RIGOROUS METHODS FOR COMPUTATIONAL NUMBER THEORY

KOEN DE BOER[1], ALICE PELLET-MARY[2], AND BENJAMIN WESOLOWSKI[3]

[1]Leiden University, Mathematical Institute, Leiden, The Netherlands
[2]Univ. Bordeaux, CNRS, Inria, Bordeaux INP, IMB, UMR 5251, F-33400, Talence, France
[3]ENS de Lyon, CNRS, UMPA, UMR 5669, Lyon, France

Abstract. We present the first algorithm for computing class groups and unit groups of arbitrary number fields that provably runs in probabilistic subexponential time, assuming the Extended Riemann Hypothesis (ERH). Previous subexponential algorithms were either restricted to imaginary quadratic fields, or relied on several heuristic assumptions that have long resisted rigorous analysis.

The heart of our method is a new general strategy to provably solve a recurring computational problem in number theory (assuming ERH): given an ideal class $[\mathfrak{a}]$ of a number field K, sample an ideal $\mathfrak{b} \in [\mathfrak{a}]$ belonging to a particular family of ideals (e.g., the family of smooth ideals, or near-prime ideals). More precisely, let $\mathcal{S}$ be an arbitrary family of ideals, and $\mathcal{S}_B$ the family of B-smooth ideals. We describe an efficient algorithm that samples ideals $\mathfrak{b} \in [\mathfrak{a}]$ such that $\mathfrak{b} \in \mathcal{S} \cdot \mathcal{S}_B$ with probability proportional to the density of $\mathcal{S}$ within the set of all ideals.

The case where $\mathcal{S}$ is the set of prime ideals yields the family $\mathcal{S} \cdot \mathcal{S}_B$ of near-prime ideals, of particular interest in that it constitutes a dense family of efficiently factorable ideals. The case of smooth ideals $\mathcal{S} = \mathcal{S}_B$ regularly comes up in index-calculus algorithms (notably to compute class groups and unit groups), where it has long constituted a theoretical obstacle overcome only by heuristic arguments.

Contents

1. Introduction 2
2. Preliminaries 6
3. Acknowledgements 19

Part 1. Sampling ideals in a class: smooth, near-prime or otherwise 21
4. Introduction 21
5. Background on the Arakelov class group 22
6. Random walks in Arakelov ray class groups 26
7. Average densities and sampling 28
8. Uniform sampling in a box intersected with an ideal lattice 34
9. Ideal sampling 45
10. Properties of the ideal sampling algorithm 57
11. Estimation of the factor $\mathcal{N}(\mathfrak{m}_0)/\phi(\mathfrak{m}_0)$ 65

Date: December 1, 2025.

Part 2. Rigorous computation of class groups and unit groups 67
12. Introduction 67
13. Specialized ideal sampling algorithm 71
14. Density of smooth ideals 74
15. Background on $\mathbb{S}$-unit lattices 75
16. The generating and covering radius of the log-$\mathbb{S}$-unit lattice 77
17. Generating a single $\mathbb{S}$-unit 83
18. Obtaining a generating set of $\mathbb{S}$-units 89
19. Post-processing phase 100
20. Generating the exceptional $\mathbb{S}$-units 103
21. Full algorithm, final theorem and discussion 105

Part 3. Provable lattice reduction techniques 114
22. Introduction 114
23. Buchmann-Kessler-Pohst algorithm 115
24. Computing the determinant of an approximate basis 121
25. Post-processing in the $\mathbb{S}$-unit computation 124
26. Lattice preliminaries 126
27. The BKZ algorithm on integer bases 130
28. The BKZ algorithm on approximate bases 134
References 142

Part 4. Appendix 145
A.1. Proof of Theorem 6.4, the random walk theorem 145
A.2. Sampling of random prime ideals 150
A.3. Upper bound on the nth successive minimum of $\mathcal{O}_K$ by Bhargava et al. 152
A.4. Upper bound on a defining polynomial of a number field 153

1. Introduction

Many number theoretic algorithms resort to heuristic assumptions for their analysis. This issue concerns even the most fundamental problems of the field, such as the computation of class groups in subexponential time. This persistent need for heuristic assumptions often stems from a step of this form: given an ideal class $[\mathfrak{a}]$ of a number field K, find a representative $\mathfrak{b} \in [\mathfrak{a}]$ belonging to a particular family $\mathcal{S}$ of ideals (for instance, the family of smooth ideals). It is relatively simple to design an algorithm for this task: sample a random representative $\mathfrak{b} \in [\mathfrak{a}]$, and hope that it belongs to the desired family $\mathcal{S}$. One then heuristically argues that the probability that $\mathfrak{b} \in \mathcal{S}$ should be proportional to the density of $\mathcal{S}$. For instance, the subexponential density of smooth ideals heuristically implies that one can find smooth representatives in subexponential time. This is the heart of state-of-the-art algorithms to compute class groups, unit groups, or generators of principal ideals in number fields [10, 17, 51], and has long constituted a theoretical obstacle overcome only by heuristic arguments (with the exception of quadratic fields [38]).

In the first part of this paper, we propose a general strategy to solve these ideal sampling tasks rigorously and efficiently, assuming only the extended Riemann hypothesis (henceforth, ERH). In the second part, we apply this new technique to present the first algorithm for computing class groups and unit groups of arbitrary number fields that provably runs in probabilistic subexponential time. These two parts of the paper can be read essentially independently.

Part 3 consists of roughly two subjects: an slightly extended analysis and an application of a known provable variant of the BKZ algorithm for ideal lattices [39], which is used in Part 1 for lattice basis reduction; and an extended analysis of the algorithm of Buchmann, Pohst and Kessler [19, 18] which is required for the post-processing part in the class group and unit group computation in Part 2. These extra analyses are necessary to apply these known results to the specific use-cases of the present work.

Main result of Part 1: sampling ideals. Let $\mathcal{S}$ be an arbitrary family of ideals, and $\mathcal{S}_B$ the family of B-smooth ideals (i.e., products of prime ideals of norm at most B). In the first part of the paper, we describe an efficient algorithm that samples $\mathfrak{b} \in [\mathfrak{a}]$ such that $\mathfrak{b} \in \mathcal{S} \cdot \mathcal{S}_B$ with probability proportional to the density of $\mathcal{S}$. The set $\mathcal{S}_B$ is used to randomize the input, and B can be chosen as small as $(\log|\Delta_K|)^{O(1)}$, where Δ_K is the discriminant of the field K. This result is formalized in Theorem 9.5, page 47, and allows to work with arbitrary ray class groups, and to restrict $\mathcal{S}_B$ to ideals whose prime factors fall in a prescribed subgroup.

For concreteness, Theorem 1.1 below is a specialization of Theorem 9.5 to the simplest case, without ray nor subgroups. Here, $\delta_{\mathcal{S}}[r^n]$ is the local density of $\mathcal{S}$ (Definition 2.11), i.e., the proportion of ideals of norm about r^n that belong to $\mathcal{S}$.

Theorem 1.1 (ERH)**.** *Assuming ERH, there is a randomized algorithm $\mathcal{A}$ such that the following holds. Let K be a number field, with degree n, discriminant Δ_K, and let an LLL-reduced basis of the ring of integers $\mathcal{O}_K$ be given. Let $\mathfrak{a} \subseteq \mathcal{O}_K$ be an integral ideal. Let $\varepsilon \in \mathbb{R}_{>0}$, let $\mathsf{b} \geq 2$ be an integer, and let $r \geq 48 \cdot \mathsf{b}^{2n/\mathsf{b}} \cdot n^{7/2} \cdot |\Delta_K|^{\frac{3}{2n}} \cdot \mathcal{N}(\mathfrak{a})^{\frac{1}{n}}$.*

Given the above data, the algorithm $\mathcal{A}$ outputs $\beta \in \mathfrak{a}$ such that $(\beta) \cdot \mathfrak{a}^{-1} \in \mathcal{S} \cdot \mathcal{S}_B$ with probability at least $\delta_{\mathcal{S}}[r^n]/3 - \varepsilon$, for some smoothness bound $B = (\log|\Delta_K| + \log\log(1/\varepsilon))^{O(1)}$ and for any set $\mathcal{S}$ of integral ideals. Furthermore, the algorithm runs in expected polynomial time in $\log|\Delta_K|$, in $\log(\mathcal{N}(\mathfrak{a}))$, in $\log(1/\varepsilon)$, in b^{b}, and in the length of the input.

Remark 1.2. *Note that the algorithm is described in a slightly different way than the above discussion: given $\mathfrak{a}$, we find $\beta \in \mathfrak{a}$ such that $\beta\mathfrak{a}^{-1} \in \mathcal{S} \cdot \mathcal{S}_B$. The ideal $\beta\mathfrak{a}^{-1}$ is in the inverse class of $\mathfrak{a}$, so up to an inversion, this problem is equivalent to the ideal sampling problem discussed above.*

Technique. The folklore strategy to solve ideal sampling tasks is the following. The input ideal $\mathfrak{a}$ can be seen as a lattice, via the Minkowski embedding. One may find a reasonably short basis of $\mathfrak{a}$ (for instance, by means of LLL [52]), which then allows one to repeatedly sample reasonably short random elements $\beta \in \mathfrak{a}$, until the ideal $\mathfrak{b} = \beta\mathfrak{a}^{-1}$ belongs to the desired family $\mathcal{S}$. One then typically argues (heuristically!) that the probability of success is proportional to the density of $\mathcal{S}$.

To obtain a rigorous sampling algorithm, we proceed in two steps. First, we prove that a fairly straightforward strategy as above indeed has the desired probability of success *when the input $\mathfrak{a}$ is treated as a random ideal lattice with uniformly random Arakelov class*. More precisely, we prove in Theorem 7.1 that there is a reasonably small box $\mathcal{B}$ (in the embedding space) such that the *expected* density (over the randomness of $\mathfrak{a}$) of elements $\beta \in \mathfrak{a} \cap \mathcal{B}$ such that $\beta\mathfrak{a}^{-1} \in \mathcal{S}$ is proportional to the density of $\mathcal{S}$.

Second, we deal with arbitrary input $\mathfrak{a}$ by randomizing its Arakelov class via a generalization of the random walks introduced in [14]. Concretely, the input $\mathfrak{a}$ is multiplied by random ideals of small prime norm (the *discrete* part of the random walk), and is randomly distorted according to some Gaussian distribution (the *continuous* part of the random walk). We prove in Corollary 6.5 that the result is close to uniformly distributed in the Arakelov ray class group. The discrete part of the random walk introduces small prime factors, hence our method samples ideals in $\mathcal{S} \cdot \mathcal{S}_B$ instead of $\mathcal{S}$. In all applications we are aware of, $\mathcal{S} = \mathcal{S} \cdot \mathcal{S}_B$.

Main result of Part 2: computing class groups and unit groups. The case of smooth ideals $\mathcal{S} = \mathcal{S}_B$ regularly comes up in index-calculus algorithms, such as the aforementioned algorithms for computing class groups or unit groups. In these cases, this sampling task is not the only source of heuristics, so significantly more work is required to obtain a rigorous algorithm. This is the object of the second part of the article.

Let K be a number field of degree n and discriminant Δ_K. The determination of the structure of its class group $\mathrm{Cl}(K)$, together with a system of fundamental units, is one of the main problems of computational number theory [22, p. 217]. It has long been believed that this task can be solved in probabilistic subexponential time. Such algorithms have been described and analyzed under a variety of heuristic assumptions [17, 10]. Despite decades of investigation, only imaginary quadratic fields have been amenable to a rigorous analysis [38], assuming ERH. The history of class group computation is discussed in further detail in Section 12.2. In Part 2 of this paper, we present the first general algorithm for this problem that provably runs in probabilistic subexponential time, assuming ERH. We use the classical L-notation

$$L_x(\alpha, c) = \exp\left((c + o(1))(\log x)^{\alpha}(\log\log x)^{1-\alpha}\right),$$

and $L_x(\alpha) = L_x(\alpha, O(1))$. We prove the following theorem.

Theorem 1.3 (ERH)**.** *There is a probabilistic algorithm which, on input a number field K of degree n and discriminant Δ_K and an LLL-reduced basis of its ring of integers, computes its ideal class group and a compact representation of a fundamental system of units, and runs in expected time polynomial in the length of the input, in $L_{|\Delta_K|}(1/2)$, in $L_{n^n}(2/3)$, and in $\min(\rho_K, L_{|\Delta_K|}(2/3 + o(1)))$, where ρ_K is the residue at 1 of the Dedekind zeta function ζ_K.*

It has been conjectured since Buchmann's 1988 heuristic algorithm [17] that this problem can be solved in subexponential time $L_{|\Delta_K|}(1/2)$ for any family of fields of fixed degree. Theorem 1.3, together with the upper bound $\rho_K = (\log|\Delta_K|)^{O(n)}$ (see Equation (5) below), implies this conjecture, assuming ERH.

Then, it was conjectured by Biasse and Fieker's 2014 algorithm [10] that this problem can be solved in subexponential time even for varying degree. Again, Theorem 1.3 implies this conjecture, assuming ERH. However, Biasse and Fieker conjectured a complexity as in Theorem 1.3 where the quantity ρ_K is replaced with $L_{n^n}(2/3)$. In our analysis, the quantity ρ_K arises from the best known estimates on the density of bounded smooth ideals. It seems ρ_K should appear in the same way in the heuristic complexity of [10], unless one expects a better bound on the density of smooth ideals.

Computing $\mathbb{S}$*-units.* While we stated our main result as an algorithm for computing units and class groups in Theorem 1.3, our algorithm actually does slightly more than that: it computes the so-called Log-$\mathbb{S}$-unit lattice for any set $\mathbb{S}$ of prime ideals. It is well known that such an algorithm for $\mathbb{S}$-units can be used to compute the class group and the unit group. Combined with Theorem 17.7, which allows to decompose any integral ideal as a product of prime ideals in a sufficiently large set $\mathbb{S}$, this can also be used to solve other algorithmic problems, such as the principal ideal problem, or the class group discrete logarithm problem.

Main results of Part 3: Detailed analyses of provable lattice techniques. Part 3 consists of an extended analysis of two already known results in lattice theory; namely the existence of a BKZ-algorithm variant [39] that has a provable running time, and the existence of an LLL-algorithm variant (called the Buchmann-Pohst-Kessler algorithm [19, 18]) that has a relatively well numerical stability, so that it can be used on 'approximated' bases. The extended analysis consists, in the case of the BKZ-algorithm, mainly of making explicit the techniques described in [39, Section 3, 'Cost of BKZ"] and applying this BKZ-variant to ideal lattices. In the case of the Buchmann-Pohst-Kessler algorithm [19, 18], the present work required an extended analysis because in our use-case (in contrast to theirs) some lattice invariants like the rank and the determinant are unknown, thus requiring a slight extension to their algorithm.

The BKZ-variant a with provable running time is used in Part 1 for lattice basis reduction of ideal lattices, and the extended algorithm of Buchmann, Pohst and Kessler is required for the post-processing part in the class group and unit group computation in Part 2. Note that both of these algorithms being non-heuristic and having a provable running time is essential for the main results of the present work: a rigorous analysis of a common number-theoretic technique (Part 1) and a rigorous algorithm for computing the unit group and the class group that has a provable upper bound on the run time (Part 2).

Further applications. Sampling smooth ideals is a task that regularly arises in computational number theory. In Part 2, we focus on the problem of class group and unit group computation, but it is more generally a common component of index-calculus algorithms, like the general number field sieve for integer factorization or the computation of discrete logarithms in finite fields. We do not investigate this direction further in the present paper.

Applying the method of Part 1 to the case where $\mathcal{S}$ is the set of prime ideals allows one to sample in the family $\mathcal{S} \cdot \mathcal{S}_B$ of near-prime ideals, of particular interest in that it constitutes a dense family of efficiently factorable ideals. Therefore, our sampling method provides a rigorous way to transform any ideal $\mathfrak{a}$ into an equivalent ideal $\mathfrak{b}$ of known factorization. Obtaining such factorable ideals (or elements) is a key step in algorithms to compute power residue symbols. Specifically, it allows to perform the 'principalization step' in [16, §5.2] efficiently. The first author of the present article has developed this idea in his PhD dissertation [12], applying the main result of Part 1 to construct the first polynomial time algorithm to compute power residue symbols.

Related work. The aforementioned difficulties of ideal sampling and class group computation have already been overcome in the special case of *imaginary quadratic* number fields. Building on a result of Seysen [70], Hafner and McCurley [38] gave a provable algorithm for computing class groups and unit groups of imaginary quadratic fields, assuming ERH. This case distinguishes itself by the finiteness of the unit group and the existence of *reduced* representatives of ideal classes. This algorithm exploits random walks in the class group to find B-smooth *principal* ideals. The idea of performing a random walk in the class group was reused in the algorithms of Buchmann [17] and Biasse and Fieker [10], in a heuristic way. Rather than random walks in class groups, our method exploits the much richer Arakelov (ray) class groups. Random walks in Arakelov class groups were first studied in [14] to prove the random self-reducibility of computational problems in ideal lattices. Their technique to study the convergence of these walks plays a key role in the present paper. We note that Schoof [69] rephrased Buchmann's algorithm in terms of Arakelov theory, and we borrow from his formalism.

2. Preliminaries

2.1. **Notation.**

We denote by $\mathbb{N}, \mathbb{Z}, \mathbb{Q}, \mathbb{R}, \mathbb{C}$ the natural numbers, the integers, the rationals, the real numbers, and the complex numbers respectively. The notation log refers to logarithms in base e, whereas the notation $\log_2$ denotes logarithms in base 2. For finite sets X we denote by $|X|$ the number of elements in X. For infinite sets X with a well-defined volume, we use both notations $\mathrm{Vol}(X) = |X|$ for the volume of X. The transpose of a matrix M is denoted by $M^\top$. For a ring R, we write $R^\times$ the set of invertible elements of R, and $R^* := R \setminus \{0\}$.

We use the classic big O and Ω asymptotic notations, and all hidden constants are absolute (in particular, they never depend on the choice of a field K). We also use the notation $f = \mathsf{poly}(g)$ as a synonym for $f = g^{O(1)}$. We use the notation O_ε to signify that the hidden constants depend on ε. As already mentioned, to denote the running time of subexponential algorithms, we use the asymptotic L-notation

$$L_x(\alpha, c) = \exp\left((c + o(1))(\log x)^\alpha (\log\log x)^{1-\alpha}\right),$$

where the $o(1)$ is asymptotic in x (and does not depend on other parameters). We also write $L_x(\alpha) = L_x(\alpha, O(1))$.

For a real vector space $V \subseteq \mathbb{R}^m$, we consider the Euclidean norm $\|\cdot\|$, and the infinity-norm $\|\cdot\|_\infty$. We sometimes write $\|\cdot\|_2$ for $\|\cdot\|$ to emphasize the type of norm. We will occasionally use the notation $\mathcal{B} = \mathcal{B}_2$ and $\mathcal{B}_\infty$ for the unit ball with respect to the Euclidean and infinity norm respectively. In particular, the Euclidean ball of radius r in V is typically denoted

$$r\mathcal{B} = \{\mathbf{v} \in V \mid \|\mathbf{v}\| < r\}.$$

The vector space V is either clear from context or explicitly mentioned when introducing $\mathcal{B}$ or $\mathcal{B}_\infty$.

2.2. **The Extended Riemann Hypothesis.**

All statements that mention (ERH), such as Theorem 1.1, assume the *Extended Riemann Hypothesis*, which refers to the Riemann Hypothesis for Hecke L-functions (see [44, §5.7]).

2.3. **Euclidean lattices.**

A lattice Λ is a discrete subgroup of a real vector space $V = \mathbb{R}^m$. We write $\mathrm{span}_\mathbb{R}(\Lambda)$ the real vector subspace of V spanned by the vectors of Λ. The rank of the lattice is the dimension of $\mathrm{span}_\mathbb{R}(\Lambda)$, and we say the lattice is full-rank if $\mathrm{span}_\mathbb{R}(\Lambda) = V$. A lattice (of rank n) can be represented by a basis $\mathbf{B} = (\mathbf{b}_1, \cdots, \mathbf{b}_n)$ such that $\Lambda = \{\sum_i x_i \mathbf{b}_i\, ,\, x_i \in \mathbb{Z}\}$. For a given basis $\mathbf{B} \in \mathbb{R}^{m\times n}$ we denote by $\mathcal{L}(\mathbf{B})$ the lattice spanned by the columns of this basis.

Geometric invariants. The covolume of Λ, denoted by $\mathrm{Vol}(\Lambda)$, is the volume of the quotient $\mathrm{span}_{\mathbb{R}}(\Lambda)/\Lambda$. If $\mathbf{B} \in \mathbb{R}^{m\times n}$ is a basis of Λ (with column vectors), then the covolume of Λ is equal to $\mathrm{Vol}(\Lambda) = \sqrt{\det(\mathbf{B}^\top \mathbf{B})}$. This quantity is also the volume of any fundamental domain of the lattice. One such domain is of special interest: the Voronoi cell.

Definition 2.1. *Let $\Lambda \subseteq V$ be a full-rank lattice. We denote by*

$$\mathcal{V}_0(\Lambda) = \{\mathbf{x} \in V \mid \|\mathbf{x}\| \le \|\mathbf{x} - \ell\| \text{ for all } \ell \in \Lambda\}$$

the Voronoi cell of Λ around zero. It is a fundamental domain for the lattice Λ (up to a set of 'faces' of measure zero), thus has volume $\mathrm{Vol}(\Lambda)$.

The i-th successive minimum of Λ is denoted by $\lambda_i(\Lambda)$. More precisely, each $\lambda_i(\Lambda)$ is the smallest real number such that there exist at least i linearly independent vectors of euclidean norm at most $\lambda_i(\Lambda)$ in Λ. We call $\lambda_1(\Lambda)$ the first minimum, and $\lambda_n(\Lambda)$ the last minimum (where n is the rank of the lattice). The covering radius $\mathrm{cov}(\Lambda)$ is the smallest $r > 0$ such that for any element $\mathbf{x} \in \mathrm{span}_{\mathbb{R}}(\Lambda)$ there exists a lattice point at distance at most r from $\mathbf{x}$. The analogous notions with respect to the maximum norm $\|\cdot\|_\infty$ instead of the Euclidean norm are denoted by $\lambda_i^{(\infty)}(\Lambda)$ and $\mathrm{cov}_\infty(\Lambda)$.

The following notion, the *generating radius* $\varrho(\Lambda)$, is closely related to $\lambda_n(\Lambda)$ and $\mathrm{cov}(\Lambda)$, but not as standard.

Definition 2.2 (Generating radius)**.** *For a lattice Λ, the* generating radius $\varrho(\Lambda)$ *is the smallest real number such that the set of all vectors of Λ of euclidean norm $\le \varrho(\Lambda)$ generates the lattice Λ (as a $\mathbb{Z}$-module). In other words,*

$$\varrho(\Lambda) = \min\Big(r > 0 \,|\, \{\mathbf{x} \in \Lambda \,|\, \|\mathbf{x}\| \le r\} \text{ generates } \Lambda\Big).$$

Lemma 2.3. *For any lattice Λ, we have*

$$\lambda_n(\Lambda) \le \varrho(\Lambda) \le 2\cdot \mathrm{cov}(\Lambda) \le \sqrt{n}\cdot\lambda_n(\Lambda).$$

Proof. The left-most inequality follows from the very definitions of $\lambda_n(\Lambda)$ and $\varrho(\Lambda)$, and the right-most inequality can be found in [54, Theorem 7.9]. We prove the middle inequality by using the concept of *Voronoi-relevant vectors* of the lattice Λ. The Voronoi cell $\mathcal{V}_0(\Lambda)$ of L contains all vectors of $\mathrm{span}_{\mathbb{R}}(\Lambda)$ that are closer or equally close to 0 than to any other lattice point (see Definition 2.1). We know (e.g., [54, Proposition 8.4]) that $\mathcal{V}_0(\Lambda)$ is compact and convex and that $\mathrm{Vol}_{\mathrm{span}_{\mathbb{R}}(\Lambda)}(\mathcal{V}_0(\Lambda)) = \mathrm{Vol}(\Lambda)$. From the definition of the Voronoi cell and of the covering radius of Λ, we also know that all $\mathbf{x} \in \mathcal{V}_0(\Lambda)$ satisfy $\|\mathbf{x}\| \le \mathrm{cov}(\Lambda)$ (see also [54, Proposition 8.4]).

The Voronoi-relevant vectors of Λ are the vectors $S_{\mathcal{V}_0} = \{\mathbf{v} \in \Lambda \,|\, \exists \mathbf{x} \in \mathcal{V}_0(\Lambda), \|\mathbf{x}\| = \|\mathbf{x}-\mathbf{v}\|\}$; these are the vectors defining the facets of the Voronoi cell. From the definition of these vectors, it holds that $\mathcal{V}_0(\Lambda) = \{\mathbf{x} \in \mathrm{span}_{\mathbb{R}}(\Lambda) \,|\, \forall \mathbf{v} \in S_{\mathcal{V}_0}, \ \|\mathbf{x}\| \le \|\mathbf{x} - \mathbf{v}\|\}$. Moreover, since $\|\mathbf{x}\| \le \mathrm{cov}(\Lambda)$ for all $\mathbf{x} \in \mathcal{V}_0(\Lambda)$, we have that $\|\mathbf{v}\| \le 2\,\mathrm{cov}(\Lambda)$ for all Voronoi-relevant vectors $\mathbf{v}$. We will show that $S_{\mathcal{V}_0}$ generates the lattice Λ, which will conclude the proof.

Let Λ' be the lattice generated by the vectors in $S_{\mathcal{V}_0}$. We know that $\Lambda' \subseteq \Lambda$ and that Λ' has rank n (since $\mathrm{span}_{\mathbb{R}}(\Lambda') = \mathrm{span}_{\mathbb{R}}(\mathcal{V}_0(\Lambda)) = \mathrm{span}_{\mathbb{R}}(\Lambda)$). Moreover, since $S_{\mathcal{V}_0} \subseteq \Lambda'$, we know that $\mathcal{V}_0(\Lambda') \subseteq \{\mathbf{x} \in \mathrm{span}_{\mathbb{R}}(\Lambda) \,|\, \forall \mathbf{v} \in S_{\mathcal{V}_0}, \|\mathbf{x}\| \le \|\mathbf{x} - \mathbf{v}\|\} = \mathcal{V}_0(\Lambda)$. From this we conclude that $\mathrm{Vol}(\Lambda') \le \mathrm{Vol}(\Lambda)$ and so $\Lambda' = \Lambda$ as desired. □

The next two results regarding counting lattice points in sets, Lemma 2.4 and Lemma 2.8, are folklore. We include proofs for completeness, as they play an important role in the article.

Estimation on the average *number of lattice points in a measurable volume.*

Lemma 2.4. *Let V be a Euclidean vector space and let $\Lambda \subseteq V$ be a full rank lattice. Let $S \subseteq V$ be a measurable set, and let $\mathbf{c} \in V/\Lambda$ be chosen uniformly. Then*

$$\mathop{\mathbb{E}}_{\mathbf{c} \leftarrow V/\Lambda}[|(\Lambda + \mathbf{c}) \cap S|] = \mathrm{Vol}(S)/\mathrm{Vol}(\Lambda).$$

Proof. By integrating the measurable indicator set 1_S, and choosing a fundamental domain F of V/Λ (which has volume $\mathrm{Vol}(\Lambda)$), we obtain

$$\mathop{\mathbb{E}}_{\mathbf{c} \leftarrow V/\Lambda}[|(\Lambda+\mathbf{c})\cap S|] = \frac{1}{\mathrm{Vol}(F)} \int_{\mathbf{c}\in F} \sum_{\ell \in \Lambda} 1_S(\mathbf{c}+\ell) d\mathbf{c} = \frac{1}{\mathrm{Vol}(\Lambda)} \int_{\mathbf{v}\in V} 1_S(\mathbf{v}) d\mathbf{v} = \frac{\mathrm{Vol}(S)}{\mathrm{Vol}(\Lambda)}.$$

□

Estimation on the number of lattice points in a convex measurable volume. For the ideal sampling algorithm (Algorithm 2) we need to efficiently sample in a shifted box (see Section 8). Lemma 2.8, which shares some similarities with [64, §4.2], provides means to estimate the number of lattice elements in such a box. This estimate is essential in the proof in Section 8. To prepare for the proof of this lemma, we will need some facts on Minkowski sums of sets.

Definition 2.5. *Let V be a Euclidean vector space. For two sets $X, Y \subseteq V$, we define the Minkowski sum $X \boxplus Y$ as follows.*

$$X \boxplus Y = \{\mathbf{x} + \mathbf{y} \mid \mathbf{x} \in X, \mathbf{y} \in Y\}.$$

For $c \in \mathbb{R}_{>0}$ we denote by cX the set

$$cX = \{c \cdot \mathbf{x} \mid \mathbf{x} \in X\}.$$

Lemma 2.6. *Let V be a Euclidean vector space and let $r, s > 0$ and let $X \subseteq V$ be a convex volume. Then*

$$(rX) \boxplus (sX) = (r+s)X.$$

Proof. We start with inclusion to the right. Suppose $\mathbf{y} \in (rX) \boxplus (sX)$, i.e., $\mathbf{y} = r\mathbf{x} + s\mathbf{x}'$ where $\mathbf{x}, \mathbf{x}' \in X$. Then $\frac{\mathbf{y}}{r+s} = \frac{r\mathbf{x}+s\mathbf{x}'}{r+s} \in X$, since it is a weighted average of two points in X and X is convex. So $\mathbf{y} \in (r+s)X$. Inclusion to the left holds because $\mathbf{y} \in (r+s)X$ means that $\mathbf{y} = (r+s)\mathbf{x} = r\mathbf{x} + s\mathbf{x} \in (rX) \boxplus (sX)$. □

Lemma 2.7. *Let V be a Euclidean vector space, let $r > 0$, let $X, Y \subseteq V$ be sets and let $S \subseteq V$ be a symmetric set, i.e., $\mathbf{x} \in S \Leftrightarrow -\mathbf{x} \in S$. Then*

$$(X \boxplus S) \cap Y \subseteq [X \cap (Y \boxplus S)] \boxplus S.$$

Proof. Suppose $\mathbf{x} + \mathbf{s} = \mathbf{y} \in (X \boxplus S) \cap Y$. Then $\mathbf{x} = \mathbf{y} - \mathbf{s} \in X \cap (Y \boxplus S)$, so $\mathbf{y} = \mathbf{x} + \mathbf{s} \in [X \cap (Y \boxplus S)] \boxplus S$. □

Lemma 2.8. *Let V be a n-dimensional Euclidean vector space, let $\Lambda \subseteq V$ be a full-rank lattice, let $X \subseteq V$ be a convex measurable volume for which $\mathcal{V}_0 \subseteq cX$ for* *some $c \in \mathbb{R}_{>0}$, where $\mathcal{V}_0$ is the Voronoi cell of Λ (see Definition 2.1). Then, for all $\mathbf{t}, \mathbf{t}' \in V$ and all $r > 2c$,*

$$|(\Lambda + \mathbf{t}) \cap r(X + \mathbf{t}')| \in [e^{-2nc/r}, e^{2nc/r}] \cdot \frac{r^n \cdot \mathrm{Vol}(X)}{\mathrm{Vol}(\Lambda)},$$

where $r(X + \mathbf{t}') = \{r \cdot (x + \mathbf{t}') \mid x \in X\}$ is the scaling of the (translated) set $X + \mathbf{t}'$ by $r \in \mathbb{R}_{>0}$.

Proof. As $|(\Lambda + \mathbf{t}) \cap (rX + r\mathbf{t}')| = |(\Lambda + \mathbf{t} - r\mathbf{t}') \cap rX|$, we just assume, without loss of generality, that $\mathbf{t}' = 0$. Note that $\mathcal{V}_0 \subseteq cX$, and that X is convex. So, by Lemma 2.6, we have $(rX) \boxplus \mathcal{V}_0 \subseteq (rX) \boxplus (cX) = (r+c)X$. Similarly, $(r-c)X \boxplus \mathcal{V}_0 \subseteq rX$. Therefore

$$[(\Lambda + \mathbf{t}) \cap rX] \boxplus \mathcal{V}_0 \subseteq (r + c)X. \tag{1}$$

Note that $\mathcal{V}_0$ is symmetric and $(\Lambda + \mathbf{t}) \boxplus \mathcal{V}_0 = V$, the whole vector space. So, by Lemma 2.7 and $(r - c)X \boxplus \mathcal{V}_0 \subseteq rX$,

$$\begin{aligned}(r - c)X &= [(\Lambda + \mathbf{t}) \boxplus \mathcal{V}_0] \cap (r - c)X & (2)\\ &\subseteq [(\Lambda + \mathbf{t}) \cap ((r - c)X \boxplus \mathcal{V}_0)] \boxplus \mathcal{V}_0 \subseteq [(\Lambda + \mathbf{t}) \cap rX] \boxplus \mathcal{V}_0 & (3)\end{aligned}$$

By Equations (1) and (3) and the fact that $\mathcal{V}_0$ is a fundamental domain of Λ with volume $\mathrm{Vol}(\Lambda)$, we obtain

$$(r - c)^n \,\mathrm{Vol}(X) \leq |(\Lambda + \mathbf{t}) \cap rX| \cdot \mathrm{Vol}(\Lambda) \leq (r + c)^n \,\mathrm{Vol}(X).$$

Dividing by $\mathrm{Vol}(\Lambda)$ and using the estimate $e^{-2nc/r} \leq (1 - c/r)^n \leq (1 + c/r)^n \leq e^{2nc/r}$ (note that $r > 2c$) we arrive at the final claim. □

2.4. **Number fields.** Throughout this paper, we consider a number field K of rank n over $\mathbb{Q}$, having ring of integers $\mathcal{O}_K$, discriminant Δ_K, regulator R_K, class number h_K and group of roots of unity μ_K. Additionally, we consider a *modulus*: a formal product $\mathfrak{m} = \mathfrak{m}_0 \mathfrak{m}_\infty$, where $\mathfrak{m}_0 \subseteq \mathcal{O}_K$ is an integral ideal and $\mathfrak{m}_\infty$ is a formal product of infinite places (see more details below). We know by Minkowski's theorem [56, pp. 261–264] that[1] $\log |\Delta_K| \geq \log(\pi/2)n \geq 0.4n$ for $n \geq 2$.

The number field K has n field embeddings into $\mathbb{C}$, which are divided in $n_\mathbb{R}$ real embeddings and $n_\mathbb{C}$ conjugate pairs of complex embeddings, with $n = n_\mathbb{R} + 2n_\mathbb{C}$. These embeddings combined yield the so-called Minkowski embedding

$$\begin{aligned} K &\longrightarrow K_\mathbb{R} \subseteq \bigoplus_{\sigma : K \hookrightarrow \mathbb{C}} \mathbb{C} \\ \alpha &\longmapsto (\sigma(\alpha))_\sigma, \end{aligned}$$

where

$$K_\mathbb{R} = \left\{ x \in \bigoplus_{\sigma : K \hookrightarrow \mathbb{C}} \mathbb{C} \;\middle|\; x_{\overline{\sigma}} = \overline{x_\sigma} \right\}.$$

Here, $\overline{\sigma}$ equals the conjugate embedding of σ whenever σ is a complex embedding and it is just σ itself whenever it is a real embedding. We index the components of the vectors in $K_\mathbb{R}$ by the embeddings of K, i.e., we write $x = (x_\sigma)_\sigma \in K_\mathbb{R}$. Embeddings up to conjugation are called infinite places, denoted by ν. With any embedding σ we denote by ν_σ the associated place; and for any place we choose a fixed embedding σ_ν. We will sometimes write $\sigma \mid \mathfrak{m}_\infty$ to mean that the associated

[1] We have $|\Delta_K| \geq \left(\frac{\pi}{4}\right)^n n^{2n}/(n!)^2 \geq \left(\frac{\pi}{2}\right)^n$ for $n \geq 2$, since $n^{2n}/(n!)^2 \geq 2^n$ for $n \geq 2$.

place ν_σ divides $\mathfrak{m}_\infty$. For any radius $r \in \mathbb{R}_{>0}$ we write $r\mathcal{B}_\infty = \{(x_\sigma)_\sigma \in K_\mathbb{R} \mid |x_\sigma| \leq r \text{ for all } \sigma\}$, if it is clear from the context that $K_\mathbb{R}$ is the vector space at hand.

Ideals. The group of fractional ideals of K is denoted by $\mathcal{I}_K$. Fractional ideals are denoted by $\mathfrak{a}, \mathfrak{b}, \ldots$, and the symbols $\mathfrak{p}, \mathfrak{q}$ are generally reserved for integral prime ideals of $\mathcal{O}_K$. Any $\mathfrak{a} \in \mathcal{I}_K$ factors uniquely as a product of prime ideals (with possibly negative exponents), and we denote by $\mathrm{ord}_\mathfrak{p}(\mathfrak{a})$ the exponent of $\mathfrak{p}$ in the factorization of $\mathfrak{a}$; by extension, for any elements $\alpha \in K^*$ we define $\mathrm{ord}_\mathfrak{p}(\alpha) = \mathrm{ord}_\mathfrak{p}(\mathfrak{a})$ for $\mathfrak{a} = (\alpha)$ the principal ideal generated by α. Two fractional ideals $\mathfrak{a}, \mathfrak{b}$ are said to be coprime if their unique decomposition into a product of prime ideals (with possibly negative coefficients) do not share any prime ideals. The algebraic norm of a fractional ideal $\mathfrak{a}$ or an element $\alpha \in K$ is denoted $\mathcal{N}(\mathfrak{a})$ and $\mathcal{N}(\alpha)$, respectively. When $\mathfrak{a} = (\alpha)$, we have $\mathcal{N}(\mathfrak{a}) = |\mathcal{N}(\alpha)|$.

The class number formula. Let h_K be the class number of K, R_K be its regulator and μ_K be its group of roots of unity. Let also ζ_K be the Dedekind zeta function of K, and ρ_K be its residue at 1. The class number formula [58, VII.§5, Cor 5.11] states that

$$\rho_K := \lim_{s\to 1}(s-1)\zeta_K(s) = \frac{2^{n_\mathbb{R}} \cdot (2\pi)^{n_\mathbb{C}} \cdot R_K \cdot h_K}{|\mu_K| \cdot \sqrt{|\Delta_K|}}. \tag{4}$$

The Dedekind residue $\rho_K \in \mathbb{R}_{>0}$ is bounded above[2] [53]: for $n = [K : \mathbb{Q}] > 1$, we have

$$\log(\rho_K) \leq (n-1) \cdot \log\left(\frac{e \log|\Delta_K|}{2(n-1)}\right) \leq \log|\Delta_K|. \tag{5}$$

It can be approximated in polynomial time up to some factor [5], and so can the related quantity $R_K \cdot h_K$. More precisely, we have the following proposition.

Proposition 2.9 (ERH)**.** *There exists a polynomial time algorithm (in $\log|\Delta_K|$) that takes as input any number field K, and an LLL-reduced basis of its ring of integers $\mathcal{O}_K$, and outputs $\rho_0 \in \mathbb{Q}$ and $\eta_0 \in \mathbb{Q}$ such that*

$$\rho_0 \in [\tfrac{3}{4}, \tfrac{5}{4}] \cdot \rho_K$$
$$\eta_0 \in [\tfrac{3}{4}, \tfrac{5}{4}] \cdot R_K \cdot h_K.$$

Density of ideals. In this article, we consider families of ideals, like smooth ideals or prime ideals. Given a specifically randomly generated ideal, we want to estimate the probability that it belongs to a given family. The notion of local density provides a certain approximation of this probability, for uniformly random ideals of bounded norm.

