

3.2 Stickelberger generators

In this section, we use [BK21, Th. 3.1] to describe a short *basis* of the so-called Stickelberger ideal, viewed as a $\mathbb{Z}$-module. These Stickelberger short relations correspond to principal ideals whose generators are surprisingly easy to compute using Jacobi sums as in [BK21, §6]. Following Sinnott [Sin80], for all $a \in \mathbb{Z}$, let:

$$\theta_m(a) = \sum_{s \in (\mathbb{Z}/m\mathbb{Z})^\times} \left\{ -\frac{as}{m} \right\} \cdot \sigma_s^{-1} \in \mathbb{Q}\big[G_m\big], \tag{3.2}$$

and let N_m be the absolute norm element $N_m = \sum_{\sigma \in G_m} \sigma$. It is easy to check that $a \equiv b \mod m$ implies $\theta_m(a) = \theta_m(b)$ and that $\theta_m(a) + \theta_m(-a) = N_m$ whenever $m \nmid a$.

Definition 3.5 (Stickelberger ideal [Sin80, p.189]). *Let $\mathcal{S}'_m$ be the $\mathbb{Z}$-module of $\mathbb{Q}\big[G_m\big]$ generated by $\big\{\theta_m(a);\ 0 < a < m\big\} \cup \big\{\frac{1}{2}N_m\big\}$. The* Stickelberger ideal *of K_m is the intersection $\mathcal{S}_m = \mathcal{S}'_m \cap \mathbb{Z}\big[G_m\big]$.*

As in [CDW21], we shall refer to the *Stickelberger lattice* when $\mathcal{S}_m$ is viewed as a $\mathbb{Z}$-module. Note that in some references, like in [Was97, §6.2], the Stickelberger ideal is defined as the smaller ideal $\mathbb{Z}\big[G_m\big] \cap \theta_m(-1)\mathbb{Z}\big[G_m\big]$, which coincides with Def. 3.5 if and only if m is a prime power [Kuč86, Pr. 4.3].

Theorem 3.6 (Stickelberger's theorem [Sin80, Th. 3.1]). *The Stickelberger ideal $\mathcal{S}_m$ of K_m annihilates the class group of K_m. Hence, for any ideal $\mathfrak{b}$ of K_m and any $\alpha = \sum_{\sigma \in G_m} a_\sigma \sigma \in \mathcal{S}_m$, the ideal $\mathfrak{b}^\alpha = \prod_{\sigma \in G_m} \sigma(\mathfrak{b})^{a_\sigma}$ is principal.*

An outstanding point is that the proof of this important result is completely explicit, i.e., for any $\alpha \in \mathcal{S}_m$, and any fractional ideal $\mathfrak{b}$ of K_m, an explicit $\gamma \in K_m$ s.t. $\langle \gamma \rangle = \mathfrak{b}^\alpha$ is constructed. It appears that when α is a short element of $\mathcal{S}_m$, this explicit generator is very efficiently computable.

A short basis of the Stickelberger lattice. An element of the integral group ring $\mathbb{Z}\big[G_m\big]$ is called *short* if it is of the form $\sum_{\sigma \in G_m} a_\sigma \sigma \in \mathbb{Z}\big[G_m\big]$, where $a_\sigma \in \{0, 1\}$ for all $\sigma \in G_m$. Short elements of $\mathcal{S}_m$ have been identified in [Sch08, Th. 9.3(i) and Ex. 9.3] in the prime conductor case, and the proof has been adapted to any conductor in [CDW21, Lem. 4.4] to prove the shortness of the following generating set of $\mathcal{S}_m$:

$$W = \big\{w_a;\ a \in [\![2, m]\!]\big\}, \quad \text{with } w_a = \theta_m(1) + \theta_m(a-1) - \theta_m(a). \tag{3.3}$$

Note that using $\theta_m(a) + \theta_m(-a) = N_m$ when $m \nmid a$, we obtain $w_a = w_{m-a+1}$ whenever $1 < a < m$, and that $w_m = N_m$ using also $\theta_m(m) = 0$. Hence, W is the set $\big\{w_a;\ 2 \le a \le \big\lceil \frac{m}{2} \big\rceil\big\} \cup \big\{N_m\big\}$.

We emphasize that only knowing a generating set of short elements as in [CDW21] is not necessarily sufficient. Though it would be possible to build a basis from this generating set to solve the Cvp like in [CDW21, Cor. 2.2], without any geometric loss using e.g. [MG02, Lem. 7.1], we observed that the slight euclidean norm growth of the obtained basis vectors translates into a dramatic increase of the size of the (possibly rational) coefficients of the corresponding generators, in a way that significantly hinders subsequent computations. In particular, in order to climb dimensions as far as possible and best approach log-$\mathcal{S}$-unit lattices using the saturation process described in §3.5, it is crucial to constrain both the number of elements we use and their size, i.e., to use a *basis* of the Stickelberger lattice containing only *short* elements. In [BK21], a very large family of short elements [BK21, Pr. 3.1] encompassing $W \setminus \{N_m\}$ is made explicit:

Proposition 3.7 ([BK21, Pr. 3.1]). *Let a, $b \in \mathbb{Z}$ satisfying $m \nmid a$, $m \nmid b$ and $m \nmid (a+b)$. Then $\alpha = \theta_m(a) + \theta_m(b) - \theta_m(a+b)$ is a short element of $\mathcal{S}_m$. Moreover, $(1+\tau)\cdot\alpha = N_m$, so exactly one half of the coefficients of α are zeros.*

Note that the second part of the proposition actually specifies [CDW21, Lem. 4.4(3)]: for any $w \in W \setminus \{N_m\}$, it implies that the ℓ_2-norm of w, viewed as a vector in $\mathbb{Z}^{\varphi(m)} \simeq_{\mathbb{Z}} \mathbb{Z}[G_m]$, is *exactly* $\sqrt{\varphi(m)/2}$. Then, from this family, a short basis is computationally easy to extract:

Theorem 3.8 ([BK21, Th. 3.6]). *Recall $M'_m \subsetneq [\![1, m]\!]$ is defined in §A.1. There exists an efficiently computable map $\alpha_m(\cdot)$ from $[\![1, m]\!]$ to the family of short elements of $\mathcal{S}_m$ described in Pr. 3.7, s.t. $\left\{\alpha_m(c);\ c \in M'_m\right\} \cup \left\{N_m\right\}$ is a $\mathbb{Z}$-basis of the Stickelberger lattice $\mathcal{S}_m$ of K_m having only short elements.*

The explicit definition of $\alpha_m(\cdot)$ can be found in [BK21, §3.2], and is included for completeness in §A.2. We stress that when m is a prime, this basis coincides with the one given by [Sch08, Th. 9.3(i)] and with the set W in Eq. (3.3).

Effective Stickelberger generators using Jacobi sums. As previously mentioned, the proof of Th. 3.6 is explicit, i.e., for any $\alpha \in \mathcal{S}_m$ and any fractional ideal $\mathfrak{b}$ of K_m, it builds an explicit $\gamma \in K_m$ such that $\langle\gamma\rangle = \mathfrak{b}^\alpha$ [Was97, §6.2], [Sin80, §3.1]. Moreover, when α is a short basis element from Th. 3.8, it turns out that γ has a simple expression using Jacobi sums [BK21, §5].

We briefly treat the split case here. Let $\ell \in \mathbb{Z}$ be a prime such that $\ell \equiv 1 \mod m$, and let $\mathfrak{L}$ be any fixed (split) prime ideal of K_m above ℓ. Let a, b be such as in Pr. 3.7, then for $\alpha = \theta_m(a) + \theta_m(b) - \theta_m(a+b)$, we have that $\mathfrak{L}^\alpha$ is a principal ideal generated by the following Jacobi sum [BK21, Pr. 5.1]:

$$\mathcal{J}_{\mathfrak{L}}(a,b) = -\sum_{u \in \mathcal{O}_{K_m}/\mathfrak{L}} \chi_{\mathfrak{L}}^{a}(u)\chi_{\mathfrak{L}}^{b}(1-u) \quad \in K_m, \tag{3.4}$$

where $\chi_{\mathfrak{L}}(u) \in \left\langle\zeta_m\right\rangle$ verifies $\chi_{\mathfrak{L}}(u) \equiv u^{(\ell-1)/m} \mod \mathfrak{L}$, for any $u \in \left(\mathcal{O}_{K_m}/\mathfrak{L}\right)^\times$, and $\chi_{\mathfrak{L}}(0) = 0$. When $\alpha = \alpha_m(c)$ for $c \in M'_m$, we shall write $\gamma_{\bar{\mathfrak{L}},c}$ for the generator of $\mathfrak{L}^{\alpha_m(c)}$. Using a discrete logarithm table for elements of $\mathcal{O}_{K_m}/\mathfrak{L}^\times$, the computation, for a fixed prime $\mathfrak{L}$, of all Jacobi sums corresponding to the short basis $\left\{\alpha_m(c);\ c \in M'_m\right\}$ is very fast. As noted in [BK21, §5], the Galois group also acts on the involved Jacobi sums in a way that allows to replace some of the Jacobi sum computations by the application of a suitable automorphism.

Finally, as a direct consequence of [Was97, Lem. 6.1], all these Jacobi sums are ℓ-Weil numbers, i.e., they verify the Weil relation $\mathcal{J}_{\mathfrak{L}}(a,b)\overline{\mathcal{J}_{\mathfrak{L}}(a,b)} = \ell$, for a and b as above. This implies $\left|\sigma\big(\mathcal{J}_{\mathfrak{L}}(a,b)\big)\right| = \sqrt{\ell}$ for all $\sigma \in G_m$, meaning that any of these elements is *the shortest* generator of its corresponding $\mathfrak{L}^\alpha$.

3.3 Real $\mathcal{S}^+$-units

A consequence of Th. 3.8, since $\left|M'_m\right| = \frac{\varphi(m)}{2}$, is that the Stickelberger lattice only has rank $\frac{\varphi(m)}{2}+1$ in $\mathbb{Z}\left[G_m\right]$; in particular, it is not full rank, hence cannot be directly used as a lattice of class relations. In previous works, obtaining a full rank lattice in $\mathbb{Z}[G_m]$ from $\mathcal{S}_m$ was done by projecting into $(1-\tau)\mathcal{S}_m$ [CDW21, §4.3], or by the adjonction of $(1+\tau)\mathbb{Z}[G_m]$ [CDW17, Def. 2]. Both can be used as a lattice of class relations for the *relative* class group Cl_m^-. In particular, the so-called *augmented* Stickelberger lattice $\mathcal{S}_m+(1+\tau)\mathbb{Z}[G_m]$ annihilates the relative class group and has full rank in $\mathbb{Z}[G_m]$, as shown in [CDW17, Lem. 2].

We generalize this result by considering the module of all real class group relations between relative norm ideals of ideals from the entire class group Cl_m. In §3.4, we shall prove that the Stickelberger lattice augmented with these real class group relations yields a lattice of class relations for the *whole* class group. Note that, as opposed to other modules like $(1-\tau)\mathcal{S}_m$ or $\mathcal{S}_m+(1+\tau)\mathbb{Z}\left[G_m\right]$, real class group relations actually depend on the underlying prime ideals.

On one hand, this affects negatively the shortness of the obtained relation vectors: putting those in Hermite Normal Form, we shall see later that each relation, viewed as a vector of integer valuations, has ℓ_2-norm at most h_m^+. On the other hand, removing the constraint to belong to the relative class group brings a significant practical and theoretical gap: first, it allows to choose prime ideals of smallest possible norms, which as shown in [BR20, §3.3] or [CDW21, Th. 4.8] lowers in practice the obtained approximation factor; second, whereas prime ideals of norm at most Bach's bound are sufficient to generate the entire class group, prime generators for the *relative* class group are only proven to be of norm bounded by the *larger* bound $(2.71\cdot h_m^+\cdot\ln\Delta_{K_m}+4.13)^2$ from [Wes18].

Lifting real class group relations. Let $\ell_1,\ldots,\ell_d$ be distinct prime integers satisfying $\ell_i\equiv 1 \mod m$, so that ℓ_i is split in K_m, for all i in $[\![1,d]\!]$. For each i, fix a prime ideal $\mathfrak{L}_i \mid \ell_i$ in K_m of norm ℓ_i, and let $\mathfrak{l}_i=\mathcal{N}_{K_m/K_m^+}\left(\mathfrak{L}_i\right)=\mathfrak{L}_i^{1+\tau}\cap K_m^+$ be the relative norm ideal of $\mathfrak{L}_i$. Since $\mathfrak{L}_i$ is a split prime ideal of K_m dividing ℓ_i, the ideal $\mathfrak{l}_i$ is a split prime ideal of K_m^+ of norm ℓ_i, and by Kummer-Dedekind's theorem we have $\mathfrak{l}_i\cdot\mathcal{O}_{K_m}=\mathfrak{L}_i^{1+\tau}$. This justifies the slight abuse of notation of writing $\mathfrak{l}_i^\sigma=\mathfrak{L}_i^{(1+\tau)\sigma}\cap K_m^+$, for any $\sigma\in G_m$.

We are interested in the real class group relations between all prime ideals in the G_m^+-orbits of the $\mathfrak{l}_i$, i.e., between the following prime ideals of K_m^+:

$$\left\{\mathfrak{l}_i^{\sigma_s};\ i\in[\![1,d]\!], 0<s<\tfrac{m}{2}, (s,m)=1\right\}. \tag{3.5}$$

The important point is, any class relation in K_m^+ between ideals from Eq. (3.5) translates to a class relation in K_m using repeatedly $\mathfrak{l}_i^\sigma\cdot\mathcal{O}_{K_m}=\mathfrak{L}_i^{(1+\tau)\sigma}$. More precisely, let $\left(r_1,\ldots,r_d\right)\in\mathbb{Z}\left[G_m^+\right]^d$ represent a real class relation in K_m^+ between ideals $\left\{\mathfrak{l}_i^{\sigma_s}\right\}$ of Eq. (3.5), i.e., there exists $\gamma_r^+\in K_m^+$ such that $\gamma_r^+\cdot\mathcal{O}_{K_m^+}=\prod_{i=1}^d\mathfrak{l}_i^{r_i}$. Then, this relation lifts naturally to a class relation $\left((1+\tau)\cdot r_1,\ldots,(1+\tau)\cdot r_d\right)$ in K_m between prime ideals in the G_m-orbits $\left\{\mathfrak{L}_i^\sigma;\ i\in[\![1,d]\!],\sigma\in G_m\right\}$ as:

$$\gamma_r^+\cdot\mathcal{O}_{K_m}=\prod_{i=1}^d\mathfrak{L}_i^{(1+\tau)r_i}. \tag{3.6}$$

Let $C^+_{\mathfrak{l}_1,\ldots,\mathfrak{l}_d}$ denote the lattice of class relations between elements of all G^+_m-orbits of $\{\mathfrak{l}_i;\ i \in [\![1,d]\!]\}$. Concretely, it is the kernel of the following map:

$$\mathfrak{f}_{\mathfrak{l}_1,\ldots,\mathfrak{l}_d} : \big(r_{i,s}\big)_{\substack{1\leq i\leq d,\\ 0<s<m/2,(s,m)=1}} \in \mathbb{Z}^{d\cdot\frac{\varphi(m)}{2}} \longmapsto \prod_{i,s} \big[\mathfrak{l}_i^{\sigma_s}\big]^{r_{i,s}} \in \mathrm{Cl}^+_m\,. \tag{3.7}$$

Using the canonical isomorphism of $\mathbb{Z}$-modules $\mathbb{Z}^{d\cdot\frac{\varphi(m)}{2}} \simeq_{\mathbb{Z}} \mathbb{Z}[G^+_m]^d$, the lattice of class relations $C^+_{\mathfrak{l}_1,\ldots,\mathfrak{l}_d}$ may be viewed as a $\mathbb{Z}$-submodule of $\mathbb{Z}[G^+_m]^d$. Lifting all these relations back to K_m as in Eq. (3.6), we therefore obtain the submodule $(1+\tau)\cdot C^+_{\mathfrak{l}_1,\ldots,\mathfrak{l}_d} \subseteq (1+\tau)\mathbb{Z}[G_m]^d$, that we shall call the lattice of *real class relations* between the G_m-orbits of $\{\mathfrak{L}_i;\ i \in [\![1,d]\!]\}$.

Remark 3.9. When $h^+_m = 1$, $C^+_{\mathfrak{l}_1,\ldots,\mathfrak{l}_d}$ is isomorphic to d copies of the integral group ring $\mathbb{Z}[G^+_m]$ and the lattice of real class relations is simply $(1+\tau)\mathbb{Z}[G_m]^d$.

Euclidean norm of real class relations. We now identify a real class group relation from $C^+_{\mathfrak{l}_1,\ldots,\mathfrak{l}_d}$ to a vector in $\mathbb{Z}^{d\cdot\frac{\varphi(m)}{2}}$. In other words, we consider only the valuations of these relations on the G^+_m-orbits of the prime ideals $\mathfrak{l}_1,\ldots,\mathfrak{l}_d$. Furthermore, $C^+_{\mathfrak{l}_1,\ldots,\mathfrak{l}_d}$ is put in Hermite Normal Form, conveniently for the proof, but better bounds might easily be obtained using e.g. the LLL algorithm.

Proposition 3.10. *Suppose the lattice $C^+_{\mathfrak{l}_1,\ldots,\mathfrak{l}_d}$ of real class relations is in HNF. Then, for all $\mathbf{w} \in C^+_{\mathfrak{l}_1,\ldots,\mathfrak{l}_d} \subseteq \mathbb{Z}[G^+_m]^d$, we have $\|\mathbf{w}\|_2 \leq \|\mathbf{w}\|_1 \leq h^+_m$.*

This means that $(1+\tau)\cdot C^+_{\mathfrak{l}_1,\ldots,\mathfrak{l}_d}$ can be used in the CDW algorithm instead of $(1+\tau)\mathbb{Z}[G_m]$, as we will see in §4, while still reaching the same asymptotic approximation factor as long as $h^+_m \leq O\big(\sqrt{\varphi(m)}\big)$. This slightly more restrictive hypothesis (see the discussion in §2.2) will be more than compensated by the fact that it removes the need for the $\mathfrak{l}_i$'s to be principal, which has a significant impact in practice on the algebraic norm of the chosen ideals, and thus on the final approximation factor reached in [CDW21, Alg. 6].

Proof. The image of the map $\mathfrak{f}_{\mathfrak{l}_1,\ldots,\mathfrak{l}_d}$ given in Eq. (3.7) is a subgroup of Cl^+_m, so the volume of its kernel $C^+_{\mathfrak{l}_1,\ldots,\mathfrak{l}_d}$ is at most h^+_m. By definition of the Hermite Normal Form,[6] $C^+_{\mathfrak{l}_1,\ldots,\mathfrak{l}_d}$ has diagonal elements $h_1,\ldots,h_{\varphi(m)/2} > 0$, and the j-th column contains integers c_{ij} such that $0 \leq c_{ij} < h_j$ for $i < j$ and $c_{ij} = 0$ for $i > j$. We shall prove $h_i + \sum_{i<j} c_{ij} \leq h_i \cdot \prod_{i<j} h_j$ for any row of fixed index $i \in [\![1, \frac{\varphi(m)}{2}]\!]$, which yields the result. This is done by induction on the dimension, using repeatedly the fact that for any integers $x, y \geq 1$, $x + (y-1) \leq (xy)$. □

Explicit real generators. For each relation $r = \big(r_1,\ldots,r_d\big) \in C^+_{\mathfrak{l}_1,\ldots,\mathfrak{l}_d}$, we compute an explicit $\gamma^+_r \in K^+_m \subsetneq K_m$ that verifies Eq. (3.6). Together with the unit group $\mathcal{O}^\times_{K^+_m}$ of K^+_m, they form a fundamental system of $\mathcal{S}^+$-units, where the finite places of $\mathcal{S}^+$ are the G^+_m-orbits of the relative norm ideals $\mathfrak{l}_i$.

[6] In this proof, we consider an upper-triangular HNF with row vectors.

In the next section, we shall see that adding the explicit Stickelberger generators of §3.2 to these real generators yields a maximal set of independent $\mathcal{S}$-units in the degree $\varphi(m)$ cyclotomic field K_m, at the much smaller cost of computing a fundamental system of real $\mathcal{S}^+$-units in K_m^+ of degree only $\frac{\varphi(m)}{2}$.

In practice, though this remains the main bottleneck of our experimental setting, it allows us to push effectively our experiments up to degree $\varphi(m) = 210$, whereas the (full) $\mathcal{S}$-units computations of [BR20] were bound to $\varphi(m) = 70$.

3.4 A $\mathcal{S}$-unit subgroup of finite index

As in §3.3, let $\ell_1, \dots, \ell_d$ be prime integers satisfying $\ell_i \equiv 1 \mod m$; for each i, fix a (split) prime ideal $\mathfrak{L}_i \mid \ell_i$ in K_m and let $\mathfrak{l}_i = \mathfrak{L}_i \cap K_m^+$. Let $\mathcal{S}$ be a set of places containing, apart the infinite places of K_m, all G_m-orbits of the $\mathfrak{L}_i$'s. Combining the results of §3.1, §3.2 and §3.3, we get the following family of $\mathcal{S}$-units:

$$\mathfrak{F} = \left\{v_a;\ a \in M_m^+\right\} \cup \left\{\gamma^-_{\mathfrak{L}_i,b};\ i \in [\![1,d]\!], b \in M'_m\right\} \cup \left\{\gamma^+_r;\ r \in C^+_{\mathfrak{l}_1,\dots,\mathfrak{l}_d}\right\} \tag{3.8}$$

where the first set is the set of *circular units* given by Th. 3.4, the second is the set of explicit *Stickelberger generators* stated at the end of §3.2 and the last one is the set of *real generators* as in Eq. (3.6).

This family has $\big(\varphi(m)/2-1\big)+d\cdot\varphi(m)$ elements, which matches precisely the multiplicative rank of the full $\mathcal{S}$-unit group modulo torsion $\mathcal{O}^\times_{K_m,\mathcal{S}}/\mu\big(\mathcal{O}^\times_{K_m}\big)$.[7] In this section, we prove that these $\mathcal{S}$-units are indeed independent and we compute the index of the subgroup of $\mathcal{O}^\times_{K_m,\mathcal{S}}$ generated by those elements.

Theorem 3.11. *Let $h_{m,(\mathfrak{L}_1,\dots,\mathfrak{L}_d)}$ (resp. $h^+_{m,(\mathfrak{l}_1,\dots,\mathfrak{l}_d)}$) be the cardinal of the subgroup of Cl_m (resp. Cl^+_m) generated by the G_m-orbits of $\mathfrak{L}_1, \dots, \mathfrak{L}_d$ (resp. the G^+_m-orbits of $\mathfrak{l}_1, \dots, \mathfrak{l}_d$). The family $\mathfrak{F}$ given in Eq. (3.8) is a maximal set of independent $\mathcal{S}$-units. The subgroup generated by $\mathfrak{F}$ in $\mathcal{O}^\times_{K_m,\mathcal{S}}/\mu\big(\mathcal{O}^\times_{K_m}\big)$ has index:*

$$\left(\frac{h_m \cdot h^+_{m,(\mathfrak{l}_1,\dots,\mathfrak{l}_d)}}{h_{m,(\mathfrak{L}_1,\dots,\mathfrak{L}_d)}}\right) \cdot 2^b \cdot \left(h^-_m\right)^{d-1} \cdot \left(2^{\frac{\varphi(m)}{2}-1} \cdot 2^a\right)^d,$$

where $a = b = 0$ if m is a prime power, and $a = 2^{t-2} - 1$, $b = 2^{t-2} + 1 - t$ whenever m has t distinct prime divisors.

Note that when the G_m-orbits of the $\mathfrak{L}_i$'s generate Cl_m, the first term in this index equals h^+_m. As we shall see in §3.5, the powers of 2 can be killed by standard saturation techniques, so the real problem comes from the $(h^-_m)^{d-1}$ part, which has generically *huge* prime factors. Intuitively, this comes from the fact that the Stickelberger relations miss all class group relations that exist between two (or more) distinct G_m-orbits.

First, we show that the lattice obtained by adding one copy of the Stickelberger ideal per G_m-orbit, to the lattice $(1+\tau)\cdot C^+_{\mathfrak{l}_1,\dots,\mathfrak{l}_d}$ of real class relations,

[7] Note that for our purpose, the torsion units play no role and can thus be put aside.

yields a full-rank submodule of $\mathbb{Z}[G_m]^d$. Hence, we have obtained a full-rank lattice of class relations for the union of all G_m-orbits above $\ell_1, \dots, \ell_d$.

We begin by restricting our attention to the case $d = 1$. We need the following lemma, which extends and proves an observation already made in [DPW19, Rem. 3] in the prime conductor case:

Lemma 3.12. *The index of $\mathcal{S}_m + (1+\tau)\cdot\mathbb{Z}[G_m^+]$ in $\mathbb{Z}[G_m]$ is finite:*

$$\left[\mathbb{Z}[G_m] : \mathcal{S}_m + (1+\tau)\cdot\mathbb{Z}[G_m^+]\right] = 2^{\varphi(m)/2-1}\cdot 2^a \cdot h_m^-,$$

where $a = 0$ if $t = 1$ and $a = 2^{t-2} - 1$ else, where m has t prime divisors.

Proof. The proof is due to R. Kučera. First, note that $(1+\tau)\cdot\mathbb{Z}[G_m^+]$ contains N_m, hence by Th. 3.8, $\mathcal{S}_m + (1+\tau)\cdot\mathbb{Z}[G_m^+]$ is generated by the following $\varphi(m)$ elements:

$$\left\{\alpha_m(b);\ b \in M'_m\right\} \cup \left\{(1+\tau)\sigma_s;\ 0 < s < \tfrac{m}{2}, (s,m) = 1\right\}.$$

Therefore, its index is given by the absolute value of the determinant of the transition matrix from the canonical basis of $\mathbb{Z}[G_m]$ to the above generating set:

$$\left[\mathbb{Z}[G_m] : \mathcal{S}_m + (1+\tau)\cdot\mathbb{Z}[G_m^+]\right] = \left|\det\left(\begin{array}{c} \left\{a_{b,s}\right\}_{\substack{b\in M'_m \\ 0<s<m,\, (s,m)=1}} \\ \hline \begin{matrix} & & 1 & 1 & & \\ & \iddots & & & \ddots & \\ 1 & & & & & 1 \end{matrix} \end{array}\right)\right|,$$

where for any $b \in M'_m$, we write $\alpha_m(b) = \sum_{\sigma_s\in G_m} a_{b,s}\sigma_s$. Subtracting suitable combinations of rows of the lower half of this matrix to rows of the upper half to cancel the upper right block, this is the absolute value of the determinant of the square matrix of dimension $\frac{\varphi(m)}{2}$ with coefficients $\left\{a_{b,s} - a_{b,-s}\right\}$, for $b \in M'_m$ and s prime with m such that $0 < s < \frac{m}{2}$. By Pr. 3.7, $a_{b,s} + a_{b,-s} = 1$, which implies that $a_{b,s} - a_{b,-s} = 2a_{b,s} - 1$, therefore we recognize the matrix appearing at the very end of the proof of [BK21, Cor. 4.1] with each coefficient being multiplied by 2. Combining this with [BK21, Eq. (26)], we obtain:

$$\left[\mathbb{Z}[G_m] : \mathcal{S}_m + (1+\tau)\cdot\mathbb{Z}[G_m^+]\right] = 2^{\frac{\varphi(m)}{2}}\cdot\tfrac{1}{2}\left[\mathcal{R}_m^- : \mathcal{S}_m^-\right],$$

and the result follows from Sinnott's theorem [Sin78, Th. p.107]. □

When $h_m^+ = 1$, the lattice of real class relations is always $(1+\tau)\cdot\mathbb{Z}[G_m^+]$, and Lem. 3.12 gives the whole story. In the general case $h_m^+ \neq 1$, we deduce:

Lemma 3.13. *Let ℓ be a prime integer that splits in K_m, let $\mathfrak{L} \mid \ell$ in K_m and let $\mathfrak{l} = \mathfrak{L}^{1+\tau} \cap K_m^+$. Let $h_{m,(\mathfrak{l})}^+$ be the cardinal of the subgroup of Cl_m^+ generated by the G_m^+-orbit of $\mathfrak{l}$ in K_m^+. The $\mathbb{Z}$-module generated by $\mathcal{S}_m$ and the lattice $(1+\tau)\cdot C_{\mathfrak{l}}^+$ of real class relations of the G_m-orbit of $\mathfrak{L}$, has finite index in $\mathbb{Z}[G_m]$:*

$$\left[\mathbb{Z}[G_m] : \mathcal{S}_m + (1+\tau)\cdot C_{\mathfrak{l}}^+\right] = 2^{\varphi(m)/2-1}\cdot 2^a\cdot h_m^-\cdot h_{m,(\mathfrak{l})}^+,$$

where $a = 0$ if $t = 1$ and $a = 2^{t-2} - 1$ else, where m has t prime divisors.

Proof. By definition of $C_{\mathfrak{l}}^+$ as the kernel of the map $\mathfrak{f}_{\mathfrak{l}}$ of Eq. (3.7), we have:

$$\big[\mathbb{Z}[G_m^+] : C_{\mathfrak{l}}^+\big] = h_{m,(\mathfrak{l})}^+ = \big[(1+\tau)\cdot\mathbb{Z}[G_m^+] : (1+\tau)\cdot C_{\mathfrak{l}}^+\big].$$

Note also that N_m belongs to $(1+\tau)\cdot C_{\mathfrak{l}}^+ \subseteq (1+\tau)\cdot\mathbb{Z}[G_m^+]$, hence, again by means of transition matrix:

$$\big[\mathcal{S}_m + (1+\tau)\cdot\mathbb{Z}[G_m^+] : \mathcal{S}_m + (1+\tau)\cdot C_{\mathfrak{l}}^+\big] = \big[(1+\tau)\cdot\mathbb{Z}[G_m^+] : (1+\tau)\cdot C_{\mathfrak{l}}^+\big].$$

Finally, putting things together with Lem. 3.12, the result comes from:

$$\begin{aligned}\big[\mathbb{Z}[G_m] : \mathcal{S}_m + (1+\tau)\cdot C_{\mathfrak{l}}^+\big] &= \big[\mathbb{Z}[G_m] : \mathcal{S}_m + (1+\tau)\cdot\mathbb{Z}[G_m^+]\big] \\ &\quad\cdot \big[\mathcal{S}_m + (1+\tau)\cdot\mathbb{Z}[G_m^+] : \mathcal{S}_m + (1+\tau)\cdot C_{\mathfrak{l}}^+\big] \\ &= \big(2^{\varphi(m)/2-1}\cdot 2^a\cdot h_m^-\big)\cdot\big[\mathbb{Z}[G_m^+] : C_{\mathfrak{l}}^+\big].\end{aligned}$$ □

Finally, for the case where there are $d \geq 1$ orbits, a reasoning very similar to the proofs of Lem. 3.12 and 3.13 leads to:

Proposition 3.14. *Let $h_{m,(\mathfrak{l}_1,\ldots,\mathfrak{l}_d)}^+$ be the cardinal of the subgroup of Cl_m^+ generated by all G_m^+-orbits of $\mathfrak{l}_1,\ldots,\mathfrak{l}_d$. Then, the $\mathbb{Z}$-module generated by the lattice $(1+\tau)\cdot C_{\mathfrak{l}_1,\ldots,\mathfrak{l}_d}^+ \subseteq (1+\tau)\cdot\mathbb{Z}[G_m^+]^d$ of real class relations between the G_m-orbits of the $\mathfrak{L}_i$'s, and the diagonal block matrix of d copies of $\big(\mathcal{S}_m \setminus N_m\mathbb{Z}\big)$, verifies:*

$$\big[\mathbb{Z}[G_m]^d : \mathcal{S}_m^d + (1+\tau)\cdot C_{\mathfrak{l}_1,\ldots,\mathfrak{l}_d}^+\big] = \big(2^{\varphi(m)/2-1}\cdot 2^a\cdot h_m^-\big)^d\cdot h_{m,(\mathfrak{l}_1,\ldots,\mathfrak{l}_d)}^+.$$

Proof of Th. 3.11. The independence comes from Pr. 3.14 and the trivial fact that circular units are independent from Stickelberger and real generators. The index of the subgroup generated by $\mathfrak{F}$ in $\mathcal{O}_{K_m,\mathcal{S}}^\times/\mu\big(\mathcal{O}_{K_m}^\times\big)$ is given by:

$$\big[\mathcal{O}_{K_m}^\times : C_m\big]\cdot\frac{\big[\mathbb{Z}[G_m]^d : \mathcal{S}_m^d + (1+\tau)\cdot C_{\mathfrak{l}_1,\ldots,\mathfrak{l}_d}^+\big]}{\big|\det\big(\ker\mathfrak{f}_{\mathcal{S}}\big)\big|},$$

where $\ker\mathfrak{f}_{\mathcal{S}}$ is the lattice of all class group relations between finite places of $\mathcal{S}$. The first term is given by Pr. 3.3, the numerator of the second term is given by Pr. 3.14, and by definition of $\mathcal{O}_{K_m,\mathcal{S}}^\times$, the denominator is precisely $h_{m,(\mathfrak{L}_1,\ldots,\mathfrak{L}_d)}$. Rearranging terms adequately yields the result. □

3.5 Saturation

Saturation is a standard tool of computational algebraic number theory that has been used in various contexts like unit and class group computations, and can be traced back at least to [PZ89, §5.7].

Intuitively, the e-saturation procedure applied to $\mathfrak{F}$ consists in detecting e-th powers in the subgroup generated by $\mathfrak{F}$, including their e-th roots in the set, using e.g. the generalized Montgomery's e-th-root algorithm from [Tho12, §3], and rebuilding a basis of multiplicatively independent elements. At the end, the index of the new basis is no longer divisible by e. Remark that the output size does not depend on e, but only on the number and size of the elements of $\mathfrak{F}$.

As the index given by Th. 3.11 is divisible by a large power of 2, it is therefore natural to 2-saturate $\mathfrak{F}$ in order to mitigate its exponential growth, obtaining the 2-saturated family $\mathfrak{F}_{\mathsf{sat}}$. However, as the relative class number h_m^- in the index of Th. 3.11 hides *huge* prime factors, we stress that this strategy is at first glance hopeless in general to obtain the full $\mathcal{S}$-unit group from $\mathfrak{F}$.

In the following, we briefly describe the 2-saturation procedure we use, and refer to e.g. [BFHP21, §4.3] for a formal exposition.

Recognizing squares. Let $U = \langle g_1, \dots, g_k \rangle$ be a finitely generated multiplicative subgroup of $\mathcal{O}_{K_m,\mathcal{S}}^\times$. The first step of the 2-saturation process is to recognize squares in $U \cap (\mathcal{O}_{K_m,\mathcal{S}}^\times)^2$. This is done by using local information provided by quadratic characters.

Fix a prime $\mathfrak{p} \notin \mathcal{S}$ such that $\mathcal{N}(\mathfrak{p}) \equiv 1 \mod \operatorname{lcm}(m, 2)$. Define $\chi_\mathfrak{p}$ as the Legendre symbol such that $\chi_\mathfrak{p}(a) \equiv a^{(\mathcal{N}(\mathfrak{p})-1)/2} \mod \mathfrak{p}$ for any $a \in U$. As $\mathfrak{p} \notin \mathcal{S}$ and $a \in \mathcal{O}_{K_m,\mathcal{S}}^\times$, we have $\chi_\mathfrak{p}(a) \in \{-1, 1\}$. If a is a square, $\chi_\mathfrak{p}(a) = 1$ as a is still a square modulo $\mathfrak{p}$. The converse is not true, but by considering many characters $\chi_{\mathfrak{p}_1}, \dots, \chi_{\mathfrak{p}_N}$ as above, it is expected that at least one of them evaluates to -1. Hence, recognizing squares boils down to compute the kernel of:

$$\begin{aligned} \log_{-1,\chi} : U &\longrightarrow \mathbb{F}_2^N \\ a &\longmapsto \left\{ \log_{-1} \chi_{\mathfrak{p}_i}(a);\ i \in [\![1, N]\!] \right\}. \end{aligned}$$

An element of this kernel is still not guaranteed to be a square. Nevertheless, a standard heuristic, first stated in the context of integer factorization [BLP93, §8] and also used in multiquadratic fields [BBV$^+$17, §4.2], [BV18, H. 4.3], is to assume that if the $\mathfrak{p}_i$ are all distinct (split) prime ideals, then the $\log_{-1} \chi_{\mathfrak{p}_i}$ behave as independent uniform random elements of $\operatorname{Hom}\big(U/\big(U \cap (K_m^\times)^2\big), \mathbb{F}_2\big)$. Concretely, this means that these should span this dual with probability at least $\left(1 - 1/2^{N-k}\right)$ [BLP93, Lem. 8.2]; in that case, any element of the kernel of $\log_{-1,\chi}$ is indeed a square. In other words, if $\sum_{1 \le i \le k} v_i \log_{1,\chi} g_i = 0$, then with high probability the product $g = \prod_{1 \le i \le k} g_i^{v_i}$ indeed belongs to $U \cap (\mathcal{O}_{K_m,\mathcal{S}}^\times)^2$.

Square roots algorithm. Once we have identified combinations of elements of U that are $\mathcal{S}$-unit squares, it remains to compute their square roots explicitly.

First, we note that it is useful to systematically reduce those products modulo all squared circular units C_m^2 to contain the coefficients size. This is done as usual by projecting the logarithmic embedding $\operatorname{Log}_{\mathcal{S}_\infty} g$ of the obtained $g \in (\mathcal{O}_{K_m,\mathcal{S}}^\times)^2$ into $2 \cdot \operatorname{Log}_{\mathcal{S}_\infty} C_m$, finding a closest vector $y = \operatorname{Log}_{\mathcal{S}_\infty} u^2$ and replacing g by g/u^2.

The traditional method to compute the square root of an element $g \in (K_m^\times)^2$ is to factor the polynomial $x^2 - g$ in $K_m[x]$, using e.g. Trager's method [Coh93, Alg. 3.6.4] or Belabas' p-adic method [Bel04]. As, according to Th. 3.11, we have many square roots to compute, we choose instead to use a batch strategy in the spirit of [LPS20, Alg. 5] using complex embeddings approximations.

Since LLL seminal paper [LLL82], it is known that one can retrieve an algebraic number from approximations of one of its complex embeddings. Indeed, fix an embedding $\sigma \in G_m$ and a $\mathbb{Q}$-basis $\big(\omega_1, \ldots, \omega_n\big)$ of $\mathcal{O}_{K_m}$, and LLL-reduce:

$$B_\kappa := \begin{pmatrix} -\sigma(\omega_1) & C & 0 & \ldots & 0 \\ -\sigma(\omega_2) & 0 & C & \ddots & \vdots \\ \vdots & \vdots & \ddots & \ddots & 0 \\ -\sigma(\omega_n) & 0 & \ldots & 0 & C \end{pmatrix}.$$

where $C > 0$ is a constant and approximations are computed at precision $\kappa \in \mathbb{N}$. Then, for any $g \in \mathcal{O}_{K_m}$, applying e.g. Babai's Nearest Plane algorithm on the LLL basis of B_κ and target $\big(\sigma(g), 0, \ldots, 0\big)$ gives a combination $(g_1, \ldots, g_n)$ such that $g = \sum_{i=1}^{n} g_i \omega_i$. As explained in [LPS20], it is possible to mutualize the computation of B_κ and reuse the unitary transformation to hasten computations when increasing κ is required.

We use an improvement that benefits from the existence of the maximal real subfield K_m^+. Each $g \in K_m = K_m^+[\zeta_m]$ can be uniquely written as $g = g_0 + g_1 \cdot \zeta_m$, with $g_0, g_1 \in K_m^+$. For $\sigma \in G_m^+$, the *relative Minkowski embedding* of σ relatively to the extension K_m/K_m^+ is defined by $\sigma_{K_m/K_m^+}(g_0^\sigma, g_1^\sigma) = \big(g^\sigma, \overline{g^\sigma}\big) \in \mathbb{C}^2$. This is a linear homomorphism of $\mathbb{C}^2$. When $g = h^2$, its square root $h_0 + h_1\zeta_m$ can be retrieved from approximations of h_0^σ and h_1^σ instead of h^σ, as follows:

1. Compute $\sigma_{K_m/K_m^+}(g_0^\sigma, g_1^\sigma) = \big(g^\sigma, \overline{g^\sigma}\big) \in \mathbb{C}^2$;
2. Choose one complex square root z of g^σ and apply $\sigma_{K_m/K_m^+}^{-1}$ to $(z, \overline{z})$ to get potential approximations $\big(\tilde{h}_0^\sigma, \tilde{h}_1^\sigma\big)$ of h_0^σ and h_1^σ respectively;
3. Using LLL as above in K_m^+ on $\tilde{h}_0^\sigma$ and $\tilde{h}_1^\sigma$, obtain $\big(\tilde{h}_0, \tilde{h}_1\big)$ in K_m^+, which are candidates for resp. h_0 and h_1.
4. If $(\tilde{h}_0 + \tilde{h}_1 \cdot \zeta_m)^2 \neq g$, then increase κ using the fast method of [LPS20].

Hence, this method amounts to LLL reducing a matrix of size $\frac{n}{2} \times (\frac{n}{2} + 1)$ and decoding using e.g. Babai's Nearest Plane algorithm. This offers a great speed-up compared to reducing a $n \times (n+1)$ matrix. For further details and generalizations to higher order polynomial roots, we refer the interested reader to [Les21].