Definition 2.10. *For any set of ideals $\mathcal{S}$, we define $\mathcal{S}(t) = \{\mathfrak{b} \in \mathcal{S} \mid \mathcal{N}(\mathfrak{b}) \leq t\}$.*

Definition 2.11 (Local density of an ideal set)**.** *Let $x > 0$ a positive real number, and let $\mathcal{S}$ be a set of integral ideals of K. We define the* local density *of $\mathcal{S}$ at x as*

$$\delta_\mathcal{S}[x] = \min_{t\in[x/e^n, x]} \frac{|\mathcal{S}(t)|}{\rho_K \cdot t} = \min_{t\in[x/e^n, x]} \frac{|\{\mathfrak{b} \in \mathcal{S} \mid \mathcal{N}(\mathfrak{b}) \leq t\}|}{\rho_K \cdot t},$$

where $\rho_K = \lim_{s\to 1}(s-1)\zeta_K(s)$ (see Equation (4)).

[2] We use here that $\log(ex/2) \leq x$ for all $x > 0$ and the fact that $\log(|\Delta_K|)/(n-1) > 0$ for $n \geq 2$.

Note that the local density tends to the (asymptotic) 'natural density' of $\mathcal{S}$ as $x \to \infty$, since $|\{\mathfrak{a} \subseteq \mathcal{O}_K \mid \mathcal{N}(\mathfrak{a}) < t\}| \sim \rho_K \cdot t$ [60, §9.5]. However, this notion of natural density is not fine enough as it is simply 0 for families of interest like smooth ideals or prime ideals.

Ray. The group of fractional ideals coprime with the finite part $\mathfrak{m}_0$ of the modulus $\mathfrak{m}$ is denoted by $\mathcal{I}_K^{\mathfrak{m}}$. We denote by

$$K^{\mathfrak{m},1} = \langle \alpha \in \mathcal{O}_K \mid \alpha \equiv 1 \bmod \mathfrak{m}_0 \text{ and } \sigma(\alpha) > 0 \text{ for all real } \sigma \mid \mathfrak{m}_\infty \rangle$$

the *ray* modulo $\mathfrak{m}$, a multiplicative subgroup of K^*. Here, the notation $\langle \cdot \rangle$ means that $K^{\mathfrak{m},1}$ is multiplicatively generated by the elements $\alpha \in \mathcal{O}_K$ satisfying $\alpha \equiv 1 \bmod \mathfrak{m}_0$ and $\sigma(\alpha) > 0$ for all real $\sigma \mid \mathfrak{m}_\infty$, which also includes fractions.

The *ideal ray class group* $\mathrm{Cl}_K^{\mathfrak{m}}$ modulo $\mathfrak{m}$ is defined as the quotient of $\mathcal{I}_K^{\mathfrak{m}}$ by the subgroup $\mathrm{Princ}_K^{\mathfrak{m}} := \{(\alpha) \in \mathcal{I}_K^{\mathfrak{m}} \mid \alpha \in K^{\mathfrak{m},1}\}$. One retrieves the (ordinary) ideal class group by taking an empty modulus $\mathfrak{m} = \mathcal{O}_K$, for which $K^{\mathfrak{m},1} = K^*$. We also define $K^{\mathfrak{m}} = \langle \alpha \in \mathcal{O}_K \mid (\alpha) + \mathfrak{m}_0 = \mathcal{O}_K \rangle$, the multiplicative subgroup of K^* generated by elements coprime to $\mathfrak{m}_0$. Note that $K^{\mathfrak{m},1} \subseteq K^{\mathfrak{m}}$ and that $K^{\mathfrak{m}} = K^*$ if $\mathfrak{m}_0 = \mathcal{O}_K$. The number of *real places* $\nu \mid \mathfrak{m}_\infty$ is denoted by $|\mathfrak{m}_{\mathbb{R}}|$. We denote $\mathcal{N}(\mathfrak{m}) = \mathcal{N}(\mathfrak{m}_0) \cdot 2^{|\mathfrak{m}_{\mathbb{R}}|}$. We will also use the generalized Euler totient $\phi(\mathfrak{m}_0) = |K^{\mathfrak{m}_0}/K^{\mathfrak{m}_0,1}|$ which equals $|(\mathcal{O}_K/\mathfrak{m}_0)^\times|$ for $\mathfrak{m}_0 \subsetneq \mathcal{O}_K$ and equals 1 for $\mathfrak{m}_0 = \mathcal{O}_K$.

We denote $K_{\mathbb{R}}^{\mathfrak{m}_\infty} = \{(x_\sigma)_\sigma \in K_{\mathbb{R}} \mid x_\sigma > 0 \text{ for real } \sigma \mid \mathfrak{m}_\infty\}$ for the 'positive part' of $K_{\mathbb{R}}$ with respect to the modulus $\mathfrak{m}$. For any $\tau \in K^{\mathfrak{m}}$, we denote $\tau K_{\mathbb{R}}^{\mathfrak{m}_\infty} = \{(x_\sigma)_\sigma \in K_{\mathbb{R}} \mid x_\sigma/\sigma(\tau) > 0 \text{ for real } \sigma \mid \mathfrak{m}_\infty\}$, which is the part of $K_{\mathbb{R}}$ that has the same sign as τ at the real embeddings $\sigma \mid \mathfrak{m}_\infty$.

2.5. **Ideal lattices.** Ideals can be viewed as lattices in the real vector space $K_{\mathbb{R}}$, where $K_{\mathbb{R}}$ has its (Euclidean or maximum) norm inherited from the complex vector space it lives in. Explicitly, the Euclidean and maximum norm of $\alpha \in K$ are respectively defined by the rules $\|\alpha\|^2 = \sum_\sigma |\sigma(\alpha)|^2$ and $\|\alpha\|_\infty = \max_\sigma |\sigma(\alpha)|$, where σ ranges over all embeddings $K \to \mathbb{C}$.

For any ideal $\mathfrak{a}$ of K, we define the associated lattice $\mathfrak{a} \subseteq K_{\mathbb{R}}$ to be the image of $\mathfrak{a} \subseteq K$ under the Minkowski embedding, which is clearly a discrete additive subgroup of $K_{\mathbb{R}}$. Abusing notation, we denote both the ideal and the associated lattice with the same symbol $\mathfrak{a}$. In particular, $\mathcal{O}_K$ is a lattice. Note that we have $\mathrm{Vol}(\mathfrak{a}) = \sqrt{|\Delta_K|}\mathcal{N}(\mathfrak{a})$ for ideals $\mathfrak{a} \in \mathcal{I}_K$. The notion of *ideal lattices* extends to a larger family of lattices in $K_{\mathbb{R}}$ as follows.

Definition 2.12 (Ideal lattices)**.** *Let K be a number field with ring of integers $\mathcal{O}_K$. An* ideal lattice *of K is a lattice in $K_{\mathbb{R}}$ of the form $x\mathfrak{a}$ where $x \in K_{\mathbb{R}}^\times$ is invertible and $\mathfrak{a}$ is a fractional ideal of K. We denote the group of ideal lattices by* IdLat_K.

The set of ideal lattices is a group with product $(x\mathfrak{a})(y\mathfrak{b}) = (xy)(\mathfrak{a}\mathfrak{b})$, inverse $x^{-1}\mathfrak{a}^{-1}$ and unit $\mathcal{O}_K$.

Bounds on invariants of ideal lattices. Denote $\Gamma(\Lambda) = \lambda_n(\Lambda)/\lambda_1(\Lambda)$, and define, for a fixed number field K:

$$\Gamma_K = \sup_{x\mathfrak{a} \in \mathrm{IdLat}_K} \Gamma(x\mathfrak{a}) \tag{6}$$

We have the following bounds.

Lemma 2.13. *For any ideal lattice $x\mathfrak{a} \in \mathrm{IdLat}_K$,*

(i) $\lambda_1(x\mathfrak{a}) \geq \sqrt{n} \cdot \mathcal{N}(x\mathfrak{a})^{1/n}$.
(ii) $\lambda_n(\mathcal{O}_K)/\sqrt{n} \leq \Gamma_K \leq \lambda_n^\infty(\mathcal{O}_K) \leq |\Delta_K|^{1/n}$.
(iii) *For cyclotomic number fields* K, $\Gamma_K = 1$.
(iv) $\lambda_n(x\mathfrak{a}) \leq \sqrt{n} \cdot \Gamma_K \cdot \mathrm{Vol}(x\mathfrak{a})^{1/n}$.
(v) $\mathrm{cov}_\infty(x\mathfrak{a}) \leq \mathrm{cov}(x\mathfrak{a}) \leq n/2 \cdot \Gamma_K \cdot \mathrm{Vol}(x\mathfrak{a})^{1/n}$.

Proof. The first item follows from the fact that for any non-zero element $z \in x\mathfrak{a}$, it holds that $\mathcal{N}(x\mathfrak{a}) \leq |\mathcal{N}(z)|$ and that $|\mathcal{N}(z)|^{2/n} \leq \frac{1}{n} \cdot \|z\|^2$ by the inequality of arithmetic and geometric means, applied to the $(|\sigma(z)|^2)_\sigma$. Applying this to a z reaching $\lambda_1(x\mathfrak{a})$ yields $\lambda_1(x\mathfrak{a}) \geq \sqrt{n}|\mathcal{N}(z)|^{1/n} \geq \sqrt{n}\mathcal{N}(x\mathfrak{a})^{1/n}$. For the second item, the inequality $\lambda_n(\mathcal{O}_K)/\sqrt{n} \leq \Gamma_K$ follows from the definition of $\Gamma(\mathcal{O}_K)$ and the fact that $\lambda_1(\mathcal{O}_K) = \sqrt{n}$ (using that $\lambda_1(\mathcal{O}_K) \geq \sqrt{n}$ by the first item and that this lower bound is reached by $1 \in \mathcal{O}_K$). To obtain the bound $\Gamma_K \leq \lambda_n^\infty(\mathcal{O}_K)$, pick an arbitrary ideal lattice $x\mathfrak{a} \in \mathrm{IdLat}_K$ and choose a shortest element $x\alpha \in x\mathfrak{a}$ with $\alpha \in \mathfrak{a} \in \mathcal{I}_K$. That means $\|x\alpha\| = \lambda_1(x\mathfrak{a})$. Then $x\mathfrak{a} \supset x \cdot (\alpha)$, and therefore,

$$\lambda_n(x\mathfrak{a}) \leq \lambda_n(x \cdot \alpha\mathcal{O}_K) \leq \|x\alpha\| \cdot \lambda_n^\infty(\mathcal{O}_K) \leq \lambda_1(x\mathfrak{a}) \cdot \lambda_n^\infty(\mathcal{O}_K). \tag{7}$$

The bound $\lambda_n^\infty(\mathcal{O}_K) \leq |\Delta_K|^{1/n}$ is obtained from [8, Theorem 3.1] and is tailored to our purposes in Theorem A.3.1. Part (iii) follows from part (ii) and the fact that $\|\zeta\| = \|1\|$ for roots of unity $\zeta \in K$. Part (iv) is essentially Minkowski's bound $\lambda_1(x\mathfrak{a}) \leq \sqrt{n}\,\mathrm{Vol}(x\mathfrak{a})^{1/n}$ combined with the definition of Γ_K. Finally, the last item follows from the fact that $\mathrm{cov}(\Lambda) \leq \sqrt{n}/2 \cdot \lambda_n(\Lambda)$ [54, Theorem 7.9]. □

2.6. **Representation of elements and ideals.** We assume throughout this paper that the number field K is represented by a monic irreducible polynomial $f \in \mathbb{Z}[x]$ satisfying $\mathrm{size}(f) := \sum_i \log_2 |f_i| \leq \mathsf{poly}(\log |\Delta_K|)$. This restriction is very mild, indeed we can prove that such a polynomial always exists (see Appendix A.4) and there are *heuristic* polynomial time algorithms computing such polynomials, e.g., [24, 35].

Additionally, we assume throughout this paper that we know an LLL-reduced basis $(\mathbf{b}_1, \cdots, \mathbf{b}_n)$ of the ring of integers $\mathcal{O}_K$ of K. Such a basis has vectors (represented as polynomials in $\mathbb{Q}[x]/(f(x))$) whose size is polynomially bounded[3] in $\log |\Delta_K|$. Note that such a basis can be computed in unconditional, probabilistic subexponential time in $\mathrm{size}(f)$ (by factoring $\mathrm{disc}(f)$ with [65], computing a basis of $\mathcal{O}_K$ with [20, Theorem 1.4], and reducing it with the LLL algorithm [52]).

For the main result of Part 1 such a basis is not required per se (see the discussion in Section 9.5), as any sub-order of $\mathcal{O}_K$ would suffice as well. This choice for a basis of the ring of integers is done purely because it simplifies the description and analysis of this main result.

[3] Let θ denote the class of x in $\mathbb{Q}[x]/(f(x))$, and write $b_i = \sum_{j=1}^n q_{i,j}\theta^{j-1}$ for the elements of the LLL-reduced basis of $\mathcal{O}_K$ (with $q_{i,j} \in \mathbb{Q}$). Let $\mathbf{B} \in \mathbb{C}^{n\times n}$ be the matrix whose columns correspond to the Minkowski embedding of the b_i's, and $\mathbf{T}$ be the one corresponding to the θ^{j-1}'s. Then $(q_{i,j})_{i,j} = \mathbf{T}^{-1} \cdot \mathbf{B}$. By LLL reducedness of the b_i's, it holds that the log of the coefficients of $\mathbf{B}$ are polynomially bounded in $\log |\Delta_K|$. The coefficients of $\mathbf{T}^{-1}$ correspond to the coefficients of the Lagrange polynomials associated to the roots of f, so their logarithm is also polynomially bounded in $\mathrm{size}(f) = \mathsf{poly}(\log |\Delta_K|)$ (using e.g., Mignotte's root separation lower bound and Cauchy's upper bound on the roots). Hence, we conclude that $|\log(q_{i,j})| \leq \mathsf{poly}(\log |\Delta_K|)$.

It remains to prove that the denominators of the q_i are bounded. As the integral basis $(1, \theta, \ldots, \theta^{n-1})$ has discriminant $\Delta(f)$, any denominator of q_i must divide $\Delta(f)/\Delta_K$ (since β is integral). Since the size of $\Delta(f)$ is bounded by $\mathsf{poly}(\log |\Delta_K|)$ (per assumption), any denominator of q_i must so, too.

All elements and ideals in K can then be represented by vectors and bases with rational coefficients, by using this basis of $\mathcal{O}_K$ as a coordinate system; which we do in this paper.

The *size* $\operatorname{size}(k)$ of a number $k \in \mathbb{Z}$ is defined to be $\log_2 |k|$ and is extended to rationals $\frac{a}{b}$ as $\operatorname{size}(a) + \operatorname{size}(b)$ for reduced fractions $\frac{a}{b}$. The size of a $\mathbb{Q}$-vector is just the sum of the sizes of its entries, which allows to define the size of an element $\alpha \in K$ as the size of the rational vector representing α in the basis of $\mathcal{O}_K$. The size of a matrix $M_{\mathfrak{a}}$ defining an ideal $\mathfrak{a}$ is defined as the sum of the sizes of the matrix entries.

Basic operations, such as addition, multiplication, inversion and approximate computation of a complex embedding of elements in K, are all polynomial in $\log |\Delta_K|$ and the (just defined) size of the elements involved.

Elements $x = (x_\sigma)_\sigma \in K_\mathbb{R}$ are represented with rational coefficients, that is, $x \in \bigoplus_\sigma (\mathbb{Q} + i\mathbb{Q}) \subseteq K_\mathbb{R}$, and the size of such an element is the size of the vector $(x_\sigma)_\sigma$ (which can be seen as a vector in $\mathbb{Q}^n$).

Representation of ideals. We assume throughout the paper that all integral ideals $\mathfrak{a} \subseteq \mathcal{O}_K$ are represented by their Hermite Normal form (HNF) basis by default. This is possible since their matrix, using the basis of $\mathcal{O}_K$ as coordinate system, has integral coefficients.

This requirement allows us to abuse terminology and write sentences like "the algorithm takes as input an integral ideal $\mathfrak{a}$", without having to specify which basis of the ideal is given to the algorithm. As the HNF basis of an ideal is a canonical representation of that ideal, it cannot lead to any confusion.

Addition, multiplication and inversion of ideals given in HNF basis can be performed in time polynomial in the input size and in $\log |\Delta_K|$. Additionally, we have that the HNF basis of an integral ideal $\mathfrak{a}$ is bounded in size by $\mathsf{poly}(n, \log \mathcal{N}(\mathfrak{a}))$. So, by using the HNF basis of an integral ideal $\mathfrak{a}$ by default, one avoids the problem of the specific representation of $\mathfrak{a}$ bearing an influence on the running time of algorithms involving $\mathfrak{a}$. We therefore define $\operatorname{size}(\mathfrak{a})$ to be the size of its HNF basis. For a finite set $\mathbb{S}$ consisting of ideals we define $\operatorname{size}(\mathbb{S}) = |\mathbb{S}| \cdot \max_{\mathfrak{a} \in \mathbb{S}} \operatorname{size}(\mathfrak{a})$. For a modulus $\mathfrak{m} = \mathfrak{m}_0 \mathfrak{m}_\infty$, where $\mathfrak{m}_0 \subseteq \mathcal{O}_K$ and $\mathfrak{m}_\infty$ is a formal product of infinite places, we define $\operatorname{size}(\mathfrak{m}) = \operatorname{size}(\mathfrak{m}_0)$. Namely, as higher powers of infinite places do not have any influence as compared to a power of one, we will assume throughout this work that the infinite part $\mathfrak{m}_\infty$ only consists of single powers of places of K.

Compact representation of elements. In Part 2 of this article, an algorithm is discussed that computes a fundamental system of $\mathbb{S}$-units. In this algorithm, the output elements are given in a so-called *compact representation*, which we will explain presently.

Given a set of elements $\{\gamma_1, \ldots, \gamma_k\}$ of K, we can write the element $\eta = \prod_{j=1}^k \gamma_j^{n_j}$ in compact representation by the pair of vectors

$$(n_1, \ldots, n_k) \in \mathbb{Z}^k, \quad (\gamma_1, \ldots, \gamma_k) \in K^n \tag{8}$$

In this way, η is not explicitly computed in terms of the basis of K (or $\mathcal{O}_K$), but rather, the product is left implicit, allowing for much larger elements $\eta \in K$ to be described. Indeed, $\eta = \prod_{j=1}^k \gamma_j^{n_j}$ written in an $\mathcal{O}_K$-basis might require $\sum_j n_j \operatorname{size}(\gamma_j)$ bits to write down, whereas the compact representation in Equation (8) only requires at most $\sum_j \operatorname{size}(n_j) + \sum_j \operatorname{size}(\gamma_j)$ bits to write down. The drawback of this representation is that it only allows polynomial time multiplication and inversion of elements, but generally not efficient addition.

2.7. **The logarithmic embedding.** The logarithmic embedding of $K^\times$ is the map

$$\mathrm{Log} : K^\times \longrightarrow \bigoplus_\nu \mathbb{R} : \alpha \longmapsto (n_\nu \log|\sigma_\nu(\alpha)|)_\nu,$$

where $n_\nu = 2$ if ν is a complex place and 1 otherwise. This map naturally extends to all invertible elements of $K_\mathbb{R}$, and this extension is surjective, i.e., $\mathrm{Log}\, K_\mathbb{R}^\times = \bigoplus_\nu \mathbb{R}$. Denoting $K_\mathbb{R}^0 = \{(x_\sigma)_\sigma \in K_\mathbb{R} \mid \prod_\sigma x_\sigma = 1\}$, we use this logarithmic map to define the vector space $H = \mathrm{Log}(K_\mathbb{R}^0)$ (called the hyperplane) and the *logarithmic unit lattice*

$$\mathrm{Log}(\mathcal{O}_K^\times) \subseteq H,$$

a full-rank lattice in H. We write its dimension by $\mathbb{r} = \dim(H) = n_\mathbb{R} + n_\mathbb{C} - 1$. Explicitly, we have

$$H = \mathrm{Log}\, K_\mathbb{R}^0 = \{(x_\nu)_\nu \in \bigoplus_\nu \mathbb{R} \mid \sum_\nu x_\nu = 0\}.$$

The volume of the logarithmic unit lattice is given by the following formula (see, e.g., [58, I.§7 & I.§5, p. 33])

$$\mathrm{Vol}(\mathrm{Log}(\mathcal{O}_K^\times)) = \sqrt{n_\mathbb{C} + n_\mathbb{R}} \cdot R_K. \tag{9}$$

The first minimum of $\mathrm{Log}(\mathcal{O}_K^\times)$*.* By a result of Kessler [46], we have a lower bound on the first minimum of the lattice $\mathrm{Log}(\mathcal{O}_K^\times)$.

Lemma 2.14. *We have* $\lambda_1(\mathrm{Log}(\mathcal{O}_K^\times)) \geq \frac{1}{1000 \cdot \sqrt{n} \cdot \log(n)^3}$.

2.8. **Divisors.** We define the divisor group Div_K of K as

$$\mathrm{Div}_K := \bigoplus_\mathfrak{p} \mathbb{Z} \times \bigoplus_\nu \mathbb{R},$$

where ν ranges over the set of all infinite places (embeddings into the complex numbers up to possible conjugation), and $\mathfrak{p}$ ranges over all prime ideals of $\mathcal{O}_K$ (also referred to as the *finite places* of K). We denote the canonical basis elements with the symbols $(\!|\mathfrak{p}|\!)$ and $(\!|\nu|\!)$ (the divisor with value 1 at $\mathfrak{p}$ or ν respectively, and 0 everywhere else). Then, an arbitrary divisor can be written as

$$\mathbf{a} = \sum_\mathfrak{p} a_\mathfrak{p} \cdot (\!|\mathfrak{p}|\!) + \sum_\nu a_\nu \cdot (\!|\nu|\!),$$

with only finitely many non-zero $a_\mathfrak{p}$ and with $a_\nu \in \mathbb{R}$. We will consistently use the symbols $\mathbf{a}, \mathbf{b}, \mathbf{e}, \ldots$ for such divisors. Given such a divisor $\mathbf{a}$, we often write $\mathbf{a}_\mathrm{f} = \sum_\mathfrak{p} a_\mathfrak{p} \cdot (\!|\mathfrak{p}|\!)$ for its "finite part" and $\mathbf{a}_\infty = \sum_\nu a_\nu \cdot (\!|\nu|\!)$ for its "infinite part".

The degree map. The degree map is defined as

$$\begin{aligned} \deg : \mathrm{Div}_K &\longrightarrow \mathbb{R} \\ \sum_\mathfrak{p} a_\mathfrak{p} \cdot (\!|\mathfrak{p}|\!) + \sum_\nu a_\nu \cdot (\!|\nu|\!) &\longmapsto \sum_\mathfrak{p} a_\mathfrak{p} \cdot \log(\mathcal{N}(\mathfrak{p})) + \sum_\nu a_\nu. \end{aligned} \tag{10}$$

The kernel of this map is the subgroup $\mathrm{Div}_K^0 = \ker(\deg)$ of degree-zero divisors.

From field elements to divisors. Denoting $\mathrm{ord}_{\mathfrak{p}}$ for the valuation at the prime $\mathfrak{p}$, there is a canonical homomorphism

$$\begin{aligned} \llparenthesis \cdot \rrparenthesis : K^\times &\longrightarrow \mathrm{Div}_K \\ \alpha &\longmapsto \llparenthesis \alpha \rrparenthesis = \sum_{\mathfrak{p}} \mathrm{ord}_{\mathfrak{p}}(\alpha) \llparenthesis \mathfrak{p} \rrparenthesis - \underbrace{\sum_{\nu} n_\nu \log|\sigma_\nu(\alpha)| \cdot \llparenthesis \nu \rrparenthesis}_{\mathrm{Log}(\alpha)}, \end{aligned}$$

with $n_\nu = 2$ if ν is complex and 1 otherwise. The *product formula* states that for any $\alpha \in K^\times$, we have $\llparenthesis \alpha \rrparenthesis \in \mathrm{Div}_K^0$, i.e., $\llparenthesis K^\times \rrparenthesis \subset \mathrm{Div}_K^0$.

The exponential maps. The finite part $\mathbf{a}_{\mathrm{f}} \in \bigoplus_{\mathfrak{p}} \mathbb{Z}$ of a divisor $\mathbf{a}$ naturally corresponds to an ideal in the group $\mathcal{I}_K$ of fractional ideals via the map

$$\mathrm{Exp}_{\mathrm{f}} : \bigoplus_{\mathfrak{p}} \mathbb{Z} \longrightarrow \mathcal{I}_K : \sum_{\mathfrak{p}} a_{\mathfrak{p}} \cdot \llparenthesis \mathfrak{p} \rrparenthesis \longmapsto \prod_{\mathfrak{p}} \mathfrak{p}^{a_{\mathfrak{p}}}.$$

This map is an isomorphism with inverse

$$d : \mathcal{I}_K \longrightarrow \bigoplus_{\mathfrak{p}} \mathbb{Z} : \mathfrak{a} \longmapsto \sum_{\mathfrak{p}} \mathrm{ord}_{\mathfrak{p}}(\mathfrak{a}) \cdot \llparenthesis \mathfrak{p} \rrparenthesis. \tag{11}$$

We will often use a normalized section defined as

$$d^0 : \mathcal{I}_K \longrightarrow \mathrm{Div}_K^0 : \mathfrak{a} \longmapsto \sum_{\mathfrak{p}} \mathrm{ord}_{\mathfrak{p}}(\mathfrak{a}) \cdot \llparenthesis \mathfrak{p} \rrparenthesis - \frac{\log(\mathcal{N}(\mathfrak{a}))}{n} \sum_{\nu} n_\nu \llparenthesis \nu \rrparenthesis, \tag{12}$$

to map into Div_K^0 instead of Div_K.

The infinite part $\mathbf{a}_\infty \in \bigoplus_\nu \mathbb{R}$ of $\mathbf{a}$ can be mapped into $K_{\mathbb{R}}^\times$ via

$$\mathrm{Exp}_\infty : \bigoplus_{\nu} \mathbb{R} \longrightarrow K_{\mathbb{R}}^\times : \sum_{\nu} a_\nu \cdot \llparenthesis \nu \rrparenthesis \longmapsto (e^{n_{\nu_\sigma}^{-1} \cdot a_{\nu_\sigma}})_\sigma \in K_{\mathbb{R}}^\times.$$

The map Exp_∞ is injective, but not surjective. Note that the logarithmic embedding Log introduced in Section 2.7 is a retraction of Exp_∞ (i.e., $\mathrm{Log} \circ \mathrm{Exp}_\infty$ is the identity). Furthermore, for any $x = (x_\sigma)_\sigma \in K_{\mathbb{R}}^\times$, we have $\mathrm{Exp}_\infty(\mathrm{Log}(x)) = (|x_\sigma|)_\sigma \in K_{\mathbb{R}}^\times$.

The main reason for us to consider divisors is that they encode ideal lattices. This naturally follows by combining the above exponential maps into the following:

$$\mathrm{Exp} : \mathrm{Div}_K \longrightarrow \mathrm{IdLat}_K : \mathbf{a} \longmapsto \mathrm{Exp}_\infty(\mathbf{a}_\infty) \cdot \mathrm{Exp}_{\mathrm{f}}(\mathbf{a}_{\mathrm{f}}).$$

For a divisor $\mathbf{a} = \sum_{\mathfrak{p}} a_{\mathfrak{p}} \cdot \llparenthesis \mathfrak{p} \rrparenthesis + \sum_\nu a_\nu \cdot \llparenthesis \nu \rrparenthesis$, the associated ideal lattice is

$$\mathrm{Exp}(\mathbf{a}) = (e^{n_{\nu_\sigma}^{-1} \cdot a_{\nu_\sigma}})_\sigma \cdot \prod_{\mathfrak{p}} \mathfrak{p}^{a_{\mathfrak{p}}} = \left\{ (e^{n_{\nu_\sigma}^{-1} \cdot a_{\nu_\sigma}} \cdot \sigma(\alpha))_\sigma \in K_{\mathbb{R}} \mid \alpha \in \prod_{\mathfrak{p}} \mathfrak{p}^{a_{\mathfrak{p}}} \right\} \subseteq K_{\mathbb{R}}.$$

It is a group homomorphism sending the additive operation in Div_K to the multiplicative operation in IdLat_K. For any divisor $\mathbf{a} \in \mathrm{Div}_K$, we have

$$\mathrm{Vol}(\mathrm{Exp}(\mathbf{a})) = \sqrt{|\Delta_K|} \cdot \prod_{\nu} e^{a_\nu} \cdot \mathcal{N}\left(\prod_{\mathfrak{p}} \mathfrak{p}^{a_{\mathfrak{p}}}\right) = \sqrt{|\Delta_K|} \cdot e^{\deg(\mathbf{a})}.$$

Euclidean norm on Div_K*.* For $\mathbf{a} \in \mathrm{Div}_K$ written as

$$\mathbf{a} = \sum_{\mathfrak{p}} a_{\mathfrak{p}} \cdot (\!|\mathfrak{p}|\!) + \sum_{\nu} a_{\nu} \cdot (\!|\nu|\!),$$

we define the norm

$$\|\mathbf{a}\| = \left(\sum_{\mathfrak{p}} a_{\mathfrak{p}}^2 + \sum_{\nu} a_{\nu}^2 \right)^{1/2},$$

Note that, naturally, $H \subseteq \mathrm{Log}(K_{\mathbb{R}}^{\times}) \hookrightarrow \mathrm{Div}_K$, where the defined Euclidean norms are compatible.

Subgroups of Div_K*.* Given a set $\mathbb{S}$ of prime ideals of $\mathcal{O}_K$, we define the subgroup

$$\mathrm{Div}_{K,\mathbb{S}} = \bigoplus_{\mathfrak{p} \in \mathbb{S}} \mathbb{Z} \times \bigoplus_{\nu} \mathbb{R} \subseteq \mathrm{Div}_K .$$

In the next two parts of this article, we will work with two important cases. In Part 1, we will consider a modulus $\mathfrak{m} = \mathfrak{m}_0 \mathfrak{m}_\infty$, and let $\mathbb{S}$ be the set of all prime ideals that do not divide $\mathfrak{m}_0$. The resulting group $\mathrm{Div}_{K^{\mathfrak{m}}} := \bigoplus_{\mathfrak{p} \nmid \mathfrak{m}_0} \mathbb{Z} \times \bigoplus_{\nu} \mathbb{R}$ is called the *Arakelov* $\mathfrak{m}$*-ray divisor group.* In Part 2, we will consider cases where the set $\mathbb{S}$ is finite. We then call $\mathrm{Div}_{K,\mathbb{S}}$ the $\mathbb{S}$-divisor group, and it relates to the classical notion of $\mathbb{S}$-units. Further details on each specific case are discussed in the relevant sections.

2.9. **Probabilities and (Discrete) Gaussian distributions.** Given a distribution D over a discrete set X and $x \in X$, we denote $D(x)$ for the probability that D outputs x and $D(A) = \sum_{x \in A} D(x)$ for any $A \subseteq X$.

Definition 2.15 (Statistical distance)**.** *Let* $(\Omega, \mathcal{S})$ *be a measurable space with probability measures* P, Q*. The statistical distance (or total variation distance) between* P *and* Q *is defined by the rule*

$$SD(P, Q) = \sup_{X \in \mathcal{S}} |P(X) - Q(X)|.$$

For a discrete space Ω, we have

$$SD(P, Q) = \frac{1}{2} \sum_{x \in \Omega} |P(x) - Q(x)| =: \frac{1}{2} \|P - Q\|_1.$$

For a continuous space Ω with probability densities P, Q, we have

$$SD(P, Q) = \frac{1}{2} \int_{x \in \Omega} |P(x) - Q(x)| =: \frac{1}{2} \|P - Q\|_1.$$

For spaces that are partially discrete and partially continuous, some well-defined mix of an integral and a sum will define the statistical distance. Due to the equivalence of these notions (up to a constant $\frac{1}{2}$) we will often describe closeness of probability distributions in terms of the distance metric $\| \cdot \|_1$, instead of $SD(\cdot, \cdot)$.

The data processing inequality captures the idea that an algorithm (by just processing a single query) cannot increase the statistical distance between two probability distributions. A proof can be found for example in [28, §2.8].

Theorem 2.16 (Data processing inequality)**.** *Let $(\Omega, \mathcal{S})$ be a measurable space with probability measures P, Q. Let f be a (potentially probabilistic) function on Ω. Then*

$$\|f(P) - f(Q)\|_1 \leq \|P - Q\|_1.$$

In the rest of this section, we recall known results about Gaussian distributions and prove some lemmas that will be useful for the rest of the article.

Gaussian distributions. We denote $g_s(x) = \exp(-\pi\|x\|^2/s^2)$ the Gaussian function defined on any Euclidean vector space V. Since $\int_{x \in V} g_s(x)dx = s^{\dim V}$, we define the continuous Gaussian distribution on a Euclidean vector space V with parameter s and center c by the density function

$$\mathcal{G}_{V,s,c}(v) = s^{-\dim(V)} \cdot g_s(v - c).$$

For discrete sets $X \subseteq V$, specifically lattices, we use the same notation for the *discrete Gaussian*, where the subscript X indicates that the Gaussian is discrete.

$$\mathcal{G}_{X,s,c}(x) = g_s(x - c)/g_s(X - c),$$

where $g_s(X - c) = \sum_{x \in X} g_s(x - c)$. This notation $f(X) = \sum_{x \in X} f(x)$ for functions f and discrete sets X will be used frequently. Whenever the Gaussian is centered at the origin, i.e., $c = 0$, we will suppress the subscript c from the notation, like this: $\mathcal{G}_{V,s}$.

Tail bounds of Gaussian distributions. The following lemma originates from Banaszczyk's paper on transference theorems in lattices [6, Lemma 1.5].

Lemma 2.17 ([6, Lemma 1.5])**.** *For any $c > 1/\sqrt{2\pi}$ and any n dimensional lattice Λ, $g_1(\Lambda \setminus c\sqrt{n}\mathcal{B}) \leq C^n g_1(\Lambda)$, where $\mathcal{B}$ is the euclidean ball of radius 1 and $C = c\sqrt{2\pi \mathrm{e}} \cdot \mathrm{e}^{-\pi c^2} < 1$.*

Corollary 2.18. *Let Λ be a lattice of rank n and $s > 0$. For any $\varepsilon \in (0, 1]$, it holds that $\Pr_{x \leftarrow \mathcal{G}_{\Lambda,s}}(\|x\| \geq s \cdot \sqrt{\log(1/\varepsilon) + 2n}) \leq \varepsilon$.*

We also have a similar result in the case of a continuous Gaussian distribution.

Lemma 2.19. *Let V be a real vector space of dimension n and $s > 0$. For any $\varepsilon \in (0, 1]$, it holds that $\Pr_{x \leftarrow \mathcal{G}_{V,s}}(\|x\| \geq s \cdot \sqrt{2n \cdot \log(2n/\varepsilon)}) \leq \varepsilon$.*

Proof. Let B be an orthonormal basis of V and write $x = (x_1, \cdots, x_n)$ the coordinates of x in this basis. Then the random variables x_i are linearly independent Gaussian distributions over $\mathbb{R}$ with standard deviation s. Moreover, for any $t > 0$, if $\|x\| \geq t$, there should exist some i such that $|x_i| \geq t/\sqrt{n}$. Hence, we obtain

$$\Pr_{x \leftarrow \mathcal{G}_{V,s}}\big(\|x\| \geq t\big) \leq n \cdot \Pr_{x \leftarrow \mathcal{G}_{\mathbb{R},s}}\big(|x| \geq t/\sqrt{n}\big) \leq 2n \cdot \exp\Big(-\frac{t^2}{2n \cdot s^2}\Big),$$

where the first inequality comes from the union bound and the last one comes from Chernoff's bound. Taking $t = s \cdot \sqrt{2n \cdot \log(2n/\varepsilon)}$ leads to the desired result. □

Smoothing on lattices. For any $\varepsilon > 0$ and a lattice Λ we define the smoothing parameter [55] $\eta_\varepsilon(\Lambda)$ to be the smallest $s \in \mathbb{R}_{>0}$ such that $g_{1/s}(\Lambda^\vee \backslash \{0\}) \leq \varepsilon$, where $\Lambda^\vee = \{x \in \mathrm{span}_\mathbb{R}(\Lambda) \mid \langle x, \ell \rangle \in \mathbb{Z} \text{ for all } \ell \in \Lambda\}$ is the dual lattice of Λ. This smoothing parameter satisfies the following bound [55, Lemma 3.3]

$$\eta_\varepsilon(\Lambda) \leq \sqrt{\frac{\log(2n(1+1/\varepsilon))}{\pi}} \cdot \lambda_n(\Lambda). \tag{13}$$

We will use the following classical results on the total Gaussian weight of a (shifted) lattice Λ.

Lemma 2.20 (Proof of [55, Lemma 4.4])**.** *Let Λ be an n-dimensional lattice and $s \geq \eta_\varepsilon(\Lambda)$ for some $\varepsilon > 0$. We have*

$$g_s(\Lambda) \in [1-\varepsilon, 1+\varepsilon] \cdot \frac{s^n}{\mathrm{Vol}(\Lambda)}.$$

Lemma 2.21 ([55, Lemma 2.9])**.** *Let Λ be an n-dimensional lattice, $s > 0$ be a positive real number and $c \in \mathrm{span}_\mathbb{R}(\Lambda)$ be a vector, then $g_s(\Lambda + c) \leq g_s(\Lambda)$.*

Approximating and computing discrete Gaussians. We will use the following lemma which says that one can sample from a distribution statistically close to a discrete Gaussian distribution. It is an adaptation of [36, Theorem 4.1] for which the Gaussian can be approximated within any error and in which only bit operations are used (as opposed to real number operations as in the original article).

Lemma 2.22 (Adapted from [36, Theorem 4.1])**.** *There exists a probabilistic algorithm that takes as input a basis $B = (\mathbf{b}_1, \cdots, \mathbf{b}_n) \in \mathbb{Q}^{n\times n}$ of an n-dimensional lattice Λ, an error bound $\varepsilon_\mathcal{G} \in (0,1]$, a parameter $s \geq \sqrt{\frac{\log(1/\varepsilon_\mathcal{G})+2\log(n)+3}{\pi}} \cdot \max_i \|\mathbf{b}_i\|$ and a center $\mathbf{c} \in \mathrm{span}_\mathbb{R}(\Lambda) \cap \mathbb{Q}^n$ and outputs a sample from a distribution $\widehat{\mathcal{G}}_{\mathbf{B},\varepsilon_\mathcal{G},s,\mathbf{c}}$ such that $SD(\widehat{\mathcal{G}}_{\mathbf{B},\varepsilon_\mathcal{G},s,\mathbf{c}}, \mathcal{G}_{\Lambda,s,\mathbf{c}}) \leq \varepsilon_\mathcal{G}$. This algorithm runs in expected time (bit complexity) polynomial in the size of the input and in $\log(1/\varepsilon_\mathcal{G})$. Additionally, the output of $\mathbf{v} \leftarrow \widehat{\mathcal{G}}_{\mathbf{B},\varepsilon_\mathcal{G},s,\mathbf{c}}$ always satisfies $\|\mathbf{v} - \mathbf{c}\| \leq s \cdot \sqrt{n \log(2n^2/\varepsilon_\mathcal{G})}$.*

Proof. The proof can be adapted from the one of [36, Theorem 4.1], which proves the result for $\varepsilon_\mathcal{G} = n^{-\omega(1)}$. In this proof, the statistical distance between the sampled distribution $\widehat{\mathcal{G}}_{\mathbf{B},\varepsilon_\mathcal{G},s,c}$ and the ideal distribution $\mathcal{G}_{\Lambda,s,\mathbf{c}}$ comes from two places. The first one is that the algorithm makes n calls to a sub-routine algorithm $\texttt{Sample}_\mathbb{Z}$, sampling from a Gaussian distribution over $\mathbb{Z}$ which is only statistically close to $\mathcal{G}_{\mathbb{Z},s',c'}$ (for some parameters c' and s' depending on the input). Lemmas 4.2 and 4.3 from [36] show that the statistical distance between $\texttt{Sample}_\mathbb{Z}$ and $\mathcal{G}_{\mathbb{Z},s',c'}$ can be made as small as any arbitrary $\delta > 0$, to the cost of increasing the running time of $\texttt{Sample}_\mathbb{Z}$ in a way that is polynomial in $\log(1/\delta)$. We choose $\delta = \varepsilon_\mathcal{G}/(2n)$, which provides a running time polynomial in $\log(1/\varepsilon_\mathcal{G})$ and $\log n$, and ensures that the n calls to $\texttt{Sample}_\mathbb{Z}$ are responsible for a change of statistical distance that is at most $\varepsilon_\mathcal{G}/2$.