Rebuilding a basis. After the square root step, we obtain new elements $h_1, \ldots, h_r$, where $r = \dim\big(\ker \log_{-1,\chi}\big)$. To extract a set of k independent elements from the extended set $\big\{h_1, \ldots, h_r, g_1, \ldots, g_k\big\}$, we compute an LLL-basis of the matrix constituted of their valuations at the places of $\mathcal{S}$. Note that this matrix can be computed entirely from the valuations of the initial set $\{g_i\}$ and the basis of $\ker \log_{-1,\chi}$. Using the same trick as for matrix A in [BBV$^+$17, Alg. 5.2], this contains the height of the transformation matrix, sufficiently for our needs.

At the end of this process we obtain a maximal set of independent $\mathcal{S}$-units of index given by Th. 3.11 where no factor 2 remains.

4 Removing quantum steps from the CDW algorithm

The complete material for this section is given in §B, and the main points are summarized here. The CDW algorithm for solving Approx-Svp was introduced in [CDW17] for cyclotomic fields of prime power conductors, using short relations of the Stickelberger lattice as a keystone. [CDW21] extended it to all conductors.

In this section, we show how to use the results of §3.2, §3.3 and §3.4 to remove most quantum steps of [CDW21]. More precisely, we first propose in §B.2 an equivalent rewriting of [CDW21, Alg. 7] that enlightens some hidden steps that reveal useful for subsequent modifications. Then, in §B.3, we plug in the explicit generators of §3.2 ([BK21]) and Eq. (3.6) for relative class group orbits, to remove the last call to the quantum Pip solver. Finally, by considering the module of *all* real class group relations, using Pr. 3.14 and Th. 3.11, we remove in §B.4 the need of a random walk mapping any ideal of K_m into Cl_m^-, at the (small) additional price of restricting to cyclotomic fields such that $h_m^+ \leq O(\sqrt{m})$ (Hyp. B.1).

An equivalent rewriting of CDW (§B.2). Omitting details, the CDW algorithm works as follows, for any challenge ideal $\mathfrak{a}$ of K_m [CDW21, Alg. 7]:

1. Random walk to Cl_m^-: find $\mathfrak{b}$ such that $\left[\mathfrak{a}\mathfrak{b}\right] \in \mathrm{Cl}_m^-$.
2. Solve the ClDl of $\mathfrak{a}\mathfrak{b}$ on G_m-orbits of the prime ideals $\mathfrak{L}_1, \ldots, \mathfrak{L}_d$ of Cl_m^-. This gives a vector[8] $\epsilon = (\epsilon_1, \ldots, \epsilon_d) \in \mathbb{Z}[G_m]^d$ such that $\mathfrak{a}\mathfrak{b} \cdot \prod_i \mathfrak{L}_i^{\epsilon_i}$ is principal.
3. Solve the Cpmp by projecting each ϵ_i in $\pi(\mathcal{S}_m) = (1-\tau)\mathcal{S}_m$, find a close vector $v_i = y_i \cdot \pi(\mathcal{S}_m)$ and lift v_i to get some η_i s.t. $\pi(\eta_i) = v_i$, $\|\epsilon - \eta\|_1$ is small *with positive coordinates*, and $\mathfrak{a}\mathfrak{b} \cdot \prod_i \mathfrak{L}_i^{\epsilon_i - \eta_i}$ is principal.
4. Apply the Pip algorithm of [BS16] to get a generator of this principal ideal.
5. Reduce the obtained generator by circular units like in [CDPR16].

This eventually outputs $h \in \mathfrak{a}$ of length $\|h\|_2 \leq \exp\big(\widetilde{O}(\sqrt{m})\big) \cdot \mathcal{N}(\mathfrak{a})^{1/\varphi(m)}$ [CDW21, Th. 5.1].

We focus on the lift procedure of Step 3. In [CDW21], $v \in \pi(\mathcal{S}_m)$ is lifted to $\eta \in \mathcal{S}_m$ with non-negative coordinates by setting $(\eta_\sigma, \eta_{\tau\sigma}) = (v_\sigma, 0)$ if $v_\sigma \geq 0$ and $(0, -v_\sigma)$ otherwise, for all $\sigma \in G_m^+$. This works because $[\mathfrak{c}]^{-1} = [\mathfrak{c}^\tau]$ for any $\mathfrak{c} \in \mathrm{Cl}_m^-$, but hides which exact product of relative norm ideals is involved. We propose a totally equivalent lift procedure: from $v = y \cdot \pi(\mathcal{S}_m)$, consider the preimage $\tilde{\eta} = y \cdot \mathcal{S}_m$, from which we remove $\min\big\{\tilde{\eta}_\sigma, \tilde{\eta}_{\tau\sigma}\big\}$ to each $\tilde{\eta}_\sigma$ coordinate to obtain η. Now, it is obvious that η is a combination y of relations in $\mathcal{S}_m$, and of relative norm relations given by the min part. Details are in Alg. B.6.

Using explicit Stickelberger generators (§B.3). Each element w_a of the generating set W of $\mathcal{S}_m$ corresponds to a generator $\mathcal{J}_{\mathfrak{L}}(1, a-1)$ (see §3.2). Similarly, each relative norm ideal writes $\langle \gamma_s^+ \rangle = \mathfrak{L}^{(1+\tau)\sigma_s}$ (see §3.3). Hence, from an (explicit) ClDl solution $\langle \alpha \rangle = \mathfrak{a}\mathfrak{b} \cdot \mathfrak{L}^{\epsilon}$, and given a Cpmp solution, explicitly written as above as $\eta = y \cdot W + u \cdot (1+\tau) \cdot \mathbb{Z}[G_m^+]$, we have that a generator of $\mathfrak{a}\mathfrak{b} \cdot \mathfrak{L}^{\epsilon - \eta}$ is directly given by $\alpha / \big(\prod_a \mathcal{J}_{\mathfrak{L}}(1, a-1)^{y_a} \prod_s (\gamma_s^+)^{u_s}\big)$. This allows us to remove the quantum Pip in dimension n in step 4 (for each query). In exchange, we need to compute (only once) all real generators for relative norm relations, which can be done in dimension $\varphi(m)/2$ by [BS16, Alg. 2].

[8] In the CDW algorithm, the explicit generator given by the ClDl solver is discarded.

Avoiding the random walk (§B.4). Finally, note that several quantum steps are performed (for each query) in the random walk that maps ideals to Cl_m^-. Using the results of §3.3, we replace the module $(1+\tau)\cdot\mathbb{Z}[G_m]^d$ by the module of all real class group relations. Asymptotically, we prove in Pr. B.7 that this does not change the bound on the approximation factor obtained in [CDW21, Th. 5.1], under the same assumption on the Galois-module structure of Cl_m [CDW21, Ass. 1], as long as we restrict to fields K_m with $h_m^+ \leq O(\sqrt{m})$ (Hyp. B.1). This additional tiny assumption is largely compensated by the fact that only two quantum steps remain: one is performed only once in dimension $\varphi(m)/2$ to compute real class group relations and generators, and the second is solving the ClDl for each query (see Tab. B.1).

5 Computing log-$\mathcal{S}$-unit sublattices in higher dimension

Our main goal is to simulate the Twisted-PHS algorithm for high degree cyclotomic fields. To this end, we compute full-rank sublattices of the full log-$\mathcal{S}$-unit lattice using the knowledge of the maximal set $\mathfrak{F}$ of independent $\mathcal{S}$-units defined by Eq. (3.8) and its 2-saturated counterpart $\mathfrak{F}_{\mathsf{sat}}$ from §3.5. These sets are lifted from a complete set of real $\mathcal{S}^+$-units (see §3.3), hence are obtained at the classically subexponential cost of working in the half degree maximal real subfield. We note that by Th. 3.11, the index of these families grows rapidly as the number of orbits increases, hence these approximated modes give an upper bound on the approximation factors that can be expected when using Twisted-PHS.

The Twisted-PHS algorithm is briefly recalled in §5.1, and our experimental setting is detailed in §5.2. Then, we analyse in §5.3 the geometric characteristics of our log-$\mathcal{S}$-unit sublattices and the obtained approximation factors in §5.4.

5.1 The Twisted-PHS algorithm

The Twisted-PHS algorithm [BR20] was introduced as an improvement of the PHS algorithm [PHS19]. Both aim at solving Approx-id-Svp in any number field and have the same theoretically proven bounds for running time and reached approximation factors. However, the explicit $\mathcal{S}$-units formalism in [BR20] leads to a proper normalization of the used log-$\mathcal{S}$-embedding, weighting coordinates according to finite places norms. This turned out to give experimentally significant improvements on the lattices' decodability and on reached approximation factors.

Both algorithms are split in a *preprocessing phase*, performed only once for a fixed number field, and a *query phase*, for each challenge ideal. More precisely:

1. The preprocessing phase consists in choosing a set of finite places $\mathcal{S}$ generating the class group, computing the corresponding log-$\mathcal{S}$-unit lattice for an appropriate log-$\mathcal{S}$-embedding, and preparing the lattice for subsequent Approx-Cvp requests using the Laarhoven's algorithm from [Laa16];
2. For each challenge ideal $\mathfrak{b}$, the query phase consists in first solving the ClDl relatively to $\mathcal{S}$, obtaining $\langle\alpha\rangle = \mathfrak{b}\cdot\prod_{\mathfrak{L}\in\mathcal{S}}\mathfrak{L}^{v_\mathfrak{L}}$. Then, this element is projected onto the span of the above log-$\mathcal{S}$-unit lattice, and a close vector of this lattice

gives a $\mathcal{S}$-unit s s.t. α/s is hopefully small. Here, guaranteeing that $\alpha/s \in \mathfrak{b}$ is achieved by applying a drift parameterized by some β on the target.

In the Twisted-PHS case, since the obtained lattice, after proper normalization, appears to have exceptionally good geometric characteristics, it was proposed to replace Laarhoven's algorithm by a lazy BKZ reduction in the preprocessing phase and Babai's Nearest Plane algorithm in the query phase [BR20, Alg. 4.2 and 4.3]. We will consider only this practical version in our experiments.

In details, for a number field K, the log-$\mathcal{S}$-unit lattice used in the Twisted-PHS algorithm is defined as $\varphi_{\mathsf{tw}}(\mathcal{O}_{K,\mathcal{S}}^{\times})$, where φ_{tw} is the log-$\mathcal{S}$-embedding given by $f_H \circ \overline{\mathrm{Log}}_{\mathcal{S}}$ [BR20, Eq. (4.1)], for an isometry f_H from the span H of $\overline{\mathrm{Log}}_{\mathcal{S}}$ to $\mathbb{R}^k$, where k equals the multiplicative rank of $\mathcal{O}_{K,\mathcal{S}}^{\times}$ modulo torsion.

Among the consequences of the proper normalization induced by $\overline{\mathrm{Log}}_{\mathcal{S}}$, the authors showed how to optimally choose a set of finite places that generate the class group [BR20, Alg. 4.1]. Namely, taking ideals of increasing prime norms in the set $\mathcal{S}$, they noticed that the density of the associated (twisted) log-$\mathcal{S}$-unit lattice $\varphi_{\mathsf{tw}}(\mathcal{O}_{K,\mathcal{S}}^{\times})$ increases up to an optimal value before decreasing.

Finally, a tricky aspect of the resolution resides in guaranteeing that the output solution is indeed an element of the challenge ideal, i.e., that $v_{\mathfrak{L}}(\alpha/s) \geq 0$ for all $\mathfrak{L} \in \mathcal{S} \cap \mathcal{S}_0$. In [BR20], this is done by applying a drift vector in the span of the log-$\mathcal{S}$-unit lattice, parameterized by some β whose optimal value is searched using a dichotomic strategy in the query phase. Concretely [BR20, Eq. (4.7)]:

$$\mathbf{t} = f_H\left(\left\{\ln|\alpha|_\sigma - \frac{k\beta + \ln\mathcal{N}(\mathfrak{b}) - \sum_{\mathfrak{L}\in\mathcal{S}}\ln\mathcal{N}(\mathfrak{L})}{[K:\mathbb{Q}]}\right\}_\sigma, \left\{\ln|\alpha|_{\mathfrak{L}}^{[K_{\mathfrak{L}}:\mathbb{Q}_\ell]} + \beta - \ln\mathcal{N}(\mathfrak{L})\right\}_{\mathfrak{L}\in\mathcal{S}}\right).$$

5.2 Experimental settings

Computing the full group of $\mathcal{S}$-units in a classical way is rapidly intractable, even in the case of cyclotomic fields; therefore, experiments performed in [BR20] on Twisted-PHS were bound to $\varphi(m) \leq 70$. We apply the Twisted-PHS algorithm using our full-rank sublattices of the whole log-$\mathcal{S}$-unit lattice induced by the independent family $\mathfrak{F}$ of Eq. (3.8), its 2-saturated counterpart $\mathfrak{F}_{\mathsf{sat}}$ (§3.5) and, when feasible, a fundamental system $\mathfrak{F}_{\mathsf{su}}$ for the full $\mathcal{S}$-unit group. Approximated modes with $\mathfrak{F}$ or $\mathfrak{F}_{\mathsf{sat}}$ give a glimpse on how Twisted-PHS scales in higher dimensions, where asymptotic phenomena like the growth of h_m start to express.

Source code and hardware description. All experiments have been implemented using SageMath v9.0 [Sag20], except for the full $\mathcal{S}$-unit groups computations for which we used Magma [BCP97], which appears much faster for this particular task and also offers an indispensable product ("Raw") representation. Moreover, fplll [FpL16] was used to perform all lattice reduction algorithms. The entire source code is provided on `https://github.com/ob3rnard/Tw-Sti`.

Most of the computations were performed in less than two weeks on a server with 72 Intel® Xeon® E5-2695v4 @2.1GHz with 768GB of RAM, using 2TB of storage for the precomputations. Real class group computations were performed on a single Intel® Core™ i7-8650U @3.2GHz CPU using 10GB of RAM.

m	$\varphi(m)$	h_m^+	m	$\varphi(m)$	h_m^+	m	$\varphi(m)$	h_m^+	m	$\varphi(m)$	h_m^+	m	$\varphi(m)$	h_m^+	m	$\varphi(m)$	h_m^+
136	64	2	408	128	2	205	160	2	356	176	†	520	192	4	265	208	†
212	104	5	268	132	†	328	160	†	376	184	†	840	192	†	424	208	†
145	112	2	284	140	†	440	160	5	191	190	11	303	200	†	636	208	†
183	120	4	292	144	†	163	162	4	221	192	†	404	200	†			
248	120	4	504	144	4	332	164	†	388	192	†	309	204	†			
272	128	2	316	156	†	344	168	†	476	192	†	412	204	†			

Table 5.1 – List of ignored conductors (†: failure to compute Cl_m^+ within a day).

Targeted cyclotomic fields. We consider cyclotomic fields of *any* conductor m s.t. $20 < \varphi(m) \leq 210$ with known real class number $h_m^+ = 1$, including those from Tab. 2.1. The restriction to $h_m^+ = 1$ is only due to technical interface obstructions, i.e., we are not aware of how to access the non-trivial real class group relations internally computed by SageMath. Additionally, for some of the conductors, we were not able to obtain the real class group within a day. Thus, we are left with 210 distinct cyclotomics fields, and Tab. 5.1 lists all ignored conductors.

Finite places choice. The optimal set of places computed by [BR20, Alg. 4.1] yields a number $d_{\max}$ of split G_m-orbits of smallest norms maximizing the density of the corresponding full log-$\mathcal{S}$-unit lattice. However, the index of our log-$\mathcal{S}$-unit *sublattices*, given by Th. 3.11, grows too quickly, roughly in $(h_m^-)^{d-1}$, so that their density always decreases as soon as $d > 1$. This remark motivates us to compute all log-$\mathcal{S}$-unit sublattices for $d = 1$ to $d_{\max}$ first split G_m-orbits.

Full rank log-$\mathcal{S}$-unit sublattices. The first maximal set of independent $\mathcal{S}$-units that we consider is $\mathfrak{F}$ from Eq. (3.8). The 2-saturation process of §3.5 mitigates the huge index of $\mathfrak{F}$, yielding family $\mathfrak{F}_{\mathsf{sat}}$. A fundamental system $\mathfrak{F}_{\mathsf{su}}$ of the full $\mathcal{S}$-unit group $\mathcal{O}_{K_m,\mathcal{S}}^{\times}$ (modulo torsion) is also used whenever it is computable in reasonable time, i.e., up to $\varphi(m) < 80$. As noted in §2.3, their images under any log-$\mathcal{S}$-embedding φ form full-rank sublattices resp. L_{urs}, L_{sat}, L_{su}, generated by resp. $\varphi(\mathfrak{F})$, $\varphi(\mathfrak{F}_{\mathsf{sat}})$, $\varphi(\mathfrak{F}_{\mathsf{su}})$, of the corresponding full log-$\mathcal{S}$-unit lattice $\varphi(\mathcal{O}_{K_m,\mathcal{S}}^{\times})$.

We consider several choices of the log-$\mathcal{S}$-embedding φ. Namely, we tried to evaluate the advantage of using the expanded $\overline{\mathrm{Log}}_{\mathcal{S}}$ (exp) over $\mathrm{Log}_{\mathcal{S}}$, labelled tw (as twisted by $[\mathbb{C} : \mathbb{R}] = 2$). We also considered versions with (iso) or without (noiso) the isometry f_H of [BR20, Eq. (4.2)]. This yields four choices for φ, e.g. tag $\mathsf{noiso/tw}$ is $\varphi = \mathrm{Log}_{\mathcal{S}}$ and $\mathsf{iso/exp}$ gives the original $\varphi_{\mathsf{tw}} = f_H \circ \overline{\mathrm{Log}}_{\mathcal{S}}$.

Compact product representation. In order to avoid the exponential growth of algebraic integers viewed in $\mathbb{Z}[x]/\langle \Phi_m(x) \rangle$, we use a compact product representation, so that any element α in $\mathfrak{F}$ (resp. $\mathfrak{F}_{\mathsf{sat}}$ or $\mathfrak{F}_{\mathsf{su}}$) is written on a set $g_1, \ldots, g_N$ of N small elements as $\alpha = \prod_{i=1}^{N} g_i^{e_i}$. Hence, besides the g_i's, each α is stored as a vector $e \in \mathbb{Z}^N$, and for any choice of φ, we have $\varphi(\alpha) = \sum_{i=1}^{N} e_i \cdot \varphi(g_i)$. This allows us to compute φ without the coefficient explosion encountered in [BR20, §5], which unlocks the full log-$\mathcal{S}$-unit lattices computations beyond degree 60.

Lattice reductions. For each of the constructed log-$\mathcal{S}$-unit sublattices, i.e. for each number of orbits $d \in [\![1, d_{\max}]\!]$, for each family of independent $\mathcal{S}$-units $\mathfrak{F}$, $\mathfrak{F}_{\mathsf{sat}}$ and

m	d	set	k	$\mathrm{Vol}^{1/k}$	δ			$\max_{1\leq i\leq k}\|\mathbf{b}_i\|_2$		
					raw	LLL	bkz$_{40}$	raw	LLL	bkz$_{40}$
152	1	urs	107	8.691	2.016	1.570	1.551	45.007	38.466	38.202
		sat	107	6.928	4.398	1.787	1.822	752.306	23.280	21.720
		su	107	6.928	28.396	1.805	1.828	3163.723	21.953	21.446
	2	urs	179	9.683	2.157	1.623	1.590	48.754	41.313	41.404
		sat	179	7.384	7.670	1.885	1.896	6273.562	23.280	22.772
		su	179	6.816	65.355	2.226	2.322	3427.134	23.221	24.741
211	1	urs	314	14.325	2.672	2.291	2.257	96.068	97.930	96.569
		sat	314	11.386	9.998	2.581	2.562	9742.552	59.387	59.578
	5	urs	1154	18.232	3.118	2.542	2.497	118.124	119.160	115.888
		sat	1154	13.341	19.443	2.918	2.901	32067.612	71.428	72.752
	7	urs	1574	18.976	3.161	2.557	2.512	120.838	121.129	119.020
		sat	1574	13.771	26.841	2.927	2.910	530646.708	71.428	72.752

TABLE 5.2 – Geometric characteristics of L_{urs}, L_{sat} and L_{su} for $\mathbb{Q}(\zeta_{152})$ and $\mathbb{Q}(\zeta_{211})$ with log-$\mathcal{S}$-embedding φ_{tw} (of type iso/exp). For *all* bases, the root-Hermite factor verifies $|\delta_0 - 1| < 10^{-3}$.

(when feasible) $\mathfrak{F}_{\mathsf{su}}$, and for each choice of log-$\mathcal{S}$-embedding, we compare several levels of reduction: no reduction ("raw"), LLL-reduction and BKZ$_{40}$-reduction.

5.3 Geometry of the lattices

For all described choices of log-$\mathcal{S}$-unit sublattices, we first evaluate several geometrical parameters (see §2.5): reduced volume $V^{1/k}$, root-Hermite factor δ_0, orthogonality defect δ. For clarity's sake, we only give here a few examples giving a glimpse of what happens in general, and additional data can be found in §C.1.

Table 5.2 contains data for cyclotomic fields $\mathbb{Q}(\zeta_{152})$ and $\mathbb{Q}(\zeta_{211})$ of degrees resp. 72 and 210. All values correspond to the iso/exp log-$\mathcal{S}$-embedding, i.e., $\varphi = \varphi_{\mathsf{tw}}$. Indeed, as illustrated by Tab. C.2, we experimentally note that using (no)iso/exp seems geometrically slightly better than using (no)iso/tw. Notice how small is the normalized orthogonality defect after only LLL reduction, unambiguously below the tight Minkowski bound $\sqrt{1+\frac{k}{4}}$.

We then look at the logarithm of the Gram-Schmidt norms, for every described choice of log-$\mathcal{S}$-unit sublattices. Figure 5.1 plots the Gram-Schmidt log norms before and after BKZ reduction of the lattices L_{sat}, using the original iso/exp log-$\mathcal{S}$-embedding φ_{tw}. As in [BR20, Fig. B.1–10], for each field the two curves are almost superposed, which is consistent with the previous observations on the orthogonality defect. We also checked the impact of the log-$\mathcal{S}$-embedding choice among all four options on the Gram-Schmidt logarithm norms of the *unreduced* basis $\varphi(\mathfrak{F}_{\mathsf{sat}})$. As expected, the isometry f_H has absolutely no influence on the Gram-Schmidt norms. On the other hand, using $\mathrm{Log}_{\mathcal{S}}$ or $\overline{\mathrm{Log}}_{\mathcal{S}}$ seems to alter only the first norms, and in a very small way. This can be seen in Fig. C.4. Again, increasing the number of orbits does not influence these behaviours.

We stress that these very peculiar geometric characteristics – shape of the logarithm of the norms of the Gram-Schmidt basis, ease of reduction, very small

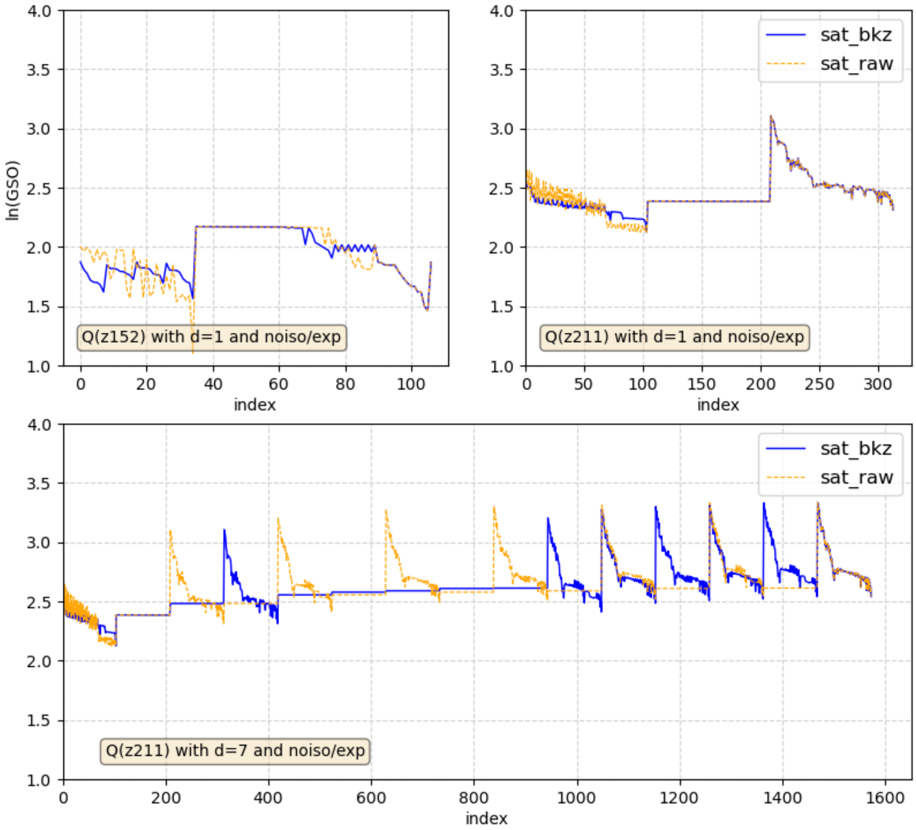

FIG. 5.1 – L_{sat} lattices for $\mathbb{Q}(\zeta_{152})$ and $\mathbb{Q}(\zeta_{211})$: Gram-Schmidt log norms before and after reduction by BKZ_{40}.

orthogonality defect (after LLL) – already observed in [BR20, §5.1–2], are consistently viewed across all conductors, degrees, log-$\mathcal{S}$-unit sublattices and number of orbits. To give a concrete idea of e.g. the striking ease of reduction of these log-$\mathcal{S}$-unit sublattices, we report that for $m = 211$, BKZ_{40} terminates in around 7 minutes (resp. 30 minutes) on the log-$\mathcal{S}$-unit sublattice of dimension $k = 1154$ (resp. 1574) corresponding to $d = 5$ (resp. $d_{\max} = 7$), which is unusually fast.

This very broad phenomenon suggests that the explanation is possibly deep, an observation that has been recently developed by Bernstein and Lange [BL21].

5.4 Evaluation of the approximation factor

In [BR20], evaluating in practice the approximation factors reached by Twisted-PHS is done by choosing random split ideals of prime norm, solving the ClDL for these challenges and comparing the length of the obtained algebraic integer with the length of the exact shortest element. As the degrees of the fields grow, solving the ClDL and exact id-SVP becomes rapidly intractable. Hence, we resort to simulating random outputs of the ClDL, similarly to [DPW19, Hyp. 8], and estimate the obtained approximation factors with inequalities from Eq. (2.3).

Simulation of ClDL *solutions.* To simulate targets that heuristically correspond to explicit generators α output by the ClDL, we assume that for each ideal $\mathfrak{L}_{\mathsf{i}} \in \mathcal{S}$, the vector $\big(v_{\mathfrak{L}_{\mathsf{i}}^{\sigma}}(\alpha)\big)_{\sigma\in G_m}$ of $\mathbb{Z}^{\frac{\varphi(m)}{2}}$ is uniform modulo the lattice of class relations, and that after projection along the $\mathbf{1}$-axis, $\big(\ln|\sigma(\alpha)|\big)_{\sigma}$ is uniform modulo the log-unit lattice. These hypotheses have already been used in [DPW19, Hyp. 8] or [BR20, H. 4.8], and are backed up by theoretical results in [BDPW20, Th. 3.3].

Drawing random elements modulo a lattice of rank k is done by following a Gaussian distribution of sufficiently large deviation. Concretely, we first choose a random split prime p in $[\![2^{97}, 2^{103}]\!]$. Then, for each $\mathfrak{L} \in \mathcal{S} \cap \mathcal{S}_0$, we pick random valuations $v_{\mathfrak{L}}(\alpha)$ modulo the lattice of class relations of rank $\left|\mathcal{S}\cap\mathcal{S}_0\right|$ and random elements $(u_\sigma)_{\sigma\in G_m^+} \in \mathbb{R}^{\varphi(m)/2}$ in the span of the log-unit lattice of rank $\frac{\varphi(m)}{2} - 1$. Finally, we simulate $(\ln|\sigma(\alpha)|)_\sigma$ by adding $\frac{\ln p + \sum_{\mathfrak{L}\in\mathcal{S}} v_{\mathfrak{L}} \ln \mathcal{N}(\mathfrak{L})}{\varphi(m)}$ to each coordinate u_σ, so that their sum is $\frac{\ln|\mathcal{N}(\alpha)|}{2}$. For each field, we thereby generate 100 random targets on which to test Twisted-PHS on all lattice versions.

Reconstruction of a solution. For each simulated ClDL generator α, given as a random vector $(\{\ln|\sigma(\alpha)|\}_{\sigma\in G_m^+}, \{v_{\mathfrak{L}}(\alpha)\}_{\mathfrak{L}\in\mathcal{S}\cap\mathcal{S}_0})$, it is easy to compute $\varphi(\alpha)$ for any log-$\mathcal{S}$-embedding φ and to derive a target as in [BR20, Eq. (4.7)], including a drift parameterized by some β. Then, considering e.g. $L_{\mathsf{sat}} = \varphi(\mathfrak{F}_{\mathsf{sat}})$, given by the *$BKZ_{40}$-reduced* basis $U_{\mathsf{bkz}} \cdot \varphi(\mathfrak{F}_{\mathsf{sat}})$, we find a close vector $v = (y \cdot U_{\mathsf{bkz}}) \cdot \varphi(\mathfrak{F}_{\mathsf{sat}})$ to this target using Babai's Nearest Plane algorithm, and from y, U_{bkz} and $\mathfrak{F}_{\mathsf{sat}}$ we easily recover, in compact representation, $s \in \mathcal{O}_{K_m,\mathcal{S}}^{\times}$ s.t. $v = \varphi(s)$ and also α/s.

The purpose of the drift parameter β is to guarantee $v_{\mathfrak{L}}(\alpha/s) \geq 0$ on all finite places. As mentioned in [BR20], the length of α/s is extremely sensitive to the value of β, so that they searched for an optimal value by dichotomy. However, this positiveness property actually does not seem to be monotonic in β, and in practice, using the same β on each finite place coordinate is too coarse when the dimension grows, resulting in unnecessarily large approximation factors. We instead obtained best results using random drifts in ℓ_∞-norm balls of radius 1 centered on the $\mathbf{1}$ axis. A first sampling of $O(\varphi(m))$ random points $\beta \cdot \mathbf{1} + \mathcal{B}_\infty(1)$ for a wide range of random β's allows us to select a β_0 around which we found the best $\|\alpha/s\|_2$ with all $v_{\mathfrak{L}}(\alpha/s)$ being positive. Then we sample $O(\varphi(m))$ uniform random points in the neighbourhood of β_0, namely in $[0.9\beta_0, 1.1\beta_0] \cdot \mathbf{1} + \mathcal{B}_\infty(1)$, and output the overall optimal $\|\alpha/s\|_2$ having all $v_{\mathfrak{L}}(\alpha/s) \geq 0$.

Estimator of the approximation factor. Since we do not have access to the shortest element of a challenge ideal, we cannot compute an exact approximation factor as in [BR20]. Instead, we estimate the retrieved approximation factor using the inequalities implied by Eq. (2.3). We focus on the Gaussian Heuristic, which gives consistent results with the exact approximation factors found in [BR20], in small dimensions. For each cyclotomic field, the plotted points are the means, over the 100 simulated random targets, of the minimal approximation factors obtained using options iso/noiso and exp/tw. For each family $\mathfrak{F}$, $\mathfrak{F}_{\mathsf{sat}}$ and $\mathfrak{F}_{\mathsf{su}}$, we chose to keep only the factor base that gives the best result. This systematically translated into using $d = 1$ G_m-orbit for $\mathfrak{F}$ and $\mathfrak{F}_{\mathsf{sat}}$, whereas we had to use $d = d_{\max}$ for $\mathfrak{F}_{\mathsf{su}}$, as predicted by the Twisted-PHS algorithm.

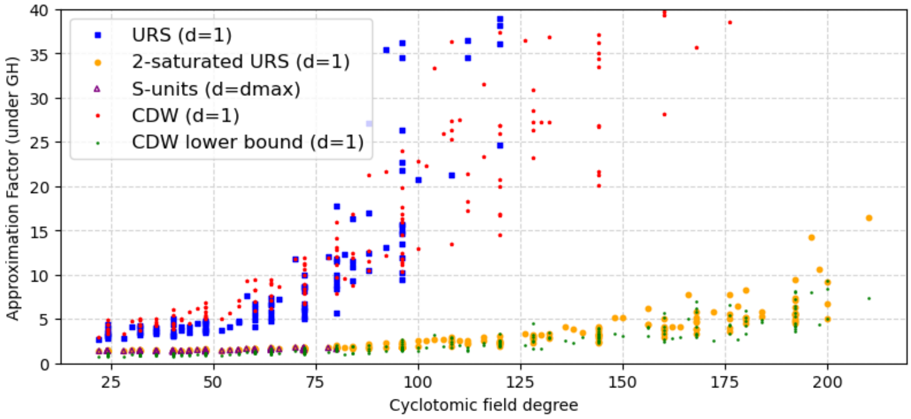

FIG. 5.2 – Approximation factors, with Gaussian Heuristic, reached by Tw-PHS for cyclotomic fields of degree up to 210, on lattices L_{urs}, L_{sat} and L_{su}.

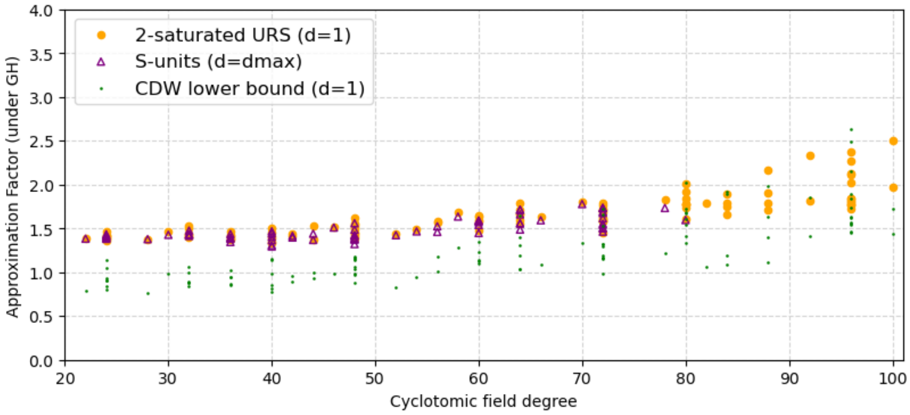

FIG. 5.3 – Approximation factors, with Gaussian Heuristic, reached by Tw-PHS for cyclotomic fields of degree up to 100, on lattices L_{sat} and L_{su}.

Figure 5.2 shows the approximation factor $\mathrm{af}_{\mathsf{gh}}$ obtained for all lattices L_{urs}, L_{sat} and L_{su} (when applicable) after BKZ_{40} reduction. Figure 5.3 is a zoom of Fig. 5.2 that focuses on L_{sat} and L_{su} on small dimensions. First, we remark that using $\mathfrak{F}$ from Eq. (3.8), the retrieved approximation factors are increasing rapidly. Using the 2-saturated family $\mathfrak{F}_{\mathsf{sat}}$ yields much better results, and looking closely at Fig. 5.3 shows that using a basis $\mathfrak{F}_{\mathsf{su}}$ of the full $\mathcal{S}$-unit group, when feasible, even improves the picture if $d_{\max} > 1$, in which case L_{su} is denser than L_{sat}. For L_{su}, we stress that we obtain estimated approximation factors very similar to the exact ones observed in [BR20].

More generally, we observe a very strong correlation between the density of our lattices and the obtained approximation factors – the denser, the better. As an important related remark, the variance seen for $\mathrm{af}_{\mathsf{gh}}$ in Fig. 5.2 for distinct fields of same degree follows the variations of the norm of the first split prime, thus of the reduced volume of the considered log-$\mathcal{S}$-unit sublattice. We expect this variance to be smoothed through conductors for the full log-$\mathcal{S}$-unit lattice.

Furthermore, considering $m = 211$, the $\mathfrak{F}$ family gives $\mathrm{Vol}^{1/314} L_{\mathsf{urs}} \approx 14.325$ and an estimated $\mathrm{af}_{\mathsf{gh}} \approx 13170$, for $\mathfrak{F}_{\mathsf{sat}}$ we get $\mathrm{Vol}^{1/314} L_{\mathsf{sat}} \approx 11.386$ and a much smaller estimated $\mathrm{af}_{\mathsf{gh}} \approx 16.4$, whereas the optimal number of orbits predicted by the Twisted-PHS factor base choice algorithm [BR20, Alg. 4.1] is $d_{\max} = 7$, which yields a full log-$\mathcal{S}$-unit lattice of reduced volume only $\mathrm{Vol}^{1/1574} L_{\mathsf{su}} \approx 9.635$.

Comparison to the CDW algorithm. Using the same experimental setting, we compute the approximation factors obtained using the CDW algorithm as implemented in [DPW19] ("Naive version") with additional BKZ_{40} lattice reductions, as well as the experimentally derived *volumetric lower bound* from [DPW19, Eq. (5) and Tab. 1]. Those values are also represented in Fig. 5.2 and 5.3.

We note that our experimental results using the $\mathfrak{F}_{\mathsf{sat}}$ family are comparable to this volumetric lower bound. Moreover, for some fields, e.g. in dimensions 96, 160, 168, 200, this lower bound is defeated by the (approximated version of the) Twisted-PHS algorithm. Note that this does not invalidate the lower bound itself, which is stated for the two-phase CDW algorithm, but indicates the power of combining both steps in only one lattice as in the Twisted-PHS algorithm.

Acknowledgements. The first author is deeply indebted to Radan Kučera for the proof of Lem. 3.12, and for thorough and invaluable discussions about the Stickelberger ideal. Andrea Lesavourey is funded by the Direction Générale de l'Armement (Pôle de Recherche CYBER), with the support of Région Bretagne. This work is supported by the European Union PROMETHEUS project (Horizon 2020 Research and Innovation Program, grant 780701).

References

Bab86. L. Babai: *On Lovász' lattice reduction and the nearest lattice point problem.* Combinatorica, **6**(1), pp. 1–13, 1986.

Bac90. É. Bach: *Explicit bounds for primality testing and related problems.* Math. Comp., **55**(191), pp. 355–380, 1990.

BBV^{+}17. J. Bauch, D. Bernstein, H. de Valence, T. Lange, C. van Vredendaal: *Short generators without quantum computers: the case of multiquadratics.* In *EUROCRYPT (1)*, vol. 10210 of *LNCS*, pp. 27–59, Springer, 2017.

BCP97. W. Bosma, J. Cannon, C. Playoust: *The Magma algebra system. I. The user language.* J. Symbolic Comput., **24**(3-4), pp. 235–265, 1997, computational algebra and number theory (London, 1993).

BDF08. K. Belabas, F. Diaz y Diaz, E. Friedman: *Small generators of the ideal class group.* Math. Comput., **77**(262), pp. 1185–1197, 2008.

BDPW20. K. d. Boer, L. Ducas, A. Pellet-Mary, B. Wesolowski: *Random self-reducibility of Ideal-SVP via Arakelov random walks.* In *CRYPTO (2)*, vol. 12171 of *LNCS*, pp. 243–273, Springer, 2020.

BEF$^+$17. J. Biasse, T. Espitau, P. Fouque, A. Gélin, P. Kirchner: *Computing generator in cyclotomic integer rings.* In *EUROCRYPT (1)*, vol. 10210 of *LNCS*, pp. 60–88, Springer, 2017.

Bel04. K. Belabas: *A relative van Hoeij algorithm over number fields.* J. Symb. Comput., **37**(5), pp. 641–668, 2004.

BF14. J. Biasse, C. Fieker: *Subexponential class group and unit group computation in large degree number fields.* LMS J. Comp. Math., **17**(A), pp. 385–403, 2014.

BFHP21. J. Biasse, C. Fieker, T. Hofmann, A. Page: *Norm relations and computational problems in number fields.* arXiv:2002.12332v3 [math.NT], 2021.

BK21. O. Bernard, R. Kučera: *A short basis of the Stickelberger ideal of a cyclotomic field.* arXiv:2109.13329 [math.NT], 2021.

BL21. D. J. Bernstein, T. Lange: *Non-randomness of s-unit lattices.* Cryptology ePrint Archive, Report 2021/1428, 2021, https://ia.cr/2021/1428.

BLP93. J. Buhler, H. Lenstra, C. Pomerance: *Factoring integers with the number field sieve*, vol. 1554 of *Lecture Notes in Math.* Springer, 1993.

BPR04. J. Buhler, C. Pomerance, L. Robertson: *Heuristics for class numbers of prime-power real cyclotomic fields.* Fields Inst. Commun., **41**, pp. 149–157, 2004.

BR20. O. Bernard, A. Roux-Langlois: *Twisted-PHS: Using the product formula to solve Approx-SVP in ideal lattices.* In *ASIACRYPT*, vol. 12492 of *LNCS*, pp. 349–380, Springer, 2020.

BS16. J.-F. Biasse, F. Song: *Efficient quantum algorithms for computing class groups and solving the principal ideal problem in arbitrary degree number fields.* In *SODA*, pp. 893–902, SIAM, 2016.

BV18. J. Biasse, C. van Vredendaal: *Fast multiquadratic S-unit computation and application to the calculation of class groups.* In *ANTS-XIII*, vol. 2 of *The Open Book Series*, pp. 103–118, Mathematical Sciences Publisher, 2018.

CDPR16. R. Cramer, L. Ducas, C. Peikert, O. Regev: *Recovering short generators of principal ideals in cyclotomic rings.* In *EUROCRYPT (2)*, vol. 9666 of *LNCS*, pp. 559–585, Springer, 2016.

CDW17. R. Cramer, L. Ducas, B. Wesolowski: *Short Stickelberger class relations and application to Ideal-SVP.* In *EUROCRYPT (1)*, vol. 10210 of *LNCS*, pp. 324–348, Springer, 2017.

CDW21. R. Cramer, L. Ducas, B. Wesolowski: *Mildly short vectors in cyclotomic ideal lattices in quantum polynomial time.* J. ACM, **68**(2), 2021.