The second reason why the algorithm only provides a distribution that is statistically close to $\mathcal{G}_{\Lambda,s,c}$ comes from the fact that $g_{s'}(\mathbb{Z}+c)$ is only approximately close to $g_{s'}(\mathbb{Z})$ for an arbitrary $c \in \mathbb{R}$ (where $s' \geq s/(\max_i \|b_i\|)$). In the proof of Theorem 4.1, the authors of [36] show that, provided that $s \geq \eta_\delta(\mathbb{Z}) \cdot \max_i \|\mathbf{b}_i\|$, then this statistical distance is at most $\delta'/2$, where $1 + \delta' = \left(\frac{1+\delta}{1-\delta}\right)^n$. We choose

$\delta = \varepsilon_{\mathcal{G}}/(8n) \leq 1/(8n)$. Using the fact that[4] $\left(\frac{1+\delta}{1-\delta}\right)^n \leq 1 + 8n\delta$ for all $\delta \leq 1/(8n)$ and $n \geq 1$, we obtain that the statistical distance between the approximate distribution and the ideal one is at most $4n\delta = \varepsilon_{\mathcal{G}}/2$. Using Equation (13), we see that $\eta_\delta(\mathbb{Z}) \leq \sqrt{\frac{\log(1/\varepsilon_{\mathcal{G}})+2\log(n)+3}{\pi}}$, hence $s \geq \eta_\delta(\mathbb{Z}) \cdot \max_i \|\mathbf{b}_i\|$ as desired.

For the bound on $v \leftarrow \widehat{\mathcal{G}}_{\mathbf{B},\varepsilon_{\mathcal{G}},s,\mathbf{c}}$ observe the output $\mathbf{v}_0$ of the algorithm in [36, Section 4.2]. By [36, Lemma 4.4], the output of this algorithm satisfies $\mathbf{v} - \mathbf{c} = \sum_{i\in[n]}(\hat{z}_i - c'_i) \cdot \tilde{\mathbf{b}}_i$, where the values $\hat{z}_i$ and c'_i are from the algorithm, and the $\tilde{\mathbf{b}}_i$ are the Gram-Schmidt vectors associated to the $\mathbf{b}_i$'s. Hence, by the Pythagorean theorem, $\|\mathbf{v}-\mathbf{c}\|^2 = \sum_{i\in[n]} |\hat{z}_i - c'_i|^2 \cdot \|\tilde{\mathbf{b}}_i\|^2$. In part 1(b) of the algorithm [36, Section 4.2], $D_{\mathbb{Z},s'_i,c'_i}$ is implemented as in [36, Section 4.1], i.e., the algorithm $\texttt{Sample}\mathbb{Z}$ is called (see the proof of [36, Theorem 4.1]). Therefore $\hat{z}_i$ can be shown to lie in $\mathbb{Z} \cap [c'_i - t(n)s'_i, c'_i + t(n)s'_i]$, where $t(n) = \sqrt{\log(n/\delta)}$ (with $\delta = \varepsilon_{\mathcal{G}}/(2n)$, as before). As a consequence,

$$\|\mathbf{v}-\mathbf{c}\|^2 = \sum_{i\in[n]} |\hat{z}_i - c'_i|^2 \cdot \|\tilde{\mathbf{b}}_i\|^2 \leq \sum_{i\in[n]} t(n)^2 (s'_i)^2 \cdot \|\tilde{\mathbf{b}}_i\|^2 = \sum_{i\in[n]} t(n)^2 s^2 = nt(n)^2 s^2,$$

by the definitions of $s'_i = s/\|\tilde{\mathbf{b}}_i\|$. Hence $\|\mathbf{v} - \mathbf{c}\| \leq \sqrt{n\log(n/\delta)}s$ with $\delta = \varepsilon_{\mathcal{G}}/(2n)$, as was required to prove.

We show now that the algorithm in [36] can be readily adapted into one without real number operations, but just bit operations. There are two places in the algorithm in [36, Section 3.2] where real arithmetic is used, namely in the subroutine $\texttt{Sample}\mathbb{Z}$ and in the computation of $s/\|\tilde{\mathbf{b}}_k\|$ where $\tilde{\mathbf{b}}_k$ are the Gram-Schmidt vectors. Note that the Gram-Schmidt orthogonalization itself (without normalizing) can be done with rational arithmetic in polynomial time and is thus not altered in this adapted version.

The subroutine $\texttt{Sample}\mathbb{Z}$ can be amended to avoid real operations by just approximating the sampling probabilities $\rho_s(x - c)$ with $x \in \mathbb{Z} \cap [c - s\cdot t(n), c + s\cdot t(n)]$ well enough. This can be done within polynomial bit complexity in the size of the input, and, by maybe slightly increasing $t(n)$, without loss in the approximation error δ.

The value of $s/\|\tilde{\mathbf{b}}_k\|$ is only used in the subroutine $\texttt{Sample}\mathbb{Z}$, where actually its *square* is used to compute $\rho_{s/\|\tilde{\mathbf{b}}_k\|}(x - c)$ and hence there is no need to compute $s/\|\tilde{\mathbf{b}}_k\|$ but rather its square $s^2/\|\tilde{\mathbf{b}}_k\|^2$ which consists of rational numbers.

Hence we can conclude that this slight adaptation of [36, Section 3] has polynomial bit-complexity in the size of its input. □

3. Acknowledgements

First and foremost, we would like to express our gratitude to Léo Ducas, who took part in early discussions that lead to this project, and suggested key ideas for the randomisation of ideals.

We thank Aurel Page and Wessel van Woerden for insightful discussions. We thank Sameera Vemulapalli for bringing to our attention the bound on $\lambda_n(\mathcal{O}_K)$ of Bhargava et al. [8].

[4] We have $\frac{1+\delta}{1-\delta} \leq 1 + 4\delta$ for $\delta \leq 1/2$. Also, $(1 + x)^n \leq (e^x)^n = e^{nx} \leq 1 + 2nx$ for $x < 1/(2n)$. Hence, for $\delta \leq 1/(8n)$ and $n \in \mathbb{N}_{>0}$, $\left(\frac{1+\delta}{1-\delta}\right)^n \leq (1 + 4\delta)^n \leq 1 + 8n\delta$.

Alice Pellet-Mary and Benjamin Wesolowski were supported by the Agence Nationale de la Recherche under grants ANR-22-PETQ-0008 (PQ-TLS) and ANR-21- CE94-0003 (CHARM). Benjamin Wesoloswki was supported by the European Research Council under grant No. 101116169 (AGATHA CRYPTY). This work started when the authors were visiting the Simon Institute during the lattice semester in 2020.

PART 1

Sampling ideals in a class: smooth, near-prime or otherwise

4. Introduction

In this first part of the article, we propose a general strategy to provably solve a recurring computational problem in number theory (assuming the extended Riemann hypothesis, ERH): given an ideal class $[\mathfrak{a}]$ of a number field K, sample an ideal $\mathfrak{c} \in [\mathfrak{a}]$ belonging to a particular family of ideals (e.g., the family of smooth ideals, or near-prime ideals). While there is a simple heuristic algorithm for this task, it has proved notoriously difficult to resolve it rigorously. It has thereby been a central roadblock explaining the heuristic nature of many major algorithms in computational number theory.

The main result of this part is Theorem 9.5 (page 47). Its formulation in full generality is postponed, as it first requires the introduction of several notions. However, a simplified statement, which may already suit many needs, is available in Theorem 1.1 (page 3).

Roadmap. Fix an arbitrary family of ideals $\mathcal{S}$. For convenience, we consider the input to be an ideal $\mathfrak{b} \in [\mathfrak{a}]^{-1}$, and are looking for an ideal $\mathfrak{c} \in \mathcal{S}$ in the inverse class of $\mathfrak{b}$. The folklore strategy consists in considering $\mathfrak{b}$ as an *ideal lattice* via the Minkowski embedding, and sampling random elements $\beta \in \mathfrak{b}$ (within some bounds, say in a "box" $r\mathcal{B}$ of radius r) until $\mathfrak{c} = \beta\mathfrak{b}^{-1}$ falls in the desired family of ideals $\mathcal{S}$. Heuristically, one *expects* that for β "sufficiently random", the ideal $\beta\mathfrak{b}^{-1}$ falls in $\mathcal{S}$ with probability proportional to the "density" of the family (think about the set of prime ideals that have norm around x having "density" $\approx 1/(\rho_K \log(x))$). This, of course, cannot be literally true for arbitrary families (e.g., principal ideals), since $\beta\mathfrak{b}^{-1}$ is confined to one ideal class. Instead, we will solve the problem for $\mathfrak{c} \in \mathcal{S} \cdot \mathcal{S}_B$, where $\mathcal{S}_B$ is the family of B-smooth ideals for some bound B. In all applications we are aware of, $\mathcal{S} = \mathcal{S} \cdot \mathcal{S}_B$ (smooth ideals, near-prime ideals).

The Arakelov class group. The notion of *ideal lattice* plays a key role in this sampling strategy. The space of ideal lattices up to isomorphism is naturally isomorphic to the so-called *Arakelov class groups* We open this part of the article in Section 5 with an introduction to Arakelov ray class groups. They can be thought of as a "combination" of the ray class group and the ray unit group of a number field. In Section 6, we state a useful result on these Arakelov ray class groups: certain *random walks* in them rapidly converge to the uniform distribution. This is a generalization of [14] from Arakelov class groups to Arakelov *ray* class groups.

Average densities. In the folklore strategy, one is hoping that for (uniformly) random $\beta \in \mathfrak{b} \cap r\mathcal{B}$ (the intersection of an ideal lattice and a "box"), the probability that $\beta\mathfrak{b}^{-1} \in \mathcal{S}$ is proportional to the density of $\mathcal{S}$. This is generally not true. However, in Section 7, we prove that it *is* true *on average* when the ideal lattice $\mathfrak{b}$ is also random, uniformly distributed in the Arakelov ray class group. This might sound too weak for our goal: the input $\mathfrak{b}$ of the algorithm is not random. It will actually be sufficient when properly combined with a randomization step.

Ideal sampling. We then turn these density results into an algorithm. First, we show in Section 8 how to sample uniformly a random element in a set of the form $\mathfrak{b} \cap r\mathcal{B}$, the intersection of an ideal lattice and a "box".

Then, Section 9 culminates with the main algorithm, Algorithm 2. It addresses the final obstacle: the density result of Section 7 only holds on average, for uniformly random ideal lattices. The input $\mathfrak{b}$ is not random, so we need to randomize it. This is where random walks in the Arakelov class group come in. Essentially, Algorithm 2 starts by multiplying $\mathfrak{b}$ with a few "small" prime ideals, resulting in a random ideal $\mathfrak{w}\mathfrak{b}$ where $\mathfrak{w}$ is smooth (i.e., $\mathfrak{w} \in \mathcal{S}_B$ for some bound B).

Algorithm 2 then samples a uniformly random $\beta \in (\mathfrak{w}\mathfrak{b}) \cap r\mathcal{B}$. Then:

- From Section 6 and [14] (on the rapid equidistribution of random walks), the random ideal lattice $\mathfrak{w}\mathfrak{b}$ is close to uniformly distributed.
- From Section 7, we obtain that $\beta(\mathfrak{w}\mathfrak{b})^{-1} \in \mathcal{S}$ with probability proportional to the density of $\mathcal{S}$.

Upon the event $\beta(\mathfrak{w}\mathfrak{b})^{-1} \in \mathcal{S}$, we get $\beta\mathfrak{b}^{-1} = \mathfrak{w} \cdot (\beta(\mathfrak{w}\mathfrak{b})^{-1}) \in \mathcal{S}_B \cdot \mathcal{S}$, as desired.

Further properties. Finally, in Section 10 and Section 11, we develop tools to ease the applicability of this "ideal sampling" algorithm. These tools are indispensable for the application presented in Part 2, and may be useful in other contexts. More precisely, in Section 10, three important properties of the ideal sampling algorithm are stated and proved: the *shifting property*, *boundedness* and *almost-Lipschitz-continuity*. In Section 11, we estimate quantities related to the modulus $\mathfrak{m}$, which affect the behavior of the ideal sampling algorithm.

5. Background on the Arakelov class group

Recall that throughout this paper, we consider a number field K of rank n over $\mathbb{Q}$, having ring of integers $\mathcal{O}_K$, discriminant Δ_K, regulator R_K, class number h_K and group of roots of unity μ_K. It has $n_\mathbb{R}$ real embeddings and $n_\mathbb{C}$ conjugate pairs of complex embeddings, with $n = n_\mathbb{R} + 2n_\mathbb{C}$.

5.1. **The Arakelov Ray Class Group.**

In this section, we rely heavily on the notation introduced in Section 2.8 for the divisor group Div_K. The *Arakelov ray divisor group* with respect to a modulus $\mathfrak{m} = \mathfrak{m}_0\mathfrak{m}_\infty$ is a subgroup $\mathrm{Div}_{K^\mathfrak{m}} \subseteq \mathrm{Div}_K$ defined as

$$\mathrm{Div}_{K^\mathfrak{m}} = \bigoplus_{\mathfrak{p}\nmid\mathfrak{m}_0} \mathbb{Z} \times \bigoplus_{\nu} \mathbb{R}$$

where $\mathfrak{p}$ ranges over the set of all prime ideals of $\mathcal{O}_K$ that do not divide the finite part $\mathfrak{m}_0$ of the modulus, and ν over the set of all infinite places (embeddings into the complex numbers up to conjugation). The case $\mathfrak{m} = \mathcal{O}_K$ yields the standard divisor group Div_K. Recall that we write an arbitrary element in $\mathrm{Div}_{K^{\mathfrak{m}}}$ as

$$\mathbf{a} = \underbrace{\sum_{\mathfrak{p} \nmid \mathfrak{m}_0} a_{\mathfrak{p}} \cdot \llparenthesis \mathfrak{p} \rrparenthesis}_{\mathbf{a}_{\mathrm{f}}} + \underbrace{\sum_{\nu} a_{\nu} \cdot \llparenthesis \nu \rrparenthesis}_{\mathbf{a}_{\infty}} . \tag{14}$$

with only finitely many non-zero $a_{\mathfrak{p}}$ and with $a_\nu \in \mathbb{R}$. The map $\llparenthesis \cdot \rrparenthesis : K^\times \to \mathrm{Div}_K$ naturally restricts and co-restrict to

$$\begin{aligned} \llparenthesis \cdot \rrparenthesis : K^{\mathfrak{m}} &\longrightarrow \mathrm{Div}_{K^{\mathfrak{m}}} \\ \alpha &\longmapsto \llparenthesis \alpha \rrparenthesis = \sum_{\mathfrak{p} \nmid \mathfrak{m}_0} \mathrm{ord}_{\mathfrak{p}}(\alpha) \llparenthesis \mathfrak{p} \rrparenthesis - \sum_{\nu} n_\nu \log |\sigma_\nu(\alpha)| \cdot \llparenthesis \nu \rrparenthesis, \end{aligned}$$

where $n_\nu = 2$ whenever ν is a complex place and 1 otherwise. The divisors of the form $\llparenthesis \alpha \rrparenthesis$ for $\alpha \in K^{\mathfrak{m},1}$ are called *principal* $\mathfrak{m}$*-ray divisors.*

Just as the ideal ray class group is the group of ideals coprime with $\mathfrak{m}$ quotiented by the ray $K^{\mathfrak{m},1}$, the *Picard ray group* is the group of Arakelov ray divisors quotiented by the group of principal ray Arakelov divisors. In other words, the Picard ray group $\mathrm{Pic}_{K^{\mathfrak{m}}}$ is defined by the following exact sequence, where $\mu_{K^{\mathfrak{m},1}} = \mu_K \cap K^{\mathfrak{m},1}$ are the roots of unity in the ray:

$$0 \to \mu_{K^{\mathfrak{m},1}} \to K^{\mathfrak{m},1} \xrightarrow{\llparenthesis \cdot \rrparenthesis} \mathrm{Div}_{K^{\mathfrak{m}}} \to \mathrm{Pic}_{K^{\mathfrak{m}}} \to 0.$$

For any Arakelov ray divisor $\mathbf{a} = \sum_{\mathfrak{p} \nmid \mathfrak{m}_0} a_{\mathfrak{p}} \cdot \llparenthesis \mathfrak{p} \rrparenthesis + \sum_\nu a_\nu \cdot \llparenthesis \nu \rrparenthesis$, we denote its class in the Picard ray group $\mathrm{Pic}_{K^{\mathfrak{m}}}$ by $[\mathbf{a}]$, in the same fashion that $[\mathfrak{a}]$ denotes the ideal class of the ideal $\mathfrak{a}$ in $\mathrm{Cl}_K^{\mathfrak{m}}$.

Since principal ray divisors $\llparenthesis \alpha \rrparenthesis$ for $\alpha \in K^{\mathfrak{m},1}$ are in the kernel of the degree map, the degree factors through $\mathrm{Pic}_{K^{\mathfrak{m}}}$. We can therefore define the *degree-zero Arakelov ray divisor group* $\mathrm{Div}^0_{K^{\mathfrak{m}}} = \{\mathbf{a} \in \mathrm{Div}_{K^{\mathfrak{m}}} \mid \deg(\mathbf{a}) = 0\}$ and the *Arakelov ray class group* $\mathrm{Pic}^0_{K^{\mathfrak{m}}} = \{[\mathbf{a}] \in \mathrm{Pic}_{K^{\mathfrak{m}}} \mid \deg([\mathbf{a}]) = 0\}$.
The maps $\mathrm{Exp}_{\mathrm{f}}$, Exp_∞, d and d^0 defined in Section 2.8 naturally restrict and co-restrict to

$$\mathrm{Exp}_{\mathrm{f}} : \bigoplus_{\mathfrak{p} \nmid \mathfrak{m}_0} \mathbb{Z} \longrightarrow \mathcal{I}_K^{\mathfrak{m}} \quad : \mathbf{a} \longmapsto \prod_{\mathfrak{p} \nmid \mathfrak{m}_0} \mathfrak{p}^{a_{\mathfrak{p}}}, \tag{15}$$

$$\mathrm{Exp}_\infty : \bigoplus_{\nu} \mathbb{R} \longrightarrow K_{\mathbb{R}}^\times \quad : \mathbf{a} \longmapsto \left(e^{n_{\nu_\sigma}^{-1} \cdot a_{\nu_\sigma}} \right)_\sigma , \tag{16}$$

$$d : \mathcal{I}_K^{\mathfrak{m}} \longrightarrow \bigoplus_{\mathfrak{p} \nmid \mathfrak{m}_0} \mathbb{Z} \; : \mathfrak{a} \longmapsto \sum_{\mathfrak{p} \nmid \mathfrak{m}_0} \mathrm{ord}_{\mathfrak{p}}(\mathfrak{a}) \cdot \llparenthesis \mathfrak{p} \rrparenthesis, \tag{17}$$

$$d^0 : \mathcal{I}_K^{\mathfrak{m}} \longrightarrow \mathrm{Div}^0_{K^{\mathfrak{m}}} : \mathfrak{a} \longmapsto d(\mathfrak{a}) - \frac{n_\nu \log(\mathcal{N}(\mathfrak{a}))}{n} \sum_{\nu} \llparenthesis \nu \rrparenthesis. \tag{18}$$

The groups and their relations treated above fit nicely in the diagram of exact sequences given in Figure 1, where the middle row sequence splits with the section d^0. In this diagram we use the notation $\mathcal{O}^\times_{K^{\mathfrak{m},1}} = \mathcal{O}_K^\times \cap K^{\mathfrak{m},1}$ and $\mu_{K^{\mathfrak{m},1}} = \mu_K \cap K^{\mathfrak{m},1}$. The group $T^{\mathfrak{m}} = H / \mathrm{Log}(\mathcal{O}^\times_{K^{\mathfrak{m},1}})$ is the *logarithmic ray unit torus*, with $\mathrm{Log}(\mathcal{O}^\times_{K^{\mathfrak{m},1}})$ the *logarithmic ray unit lattice.*

Relations between different ray groups. The (ray) unit groups $\mathcal{O}_K^\times, \mathcal{O}^\times_{K^{\mathfrak{m},1}}$, the (ray) class groups $\mathrm{Cl}_K, \mathrm{Cl}_K^{\mathfrak{m}}$, and the ray groups $K^{\mathfrak{m},1}$ and $K^{\mathfrak{m}}$ are tightly related by an

$$\begin{tikzcd}
 & 0 \arrow[d] & 0 \arrow[d] & 0 \arrow[d] & \\
0 \arrow[r] & \mathcal{O}^\times_{K^{\mathfrak{m},1}}/\mu_{K^{\mathfrak{m},1}} \arrow[r] \arrow[d, "\mathrm{Log}"'] & K^{\mathfrak{m},1}/\mu_{K^{\mathfrak{m},1}} \arrow[r] \arrow[d, "(\!|\cdot|\!)"'] & \mathrm{Princ}^{\mathfrak{m}}_K \arrow[r] \arrow[d] & 0 \\
0 \arrow[r] & H \arrow[r] \arrow[d] & \mathrm{Div}^0_{K^{\mathfrak{m}}} \arrow[r, "\mathbf{a} \mapsto \mathrm{Exp}_{\mathrm{f}}(\mathbf{a}_{\mathrm{f}})"'] \arrow[d] & \mathcal{I}^{\mathfrak{m}}_K \arrow[r] \arrow[d] \arrow[l, bend right, "d^0"'] & 0 \\
0 \arrow[r] & T^{\mathfrak{m}} \arrow[r] \arrow[d] & \mathrm{Pic}^0_{K^{\mathfrak{m}}} \arrow[r] \arrow[d] & \mathrm{Cl}^{\mathfrak{m}}_K \arrow[r] \arrow[d] & 0 \\
 & 0 & 0 & 0 &
\end{tikzcd}$$

FIGURE 1. A commutative diagram of short exact sequences involving the Arakelov ray class group.

exact sequence (e.g., [49, Chapter VI, §1]), relating the (relative) cardinalities of these groups. Namely,

$$|\mathcal{O}_K^\times/\mathcal{O}^\times_{K^{\mathfrak{m},1}}| \cdot |\,\mathrm{Cl}^{\mathfrak{m}}_K\,| = \phi(\mathfrak{m}_0) \cdot 2^{|\mathfrak{m}_{\mathbb{R}}|} \cdot |\,\mathrm{Cl}_K\,|, \tag{19}$$

where $\phi(\mathfrak{m}_0) = |K^{\mathfrak{m}_0}/K^{\mathfrak{m}_0,1}|$ and $|\mathfrak{m}_{\mathbb{R}}|$ is the number of real places dividing $\mathfrak{m}_\infty$, (note that $|K^{\mathfrak{m}}/K^{\mathfrak{m},1}| = \phi(\mathfrak{m}_0) \cdot 2^{|\mathfrak{m}_{\mathbb{R}}|}$). Also, by observing the kernel-cokernel sequence of the inclusions $\mathcal{O}^\times_{K^{\mathfrak{m},1}} \subseteq \mathcal{O}_K^\times \subseteq H = \log K^0_{\mathbb{R}}$, we obtain,

$$|\mu_{K^{\mathfrak{m},1}}| \cdot |\mathcal{O}_K^\times/\mathcal{O}^\times_{K^{\mathfrak{m},1}}| = |\mu_K| \cdot \mathrm{Vol}(T^{\mathfrak{m}})/\,\mathrm{Vol}(T). \tag{20}$$

Geometric properties of the divisor group and the Arakelov class group. Recall that the norm of a divisor is defined as

$$\|\mathbf{a}\| = \left(\sum_{\mathfrak{p}} a_{\mathfrak{p}}^2 + \sum_{\nu} a_\nu^2\right)^{1/2}.$$

In the following lemma, we show that the volume of the Arakelov ray class group roughly follows the square root of the absolute value of the field discriminant times $\phi(\mathfrak{m}_0) \cdot 2^{|\mathfrak{m}_{\mathbb{R}}|}$; with a possible additional correction factor due to the roots of unity.

Lemma 5.1 (Volume of $\mathrm{Pic}^0_{K^{\mathfrak{m}}}$)**.** *For $n = [K : \mathbb{Q}] > 1$, we have*

$$\mathrm{Vol}\left(\mathrm{Pic}^0_{K^{\mathfrak{m}}}\right) = |\,\mathrm{Cl}^{\mathfrak{m}}_K\,| \cdot \mathrm{Vol}(T^{\mathfrak{m}}) = \frac{|\mu_{K^{\mathfrak{m},1}}|}{|\mu_K|} \cdot \phi(\mathfrak{m}_0) \cdot 2^{|\mathfrak{m}_{\mathbb{R}}|} \cdot h_K \cdot \mathrm{Vol}(T) \tag{21}$$

$$= \frac{|\mu_{K^{\mathfrak{m},1}}|}{|\mu_K|} \cdot \phi(\mathfrak{m}_0) \cdot h_K \cdot R_K \cdot \sqrt{\mathbb{r}+1} \cdot 2^{|\mathfrak{m}_{\mathbb{R}}|}, \tag{22}$$

where $\phi(\mathfrak{m}_0) = |K^{\mathfrak{m}_0}/K^{\mathfrak{m}_0,1}|$ and $|\mathfrak{m}_{\mathbb{R}}|$ is the number of real places dividing $\mathfrak{m}_\infty$. Additionally,

$$\log\left(\mathrm{Vol}\left(\mathrm{Pic}^0_{K^{\mathfrak{m}}}\right)\right) \leq \log\left(\mathcal{N}(\mathfrak{m}_0) \cdot 2^{|\mathfrak{m}_{\mathbb{R}}|}\right) + \log|\Delta_K|. \tag{23}$$

Proof. The first identity involving the volume of the Arakelov ray class group follows from the exact sequence in Figure 1. The second one can be deduced from Equations (19) and (20). The third one follows from the volume computation of T in Equation (9).

The bound on the logarithm is obtained by using $\frac{|\mu_{K^{\mathfrak{m},1}}|}{|\mu_K|} \leq 1$, applying the class number formula [58, VII.§5, Cor 5.11] and Louboutin's bound [53] on the residue ρ_K of the Dedekind zeta function at $s = 1$:

$$\begin{aligned}|\mathrm{Pic}^0_{K^{\mathfrak{m}}}| &\leq \phi(\mathfrak{m}_0) \cdot h_K \cdot R_K \cdot \sqrt{\mathbb{r}+1} \cdot 2^{|\mathfrak{m}_{\mathbb{R}}|} = \frac{\phi(\mathfrak{m}_0) \cdot \rho_K \cdot \sqrt{|\Delta_K|} \cdot |\mu_K| \cdot \sqrt{n_{\mathbb{R}} + n_{\mathbb{C}}}}{2^{n_{\mathbb{R}} - |\mathfrak{m}_{\mathbb{R}}|} \cdot (2\pi)^{n_{\mathbb{C}}}} \\ &\leq \phi(\mathfrak{m}_0) \cdot 2^{|\mathfrak{m}_{\mathbb{R}}|} \cdot \sqrt{|\Delta_K|} \cdot \rho_K \leq \phi(\mathfrak{m}_0) \cdot 2^{|\mathfrak{m}_{\mathbb{R}}|} \cdot \sqrt{|\Delta_K|} \cdot \left(\frac{e \log |\Delta_K|}{2(n-1)} \right)^{n-1} \\ &\leq \phi(\mathfrak{m}_0) \cdot 2^{|\mathfrak{m}_{\mathbb{R}}|} \cdot |\Delta_K|.\end{aligned}$$

The first inequality follows from $\frac{|\mu_K|\sqrt{n_{\mathbb{R}}+n_{\mathbb{C}}}}{2^{n_{\mathbb{R}}}(2\pi)^{n_{\mathbb{C}}}} \leq \frac{n^{3/2}}{2^n} \leq 1$. The last inequality follows from the fact that $\frac{e \log |x|}{|x|} \leq 1$ for all $x \in \mathbb{R}$. This inequality instantiated with $x = |\Delta_K|^{\frac{1}{2(n-1)}}$ then yields $\left(\frac{e \log |\Delta_K|}{2(n-1)} \right)^{n-1} \leq \sqrt{|\Delta_K|}$. □

5.2. **Divisors and ideal lattices.** The Exp map introduced in Section 2.8 restricts to the group homomorphism

$$\mathrm{Exp} : \mathrm{Div}_{K^{\mathfrak{m}}} \longrightarrow \mathrm{IdLat}_K : \mathbf{a} \longmapsto \mathrm{Exp}_\infty(\mathbf{a}_\infty) \cdot \mathrm{Exp}_{\mathrm{f}}(\mathbf{a}_{\mathrm{f}}),$$

sending each Arakelov divisor to an ideal lattice. Recall that for any divisor $\mathbf{a} \in \mathrm{Div}_{K^{\mathfrak{m}}}$, we have $\mathrm{Vol}(\mathrm{Exp}(\mathbf{a})) = \sqrt{|\Delta_K|} \cdot e^{\deg(\mathbf{a})}$.

5.3. **τ-equivalent elements and generators of an Arakelov ray divisor.** Suppose that $\mathfrak{m}$ is a modulus, and let $\mathfrak{a}$ be any ideal coprime to $\mathfrak{m}_0$. An element $\alpha \in \mathfrak{a}$ is said to be τ-equivalent (with respect to $\mathfrak{m}$) if $\alpha \equiv \tau \bmod \mathfrak{m}_0$ and $\sigma_\nu(\alpha)/\sigma_\nu(\tau) \in \mathbb{R}_{>0}$ for all real $\nu \mid \mathfrak{m}_\infty$. If additionally, $\mathfrak{a}$ is *principal*, any τ-equivalent element α such that $\mathfrak{a} = (\alpha)$ is called a τ-equivalent *generator* of $\mathfrak{a}$.

These notions generalize to Arakelov ray divisors. As we can see Arakelov ray divisors as ideal lattices $x\mathfrak{a}$, an element of such a divisor is just an element of the shape $x\alpha$ where $\alpha \in \mathfrak{a}$ and $x \in K_{\mathbb{R}}$. Similarly, a generator of such divisor is an element in $K_{\mathbb{R}}^\times$ of the shape $x\alpha$, where α is a generator of $\mathfrak{a}$. The precise definitions are as follows.

Definition 5.2 (τ-equivalent elements of an Arakelov ray divisor)**.** *Let $\tau \in K^{\mathfrak{m}}$ and let $\mathbf{a} \in \mathrm{Div}_{K^{\mathfrak{m}}}$ be an Arakelov ray divisor with an infinite part $\mathbf{a}_\infty$ and a finite part $\mathbf{a}_{\mathrm{f}}$ (see Equation (14)). We define the set of τ-equivalent elements $\mathrm{Exp}(\mathbf{a})_\tau \subseteq K_{\mathbb{R}}$ of $\mathbf{a}$ by the following rule*

$$\mathrm{Exp}(\mathbf{a})_\tau := \mathrm{Exp}_\infty(\mathbf{a}_\infty) \cdot (\mathrm{Exp}_{\mathrm{f}}(\mathbf{a}_{\mathrm{f}}) \cap \tau K^{\mathfrak{m},1})$$

Equivalently, we can write

$$\mathrm{Exp}(\mathbf{a})_\tau = \{\alpha \in \mathrm{Exp}(\mathbf{a}) \mid \mathrm{Exp}_\infty(-\mathbf{a}_\infty) \cdot \alpha \in \tau K^{\mathfrak{m},1}\}.$$

Definition 5.3 (τ-equivalent generators of an Arakelov ray divisor)**.** *Let $\tau \in K^{\mathfrak{m}}$ and let $\mathbf{a} \in \mathrm{Div}_{K^{\mathfrak{m}}}$ be an Arakelov ray divisor with an infinite part $\mathbf{a}_\infty$ and a finite part $\mathbf{a}_{\mathrm{f}}$ (see Equation (14)). We define the set of τ-equivalent generators $\mathrm{Exp}(\mathbf{a})_\tau^\times \subseteq K_{\mathbb{R}}$ of $\mathbf{a}$ by the following rule*

$$\mathrm{Exp}(\mathbf{a})_\tau^\times := \begin{cases} \mathrm{Exp}_\infty(\mathbf{a}_\infty) \cdot (\kappa \cdot \mathcal{O}_K^\times \cap \tau K^{\mathfrak{m},1}) \subseteq \mathrm{Exp}(\mathbf{a}) & \textit{if } \mathrm{Exp}_{\mathrm{f}}(\mathbf{a}_{\mathrm{f}}) = (\kappa) \\ & \textit{for some } \kappa \in K^{\mathfrak{m}} \\ \emptyset & \textit{otherwise} \end{cases}$$

Equivalently, we can write

$$\mathrm{Exp}(\mathbf{a})_\tau^\times = \{\alpha \in \mathrm{Exp}(\mathbf{a}) \mid \mathrm{Exp}_\infty(-\mathbf{a}_\infty)\cdot\alpha \in \tau K^{\mathfrak{m},1} \textit{ is a generator of } \mathrm{Exp}_{\mathrm{f}}(\mathbf{a}_{\mathrm{f}})\}.$$

5.4. **Uniform sampling of prime ideals in a certain Arakelov class.** In the main result of this part, we will need to sample primes $\mathfrak{p}$ that satisfy $[d^0(\mathfrak{p})] \in G$ for a certain finite-index subgroup of $\mathrm{Pic}^0_{K^\mathfrak{m}}$. The procedure explaining this and the respective running time can be found in this lemma; the proof is provided in Appendix A.2.

Lemma 5.4 (Uniform sampling of prime ideals, ERH)**.** *Let a basis of $\mathcal{O}_K$ be known, and let $\mathfrak{m}_0 \subseteq \mathcal{O}_K$ be a modulus. Let $G \subseteq \mathrm{Pic}^0_{K^\mathfrak{m}}$ a finite-index subgroup and let $\mathbf{O}_G$ be an oracle that on input an ideal $\mathfrak{c}$ returns whether $[d^0(\mathfrak{c})] \in G$ or not. Let $\mathcal{P}_B = \{\mathfrak{p}$ prime ideal of $K \mid \mathcal{N}(\mathfrak{p}) \le B, \mathfrak{p} \nmid \mathfrak{m}_0$ and $[d^0(\mathfrak{p})] \in G\}$.*

There exists a bound

$$B_0 = \widetilde{O}\Big([\mathrm{Pic}^0_{K^\mathfrak{m}} : G]^2 \cdot \big[n^2(\log\log(1/\varepsilon))^2 + n^2(\log(1/\tilde{s}))^2 + (\log(|\Delta_K|\mathcal{N}(\mathfrak{m})))^2\big]\Big)$$

such that for all $B \ge B_0$, one can sample uniformly from $\mathcal{P}_B$ in expected time $O([\mathrm{Pic}^0_{K^\mathfrak{m}} : G]\cdot n^3 \log^2 B)$ and using $O([\mathrm{Pic}^0_{K^\mathfrak{m}} : G]\cdot n \log B)$ queries to $\mathbf{O}_G$.

6. Random walks in Arakelov ray class groups

In this section, we present a theorem on the rapid mixing of random walks on finite-index subgroups of Arakelov ray class groups. It is a generalization of the main theorem of [14], from Arakelov class groups to finite-index subgroups G of Arakelov ray class groups. Starting with a point in the hyperplane $H \hookrightarrow \mathrm{Div}^0_{K^\mathfrak{m}}$, sampled according to a Gaussian distribution, we prove that multiplying this point sufficiently often by small random prime ideals whose Arakelov classes are in $G \subseteq \mathrm{Pic}^0_{K^\mathfrak{m}}$ yields a random ray divisor that is very close to uniformly distributed in G, (the concerning subgroup of the Arakelov ray class group).

Definition 6.1 (Random Walk Distribution in $\mathrm{Div}^0_{K^\mathfrak{m}}$)**.** *For a number field K and a finite-index subgroup $G \subseteq \mathrm{Pic}^0_{K^\mathfrak{m}}$, we denote by $\mathcal{W}_G(B,N,s) \in L_1(\mathrm{Div}^0_{K^\mathfrak{m}})$ the distribution on $\mathrm{Div}^0_{K^\mathfrak{m}}$ that is obtained by the following random walk procedure.*

Sample $a = (a_\nu)_\nu \in H \subseteq \mathrm{Div}^0_{K^\mathfrak{m}}$ according to a centered Gaussian distribution with standard deviation $s > 0$ (see Section 2.9). Subsequently, sample N ideals $\mathfrak{p}_j$ uniformly from the set of all prime ideals coprime with $\mathfrak{m}_0$, with norm bounded by B and whose Arakelov class $[d^0(\mathfrak{p}_j)]$ lies in G. Finally, output $a + \sum_{j=1}^N d^0(\mathfrak{p}_j)$, where $a \in \mathrm{Div}^0_{K^\mathfrak{m}}$ is understood via the inclusion $H \subseteq \mathrm{Div}^0_{K^\mathfrak{m}}$.

Definition 6.2. *For any distribution $\mathcal{D}$ on $\mathrm{Div}^0_{K^\mathfrak{m}}$, we define the distribution $[\mathcal{D}]$ on $\mathrm{Pic}^0_{K^\mathfrak{m}}$ by the following rule:*

$$[\mathcal{D}](\cdot) = \sum_{\kappa \in K^{\mathfrak{m},1}/\mu_{K^{\mathfrak{m},1}}} \mathcal{D}(\cdot + (\!|\kappa|\!)).$$

This distribution $[\mathcal{D}]$ on $\mathrm{Pic}^0_{K^\mathfrak{m}}$ arises whenever one samples $\mathbf{a} \leftarrow \mathcal{D}$ and subsequently takes the Arakelov ray class $[\mathbf{a}] \in \mathrm{Pic}^0_{K^\mathfrak{m}}$.