CGS14. P. Campbell, M. Groves, D. Shepherd: *Soliloquy: A cautionary tale*, 2014, available at http://docbox.etsi.org/Workshop/2014/201410_CRYPTO/S07_Systems_and_Attacks/S07_Groves_Annex.pdf.

Che13. Y. Chen: *Réduction de réseau et sécurité concrète du chiffrement complètement homomorphe.* Ph.D. thesis, Paris 7, 2013.

CN11. Y. Chen, P. Q. Nguyen: *BKZ 2.0: Better lattice security estimates.* In *ASIACRYPT*, vol. 7073 of *LNCS*, pp. 1–20, Springer, 2011.

Coh93. H. Cohen: *A course in computational algebraic number theory*, vol. 138 of *Graduate texts in mathematics.* Springer, 1993.

DPW19. L. Ducas, M. Plançon, B. Wesolowski: *On the shortness of vectors to be found by the Ideal-SVP quantum algorithm.* In *CRYPTO (1)*, vol. 11692 of *LNCS*, pp. 322–351, Springer, 2019.

EHKS14. K. Eisenträger, S. Hallgren, A. Y. Kitaev, F. Song: *A quantum algorithm for computing the unit group of an arbitrary degree number field.* In *STOC*, pp. 293–302, ACM, 2014.

FpL16. FpLLL development team: *fplll, a lattice reduction library*, 2016, available at https://github.com/fplll/fplll.

GK89. R. Gold, J. Kim: *Bases for cyclotomic units.* Compos. Math., **71**(1), pp. 13–27, 1989.

GN08. N. Gama, P. Q. Nguyen: *Predicting lattice reduction.* In *EUROCRYPT*, vol. 4965 of *LNCS*, pp. 31–51, Springer, 2008.

HW38. G. H. Hardy, E. M. Wright: *An Introduction to the Theory of Numbers.* Oxford University Press, 1938, Fourth Edition.

Kuč86. R. Kučera: *On a certain subideal of the Stickelberger ideal of a cyclotomic field.* Archivum Mathematicum, **22**(1), pp. 7–19, 1986.

Kuč92. R. Kučera: *On bases of the Stickelberger ideal and of the group of circular units of a cyclotomic field.* J. Number Theory, **40**(3), pp. 284–316, 1992.

Laa16. T. Laarhoven: *Sieving for closest lattice vectors (with preprocessing).* In *SAC*, vol. 10532 of *LNCS*, pp. 523–542, Springer, 2016.

Les21. A. Lesavourey: *Usability of structured lattices for a post-quantum cryptography: practical computations, and a study of some real Kummer extensions.* Ph.D. thesis, University of Wollongong, 2021.

LLL82. A. K. Lenstra, H. W. Lenstra, L. Lovász: *Factoring polynomials with rational coefficients.* Math. Ann., **261**, pp. 515–534, 1982.

LPR10. V. Lyubashevsky, C. Peikert, O. Regev: *On ideal lattices and learning with errors over rings.* In *EUROCRYPT*, vol. 6110 of *LNCS*, pp. 1–23, Springer, 2010.

LPS20. A. Lesavourey, T. Plantard, W. Susilo: *Short principal ideal problem in multicubic fields.* J. Math. Cryptol., **14**(1), pp. 359–392, 2020.

MG02. D. Micciancio, S. Goldwasser: *Complexity of Lattice Problems*, vol. 671 of *The Kluwer International Series in Engineering and Computer Science.* Springer, 2002.

Mil14. J. C. Miller: *Class numbers of real cyclotomic fields of composite conductor.* LMS J. Comput. Math., **17**, pp. 404–417, 2014.

Nar04. W. Narkiewicz: *Elementary and Analytic Theory of Algebraic Numbers.* Springer Monographs in Mathematics, Springer, 3 edn., 2004.

Neu99. J. Neukirch: *Algebraic Number Theory*, vol. 322 of *Grundlehren des mathematischen Wissenschaften.* Springer, 1999.

PHS19. A. Pellet-Mary, G. Hanrot, D. Stehlé: *Approx-SVP in Ideal lattices with pre-processing.* In *EUROCRYPT (2)*, vol. 11477 of *LNCS*, pp. 685–716, Springer, 2019.

PZ89. M. Pohst, H. Zassenhaus: *Algorithmic Algebraic Number Theory.* Encyclop. Math. Appl., Cambridge University Press, 1989.

Sag20. Sage Developers: *SageMath, the Sage Mathematics Software System (Version 9.0)*, 2020, available at https://www.sagemath.org.

Sch87. C. Schnorr: *A hierarchy of polynomial time lattice basis reduction algorithms.* Theor. Comput. Sci., **53**, pp. 201–224, 1987.

Sch03. R. Schoof: *Class numbers of real cyclotomic fields of prime conductor.* Math. Comput., **72**(242), pp. 913–937, 2003.

Sch08. R. Schoof: *Catalan's Conjecture.* Universitext, Springer, 2008.

SE94. C. Schnorr, M. Euchner: *Lattice basis reduction: Improved practical algorithms and solving subset sum problems.* Math. Program., **66**, pp. 181–199, 1994.

Sin78. W. Sinnott: *On the Stickelberger ideal and the circular units of a cyclotomic field*. Ann. Math., **108**(1), pp. 107–134, 1978.

Sin80. W. Sinnott: *On the Stickelberger ideal and the circular units of an abelian field*. Invent. Math., **62**, pp. 181–234, 1980.

SSTX09. D. Stehlé, R. Steinfeld, K. Tanaka, K. Xagawa: *Efficient public key encryption based on ideal lattices*. In *ASIACRYPT*, vol. 5912 of *LNCS*, pp. 617–635, Springer, 2009.

Tho12. E. Thomé: *Square root algorithms for the Number Field Sieve*. In *WAIFI*, vol. 7369 of *LNCS*, pp. 208–224, Springer, 2012.

Was97. L. C. Washington: *Introduction to Cyclotomic Fields*, vol. 83 of *Graduate Texts in Mathematics*. Springer, 2 edn., 1997.

Wes18. B. Wesolowski: *Generating subgroups of ray class groups with small prime ideals*. In *ANTS-XIII*, vol. 2 of *The Open Book Series*, pp. 461–478, Mathematical Sciences Publisher, 2018.

A Arithmetic details for the description of an explicit full-rank family of independent $\boldsymbol{\mathcal{S}}$-units

A.1 Two special subsets of $[\![1, m]\!]$

We recall here from resp. [Kuč92, p.293] and [BK21, Eq. (11)] the definition of two subsets M_m^+ and M_m' of $[\![1, m]\!]$ that are useful to describe resp. a fundamental family of circular units and a short $\mathbb{Z}$-basis of the Stickelberger ideal of K_m.

Recall that m has prime factorization $m = q_1 q_2 \cdots q_t \not\equiv 2 \mod 4$, where $q_i = p_i^{e_i} > 2$ for $i \in [\![1, t]\!]$. Let X_m be the set of all positive integers $a < m$ that are either divisible by q_i or relatively prime to q_i for each $i \in [\![1, t]\!]$, i.e.:

$$X_m = \Big\{ a \in \mathbb{Z};\ 0 < a < m, \big(a, \tfrac{m}{(a,m)}\big) = 1 \Big\}.$$

Let $M_m^\pm \subseteq X_m$ be the sets of all $a \in X_m$ satisfying ([Kuč92, p.293]):[9]

- for all $i \in [\![1, t]\!]$, if $q_i \nmid a$ then $a \not\equiv -(a, m) \mod q_i$,
- if $a \nmid m$, let $k = \max\big\{i \in [\![1, t]\!];\ a \not\equiv (a, m) \mod q_i\big\}$, then $\big\{\frac{a}{(a,m)q_k}\big\} < \frac{1}{2}$,
- if $a \mid m$ then the set $\big\{i \in [\![1, t]\!];\ q_i \nmid a\big\}$ has an even (resp. odd) number of elements when defining M_m^+ (resp. when defining M_m^-).

Finally, the set M_m' is defined from the previous set M_m^- using [BK21, Eq. (11)]:

$$M_m' = \Big\{ a \in M_m^-;\ \forall i \in [\![1, t]\!], \tfrac{m}{q_i} \nmid a \Big\} \cup \Big(\bigcup_{i=1}^{t} \Big\{ \tfrac{mb}{q_i};\ 1 \le b \le \tfrac{\varphi(q_i)}{2} \Big\} \Big).$$

Note that M_m^+ (resp. M_m') contains $\frac{\varphi(m)}{2} - 1$ elements (resp. $\frac{\varphi(m)}{2}$ elements). Both sets are obviously easy to compute, using only simple arithmetic criteria.

A.2 Explicit description of the map $\boldsymbol{\alpha_m(\cdot)}$

For the sake of being self-contained, we recall here, sticking to the exposition of [BK21, §3.2], the description of the map $\alpha_m(\cdot) : [\![1, m]\!] \to \mathcal{S}_m$, whose image lies

[9] Actually, the set M_+ defined in [Kuč92, p.293] is $M_+ = M_m^+ \cup \{0\}$.

in the family of short elements described in Pr. 3.7. It yields a short basis of the lattice $\mathcal{S}_m \setminus \{N_m\}$ when applied on the set $M'_m \subsetneq [\![1, m]\!]$ defined in §A.1.

For any positive $b \in \mathbb{Z}$, define J_b as the set $\{i \in [\![1,t]\!];\ q_i \mid b\}$, hence $r_b = \prod_{i \in J_b} q_i$ is the maximal divisor of (b, m) s.t. $\left(r_b, \frac{m}{r_b}\right) = 1$. Let $J'_b = [\![1,t]\!] \setminus J_b$ be the set of indices i s.t. $q_i \nmid b$. If $b < m$, then $J'_b \neq \emptyset$ and $\alpha_m(b)$ is defined by:

1. If $J'_b = \{j\}$, then $b = c \cdot \frac{m}{q_j}$ for $0 < c < q_j$, and [BK21, Eq. (16) and (15)]:
$$\alpha_m(b) = \begin{cases} 2\theta_m\left(\frac{\varphi(q_j)\cdot m}{2\cdot q_j}\right) - \theta_m\left(\frac{\varphi(q_j)\cdot m}{q_j}\right) & \text{if } c = 1, \\ \theta_m\left(\frac{m}{q_j}\right) + \theta_m\left(b - \frac{m}{q_j}\right) - \theta_m(b) & \text{otherwise.} \end{cases}$$
2. If $\left|J'_b\right| > 1$, let $u = q_i$ for some $i \in J'_b$ and $v = \frac{m}{ur_b}$. Since $(u, v) = 1$, there exist x, $y \in \mathbb{Z}$ s.t. $ux + vy = 1$, and [BK21, Eq. (14)]:
$$\alpha_m(b) = \theta_m(bux) + \theta_m(bvy) - \theta_m(b).$$

In [BK21, Lem. 3.2], these elements are shown to satisfy the conditions of Pr. 3.7. In particular, for any $b \in \mathbb{Z}$ s.t. $0 < b < m$, this implies that $\alpha_m(b) \in \mathcal{S}_m$ is short and $(1 + \tau) \cdot \alpha_m(b) = N_m$.

B Removing quantum steps from the CDW algorithm

In §3.2 a short basis for the Stickelberger lattice has been introduced in Th. 3.8, as well as associated generators. We make use of these new elements and see how they can be applied to the approx-Svp algorithm from [CDW17,CDW21]. First, we recall the original algorithms with only aesthetic rearrangement that will reveal useful later on. Then, using explicit Stickelberger elements corresponding to the class group relations of the relatively short generating family W of [CDW21], as well as principal relative norm ideals generators, we replace the last Pip call in the query phase by a class group computation in the preprocessing phase in the maximal real subfield, hence in dimension half of the initial field. Finally, we remove the need of using the random walk mapping challenge ideals into the minus part of the class group, by using the module of *all* real class group relations $C^{+}_{\mathfrak{l}_1,\ldots,\mathfrak{l}_d}$ introduced in §3.3, under the restriction that $h^+_m \leq O(\sqrt{m})$ (Hyp. B.1). Note that this last part requires using the index formula from Pr. 3.14.

B.1 Hypothesis on the plus part of the class number

The CDW algorithm from [CDW21] assumes that $h^+_m \leq \operatorname{poly}(m)$ for any conductor m [CDW21, Ass. 2]. This is needed for their random walk procedure mapping any ideal to Cl^-_m to have a running time in $\operatorname{poly}(m)$. To remove this reduction to Cl^-_m constraint, we use a slightly more restrictive hypothesis.

Hypothesis B.1. We restrict to cyclotomic fields K_m verifying $h^+_m \leq O(\sqrt{m})$.

This assumption is certainly not true in general. Nevertheless, by the discussion in Section 2.2, it should be valid when m is a power of 2 and asymptotically when m is a prime power. Finally, according to Schoof's table, we note that $h^+_m \leq \sqrt{m}$ holds for more than 96.6% of all prime conductors $m = p < 10000$. We stress that this restriction only impacts the results of §B.4.

B.2 An equivalent rewriting of the CDW algorithm

The following general proposition will be useful for fully understanding algorithms from [CDW21] as well as the improvements we provide.

As stated in §2.1, given a cyclotomic field K_m, recall we identify $G_m/\langle\tau\rangle$ with G_m^+, and we consider the natural lift of those elements to G_m. For any $\sigma \in G_m^+$, and any $\alpha \in \mathbb{Z}[G_m]$, we write $\mathit{\Delta}_\sigma(\alpha) := \alpha_\sigma - \alpha_{\sigma\tau}$.

Proposition B.2. *Let $\alpha \in \mathbb{Z}[G_m]$. Then for all $\beta \in \mathbb{Z}[G_m]$, we have:*

$$\beta \equiv \alpha \bmod (1+\tau) \iff \forall \sigma \in G_m^+, \quad \mathit{\Delta}_\sigma(\beta) = \mathit{\Delta}_\sigma(\alpha).$$

Moreover, let $\beta \equiv \alpha \bmod (1+\tau)$, then:

1. *For any $\sigma \in G_m^+$:*
   - *$\beta_{\sigma\tau} = 0$ if, and only, if $\beta_\sigma = \mathit{\Delta}_\sigma(\alpha)$,*
   - *$\beta_\sigma = 0$ if, and only, if $\beta_{\sigma\tau} = -\mathit{\Delta}_\sigma(\alpha)$.*
2. *There is a unique $\beta \equiv \alpha \bmod (1+\tau)$ with nonnegative integer coordinates and minimal ℓ_1-norm, it is defined by:*

$$\forall \sigma \in G_m^+, (\beta_\sigma, \beta_{\sigma\tau}) = \begin{cases} (\mathit{\Delta}_\sigma(\alpha), 0) & \text{if } \mathit{\Delta}_\sigma(\alpha) \geqslant 0 \\ (0, -\mathit{\Delta}_\sigma(\alpha)) & \text{if } \mathit{\Delta}_\sigma(\alpha) < 0. \end{cases} \tag{B.1}$$

Proof. The first assertion is easy since $\beta \equiv \alpha \bmod (1+\tau)$ if, and only if, for all $\sigma \in G_m^+$, $(\beta_\sigma, \beta_{\sigma\tau}) \in (\alpha_\sigma, \alpha_{\sigma\tau}) + (1,1)\cdot\mathbb{Z}$. Thus, locally in the coordinates $\sigma, \sigma\tau$ (with a fixed σ), there is in the class of α modulo $(1+\tau)$ a unique β such that $\beta_{\sigma\tau} = 0$ and a unique β such that $\beta_\sigma = 0$. These are exactly $(\alpha_\sigma - \alpha_{\sigma\tau}, 0)$ and $(0, \alpha_{\sigma\tau} - \alpha_\sigma)$. A coordinate pair $(\beta_\sigma, \beta_{\sigma\tau}) \in \mathbb{Z}^2$ (of $\beta \in \mathbb{Z}[G_m]$) is parametrized as $\mathit{\Delta}_\sigma(\alpha)(1-\lambda, -\lambda)$ for some $\lambda \in \mathbb{R}$. The segment delimited by $(\mathit{\Delta}_\sigma(\alpha), 0)$ and $(0, \mathit{\Delta}_\sigma(\alpha))$ are the points such that $\lambda \in [0,1]$. For any $\lambda > 1$ we have:

$$\|\mathit{\Delta}_\sigma(\alpha)(1-\lambda, -\lambda)\|_1 = |\mathit{\Delta}_\sigma(\alpha)|(2\lambda - 1) > |\mathit{\Delta}_\sigma(\alpha)|,$$

and for $\lambda < 0$ one has:

$$\|\mathit{\Delta}_\sigma(\alpha)(1-\lambda, -\lambda)\|_1 = |\mathit{\Delta}_\sigma(\alpha)|(2|\lambda| + 1) > |\mathit{\Delta}_\sigma(\alpha)|.$$

Last, if $\lambda \in [0,1]$ the norm is $|\mathit{\Delta}_\sigma(\alpha)|$. Finally, in order to find a minimal element in a given class of $\mathbb{Z}[G_m]$ modulo $(1+\tau)$ with nonnegative coefficients only, it is sufficient to find a minimal pair $(\beta_\sigma, \beta_{\sigma\tau})$ with nonnegative coefficients for each $\sigma \in G_m^+$. Fix $\sigma \in G_m^+$ and assume without loss of generality that $\mathit{\Delta}_\sigma(\alpha) \geqslant 0$. Then following the characterisation above, any equivalent pair with minimal norm can be written $\mathit{\Delta}_\sigma(\alpha)(1-\lambda, -\lambda)$ with $\lambda \in [0,1]$. Among them, $(\mathit{\Delta}_\sigma(\alpha), 0)$ is clearly the only pair such that both coefficients are nonnegative. □

Algorithm B.1 $\textsc{WalkToCl}^-(\mathfrak{a})$: random walk to Cl_m^-

```
Input:  an ideal $\mathfrak{a} \subset \mathcal{O}_{K_m}$.
Output: an ideal $\mathfrak{b} \subset \mathcal{O}_{K_m}$ s.t. $[\mathfrak{a}\mathfrak{b}] \in \mathrm{Cl}_m^-$ and $\mathcal{N}(\mathfrak{b}) \leq \exp(\tilde{O}(m))$.
$\ell = \tilde{O}(m), B = poly(m)$
repeat
    for $i = 1, \dots, \ell$ do
        Choose $\mathfrak{L}_i$ uniformly at random among prime ideals of norm less than $B$
        $\mathfrak{b} \leftarrow \prod_{i=1}^{d} \mathfrak{L}_i$
until $\mathcal{N}_{K_m/K_m^+}(\mathfrak{a}\mathfrak{b})$ is principal, using the (quantum) Pip algorithm from [BS16]
return $\mathfrak{b}$
```

Algorithm B.2 $\textsc{Reduce}(W, \xi)$: finds a reduction of ξ

```
Input:  $\alpha \in \mathbb{Z}[G_m]$ and $W \subset \mathbb{Z}[G_m]$ a generating set of the Stickelberger lattice.
Output: $\beta \in \mathbb{Z}[G_m]$ s.t. $\|\beta\|_1 \leq \frac{1}{4} \cdot \varphi(m)^{3/2}$, and $C^\alpha = C^\beta$ for any $C \in \mathrm{Cl}_m^-$.
$v \leftarrow \textsc{Cvp}(\pi(W), \pi(\alpha))$
$\gamma \leftarrow \pi(\alpha) - v \cdot \pi(W)$
Define $(a_\sigma)_{\sigma \in G_m^+}$ as the integral coordinates of $\gamma$ in the basis $(\pi(\sigma))_{\sigma \in G_m^+}$ of
    $\mathbb{Z}[G_m]/(1 + \tau)$
$\beta \leftarrow \sum_{\sigma \in G_m^+} a_\sigma \sigma \in \mathbb{Z}[G_m]$
return $\beta$
```

Algorithm B.3 $\textsc{Cpm}^-(W, \mathfrak{L}, \alpha)$: solves the CPM problem for ideal $\mathfrak{L}^\alpha$

```
Input:  A generating set $W$ [CDW21, Lem. 4.4] of the Stickelberger lattice, an ideal
    $\mathfrak{L}$ such that $[\mathfrak{L}] \in \mathrm{Cl}_{K_m}^-$ and an element $\alpha \in \mathbb{Z}[G_m]$.
Output: an integral ideal $\mathfrak{b} = \mathfrak{L}^\gamma$ s.t. $\mathfrak{L}^\alpha \mathfrak{b}$ is principal and and $\mathcal{N}(\mathfrak{b}) =$
    $\mathcal{N}(\mathfrak{L})^{O(\varphi(m)^{3/2})}$.
$\beta \leftarrow \textsc{Reduce}(W, \alpha)$
Write $\beta$ as $\beta = \sum_{\sigma \in G_m^+} a_\sigma \sigma$
for $\sigma \in G_m^+$ do
    $(a_\sigma^+, a_\sigma^-) \leftarrow \begin{cases} (a_\sigma, 0) & a_\sigma \geq 0, \\ (0, -a_\sigma) & otherwise \end{cases}$
$\gamma \leftarrow \sum_{\sigma \in G_m^+} (a_\sigma^+ + a_\sigma^- \tau)\sigma$
return $\mathfrak{L}^\gamma$.
```

We can now recall the main algorithms from [CDW21]. Algorithm B.1 is $\textsc{WalkToCl}^-$ [CDW21, Alg. 5]. This algorithm gives reduces the general case to the case where the input ideal is in the relative class groups, for which the Stickelberger ideal is a natural lattice of class relations.