Definition 6.3 (Random Walk Distribution in $\mathrm{Pic}^0_{K^\mathfrak{m}}$)**.** *We denote by $[\mathcal{W}_G(B,N,s)]$ the distribution on the Arakelov class group obtained by sampling $\mathbf{a}$ from $\mathcal{W}_G(B,N,s)$ and taking the Arakelov ray class $[\mathbf{a}] \in \mathrm{Pic}^0_{K^\mathfrak{m}}$ (as in Definition 6.2).*

Theorem 6.4 (Random Walks in finite-index subgroups of the Arakelov Ray Class Group, ERH)**.** *Let $\varepsilon > 0$ and $s > 0$ be any positive real numbers and let $k \in \mathbb{R}_{>0}$ be a positive real number as well. Let $G \subseteq \mathrm{Pic}^0_{K^{\mathfrak{m}}}$ be a finite-index subgroup of the Arakelov ray class group. Putting*[5] $\tilde{s} = \min(\sqrt{2} \cdot s, 1/\eta_1(\mathrm{Log}(\mathcal{O}^{\times}_{K^{\mathfrak{m}},1})^{\vee}))$*, there exists a bound*

$$B = \widetilde{O}\Big([\mathrm{Pic}^0_{K^{\mathfrak{m}}} : G]^2 \cdot n^{2k} \left[n^2(\log\log(1/\varepsilon))^2 + n^2(\log(1/\tilde{s}))^2 + (\log(|\Delta_K|\mathcal{N}(\mathfrak{m})))^2\right]\Big)$$

such that for any integer

$$N \geq \left\lceil \frac{1}{2k\log n} \cdot (\mathbb{r} \cdot \log(1/\tilde{s}) + 2\log(1/\varepsilon) + \log|\mathrm{Pic}^0_{K^{\mathfrak{m}}}| - \log[\mathrm{Pic}^0_{K^{\mathfrak{m}}} : G] + 2) \right\rceil, \tag{24}$$

the random walk distribution $[\mathcal{W}_G(B, N, s)]$ is ε-close to uniform in $L_1(G)$, i.e.,

$$\|[\mathcal{W}_G(B, N, s)] - \mathcal{U}(G)\|_1 \leq \varepsilon,$$

where $\mathcal{N}(\mathfrak{m}) = \mathcal{N}(\mathfrak{m}_0) \cdot 2^{|\mathfrak{m}_{\mathbb{R}}|}$, $\mathbb{r} = \dim(H) = n_{\mathbb{R}} + n_{\mathbb{C}} - 1$ and $|\mathfrak{m}_{\mathbb{R}}|$ is the number of different real places dividing $\mathfrak{m}$ (see Section 2.4).

Proof. The proof is very similar to that of [14], adapted to account for the ray and the finite-index subgroup. Details can be found in Appendix A.1. □

The following instantiation of the random walk theorem gives the appropriate parameters for the main application of this paper, the sampling algorithm (see Algorithm 2 and Theorem 9.5)

Corollary 6.5 (ERH)**.** *Let $n = [K : \mathbb{Q}] \geq 2$, $s = 1/n^2$, let $G \subseteq \mathrm{Pic}^0_{K^{\mathfrak{m}}}$ be a finite-index subgroup and let $\varepsilon > 0$ be an error parameter. There exists a bound $B = \widetilde{O}\Big([\mathrm{Pic}^0_{K^{\mathfrak{m}}} : G]^2 \cdot n^2 \cdot \left[n^2(\log\log(1/\varepsilon))^2 + (\log(|\Delta_K|\mathcal{N}(\mathfrak{m})))^2\right]\Big)$ such that for $N = \lceil 7n + 2\log(1/\varepsilon) + \log|\mathrm{Pic}^0_{K^{\mathfrak{m}}}| - \log[\mathrm{Pic}^0_{K^{\mathfrak{m}}} : G] + 2\rceil$ the random walk distribution $[\mathcal{W}(B, N, s)]$ on $\mathrm{Pic}^0_{K^{\mathfrak{m}}}$ is ε-close to uniform in $L_1(G)$, i.e.,*

$$\|[\mathcal{W}(B, N, s)] - \mathcal{U}(G)\|_1 \leq \varepsilon.$$

Proof. This formulation of the random walk theorem is obtained by instantiating Theorem 6.4 with $s = 1/n^2$ and $k = 1$. To obtain the bounds on B and N, we use the inequality $1/\tilde{s} = \max(n^2/\sqrt{2}, \eta_1(\mathrm{Log}(\mathcal{O}^{\times}_{K^{\mathfrak{m}},1})^{\vee})) \leq 2800 \cdot n^2$, which we will verify at the end of this proof.

One then gets the bound on B by applying Theorem 6.4 and simply moving the $n^2(\log(1/\tilde{s}))^2$ into the polylogarithmic factors. For the lower bound on N, note that $\frac{1}{2\log(2)} \leq 1$ and[6] $\mathbb{r} \cdot \log(1/\tilde{s})/(2\log n) \leq 7n$. Hence, a sufficient lower bound on N is the one in Theorem 6.4 with $\frac{1}{2k\log n}$ removed and $\mathbb{r} \cdot \log(1/\tilde{s})$ replaced by $7n$.

As promised, we finish the proof by showing $1/\tilde{s} \leq 2800 \cdot n^2$. As $\mathrm{Log}(\mathcal{O}^{\times}_{K^{\mathfrak{m}},1}) \subseteq \mathrm{Log}(\mathcal{O}^{\times}_{K})$, we have $\mathrm{Log}(\mathcal{O}^{\times}_{K^{\mathfrak{m}},1})^{\vee} \supseteq \mathrm{Log}(\mathcal{O}^{\times}_{K})^{\vee}$. Therefore, the smoothing parameter

[5] Recall that for any lattice Λ, we write $\Lambda^{\vee}$ for its dual, and $\eta_1(\Lambda)$ for its smoothing parameter (see page 18).

[6] Using the bound $1/\tilde{s} \leq 2800 \cdot n^2$, we have $\frac{\mathbb{r}\cdot\log(1/\tilde{s})}{2\log n} \leq \frac{\mathbb{r}\cdot\log(2800\cdot n^2)}{2\log n} = \frac{\mathbb{r}\cdot[\log(2800)+2\log n]}{2\log n} \leq \mathbb{r} \cdot (\frac{\log(2800)}{2\log n} + 1) \leq (\frac{\log(2800)}{2\log 2} + 1)n \leq 7n$.

of $\mathrm{Log}(\mathcal{O}_{K^{\mathfrak{m},1}}^{\times})^{\vee}$ satisfies

$$\eta_1(\mathrm{Log}(\mathcal{O}_{K^{\mathfrak{m},1}}^{\times})^{\vee}) \leq \eta_1(\mathrm{Log}(\mathcal{O}_{K}^{\times})^{\vee}) \leq \frac{\sqrt{\mathbb{r}}}{\lambda_1(\mathrm{Log}(\mathcal{O}_{K}^{\times}))} \leq 1000 \cdot n \cdot \log(n)^3 \leq 2800 \cdot n^2.$$

Here, the first inequality follows from the fact that $\eta_1(\Lambda) \leq \eta_1(\Lambda')$ if $\Lambda \supseteq \Lambda'$. The second inequality holds for general lattices [55, Lemma 3.2], the third inequality holds by the fact that $1/\lambda_1(\mathrm{Log}(\mathcal{O}_{K}^{\times})) \leq 1000 \cdot \sqrt{n} \cdot \log(n)^3$ (see Lemma 2.14) and $\mathbb{r} \leq n$, and the last inequality by $x \log(x)^3 \leq 1.4 \cdot x^2$ for all $x > 0$. □

7. Average densities and sampling

7.1. **Result.** Let $\mathcal{S} \subseteq \mathcal{I}_K$ be a set of integral ideals, and assume that $\mathbf{a}$ is an Arakelov ray divisor whose class is uniformly distributed. In this section, we show that one can sample elements β such that $\beta\mathbf{a}^{-1}$ corresponds to an ideal of the family $\mathcal{S}$ with the probability one would naturally expect, i.e., proportional to the *density* of $\mathcal{S}$. This is made precise in Theorem 7.1 via the notion of *local density*. Recall that the local density of a set of ideals $\mathcal{S}$ is defined (Definition 2.11) as

$$\delta_{\mathcal{S}}[x] = \min_{t \in [x/e^n, x]} \frac{|\mathcal{S}(t)|}{\rho_K \cdot t} = \min_{t \in [x/e^n, x]} \frac{|\{\mathfrak{b} \in \mathcal{S} \mid \mathcal{N}(\mathfrak{b}) \leq t\}|}{\rho_K \cdot t},$$

where, $\rho_K = \lim_{s \to 1}(s-1)\zeta_K(s)$ (see Equation (4)).

Recall that for $r \in \mathbb{R}_{>0}$, the box of radius r in $K_{\mathbb{R}}$ is $r\mathcal{B}_\infty = \{(x_\sigma)_\sigma \in K_{\mathbb{R}} \mid |x_\sigma| \leq r\}$.

Theorem 7.1. *Let $G \subseteq \mathrm{Pic}_{K^{\mathfrak{m}}}^0$ be a finite-index subgroup of the Arakelov ray class group, and let $[\mathbf{b}] \in \mathrm{Pic}_{K^{\mathfrak{m}}}^0$ arbitrary, and let $\mathcal{D}$ be a distribution on $\mathrm{Div}_{K^{\mathfrak{m}}}^0$ such that $[\mathcal{D}]$ is uniform on the coset $G + [\mathbf{b}]$. Let $r \geq 8 \cdot n^2 \cdot \Gamma_K \cdot |\Delta_K|^{\frac{1}{2n}} \cdot \mathcal{N}(\mathfrak{m})^{1/n}$, where $\Gamma_K \leq \lambda_n^{\infty}(\mathcal{O}_K) \leq |\Delta_K|^{1/n}$ is defined in Equation (6), let $\tau \in K^{\mathfrak{m}}$ and let $\mathcal{S}$ be a set of integral ideals coprime to $\mathfrak{m}_0$ for which holds $[d^0(\mathcal{S})] \subseteq G + [\llparenthesis\tau\rrparenthesis] - [\mathbf{b}]$, with local density $\delta_{\mathcal{S}}[r^n]$ at r^n. Then*

$$\mathop{\mathbb{E}}_{\mathbf{a} \leftarrow \mathcal{D}} \left[\Pr_{\alpha \leftarrow \mathrm{Exp}(\mathbf{a}) \cap r\mathcal{B}_\infty} \left[(\alpha) \cdot \mathrm{Exp}(-\mathbf{a}) \in \mathcal{S} \;\middle|\; \alpha \cdot \mathrm{Exp}_\infty(-\mathbf{a}_\infty) \in \tau K^{\mathfrak{m},1} \right] \right] \tag{25}$$

$$\geq \frac{\mathcal{N}(\mathfrak{m}_0)}{\phi(\mathfrak{m}_0)} \cdot \frac{[\mathrm{Pic}_{K^{\mathfrak{m}}}^0 : G]}{3} \cdot \delta_{\mathcal{S}}[r^n] \tag{26}$$

$$\geq \frac{[\mathrm{Pic}_{K^{\mathfrak{m}}}^0 : G]}{3} \cdot \delta_{\mathcal{S}}[r^n] \tag{27}$$

where $\alpha \leftarrow \mathrm{Exp}(\mathbf{a}) \cap r\mathcal{B}_\infty$ is uniformly sampled from the finite set $\mathrm{Exp}(\mathbf{a}) \cap r\mathcal{B}_\infty$ and $\phi(\mathfrak{m}_0) = |K^{\mathfrak{m}_0}/K^{\mathfrak{m}_0,1}| \leq \mathcal{N}(\mathfrak{m}_0)$.

Remark 7.2. *The factor $1/3$ in Equations (26) and (27) can be made arbitrarily close to 1 by increasing the radius $r \in \mathbb{R}$ and widening the density interval $[x/e^n, x]$ in Definition 2.11.*

Remark 7.3. *It is possible, with essentially the same proof, to rephrase this theorem in such a way that it concerns an* intersection *of events instead of a conditional probability. The probability then also depends on the number $\mathcal{N}(\mathfrak{m}) = \mathcal{N}(\mathfrak{m}_0) \cdot 2^{|\mathfrak{m}_{\mathbb{R}}|}$, where $|\mathfrak{m}_{\mathbb{R}}|$ is the number of real places dividing $\mathfrak{m}_\infty$. Under the same conditions as* *in Theorem 7.1, one can prove that*

$$\Pr_{\substack{\mathbf{a}\leftarrow\mathcal{D}\\ \alpha\leftarrow\mathrm{Exp}(\mathbf{a})\cap r\mathcal{B}_\infty}} \left[\begin{array}{c}(\alpha)\cdot\mathrm{Exp}(-\mathbf{a})\in\mathcal{S},\ \textit{and}\\ \alpha\cdot\mathrm{Exp}_\infty(-\mathbf{a}_\infty)\in\tau K^{\mathfrak{m},1}\end{array}\right] \geq \frac{\mathcal{N}(\mathfrak{m}_0)}{\phi(\mathfrak{m}_0)}\cdot\frac{[\mathrm{Pic}^0_{K^\mathfrak{m}}:G]}{3\cdot\mathcal{N}(\mathfrak{m})}\cdot\delta_{\mathcal{S}}[r^n]. \tag{28}$$

7.2. **Proof of Theorem 7.1.**

7.2.1. *Fixing the ideal $\mathfrak{c}\in\mathcal{S}$ and the Arakelov divisor $\mathbf{a}\in\mathrm{Div}^0_{K^\mathfrak{m}}$.* We concentrate on the 'inner probability' of Equation (25) in Theorem 7.1 in the case where $\mathcal{S}=\{\mathfrak{c}\}$ consists of a *single* integral ideal. We denote

$$p_{\mathbf{a},\mathfrak{c}} = \Pr_{\alpha\leftarrow\mathrm{Exp}(\mathbf{a})\cap r\mathcal{B}_\infty}\left[(\alpha)\cdot\mathrm{Exp}(-\mathbf{a})=\mathfrak{c}\ \middle|\ \alpha\cdot\mathrm{Exp}_\infty(-\mathbf{a}_\infty)\in\tau K^{\mathfrak{m},1}\right], \tag{29}$$

where we leave the dependency on $r\in\mathbb{R}_{>0}$, the modulus $\mathfrak{m}$ and $\tau\in K^\mathfrak{m}/K^{\mathfrak{m},1}$ implicit. By the law of conditional probability, we have that $p_{\mathbf{a},\mathfrak{c}}$ in Equation (29) equals

$$\Pr_{\alpha\leftarrow\mathrm{Exp}(\mathbf{a})\cap r\mathcal{B}_\infty}\left[\begin{array}{c}(\alpha)\cdot\mathrm{Exp}(-\mathbf{a})=\mathfrak{c}\\ \text{and}\\ \alpha\cdot\mathrm{Exp}_\infty(-\mathbf{a}_\infty)\in\tau K^{\mathfrak{m},1}\end{array}\right]\Bigg/\Pr_{\alpha\leftarrow\mathrm{Exp}(\mathbf{a})\cap r\mathcal{B}_\infty}\left[\alpha\cdot\mathrm{Exp}_\infty(-\mathbf{a}_\infty)\in\tau K^{\mathfrak{m},1}\right] \tag{30}$$

The following lemma addresses the probability values of the numerator and denominator in Equation (30) separately.

Lemma 7.4. *Let $\mathfrak{m}=\mathfrak{m}_0\mathfrak{m}_\infty$ be a modulus, let $\tau\in K^\mathfrak{m}/K^{\mathfrak{m},1}$, let $\mathbf{a}\in\mathrm{Div}^0_{K^\mathfrak{m}}$ be a fixed Arakelov ray divisor, and let $\mathfrak{c}\in\mathcal{I}_K^\mathfrak{m}$ be an integral ideal. Then*

$$\Pr_{\alpha\leftarrow\mathrm{Exp}(\mathbf{a})\cap r\mathcal{B}_\infty}\left[\begin{array}{c}(\alpha)\cdot\mathrm{Exp}(-\mathbf{a})=\mathfrak{c}\\ \textit{and}\\ \alpha\cdot\mathrm{Exp}_\infty(-\mathbf{a}_\infty)\in\tau K^{\mathfrak{m},1}\end{array}\right] = \frac{|\mathrm{Exp}(\mathbf{a}+d(\mathfrak{c}))^\times_\tau\cap r\mathcal{B}_\infty|}{|\,\mathrm{Exp}(\mathbf{a})\cap r\mathcal{B}_\infty|}, \tag{31}$$

and, there exists some $\tilde{\tau}\in K_\mathbb{R}$ such that

$$\Pr_{\alpha\leftarrow\mathrm{Exp}(\mathbf{a})\cap r\mathcal{B}_\infty}\left[\alpha\cdot\mathrm{Exp}_\infty(-\mathbf{a}_\infty)\in\tau K^{\mathfrak{m},1}\right] = \frac{|(\mathrm{Exp}(\mathbf{a})\mathfrak{m}_0+\tilde{\tau})\cap r\mathcal{B}_\infty\cap\tau K_\mathbb{R}^{\mathfrak{m}_\infty}|}{|\,\mathrm{Exp}(\mathbf{a})\cap r\mathcal{B}_\infty|}, \tag{32}$$

where the sampling $\alpha\leftarrow\mathrm{Exp}(\mathbf{a})\cap r\mathcal{B}_\infty$ is uniform in both expressions, and where $\tau K_\mathbb{R}^{\mathfrak{m}_\infty}=\{(x_\sigma)_\sigma\in K_\mathbb{R}\mid x_\sigma/\sigma(\tau)>0 \text{ for real } \sigma\mid\mathfrak{m}_\infty\}$.

Proof. By examining Definition 5.3 closely, noting that $\mathrm{Exp}_\mathrm{f}((\mathbf{a}+d(\mathfrak{c}))_\mathrm{f})=\mathrm{Exp}_\mathrm{f}(\mathbf{a}_\mathrm{f})\cdot\mathfrak{c}\in\mathcal{I}_K^\mathfrak{m}$, we see that for all $\alpha\in\mathrm{Exp}(\mathbf{a})$,

$$(\alpha)\cdot\mathrm{Exp}(-\mathbf{a})=\mathfrak{c} \text{ and } \alpha\cdot\mathrm{Exp}_\infty(-\mathbf{a}_\infty)\in\tau K^{\mathfrak{m},1} \iff \alpha\in\mathrm{Exp}(\mathbf{a}+d(\mathfrak{c}))^\times_\tau.$$

As the number of choices for $\alpha\in\mathrm{Exp}(\mathbf{a})\cap r\mathcal{B}_\infty$ equals $|\,\mathrm{Exp}(\mathbf{a})\cap r\mathcal{B}_\infty|$, the number of good choices equals $|\mathrm{Exp}(\mathbf{a}+d(\mathfrak{c}))^\times_\tau\cap r\mathcal{B}_\infty|$ and since the sampling procedure is uniform, we arrive at the first probability claim. For the second probability claim, write $\mathfrak{a}=\mathrm{Exp}_\mathrm{f}(\mathbf{a}_\mathrm{f})\in\mathcal{I}_K^\mathfrak{m}$, for conciseness. We note that for $\alpha\in\mathrm{Exp}(\mathbf{a})$, $\alpha\cdot\mathrm{Exp}_\infty(-\mathbf{a}_\infty)\in\tau K^{\mathfrak{m},1}$ is equivalent to

$$\alpha\cdot\mathrm{Exp}_\infty(-\mathbf{a}_\infty)\in\mathrm{Exp}_\mathrm{f}(\mathbf{a}_\mathrm{f})\cap\tau K^{\mathfrak{m},1}=\mathfrak{a}\cap\tau K^{\mathfrak{m},1}=(\mathfrak{a}\mathfrak{m}_0+\tau')\cap\tau K_\mathbb{R}^{\mathfrak{m}_\infty},$$

where $\tau'\in\mathfrak{a}$ is such that $\tau'\equiv\tau$ modulo $\mathfrak{m}_0$ (note that $\mathfrak{a}$ and $\mathfrak{m}_0$ are coprime).

So, for $\alpha\in\mathrm{Exp}(\mathbf{a})\cap r\mathcal{B}_\infty$, the statement $\alpha\cdot\mathrm{Exp}_\infty(-\mathbf{a}_\infty)\in\tau K^{\mathfrak{m},1}$ is equivalent to

$$\alpha\in\mathrm{Exp}_\infty(\mathbf{a}_\infty)\big((\mathfrak{a}\mathfrak{m}_0+\tau')\cap\tau K_\mathbb{R}^{\mathfrak{m}_\infty}\big)\cap\mathcal{B}_\infty=(\mathrm{Exp}(\mathbf{a})\mathfrak{m}_0+\tilde{\tau})\cap\mathcal{B}_\infty\cap\tau K_\mathbb{R}^{\mathfrak{m}_\infty}$$

where we use the fact that $\operatorname{Exp}(\mathbf{a})\tau K_{\mathbb{R}}^{\mathfrak{m}_\infty} = \tau K_{\mathbb{R}}^{\mathfrak{m}_\infty}$ and where we put $\tilde{\tau} = \operatorname{Exp}_\infty(\mathbf{a}_\infty)\tau' \in K_{\mathbb{R}}$. This proves the claim. □

By combining Equations (30) to (32) and scratching redundant factors, one concludes that there exists $\tilde{\tau} \in \operatorname{Exp}(\mathbf{a})$ such that

$$p_{\mathbf{a},\mathfrak{c}} = \frac{|\operatorname{Exp}(\mathbf{a} + d(\mathfrak{c}))_\tau^\times \cap r\mathcal{B}_\infty|}{|(\operatorname{Exp}(\mathbf{a})\mathfrak{m}_0 + \tilde{\tau}) \cap r\mathcal{B}_\infty \cap \tau K_{\mathbb{R}}^{\mathfrak{m}_\infty}|}. \tag{33}$$

7.2.2. *Estimating* $|(\operatorname{Exp}(\mathbf{a})\mathfrak{m}_0 + \tilde{\tau}) \cap r\mathcal{B}_\infty^{\mathfrak{m}} \cap \tau K_{\mathbb{R}}^{\mathfrak{m}_\infty}|$. When the radius r is sufficiently large compared to the lattice $\operatorname{Exp}(\mathbf{a})\mathfrak{m}_0 \subseteq K_{\mathbb{R}}$, one can deduce that for $\mathbf{a} \in \operatorname{Div}_{K^{\mathfrak{m}}}^0$ the number of points in $(\operatorname{Exp}(\mathbf{a})\mathfrak{m}_0 + \tilde{\tau}) \cap r\mathcal{B}_\infty \cap \tau K_{\mathbb{R}}^{\mathfrak{m}_\infty}$ is approximately $\operatorname{Vol}(r\mathcal{B}_\infty) \cdot 2^{-|\mathfrak{m}_{\mathbb{R}}|}/\det(\mathfrak{m}_0)$, where the $2^{-|\mathfrak{m}_{\mathbb{R}}|}$ accounts for the fact that we only take the halves of the box for which $x_\sigma/\sigma(\tau) > 0$ at the real embeddings $\sigma \mid \mathfrak{m}_\infty$. More precisely, we instantiate Lemma 2.8 with

- $\Lambda = \operatorname{Exp}(\mathbf{a} + d(\mathfrak{m}_0)) \subseteq K_{\mathbb{R}}$, for which holds $\operatorname{Vol}(\Lambda) = \mathcal{N}(\mathfrak{m}_0) \cdot \sqrt{|\Delta_K|}$,
- $t = \tilde{\tau} \in K_{\mathbb{R}}$,
- $(t')_\sigma = \operatorname{sgn}(\sigma(\tau)) \cdot 1/2$ for real $\nu \mid \mathfrak{m}_\infty$, and $(t')_\sigma = 0$ otherwise. Here $\operatorname{sgn}(\sigma(\tau))$ is the sign of $\sigma(\tau)$.
- $X = \{(x_\sigma)_\sigma \in K_{\mathbb{R}} \mid |x_\sigma| \leq 1$ and $|x_\sigma| \leq \frac{1}{2}$ if σ is real and $\sigma \mid \mathfrak{m}_\infty\}$, such that $X + t' = \mathcal{B}_\infty \cap \tau K_{\mathbb{R}}^{\mathfrak{m}_\infty}$. Note that $\operatorname{Vol}(X) = 2^{n_{\mathbb{R}} - |\mathfrak{m}_{\mathbb{R}}|} \cdot (2\pi)^{n_{\mathbb{C}}}$, due to the hybrid complex-real nature of $K_{\mathbb{R}}^{\mathfrak{m}_\infty}$ and taking account for the required positivity (or negativity, depending on τ) at the real places $\nu \mid \mathfrak{m}_\infty$.
- $c = n \cdot \mathcal{N}(\mathfrak{m}_0)^{\frac{1}{n}} \cdot \Gamma_K \cdot |\Delta_K|^{\frac{1}{2n}} \geq 2 \cdot \operatorname{cov}(\operatorname{Exp}(\mathbf{a} + d(\mathfrak{m}_0)))$ (see Lemma 2.13(i) and (iv)), so that the Voronoi cell $\mathcal{V}_0$ of $\operatorname{Exp}(\mathbf{a} + d(\mathfrak{m}_0))$ lies in cX.

This yields, for $r > 8 \cdot n^2 \cdot \mathcal{N}(\mathfrak{m}_0)^{\frac{1}{n}} \cdot \Gamma_K \cdot |\Delta_K|^{\frac{1}{2n}} \geq 8nc$,

$$|(\operatorname{Exp}(\mathbf{a})\mathfrak{m}_0 + \tilde{\tau}) \cap r\mathcal{B}_\infty \cap \tau K_{\mathbb{R}}^{\mathfrak{m}_\infty}| \in [e^{-1/4}, e^{1/4}] \cdot \frac{r^n \cdot 2^{n_{\mathbb{R}} - |\mathfrak{m}_{\mathbb{R}}|} \cdot (2\pi)^{n_{\mathbb{C}}}}{\mathcal{N}(\mathfrak{m}_0) \cdot \sqrt{|\Delta_K|}}. \tag{34}$$

Applying this to the denominator of Equation (33), we directly deduce that

$$p_{\mathbf{a},\mathfrak{c}} \in [e^{-1/4}, e^{1/4}] \cdot \frac{\sqrt{|\Delta_K|} \cdot \mathcal{N}(\mathfrak{m}_0) \cdot 2^{|\mathfrak{m}_{\mathbb{R}}|}}{r^n \cdot 2^{n_{\mathbb{R}}} \cdot (2\pi)^{n_{\mathbb{C}}}} \cdot |\operatorname{Exp}(\mathbf{a} + d(\mathfrak{c}))_\tau^\times \cap r\mathcal{B}_\infty| \tag{35}$$

7.2.3. *Estimating the probability of sampling a fixed ideal for a* <u>*random*</u> *Arakelov divisor.* Still focusing on the simplified case where $\mathcal{S} = \{\mathfrak{c}\}$, the goal of this proof is to find a lower bound for $\underset{\mathbf{a} \leftarrow \mathcal{D}}{\mathbb{E}}[p_{\mathbf{a},\mathfrak{c}}]$. By linearity of expectation, we have

$$\underset{\mathbf{a} \leftarrow \mathcal{D}}{\mathbb{E}}[p_{\mathbf{a},\mathfrak{c}}] \in [e^{-1/4}, e^{1/4}] \cdot \frac{\sqrt{|\Delta_K|} \cdot \mathcal{N}(\mathfrak{m}_0) \cdot 2^{|\mathfrak{m}_{\mathbb{R}}|}}{r^n \cdot 2^{n_{\mathbb{R}}} \cdot (2\pi)^{n_{\mathbb{C}}}} \cdot \underset{\mathbf{a} \leftarrow \mathcal{D}}{\mathbb{E}}\left[|\operatorname{Exp}(\mathbf{a} + d(\mathfrak{c}))_\tau^\times \cap r\mathcal{B}_\infty|\right]. \tag{36}$$

So it remains to focus on the expected value of $|\operatorname{Exp}(\mathbf{a} + d(\mathfrak{c}))_\tau^\times \cap r\mathcal{B}_\infty|$ for $\mathbf{a} \leftarrow \mathcal{D}$.

7.2.4. *The number* $|\operatorname{Exp}(\mathbf{a} + d(\mathfrak{c}))_\tau^\times \cap r\mathcal{B}_\infty|$ *only depends on the Arakelov ray* <u>*class*</u> *of* $\mathbf{a} \in \operatorname{Div}_{K^{\mathfrak{m}}}^0$. A fact that plays a large role in the full proof, is that the number $|\operatorname{Exp}(\mathbf{a} + d(\mathfrak{c}))_\tau^\times \cap r\mathcal{B}_\infty|$ of τ-equivalent generators in a box depends on the Arakelov ray <u>class</u> $[\mathbf{a}]$ rather than the divisor $\mathbf{a}$ itself. This has as a consequence that the involved probability distribution changes from $\mathcal{D}$ to $[\mathcal{D}] = \mathcal{U}(G)$, uniform on $G \subseteq \operatorname{Pic}_{K^{\mathfrak{m}}}^0$, which is easier to analyze. This fact, among others, is proven in the following lemma.

Lemma 7.5. *For all ray divisors $\mathbf{a} \in \mathrm{Div}^0_{K^{\mathfrak{m}}}$, elements $\tau, \tau' \in K^{\mathfrak{m}}$, ideals $\mathfrak{c} \in \mathcal{I}^{\mathfrak{m}}_K$ and real numbers $r > 0$ we have the following list of facts.*

(i) $|\mathrm{Exp}(\mathbf{a})^\times_\tau \cap r\mathcal{B}_\infty| = |\mathrm{Exp}(\mathbf{a} + \llparenthesis \tau' \rrparenthesis)^\times_{\tau\tau'} \cap r\mathcal{B}_\infty|$*, i.e., the number of τ-equivalent ray generators of $\mathbf{a}$ in a fixed box of radius r is equal to the number of $\tau\tau'$-equivalent ray generators of $\mathbf{a} + \llparenthesis \tau' \rrparenthesis$ in the same box.*

(ii) $|\mathrm{Exp}(\mathbf{a} + d(\mathfrak{c}))^\times_\tau \cap r\mathcal{B}_\infty| = |\mathrm{Exp}(\mathbf{a} + d^0(\mathfrak{c}))^\times_\tau \cap \frac{r}{\mathcal{N}(\mathfrak{c})^{1/n}}\mathcal{B}_\infty|$*, since the maps d^0 and d only differ by some scaling $\mathcal{N}(\mathfrak{c})^{1/n}$.*

(iii) Writing $\mathbf{a}_\infty = \sum_\nu a_\nu \cdot \llparenthesis \nu \rrparenthesis \in \mathrm{Div}^0_{K^{\mathfrak{m}}}$, we have

$$|\mathrm{Exp}(\mathbf{a}_\infty)^\times_1 \cap r\mathcal{B}_\infty| = |\mu_{K^{\mathfrak{m},1}}| \cdot |(\mathrm{Log}(\mathcal{O}^\times_{K^{\mathfrak{m},1}}) + (a_\nu)_\nu) \ \cap \ S_{\log(r)}|, \tag{37}$$

where $S_{\log r} = \{(b_\nu)_\nu \in \bigoplus_\nu \mathbb{R} \mid b_\nu \leq n_\nu \log(r)\,, \sum_\nu b_\nu = 0\} \subseteq H$ is a simplex (as in Lemma 7.6) and $\mathrm{Log}(\mathcal{O}^\times_{K^{\mathfrak{m},1}}) = \mathrm{Log}(\mathcal{O}^\times_K \cap K^{\mathfrak{m},1})$. Here, $n_\nu = 2$ if ν is complex and 1 otherwise.

Proof. For part (i), observe that multiplying by $\left(\frac{\sigma(\tau')}{|\sigma(\tau')|}\right)_\sigma \in K_\mathbb{R}$ yields a bijection from $\mathrm{Exp}(\mathbf{a})$ to $\mathrm{Exp}(\mathbf{a} + \llparenthesis \tau' \rrparenthesis)$, preserving the maximum norm. It remains to show that this bijection sends $\mathrm{Exp}(\mathbf{a})^\times_\tau$ to $\mathrm{Exp}(\mathbf{a} + \llparenthesis \tau' \rrparenthesis)^\times_{\tau'\tau}$. Using Definition 5.3 and assuming $\mathrm{Exp}_\mathrm{f}(\mathbf{a}_\mathrm{f}) = \kappa\mathcal{O}_K$ (and therefore $\mathrm{Exp}_\mathrm{f}([\mathbf{a} + \llparenthesis \tau' \rrparenthesis]_\mathrm{f}) = \tau'\kappa\mathcal{O}_K$), we have

$$\left(\frac{\sigma(\tau')}{|\sigma(\tau')|}\right)_\sigma \cdot \mathrm{Exp}(\mathbf{a})^\times_\tau = \left(\frac{1}{|\sigma(\tau')|}\right)_\sigma \cdot (\tau') \cdot \underbrace{\mathrm{Exp}_\infty(\mathbf{a}_\infty) \cdot (\kappa\mathcal{O}^\times_K \cap \tau K^{\mathfrak{m},1})}_{\mathrm{Exp}(\mathbf{a})^\times_\tau}$$

$$= \underbrace{\left(\frac{1}{|\sigma(\tau')|}\right)_\sigma \cdot \mathrm{Exp}_\infty(\mathbf{a}_\infty)}_{\mathrm{Exp}_\infty((\mathbf{a} + \llparenthesis \tau' \rrparenthesis)_\infty)} \cdot (\tau'\kappa\mathcal{O}^\times_K \cap \tau'\tau K^{\mathfrak{m},1}) = \mathrm{Exp}(\mathbf{a} + \llparenthesis \tau' \rrparenthesis)^\times_{\tau'\tau}$$

For part (ii), recall that multiplying the ideal lattice $\mathrm{Exp}(d(\mathfrak{c})) = \mathfrak{c} \subseteq K_\mathbb{R}$ by the scalar $\mathcal{N}(\mathfrak{c})^{-1/n}$ results in the ideal lattice $\mathrm{Exp}(d^0(\mathfrak{c}))$. Applying this scalar multiplication to the set $\mathrm{Exp}(\mathbf{a} + d(\mathfrak{c})) \cap r\mathcal{B}_\infty$ yields a bijective correspondence with $\mathrm{Exp}(\mathbf{a} + d^0(\mathfrak{c})) \cap \frac{r}{\mathcal{N}(\mathfrak{c})^{1/n}}\mathcal{B}_\infty$.

In part (iii) it is enough to show that the logarithm $\mathrm{Log} : K^\times_\mathbb{R} \to \mathrm{Log}(K^\times_\mathbb{R})$ takes $\mathrm{Exp}(\mathbf{a}_\infty)^\times_1$ to the shifted lattice $\mathrm{Log}(\mathcal{O}^\times_{K^{\mathfrak{m},1}}) + (a_\nu)_\nu \subset H$ and takes $r\mathcal{B}_\infty \cap K^0_\mathbb{R}$ to the simplex $S_{\log(r)} \subset H$. This logarithmic map is $|\mu_{K^{\mathfrak{m},1}}|$-to-one on $\mathrm{Exp}(\mathbf{a}_\infty)^\times_1$, as it sends roots of unity to the all-zero vector in $\mathrm{Log}(K^\times_\mathbb{R})$ (which is the unit in that group), yielding the extra factor $|\mu_{K^{\mathfrak{m},1}}|$ in Equation (37). Here, $\mu_{K^{\mathfrak{m},1}} = \mu_K \cap K^{\mathfrak{m},1}$, i.e., the roots of unity in $K^{\mathfrak{m},1}$. □

As a corollary of Lemma 7.5(i) we deduce that $|\mathrm{Exp}(\mathbf{a})^\times_\tau \cap r\mathcal{B}_\infty| = |\mathrm{Exp}(\mathbf{a} + \llparenthesis \kappa \rrparenthesis)^\times_\tau \cap r\mathcal{B}_\infty|$ for $\kappa \in K^{\mathfrak{m},1}$, i.e., the number of elements $|\mathrm{Exp}(\mathbf{a})^\times_\tau \cap r\mathcal{B}_\infty|$ only depends on the Arakelov ray <u>class</u> of $\mathbf{a}$ (next to $r \in \mathbb{R}$, $\mathfrak{m}$ and $\tau \in K^{\mathfrak{m}}$). Choose a (measurable) fundamental domain $F \subseteq \mathrm{Div}^0_{K^{\mathfrak{m}}}$ of the quotient group $\mathrm{Pic}^0_{K^{\mathfrak{m}}}$. Put $F_G = \{\mathbf{a} \in F \mid [\mathbf{a}] \in G\}$, which is a subdomain of F for the subgroup $G \subseteq \mathrm{Pic}^0_{K^{\mathfrak{m}}}$ and likewise put $F_{T^{\mathfrak{m}}} = \{\mathbf{a} \in F \mid [\mathbf{a}] \in T^{\mathfrak{m}}\}$, a fundamental domain of $T^{\mathfrak{m}}$ in $\mathrm{Pic}^0_{K^{\mathfrak{m}}}$. Note that $F_{T^{\mathfrak{m}}} \subseteq F_G$. By the assumption that $[\mathcal{D}]$ is uniform on $G + [\mathbf{b}]$, and $[d^0(\mathfrak{c})] \in [d^0(\mathcal{S})] \subseteq G + [(\!|\tau|\!)] - [\mathbf{b}]$, we deduce, writing $\tilde{r} = r\,\mathcal{N}(\mathfrak{c})^{-1/n}$,

$$\mathop{\mathbb{E}}_{\mathbf{a}\leftarrow\mathcal{D}}\left[|\mathrm{Exp}(\mathbf{a}+d(\mathfrak{c}))_\tau^\times \cap r\mathcal{B}_\infty|\right] = \mathop{\mathbb{E}}_{\mathbf{a}\leftarrow\mathcal{D}}\left[|\mathrm{Exp}(\mathbf{a}+d^0(\mathfrak{c}))_\tau^\times \cap \tilde{r}\mathcal{B}_\infty|\right] \tag{38}$$

$$= \mathop{\mathbb{E}}_{\mathbf{a}\leftarrow\mathcal{U}(F_G+[\mathbf{b}])}\left[|\mathrm{Exp}(\mathbf{a}+d^0(\mathfrak{c}))_\tau^\times \cap \tilde{r}\mathcal{B}_\infty|\right] = \mathop{\mathbb{E}}_{\mathbf{a}\leftarrow\mathcal{U}(F_G)}\left[|\mathrm{Exp}(\mathbf{a}+(\!|\tau|\!))_\tau^\times \cap \tilde{r}\mathcal{B}_\infty|\right] \tag{39}$$

$$= \mathop{\mathbb{E}}_{\mathbf{a}\leftarrow\mathcal{U}(F_G)}\left[|\mathrm{Exp}(\mathbf{a})_1^\times \cap \tilde{r}\mathcal{B}_\infty|\right] = \frac{1}{|G/T^{\mathfrak{m}}|}\mathop{\mathbb{E}}_{\mathbf{a}\leftarrow\mathcal{U}(F_{T^{\mathfrak{m}}})}\left[|\mathrm{Exp}(\mathbf{a})_1^\times \cap \tilde{r}\mathcal{B}_\infty|\right] \tag{40}$$

where the first equality follows from scaling (Lemma 7.5(ii)) and the second one by the fact that the random variable is an Arakelov ray class invariant (Lemma 7.5(i)) and that $[\mathcal{D}]$ is uniform on $G + [\mathbf{b}]$. The third equality holds because the class $[\mathbf{a} + d^0(\mathfrak{c})]$ for $\mathbf{a} \in \mathrm{Pic}^0_{K^{\mathfrak{m}}}$ uniformly distributed over $F_G + [\mathbf{b}]$ is distributed as $[\mathbf{a}' + (\!|\tau|\!)]$ for $\mathbf{a}' \leftarrow \mathcal{U}(F_G)$. This follows from the assumption $d^0(\mathfrak{c}) \in G + [(\!|\tau|\!)] - [\mathbf{b}]$. The fourth equality follows directly from Lemma 7.5(i), and the last equality follows from Definition 5.3. Namely, an Arakelov divisor $\mathbf{a}$ can only have generators if the ideal class of $\mathrm{Exp}(\mathbf{a}_{\mathfrak{f}})$ is trivial, i.e., if $[\mathbf{a}] \in T^{\mathfrak{m}}$. So, instead, $\mathbf{a}$ can be chosen uniformly from a fundamental domain $F_{T^{\mathfrak{m}}}$ of $T^{\mathfrak{m}}$ in $\mathrm{Div}^0_{K^{\mathfrak{m}}}$, with a correction factor of $\frac{1}{|G/T^{\mathfrak{m}}|}$ in the expected value.

7.2.5. *Volume of the simplex in H.* For the next step in the proof, we need to know the volume of the simplex $S_{\log r} \subseteq H$, hence the following lemma.

Lemma 7.6. *The volume of the simplex $S_\alpha = \{(b_\nu)_\nu \in H \mid b_\nu \leq n_\nu\alpha\} \subseteq H = \mathrm{Log}\, K^0_{\mathbb{R}}$ for $\alpha \in \mathbb{R}_{>0}$ is given by*

$$\mathrm{Vol}(S_\alpha) = \frac{\sqrt{\mathbb{r}+1}\cdot(n\alpha)^{\mathbb{r}}}{\mathbb{r}!},$$

where $\mathbb{r} = n_{\mathbb{R}} + n_{\mathbb{C}} - 1$ and where $n_\nu = 2$ whenever ν is complex and $n_\nu = 1$ when ν is real.

Proof. By applying to $S_\alpha \subseteq H$ the translation $c_\nu = n_\nu\alpha - b_\nu$, one can see that S_α is a regular $\mathbb{r}$-simplex $\{c \in \prod_\nu \mathbb{R} \mid \sum_\nu c_\nu = n \cdot \alpha \text{ and } c_\nu \geq 0\}$ with edge length $\sqrt{2}\cdot n \cdot \alpha$. Therefore, the volume of S_α equals $\mathrm{Vol}(S_\alpha) = \frac{(n\alpha)^{\mathbb{r}}\sqrt{\mathbb{r}+1}}{\mathbb{r}!}$ [66]. □

7.2.6. *Taking the logarithmic map into $H = \mathrm{Log}\, K^0_{\mathbb{R}}$.* Applying the logarithmic map on the set $\mathrm{Exp}(\mathbf{a})_1^\times \cap r \cdot \mathcal{N}(\mathfrak{c})^{-1/n}\mathcal{B}_\infty$, sends $\mathrm{Exp}(\mathbf{a})_1^\times$ to a shift of the logarithmic ray unit lattice $\mathrm{Log}(\mathcal{O}^\times_{K^{\mathfrak{m}},1}) \subseteq H$ and $r \cdot \mathcal{N}(\mathfrak{c})^{-1/n}\mathcal{B}_\infty$ to a simplex $S_{n\log r - \log\mathcal{N}(\mathfrak{c})}$, where $S_x = \mathrm{Log}(x\mathcal{B}_\infty) \subseteq H = \mathrm{Log}\, K^0_{\mathbb{R}}$ as in Lemma 7.6 (see also Lemma 7.5(iii)). Note that $\mathrm{Vol}(T^{\mathfrak{m}}) = \mathrm{Vol}(\mathrm{Log}(\mathcal{O}^\times_{K^{\mathfrak{m}},1}))$.

The expected value as in Equation (40) then equals the average number of points of a randomly shifted logarithmic ray unit lattice into this simplex, which equals $\mathrm{Vol}(S_{n\log r - \log\mathcal{N}(\mathfrak{c})})/\,\mathrm{Vol}(T^{\mathfrak{m}})$ (see Lemma 2.4). Therefore,

$$\frac{1}{|G/T^{\mathfrak{m}}|}\mathop{\mathbb{E}}_{\mathbf{a}\leftarrow\mathcal{U}(F_{T^{\mathfrak{m}}})}[|\mathrm{Exp}(\mathbf{a})_1^\times \cap r\,\mathcal{N}(\mathfrak{c})^{-1/n}\mathcal{B}_\infty|] = \frac{|\mu_{K^{\mathfrak{m}},1}|\cdot\mathrm{Vol}(S_{n\log r-\log\mathcal{N}(\mathfrak{c})})}{|G/T^{\mathfrak{m}}|\cdot|T^{\mathfrak{m}}|} \tag{41}$$

$$= \frac{[\mathrm{Pic}^0_{K^{\mathfrak{m}}} : G]\cdot|\mu_K|\cdot\mathrm{Vol}(S_{n\log r-\log\mathcal{N}(\mathfrak{c})})}{\phi(\mathfrak{m}_0)\cdot 2^{|\mathfrak{m}_{\mathbb{R}}|}\cdot h_K\cdot|T|} = \frac{[\mathrm{Pic}^0_{K^{\mathfrak{m}}} : G]\cdot|\mu_K|\cdot C(r,\mathcal{N}(\mathfrak{c}))}{\phi(\mathfrak{m}_0)\cdot 2^{|\mathfrak{m}_{\mathbb{R}}|}\cdot h_K\cdot R_K} \tag{42}$$

where, for the second equation, we use that $|G/T^{\mathfrak{m}}| \cdot |T^{\mathfrak{m}}| = |G| = |\operatorname{Pic}^0_{K^{\mathfrak{m}}}|/[\operatorname{Pic}^0_{K^{\mathfrak{m}}} : G]$ and Equation (21) of Lemma 5.1. For the third equation we use the fact that $\frac{\mathrm{Vol}(S_{n \log r - \log \mathcal{N}(\mathfrak{c})})}{|T|} = \frac{C(r,\mathcal{N}(\mathfrak{c}))}{R_K}$, where we define $C(r, \mathcal{N}(\mathfrak{c})) := (n \log r - \log \mathcal{N}(\mathfrak{c}))^{\mathbb{r}}/\mathbb{r}!$ (see Lemma 7.6 and Equation (9)).

7.2.7. *Applying the Abel summation formula to get the probability for the ideal set* $\mathcal{S}$. By combining Equations (36), (40) and (42), using the class number formula (see Equation (4)) and the fact that $\frac{\mathcal{N}(\mathfrak{m}_0)}{\phi(\mathfrak{m}_0)} = \frac{|\mathcal{O}_K/\mathfrak{m}_0|}{|(\mathcal{O}_K/\mathfrak{m}_0)^\times|} \geq 1$, one obtains,

$$\mathop{\mathbb{E}}_{\mathbf{a} \leftarrow \mathcal{D}} [p_{\mathbf{a},\mathfrak{c}}] \geq e^{-1/4} \cdot \frac{\sqrt{|\Delta_K|} \cdot \mathcal{N}(\mathfrak{m}_0) \cdot 2^{|\mathfrak{m}_{\mathbb{R}}|}}{r^n \cdot 2^{n_{\mathbb{R}}} \cdot (2\pi)^{n_{\mathbb{C}}}} \cdot \frac{[\operatorname{Pic}^0_{K^{\mathfrak{m}}} : G] \cdot |\mu_K| \cdot C(r, \mathcal{N}(\mathfrak{c}))}{\phi(\mathfrak{m}_0) \cdot 2^{|\mathfrak{m}_{\mathbb{R}}|} \cdot h_K \cdot R_K} \tag{43}$$

$$= e^{-1/4} \cdot \frac{[\operatorname{Pic}^0_{K^{\mathfrak{m}}} : G] \cdot C(r, \mathcal{N}(\mathfrak{c}))}{r^n \cdot \rho_K} \cdot \frac{\mathcal{N}(\mathfrak{m}_0)}{\phi(\mathfrak{m}_0)} \tag{44}$$

where $|\mathfrak{m}_{\mathbb{R}}|$ is the number of real places dividing $\mathfrak{m}_\infty$ and where $C(r, \mathcal{N}(\mathfrak{c})) = (n \log r - \log \mathcal{N}(\mathfrak{c}))^{\mathbb{r}}/\mathbb{r}!$. By taking the sum over all $\mathfrak{c} \in \mathcal{S}$ (note that $[d^0(\mathcal{S})] \subseteq G$), using linearity of the expected value operator, one can achieve the following lower bound.

$$\mathop{\mathbb{E}}_{\mathbf{a} \leftarrow \mathcal{D}} \left[\sum_{\mathfrak{c} \in \mathcal{S}} p_{\mathbf{a},\mathfrak{c}} \right] = \sum_{\mathfrak{c} \in \mathcal{S}} \mathop{\mathbb{E}}_{\mathbf{a} \leftarrow \mathcal{D}} [p_{\mathbf{a},\mathfrak{c}}] \tag{45}$$

$$\in [e^{-1/4}, e^{1/4}] \cdot \frac{\mathcal{N}(\mathfrak{m}_0)}{\phi(\mathfrak{m}_0)} \cdot [\operatorname{Pic}^0_{K^{\mathfrak{m}}} : G] \cdot \sum_{\mathfrak{c} \in \mathcal{S}} \frac{C(r, \mathcal{N}(\mathfrak{c}))}{\rho_K \cdot r^n} \tag{46}$$

By an application of the Abel summation formula, one can relate the sum $\sum_{\mathfrak{c} \in \mathcal{S}} C(r, \mathcal{N}(\mathfrak{c}))$ to an integral involving the counting function $|\mathcal{S}(t)| = |\{\mathfrak{c} \in \mathcal{S} \mid \mathcal{N}(\mathfrak{a}) \leq t\}|$ of the ideal set $\mathcal{S}$ and the derivative of the volume function $C(r, N) = (n \log r - \log N)^{\mathbb{r}}/\mathbb{r}!$ with respect to the variable N. More precisely, we have

$$\sum_{\mathfrak{c} \in \mathcal{S}} \frac{C(r, \mathcal{N}(\mathfrak{c}))}{\rho_K \cdot r^n} = -\int_{t=1}^{r^n} \frac{|\mathcal{S}(t)|}{\rho_K \cdot r^n} \cdot \left[\tfrac{d}{dN} C(r, N) \Big|_{N=t} \right] dt \tag{47}$$

$$= \frac{1}{(\mathbb{r}-1)!} \int_{t=1}^{r^n} \frac{|\mathcal{S}(t)|}{\rho_K \cdot t} \cdot \left[\tfrac{d}{dN} \Gamma\big(\mathbb{r}, n \log r - \log N\big) \Big|_{N=t} \right] dt, \tag{48}$$

Where the first equality is the Abel summation formula [2, Theorem 4.2] and the second equality follows from computing the derivative of the upper incomplete Gamma function $\Gamma(\mathbb{r}, x) = \int_x^\infty u^{\mathbb{r}-1} e^{-u} du$.

$$-\tfrac{d}{dN} C(r, N)\Big|_t = \frac{(n \log r - \log t)^{\mathbb{r}-1}}{t \cdot (\mathbb{r}-1)!} = \frac{r^n}{t \cdot (\mathbb{r}-1)!} \cdot \big[\tfrac{d}{dN} \Gamma\big(\mathbb{r}, n \log r - \log N\big) \big]\Big|_t.$$

Using Definition 2.11 about ideal density and the fact that the integrand is positive, Equation (48) is lower bounded by

$$\frac{1}{(\mathbb{r}-1)!} \int_{t=(r/e)^n}^{r^n} \frac{|\mathcal{S}(t)|}{\rho_K \cdot t} \cdot \left[\tfrac{d}{dN} \Gamma\big(\mathbb{r}, n \log r - \log N\big) \Big|_{N=t} \right] dt \tag{49}$$

$$\geq \frac{\delta_{\mathcal{S}}[r^n]}{(\mathbb{r}-1)!} \int_{t=(r/e)^n}^{r^n} \left[\tfrac{d}{dN} \Gamma\big(\mathbb{r}, n \log r - \log N\big) \Big|_{N=t} \right] dt \geq \tfrac{1}{2} \cdot \delta_{\mathcal{S}}[r^n], \tag{50}$$

where the last inequality (Equation (50)) follows from the definition of the upper incomplete Gamma function,

$$\frac{1}{(\mathbb{r}-1)!}\int_{t=(r/e)^n}^{r^n}\left(\frac{d}{dt}\Gamma(\mathbb{r}, n\log r-\log N)\big|_{N=t}\right)dt = \frac{1}{(\mathbb{r}-1)!}\cdot(\Gamma(\mathbb{r},0)-\Gamma(\mathbb{r},n))$$
$$= 1-e^{-n}\sum_{k=0}^{\mathbb{r}-1}\frac{n^k}{k!}\geq 1/2,$$

where we used the fact that $e^{-n}\sum_{k=0}^{\mathbb{r}-1}\frac{n^k}{k!}$ equals the probability that a Poisson distribution with parameter n yields at most $\mathbb{r}-1\leq n-1$ occurrences, which is bounded by a half.

By combining Equations (46), (48) and (50) and using $e^{-1/4}/2>1/3$, we obtain

$$\mathop{\mathbb{E}}_{\mathbf{a}\leftarrow\mathcal{D}}\left[\Pr_{\alpha\leftarrow\operatorname{Exp}(\mathbf{a})\cap r\mathcal{B}_\infty}\left[(\alpha)\cdot\operatorname{Exp}(-\mathbf{a})\in\mathcal{S}\;\middle|\;\alpha\cdot\operatorname{Exp}_\infty(-\mathbf{a}_\infty)\in\tau K^{\mathfrak{m},1}\right]\right]$$
$$\geq\frac{\mathcal{N}(\mathfrak{m}_0)}{\phi(\mathfrak{m}_0)}\cdot[\operatorname{Pic}^0_{K^{\mathfrak{m}}}:G]\cdot\delta_{\mathcal{S}}[r^n]/3.$$

which finishes the proof. □

8. Uniform sampling in a box intersected with an ideal lattice

8.1. Introduction. In this section, we explain how one can efficiently sample in a (shifted) ideal lattice $x(\mathfrak{b}+\gamma)$ intersected with a (partially positive) box, provided that the dimensions of the box are sufficiently large. More precisely, let $r>0$ and let $r\mathcal{B}_\infty$ denote the box

$$r\mathcal{B}_\infty := \{(x_\sigma)_\sigma\in K_{\mathbb{R}}\,|\,|x_\sigma|\leq r,\ \forall\sigma\}.$$

Now, the box of interest is $r\mathcal{B}_\infty\cap\tau K_{\mathbb{R}}^{\mathfrak{m}_\infty}$, whose coordinates at σ have the same sign as $(\sigma(\tau))_\sigma$ for the real embeddings $\sigma\mid\mathfrak{m}_\infty$. Our aim of this section is to perfectly uniform sample from the intersection $r\mathcal{B}_\infty\cap\tau K_{\mathbb{R}}^{\mathfrak{m}_\infty}\cap x(\mathfrak{b}+\gamma)$.

The algorithm we use to sample uniformly in this box follows the framework described by Plançon and Prest in [64], and is very similar to their instantiation of that framework [64, Sec. 4.2]. The main two differences are that [64] analyzes the running time of their algorithm only heuristically, whereas here we provide a provable running time; also, [64] assumes perfect uniform samples from real intervals, whereas here we only rely on perfectly uniform bits.

Technical idea of the algorithm. The core idea of the algorithm applies to general lattices and relies on two observations. The first observation: for lattices Λ satisfying $\Lambda\subseteq\frac{1}{N}\mathbb{Z}^n$ and sufficiently large, bounded convex sets $S\subseteq\mathbb{R}^n$, the task of uniformly sampling from $S\cap\Lambda$ reduces to uniformly sampling in $(1+c)S\cap\frac{1}{N}\mathbb{Z}^n$ for some constant $c>0$, where we understand $(1+c)\cdot S:=\{(1+c)\cdot s\mid s\in S\}$. Indeed, we can use a 'good' fundamental domain F of Λ to round a uniform sample of $(1+c)\cdot S\cap\frac{1}{N}\mathbb{Z}^n$ to a point in Λ. Since $(F+\ell)\cap\frac{1}{N}\mathbb{Z}^n$ contains the same number of elements for all $\ell\in\Lambda\cap S$ (since $\Lambda\subseteq\frac{1}{N}\mathbb{Z}^n$), this yields a perfect uniform distribution over $S\cap\Lambda$, if $c>0$ is chosen adequately. This observation is formalized in Lemma 8.7.

The second observation is that each lattice Λ given by a basis B can be *approximated* by a lattice $\tilde{\Lambda}$ given by a basis $\tilde{B}$ that satisfies $\tilde{\Lambda}\subseteq\frac{1}{N}\mathbb{Z}^n$. Hence, by the first observation, one can perfectly uniformly sample from $\tilde{\Lambda}\cap S$ for suitable S. If the approximation $\tilde{B} \approx B$ is good enough, one can transform this perfectly uniform sample from $\tilde{\Lambda} \cap S$ to a perfectly uniform sample from $\Lambda \cap S$. This observation is formalized in Proposition 8.8.

Note that in the first observation, a 'good' fundamental domain of the lattice is required. For such a fundamental domain, we need to know a sufficiently good basis of the ideal we want to sample from. We obtain such a basis by using one of the provable variants of the BKZ algorithm [68, 39, 33, 1]. Lemma 8.5 captures the end result of using a provable variant of the BKZ algorithm; for an in-depth proof of this lemma, we refer to Sections 26 to 28.

Remark 8.1. *We note that this choice of basis reduction algorithm (BKZ) does not exploit the ideal structure of the lattice. There do exist algorithms that exploit the ideal structure* [29, 63] *and might outperform BKZ in some context. However, these algorithms are heuristic, require a quantum computer and/or some exponential preprocessing on the number field and in the case of* [29] *are restricted to cyclotomic number fields. Hence, we chose to use the BKZ algorithm here in order to obtain a non-heuristic algorithm. If one allows heuristics and quantum algorithms, it may be possible to diminish the block-size dependent quantity* b^{b} *that appears in the running time of the basis reduction algorithm.*

Ray. Actually, in this section, a slightly more general algorithm is described. In this more general algorithm, we sample elements from an intersection of a (shifted) ideal lattice $x(\mathfrak{b} + \gamma)$ and a box $r\mathcal{B}_\infty$, that also fall in the τ-*ray* mod $\mathfrak{m}$, for some $\tau \in K^{\mathfrak{m}}$.

That is, we will sample from $x((\mathfrak{b} + \gamma) \cap \tau K^{\mathfrak{m},1})$ intersected with $r\mathcal{B}_\infty$; those are the elements of the shape $x(\beta + \gamma)$ for which $\beta + \gamma \in \tau K^{\mathfrak{m},1}$ and $|x_\sigma \sigma(\beta + \gamma)| \leq r$ for all embeddings σ. So, for the algorithm including the modulus $\mathfrak{m}$, the output additionally satisfies certain modular conditions depending on the modulus $\mathfrak{m}$.

One retrieves the simpler algorithm (without a modulus) that samples in $x(\mathfrak{b} + \gamma) \cap r\mathcal{B}_\infty$ by just putting $\mathfrak{m} = \mathcal{O}_K$ (so that $\tau K^{\mathfrak{m},1} = K^{\mathfrak{m},1} = K^{\mathfrak{m}} = K^*$ for all $\tau \in K^*$).

8.2. **On efficiently deciding whether an algebraic number is greater or equal to some rational number.** To be able to perfectly sample from an ideal lattice intersected with a box, one needs to be able to efficiently decide, for an algebraic number $\alpha \in K$ in symbolic representation, whether $|\sigma(\alpha)| > r$ or $|\sigma(\alpha)| \leq r$ (or, for real embeddings, whether $\sigma(\alpha) > r$ or $\leq r$) for some rational $r \in \mathbb{Q}$ and embedding σ. Namely, the combined information over all places signifies whether the algebraic number α is inside some box $r\mathcal{B}_\infty$ or not.

The challenge in this question is that the algebraic number is given in some symbolic representation, like $\alpha = 408\sqrt{2} - 577(\approx -0.0008665)$. In order to decide, for example, whether $\alpha > 0$ or ≤ 0, we need to know *how well* to approximate this number (in this particular case at least up to 3 decimals).

The result of this section, in a nutshell, is that for any algebraic number $\alpha \in K$ we can efficiently decide whether $|\sigma(\alpha)|$ (or $\sigma(\alpha)$ in the case of a real place) is larger than some rational $r \in \mathbb{Q}$ or not. This is proven in the following sequence of lemmas. The next lemma is coined Liouville's lemma, since the technique comes from the tendency of algebraic numbers to avoid rationals, which was originally used to show the explicit existence of transcendental numbers (Liouville numbers).

Lemma 8.2 (Louiville's lemma)**.** *Let $f(x) \in \mathbb{Z}[x]$ be a degree n polynomial. Let $\omega \in \mathbb{R}$ be a real root of f. Then, if $\omega \notin \mathbb{Z}$, we have, for all $z \in \mathbb{Z}$,*

$$|\omega - z| > \left[(n+1)^3 \cdot 2^n \cdot (2 + \|f\|_\infty)^n\right]^{-1} =: \mathfrak{A}(f).$$

Proof. Let $h \mid f$ be an irreducible factor of f and let ω be a real root of $h(x) \in \mathbb{Z}[x]$. If h is linear, $\omega = \frac{c}{d} \in \mathbb{Q}$, and by the rational root theorem, d is an integer factor of the leading coefficient of h (and thus of that of f). Hence, if $\omega \notin \mathbb{Z}$, we have $|\omega - z| \geq 1/d \geq 1/\|f\|_\infty > \mathfrak{A}(f)$ for every $z \in \mathbb{Z}$.

Assume now that h is non-linear. By the mean value theorem, we have, for any $t \in [\omega - 1, \omega + 1]$,

$$|h(t)| = |h(t) - h(\omega)| \leq \max_{t_0 \in [\omega-1,\omega+1]} |h'(t_0)| \cdot |t - \omega| \tag{51}$$

Additionally, for any *integer* $z \in \mathbb{Z}$, we have $|h(z)| \geq 1$, since $h(x) \in \mathbb{Z}[x]$ is irreducible and of degree > 1 (and hence cannot have integer roots). This immediately yields, for any *integer* $z \in \mathbb{Z}$,

$$|z - \omega| \geq \frac{|h(z)|}{\max_{x_0 \in [\omega-1,\omega+1]} |h'(x_0)|} \geq \Big(\max_{x_0 \in [\omega-1,\omega+1]} |h'(x_0)|\Big)^{-1} \tag{52}$$

$$\geq \Big(\max_{h|f} \max_{\omega \text{ root of } h} \max_{x_0 \in [\omega-1,\omega+1]} |h'(x_0)|\Big)^{-1} \tag{53}$$

where, in the last line, the first maximum is over all divisors of f (also the possible linear ones) and the second maximum is over all real roots. We now aim to find a easy-to-compute lower bound on Equation (53), which we call $\mathfrak{A}(f)$, in terms of the coefficients of f alone, using the Landau-Mignotte bound [76, Theorem 6.32] and Cauchy's bound [27, Section 3.6].

By the triangle inequality, writing $h(x) = \sum_{j=0}^{\deg(h)} h_j \cdot x^j$, $\|h\|_\infty = \max_j |h_j|$ and $\|f\|_\infty = \max_j |f_j|$, we can deduce

$$\begin{aligned}
\max_{x_0 \in [\omega-1,\omega+1]} |h'(x_0)| &\leq \sum_{j=1}^{\deg(h)} j \cdot |h_j| \cdot \max_{x_0 \in [\omega-1,\omega+1]} |x_0|^{j-1} \\
&\leq \deg(h)^2 \cdot \|h\|_\infty \cdot \max_{x_0 \in [\omega-1,\omega+1]} |x_0|^{\deg(h)-1} \\
&\leq \deg(h)^2 \cdot \|h\|_\infty \cdot (|\omega| + 1)^{\deg(h)-1} \\
&\leq n^2 \cdot (n+1) \cdot 2^n \cdot \|f\|_\infty \cdot (|\omega| + 1)^{n-1} \quad \text{(Landau-Mignotte bound)} \\
&\leq n^2 \cdot (n+1) \cdot 2^n \cdot \|f\|_\infty \cdot (2 + \|f\|_\infty)^{n-1} \quad \text{(Cauchy's bound)} \\
&< (n+1)^3 \cdot 2^n \cdot (2 + \|f\|_\infty)^n = \mathfrak{A}(f)^{-1}.
\end{aligned}$$

where we recall that $n = \deg(f) \geq \deg(h)$. Thus, for any $z \in \mathbb{Z}$ and any real root ω of f satisfying $\omega \notin \mathbb{Z}$, we have

$$|\omega - x| \geq \mathfrak{A}(f) = \left[(n+1)^3 \cdot 2^n \cdot (2 + \|f\|_\infty)^n\right]^{-1}. \tag{54}$$

□

Corollary 8.3. *There exists an algorithm that, given as input a degree n polynomial $f(x) \in \mathbb{Z}[x]$ and some rational $r \in \mathbb{Q}$, decides whether the roots $\omega_i \in \mathbb{C}$ of $f(x)$ satisfy $|\omega_i| > r$ or $|\omega_i| \leq r$ for complex roots ω_i, respectively, $\omega_i > r$ or $\omega_i \leq r$ for* *real roots ω_i. The running time of the algorithm is polynomial in the size of $f(x)$ and r.*

Proof. Denote $r = \frac{a}{b} \in \mathbb{Q}$. We replace the roots ω_i by $b \cdot \omega_i$, the polynomial $f(x)$ by $b^n \cdot f(b^{-1} \cdot x)$ and the rational r by $a \in \mathbb{Z}$. Hence, we may without loss of generality assume that $r \in \mathbb{Z}$, with the expense of an increase in the size of the polynomial f by $n \cdot \mathrm{size}(r)$.

We start with the case of a real root ω of f, and we then proceed with the case of a complex root.

Real root case.

According to Lemma 8.2, we have for any real but non-integral root ω of $f(x)$ and any $z \in \mathbb{Z}$, $|\omega - z| > \mathfrak{A}(f)$. This lower bound gives us means to decide whether a real root of f is larger (or equal) or smaller than some integer. Namely, approximate the real roots of f within precision $< \mathfrak{A}(f)/2 = \left[(n+1)^3 \cdot (2 + \|f\|_\infty)^n\right]^{-1}/2$, which can be done in time $\mathsf{poly}(n, \log \|f\|_\infty) = \mathsf{poly}(n, \mathrm{size}(f), \mathrm{size}(r))$. Denote the approximations of these real roots by $\tilde{\omega_i} \approx \omega_i$. There are two cases:

- $|\tilde{\omega_i} - z| < \mathfrak{A}(f)/2$ for some integer $z \in \mathbb{Z}$. In that case, $\omega_i = z$. Indeed, if ω_i were not an integer, we must have
$$\mathfrak{A}(f) \leq |\omega_i - z| \leq |\omega_i - \tilde{\omega_i}| + |\tilde{\omega_i} - z| < \mathfrak{A}(f),$$
which is a contradiction. Hence, ω_i is an integer and equal to $z \in \mathbb{Z}$. In such case, deciding whether $|\omega_i| > r$ or $\leq r$ is easily and efficiently done.
- $|\tilde{\omega_i} - z| \geq \mathfrak{A}(f)/2$ for all $z \in \mathbb{Z}$. Then, by the triangle inequality, we have $\omega_i \notin \mathbb{Z}$, since, for all $z \in \mathbb{Z}$,
$$|\omega_i - z| \geq \underbrace{|\tilde{\omega_i} - z|}_{\geq \mathfrak{A}(f)/2} - \underbrace{|\omega_i - \tilde{\omega_i}|}_{< \mathfrak{A}(f)/2} > 0.$$
Hence, by Lemma 8.2, $|\omega_i - z| \geq \mathfrak{A}(f)$ for any integer $z \in \mathbb{Z}$. We claim that $\tilde{\omega_i} > r$ if and only if $\omega_i > r$. That is, by using the approximation $\tilde{\omega_i}$, we can decide whether $\omega_i > r$ or $\leq r$.

  Indeed, suppose (as to achieve a contradiction) that $\omega_i \leq r$ but $\tilde{\omega_i} > r$. Then, since $|\omega_i - r| \geq \mathfrak{A}(f)$, and thus $r - \omega_i \geq \mathfrak{A}(f)$.
$$\underbrace{r - \omega_i}_{\geq \tilde{\mathfrak{A}}(f)} = \underbrace{(r - \tilde{\omega}_i)}_{\leq 0} + \underbrace{(\omega_i - \tilde{\omega}_i)}_{\in[-\mathfrak{A}(f)/2, \mathfrak{A}(f)/2]} \leq \mathfrak{A}(f)/2$$
which is a contradiction. The other case ($\omega_i > r$ but $\tilde{\omega_i} \leq r$) can be excluded similarly.

Complex root case.

For the complex root case, we take the companion matrix M_f of $f = \sum_{j=0}^{n} f_j x^j$. This $n \times n$ matrix M_f is defined by $(M_f)_{j+1,j} = 1$ for $j \in \{1, \ldots, n-1\}$ and $(M_f)_{j,n} = -f_{j-1}/f_n$ for $j \in \{1, \ldots, n\}$ (this is the companion matrix of the monic polynomial $f_n^{-1} \cdot f(x)$). It is known that M_f has the roots ω_i of f as eigen values. Therefore, the matrix resulting from the Kronecker product $M_f \otimes M_f$ has all products $\omega_i \omega_j$ ($i, j \in \{1, \ldots, n\}$) as eigen values, in particular the square absolute values $|\omega_i|^2$ for all original roots of f.

Computing the characteristic polynomial of $M_f \otimes M_f$, and clearing denominators of that polynomials yields the n^2-degree integer polynomial $g(x) \in \mathbb{Z}[x]$. Identifying the roots of g that are the square absolute values of those of f (by approximating the roots of both), one can apply the same technique as in the real root case to decide whether $|\omega_i|^2 > r^2$ or not.

It remains to be shown that the polynomial g is polynomially bounded in size by the size of f, so that all operations can be done in polynomial time in size(f) and size(r). The j-th coefficient p_j of the characteristic polynomial $p(x)$ of $M_f \otimes M_f$ can be computed (up to sign) by taking the sum of all $\binom{n^2}{j}$ principal minors of $M_f \otimes M_f$ of size j [42, §1.2, Equation (1.2.13)]. By Hadamard's bound, each of these principal minors has size at most $\|f\|_\infty^{2j}$ and can only have a denominator that is a divisor of f_n^{2j}. Hence, $f_n^{2n} \cdot p(x)$ must be a multiple of $g(x)$. This means that $\|g\|_\infty \leq \|p\|_\infty \leq (n^2)!\|f\|_\infty^{2n}$. Hence, $\text{size}(g) \leq \mathsf{poly}(n, \text{size}(f))$, which was to be demonstrated. □

Corollary 8.4. *There exist an algorithm that on input a degree n number field $K = \mathbb{Q}(\theta)$, an ideal $\mathfrak{b}$, an element $x = (x_\sigma)_\sigma \in K_\mathbb{R}^\times$ represented by rational numbers, an element $\alpha \in \mathfrak{b}$ represented by the $\mathbb{Q}$-basis of K (the power-basis formed by θ), a rational $g \in \mathbb{Q}$ and an integer $k \in \mathbb{Z}_{>0}$, decides whether $x_\sigma|\sigma(\alpha)| > g^{1/k}$ or $x_\sigma|\sigma(\alpha)| \leq g^{1/k}$, for every complex embedding σ; and decides whether $x_\sigma\sigma(\alpha) > g^{1/k}$ or $x_\sigma\sigma(\alpha) \leq g^{1/k}$, for every real embedding σ. Moreover, this algorithm runs in time $\mathsf{poly}(\text{size}(\alpha), \text{size}(x), \text{size}(\mathfrak{b}), \text{size}(g), n, \log|\Delta_K|, k)$.*

Proof. Throughout this paper we assume that the polynomial f defining the number field K satisfies $\text{size}(f) \leq \mathsf{poly}(\log|\Delta_K|)$ (see Section 2.6). As $x_\sigma|\sigma(\alpha)| > g^{1/k}$ is equivalent to $|\sigma(\alpha^k)| > g/x_\sigma^k$ (with a possible sign change), we can reduce this result to that of Corollary 8.3.

The sole challenge is to find and bound the polynomial of which α^k is a root. For this we start by bounding the polynomial for which $\alpha \in \mathfrak{b}$ is a root. Write down the multiplication matrix of α in the θ-basis. That is, the matrix defined by the coefficients q_{ij} occurring in the identities $\alpha\theta^j = \sum_{i=0}^{n-1} q_{ij}\theta^i$ for $j \in \{0, \ldots, n-1\}$. Then the characteristic polynomial of this matrix is one of which $\alpha \in \mathfrak{b}$ must be a root.

The size of the multiplication matrix of α is bounded by $\mathsf{poly}(\text{size}(\alpha), \text{size}(f)) = \mathsf{poly}(\text{size}(\alpha), \log|\Delta_K|)$. Hence, the size of the characteristic polynomial (whose j-th coefficient can be shown to be the sum of all principal minors of size j of the multiplication matrix, which can be bounded by Hadamard's bound) must be polynomially bounded in size(α), $\log|\Delta_K|$ and n as well. By a very similar reasoning, the size of the characteristic polynomial of α^k must be $\mathsf{poly}(n, \log|\Delta_K|, \text{size}(\alpha), k)$. By applying Corollary 8.3, we obtain the final result. □

8.3. Lattice reduction.

Being able to successfully sample in an ideal lattice intersected with a box largely depends on the maximum vector length (i.e., the quality) of the basis of the ideal lattice compared to the dimensions of the box. If these dimensions of the box are somewhat larger than the basis vectors, this sampling can be done efficiently. For the applications and algorithms of the present work, we generally require the dimensions of this box to be as small as possible, since this benefits their complexity. As a consequence, we would like to obtain a basis of the ideal lattice with as small as possible maximum vector length (i.e., best basis quality), but without paying too much time.

The standard algorithm for finding lattice bases of good quality is the Block Korkine Zolotarev (BKZ) algorithm, which allows for a controllable trade-off between the output basis quality and the heuristic run time. Unfortunately, no published versions of the BKZ-algorithm exist that have a *provable* run-time, which is what we require in the present work (as our end goal is an algorithm with a provable run-time). Hence, Sections 26 to 28 of this paper is devoted to showing that there exists a variant of the BKZ-algorithm with the same trade-off between output basis quality and run time as the textbook version, but in which the run time is actually proven and non-heuristic.

The proof of this result requires both careful numerical analysis (for the ideal lattice case) and precise monitoring of the sizes ('potential') of the bases occurring intermediately in the algorithm. To stay relevant with the current topic, this proof is therefore postponed to Part 3. The overall result can be summarized in the following lemma.

Lemma 8.5. *Let K be a number field of degree n. Let $x\mathfrak{a}$ be an ideal lattice where $x \in K_{\mathbb{R}}^{\times}$ is represented by rational numbers, and where $\mathfrak{a} \in \mathcal{I}_K$ is represented by a rational Hermite Normal Form matrix $M_{\mathfrak{a}}$ with respect to a given LLL-reduced basis of $\mathcal{O}_K$.*

Then there exists an algorithm that computes a $\mathbb{Z}$-basis $(x\alpha_1, \ldots, x\alpha_n)$ of $x\mathfrak{a}$ with $\alpha_i \in K$ such that

$$\|x \cdot \alpha_i\| \leq 2n \cdot \mathsf{b}^{2n/\mathsf{b}} \cdot \lambda_n(x \cdot \mathfrak{a}),$$

using time at most $T = \mathit{poly}(\mathsf{b}^{\mathsf{b}}, \mathrm{size}(\mathfrak{a}), \log|\Delta_K|, \mathrm{size}(x))$.

Proof. The proof can be found in Section 28.6. □

8.4. Perfectly uniform sampling in a bounded convex set intersected with a lattice.

Notation 8.6. *We denote $B_r = \{x \in \mathbb{R}^n \mid \|x\| < r\}$. For any bounded convex set $S \subseteq \mathbb{R}^n$ and $c \in \mathbb{R}_{>0}$, we denote $cS := \{c \cdot s \mid s \in S\}$. For a basis $\mathbf{B} = (\mathbf{b}_1, \ldots, \mathbf{b}_n)$ of $\mathbb{R}^n$, we denote $\mathcal{L}(\mathbf{B}) = \mathbf{B}\mathbb{Z}^n := \{\sum_{i=1}^n \mathbf{b}_i z_i \mid z_i \in \mathbb{Z}$ for all $i\}$ (which is a lattice).*

Lemma 8.7. *Let $\mathbf{B} = (\mathbf{b}_1, \ldots, \mathbf{b}_n)$ with $\mathbf{b}_i \in \frac{1}{N}\mathbb{Z}^n$ be a basis, let $D = \sum_i \|\mathbf{b}_i\|$, let $U, c, \varepsilon \in \mathbb{R}_{>0}$ such that $c > \varepsilon > 0$. Suppose S is a convex set satisfying $B_{D/\varepsilon} \subseteq S \subseteq B_U$ for which we can perfectly uniformly sample in $(c+\varepsilon)S \cap \frac{1}{N}\mathbb{Z}^n$ within expected time T. Additionally, suppose we can efficiently decide membership in cS.*

Then we can perfectly uniformly sample in $cS \cap \mathbf{B}\mathbb{Z}^n$ within time $\mathit{poly}(T, \mathrm{size}(\mathbf{B}), \log U, \log N)$ with success probability lower bounded by

$$\frac{|(c-\varepsilon)S \cap \frac{1}{N}\mathbb{Z}^n|}{|(c+\varepsilon)S \cap \frac{1}{N}\mathbb{Z}^n|} \tag{55}$$

Proof. (**Algorithm description**) Sample a perfectly uniform $u \in (c+\varepsilon)S \cap \frac{1}{N}\mathbb{Z}^n$. Compute $v = \mathbf{B}^{-1}u$ (which can be done in time $\mathsf{poly}(T, \mathrm{size}(\mathbf{B}), \log U)$) and put $w_i = \lfloor v_i \rceil$, rounding to the nearest integer. Output $\mathbf{B}w$ if $\mathbf{B}w \in cS$, otherwise output 'failure'.

(**Analysis**) We will prove that this procedure yields a perfectly uniform sample in $\mathbf{B}\mathbb{Z}^n \cap cS$ and that the success probability of a single iteration is lower bounded by the quantity in Equation (55).

For a fixed $\mathbf{b} \in (\mathbf{B}\mathbb{Z}^n) \cap cS$, the probability that $\mathbf{b}$ is outputted is proportional to the number $|(\mathbf{b} + \mathbf{B} \cdot [-\frac{1}{2}, \frac{1}{2})^n) \cap (c+\varepsilon)S \cap \frac{1}{N}\mathbb{Z}^n|$. We have $\mathbf{b} \in cS$; and each $\mathbf{x} \in \mathbf{B} \cdot [-\frac{1}{2}, \frac{1}{2})^n$ satisfies $\|\mathbf{x}\| \leq D/2 \leq D$, hence $\mathbf{x} \in B_D \subseteq \varepsilon \cdot S$ (since $B_{D/\varepsilon} \subseteq S$).

So, we obtain that $\mathbf{b} + \mathbf{B} \cdot [-\frac{1}{2}, \frac{1}{2})^n \subseteq (c+\varepsilon)S$ by the convexity of S. Therefore

$$|(\mathbf{b} + \mathbf{B} \cdot [-\tfrac{1}{2}, \tfrac{1}{2})^n) \cap (c+\varepsilon)S \cap \frac{1}{N}\mathbb{Z}^n| = |(\mathbf{b} + \mathbf{B} \cdot [-\tfrac{1}{2}, \tfrac{1}{2})^n) \cap \frac{1}{N}\mathbb{Z}^n|$$
$$= N^n \det(\mathbf{B}),$$

for all $\mathbf{b} \in (\mathbf{B}\mathbb{Z}^n) \cap S$. Note that this relies critically on $\frac{1}{N}\mathbb{Z}^n \subseteq \mathbf{B}\mathbb{Z}^n$. So the sampling probability is the same for all $\mathbf{b} \in (\mathbf{B}\mathbb{Z}^n) \cap cS$, which is the property of a perfect uniform distribution. It remains to show that the probability of success (a sample in $\mathbf{B}\mathbb{Z}^n \cap cS$) is lower bounded.