Once this technical requirement is satisfied, the main steps described in [CDW21] are given by the $\textsc{Reduce}$ algorithm in Alg. B.2 which corresponds to [CDW21, Alg. 3]. This algorithm is subsequently used in algorithm $\textsc{Cpm}^-$ described in Alg. B.3 and which corresponds to [CDW21, Alg. 4][10]. Note also that

[10] This algorithm was originally called $\textsc{ClosePrincipalMultiple}^-$ in [CDW21].

compared to [CDW21, Alg. 4], the end of CPM$^-$ algorithm is slightly modified to satisfy the convention we use for the ClDL algorithm.

Finally, all the previously introduced algorithms are used to define the algorithm CDW [CDW21, Alg. 7] solving Approx-SVP for ideal lattice algorithm[11]. For this last algorithm, it will be useful for us to use an equivalent rewriting of it in a preprocessing phase (Alg. B.4) and a query phase (Alg. B.5). We also recall there exists an algorithm SHORTGENERATOR [CDW21, Alg. 1] whose property is described in Th. B.3.

In order to be coherent with future algorithms that will be described with a preprocessing phase and a query phase, we argue that the (randomized) ClDL step of [CDW21, Alg. 6, lines 4–8] can be rewritten as follows. Essentially, instead of testing whether the ClDL algorithm succeeds within the algorithm, we fix a number of orbits d during the preprocessing phase (Alg. B.4) before moving to the query phase (Alg. B.5). If the ClDL step of the query phase fails then we go back to the preprocessing phase with a higher d. Also, for any of such d, we choose the bound $B = \mathit{poly}(m)$ so that $\mathfrak{M} = \{\mathfrak{L} \mid \mathcal{N}(\mathfrak{L}) \leq B, [\mathfrak{L}] \in \mathrm{Cl}_m^-\}$ has at least d elements allowing us to pick the d ideals of smallest norm within $\mathfrak{M}$, as in step 3 of Alg. B.4. After a small number of query we expect to find a sufficiently big d such that the ideals $\mathfrak{L}_1, \ldots, \mathfrak{L}_d$ generates Cl_m^-.

Algorithm B.4 CDW$_{\text{pre-proc}}$: find a generating family of Cl_m^-

```
Input:   a cyclotomic field $K_m$ of conductor $m$ and an integer $d$
Output:  a family $\mathfrak{B}$ of prime ideals (expected to generate $\mathrm{Cl}_m^-$)
1: $B = \mathit{poly}(m)$
2: $\mathfrak{M} \leftarrow \{\mathfrak{L} \mid \mathcal{N}(\mathfrak{L}) \leq B, [\mathfrak{L}] \in \mathrm{Cl}_m^-\}$
3: Choose $\mathfrak{L}_1, \ldots, \mathfrak{L}_d$ with smallest norm in $\mathfrak{M}$
4: $\mathfrak{B} \leftarrow \{\mathfrak{L}_i^\sigma \mid \sigma \in G_m, i = 1, \ldots, d\}$
5: return $\mathfrak{B}$
```

Algorithm B.5 CDW$_{\text{query}}(\mathfrak{a})$: finding mildly short vectors in the ideal $\mathfrak{a}$

```
Input:   an ideal $\mathfrak{a} \in \mathcal{O}_{K_m}$, a family $\mathfrak{B} \leftarrow \{\mathfrak{L}_i^\sigma \mid \sigma \in G_m, i = 1, \ldots, d\}$
Output:  an element $h \in \mathfrak{a}$ of norm $\|h\|_2 \leq \exp(\tilde{O}(\sqrt{m})) \cdot \mathcal{N}(\mathfrak{a})^{1/\varphi(m)}$
$\mathfrak{b}' \leftarrow \mathrm{WalkToCl}^-(\mathfrak{a})$
$(y_{i,\sigma})_{\sigma \in G_m, i=1,\ldots,d} \leftarrow \mathrm{ClDL}_{\mathfrak{B}}(\mathfrak{a}\mathfrak{b}')$        ▷ $\mathfrak{a}\mathfrak{b}' \prod_{i,\sigma} (\mathfrak{L}_i^\sigma)^{y_{i,\sigma}} \sim 1$
for $i = 1, \ldots, d$ do
    $\xi_i \leftarrow \sum_{\sigma \in G_m} y_{i,\sigma}\sigma \in \mathbb{Z}[G_m]$
    $\mathfrak{b}'_i \leftarrow \mathrm{CPM}^-(W, \mathfrak{L}_i, \xi_i)$
$\mathfrak{b} \leftarrow \mathfrak{b}' \prod_{i=1}^d \mathfrak{b}'_i$
$g \leftarrow \mathrm{PIP}(\mathfrak{a}\mathfrak{b})$
$h \leftarrow \mathrm{SHORTGENERATOR}(\mathrm{G})$
return $h$
```

Theorem B.3 ([CDW21, Th. 3.6]). *There is a randomized algorithm* SHORTGENERATOR *that for any $g \in \mathcal{O}_{K_m}$ (in compact representation), finds*

[11] This algorithm was originally called IDEALSVP in [CDW21].

an element $h \in \mathcal{O}_{K_m}$ (in compact representation) such that $g \cdot \mathcal{O}_{K_m} = h \cdot \mathcal{O}_{K_m}$ and $\|h\|_2 = \exp\left(O(\sqrt{m \log m})\right) \cdot \mathcal{N}(g)^{1/\varphi(m)}$, and runs in polynomial time in the size of the input.

Now that we have introduced algorithms used in [CDW21], we first look into steps 2–5 of Alg. B.3. Essentially, these steps guarantee that the exponent $\gamma \in \mathbb{Z}[G_m]$ has only nonnegative coordinates in the basis $((\sigma)_{\sigma \in G_m^+}, (\sigma\tau)_{\sigma \in G_m^+})$, using the property $\mathfrak{L}^{-1} \sim \mathfrak{L}^{\tau}$ that was ensured by the restriction to the relative class group. However, it is also important that the resulting γ has small norm $\|\gamma\|_1$. In Pr. B.4, we show that steps 2–5 of Alg. B.3 guarantee that the returned exponent γ is actually minimal in a certain sense. Before that, we introduce the subroutine PositiveOptim in Alg. B.6 that generalizes steps 2–5 of Alg. B.3. This algorithm also applies to elements whose "right part" of coordinates are not all zero and explicitely shows that the modification are done using elements of $(1+\tau) \cdot \mathbb{Z}[G_m^+]$.

Algorithm B.6 PositiveOptim(α): returns an element in the class of α modulo $(1+\tau) \cdot \mathbb{Z}[G_m^+]$ whose coordinates in basis $((\sigma)_{\sigma \in G_m^+}, (\sigma\tau)_{\sigma \in G_m^+})$ are nonnegative integers and which is minimal for ℓ_1-norm inside the equivalence class.

Input: an element $\alpha \in \mathbb{Z}[G_m]$

Output: an element $\tilde{\alpha} \equiv \alpha$ of minimal ℓ_1-norm and whose coordinates in basis $((\sigma)_{\sigma \in G_m^+}, (\sigma\tau)_{\sigma \in G_m^+})$ are nonnegative integers.

```
Write $\alpha$ as $((a_\sigma)_{\sigma \in G_m^+}, (a_{\sigma\tau})_{\sigma \in G_m^+})$ on the basis $\left((\sigma)_{\sigma \in G_m^+}, (\sigma\tau)_{\sigma \in G_m^+}\right)$ of $\mathbb{Z}[G_m]$
$\tilde{\alpha} \leftarrow \alpha$
for $\sigma \in G_m^+$ do                              ▷ Dealing with negative coordinates
    if $a_\sigma \leq a_{\sigma\tau}$ then
        $\tilde{\alpha} \leftarrow \tilde{\alpha} - a_\sigma(1+\tau)\sigma$
    else if $a_{\sigma\tau} \leq a_\sigma$ then
        $\tilde{\alpha} \leftarrow \tilde{\alpha} - a_{\sigma\tau}(1+\tau)\sigma$
return $\tilde{\alpha}$.
```

Proposition B.4. *In Alg. B.3, it is possible de replace steps 2–5 by subroutine* PositiveOptim. *Moreover, this shows the resulting γ has minimal ℓ_1-norm given $\beta \leftarrow$* Reduce(W, α) *as in step 1 of Alg. B.3.*

Proof. We identify $\alpha \in \mathbb{Z}[G_m]$ with its coordinates $((a_\sigma)_{\sigma \in G_m^+}, (a_{\sigma\tau})_{\sigma \in G_m^+})$ in the basis $\left((\sigma)_{\sigma \in G_m^+}, (\sigma\tau)_{\sigma \in G_m^+}\right)$. Then, PositiveOptim, act the following way. For any $\sigma \in G_m^+$, $\alpha := (\ldots, a_\sigma, \ldots, a_{\sigma\tau}, \ldots)$ is mapped to $(\ldots, \Delta_\sigma(a), \ldots, 0, \ldots)$ if $\Delta_\sigma(a) \geq 0$, and if $\Delta_\sigma(a) < 0$, it is mapped to $(\ldots, 0, \ldots, -\Delta_\sigma(a), \ldots)$. This is precisely what steps 2–5 returns in the particular case where for all $\sigma \in G_m^+$, $a_{\sigma\tau} = 0$. Note that by Pr. B.2, those images are precisely of minimal norm (inside a fixed equivalence class). All in all, we conclude that the transformation $\beta \mapsto \gamma$ of Alg. B.3 remains inside the equivalence class of β and that it returns the element of minimal ℓ_1-norm inside this class. □

In Pr. B.4, we proved that given a particular class modulo $(1+\tau)$, algorithm PositiveOptim returns an element (with nonnegative coordinates) of the class whose ℓ_1-norm is minimal among all elements of the class. Nevertheless, among all the coset $\alpha + S_m$ how can we find the class whose associated minimal value is the smallest value among all the possible lower bounds? In previous works such as [CDW17,DPW19,CDW21] the question is raised when discussing whether to use a Cvp solver on $\pi(S_m)$ and then lifting it back, or directly on the extended Stickelberger lattice $S_m + (1+\tau)\mathbb{Z}[G_m]$[12]. The following proposition proves that, given an exact Cvp solver, the construction using $\pi(S_m)$ is optimal.

Proposition B.5. *Let $\alpha \in \mathbb{Z}[G_m]$, W_{bk} the short basis of the Stickelberger ideal S_m as introduced in Th. 3.8 and note* Cvp *an exact close vector problem solver on $\pi(W_{\mathsf{bk}})$, for ℓ_1 norm. Define $\gamma(v) := \textsc{PositiveOptim}(\alpha - v \cdot W_{\mathsf{bk}})$, as a function of $v \in \mathbb{Z}^{\varphi(m)/2}$, then: $\mathrm{argmin}_{v\in\mathbb{Z}^{\varphi(m)/2}} \|\gamma(v)\|_1 = \textsc{Cvp}(\pi(W_{\mathsf{bk}}), \pi(\alpha))$.*

Proof. We note $\{w_1, \ldots, w_{\varphi(m)/2}\}$ the elements of W_{bk} and we write v as the vector $(v_1, \ldots, v_{\varphi(m)/2}) \in \mathbb{Z}^{\varphi(m)/2}$. Then:

$$\begin{aligned}\|\gamma(v)\|_1 &= \sum_{\sigma\in G_m^+} |\mathit{\Delta}_\sigma(\textsc{PositiveOptim}(\alpha - v\cdot W_{\mathsf{bk}}))| \\ &= \sum_{\sigma\in G_m^+} |\mathit{\Delta}_\sigma(\alpha - \sum_{i=1}^{\varphi(m)/2} v_i w_i)|\end{aligned}$$

by applying Pr. B.2, since subroutine PositiveOptim does not alter the equivalence class. Now, by definition of the projection π,

$$\begin{aligned}\sum_{\sigma\in G_m^+} |\mathit{\Delta}_\sigma(\alpha - \sum_{i=1}^{\varphi(m)/2} v_i w_i)| &= \sum_{\sigma\in G_m^+} |\pi(\alpha)_\sigma - \sum_i v_i \pi(w_i)_\sigma| \\ &= \|\pi(\alpha) - v\cdot\pi(W_{\mathsf{bk}})\|_1.\end{aligned}$$

Hence, minimizing $\|\gamma(v)\|_1$ is equivalent to minimizing $\|\pi(\alpha) - v\cdot\pi(W_{\mathsf{bk}})\|_1$, which is achieved by taking $v = \textsc{Cvp}(\pi(W_{\mathsf{bk}}), \pi(\alpha))$. □

B.3 Using explicit Stickelberger generators

Many quantum steps are required in the query phase of the CDW algorithm (Alg. B.5). First, the random walk to reach Cl_m^- requires a polynomial number (in $h_{K_m}^+$) of steps and each of these steps requires a Pip test in the maximal real subfield. Second, a ClDl step is performed in the cyclotomic field to obtain inputs used in the Cpm$^-$ subroutine. Finally, a final Pip is performed in the cyclotomic field in order to recover a short generator.

Our goal in this subsection is to use Th. 3.8, the associated generators and the subroutine PositiveOptim, to reduce the cost of the last Pip call (inside

[12] Both modules being used as a replacement for S_m not being full-rank.

Alg. B.5). In order to do so, one key ingredient is to replace the generating set W by the short basis W_{bk} of the Stickelberger lattice, introduced in Th. 3.8. This last switch is beneficial for several reasons:

1. In order to solve Cvp, using [CDW21, Cor. 2.2], one does not need anymore to compute a maximal set of linearly independent vectors inside W (in a greedy manner). We also note that this full-rank set of vectors only ensures that the Cvp algorithm is done inside a (full rank) sublattice of the Stickelberger lattice. Whereas, using the complete Stickelberger lattice basis ensures the best result for the Cvp algorithm, regarding the approximation factor.
2. The second advantage is that, we can use the explicit Stickelberger generators (associated to the principal ideals resulting from the action of W_{bk}). Exhibiting such Stickelberger generators is (in general) not possible for elements of the generating set W. This point will be of importance for replacing the last Pip call in dimension n (which is done for any challenge) in the CDW algorithm, by the computation of the real class group. Note that this last part also required the introduction of PositiveOptim (Alg. B.6).

Concretely, we define $(\text{Cpm}^-)'$ as Alg. B.7 and $\text{CDW}^{\mathsf{explicit}}$ as the successive combinaison of algorithms Alg. B.8 and B.9, defining a preprocessing phase and a query phase.

We first prove correctness of the new algorithms we introduced. Notably, we prove that prove that Cpm^- returns the same result as $(\text{Cpm}^-)'$. Subsequently we deduce the correctness of algorithm $\text{CDW}^{\mathsf{explicit}}$ which is splitted in a preprocessing phase (Alg. B.8) and then a query phase (Alg. B.9).

Corollary B.6. *Algorithm* $(\text{Cpm}^-)'$ *and* $\text{CDW}^{\mathsf{explicit}}$ *are correct. Notably, let* $\mathfrak{L}$ *be a an ideal st.* $[\mathfrak{L}] \in \text{Cl}_m^-$ *and* $\alpha \in \mathbb{Z}[G_m]$*, then, algorithms* Cpm^- *and* $(\text{Cpm}^-)'$ *output the same result on input* $(W_{\mathsf{bk}}, \mathfrak{L}, \alpha)$*.*

Proof. The correctness of algorithm $(\text{Cpm}^-)'$ is a straightforward corollary of Pr. B.4, since this proposition essentially shows that using steps 2–5 of Alg. B.2 or subroutine PositiveOptim (Alg. B.6) returns the same element (and note that we use the same Cvp solver in both Cpm^- and $(\text{Cpm}^-)'$). This result is not dependant on W_{bk} and would have been true for the original generating family W from [CDW21]. For the correctness of $\text{CDW}^{\mathsf{explicit}}$, we first use the fact that Cpm^- and $(\text{Cpm}^-)'$ output the same result on input $(W_{\mathsf{bk}}, \mathfrak{L}, \alpha)$. Secondly, using results specific to W_{bk}, we note that §3.2 provides us with Stickelberger

Algorithm B.7 $\left(\text{CPM}^-\right)'(W_{\mathsf{bk}}, \mathfrak{L}, \alpha)$: solves the CPM problem for ideal $\mathfrak{L}^\alpha$

Input: W_{bk} the basis of the Stickelberger lattice defined in §3.2, an ideal $\mathfrak{L}$ such that $[\mathfrak{L}] \in \mathrm{Cl}^-_{K_m}$ and an element $\alpha \in \mathbb{Z}[G_m]$.

Output: an integral ideal $\mathfrak{b} = \mathfrak{L}^\gamma$ s.t. $\mathfrak{L}^\alpha\mathfrak{b}$ is principal and $\mathcal{N}(\mathfrak{b}) = \mathcal{N}(\mathfrak{L})^{O(\varphi(m)^{3/2})}$.

```
$v \leftarrow \text{CVP}(\pi(W_{\mathsf{bk}}), \pi(\alpha))$
$\beta \leftarrow \alpha - v \cdot W_{\mathsf{bk}}$
$\gamma \leftarrow \text{PositiveOptim}(\beta)$
return $\mathfrak{L}^\gamma$.
```

Algorithm B.8 $\mathrm{CDW}^{\mathsf{explicit}}_{\text{pre-proc}}$: finding a generating family for the relative class group and generators for certain principal ideals

Input: a cyclotomic field K_m of conductor m and an integer d

Output: a family $\mathfrak{B} \leftarrow \{\mathfrak{L}_i^\sigma \mid \sigma \in G_m, i = 1, \ldots, d\}$ generating Cl^-_m and generators of the principal ideals $\{\mathfrak{L}_i^{\alpha_m(b)}\}_{i,b}$ ($\alpha_m(b) \in W_{\mathsf{bk}}$) and $\{\mathfrak{L}_i^{1+\tau}\}_i$

```
$d = \mathrm{polylog}(m), B = \mathrm{poly}(m)$
$\mathfrak{M} \leftarrow \{\mathfrak{L} \mid \mathcal{N}(\mathfrak{L}) \le B, [\mathfrak{L}] \in \mathrm{Cl}^-_m\}$
Choose $\mathfrak{L}_1, \ldots, \mathfrak{L}_d$ with smallest norm in $\mathfrak{M}$
$\mathfrak{B} \leftarrow \{\mathfrak{L}_i^\sigma \mid \sigma \in G_m, i = 1, \ldots, d\}$
    subsec:stigens
Compute generators $\{\gamma^-_{\mathfrak{L}_i,b}\}_b$ st. $\mathfrak{L}_i^{\alpha_m(b)} = \langle\gamma^-_{\mathfrak{L}_i,b}\rangle$ for $\alpha_m(b) \in W_{\mathsf{bk}}$ and $i = 1, \ldots, d$
    ▷ See §3.2
Compute generators $\{\gamma^+_r\}_r$ st. $\langle\gamma^+_r\rangle = \prod_{i=1}^d \mathfrak{L}_i^{(1+\tau)r_i}$ for $r \in \mathbb{Z}[G^+_m]^d$ ▷ See Eq. (3.8) and Rem. 3.9
return $\mathfrak{B}$, $\{\gamma^-_{\mathfrak{L}_i,b}\}_{\substack{i=1,\ldots,d,\\ b\in M'_m}}$, $\{\gamma^+_{\mathfrak{L}_r}\}_{r\in\mathbb{Z}[G^+_m]}$
```

Algorithm B.9 $\mathrm{CDW}^{\mathsf{explicit}}_{\text{query}}(\mathfrak{a})$: finding mildly short vectors in the ideal $\mathfrak{a}$

Input: an ideal $\mathfrak{a} \in \mathcal{O}_{K_m}$, a family $\mathfrak{B} \leftarrow \{\mathfrak{L}_i^\sigma \mid \sigma \in G_m, i = 1, \ldots, d\}$ generating Cl^-_m and generators $\{\gamma^-_{\mathfrak{L}_i,b}\}_{\substack{i=1,\ldots,d,\\ b\in M'_m}}$, $\{\gamma^+_{\mathfrak{L}_r}\}_{r\in\mathbb{Z}[G^+_m]}$