If the initial uniform sample u satisfies $u \in (c-\varepsilon)S \cap \frac{1}{N}\mathbb{Z}^n$, then $\mathbf{B}w = \mathbf{B}\lfloor \mathbf{B}^{-1}u \rceil = u + y$ for some $y \in \mathbf{B}[-\frac{1}{2}, \frac{1}{2})^n \subseteq \varepsilon S$ (where we mean with $\lfloor \cdot \rceil$ that every coordinate is rounded to the nearest integer). For such u holds that $u+y$ must lie in cS. Therefore, the success probability is lower bounded by $\frac{|(c-\varepsilon)S \cap \frac{1}{N}\mathbb{Z}^n|}{|(c+\varepsilon)S \cap \frac{1}{N}\mathbb{Z}^n|}$. □

In our context, the basis $\mathbf{B}_\mathfrak{b}$ of an ideal lattice $\mathfrak{b}$ (as embedded in the Minkowski space) cannot be represented by rational numbers, but is rather symbolically represented. In the following proposition we show that this is not a real complication; namely, it is enough to be able to compute a sufficiently good rational approximation $\mathbf{C} \approx \mathbf{B}_\mathfrak{b}$ in order to be able to perfectly uniformly sample in this lattice $\mathfrak{b}$ intersected with a certain convex set S.

As a consequence, the algorithm of the following proposition takes as an input a sufficiently good approximation $\mathbf{C}$ of $\mathbf{B}$ and uses only this rational matrix $\mathbf{C}$ to compute with. To be able to check whether an element of the lattice $\mathbf{B}\mathbb{Z}^n$ lies in the convex set S or not, an auxiliary oracle is needed. In our use-case, this oracle can be implemented by using the results of Section 8.2.

Proposition 8.8. *Let $\mathbf{B} = (\mathbf{b}_1, \ldots, \mathbf{b}_n)$ with $\mathbf{b}_i \in \mathbb{R}^n$ be a basis of lattice Λ and let $D = 2\sum_i \|\mathbf{b}_i\|$.*

(1) Let $t \in \mathbb{R}^n$ satisfying $\|t\| < U$ for some $U \in \mathbb{R}_{>0}$.
(2) Let $N \in \mathbb{Z}_{>0}$ satisfying $\frac{n}{N} \leq D$ and let $\mathbf{C} \in \frac{1}{N}\mathbb{Z}^n$ be an approximation of $\mathbf{B}$ satisfying $\|\mathbf{C} - \mathbf{B}\| \leq \|\mathbf{B}^{-1}\|^{-1} \cdot D/U$ and $\sum_i \|\mathbf{c}_i\| \leq D$.
(3) Let $\tilde{t} \in \frac{1}{N}\mathbb{Z}^n$ an approximation of $t \in \mathbb{R}^n$ satisfying $\|t - \tilde{t}\|_\infty \leq \frac{1}{2N}$ and $\|\tilde{t}\| < 2U$.
(4) Let $1/5 > \varepsilon > 0$ and let $S \subseteq \mathbb{R}^n$ be a convex set satisfying $B_{D/\varepsilon} \subseteq S \subseteq B_U$.
(5) Suppose we can perfectly uniformly sample in $(1+4\varepsilon)S \cap \frac{1}{N}\mathbb{Z}^n$ within expected time T.
(6) Additionally, suppose we have an oracle $O_\mathbf{B}$ that can decide, on input $v \in \mathbb{Z}^n$, whether $\mathbf{B}v \in S + t$ or not.

Then we can sample $v \in \mathbb{Z}^n$ such that $\mathbf{B}v$ is perfectly uniformly distributed in $(S+t) \cap \mathbf{B}\mathbb{Z}^n$, within expected time $\mathsf{poly}(T, \mathrm{size}(\mathbf{C}), \log U, \log N)$, using a single call to $O_\mathbf{B}$, and with success probability lower bounded by

$$\frac{|(1-5\varepsilon)S \cap \frac{1}{N}\mathbb{Z}^n|}{|(1+5\varepsilon)S \cap \frac{1}{N}\mathbb{Z}^n|}$$

Proof. (**Algorithm description**)
(1) We start with processing the shift $t \in \mathbb{R}^n$. We use the approximation $\tilde{t} \in \frac{1}{N}\mathbb{Z}^n$ that satisfies $\|t - \tilde{t}\|_\infty \leq \frac{1}{2N} \leq D/(2n)$ and hence $\|t - \tilde{t}\|_2 \leq D/2$. We compute $w_0 = \lfloor \mathbf{C}^{-1}\tilde{t} \rceil$ (where $\lfloor \cdot \rceil$ means that each component is rounded to the nearest

integer).
(2) Imitate the approach in Lemma 8.7, (taking $c = 1 + 3\varepsilon$) by sampling $u \in \frac{1}{N}\mathbb{Z}^n \cap (1 + 4\varepsilon)S$ and computing $\mathbf{c} = \mathbf{C}\lfloor \mathbf{C}^{-1}u \rceil$ until $\mathbf{c} \in (1 + 3\varepsilon)S$. By the exact same reasoning as in Lemma 8.7, one then obtains a perfectly uniform sample $\mathbf{c} \in \mathbf{C} \cdot \mathbb{Z}^n \cap (1 + 3\varepsilon)S$.
(3) We compute $v = \mathbf{C}^{-1}\mathbf{c} = \lfloor \mathbf{C}^{-1}u \rceil$ and we output $v+w_0 \in \mathbb{Z}^n$ if $\mathbf{B}(v+w_0) \in S+t$ (using the oracle $O_{\mathbf{B}}$). Otherwise output 'failure'.

(**Analysis**) In this analysis we will denote $\mathbf{b} = \mathbf{B}v, \mathbf{b}_0 = \mathbf{B}w_0, \mathbf{c} = \mathbf{C}v$ and $\mathbf{c}_0 = \mathbf{C}w_0$. We will first show that the output $v + w_0 \in \mathbb{Z}^n$ has the property that $\mathbf{B}(v + w_0)$ is perfectly uniformly distributed in $\mathbf{B}\mathbb{Z}^n \cap (S + t)$. We will finish the proof with a lower bound on the success probability and a run time analysis.

By Lemma 8.7, we know that $v + w_0$ has the property that $\mathbf{C}(v + w_0) = \mathbf{c} + \mathbf{c}_0$ is perfectly uniform in $\mathbf{C}\mathbb{Z}^n \cap [(1 + 3\varepsilon)S + \mathbf{c}_0]$. Hence $\mathbf{B}(v + w_0) = \mathbf{B}\mathbf{C}^{-1}(\mathbf{c} + \mathbf{c}_0)$ is perfectly uniform in $\mathbf{B}\mathbb{Z}^n \cap [\mathbf{B}\mathbf{C}^{-1}(1 + 3\varepsilon)S + \mathbf{b}_0]$. So, since we reject those $v + w_0$ for which $\mathbf{B}(v+w_0) \notin S+t$, it is sufficient to show that $S+t \subset \mathbf{B}\mathbf{C}^{-1}(1+3\varepsilon)S+\mathbf{b}_0$. This is equivalent to $\mathbf{C}\mathbf{B}^{-1}(S + t) \subset (1 + 3\varepsilon)S + \mathbf{c}_0$ which is in turn equivalent to $\mathbf{C}\mathbf{B}^{-1}(S + t) - \mathbf{c}_0 \subset (1 + 3\varepsilon)S$. To show this, take an arbitrary $s \in S$, and put

$$\mathbf{C}\mathbf{B}^{-1}(s + t) - \mathbf{c}_0 = s + t - \mathbf{c}_0 + (\mathbf{C}\mathbf{B}^{-1} - I)s + (\mathbf{C}\mathbf{B}^{-1} - I)t. \tag{56}$$

We will now show that $\|t - \mathbf{c}_0\| \leq D$, $\|(\mathbf{C}\mathbf{B}^{-1} - I)s\| \leq D$ and $\|(\mathbf{C}\mathbf{B}^{-1} - I)t\| \leq D$, so that the right-hand side of Equation (56) lies in $S \boxplus B_{3D} \subseteq S \boxplus (3\varepsilon)S = (1+3\varepsilon)S$ (where $\boxplus$ denotes the Minkowski sum). We have $\|t - \mathbf{c}_0\| \leq \|t - \tilde{t}\| + \|\tilde{t} - \mathbf{c}_0\| \leq D/2 + D/2$ since $\|t - \tilde{t}\| \leq D/2$ by construction and $\tilde{t} - \mathbf{c}_0 = \tilde{t} - \mathbf{C}\lfloor \mathbf{C}^{-1}\tilde{t} \rceil = \tilde{t} - \mathbf{C}(\mathbf{C}^{-1}\tilde{t} + u) = \mathbf{C}u$ with $u \in [-1/2, 1/2)^n$. Hence $\|\tilde{t} - \mathbf{c}_0\| = \|\mathbf{C}u\| \leq D/2$, by assumption. Using the properties of matrix norms, we obtain $\|(\mathbf{C}\mathbf{B}^{-1} - I)\| = \|(\mathbf{C} - \mathbf{B})\mathbf{B}^{-1}\| \leq \|\mathbf{C} - \mathbf{B}\|\|\mathbf{B}^{-1}\| \leq D/U$. Combining this with the assumptions $\|s\| \leq U$ and $\|t\| \leq U$, we obtain that all the summands' norms are bounded by D.

(**Success probability**) For the success probability, note that, if $u \in \frac{1}{N}\mathbb{Z}^n$ from step (2) of the algorithm description were to be in $(1 - 5\varepsilon)S$, a similar reasoning as above shows that, for $v = \lfloor \mathbf{C}^{-1}u \rceil$, we surely have $\mathbf{c} = \mathbf{C}v \in (1 - 4\varepsilon)S$.

We have, by the Neumann series of $(I - (I - \mathbf{C}^{-1}\mathbf{B}))^{-1}$, noting that $\|I - \mathbf{C}^{-1}\mathbf{B}\| \leq D/U \leq \varepsilon < 1/5$,

$$\begin{aligned}\|\mathbf{B}\mathbf{C}^{-1} - I\| = \|(I - (I - \mathbf{C}^{-1}\mathbf{B}))^{-1} - I\| = \Big\| \sum_{j=1}^{\infty} (I - \mathbf{C}^{-1}\mathbf{B})^j \Big| \leq \sum_{j=1}^{\infty} \|I - \mathbf{C}^{-1}\mathbf{B}\|^j \\ \leq \tfrac{5}{4} D/U.\end{aligned}$$

Hence, by similar computations as above, we can bound the norms of the following summands:

$$\begin{aligned}\mathbf{B}(v + w_0) &= \mathbf{B}\mathbf{C}^{-1}(\mathbf{c} + \mathbf{c}_0) - t \\ &= \mathbf{c} + \underbrace{(\mathbf{B}\mathbf{C}^{-1} - I)\mathbf{c}}_{\leq \frac{5}{4}D} + \underbrace{(\mathbf{B}\mathbf{C}^{-1} - I)t}_{\leq \frac{5}{4}D} + \underbrace{(\mathbf{B}\mathbf{C}^{-1} - I)(\mathbf{c}_0 - t)}_{\leq \frac{5}{4}D^2/U \leq D/4} + \underbrace{\mathbf{c}_0 - t}_{\leq D}.\end{aligned}$$

So, since $\mathbf{c} = \mathbf{C}v \in (1 - 4\varepsilon)S$ and all the other summands together have norm at most $4D$ and their sum thus lies in $B_{4D} \subseteq 4\varepsilon S$, we see that $\mathbf{B}(v + w_0) \in (1 - 4\varepsilon)S \boxplus 4\varepsilon S \subseteq S$.

Therefore, the success probability of the entire procedure is lower bounded by $\frac{|(1-5\varepsilon)S\cap\frac{1}{N}\mathbb{Z}^n|}{|(1+4\varepsilon)S\cap\frac{1}{N}\mathbb{Z}^n|} \geq \frac{|(1-5\varepsilon)S\cap\frac{1}{N}\mathbb{Z}^n|}{|(1+5\varepsilon)S\cap\frac{1}{N}\mathbb{Z}^n|}$.

(**Run time analysis**) Note that the additional running time of this algorithm compared to Lemma 8.7 is caused by the (possible) extra computations on $\tilde{t} \in \frac{1}{N}\mathbb{Z}^n$ (which satisfies $\|\tilde{t}\| \leq 2U$, which can be at most $\mathsf{poly}(\mathrm{size}(\mathbf{C}), \log U, \log N)$. The dependency on n is hidden in $\mathrm{size}(\mathbf{C})$. □

Notation 8.9. *We define $\phi : \prod_\sigma \mathbb{Z} \hookrightarrow K_\mathbb{R}$, $(n_\sigma)_\sigma \mapsto (x_\sigma)_\sigma$ by putting*

$$\begin{cases} x_{\sigma_\nu} = n_{\sigma_\nu} + i n_{\bar{\sigma}_\nu} & \textit{if } \nu \textit{ is complex} \\ x_{\bar{\sigma}_\nu} = n_{\sigma_\nu} - i n_{\bar{\sigma}_\nu} & \textit{if } \nu \textit{ is complex} \\ x_{\sigma_\nu} = n_{\sigma_\nu} & \textit{if } \nu \textit{ is real} \end{cases} \tag{57}$$

Abusing notation, we will just denote $\mathbb{Z}^n$ or $\prod_\sigma \mathbb{Z}$ for $\phi(\prod_\sigma \mathbb{Z})$.

Corollary 8.10. *Let K be a degree n number field, let $x \in K_\mathbb{R}^\times$ be represented by rational numbers, let $\mathfrak{b} \in \mathcal{I}_K$ be an ideal and let $\mathfrak{m} = \mathfrak{m}_0\mathfrak{m}_\infty$ be a modulus. Let $\omega \in \mathbb{Q}_{\geq 1}$ and let $r = 48 \cdot \omega \cdot \mathsf{b}^{2n/\mathsf{b}} \cdot n^{7/2} \cdot |\Delta_K|^{\frac{3}{2n}} \cdot \mathcal{N}(x\mathfrak{b}\mathfrak{m}_0)^{\frac{1}{n}}$. Let $\mathbf{B}_{\mathfrak{b}\mathfrak{m}_0} = (\mathbf{b}_1, \ldots, \mathbf{b}_n)$ with $\mathbf{b}_j \in K_\mathbb{R}$ (i.e., in the Minkowski space) be a basis of $\mathfrak{b}\mathfrak{m}_0$ satisfying*

$$\|x \cdot \mathbf{b}_i\| \leq 2n \cdot \mathsf{b}^{2n/\mathsf{b}} \cdot \lambda_n(x \cdot \mathfrak{b} \cdot \mathfrak{m}_0)$$

Let $\tau \in K^\mathfrak{m}$ satisfying $\|x\tau\| \leq r$. Then we can perfectly uniformly sample from

$$x(\mathfrak{b}\mathfrak{m}_0 + \tau) \cap r\mathcal{B}_\infty^\tau.$$

within expected time $\mathit{poly}(n, \log|\Delta_K|, \mathrm{size}(\mathfrak{b}), \mathrm{size}(\mathfrak{m}), \mathrm{size}(x), \mathrm{size}(\omega))$

Proof. We apply Proposition 8.8. For this, we need to satisfy all requirements (1) - (6) in Proposition 8.8. We start with (1) and (4), then proceed with (2), (3), (5) and (6). Note that Proposition 8.8 only outputs some $v \in \mathbb{Z}^n$, but because the elements in $x(\mathfrak{b}\mathfrak{m}_0 + \tau)$ can be symbolically represented, we will see that such an output $v = (v_1, \ldots, v_n) \in \mathbb{Z}^n$ can be converted into $x\sum_{j=1}^n \beta_j v_j + x\tau$ where $\beta_j \in K$ is the element in $\mathfrak{b}\mathfrak{m}_0$ associated with the basis element $\mathbf{b}_j$.

Requirements (1) and (4). Since a uniform sample from $x(\mathfrak{b}\mathfrak{m}_0 + \tau) \cap r\mathcal{B}_\infty^\tau$ can be simply obtained by taking a uniform sample of $x\mathfrak{b}\mathfrak{m}_0 \cap (r\mathcal{B}_\infty^\tau - x\tau)$ and adding $x\tau$ afterwards, we concentrate on taking a uniform sample from this latter set. Recall that

$$r\mathcal{B}_\infty^\tau = \{(x_\sigma)_\sigma \in K_\mathbb{R} \mid |x_\sigma| \leq r \text{ and } x_\sigma/\sigma(\tau) > 0 \text{ for real } \sigma \mid \mathfrak{m}\}.$$

Hence we can write $r\mathcal{B}_\infty^\tau = (r'_\sigma)_\sigma \mathcal{B}_\infty + t' = S + t'$ with $r'_\sigma = r/2$ for real $\sigma \mid \mathfrak{m}_\infty$ and $r'_\sigma = r$ otherwise, and $t'_\sigma = r/2$ for real $\sigma \mid \mathfrak{m}_\infty$ and $t'_\sigma = 0$ otherwise; and $S = (r'_\sigma)\mathcal{B}_\infty$ (which is a convex set). Then, putting $t := t' - x\tau$, we have

$$r\mathcal{B}_\infty^\tau - x\tau = S + t' - x\tau = S + t$$

Note that $\|x \cdot \mathbf{b}_i\| \leq r/(24n^2)$ (by the definition of r and Lemma 2.13(ii) and (iv); see also Equation (61)). Hence, writing $D = 2\sum_i \|x \cdot \mathbf{b}_i\| \leq r/(12n)$, we see that $B_{6nD} \subseteq B_{r/2} \subseteq S \subseteq B_r$ and $\|t\|_\infty \leq \|t'\| + \|x\tau\| \leq 2nr$. So, taking $\varepsilon := 1/(6n)$ and $U := 2nr$ satisfies the requirements (1) and (4) of Proposition 8.8.

Requirement (2) and (3). Let us first compute an upper bound on U/D. We have (see Lemma 2.13) $\lambda_1(x\mathfrak{m}_0\mathfrak{b}) \geq \sqrt{n}\mathcal{N}(x\mathfrak{m}_0\mathfrak{b})^{1/n}$ and hence $D \geq 2n^{3/2}\mathcal{N}(x\mathfrak{m}_0\mathfrak{b})^{1/n}$.

Hence, since $U = 2nr$, and $\mathsf{b}^{1/\mathsf{b}} \leq e^{1/e}$,

$$U/D \leq \frac{(2 \cdot n) \cdot 48 \cdot \omega \cdot \mathsf{b}^{2n/\mathsf{b}} \cdot n^{7/2} \cdot |\Delta_K|^{\frac{3}{2n}} \cdot \mathcal{N}(x\mathfrak{b}\mathfrak{m}_0)^{\frac{1}{n}}}{2n^{3/2} \cdot \mathcal{N}(x\mathfrak{m}_0\mathfrak{b})^{1/n}} = 48 \cdot \omega \cdot e^{2n/e} \cdot n^3 \cdot |\Delta_K| \tag{58}$$

We need to (efficiently) approximate $\mathbf{C} \in \frac{1}{N}\mathbb{Z}^n$ with $\|\mathbf{C} - \mathbf{B}\| \leq \|\mathbf{B}^{-1}\|^{-1} \cdot D/U$, with $\mathbf{B} = x\mathbf{B}_{\mathfrak{b}\mathfrak{m}_0}$. For that it is sufficient[7] to choose $N \geq n \cdot \|\mathbf{B}^{-1}\| \cdot U/D$. Using Lemma 24.2, noting that $\lambda_n(x \cdot \mathfrak{b} \cdot \mathfrak{m}_0)/\lambda_1(x \cdot \mathfrak{b} \cdot \mathfrak{m}_0) \leq |\Delta_K|^{1/n}$ (see Lemma 2.13) and $\mathsf{b}^{1/\mathsf{b}} \leq e^{1/e}$, we obtain

$$\begin{aligned} \|\mathbf{B}^{-1}\| &\leq n^{n/2+1}\lambda_1(x\mathfrak{b}\mathfrak{m}_0) \cdot \left(\prod_{j=1}^{n} \frac{2n \cdot \mathsf{b}^{2n/\mathsf{b}} \cdot \lambda_n(x \cdot \mathfrak{b} \cdot \mathfrak{m}_0)}{\lambda_j(x\mathfrak{b}\mathfrak{m}_0)} \right) \\ &\leq n^{n/2+1}|\Delta_K| \cdot (2n)^n \cdot e^{2n^2/e}. \end{aligned} \tag{59}$$

Combining Equation (58) and Equation (59), instantiating

$$N := \lceil n \cdot n^{n/2+1}|\Delta_K| \cdot (2n)^n \cdot e^{2n^2/e} \cdot 48 \cdot \omega \cdot e^{2n/e} \cdot n^3 \cdot |\Delta_K| \rceil \geq n \cdot \|\mathbf{B}^{-1}\|^{-1} \cdot U/D \tag{60}$$

we can certainly compute such approximation $\mathbf{C}$ within time $\mathsf{poly}(n, \log(N), \log(r))$ for such N (since $\|x\mathbf{b}_i\| < r$). Similarly, for requirement (3), we can efficiently compute $\tilde{t} \in \frac{1}{N}\mathbb{Z}^n$ satisfying $\|t - \tilde{t}\| \leq 1/(2N)$. Since $1/(2N) \leq D/2 \leq U$ we certainly have $\|\tilde{t}\| \leq 2U$.

Requirement (5). We can efficiently and perfectly uniformly sample in $\frac{1}{N}\mathbb{Z}^n \cap (1+c)S$ for every $c > 0$ (where we understand $\mathbb{Z}^n \hookrightarrow K_{\mathbb{R}}$ as in Notation 8.9) by simple rejection sampling (where the 'circles' associated to the complex places are done one-by-one). Indeed, for this N, the uniform random sampling in $(1+4\varepsilon)S \cap \frac{1}{N}\mathbb{Z}^n$ (with $\varepsilon = 1/(6n)$) has bit complexity $O(n \log(rN))$.

Requirement (6). For the implementation of the oracle $O_{x\mathbf{B}_{\mathfrak{b}\mathfrak{m}_0}}$, it is enough to show that we can efficiently decide for any rational number $g \in \mathbb{Q}$, integer $k \in \mathbb{Z}_{>0}$ and any embedding σ of K, whether an algebraic number $x\beta$ with $\beta \in \mathfrak{b}\mathfrak{m}_0$ satisfies $|\sigma(x\beta)| > g^{1/k}$ or $|\sigma(x\beta)| \leq g^{1/k}$. Indeed, this holds since the set S is entirely defined in terms of these (absolute values of) embeddings. By putting $k = 2 \cdot \mathsf{b} \cdot n$ we can write $r = g^{1/k}$ for some rational r. The existence and the effectiveness of such an oracle is precisely the object of Corollary 8.4; it is clear that it runs in expected time $\mathsf{poly}(n, \log|\Delta_K|, \mathrm{size}(\mathfrak{b}), \mathrm{size}(\mathfrak{m}), \mathrm{size}(x), \mathrm{size}(\omega))$.

Bit complexity Since $\log(r), \log(N)$ are both $\mathsf{poly}(n, \log|\Delta_K|, \mathrm{size}(\mathfrak{b}), \mathrm{size}(\mathfrak{m}), \mathrm{size}(x), \mathrm{size}(\omega))$, a single run of the algorithm takes bit complexity $\mathsf{poly}(n, \log|\Delta_K|, \mathrm{size}(\mathfrak{b}), \mathrm{size}(\mathfrak{m}), \mathrm{size}(x), \mathrm{size}(\omega))$. If the success probability of a single run is bounded from below by a constant, the entire algorithm (until success) has the same expected bit complexity. It thus remains to show that, for the box $S = (r'_\sigma)\mathcal{B}_\infty$, $\varepsilon = 1/(6n)$, and N as in Equation (60), the success probability is lower bounded by a constant. We have that the success probability is lower bounded by (using Lemma 2.8 with $c = (\min_\sigma |r'_\sigma|)^{-1}$ and (the r of that lemma) $r = (1 \pm 5\varepsilon) \in [\frac{1}{6}, 2]$)

$$\frac{|(1-5\varepsilon)S \cap \frac{1}{N}\mathbb{Z}^n|}{|(1+5\varepsilon)S \cap \frac{1}{N}\mathbb{Z}^n|} \geq \frac{(1-5\varepsilon)^n|S|}{(1+5\varepsilon)^n|S|} \frac{N^n e^{-12nc}}{N^n e^{4nc}} \geq \frac{(1-1/n)^n e^{-16nc}}{(1+1/n)^n} \geq e^{-2} \cdot e^{-1} \geq e^{-3},$$

[7] We have $\|A\| \leq \|A\|_F = (\sum_{ij} |A_{ij}|^2)^{1/2}$ (Frobenius norm) and hence approximating $\mathbf{B}$ by $\mathbf{C} \in N^{-1}\mathbb{Z}^{n \times n}$ yields a $\mathbf{C}$ for which $|\mathbf{C}_{ij} - \mathbf{B}_{ij}| \leq N^{-1}$ and hence $\|\mathbf{B} - \mathbf{C}\| \leq \|\mathbf{B} - \mathbf{C}\|_F \leq nN^{-1}$.

Algorithm 1 Uniform sampling in $x \cdot ((\mathfrak{b}+\gamma) \cap \tau K^{\mathfrak{m},1}) \cap r\mathcal{B}_\infty$

```
Require:
    • An LLL-reduced basis of $\mathcal{O}_K$,
    • A modulus $\mathfrak{m} = \mathfrak{m}_0\mathfrak{m}_\infty$ of the degree $n$ number field $K$.
    • An ideal $\mathfrak{b} \in \mathcal{I}_K^{\mathfrak{m}}$,
    • An element $\gamma \in K$,
    • An element $\tau \in K^{\mathfrak{m}}$,
    • A block size parameter $\mathsf{b} \in \mathbb{Z}$, with $2 \leq \mathsf{b} \leq n$.
    • A element $x \in K_{\mathbb{R}}^{\times}$,
    • A real number $\omega \in \mathbb{Q}_{\geq 1}$.
Ensure: A uniformly distributed element $\beta \in x \cdot ((\mathfrak{b}+\gamma) \cap \tau K^{\mathfrak{m},1}) \cap r\mathcal{B}_\infty$ with
    $r = 48 \cdot \omega \cdot \mathsf{b}^{2n/\mathsf{b}} \cdot n^{7/2} \cdot |\Delta_K|^{\frac{3}{2n}} \cdot \mathcal{N}(x\mathfrak{b}\mathfrak{m}_0)^{\frac{1}{n}}$.
Defining the radius of the ball.
    Define $r = 48 \cdot \omega \cdot \mathsf{b}^{2n/\mathsf{b}} \cdot n^{7/2} \cdot |\Delta_K|^{\frac{3}{2n}} \cdot \mathcal{N}(x\mathfrak{b}\mathfrak{m}_0)^{\frac{1}{n}}$.
Obtaining a short basis of $x\mathfrak{b}\mathfrak{m}_0$.
    Apply Lemma 8.5 to obtain a $\mathsf{b}$-BKZ reduced basis $B_{\mathfrak{b}\mathfrak{m}_0} = (\mathbf{b}_1, \ldots, \mathbf{b}_n)$ of $\mathfrak{b}\mathfrak{m}_0$
    that satisfies
        $\|x \cdot \mathbf{b}_i\| \leq 2n \cdot \mathsf{b}^{2n/\mathsf{b}} \cdot \lambda_n(x \cdot \mathfrak{b} \cdot \mathfrak{m}_0)$
        $\leq 2n \cdot \mathsf{b}^{2n/\mathsf{b}} \cdot \sqrt{n} \cdot |\Delta_K|^{3/(2n)} \cdot \mathcal{N}(x\mathfrak{b}\mathfrak{m}_0)^{1/n} \leq r/(24n^2),$   (61)
    where the second inequality follows from Lemma 2.13(ii) and (iv) and the third
    from the definition of $r$.
Computing a new shift $\gamma_{\mathfrak{m}}$ to take account for the modulus.
    Compute $\gamma_{\mathfrak{m}} \in \mathfrak{b} + \gamma$ such that $\gamma_{\mathfrak{m}} \equiv \tau$ modulo $\mathfrak{m}_0$, which is possible by the fact
    that $\mathfrak{b}$ and $\mathfrak{m}_0$ are coprime. If $\mathfrak{m}_0 = \mathcal{O}_K$, we put $\gamma_{\mathfrak{m}} = \gamma$.
Reducing the shift $\gamma_{\mathfrak{m}}$ modulo the short basis of $\mathfrak{b}\mathfrak{m}_0$.
    Reduce $\gamma_{\mathfrak{m}} \in K$ modulo this $\mathsf{b}$-BKZ-reduced basis $(\mathbf{b}_1, \cdots, \mathbf{b}_n)$ of $\mathfrak{b}\mathfrak{m}_0$, yielding
    $\gamma_{\mathrm{red}} \in \gamma_{\mathfrak{m}} + \mathfrak{b}\mathfrak{m}_0$. That is, write $\gamma_{\mathfrak{m}} = \sum_i t_i \mathbf{b}_i$ and put $\gamma_{\mathrm{red}} = \sum_i (t_i - \lfloor t_i \rceil)\mathbf{b}_i$.
Sampling an element in $x \cdot ((\mathfrak{b}\mathfrak{m}_0 + \gamma_{\mathrm{red}}) \cap r\mathcal{B}_\infty^{\tau}$.
    Using Corollary 8.10, sample $\beta \in x \cdot ((\mathfrak{b}\mathfrak{m}_0 + \gamma_{\mathrm{red}}) \cap r\mathcal{B}_\infty^{\tau}$.
return $\beta$.
```

since $c = (\min_\sigma |r'_\sigma|)^{-1} \leq (r/2)^{-1} \leq 1/(24n)$. □

8.5. Algorithm for sampling in a box intersected with an ideal lattice.

Lemma 8.11. *Let K be a number field, let $\mathfrak{m} = \mathfrak{m}_0\mathfrak{m}_\infty$ be a modulus of $\mathcal{O}_K$, let $\tau \in K^{\mathfrak{m}}$. Let $\mathfrak{b} \in \mathcal{I}_K^{\mathfrak{m}}$ be an ideal of K coprime with $\mathfrak{m}_0$ and let $\gamma \in K$ be a shift. Let $\gamma_{\mathfrak{m}} \in \mathfrak{b} + \gamma$ such that $\gamma_{\mathfrak{m}} \equiv \tau$ modulo $\mathfrak{m}_0$.*

Then

$$(\mathfrak{b}+\gamma) \cap \tau K^{\mathfrak{m},1} = (\mathfrak{b}\mathfrak{m}_0 + \gamma_{\mathfrak{m}}) \cap \tau K_{\mathbb{R}}^{\mathfrak{m}_\infty}$$

Proof. By scaling up $\mathfrak{b}, \gamma, \tau$ by an integer $M \in \mathbb{N}_{>0}$ coprime with $\mathfrak{m}_0$, we can assume that they lie in $\mathcal{O}_K$. It is then enough to show that $(M\mathfrak{b} + M\gamma) \cap M\tau K^{\mathfrak{m},1} = (M\mathfrak{b}\mathfrak{m}_0 + M\gamma_{\mathfrak{m}}) \cap M\tau K_{\mathbb{R}}^{\mathfrak{m}_\infty}$.

We start with inclusion to the right. Since $M\mathfrak{b} + M\gamma \subseteq \mathcal{O}_K$, we have that $\alpha \in (M\mathfrak{b} + M\gamma) \cap M\tau K^{\mathfrak{m},1}$ satisfies $\alpha = M\beta + M\gamma = M\tau + \mu$ with $\beta \in \mathfrak{b}$ and $\mu \in \mathfrak{m}_0 \subseteq \mathcal{O}_K$. Additionally $\sigma(\alpha)/\sigma(\tau) > 0$ for real $\sigma \mid \mathfrak{m}_\infty$ (since multiplying with M does not change the sign of $\sigma(\tau)$).

By the definition of $\gamma_\mathfrak{m} \in K$, we have $\gamma_\mathfrak{m} \in \mathfrak{b} + \gamma$ and $\gamma_\mathfrak{m} \in \mathfrak{m}_0 + \tau$. Hence $\alpha - M\gamma_\mathfrak{m} \equiv 0 \mod$ both $M\mathfrak{b}$ and $\mathfrak{m}_0$. Hence $\alpha \equiv M\gamma_\mathfrak{m}$ modulo $M\mathfrak{b}\mathfrak{m}_0$, i.e., $\alpha \in M\mathfrak{b}\mathfrak{m}_0 + M\gamma_\mathfrak{m}$. Also, $\sigma(\alpha)/\sigma(\tau) > 0$ for real $\sigma \mod \mathfrak{m}_\infty$, so $\alpha \in \tau K_\mathbb{R}^{\mathfrak{m}_\infty}$.

The inclusion to the left can be done similarly. Any $\alpha \in (M\mathfrak{b}\mathfrak{m}_0 + M\gamma_\mathfrak{m}) \subseteq \mathcal{O}_K$ satisfies $\alpha \in M\mathfrak{b} + M\gamma$. Also, since $\alpha \equiv M\gamma_\mathfrak{m} \equiv M\tau \mod \mathfrak{m}_0$ and $\alpha \in \tau K_\mathbb{R}^{\mathfrak{m}_\infty}$, we have $\alpha \in M\tau K^{\mathfrak{m},1}$. This concludes the proof. □

Proposition 8.12 (Correctness and efficiency of Algorithm 1)**.** *Let K be a number field of degree n, let $\mathfrak{m} = \mathfrak{m}_0\mathfrak{m}_\infty$ be a modulus of $\mathcal{O}_K$, let $\mathfrak{b} \in \mathcal{I}_K^\mathfrak{m}$ be an ideal, let $\gamma \in K$ be a shift, let $\tau \in \mathcal{O}_K$ be coprime to $\mathfrak{m}_0$, let $\mathsf{b} \in \{1, \dots, n\}$ be a block size parameter, let $x \in K_\mathbb{R}^\times$ be represented by rational numbers, let $\omega \in \mathbb{Q}_{\geq 1}$ and put $r = 48 \cdot \omega \cdot \mathsf{b}^{2n/\mathsf{b}} \cdot n^{7/2} \cdot |\Delta_K|^{\frac{3}{2n}} \cdot \mathcal{N}(x\mathfrak{b}\mathfrak{m}_0)^{\frac{1}{n}}$.*

Then, the randomized algorithm Algorithm 1 samples from a uniform distribution over $x \cdot ((\mathfrak{b} + \gamma) \cap \tau K^{\mathfrak{m},1}) \cap r\mathcal{B}_\infty$. Moreover, this algorithm runs in expected time $\mathsf{poly}(n, \log|\Delta_K|, \mathrm{size}(\mathfrak{m}), \mathrm{size}(\mathfrak{b}), \mathrm{size}(\gamma), \mathrm{size}(\tau), \mathrm{size}(x), \mathrm{size}(\omega))$.

Proof. The correctness of the algorithm follows from Corollary 8.10 and Lemma 8.11. The only thing that needs to be checked is that γ_{red} satisfies $\|x\gamma_{\mathrm{red}}\| \leq r$. We have $\gamma_{\mathrm{red}} = \sum_i c_i \mathbf{b}_i$ with $c_i \in [-1/2, 1/2)$, hence $\|x\gamma_{\mathrm{red}}\| \leq \sum_i |c_i| \|x\mathbf{b}_i\| \leq r$, since the basis was BKZ-reduced (see line 2).

We conclude the proof with the estimate on the running time. In line 1, a mere instantiation of the radius r is given.

In line 2, a short basis of $x\mathfrak{b}\mathfrak{m}_0$ is computed using a BKZ-like algorithm. This takes, by Lemma 8.5, time $\mathsf{poly}(\mathsf{b}^\mathsf{b}, \mathrm{size}(\mathfrak{b}), \mathrm{size}(\mathfrak{m}_0), \log|\Delta_K|, \mathrm{size}(x))$.

In line 3, a $\gamma_\mathfrak{m} \in K$ is computed satisfying $\gamma_\mathfrak{m} \in (\mathfrak{b} + \gamma) \cap (\mathfrak{m}_0 + \tau)$. For this, it suffices to compute elements $\beta \in \mathfrak{b}$ and $\mu \in \mathfrak{m}_0$ such that $\beta + \mu = 1$ (by putting $\gamma_\mathfrak{m} = \tau\beta + \gamma\mu$). Such a pair $(\beta, \mu) \in \mathfrak{b} \times \mathfrak{m}$ can be found by applying the Hermite normal form to the concatenated basis matrices of $\mathfrak{b}$ and $\mathfrak{m}$ [23, Proposition 1.3.1]. This requires $\tilde{O}(n^5 \log(M)^2)$ time [72], where M is the maximum entry occurring in the basis matrices. Clearly this overall process takes time polynomial in $n, \mathrm{size}(\mathfrak{m}_0), \mathrm{size}(\gamma)$ and $\mathrm{size}(\mathfrak{b})$.

In line 4, this element $\gamma_\mathfrak{m}$ is reduced modulo $\mathfrak{b}\mathfrak{m}_0$, which takes time at most $\mathsf{poly}(\mathrm{size}(\mathfrak{b}), \mathrm{size}(\mathfrak{m}_0), \mathrm{size}(\gamma))$, since $\mathrm{size}(\gamma_\mathfrak{m}) = \mathsf{poly}(\mathrm{size}(\mathfrak{b}), \mathrm{size}(\mathfrak{m}_0), \mathrm{size}(\gamma))$.

Lastly, in line 5, a uniform sample is taken, following Corollary 8.10. The expected bit complexity then follows. □

9. Ideal sampling

9.1. **Introduction.** In this section, we prove the main result of this part, Theorem 9.5. Recall the task at hand. Fix a family of ideals $\mathcal{S}$. Given an ideal $\mathfrak{a}$, find $\beta \in \mathfrak{a}$ such that $\beta\mathfrak{a}^{-1} \in \mathcal{S} \cdot \mathcal{S}_B$ with probability proportional to the density of $\mathcal{S}$.

In Section 6 is proven that the input ideal lattice $\mathfrak{a}$ can be randomized so that its Arakelov class is uniformly distributed. For such random ideal lattices, by Section 7, the event $\beta\mathfrak{a}^{-1} \in \mathcal{S} \cdot \mathcal{S}_B$ happens with the anticipated probability (the ideal density of $\mathcal{S}$) when β is sampled in a large enough box. Subsequently, we proved in Section 8 that one can efficiently sample from such a box. Combining these results together leads to Algorithm 2 and Theorem 9.5, of which the latter can be informally rephrased as follows: There is an efficient way to sample $\beta \in \mathfrak{a}$ satisfying $\beta\mathfrak{a}^{-1} \in \mathcal{S} \cdot \mathcal{S}_B$ with a provable lower bound on the sampling probability.

In the proof of Theorem 9.5 we require three technical results, which are in the later separate Section 9.3 and Section 9.4. The first of these three results is a lemma that states that part of Algorithm 2 is exactly a random walk as in Theorem 6.4, applied to the input ideal lattice $y \cdot \mathfrak{b}$. This allows to apply the density result as in Theorem 7.1. The second result consists of a lemma that states that a distribution $\mathcal{D}$ on $\mathrm{Div}^0_{K^\mathfrak{m}}$ for which holds that the 'folded' distribution $[\mathcal{D}]$ is close to uniform in $\mathrm{Pic}^0_{K^\mathfrak{m}}$, the original distribution $\mathcal{D}$ is close to some $\mathcal{D}_U$ on $\mathrm{Div}^0_{K^\mathfrak{m}}$ satisfying $[\mathcal{D}_U] = U(\mathrm{Pic}^0_{K^\mathfrak{m}})$. This allows for the statistical argument that we may assume the 'randomized' ideal lattice being drawn from $\mathcal{D}_U$ rather than the original distribution $\mathcal{D}$, for the cost of a small error coming from the statistical distance.