Output: an element $h \in \mathfrak{a}$ of norm $\|h\|_2 \le \exp(\tilde{O}(\sqrt{m})) \cdot \mathcal{N}(\mathfrak{a})^{1/\varphi(m)}$

```
$\mathfrak{b}' \leftarrow \mathrm{WalkToCl}^-(\mathfrak{a})$
$\xi, (y_{i,\sigma})_{\sigma\in G_m, i=1,\ldots,d} \leftarrow \text{ClDL}_{\mathfrak{B}}(\mathfrak{a}\mathfrak{b}')$      ▷ $\langle\xi\rangle \sim \mathfrak{a}\mathfrak{b}' \prod_{i,\sigma}(\mathfrak{L}_i^\sigma)^{y_{i,\sigma}}$
for $i = 1, \ldots, d$ do
    $\xi_i \leftarrow \sum_{\sigma\in G_m} y_{i,\sigma}\sigma \in \mathbb{Z}[G_m]$
    $\mathfrak{L}_i^{\gamma_i} \leftarrow (\text{CPM}^-)'(W_{\mathsf{bk}}, \mathfrak{L}_i, \xi_i)$
        where $\gamma_i = \xi_i - \sum_b v_{i,b}\alpha_m(b) - (1+\tau)r_i$
        for integers $(v_{i,b})_{b\in M'_m}$ and $r_i \in \mathbb{Z}[G^+_m]$
$g \leftarrow \xi \Big/ \left(\gamma^+_r \cdot \prod_{\substack{i=1,\ldots,d\\ b\in M'_m}} (\gamma^-_{\mathfrak{L}_i,b})^{v_{i,b}}\right)$ where $r = (r_1, \ldots, r_d) \in \mathbb{Z}[G^+_m]$
$h \leftarrow \text{ShortGenerator}(\mathrm{g})$
return $h$
```

generators associated to the lattice basis W_{bk}. In other words, for any ideal $\mathfrak{L}_i$ of the basis, there exists elements $\{\gamma^{-}_{\mathfrak{L}_i,b}\}_b$ st. $\mathfrak{L}_i^{\alpha_m(b)} = \langle\gamma^{-}_{\mathfrak{L}_i,b}\rangle$ for $b \in M'_m$. Moreover, the ClDl algorithm from Biasse and Song (used in Alg. B.5) not only recovers the family $(y_{i,\sigma})_{\sigma\in G_m, i=1,\ldots,d}$ but also the element $\xi \in \mathcal{O}_{K_m}$ such that:

$$\mathfrak{ab}' = \langle\xi\rangle \prod_{\substack{i=1,\ldots,d\\ \sigma\in G_m}} (\mathfrak{L}_i^{\sigma})^{-y_{i,\sigma}} = \langle\xi\rangle \prod_i \mathfrak{L}_i^{-\xi_i}$$

with the notation $\xi_i := \sum_{\sigma\in G_m} y_{i,\sigma}\sigma$ for $i = 1,\ldots,d$. Now, for a fixed ideal $\mathfrak{L}_i$, on input $(\mathfrak{L}_i, \xi_i)$, algorithm $(\text{Cpm}^{-})'$ returns an element $\mathfrak{L}_i^{\gamma_i}$ with $\gamma_i = \xi_i - \sum_{b\in M'_m} v_{i,b}\alpha_m(b) - (1+\tau)r_i$ where $(v_{i,b})_{b\in M'_m}$ is the vector with integral coordinates obtain by the Cvp subroutine inside $(\text{Cpm}^{-})'$, and $r_i \in \mathbb{Z}[G_m^{+}]$. Then:

$$\mathfrak{ab}'\left(\prod_{i=1,\ldots,d} \mathfrak{L}_i^{\gamma_i}\right) = \langle\xi\rangle \prod_{\substack{i=1,\ldots,d\\ b\in M'_m}} \mathfrak{L}_i^{-\sum_b v_{i,b}\alpha_m(b)} \mathfrak{L}_i^{-(1+\tau)r_i}.$$

To conclude, recall from Eq. (3.6) and Rem. 3.9 that for $r = (r_1,\ldots,r_d) \in \mathbb{Z}[G_m]^d$, we have $\langle\gamma_r^{+}\rangle = \prod_{i=1,\ldots,d} \mathfrak{L}_i^{(1+\tau)r_i}$. □

In terms of calls to quantum algorithms, we replaced the last Pip in dimension n (for each query) by the computation of some generators during the preprocessing phase (step 6, Alg. B.8). Now, these generators can be all obtained by the computation of the real class group. Indeed, following [BS16, Th. 1.1 and Alg. 1], we note that the computation of the class group reduces to the calculation of $\mathcal{S}$-units for a particular set $\mathcal{S}$. In particular, this implies that the calculation of the class group also yields, at the same time, the generators associated to those class relations. Finally, from [BS16, Alg. 2], we deduce that the cost of computing $\mathcal{S}$-units is similar to the cost of the Pip algorithm. Concretely, this means that computing the relations during the preprocessing has a quantum cost equivalent to the cost of a single query to the Pip algorithm in dimension $n/2$.

B.4 Avoiding the random walk

In the previous algorithms presented, in the original, as well as the first modification, working on the minus part of the class group is still required. Hence doing the random walk from Alg. B.1 is still required during the query phase. We note this random walk calls for polynomially (in $h^{+}_{K_m}$) many calls to the Pip algorithm (in dimension $n/2$), in order to test membership to Cl^{-}_m of candidate ideals by testing principality in $\mathcal{O}^{+}_{K_m}$ (of the images by the relative norm map $\mathcal{N}_{K_m/K_m^{+}}$). A possible theoretical solution to bypass those Pip calls is to use relations induced from relations on $\mathrm{Cl}_{K_m^{+}}$. These relations were introduced in §3.3 as $C^{+}_{\mathfrak{l}_1,\ldots,\mathfrak{l}_d}$ for ideals $\mathfrak{l}_1,\ldots,\mathfrak{l}_d$ associated to ideals $\mathfrak{L}_1,\ldots,\mathfrak{L}_d$ generating Cl_{K_m} and required the computation of the real class group. Using the same argument made at the end of the previous paragraph means that steps 3–4 of Alg. B.10 can be done using a single call to a $\mathcal{S}$-units computation, whose cost is equivalent to the a single call to Pip in dimension $n/2$.

One technical issue is that using relations coming from real classes does not let us use algorithm PositiveOptim anymore, yet we still need to recover an element $\gamma \in \mathbb{Z}[G_m]$ with nonnegative integer coordinates. We proceed "à la PHS" (or Twisted-PHS) by computing a "drifted" Cvp, the added drift being chosen greater than the infinity decoding radius of the Cvp solver used. Note that using the Cvp algorithm requires the lattice to be full-rank, and this is precisely the result from Pr. 3.14. Like previous CDW and $\mathrm{CDW}^{\textsf{explicit}}$ algorithms, the $\mathrm{CDW}^{\textsf{no-walk}}$ algorithm is splitted in a preprocessing phase (Alg. B.10) followed by a query phase (Alg. B.11).

Algorithm B.10 $\mathrm{CDW}^{\textsf{no-walk}}_{\text{pre-proc}}$: finding a generating family for the relative class group and generators for certain principal ideals

```
Input:  a cyclotomic field $K_m$ of conductor $m$
Output: a family $\mathfrak{B} \leftarrow \{\mathfrak{L}_i^\sigma \mid \sigma \in G_m, i = 1, \dots, d\}$ generating $\mathrm{Cl}_{K_m}$, the generators
    of the principal ideals $\{\mathfrak{L}_i^{\alpha_m(b)}\}_{i,b}$ ($\alpha_m(b) \in W_{\mathsf{bk}}$), the real class relations $C^+_{\mathfrak{l}_1,\dots,\mathfrak{l}_d}$
    for $\mathfrak{l}_i = \mathcal{N}_{K_m/K_m^+}(\mathfrak{L}_i)$ ($i = 1, \dots, d$) as well as the associated generators
1: Compute $\mathrm{Cl}_{K_m} = \langle \mathfrak{L}_1, \dots, \mathfrak{L}_d \rangle$ with $d \leq \mathrm{polylog}(m), \max_i \mathcal{N}(\mathfrak{L}_i) := B \leq \mathrm{poly}(m)$
2: Compute generators $\{\gamma^-_{i,b}\}_{i,b}$ st. $\mathfrak{L}_i^{\alpha_m(b)} = \langle \gamma^-_{i,b} \rangle$ for $\alpha_m(b) \in W_{\mathsf{bk}}$ and $i = 1, \dots, d$
    ▷ Using §3.2
3: Compute the real class relations $C^+_{\mathfrak{l}_1,\dots,\mathfrak{l}_d}$ associated to ideals $\mathfrak{l}_i = \mathcal{N}_{K_m/K_m^+}(\mathfrak{L}_i)$
    ($i = 1, \dots, d$)                                                    ▷ See §3.3
4: Compute generators $\{\gamma^+_r\}_r$ st. $\langle \gamma_r \rangle = \prod_{i=1}^d \mathfrak{L}_i^{(1+\tau)r_i}$ for $r = (r_1, \dots, r_d) \in C^+_{\mathfrak{l}_1,\dots,\mathfrak{l}_d}$
5: return $\mathfrak{B}$, $\{\gamma^-_{i,b}\}_{\substack{i=1,\dots,d\\ b \in M'_m}}$, $\{\gamma^+_r\}_{r \in C^+_{\mathfrak{l}_1,\dots,\mathfrak{l}_d}}$
```

Algorithm B.11 $\mathrm{CDW}^{\textsf{no-walk}}_{\text{query}}(\mathfrak{a})$: finding mildly short vectors in the ideal $\mathfrak{a}$

```
Input:  an ideal $\mathfrak{a}$ in $\mathcal{O}_{K_m}$, a family $\mathfrak{B} \leftarrow \{\mathfrak{L}_i^\sigma \mid \sigma \in G_m, i = 1, \dots, d\}$ generat-
    ing $\mathrm{Cl}_{K_m}$, generators $\{\gamma^-_{i,b}\}_{\substack{i=1,\dots,d\\ b \in M'_m}}$, $\{\gamma^+_r\}_{r \in C^+_{\mathfrak{l}_1,\dots,\mathfrak{l}_d}}$ and a drift $\beta$ greater than the
    decoding radius of the Cvp algorithm
Output: $h \in \mathfrak{a}$ of norm $\|h\|_2 \leq \exp\left(\tilde{O}(\max(\sqrt{\varphi(m)}, h^+_{K_m}))\right) \cdot \mathcal{N}(\mathfrak{a})^{1/\varphi(m)}$
1: $\xi, y \leftarrow \mathrm{ClDL}_{\mathfrak{B}}(\mathfrak{a})$ where $y := (y_{i,\sigma})_{\sigma \in G_m, i=1,\dots,d}$        ▷ $\langle \xi \rangle \sim \mathfrak{a} \prod_{i,\sigma} (\mathfrak{L}_i^\sigma)^{y_{i,\sigma}}$
2: Let $B = \mathcal{S}_m^d + (1+\tau) C^+_{\mathfrak{l}_1,\dots,\mathfrak{l}_d}$
3: $v \leftarrow \mathrm{Cvp}\big(B, y + (\beta, \dots, \beta)\big)$, where $v := \left[ (v_{i,b})_{1 \leq i \leq d, b \in M'_m}, (v_r)_{r \in C^+_{\mathfrak{l}_1,\dots,\mathfrak{l}_d}} \right]$
4: $g \leftarrow \xi \Big/ \Big( \prod_{r \in C^+_{\mathfrak{l}_1,\dots,\mathfrak{l}_d}} (\gamma^+_r)^{v_r} \prod_{1 \leq i \leq d, b \in M'_m} (\gamma^-_{\mathfrak{L}_i,c})^{v_{i,b}} \Big)$
5: $h \leftarrow$ ShortGenerator$(g)$
6: return $h$
```

Proposition B.7. *Algorithm* $\mathrm{CDW}^{\textit{no-walk}}$ *is correct.*

Proof. We first note that the Cvp algorithm in the query phase is meaningful since the lattice $\Lambda = \mathcal{S}_m^d + (1+\tau) C^+_{\mathfrak{l}_1,\dots,\mathfrak{l}_d}$ is full-rank, using Pr. 3.14. Now, the drifted Cvp algorithm described in step 3 ensures that g (given in step 4) is in

$\mathfrak{a}$. Indeed, fix $i \in \{1, \ldots, d\}$. Note $z = [(v_{i,b})_{i,b}, (v'_r)_r] \cdot \mathit{\Lambda}$, by definition of the decoding radius D of the Cvp algorithm, $\|y + (\beta, \ldots, \beta) - z\|_\infty \leq D \leq \beta$. Taking coordinates, it follows that for any $i = 1, \ldots, d$ and $\sigma \in G_m$, $|y_{i,\sigma} + \beta - z_{i,\sigma}| \leq \beta$ and then $0 \leq y_{i,\sigma} - z_{i,\sigma} \leq 2\beta$. Since by definition, $\langle g \rangle = \mathfrak{a} \prod_{i,\sigma} (\mathfrak{L}_i^\sigma)^{y_{i,\sigma} - z_{i,\sigma}}$, the algorithm returns $g \in \mathfrak{a}$. We now use Th. B.3 and corollary [CDW21, Cor. 2.2]. Notably, in our situation,

$$\max_{w \in [W_{\mathsf{bk}} | C^+_{\mathfrak{l}_1, \ldots, \mathfrak{l}_d}]} \|w\|_2 = \max(\sqrt{\varphi(m)/2}, h^+_{K_m}),$$

using Pr. 3.10 and that short elements of the basis W_{bk} have ℓ_2-norm equal to $\sqrt{\varphi(m)/2}$ (see Th. 3.8). It follows that if $h = \textsc{ShortGenerator}(g)$, then:

$$\begin{aligned}
\|h\|_2 &= \exp\left(O(\sqrt{m \log m})\right) \cdot \mathcal{N}(g)^{1/\varphi(m)} \\
&\leq \exp\left(O(\sqrt{m \log m})\right) \cdot (B^{d \cdot |G_m| \cdot \max(\sqrt{\varphi(m)/2}, h^+_{K_m})})^{1/\varphi(m)} \cdot \mathcal{N}(\mathfrak{a})^{1/\varphi(m)} \\
&\leq \exp\left(\tilde{O}(\max(\sqrt{\varphi(m)}, h^+_{K_m}))\right) \cdot \mathcal{N}(\mathfrak{a})^{1/\varphi(m)}.
\end{aligned}$$

The last inequality is obtained using the assumption [CDW21, Ass. 1] stating that B can be chosen as poly(m) while d can be chosen as polylog(m) (line 1 of Alg. B.10). Originally, this assumption was stated for Cl^-_m but the justification in [CDW21, §6] readily extends to the whole class group under Hyp. B.1. □

The quantum steps used in algorithms CDW, CDW$^{\mathsf{explicit}}$ and CDW$^{\mathsf{no\text{-}walk}}$ are summarized in Tab. B.1. We emphasize that the computation of the class group has a cost equivalent to the Pip algorithm (in the same dimension), since they both reduce to a single call to the computation of $\mathcal{S}$-units, for suitable sets $\mathcal{S}$ of prime ideals.

	Preproccessing phase	Query phase		
	Class group computation (dim. $n/2$)	Pip (dim. n)	Pip (dim. $n/2$)	ClDl
CDW	0	1	$O(\mathrm{poly}(h^+_{K_m}))$	1
CDW$^{\mathsf{explicit}}$	1	0	$O(\mathrm{poly}(h^+_{K_m}))$	1
CDW$^{\mathsf{no\text{-}walk}}$	1	0	0	1

Table B.1 – Number of quantum steps used for algorithms CDW, CDW$^{\mathsf{explicit}}$ and CDW$^{\mathsf{no\text{-}walk}}$.

C Additional experimental results

C.1 Geometry of log-$\mathcal{S}$-unit sublattices

In the following, we provide data regarding the geometry of the log-$\mathcal{S}$-unit sublattices L_{urs} and L_{sat} for additional cyclotomic fields.

m	d	set	k	$\mathrm{Vol}^{1/k}$	δ			$\max_{1\leq i\leq k}\|\mathbf{b}_i\|_2$		
					raw	LLL	bkz$_{40}$	raw	LLL	bkz$_{40}$
159	1	urs	155	11.291	2.177	1.702	1.686	71.228	62.253	60.096
		sat	155	8.989	6.143	1.898	1.921	3168.773	35.391	35.703
	2	urs	259	12.576	2.350	1.781	1.739	72.069	62.357	60.675
		sat	259	9.572	6.902	2.028	2.036	3168.773	36.062	35.703
	3	urs	363	13.364	2.419	1.798	1.750	75.913	65.973	63.701
		sat	363	9.978	7.602	2.066	2.066	3168.773	37.480	37.132
149	1	urs	221	12.192	2.828	2.091	1.999	74.637	71.073	68.291
		sat	221	9.697	12.473	2.305	2.244	12554.466	44.327	44.326
	2	urs	369	13.353	3.134	2.233	2.149	78.906	74.039	71.298
		sat	369	10.150	14.472	2.507	2.467	12554.466	47.719	46.438
	3	urs	517	13.962	3.269	2.271	2.190	80.529	76.289	76.007
		sat	517	10.410	22.211	2.569	2.531	85211.593	47.719	48.556
	4	urs	665	14.415	3.327	2.300	2.223	83.176	78.268	77.926
		sat	665	10.632	20.731	2.606	2.576	85211.593	47.768	48.556
516	1	urs	251	11.815	2.535	2.026	2.013	77.904	73.051	72.993
		sat	251	9.395	6.508	2.341	2.359	4850.233	44.290	43.783
	2	urs	419	12.921	2.833	2.156	2.129	82.452	76.629	75.586
		sat	419	9.818	8.208	2.550	2.565	5761.443	46.559	46.426
	3	urs	587	13.850	2.945	2.202	2.167	91.958	84.961	86.487
		sat	587	10.321	10.348	2.620	2.623	9544.834	49.096	49.971
	4	urs	755	14.445	2.998	2.222	2.188	93.457	86.198	87.794
		sat	755	10.650	12.682	2.652	2.652	26820.239	54.045	52.543
181	1	urs	269	12.855	2.747	2.308	2.146	81.230	79.924	79.204
		sat	269	10.220	7.486	2.537	2.499	5185.677	49.694	48.264
	2	urs	449	14.033	2.958	2.456	2.268	87.161	85.755	84.008
		sat	449	10.661	9.849	2.736	2.706	5185.677	50.406	51.466
	3	urs	629	14.823	3.064	2.508	2.311	92.620	90.665	88.578
		sat	629	11.045	12.340	2.801	2.778	9957.084	52.207	51.880
	4	urs	809	15.330	3.096	2.529	2.330	93.988	91.158	89.982
		sat	809	11.300	12.307	2.829	2.814	9957.084	53.598	53.519
209	1	urs	269	10.796	2.678	2.239	2.238	70.154	70.428	68.371
		sat	269	8.583	8.273	2.599	2.609	8920.663	42.887	42.683
	2	urs	449	12.651	2.921	2.320	2.300	92.739	89.996	88.251
		sat	449	9.612	14.860	2.729	2.722	45374.160	53.927	53.643

m	d	set	k	$\mathrm{Vol}^{1/k}$	δ			$\max_{1\leq i\leq k}\lVert\mathbf{b}_i\rVert_2$		
					raw	LLL	bkz_{40}	raw	LLL	bkz_{40}
217	1	urs	269	12.110	2.608	2.137	2.115	83.336	76.670	76.186
		sat	269	9.629	6.814	2.420	2.410	4415.772	47.546	46.464
	2	urs	449	13.741	2.857	2.270	2.251	96.095	87.194	87.023
		sat	449	10.440	10.474	2.630	2.623	14735.404	56.381	56.328
	3	urs	629	14.646	2.941	2.319	2.313	99.437	89.912	93.209
		sat	629	10.913	11.667	2.696	2.696	14735.404	56.381	57.135
279	1	urs	269	12.059	2.573	2.080	2.064	81.546	76.724	84.960
		sat	269	9.588	11.575	2.391	2.397	12586.042	51.509	50.663
	2	urs	449	13.528	2.836	2.212	2.195	92.187	86.744	96.124
		sat	449	10.278	12.899	2.603	2.604	12586.042	57.098	57.696
	3	urs	629	14.378	2.965	2.263	2.250	96.095	89.520	96.124
		sat	629	10.713	16.966	2.677	2.683	25638.489	57.098	57.696
	4	urs	809	14.971	3.010	2.285	2.268	99.014	92.948	99.817
		sat	809	11.036	17.733	2.709	2.713	25638.489	58.977	58.807
	5	urs	989	15.396	3.053	2.302	2.280	100.238	93.692	99.817
		sat	989	11.271	18.878	2.729	2.731	26995.083	61.123	59.322
297	1	urs	269	12.331	3.169	2.074	2.005	86.980	81.006	81.451
		sat	269	9.804	21.668	2.308	2.319	94056.513	48.941	48.984
	2	urs	449	13.513	3.676	2.252	2.148	90.321	83.985	85.236
		sat	449	10.266	36.211	2.540	2.546	94056.513	50.795	51.447
	3	urs	629	14.165	3.895	2.327	2.196	92.913	86.090	85.236
		sat	629	10.555	37.241	2.645	2.640	94056.513	51.969	51.524
	4	urs	809	14.674	4.007	2.356	2.224	96.821	89.321	87.488
		sat	809	10.816	40.952	2.688	2.685	94056.513	52.120	53.167
235	1	urs	275	11.873	2.631	2.183	2.132	80.433	77.904	79.127
		sat	275	9.439	7.618	2.479	2.470	5297.502	47.586	46.684
	2	urs	459	13.287	2.936	2.347	2.275	91.190	87.506	82.926
		sat	459	10.094	12.645	2.706	2.699	28003.197	51.044	51.229
	3	urs	643	14.178	3.061	2.398	2.328	96.709	91.765	91.485
		sat	643	10.563	13.258	2.780	2.772	28003.197	52.348	52.334
	4	urs	827	14.743	3.099	2.423	2.349	98.093	93.292	92.979
		sat	827	10.867	13.861	2.815	2.807	28003.197	55.931	54.179
564	1	urs	275	12.264	2.551	2.035	2.061	82.573	77.166	76.021
		sat	275	9.750	14.624	2.390	2.370	39653.048	46.848	46.757
	2	urs	459	13.384	2.831	2.193	2.230	87.333	81.561	80.426
		sat	459	10.168	15.707	2.655	2.637	39653.048	50.285	49.290
	3	urs	643	14.393	2.984	2.240	2.274	98.851	90.926	90.825
		sat	643	10.724	17.342	2.727	2.714	39653.048	53.003	53.868
	4	urs	827	15.032	3.029	2.256	2.292	100.234	91.997	92.037
		sat	827	11.080	18.829	2.757	2.744	39653.048	55.358	55.921

Table C.1 – Geometric characteristics of L_{urs}, L_{sat} and L_{su} for some cyclotomic fields with log-$\mathcal{S}$-embedding φ_{tw} (of type $\mathsf{iso}/\mathsf{exp}$). For *all* bases, the root-Hermite factor verifies $|\delta_0 - 1| < 10^{-3}$.

m	d	φ_{tw}-type	k	$\mathrm{Vol}^{1/k}$	δ			$\max_{1\leq i\leq k}\lVert\mathbf{b}_i\rVert_2$		
					raw	LLL	bkz$_{40}$	raw	LLL	bkz$_{40}$
159	1	iso/exp	155	8.989	6.143	1.898	1.921	3168.773	35.391	35.703
		iso/tw	155	10.088	7.533	2.117	2.143	4481.257	38.437	37.421
		noiso/exp	155	8.989	6.143	1.894	1.905	3168.773	34.229	34.689
		noiso/tw	155	10.088	7.533	2.119	2.139	4481.257	37.723	38.596
	2	iso/exp	259	9.572	6.902	2.028	2.036	3168.773	36.062	35.703
		iso/tw	259	10.258	8.805	2.313	2.337	4481.257	38.437	37.670
		noiso/exp	259	9.572	6.902	2.024	2.024	3168.773	35.579	35.802
		noiso/tw	259	10.258	8.805	2.317	2.334	4481.257	37.723	38.596
	3	iso/exp	363	9.978	7.602	2.066	2.066	3168.773	37.480	37.132
		iso/tw	363	10.484	9.857	2.373	2.397	4481.257	39.327	39.938
		noiso/exp	363	9.978	7.602	2.064	2.064	3168.773	38.643	38.255
		noiso/tw	363	10.484	9.857	2.376	2.392	4481.257	39.286	41.548
149	1	iso/exp	221	9.697	12.473	2.305	2.244	12554.466	44.327	44.326
		iso/tw	221	10.883	15.626	2.672	2.602	17754.669	49.653	49.399
		noiso/exp	221	9.697	12.473	2.307	2.266	12554.466	43.736	45.013
		noiso/tw	221	10.883	15.626	2.668	2.612	17754.669	49.143	48.693
	2	iso/exp	369	10.150	14.472	2.507	2.467	12554.466	47.719	46.438
		iso/tw	369	10.878	18.958	2.982	2.936	17754.669	52.622	53.154
		noiso/exp	369	10.150	14.472	2.509	2.483	12554.466	48.576	47.820
		noiso/tw	369	10.878	18.958	2.982	2.949	17754.669	54.041	50.666
	3	iso/exp	517	10.410	22.211	2.569	2.531	85211.593	47.719	48.556
		iso/tw	517	10.938	29.658	3.084	3.050	120507.386	52.788	53.154
		noiso/exp	517	10.410	22.211	2.569	2.552	85211.593	48.576	48.778
		noiso/tw	517	10.938	29.658	3.085	3.058	120507.386	54.041	52.131
	4	iso/exp	665	10.632	20.731	2.606	2.576	85211.593	47.768	48.556
		iso/tw	665	11.050	27.968	3.149	3.117	120507.386	53.017	53.154
		noiso/exp	665	10.632	20.731	2.606	2.594	85211.593	48.576	48.778
		noiso/tw	665	11.050	27.968	3.149	3.128	120507.386	54.041	52.385
516	1	iso/exp	251	9.395	6.508	2.341	2.359	4850.233	44.290	43.783
		iso/tw	251	10.544	8.112	2.739	2.733	6859.195	49.680	50.548
		noiso/exp	251	9.395	6.508	2.342	2.354	4850.233	42.774	44.385
		noiso/tw	251	10.544	8.112	2.730	2.739	6859.195	52.260	50.964
	2	iso/exp	419	9.818	8.208	2.550	2.565	5761.443	46.559	46.426
		iso/tw	419	10.522	10.682	3.059	3.062	8147.832	51.931	53.538
		noiso/exp	419	9.818	8.208	2.549	2.557	5761.443	46.306	47.683
		noiso/tw	419	10.522	10.682	3.055	3.064	8147.832	52.534	51.448
	3	iso/exp	587	10.321	10.348	2.620	2.623	9544.834	49.096	49.971
		iso/tw	587	10.845	13.713	3.168	3.167	13498.373	56.763	56.892
		noiso/exp	587	10.321	10.348	2.617	2.615	9544.834	51.019	51.870
		noiso/tw	587	10.845	13.713	3.169	3.167	13498.373	54.998	57.177
	4	iso/exp	755	10.650	12.682	2.652	2.652	26820.239	54.045	52.543
		iso/tw	755	11.068	16.973	3.221	3.219	37929.528	58.551	56.892
		noiso/exp	755	10.650	12.682	2.649	2.650	26820.239	51.019	51.870
		noiso/tw	755	11.068	16.973	3.221	3.220	37929.528	57.437	57.177

Table C.2 – Geometric characteristics of L_{sat} for some cyclotomic fields. Comparison between choices iso/noiso and exp/tw.

C.2 Gram-Schmidt logarithm norms

Here, we provide figures showing the Gram-Schmidt log norms for other cyclotomic fields and number of orbits, comparing values before and after reduction.

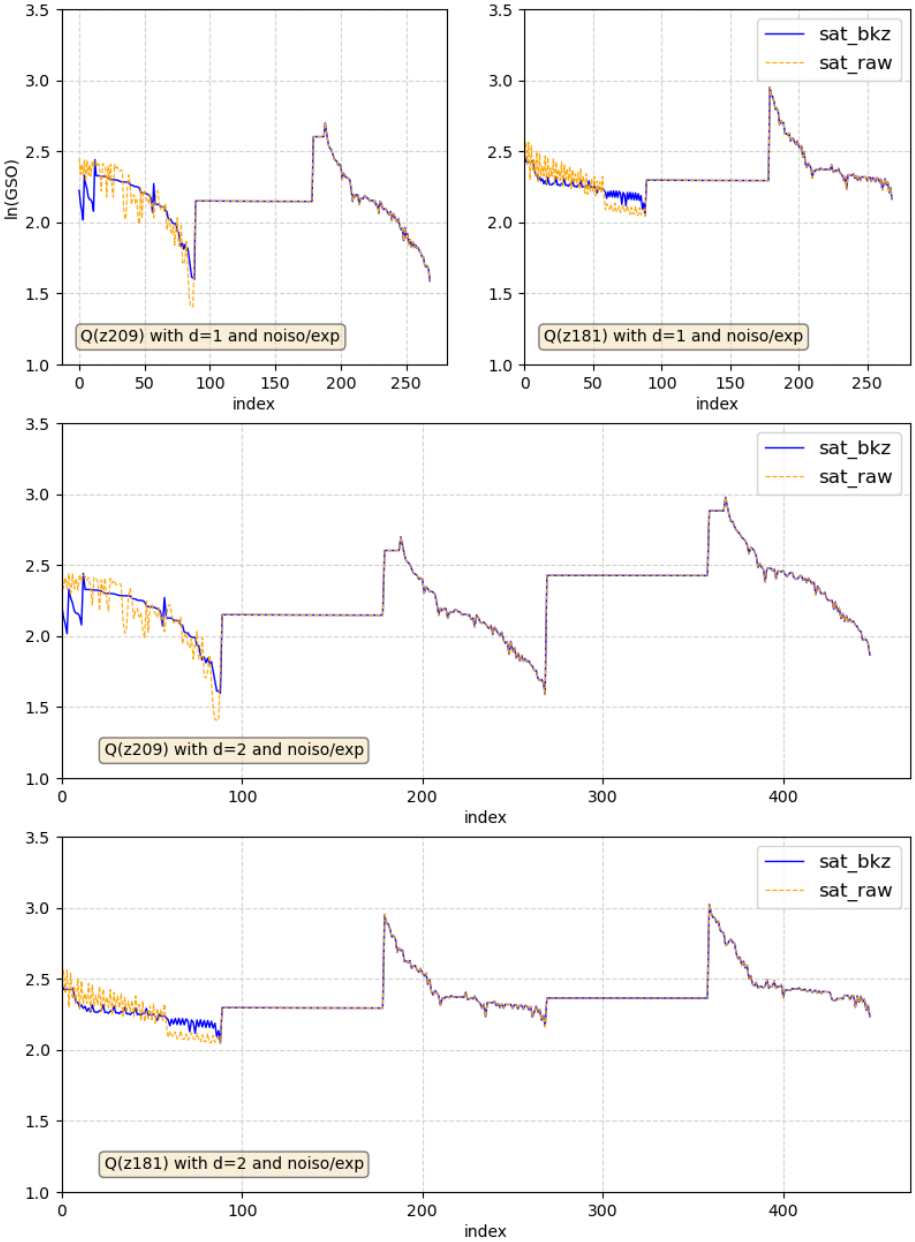

FIG. C.1 – L_{sat} lattices for $\mathbb{Q}(\zeta_{209})$ and $\mathbb{Q}(\zeta_{181})$: Gram-Schmidt log norms before and after reduction by BKZ_{40}, for $d = 1$ and $d = 2$ G_m-orbits.

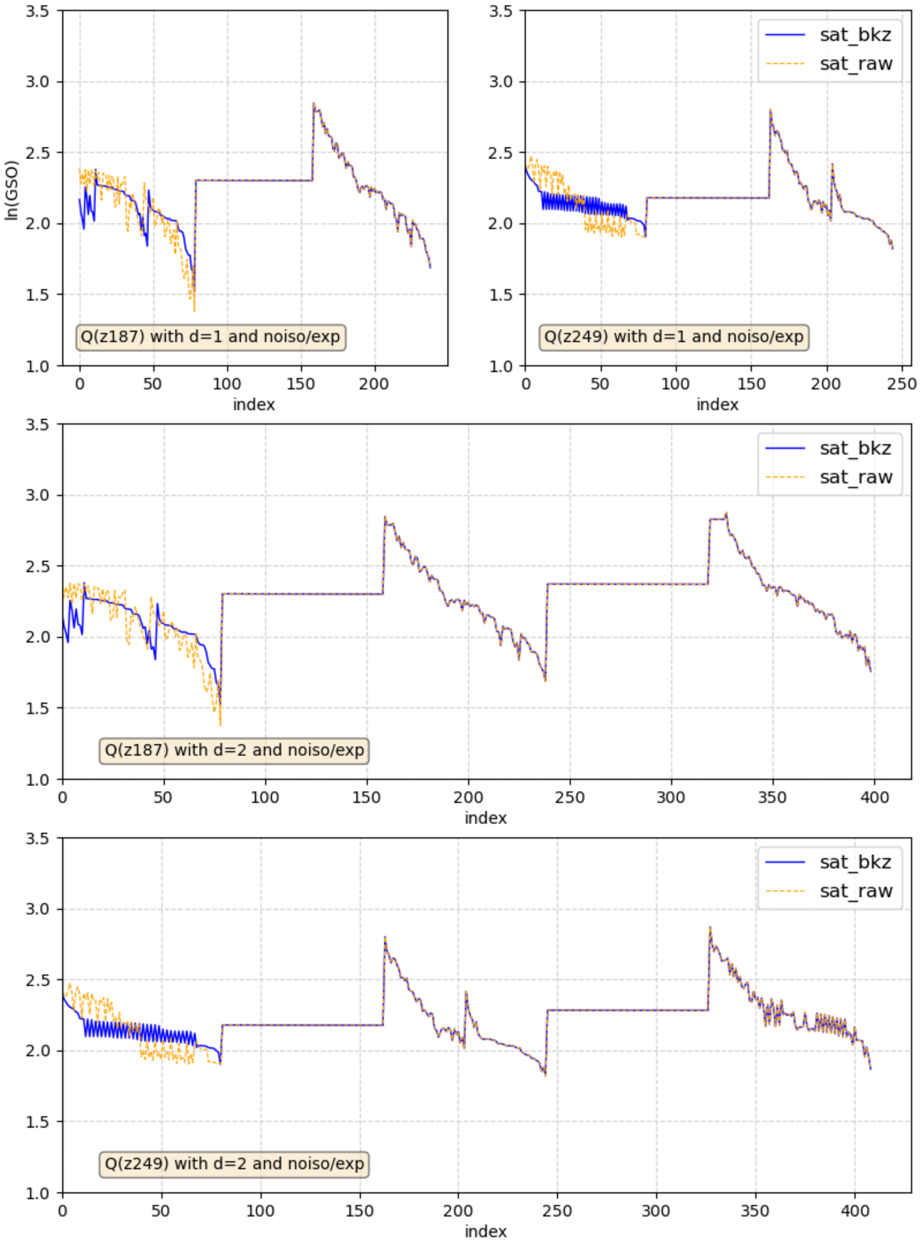

FIG. C.2 – L_{sat} lattices for $\mathbb{Q}(\zeta_{187})$ and $\mathbb{Q}(\zeta_{249})$: Gram-Schmidt log norms before and after reduction by BKZ_{40}, for $d = 1$ and $d = 2$ G_m-orbits.

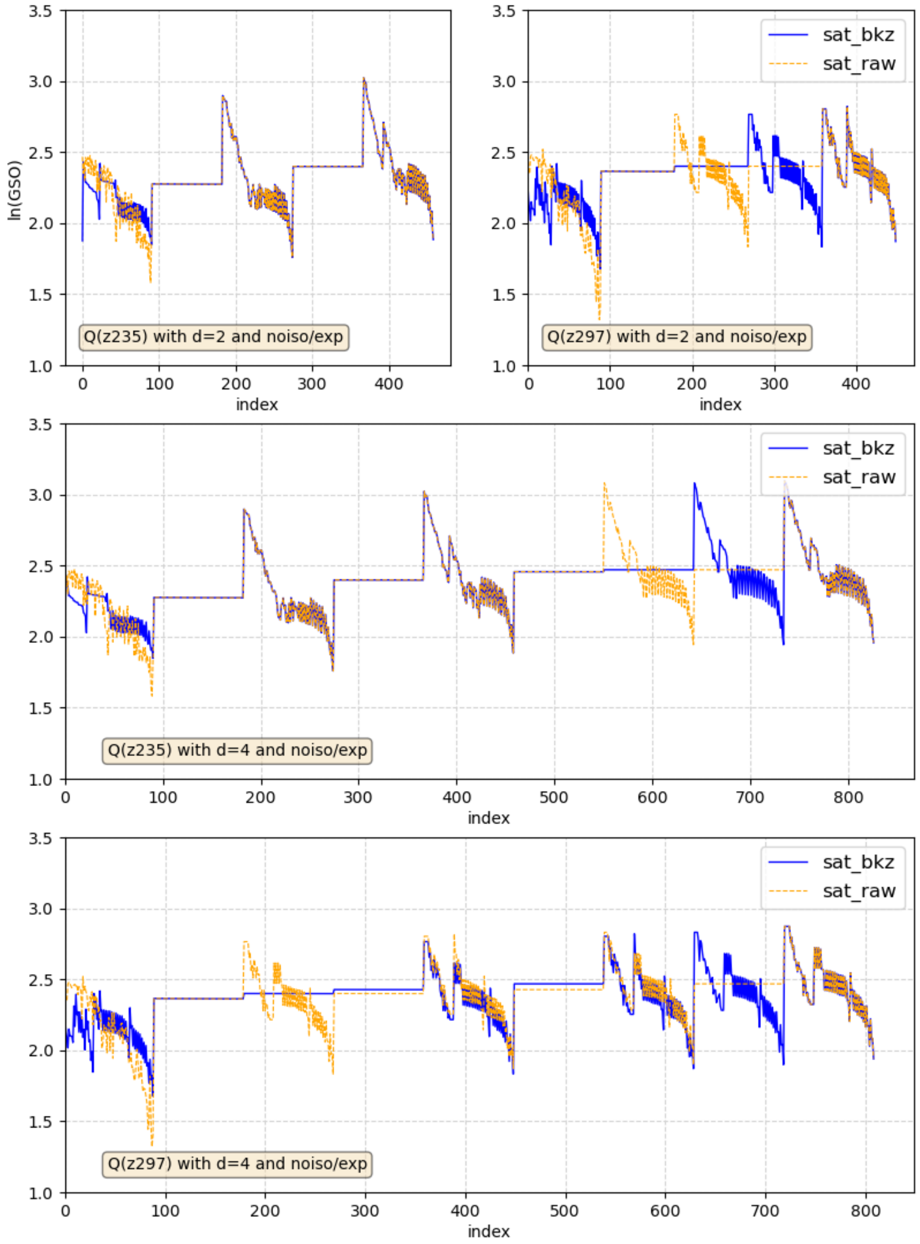

Fig. C.3 – L_{sat} lattices for $\mathbb{Q}(\zeta_{235})$ and $\mathbb{Q}(\zeta_{297})$: Gram-Schmidt log norms before and after reduction by BKZ_{40}, for $d = 2$ and $d = 4$ G_m-orbits.

Finally, Fig. C.4 shows the impact of the four choices of log-$\mathcal{S}$-embedding on the Gram-Schmidt logarithm norms of the unreduced basis $\varphi(\mathfrak{F}_{\mathsf{sat}})$.

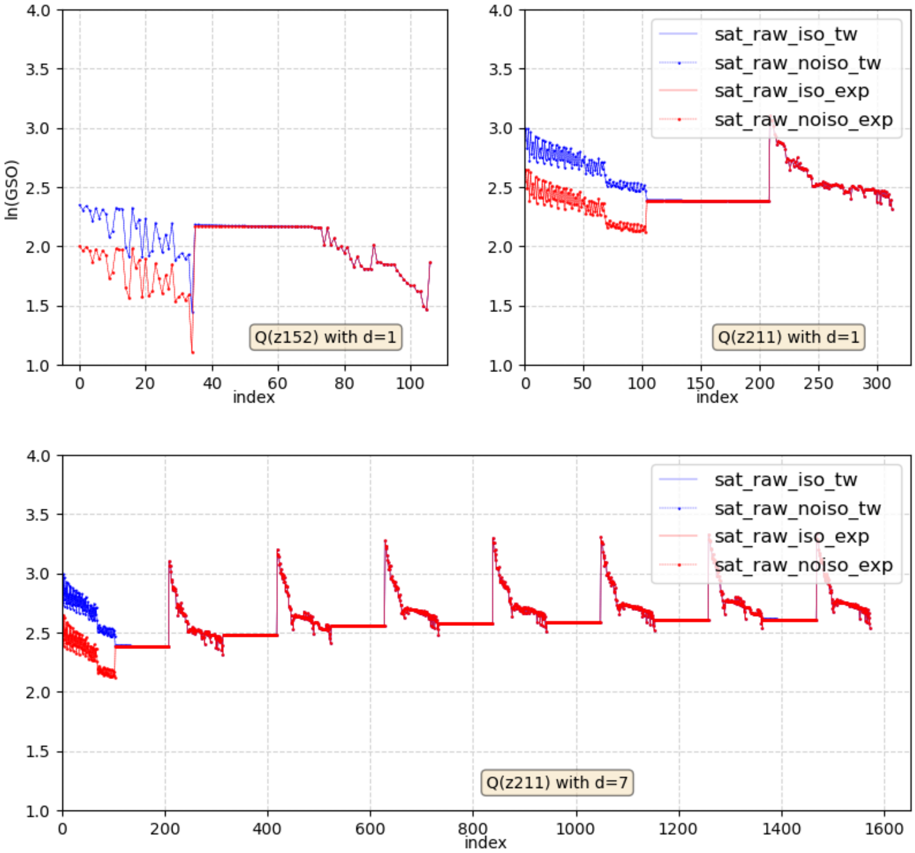

FIG. C.4 – L_{sat} lattices for $\mathbb{Q}(\zeta_{149})$ and $\mathbb{Q}(\zeta_{211})$: effect of the log-$\mathcal{S}$-embedding choices iso/noiso and exp/tw.