The last of these three results is Proposition 9.12, which shows that the output distribution of Algorithm 2 and that of the 'continuous variant of Algorithm 2' (which is to be specified precisely later) are close. This result is very useful because some algorithmic properties are much easier to prove for this 'continuous variant'; this closeness of the output distributions then show that these properties must also hold for the original variant of Algorithm 2, though with a small error due to the slight difference between the output distributions of the two variants.

9.2. **Ideal sampling.**

Definition 9.1. *We denote by $\mathcal{S}_B^G$ the set of B-smooth integral ideals coprime with $\mathfrak{m}_0$ and whose prime divisors lie in the subgroup $G \subseteq \mathrm{Pic}^0_{K^\mathfrak{m}}$, i.e.,*

$$\mathcal{S}_B^G = \big\{\mathfrak{a} \text{ ideal of } \mathcal{O}_K \;\big|\; \mathfrak{p} \mid \mathfrak{a} \text{ implies } \mathfrak{p} \nmid \mathfrak{m}_0, \mathcal{N}(\mathfrak{p}) \leq B \text{ and } [d^0(\mathfrak{p})] \in G\}.$$

For $G = \mathrm{Pic}^0_{K^\mathfrak{m}}$ we just get the set of smooth ideals $\mathcal{S}_B$ coprime with $\mathfrak{m}_0$.

Notation 9.2. *Let $\mathfrak{m} = \mathfrak{m}_0\mathfrak{m}_\infty$ be a modulus of K and let $\tau \in \mathcal{O}_K$ be an element coprime with $\mathfrak{m}_0$. Then we denote*

$$\tau K_\mathbb{R}^{\mathfrak{m}_\infty} := \{(x_\sigma)_\sigma \in K_\mathbb{R} \mid \mathrm{sign}(x_\sigma) = \mathrm{sign}(\sigma(\tau)) \text{ for all real } \sigma \mid \mathfrak{m}_\infty\}.$$

Alternatively, this set consists of all $(x_\sigma)_\sigma \in K_\mathbb{R}$ for which holds $x_\sigma/\sigma(\tau) > 0$ for all real $\sigma \mid \mathfrak{m}_\infty$.

Notation 9.3. *Let $H = \mathrm{Log}(K^0_\mathbb{R})$ be the hyper space of the number field K, i.e., $H = \{(x_\nu)_\nu \in H \mid \sum_\nu x_\nu = 0\}$. Let $(\nu_1, \ldots, \nu_{\mathbb{r}+1})$ be an ordering of the places (with $\mathbb{r} + 1 = n_\mathbb{R} + n_\mathbb{C}$). We denote $\mathbb{Z}_H \subseteq H$ for the integral lattice with basis $\mathbf{B}_H = (e_{\nu_1} - e_{\nu_2}, \ldots, e_{\nu_i} - e_{\nu_{i+1}}, \ldots, e_{\nu_\mathbb{r}} - e_{\nu_{\mathbb{r}+1}})$, where e_{ν_i} is the standard basis. It satisfies $\lambda_1(\mathbb{Z}_H) = \lambda_\mathbb{r}(\mathbb{Z}_H) = \sqrt{2}$ and hence $\mathrm{cov}(\mathbb{Z}_H) \leq n$.*

We will denote $\ddot{H} = \frac{\delta}{n}\mathbb{Z}_H$ for the 'discretized' hyper space, where $\delta > 0$ is some grid parameter. We then have $\mathrm{cov}(\ddot{H}) \leq \delta$.

Lemma 9.4. *The output distribution of Algorithm 2 is independent of the absolute value of the norm $|\mathcal{N}(y)|$ of y, and independent of the signs (and complex phase) of the entries y_σ of $y = (y_\sigma)_\sigma \in K_\mathbb{R}^\times$.*

Proof. The variable y only occurs in lines 6 and 7. The element $\tilde{\beta}$ is perfectly uniformly random over $\big((A_\sigma \cdot y_\sigma)_\sigma \cdot \big[\bar{\mathfrak{b}} \cap \tau K^{\mathfrak{m},1}\big]\big) \cap r \cdot \mathcal{N}(y\bar{\mathfrak{b}})^{1/n} \cdot \mathcal{B}_\infty$. Hence,

$\beta := (A_\sigma^{-1} \cdot y_\sigma^{-1})_\sigma \cdot \tilde{\beta}$ is uniformly distributed over

$$\begin{aligned}&\big(\big[\bar{\mathfrak{b}} \cap \tau K^{\mathfrak{m},1}\big]\big) \cap r \cdot (A_\sigma^{-1} \cdot y_\sigma^{-1})_\sigma \cdot |\mathcal{N}(y)|^{1/n} \cdot \mathcal{N}(\bar{\mathfrak{b}})^{1/n} \cdot \mathcal{B}_\infty \\ &\quad = \big(\big[\bar{\mathfrak{b}} \cap \tau K^{\mathfrak{m},1}\big]\big) \cap r \cdot (A_\sigma^{-1} \cdot (y^0)_\sigma^{-1})_\sigma \cdot \mathcal{N}(\bar{\mathfrak{b}})^{1/n} \cdot \mathcal{B}_\infty,\end{aligned}$$

where $y^0 := y/|\mathcal{N}(y)|^{1/n}$. Hence, the algorithm depends on y^0 and in particular not on $|\mathcal{N}(y)|$. Since the set $\mathcal{B}_\infty$ is symmetric around zero, the signs (and complex phases) of y_σ^0 (and hence of y) do not have any influence on the set $(A_\sigma^{-1} \cdot (y^0)_\sigma^{-1})_\sigma \cdot \mathcal{N}(\bar{\mathfrak{b}})^{1/n} \cdot \mathcal{B}_\infty$. □

Theorem 9.5 (ERH, Ideal sampling theorem)**.** *Let K be a number field of degree n, let $\mathfrak{m} = \mathfrak{m}_0\mathfrak{m}_\infty$ be a modulus, let $G \subseteq \mathrm{Pic}_{K^\mathfrak{m}}^0$ a finite-index subgroup of $\mathrm{Pic}_{K^\mathfrak{m}}^0$ and let $\mathbf{O}_G$ be an oracle that on input an ideal $\mathfrak{c}$ outputs whether $[d^0(\mathfrak{c})] \in G$ or not. Let $\mathfrak{b} \in \mathcal{I}_K$ be an ideal coprime with $\mathfrak{m}_0$, let $\tau \in K^\mathfrak{m}$ (see Section 2.4) satisfying $[(\!|\tau|\!)] \in G$, and let $y \in K_\mathbb{R}^\times$ be represented by rational coordinates. Let $\mathsf{b} \in \{2, \cdots, n\}$ be an integer, let $0 < \varepsilon < \min(1, \frac{20}{n})$ be an error parameter and let $\omega \in \mathbb{Q}_{\geq 1}$. Let $\mathcal{S}$ be a set of integral ideals coprime with $\mathfrak{m}_0$, satisfying $[d^0(\mathcal{S})] \subseteq G - [d^0(\mathfrak{b})]$.*

(A) Correctness. *Algorithm 2 outputs an element $\beta \in \mathfrak{b}$ such that*

- $(\beta) \cdot \mathfrak{b}^{-1} \in \mathcal{S} \cdot \mathcal{S}_B^G$,
- $\beta \in \tau K^{\mathfrak{m},1}$ *(i.e.,* $\mathrm{ord}_\mathfrak{p}(\beta - \tau) \geq \mathrm{ord}_\mathfrak{p}(\mathfrak{m}_0)$ *for all* $\mathfrak{p} \mid \mathfrak{m}_0$ *and* $\sigma(\beta/\tau) > 0$ *for all real* $\sigma \mid \mathfrak{m}_\infty$*),*
- $|\mathcal{N}(\beta)| \leq \mathcal{N}(\mathfrak{b}) \cdot B^N \cdot r^n$

with probability at least

$$\frac{\mathcal{N}(\mathfrak{m}_0)}{\phi(\mathfrak{m}_0)} \cdot \frac{[\mathrm{Pic}_{K^\mathfrak{m}}^0 : G]}{3} \cdot \delta_\mathcal{S}[r^n] - \varepsilon \geq \frac{[\mathrm{Pic}_{K^\mathfrak{m}}^0 : G]}{3} \cdot \delta_\mathcal{S}[r^n] - \varepsilon. \tag{62}$$

Here, $B = \widetilde{O}\Big([\mathrm{Pic}_{K^\mathfrak{m}}^0 : G]^2 \cdot n^2 \cdot \big[n^2 \cdot (\log\log(1/\varepsilon))^2 + (\log(|\Delta_K|\mathcal{N}(\mathfrak{m})))^2\big]\Big)$, $N = \lceil 7n + \log(\mathcal{N}(\mathfrak{m})) + \log|\mathrm{Pic}_{K^\mathfrak{m}}^0| - \log[\mathrm{Pic}_{K^\mathfrak{m}}^0 : G] + 2\log(1/\varepsilon) + 1\rceil$ *as in Corollary 6.5,* $r = 48 \cdot \omega \cdot \mathsf{b}^{2n/\mathsf{b}} \cdot n^{7/2} \cdot |\Delta_K|^{\frac{3}{2n}} \cdot \mathcal{N}(\mathfrak{m}_0)^{\frac{1}{n}}$*, and* $\mathcal{N}(\mathfrak{m}) = 2^{|\mathfrak{m}_\mathbb{R}|} \cdot \mathcal{N}(\mathfrak{m}_0)$ *with* $|\mathfrak{m}_\mathbb{R}|$ *being the number of different real embeddings dividing* $\mathfrak{m}$ *(see Section 2.4).*

(B) Running time. *Furthermore, Algorithm 2 has a bit complexity of* $\mathsf{poly}(\mathsf{b}^\mathsf{b}, \log|\Delta_K|, \mathrm{size}(\mathfrak{b}), \log(1/\varepsilon), \log(\mathcal{N}(\mathfrak{m})), [\mathrm{Pic}_{K^\mathfrak{m}}^0 : G], \mathrm{size}(y), \mathrm{size}(\tau), \mathrm{size}(\omega))$ *and uses at most* $\mathsf{poly}(\log|\Delta_K|, \log\mathcal{N}(\mathfrak{m})) \cdot [\mathrm{Pic}_{K^\mathfrak{m}}^0 : G]$ *queries to* $\mathbf{O}_G$*.*

Proof of Theorem 9.5 (A) Correctness. This proof is structured as follows. We will assume, purely for the sake of argument, that lines 4 and 5 in Algorithm 2 are replaced by the following,

$$\text{“}\mathtt{Sample}\ a = (a_\sigma)_\sigma \leftarrow \mathcal{G}_{H,s}\ \mathtt{and\ put}\ (A_\sigma)_\sigma = (e^{n_{\nu_\sigma}^{-1} a_\sigma})_\sigma\text{”}. \tag{63}$$

and that $y \in K_\mathbb{R}^\times$ is replaced[8] by $y^0 = y/|\mathcal{N}(y)|^{1/n}$ (so that $b = \mathrm{Log}(y) \in H$). This is indeed purely for the sake of argument, since these changes renders this algorithm unprocessable by a computer, due to the real arithmetic. We will show two things: (1) After these changes, which we will call the ‘continuous version of Algorithm 2’, both the correctness and the success probability claim as stated in Equation (62) of (A) do hold (but with $\varepsilon/2$ instead of ε); and (2) the output distribution of this ‘continuous version of Algorithm 2’ is $\varepsilon/2$-statistically close to the output distribution of the ordinary (or discrete) version of Algorithm 2 (that

[8] Note that this particular change of y by y^0 does not impact the output distribution, by Lemma 9.4.

is, without the changes on y and line 4 and 5). Together, we may then conclude that the correctness and success probability claim holds for the ordinary version of Algorithm 2, which is what we aimed to show.

Part (1): Showing the correctness and success probability for the 'continuous version of Algorithm 2'. We assume that lines 4 and 5 from Algorithm 2 are replaced by Equation (63) and that $y \in K_{\mathbb{R}}^0$. We will denote $b = \mathrm{Log}(y) \in H$. Then, by Lemma 9.8, which we will treat later, the ideal-element pair $\big((\beta)\bar{\mathfrak{b}}^{-1}, \beta\big) \in \mathcal{I}_K^{\mathfrak{m}} \times \mathfrak{b}$ from Algorithm 2 is distributed as

$$\big((\alpha) \cdot \mathrm{Exp}(-\mathbf{a}), \alpha \cdot \mathrm{Exp}_\infty(-\mathbf{a}_\infty)\big),$$

with $\mathbf{a} \leftarrow \mathcal{W} = \mathcal{W}_G(N, B, s) + d^0(\mathfrak{b}) + b$ and subsequently $\alpha \leftarrow \mathrm{Exp}(\mathbf{a})_\tau \cap r\mathcal{B}_\infty$ uniformly. Here $\mathcal{W} = \mathcal{W}_G(N, B, s) + d^0(\mathfrak{b}) + b$ is the random walk distribution starting on the point $d^0(\mathfrak{b}) + b \in \mathrm{Div}_{K^\mathfrak{m}}^0$ (see Definition 6.1). Here, $\mathrm{Exp}(\mathbf{a})_\tau$ is defined in Definition 5.2 and $b \in H \subseteq \mathrm{Div}_{K^\mathfrak{m}}^0$.

For the random walk distribution $\mathcal{W} = \mathcal{W}_G(N, B, s) + d^0(\mathfrak{b}) + b$ on $\mathrm{Div}_{K^\mathfrak{m}}^0$ with these parameters (N, B, s) holds that $[\mathcal{W}]$ on $\mathrm{Pic}_{K^\mathfrak{m}}^0$ is $\varepsilon/2$-close to the uniform distribution $\mathcal{U}(G + d^0(\mathfrak{b}))$ on the coset $G + d^0(\mathfrak{b})$ in the total variation distance. So, allowing an error of $\varepsilon/2$ we may as well assume that $\mathbf{a}$ instead comes from a distribution $\mathcal{D}$ on $\mathrm{Div}_{K^\mathfrak{m}}^0$ that satisfies $[\mathcal{D}] = \mathcal{U}(G + d^0(\mathfrak{b}))$ (see Lemma 9.9).

By applying Theorem 7.1, using that $[(\!|\tau|\!)] \in G$, one then obtains that the probability that $(\beta) \cdot \bar{\mathfrak{b}}^{-1} = (\alpha)\,\mathrm{Exp}(-\mathbf{a}) \in \mathcal{S}$ given that $\beta = \alpha \mathrm{Exp}_\infty(-\mathbf{a}_\infty) \in \tau K^{\mathfrak{m},1}$ (which is precisely the way how α is sampled) is at least $\frac{\mathcal{N}(\mathfrak{m}_0)}{\phi(\mathfrak{m}_0)} \frac{[\mathrm{Pic}_{K^\mathfrak{m}}^0 : G]}{3} \cdot \delta_{\mathcal{S}}[r^n] - \varepsilon/2$. From the fact that $\bar{\mathfrak{b}} = \mathfrak{b} \cdot \prod_j \mathfrak{p}_j$ with $\mathfrak{p}_j \nmid \mathfrak{m}$, $[d^0(\mathfrak{p}_j)] \in G$, and $\mathcal{N}(\mathfrak{p}_j) \leq B$, we have that $(\beta) \cdot \mathfrak{b}^{-1} \in \mathcal{S} \cdot \mathcal{S}_B^G$ in that case, and the probability claim of Equation (62) (with $\varepsilon/2$ instead of ε) follows.

We finish part (1) of this proof by showing that the output $\beta \in \mathfrak{b}$ of Algorithm 2 satisfies all bullet points of the theorem. By lines 6 and 7 it follows that $\beta \in \tau K^{\mathfrak{m},1}$. By line 6 it follows that $\tilde{\beta} \in r \cdot \mathcal{N}(\bar{\mathfrak{b}})^{1/n} \cdot \mathcal{B}_\infty$, hence $|\mathcal{N}(\tilde{\beta})| \leq r^n \cdot \mathcal{N}(\bar{\mathfrak{b}}) \leq r^n \cdot B^N \cdot \mathcal{N}(\mathfrak{b})$. Since the Gaussian distortion $(A_\sigma)_\sigma$ (in the 'continuous version') does not change the norm (as $a \in H$), we have $|\mathcal{N}(\beta)| = |\mathcal{N}(\tilde{\beta})|$.

Part (2): Showing that the output distribution of the 'continuous version of Algorithm 2' and the discrete version are $\varepsilon/2$-close. By the proposition in Section 9.4, together with the fact that the discrete Gaussian in line 4 is approximated within statistical distance $\varepsilon/4$, we conclude that with the choice of the 'grid parameter' δ of $\ddot{H}$ in line 4 of Algorithm 2, the statistical difference between the 'continuous variant' and the ordinary variant of Algorithm 2 is at most $\varepsilon/4 + \varepsilon/4 = \varepsilon/2$.

Hence, the success probability of the ordinary Algorithm 2 as in the theorem statement is lower bounded by

$$\frac{\mathcal{N}(\mathfrak{m}_0)}{\phi(\mathfrak{m}_0)} \frac{[\mathrm{Pic}_{K^\mathfrak{m}}^0 : G]}{3} \cdot \delta_{\mathcal{S}}[r^n] - \varepsilon/2 - \varepsilon/2 = \frac{\mathcal{N}(\mathfrak{m}_0)}{\phi(\mathfrak{m}_0)} \frac{[\mathrm{Pic}_{K^\mathfrak{m}}^0 : G]}{3} \cdot \delta_{\mathcal{S}}[r^n] - \varepsilon,$$

as was required to prove. □

Proof of Theorem 9.5 (B) Running time. In the following complexity analysis, any complexity that is within $\mathsf{poly}(\log|\Delta_K|, \mathrm{size}(\mathfrak{b}), \log(1/\varepsilon), \log(\mathcal{N}(\mathfrak{m})), [\mathrm{Pic}_{K^\mathfrak{m}}^0 : G], \mathrm{size}(y), \mathrm{size}(\tau))$ we will call 'polynomial in the size of the input'. Note that $\log B$ and N are $\mathsf{poly}(\log|\Delta_K|, \mathrm{size}(\mathfrak{b}), \log(1/\varepsilon), \log(\mathcal{N}(\mathfrak{m})), \log([\mathrm{Pic}_{K^\mathfrak{m}}^0 : G]))$, so any complexity polynomially bounded by $\log B$ and N must be polynomial in the size of the input as well. We can omit the dependency here on $|G|$ since $\log |G| \leq \log |\operatorname{Pic}^0_{K^{\mathfrak{m}}}| \leq \log(|\Delta_K| \mathcal{N}(\mathfrak{m}))$ (see Lemma 5.1).

We go through lines 1 to 7 of Algorithm 2. Line 1 just initializes B and N and s. Line 2 uses Lemma 5.4 N times to obtain the random primes $\mathfrak{p}_1, \ldots, \mathfrak{p}_N$. This takes complexity $O([\operatorname{Pic}^0_{K^{\mathfrak{m}}} : G] \cdot n^3 \cdot \log^2(B) \cdot N)$ and $O(N \cdot [\operatorname{Pic}^0_{K^{\mathfrak{m}}} : G] \cdot n \log B) = \mathsf{poly}(\log |\Delta_K|, \log \mathcal{N}(\mathfrak{m}))) \cdot [\operatorname{Pic}^0_{K^{\mathfrak{m}}} : G]$ queries to $\mathbf{O}_G$.

In line 3, $\mathfrak{b}$ is multiplied by the N sampled prime ideals. Multiplication of two ideals can be done by LLL-reducing the $n^2 \times n$ matrix involving all products of the $\mathbb{Z}$-generators of the respective ideals, taking bit complexity $\tilde{O}(n^{11} \log(M)^3)$ [59], where M is the maximum entry of the matrix involved. This multiplication is done with N ideals for which $\log(M)$ is bounded by $\mathsf{poly}(\log |\Delta_K|, \operatorname{size}(\mathfrak{b}), \log(B))$, which means that the total bit complexity of this ideal multiplication is in polynomially bounded in the size of the input. An alternative way to see this is by using the two-element representation of ideals (e.g., [26, §4.7]).

Line 4 requires to sample $\ddot{a}$ from an approximate discrete Gaussian $\widehat{\mathcal{G}}_{\frac{\delta}{n}\mathbf{B}_H, \varepsilon/4, s, 0}$ in $\ddot{H} = \frac{\delta}{n}\mathbb{Z}_H$ as in Lemma 2.22 (with $\mathbf{B}_H$ as in Notation 9.3), which is within statistical distance $\varepsilon_{\mathcal{G}} := \varepsilon/4$ of an exact discrete Gaussian. Indeed, observing that the maximum length of the basis of $\frac{\delta}{n}\mathbb{Z}_H$ (see Notation 9.3) is at most $\sqrt{2}\delta/n$ we see that certainly (see line 4)

$$s \geq \delta \sqrt{n \log(2n/(\varepsilon/40))} \geq \sqrt{\frac{\log(4/\varepsilon) + 2\log(n) + 3}{\pi}} \cdot \frac{\sqrt{2}\delta}{n}.$$

This, according to Lemma 2.22, takes expected time polynomial in the input size and $\log(1/\varepsilon)$. Since the input size is $\mathsf{poly}(1/\delta) = \mathsf{poly}(n, \log |\Delta_K|, \log(1/\varepsilon), \log(\omega))$ this Gaussian sampling algorithm has a expected bit complexity of $\mathsf{poly}(n, \log |\Delta_K|, \log(1/\varepsilon), \log(\omega))$. Note that, by Lemma 2.22, the length of $\ddot{a}$ cannot exceed $s \cdot \sqrt{n \log(2n^2/\varepsilon)}$.

In line 5 we need to approximate $(e^{n_{\nu_\sigma}^{-1} \ddot{a}_\sigma})_\sigma$ by rational $(A_\sigma)_\sigma$ such that the relative multiplicative error is small enough. Since the length of $\ddot{a}$ does not exceed $s \cdot \sqrt{n \log(2n^2/\varepsilon)}$ (note that $s = 1/n^2$) we clearly see that $\min_\sigma \ddot{a}_\sigma$ cannot be smaller than $\varepsilon/(2n^2) < e^{-\sqrt{\log(2n^2/\varepsilon)}}$. Hence, approximating $(A_\sigma)_\sigma$ as in line 5 can be done in time $\mathsf{poly}(n, \log(1/\delta), \log(1/\varepsilon))$ which is $\mathsf{poly}(n, \log |\Delta_K|, \log(1/\varepsilon), \log(\omega))$. Note that the size of $(A_\sigma)_\sigma$ itself is also bounded polynomially in these latter parameters.

Line 6 requires Algorithm 1, which uses expected bit complexity at most $\mathsf{poly}(\mathsf{b}^{\mathsf{b}}, \operatorname{size}(\bar{\mathfrak{b}}), \operatorname{size}(\mathfrak{m}_0), \log |\Delta_K|, \operatorname{size}(y), \operatorname{size}((A_\sigma)_\sigma), \operatorname{size}(\tau))$. Note that $\operatorname{size}(\bar{\mathfrak{b}}) = \mathsf{poly}(\operatorname{size}(\mathfrak{b}), N, \log(B))$. Proposition 8.12 can be applied here because we have $r \cdot \mathcal{N}(\bar{\mathfrak{b}})^{1/n} \cdot |\mathcal{N}(y)| = 48 \cdot \omega \cdot \mathsf{b}^{2n/\mathsf{b}} \cdot n^{7/2} \cdot |\Delta_K|^{\frac{3}{2n}} \cdot \mathcal{N}(\mathfrak{m}_0)^{\frac{1}{n}} \cdot \mathcal{N}(\bar{\mathfrak{b}})^{\frac{1}{n}} \cdot |\mathcal{N}(y)|$.

The last line, line 7, only multiplies $\tilde{\beta}$ with $(A_\sigma^{-1} \cdot y_\sigma^{-1})_\sigma$, which has bit complexity polynomial in $\operatorname{size}((A_\sigma)_\sigma, (y_\sigma)_\sigma, \operatorname{size}(\tilde{\beta}))$ which is polynomial in the size of the input, as $\operatorname{size}(\tilde{\beta})$ must be $\mathsf{poly}(r, \operatorname{size}(\bar{\mathfrak{b}}), \operatorname{size}(y), \operatorname{size}((A_\sigma)_\sigma)$.

Therefore, all steps require a bit complexity at most polynomial in the size of the input and b^{b}. The total number of queries to $\mathbf{O}_G$ is at most $\mathsf{poly}(\log |\Delta_K|, \log \mathcal{N}(\mathfrak{m}))) \cdot [\operatorname{Pic}^0_{K^{\mathfrak{m}}} : G]$. □

Remark 9.6. *If $\mathfrak{m} = \mathcal{O}_K$, we have $\mathcal{N}(\mathfrak{m}) = \phi(\mathfrak{m}) = 1$, and $K^{\mathfrak{m}} = K^{\mathfrak{m},1} = K^*$. In this case, $\tau K^{\mathfrak{m},1} = K^*$*

```
Algorithm 2 Sampling of $\beta \in \mathfrak{b}$ such that $\beta \in \tau K^{\mathfrak{m},1}$
Require:
    • An LLL-reduced basis of $\mathcal{O}_K$,
    • A modulus $\mathfrak{m} = \mathfrak{m}_0\mathfrak{m}_\infty$ of the degree $n$ number field $K$.
    • A finite-index subgroup $G \subseteq \mathrm{Pic}^0_{K^{\mathfrak{m}}}$,
    • An ideal $\mathfrak{b} \in \mathcal{I}_K$ coprime with $\mathfrak{m}_0$,
    • An element $y \in K_{\mathbb{R}}^{\times}$,
    • An element $\tau \in K^{\mathfrak{m}}$, satisfying $[(\!(\tau)\!)] \in G$,
    • An error parameter $\varepsilon > 0$.
    • A block size parameter $\mathsf{b} \in \mathbb{Z}$, with $2 \leq \mathsf{b} \leq n$.
    • An oracle $\mathbf{O}_G$ that on input an ideal $\mathfrak{c}$ outputs whether $[d^0(\mathfrak{c})] \in G$.
    • An element $\omega \in \mathbb{Q}_{\geq 1}$.
Ensure: An element $\beta \in \mathfrak{b}$
Let $B, N$ be as in Corollary 6.5, put $s = 1/n^2$ and put $r = 48 \cdot \omega \cdot \mathsf{b}^{2n/\mathsf{b}} \cdot n^{7/2} \cdot |\Delta_K|^{\frac{3}{2n}} \cdot \mathcal{N}(\mathfrak{m}_0)^{\frac{1}{n}}$.
Sample $N$ random prime ideals $\mathfrak{p}_1, \ldots, \mathfrak{p}_N$ uniformly from the set $\{\mathfrak{p} \text{ prime ideal of } K \mid N(\mathfrak{p}) \leq B, \mathfrak{p} \nmid \mathfrak{m}_0, [d^0(\mathfrak{p})] \in G\}$, using the oracle $\mathbf{O}_G$ and Lemma 5.4
Multiply $\mathfrak{b}$ by these $N$ random prime ideals $\mathfrak{p}_j$, obtaining $\bar{\mathfrak{b}} = \mathfrak{b} \cdot \prod_j \mathfrak{p}_j$.
Sample, using Lemma 2.22, an approximation of the discrete Gaussian
    $\ddot{a} = (\ddot{a}_\sigma)_\sigma \leftarrow \widehat{\mathcal{G}}_{\frac{\delta}{n}\mathbf{B}_H, \varepsilon_{\mathcal{G}}, s, 0}$
  with error parameter $\varepsilon_{\mathcal{G}} := \varepsilon/4$, where $\frac{\delta}{n}\mathbf{B}_H$ is a basis of $\ddot{H} = \frac{\delta}{n}\mathbb{Z}_H$ (see Notation 9.3) and
    $\delta := \frac{(\varepsilon/40)^{4n^2 s+1} \cdot s}{\omega^n \cdot e^{10n^2} \cdot |\Delta_K| \cdot \sqrt{n \log(2n/(\varepsilon/40))}}.$
Compute a rationally represented $(A_\sigma)_\sigma \in K^0_{\mathbb{R}}$ such that
    $\max_\sigma |A_\sigma / e^{n_{\nu_\sigma}^{-1} \ddot{a}_{\nu_\sigma}} - 1| \leq \delta/(2n),$
  where $n_{\nu_\sigma} = 2$ if $\sigma$ is complex and 1 otherwise.
Sample, following Algorithm 1 with input $\gamma = 0$, a uniformly random element
    $\tilde{\beta} \in \left((A_\sigma \cdot y_\sigma)_\sigma \cdot \left[\bar{\mathfrak{b}} \cap \tau K^{\mathfrak{m},1}\right]\right) \cap r \cdot |\mathcal{N}(y)|^{1/n} \cdot \mathcal{N}(\bar{\mathfrak{b}})^{1/n} \cdot \mathcal{B}_\infty$
return $\beta = (A_\sigma^{-1} \cdot y_\sigma^{-1})_\sigma \cdot \tilde{\beta} \in \mathfrak{b}$.
```

Remark 9.7. *The requirement $[(\!(\tau)\!)] \in G$ is not necessary. The same proof and algorithm applies for arbitrary $[(\!(\tau)\!)]$; but, as $\beta \in \tau K^{\mathfrak{m},1}$, we have $[(\!(\beta)\!)] = [(\!(\tau)\!)]$. Hence, if we do not assume $[(\!(\tau)\!)] \in G$, the property $[(\!(\beta)\!)] \in G$ is generally not valid anymore.*

9.3. **Two help lemmas.** The following two lemmas are used in the proof of Theorem 9.5. The first of these lemmas shows that part of Algorithm 2 consists of a random walk, whereas the second one shows that one can replace the random walk distribution over $\mathrm{Div}^0_{K^\mathfrak{m}}$ with a close distribution $\mathcal{D}_U$ whose 'folded analogue' $[\mathcal{D}_U]$ in $\mathrm{Pic}^0_{K^\mathfrak{m}}$ is uniform.

Lemma 9.8 (The 'continuous version of Algorithm 2' mimicks a random walk)**.** *Let $\mathfrak{m}$ a modulus, let N, B, s and r as in Algorithm 2 and let $\mathcal{W}_G = \mathcal{W}_G(N, B, s)$ be the random walk distribution on $G \subseteq \mathrm{Pic}^0_{K^\mathfrak{m}}$ (see Definition 6.1). Let $\mathcal{W}_r$ be the distribution on $(\alpha, \mathbf{a}) \in K_\mathbb{R} \times \mathrm{Div}^0_{K^\mathfrak{m}}$ obtained by sampling $\mathbf{a} \leftarrow \mathcal{W}_G(N, B, s) + d^0(\mathfrak{b}) + b$ with $b = \mathrm{Log}(y)$ where $y \in K^0_\mathbb{R}$ followed by sampling*

$$\alpha \in \mathrm{Exp}(\mathbf{a})_\tau \cap r\mathcal{B}_\infty$$

uniformly. Then the pair $\big((\beta) \cdot \bar{\mathfrak{b}}^{-1}, \beta\big) \in \mathcal{I}^\mathfrak{m}_K \times \mathfrak{b}$ obtained by running Algorithm 2 where lines 4 and 5 are replaced by Equation (63) (coined the 'continuous version'), follows the same distribution as $\big((\alpha) \cdot \mathrm{Exp}(-\mathbf{a}), \alpha \cdot \mathrm{Exp}_\infty(-\mathbf{a}_\infty)\big)$ with $(\alpha, \mathbf{a}) \leftarrow \mathcal{W}_r$.

Proof. Both the distribution $\mathcal{W}_G$ and the 'continuous version of Algorithm 2' (where lines 4 and 5 are replaced by Equation (63), as in the proof of Theorem 9.5) involve the following two random processes: picking N uniformly random primes from $\{\mathfrak{p} \in \mathcal{I}^\mathfrak{m}_K \text{ prime } \mid \mathcal{N}(\mathfrak{p}) \leq B \text{ and } [d^0(\mathfrak{p})] \in G\}$ and sampling a Gaussian $a = (a_\nu)_\nu \in H$; both with the exact same parameters. Without loss of generality, we can therefore focus on one fixed sample $\{\mathfrak{p}_j \mid 1 \leq j \leq N\}$ of primes and one fixed vector $a = (a_\nu)_\nu \in H$. This means that we consider the fixed $\mathbf{a} = \sum_{j=1}^N d^0(\mathfrak{p}_j) + a + d^0(\mathfrak{b}) + b \in \mathrm{Div}^0_{K^\mathfrak{m}}$ for the procedure involving $\mathcal{W}_r$ and the fixed ideal $\bar{\mathfrak{b}} = \mathfrak{b} \prod_{j=1}^N \mathfrak{p}_j$ and distortion $\mathrm{Exp}(a + b) = (e^{n_{\nu_\sigma}^{-1}(a_{\nu_\sigma} + b_{\nu_\sigma})})_\sigma$ (recall that $(y_\sigma)_\sigma = (e^{n_{\nu_\sigma}^{-1} b_{\nu_\sigma}})_\sigma$; note that, by Lemma 9.4, we may assume that the entries of y_σ are all positive) for the procedure involving the 'continuous version of Algorithm 2'. Then, writing $\bar{b} = \mathcal{N}(\bar{\mathfrak{b}})^{1/n}$ and $\bar{\mathfrak{b}} = \mathrm{Exp}(\mathbf{a}_\mathrm{f})$, we have

$$\mathrm{Exp}(\mathbf{a})_\tau = \mathrm{Exp}(a + b)(\bar{\mathfrak{b}} \cap \tau K^{\mathfrak{m},1})/\bar{b} \quad \text{and} \quad \mathrm{Exp}_\infty(\mathbf{a}_\infty) = \mathrm{Exp}(a + b)/\bar{b}$$

Thus, $\alpha\, \mathrm{Exp}(-\mathbf{a}_\infty)$ for uniformly random $\alpha \in \mathrm{Exp}(\mathbf{a})_\tau \cap r\mathcal{B}_\infty$ is distributed as

$$\mathrm{Exp}_\infty(-\mathbf{a}_\infty) \cdot \mathcal{U}\big(\,\mathrm{Exp}(\mathbf{a})_\tau \cap r\mathcal{B}_\infty\big) \tag{64}$$

$$= \underbrace{\mathrm{Exp}(-a - b) \cdot \bar{b}}_{\mathrm{Exp}_\infty(-\mathbf{a}_\infty)} \cdot \mathcal{U}\Big(\underbrace{\mathrm{Exp}(a + b)(\bar{\mathfrak{b}} \cap \tau K^{\mathfrak{m},1})/\bar{b}}_{\mathrm{Exp}(\mathbf{a})_\tau} \cap r\mathcal{B}_\infty\Big) \tag{65}$$

$$= \mathcal{U}\Big((\bar{\mathfrak{b}} \cap \tau K^{\mathfrak{m},1}) \cap \mathrm{Exp}(-a - b) \cdot \bar{b} \cdot r\mathcal{B}_\infty\Big) \tag{66}$$

which is exactly the distribution of $\beta \in \bar{\mathfrak{b}}$ in Algorithm 2 for fixed $\bar{\mathfrak{b}}$ and $(a_\nu)_\nu$. It follows that

$$(\alpha)\, \mathrm{Exp}(-\mathbf{a}) = (\alpha)\mathrm{Exp}_\infty(-\mathbf{a}_\infty)\mathrm{Exp}_\mathrm{f}(-\mathbf{a}_\mathrm{f}) = (\alpha)\mathrm{Exp}_\infty(-\mathbf{a}_\infty)/\bar{\mathfrak{b}}$$

is distributed as $(\beta)\bar{\mathfrak{b}}^{-1}$, which finishes the proof. □

We will need the following 'lifting' lemma, which states that the random walk distribution $\mathcal{W}_G(N, B, s)$ over $\mathrm{Div}^0_{K^\mathfrak{m}}$ is close to a distribution $\mathcal{D}$ such that the class $[\mathcal{D}]$ is uniform over a coset $G \subseteq \mathrm{Pic}^0_{K^\mathfrak{m}}$.

Lemma 9.9 (Lifting property of distributions)**.** *Let $G \subseteq \mathrm{Pic}^0_{K^\mathfrak{m}}$ be a finite-index subgroup and suppose that a distribution $\mathcal{D} : \mathrm{Div}^0_{K^\mathfrak{m}} \to \mathbb{R}_{\geq 0}$ satisfies $\|[\mathcal{D}] - \mathcal{U}(G + [\mathbf{b}])\|_1 < \varepsilon$ for some coset $G + [\mathbf{b}]$, where $\mathbf{b} \in \mathrm{Pic}^0_{K^\mathfrak{m}}$. Then there exists a 'lifted' distribution $\mathcal{D}_U : \mathrm{Div}_{K^\mathfrak{m}} \to \mathbb{R}_{\geq 0}$ such that $[\mathcal{D}_U] = \mathcal{U}(G + [\mathbf{b}])$ and $\|\mathcal{D} - \mathcal{D}_U\|_1 < \varepsilon$.*

Proof. Put

$$\mathcal{D}_U(\mathbf{a}) = \begin{cases} \frac{1}{\mathrm{Vol}(G)} \cdot \frac{\mathcal{D}(\mathbf{a})}{[\mathcal{D}]([\mathbf{a}])} & \text{if } [\mathcal{D}]([\mathbf{a}]) \neq 0 \\ u & \text{otherwise} \end{cases},$$

for some $u : \mathrm{Div}^0_{K^\mathfrak{m}} \to \mathbb{R}_{\geq 0}$ that satisfies $[u] = \frac{1}{\mathrm{Vol}(G)} \cdot 1_G \in L_1(\mathrm{Pic}^0_{K^\mathfrak{m}})$. Then, one can check that $[\mathcal{D}_U] = \frac{1}{\mathrm{Vol}(G)} \cdot 1_G$ is uniform on $G + [\mathbf{b}]$. Furthermore, writing F for a fundamental domain in $\mathrm{Div}^0_{K^\mathfrak{m}}$ for $\mathrm{Pic}^0_{K^\mathfrak{m}}$, we have

$$\begin{aligned} \|\mathcal{D} - \mathcal{D}_U\|_1 &= \int_{\mathbf{a}\in F} \sum_{\alpha \in K^{\mathfrak{m},1}/\mu_{K^{\mathfrak{m},1}}} |\mathcal{D}(\mathbf{a} + (\!|\alpha|\!)) - \mathcal{D}_U(\mathbf{a} + (\!|\alpha|\!))| d\mathbf{a} \\ &= \int_{[\mathbf{a}]\in \mathrm{Pic}^0_{K^\mathfrak{m}}} \left| [\mathcal{D}]([\mathbf{a}]) - \frac{1_{G+[\mathbf{b}]}([\mathbf{a}])}{\mathrm{Vol}(G)} \right| d([\mathbf{a}]) = \|[\mathcal{D}] - \mathcal{U}(G + [\mathbf{b}])\|_1 \leq \varepsilon. \end{aligned}$$

The first equation holds by definition, the second equation by the fact that the sign of $(\mathcal{D}(\mathbf{a} + (\!|\alpha|\!)) - \mathcal{D}_U(\mathbf{a} + (\!|\alpha|\!)))$ depends (by construction) solely on the coset $[\mathbf{a}]$. □

9.4. **Statistical distance between the 'continuous variant' and the ordinary variant of Algorithm 2.** In this subsection we show that there is only a small statistical distance between the output distributions of the 'continuous variant of Algorithm 2' (where lines 4 and 5 are replaced by Equation (63)) and the ordinary variant of Algorithm 2 (with no changes). This result is used in the proof of part (A) of Theorem 9.5 and in the proof that Algorithm 2 is almost-Lipschitz in the later Lemma 10.6.

To prove the closeness of these two distributions, we fix all the input parameters and compare the two output distributions; where we note that we may assume that $y \in K^0_\mathbb{R}$ and that the entries of y are positive, by Lemma 9.4. By the law of total probability, it is enough to show the closeness of these two distributions for a fixed sample of the primes $\{\mathfrak{p}_j\}$ (as in line 3), and hence a fixed ideal $\bar{\mathfrak{b}}$. Indeed, the difference between the two distributions can be attributed solely to the way the Gaussian $a = (a_\nu)_\nu \in H$ (discrete or continuous) is sampled and the influence it bears on the uniform distribution of β.

Though the input of Algorithm 2 is given in terms of an ideal $\mathfrak{b}$ and an element $y = \mathrm{Exp}((b_\nu)_\nu) \in K^0_\mathbb{R}$ with $b \in H$ we will often consider the divisor $\mathbf{a} = d^0(\mathfrak{b}) + b$ as input instead. As this is a one-to-one correspondence, and is used only theoretically (for cleanness of notation), no harm is done.

By a similar computation as in the proof of Lemma 9.8, one can deduce that the output β is distributed as (denoting $b = \mathrm{Log}(y) \in H$)

$$\mathcal{U}\Big((\bar{\mathfrak{b}} \cap \tau K^{\mathfrak{m},1}) \cap \mathrm{Exp}(-\mathrm{Log}((A_\sigma)_\sigma) - b) \cdot \mathcal{N}(\bar{\mathfrak{b}})^{1/n} \cdot r\mathcal{B}_\infty\Big) \tag{67}$$

$$= y^{-1} \cdot \mathcal{U}\Big((y \cdot (\bar{\mathfrak{b}} \cap \tau K^{\mathfrak{m},1})) \cap \mathrm{Exp}(-\mathrm{Log}((A_\sigma)_\sigma)) \cdot \mathcal{N}(\bar{\mathfrak{b}})^{1/n} \cdot r\mathcal{B}_\infty\Big) \tag{68}$$

$$= y^{-1} \cdot \mathcal{U}\Big((y \cdot (\bar{\mathfrak{b}}\mathfrak{m}_0 + \gamma)) \cap \mathrm{Exp}(-\mathrm{Log}((A_\sigma)_\sigma)) \cdot \mathcal{N}(\bar{\mathfrak{b}})^{1/n} \cdot r\mathcal{B}^\tau_\infty\Big) \tag{69}$$

both in the 'continuous version' as in the ordinary version (since the definition of $(A_\sigma)_\sigma \in K_\mathbb{R}$ changes accordingly). Equation Equation (69) follows from Lemma 8.11; where $\mathcal{B}^\tau_\infty = \mathcal{B}_\infty \cap \tau K^{\mathfrak{m}_0}$ (see Notation 9.2). Note that from Equation (69) follows that $(A_\sigma)_\sigma$ has only an influence on the shape ('skewness') of the box $\mathcal{N}(\bar{\mathfrak{b}})^{1/n} \cdot r\mathcal{B}^\tau_\infty$ where the uniform sampling takes place. All other things are equal; hence we introduce the following notation.

Notation 9.10. *For $a \in H$, we denote $\mathcal{U}_a$ for the uniform distribution over the set $(y(\bar{\mathfrak{b}}\mathfrak{m}_0+\gamma)) \cap \operatorname{Exp}(a) \cdot \mathcal{N}(\bar{\mathfrak{b}})^{1/n} \cdot r\mathcal{B}_\infty^\tau$, where $r = 48 \cdot \omega \cdot \mathsf{b}^{2n/\mathsf{b}} \cdot n^{7/2} \cdot |\Delta_K|^{\frac{3}{2n}} \cdot \mathcal{N}(\mathfrak{m}_0)^{\frac{1}{n}}$ with $\omega \in \mathbb{Q}_{\geq 1}$; as in Algorithm 2, i.e.,*

$$\mathcal{U}_a := \mathcal{U}\Big((y\cdot(\bar{\mathfrak{b}}\mathfrak{m}_0+\gamma)) \cap \operatorname{Exp}(a) \cdot \mathcal{N}(\bar{\mathfrak{b}})^{1/n} \cdot r\mathcal{B}_\infty^\tau\Big) = \mathcal{U}\Big((y\cdot(\bar{\mathfrak{b}}\mathfrak{m}_0+\gamma)) \cap \operatorname{Exp}(a) \cdot r_0\mathcal{B}_\infty^\tau\Big).$$

where we write $r_0 := \mathcal{N}(\bar{\mathfrak{b}})^{1/n} \cdot r = 48 \cdot \omega \cdot \mathsf{b}^{2n/\mathsf{b}} \cdot n^{7/2} \cdot |\Delta_K|^{\frac{3}{2n}} \cdot \mathcal{N}(\bar{\mathfrak{b}}\mathfrak{m}_0)^{\frac{1}{n}}$. Note that, for $y\beta \in y(\bar{\mathfrak{b}}\mathfrak{m}_0 + \gamma) \cap \tau K_{\mathbb{R}}^{\mathfrak{m}_\infty}$, we have that

$$\mathcal{U}_a[y\beta] > 0 \iff \log|\sigma_\nu(y\beta)| \leq \log r_0 + n_\nu^{-1} a_\nu \textit{ for all } \nu$$

where $n_\nu = 2$ if ν is complex and 1 otherwise. We denote

$$S_{y\beta} = \{a \in H \mid n_\nu^{-1} a_\nu \geq \log|\sigma_\nu(y\beta)| - \log r_0 \textit{ for all } \nu\}.$$

I.e., we have, for $y\beta \in y(\bar{\mathfrak{b}}\mathfrak{m}_0 + \gamma) \cap \tau K_{\mathbb{R}}^{\mathfrak{m}_\infty}$ and $a \in H$,

$$a \in S_{y\beta} \Leftrightarrow y\beta \in \operatorname{Exp}(a) \cdot r_0\mathcal{B}_\infty^\tau \Leftrightarrow \mathcal{U}_a[\beta] > 0.$$

We denote $\partial S_{y\beta}$ for the boundary of $S_{y\beta}$; note that this boundary is a subset of $\mathbb{r} = \dim(H)$ hyperplanes in H of dimension $\mathbb{r} - 1$.

In the ordinary variant of Algorithm 2, an approximation $(A_\sigma)_\sigma$ of $(e^{n_{\nu_\sigma}^{-1}\ddot{a}_{\nu_\sigma}})_\sigma$ with $\ddot{a} \in \ddot{H}$ is computed, hence leading to a slight error. For ease of notation we denote $(A_\sigma)_\sigma = \operatorname{Exp}(\tilde{\ddot{a}}) = (e^{n_{\nu_\sigma}^{-1}\tilde{\ddot{a}}_\sigma})_\sigma$. By construction (see line 4) we have (for $\delta < 1/2$)

$$\begin{aligned} \|\ddot{a} - \tilde{\ddot{a}}\|_2 &\leq n \max_\sigma |\log(A_\sigma) - \log(e^{n_{\nu_\sigma}^{-1}\ddot{a}_{\nu_\sigma}})| = n \max_\nu |\log(A_{\sigma_\nu}/e^{n_{\nu_\sigma}^{-1}\cdot\ddot{a}_\nu})| &(70)\\ &\leq n \max_\sigma \log|1-\delta| \leq 2n \cdot \delta/(2n) = \delta. &(71)\end{aligned}$$

So, the ordinary variant of Algorithm 2, with fixed choices of primes $\{\mathfrak{p}_j\}$ leading to $\bar{\mathfrak{b}}$, has output distribution

$$\sum_{\ddot{a} \in \ddot{H}} \mathcal{U}_{\tilde{\ddot{a}}} \mathcal{G}_{\ddot{H},s}(\ddot{a}) = \sum_{\ddot{a} \in \ddot{H}} \mathcal{U}_{\tilde{\ddot{a}}}(-) \cdot \mathcal{G}_{\ddot{H},s}(\ddot{a}),$$

whereas the 'continuous variant of Algorithm 2' (with the same choices of $\{\mathfrak{p}_j\}$), has output distribution

$$\int_{a \in H} \mathcal{U}_a \mathcal{G}_{H,s}(a) = \int_{a \in H} \mathcal{U}_a(-) \cdot \mathcal{G}_{H,s}(a).$$

Lemma 9.11. *Using Notation 9.10, let $\delta > 0$ be a distance parameter and denote $B_\delta(a) \subseteq H$ for the 2-ball of radius δ around $a \in H$ (where we denote $B_\delta := B_\delta(0)$). Then, for $a \notin \bigcup_{y\beta \in y(\bar{\mathfrak{b}}\mathfrak{m}_0+\gamma)} (\partial S_{y\beta} \boxplus B_\delta)$ and $a' \in B_\delta(a)$,*

$$\mathcal{U}_a = \mathcal{U}_{a'}.$$

Proof. For $a \notin \partial S_{y\beta} \boxplus B_\delta$ holds that $B_\delta(a) \cap \partial S_{y\beta} = \emptyset$. Hence for all $a' \in B_\delta$, the status of $y\beta \in e^{a+a'} \cdot \mathcal{N}(\bar{\mathfrak{b}})^{1/n} \cdot r_0\mathcal{B}_\infty^\tau$ (where $r_0 = \mathcal{N}(\bar{\mathfrak{b}})^{1/n} \cdot r$) does not change. Hence, if $a \notin \bigcup_{y\beta \in y(\bar{\mathfrak{b}}\mathfrak{m}_0+\gamma)} (\partial S_{y\beta} \boxplus B_\delta)$, the entire distribution $\mathcal{U}_{a-a'} = \mathcal{U}_a$ is constant for $a' \in B_\delta$. This concludes the proof. □

Proposition 9.12. *Using the notation of Notation 9.10, let $1/(2n) > \varepsilon > 0$ be an error parameter. Let $s > 0$ be a Gaussian parameter and let $\ddot{H} \subseteq H$ be a full-rank lattice whose Voronoi domain F satisfies*

$$\delta := \max_{a \in F} \|a\| \leq \frac{\varepsilon^{4n^2 s+1} \cdot s}{\omega^n \cdot e^{10n^2} \cdot |\Delta_K| \cdot \sqrt{n \log(2n/\varepsilon)}} \tag{72}$$

Then

$$\left\| \int_{a \in H} \mathcal{U}_x \mathcal{G}_{H,s}(a) da - \sum_{\ddot{a} \in \ddot{H}} \mathcal{U}_{\tilde{\ddot{a}}} \mathcal{G}_{\ddot{H},s}(\ddot{a}) \right\|_1 \leq 9\varepsilon$$

Proof. Writing F for the Voronoi fundamental domain of the lattice $\ddot{H}$, using the identity $AB - A'B' = (A - A')B + A'(B - B')$, the triangle inequality and the fact that $\|\mathcal{U}_{\tilde{\ddot{a}}}\|_1 = 1$, we obtain

$$\begin{aligned}
&\left\| \int_{a \in H} \mathcal{U}_x \mathcal{G}_{H,s}(a) da - \sum_{\ddot{a} \in \ddot{H}} \mathcal{U}_{\tilde{\ddot{a}}} \mathcal{G}_{\ddot{H},s}(\ddot{a}) \right\|_1 \\
&\leq \left\| \int_{a \in F} \sum_{\ddot{a} \in \ddot{H}} \mathcal{U}_{\ddot{a}+a} \mathcal{G}_{H,s}(\ddot{a} + a) - \mathcal{U}_{\tilde{\ddot{a}}} \frac{\mathcal{G}_{\ddot{H},s}(\ddot{a})}{\mathrm{Vol}(F)} da \right\|_1 \\
&\leq \underbrace{\int_{a \in F} \sum_{\ddot{a} \in \ddot{H}} \|\mathcal{U}_{\ddot{a}+a} - \mathcal{U}_{\tilde{\ddot{a}}}\|_1 \mathcal{G}_{H,s}(\ddot{a} + a) da}_{(I)} + \underbrace{\int_{a \in F} \sum_{\ddot{a} \in \ddot{H}} \left| \mathcal{G}_{H,s}(\ddot{a} + a) - \frac{\mathcal{G}_{\ddot{H},s}(\ddot{a})}{\mathrm{Vol}(F)} \right| da}_{(II)} .
\end{aligned} \tag{73}$$

We will bound the parts (I) and (II) separately, starting with (I).
Bound on (I).
In the following sequence of inequalities we use respectively Lemma 9.11 (in combination with Equation (70)), $\mathrm{cov}(\ddot{H}) = \max_{x \in F} \|x\| = \delta$ and the fact that total variation distances are always bounded by 2. Also, we use a Gaussian tail bound (see Lemma 2.19) with radius $R := s\sqrt{n \log(2n/\varepsilon)} > 2\delta > 0$ (since we can deduce that $\delta < s/2$ from the assumption in Equation (72)). Combining this, writing $T_\delta = \bigcup_{y\beta \in y(\bar{\mathfrak{b}}\mathfrak{m}_0 + \gamma)} (\partial S_{y\beta} \boxplus B_\delta)$, we obtain

$$(I) \leq \int_{a \in F} \sum_{\substack{\ddot{a} \in \ddot{H} \\ \ddot{a} \in T_\delta}} \|\mathcal{U}_{\ddot{a}+a} - \mathcal{U}_{\tilde{\ddot{a}}}\|_1 \mathcal{G}_{H,s}(\ddot{a} + a) da \leq 2 \int_{a \in F} \sum_{\substack{\ddot{a} \in \ddot{H} \\ \ddot{a} \in T_\delta}} \mathcal{G}_{H,s}(\ddot{a} + a) da \tag{74}$$

$$\leq 2 \int_{a \in T_{2\delta}} \mathcal{G}_{H,s}(a) da \leq \varepsilon + 2 \int_{\substack{a \in T_{2\delta} \\ \|a\| < R}} \mathcal{G}_{H,s}(a) da \tag{75}$$

The third inequality follows from the observation that $\ddot{a} + a \in \partial S_{y\beta} \boxplus B_{2\delta}$ if $\ddot{a} \in \partial S_{y\beta} \boxplus B_\delta$ (since $a \in F \subseteq B_\delta$).

If $a \in \partial S_{y\beta} \boxplus B_{2\delta}$ and $\|a\| < R$, then (for all ν) $n_\nu^{-1} a_\nu \geq \log|\sigma_\nu(y\beta)| - \log r_0 - R$ (since $R > 2\delta$), and hence $\log|\sigma_\nu(y\beta)| \leq n_\nu^{-1} a_\nu + \log r_0 + R$, i.e., $\max_\sigma |\sigma(y\beta)| \leq r_0 \cdot \exp(\max_\nu a_\nu) e^R \leq r_0 e^{2R}$ (since we assumed $\|a\| < R$). Recall, by definition, $r_0 = \mathcal{N}(\bar{\mathfrak{b}})^{1/n} \cdot r$. Hence, by the condition $\|a\| < R$, we may take the union $T_{2\delta} = \bigcup_{y\beta \in y(\bar{\mathfrak{b}}\mathfrak{m}_0 + \gamma)} (\partial S_{y\beta} \boxplus B_{2\delta})$ only over those $y\beta$ satisfying $y\beta \in r_0 e^{2R} \mathcal{B}_\infty$.

We concentrate for the moment on the Gaussian weight on a single $S_{y\beta} \boxplus B_{2\delta}$ in the union of $T_{2\delta}$. Note that $\partial S_{y\beta}$ lies in a union of $\mathfrak{r} = \dim(H) \leq n$ hyperplanes in H. Denoting P for a hyperplane in H of dimension $\dim(H) - 1$ that goes through the origin, we have, for all $y\beta \in y(\bar{\mathfrak{b}}\mathfrak{m}_0 + \gamma)$,

$$\int_{a \in \partial S_{y\beta} \boxplus B_{2\delta}} \mathcal{G}_{H,s}(a) da \leq n \int_{a \in P \boxplus B_{2\delta}} \mathcal{G}_{H,s}(a) da \leq n \cdot \mathrm{erf}(2\delta/s) \leq 4n \cdot \delta/s, \tag{76}$$

where erf is the standard error function of the standard normal distribution that satisfies $\mathrm{erf}(x) \leq \frac{2}{\sqrt{\pi}} x$. The inequality in Equation (76) holds because the hyperplane $P \subseteq H$ going through the origin is the one where $P \boxplus B_{2\delta}$ has the largest Gaussian weight.

Writing $e^{2nR} \cdot N_0$ for an upper bound for $|(y(\bar{\mathfrak{b}}\mathfrak{m}_0 + \gamma)) \cap r_0 e^{2R} \mathcal{B}_\infty|$ for which we will instantiate N_0 later, we combine Equation (75) and Equation (76), and the assumption in Equation (72) on δ, to obtain

$$(I) \leq \varepsilon + 2 \int_{\substack{a \in \bigcup_{\beta \in \mathfrak{b}} (\partial S_\beta \boxplus B_{2\delta}) \\ \|a\| < R}} \mathcal{G}_{H,s}(a) da \leq \varepsilon + \frac{8n \cdot e^{2nR} \cdot N_0 \cdot \delta}{s} \leq 2\varepsilon. \tag{77}$$

It remains to show that δ as in the assumptions of this proposition indeed satisfies $\delta \leq \varepsilon \cdot s \cdot (8n)^{-1} e^{-2nR} \cdot N_0^{-1}$. We have $\exp(-2nR) = \exp(-2n^{3/2} s \sqrt{\log(2n/\varepsilon)}) \geq \exp(-4n^2 s \log(1/\varepsilon)) = \varepsilon^{4n^2 s}$ (where we use that $\varepsilon < 1/(2n)$). Also, by a similar reasoning as in Section 7.2.2, using $r_0 = \mathcal{N}(\bar{\mathfrak{b}})^{1/n} r$, and the definition $r = 48 \cdot \omega \cdot \mathsf{b}^{2n/\mathsf{b}} \cdot n^{7/2} \cdot |\Delta_K|^{\frac{3}{2n}} \cdot \mathcal{N}(\mathfrak{m}_0)^{\frac{1}{n}}$, we have

$$\begin{aligned} N_0 &\in [e^{-1/4}, e^{1/4}] \cdot \frac{\mathcal{N}(\bar{\mathfrak{b}}) \cdot 2^{n_\mathbb{R}} \cdot (2\pi)^{n_\mathbb{C}} \cdot r^n}{\mathcal{N}(\bar{\mathfrak{b}}\mathfrak{m}_0) \cdot \sqrt{|\Delta_K|}} \\ &\in [e^{-1/4}, e^{1/4}] \cdot 48^n \cdot \omega^n \cdot 2^{n_\mathbb{R}} \cdot (2\pi)^{n_\mathbb{C}} \cdot \mathsf{b}^{2n^2/\mathsf{b}} \cdot n^{7n/2} \cdot |\Delta_K| \end{aligned}$$

And hence $N_0 \leq \omega^n \cdot e^{6n^2} \cdot |\Delta_K|$ since $48^n \cdot \mathsf{b}^{2n^2/\mathsf{b}} \cdot n^{7n/2} \leq e^{n \log(48) + 2n^2/e + 7n \log(n)/2} \leq e^{6n^2}$ (because $\log(48) + (2/e) \cdot n + 7 \log(n)/2 \leq 6n$ for all $n > 0$). Therefore, indeed,

$$\begin{aligned} \delta &\leq \frac{\varepsilon^{4n^2 s + 1} \cdot s}{\omega^n \cdot e^{10n^2} \cdot |\Delta_K| \cdot \sqrt{n \log(2n/\varepsilon)}} \\ &= \varepsilon \cdot s \cdot e^{-4n^2} \cdot \varepsilon^{4n^2 s} \cdot \omega^{-n} \cdot e^{-6n^2} \cdot |\Delta_K|^{-1} \cdot (n \log(2n/\varepsilon))^{-1/2} \\ &\leq \varepsilon \cdot s \cdot (8n)^{-1} e^{-2nR} \cdot N_0^{-1} \end{aligned} \tag{78}$$

Bound on (II)

We still write F for the Voronoi domain of $\ddot{H}$. We apply Lemma 9.13 to obtain a bound on (II). Indeed, by putting $\Lambda = \ddot{H}$ and noting that $\lambda_n(\ddot{H}) \leq 2\delta = 2\,\mathrm{cov}_2(\ddot{H})$ (see Lemma 2.3) we surely have

$$\lambda_n(\ddot{H}) \leq 2\delta \leq 2 \cdot \frac{\varepsilon^{4n^2 s + 1} \cdot s}{e^{10n^2} \cdot |\Delta_K| \cdot \sqrt{n \log(2n/\varepsilon)}} \leq \frac{\varepsilon \cdot s}{5\pi n \cdot 2 \cdot \sqrt{n \log(2n/\varepsilon)}}$$

where we use that $e^{10n^2} > 20\pi n$ for $n \geq 1$. Now, since $2 \cdot \sqrt{n \log(2n/\varepsilon)} > \sqrt{n \log(4n/\varepsilon)}$ the parameter s satisfies the assumptions of Lemma 9.13, hence we can bound (II) by 7ε.

Combining Equation (73) and the bound on (II), we obtain

$$\left\| \int_{a\in H} \mathcal{U}_x \mathcal{G}_{H,s}(a)da - \sum_{\ddot{a}\in\ddot{H}} \mathcal{U}_{\ddot{a}} \mathcal{G}_{\ddot{H},s}(\ddot{a}) \right\|_1 \leq 9\varepsilon,$$

as required. □

Lemma 9.13. *Let $\Lambda \subseteq V$ be a full-rank lattice, let F be the Voronoi domain of Λ, let $\varepsilon < 1/2$ be some error parameter and let $s > 5\pi \cdot \sqrt{n\log(4n/\varepsilon)} \cdot \varepsilon^{-1} \cdot n \cdot \lambda_n(\Lambda)$. Then*

$$\int_{x\in F} \sum_{\ell\in\Lambda} \left|\mathcal{G}_{V,s}(\ell+x) - |F|^{-1}\mathcal{G}_{\Lambda,s}(\ell)\right| dx \leq 7\varepsilon$$

Proof. Note that $\operatorname{cov}_2(\Lambda) = \max_{x\in F}\|x\| \leq n\cdot\lambda_n(\Lambda)/2$ (see Lemma 2.3). Using smoothing arguments, (using that $s > \sqrt{\log(2n(1+1/\varepsilon))/\pi}\cdot\lambda_n(\Lambda)$) writing $d = \dim(V)$, we see that (see Lemma 2.20)

$$|F|^{-1}\cdot\mathcal{G}_{\Lambda,s}(\ell) = g_s(\ell)\cdot\frac{1}{|F|\cdot g_s(\Lambda)} \in [1-2\varepsilon, 1+2\varepsilon]\cdot s^{-d}\cdot g_s(\ell) = [1-2\varepsilon, 1+2\varepsilon]\cdot\mathcal{G}_{V,s}(\ell),$$

where we used that $1-2\varepsilon \leq (1+\varepsilon)^{-1} \leq (1-\varepsilon)^{-1} \leq 1+2\varepsilon$ for $\varepsilon < 1/2$. Using this, in combination with the tail bound (both using Corollary 2.18 and Lemma 2.19) with tail cut parameter $R := s\sqrt{n\log(2n/\varepsilon)} > \operatorname{cov}_2(\Lambda)$, we obtain

$$\int_{x\in F} \sum_{\ell\in\Lambda} \left|\mathcal{G}_{V,s}(\ell+x) - |F|^{-1}\mathcal{G}_{\Lambda,s}(\ell)\right| dx \tag{79}$$

$$\leq 2\varepsilon + \int_{x\in F} \sum_{\substack{\ell\in\Lambda\\ \|\ell\|\leq 2R}} \left|\mathcal{G}_{V,s}(\ell+x) - |F|^{-1}\cdot\mathcal{G}_{\Lambda,s}(\ell)\right| dx$$

$$\leq 2\varepsilon + 4\varepsilon + \int_{a\in F} \sum_{\substack{\ell\in\Lambda\\ \|\ell\|\leq 2R}} \left|\mathcal{G}_{V,s}(\ell+x) - \mathcal{G}_{V,s}(\ell)\right| dx$$

$$\leq 6\varepsilon + \int_{x\in F} \sum_{\substack{\ell\in\Lambda\\ \|\ell\|\leq 2R}} \mathcal{G}_{V,s}(\ell+x)|1 - e^{5\pi\cdot\operatorname{cov}_2(\Lambda)\cdot R/s^2}| dx \tag{80}$$

$$\leq 6\varepsilon + \frac{10\pi\operatorname{cov}_2(\Lambda)R}{s^2} \leq 6\varepsilon + \frac{5\pi n\lambda_n(\Lambda)R}{s^2} < 7\varepsilon. \tag{81}$$

The inequality in Equation (80) follows from the fact that

$$-\frac{s^2}{\pi}\log g_s(\ell+x) = \|\ell+x\|^2 = \|\ell\|^2 + 2\langle\ell,x\rangle + \|x\|^2$$
$$\in [\|\ell\|^2 - 4R\operatorname{cov}_2(\Lambda), \|\ell\|^2 + 5R\operatorname{cov}_2(\Lambda)]$$

which follows by Cauchy-Schwarz and the fact that $\max_{x\in F}\|x\| \leq \operatorname{cov}_2(\Lambda) \leq R$. The last inequalities in Equation (81) follow from $|1-e^x| \leq 2x$ for $x < 1/2$, and

$$\frac{5\pi n\lambda_n(\Lambda)R}{s^2} = \frac{5\pi n\lambda_n(\Lambda)\sqrt{n\log(2n/\varepsilon)}}{s} < \varepsilon.$$

□

9.5. **The sampling theorem in arbitrary orders of a number field.** The sampling theorem (Theorem 9.5) is not restricted to the ring of integers $\mathcal{O}_K$ of a number field K. Below we give a quick sketch how to amend the reasoning as to obtain a similar result for arbitrary orders within K.

Write $R \subseteq \mathcal{O}_K$ for an order, $\mathfrak{f}_R = \{x \in K \mid x\mathcal{O}_K \subseteq R\}$ for its conductor and Δ_R for its discriminant. Then we can in fact apply the sampling algorithm (Algorithm 2) for R in place of $\mathcal{O}_K$ if $\mathfrak{f}_R \mid \mathfrak{m}$. Indeed, for all ideals and elements involved are assumed to be coprime with $\mathfrak{f}_R$, there is a one-to-one correspondence between ideal arithmetic in R and $\mathcal{O}_K$, by the maps $\mathfrak{A} \mapsto \mathfrak{A}\mathcal{O}_K$ and $\mathfrak{a} \mapsto \mathfrak{a} \cap R$ for ideals $\mathfrak{A}$ of R and ideals $\mathfrak{a}$ of $\mathcal{O}_K$. These maps are compatible with ideal multiplication, as long as all ideals involved are coprime to $\mathfrak{f}_R$ (e.g., combine [58, Chapter I, Proposition 12.10] and [3, Chapter 1, Extension and Contraction]).

Lemma 9.14. *Let $R \subseteq \mathcal{O}_K$ an order of a number field K with conductor $\mathfrak{f}_R$. Suppose that $\mathfrak{f}_R \mid \mathfrak{m}$ and that $\tau \in R$. Then replacing every occurring ideal $\mathfrak{a}$ (that is, $\mathfrak{b}, \mathfrak{p}_j, \mathfrak{m}_0$) by the ideal $\mathfrak{A} = \mathfrak{a} \cap R$ in Algorithm 2 and using only arithmetic in R, does not change the output distribution.*

Proof. We go through every step of Algorithm 2, where we will use that $\mathfrak{b}$ and the primes $\mathfrak{p}_j$ are coprime with $\mathfrak{m}_0$. In step 3 we multiply $\mathfrak{B} = \mathfrak{b} \cap R$ by random prime ideals $\mathfrak{P}_j = \mathfrak{p}_j \cap R$ satisfying $[d^0(\mathfrak{p}_j)] \in G$. Hence we obtain $\bar{\mathfrak{B}} = \bar{\mathfrak{b}} \cap R$ with $\bar{\mathfrak{b}} = \mathfrak{b} \cdot \prod_j \mathfrak{p}_j$. The Gaussian distortion in step 4, is not affected by the specific order R.

In step 6, we have $\bar{\mathfrak{b}} \cap \tau K^{\mathfrak{m},1} = (\bar{\mathfrak{b}}\mathfrak{m}_0 + \tilde{\tau}) \cap \tau K_{\mathbb{R}}^{\mathfrak{m}_\infty}$ for $\tilde{\tau} \in \bar{\mathfrak{b}}$ satisfying $\tilde{\tau} \equiv \tau \bmod \mathfrak{m}_0$ (see Lemma 8.11). Since $\mathfrak{f}_R \mid \mathfrak{m}_0$, we have that $\mathfrak{M}_0 := \mathfrak{m}_0 \cap R = \mathfrak{m}_0$, by the very definition of the conductor; it is the largest ideal of $\mathcal{O}_K$ that is also an ideal in R. Therefore, $\tilde{\tau} \in \tau + \mathfrak{m}_0 \subseteq R$, as τ is assumed to be in R.

So, $\bar{\mathfrak{b}} \cap \tau K^{\mathfrak{m},1} \cap R = (\bar{\mathfrak{b}}\mathfrak{m}_0 + \tilde{\tau}) \cap R$. Since $\mathfrak{f}_R \mid \mathfrak{m}_0$, we have that $\bar{\mathfrak{B}}\mathfrak{M}_0 = (\bar{\mathfrak{b}}\mathfrak{m}_0) \cap R = \bar{\mathfrak{b}}\mathfrak{m}_0$. Hence, $\bar{\mathfrak{b}}\mathfrak{m}_0 + \tilde{\tau} = \bar{\mathfrak{B}}\mathfrak{M}_0 + \tilde{\tau} \subseteq R$.

Then it follows that the sampling an output distribution of steps 6 and 7 are the same as in the original algorithm. □

Remark 9.15. *Note that, if only a sub-order $R \subseteq \mathcal{O}_K$ is known, the call to Algorithm 1 in line 6 of Algorithm 2 cannot be done with $r = 48 \cdot \omega \cdot \mathsf{b}^{2n/\mathsf{b}} \cdot n^{7/2} \cdot |\Delta_K|^{\frac{3}{2n}} \cdot \mathcal{N}(\mathfrak{m}_0)^{\frac{1}{n}}$. Instead, the larger $r = 48 \cdot \omega \cdot \mathsf{b}^{2n/\mathsf{b}} \cdot n^{7/2} \cdot |\Delta(R)|^{\frac{3}{2n}} \cdot \mathcal{N}(\mathfrak{m}_0)^{\frac{1}{n}}$ must then be used, where $|\Delta(R)| = [\mathcal{O}_K : R]^2 |\Delta_K|$ is the discriminant of R.*

10. Properties of the ideal sampling algorithm

10.1. **Introduction.** In this section we will treat several important properties of the ideal sampling algorithm (Algorithm 2). These properties are phrased in terms of the *output distribution* of Algorithm 2, that is, the distribution of $\beta \in K$ on line 5. This distribution has three important properties: having the *shifting property*, being *bounded* and being *almost Lipschitz continuous.*

The boundedness and the shifting property are useful to show certain *randomness properties* of the output distribution of Algorithm 2 on a random input. In Part 2 of this paper, we will use Algorithm 2 on a 'Gaussian' input to rigorously compute $\mathbb{S}$-unit groups. For this, these two properties will be of fundamental importance to show that sampling sufficiently many 'relations' will eventually lead to the full $\mathbb{S}$-unit group.

The (almost-) Lipschitz continuous property is there to show that rounding or slightly disturbing the input of the ideal sampling algorithm does not impact the output of this algorithm significantly (this concerns the variable $\mathrm{Log}(y) = b = (b_\nu)_\nu \in H$). From this one can conclude that rounding or finite precision issues do not play a role at all for the input of Algorithm 2.

By Lemma 9.4, the output distribution of Algorithm 2 is independent on the norm of $y \in K_{\mathbb{R}}^{\times}$ and the signs of the entries. Hence, in theoretical arguments, we are always allowed to replace y by $y^0 = y/|\mathcal{N}(y)|^{1/n} \in K_{\mathbb{R}}^0$ and to assume that the entries of y^0 are positive. In the actual running algorithm we will not do this, because in general y^0 does not consist of rational numbers.

The shifting property. The shifting property of Algorithm 2 relates a certain change in the input to a similar change in the output. For this property to be phrased succinctly it is useful to consider the input of the algorithm to be the zero-degree Arakelov divisor $\mathbf{a} = d^0(\mathfrak{b}) + b$ (one retrieves $\mathfrak{b} = \mathrm{Exp}(\mathbf{a}_{\mathrm{f}})$ and $b = \mathbf{a} - d^0(\mathfrak{b})$, where $b = \mathrm{Log}(y)$; here we use $H \subseteq \mathrm{Div}^0_{K^{\mathfrak{m}}}$).

Shifting the input $\mathbf{a}$ by a principal divisor, yielding $\mathbf{a} + (\!|\alpha|\!)$ (with $\alpha \in K^{\mathfrak{m},1}$), has as a result that the output distribution of Algorithm 2 is 'multiplied' by α. More precisely, the probability of sampling $\eta \in K$ in the ideal sampling algorithm on input divisor $\mathbf{a}$ is precisely the same as the probability of sampling $\alpha \cdot \eta$ on input $\mathbf{a} + (\!|\alpha|\!)$. Writing $\mathcal{D}_{\mathbf{a}}$ for the output distribution of Algorithm 2 (see Notation 10.1) this can be more succinctly written as:

$$\mathcal{D}_{\mathbf{a}}(\eta) = \mathcal{D}_{\mathbf{a}+(\!|\alpha|\!)}(\alpha \cdot \eta).$$

This is called the 'shifting' property because a shift of the input divisor by a principal divisor $(\!|\alpha|\!)$ causes the output distribution to 'shift' (multiplicatively) by the same element α.

This property of Algorithm 2 is proven in Section 10.3.

Boundedness. The output distribution of Algorithm 2 being bounded means here that the output cannot exceed a certain size. This property is almost immediate by the fact that the output element is sampled in a (slightly) distorted box, where the distortion is dictated by a truncated discrete Gaussian (see Lemma 2.22). Exactly the dimensions of this box give the upper bound on the size of the elements; this is proven in Section 10.4.

Lipschitz continuous. The output distribution being almost-*Lipschitz* is useful when one considers the input to be discretized or rounded. Again, using the notation $\mathcal{D}_{\mathbf{a}}$ for the output distribution of Algorithm 2 on input divisor $\mathbf{a}$ (see Notation 10.1), being almost Lipschitz means that the total variation distance between $\mathcal{D}_{\mathbf{a}}$ and $\mathcal{D}_{\mathbf{a}+a}$ with $a = (a_\nu)_\nu \in H$ is linearly upper bounded by $\|a\|$, allowing some slack by adding a small error. More precisely, there exist $L \in \mathbb{R}_{>0}$ and a small $\eta \in [0, 1)$ such that for all $\mathbf{a}, a \in H$,

$$\|\mathcal{D}_{\mathbf{a}+a} - \mathcal{D}_{\mathbf{a}}\|_1 \leq L \cdot \|a\| + \eta.$$

This fact is proven in Section 10.5.

10.2. **Preliminaries.** The input of Algorithm 2 is given in terms of an ideal $\mathfrak{b}$ and an element $y = \operatorname{Exp}((b_\nu)_\nu) \in K^0_{\mathbb{R}}$ with $b \in H$. In this section about properties of this algorithm, it is cleaner to use an Arakelov divisor $\mathbf{a} = d^0(\mathfrak{b}) + b$ as input instead[9].

Notation 10.1 (Output distribution)**.** *For a fixed number field K, with modulus $\mathfrak{m}$, finite-index subgroup $G \subseteq \operatorname{Pic}^0_{K^{\mathfrak{m}}}$, element $\tau \in \mathcal{O}_K$ coprime with $\mathfrak{m}_0$ and fixed error parameter $\varepsilon > 0$, block size b, $\omega \in \mathbb{Q}_{\geq 1}$ and $\mathbf{a} = d^0(\mathfrak{b}) + b \in \operatorname{Div}^0_{K^{\mathfrak{m}}}$ (where b is understood via the inclusion $H \subseteq \operatorname{Div}^0_{K^{\mathfrak{m}}}$), we denote*

$$\mathcal{D}_{\mathbf{a}} \in L_1(K^*)$$

for the output distribution of β from Algorithm 2 on input $\mathfrak{b}$ (coprime with $\mathfrak{m}_0$) and $y = \operatorname{Exp}(b) \in K^0_{\mathbb{R}}$. We will use the notation $\mathcal{D}_{\mathbf{a}}(\alpha)$ for the probability of sampling $\alpha \in K$ from $\mathcal{D}_{\mathbf{a}}$.

$$\mathcal{D}_{\mathbf{a}}(\alpha) = \mathop{\mathbb{P}}_{x \leftarrow \mathcal{D}_{\mathbf{a}}}[x = \alpha].$$

10.3. **Shifting Property.** In order to prove the shifting property of Algorithm 2, we first start by the following lemma, that shows what impact multiplication by an element $\alpha \in K^*$ has on a (τ-equivalent) ideal lattice $\operatorname{Exp}(\mathbf{a})_\tau$. This will then be used to show that the output distribution of Algorithm 2 'shifts' (i.e., multiplies) by α if the input divisor is shifted by the principal divisor $\llparenthesis \alpha \rrparenthesis$ in Lemma 10.3.

Lemma 10.2. *For all $\mathbf{a} \in \operatorname{Div}^0_{K^{\mathfrak{m}}}$, multiplication by $\alpha \in K^{\mathfrak{m},1}$ gives an additive group isomorphism*

$$\operatorname{Exp}(\mathbf{a})_\tau \to \operatorname{Exp}(\mathbf{a} + \llparenthesis \alpha \rrparenthesis_{\mathfrak{f}})_\tau, \ \gamma \mapsto (\sigma(\alpha))_\sigma \cdot \gamma$$

and additionally, restricted to $r\mathcal{B}_\infty$, a bijection

$$\operatorname{Exp}(\mathbf{a})_\tau \cap r\mathcal{B}_\infty \to \operatorname{Exp}(\mathbf{a} + \llparenthesis \alpha \rrparenthesis_{\mathfrak{f}})_\tau \cap \operatorname{Exp}(-\llparenthesis \alpha \rrparenthesis_\infty) \cdot r\mathcal{B}_\infty.$$

Proof. Very similar as in Lemma 7.5 part (i), multiplication by $(\sigma(\alpha))_\sigma \in K_{\mathbb{R}}$ yields a bijection from $\operatorname{Exp}(\mathbf{a})_\tau$ to $\operatorname{Exp}(\mathbf{a} + \llparenthesis \alpha \rrparenthesis)_\tau \operatorname{Exp}(-\llparenthesis \alpha \rrparenthesis_\infty) = \operatorname{Exp}(\mathbf{a} + \llparenthesis \alpha \rrparenthesis_{\mathfrak{f}})_\tau$. Multiplication by an element is straightforwardly a group morphism, so the first claim follows by the fact that bijective group morphisms are isomorphisms.

For the second part, observe that multiplication by $(\sigma(\alpha))_\sigma$ transforms the box $r\mathcal{B}_\infty$ into $(\sigma(\alpha))_\sigma \cdot r\mathcal{B}_\infty = (|\sigma(\alpha)|)_\sigma \cdot r\mathcal{B}_\infty = \operatorname{Exp}_\infty(-\llparenthesis \alpha \rrparenthesis_\infty) \cdot r\mathcal{B}_\infty$. □

Lemma 10.3. *For all degree-zero Arakelov divisors $\mathbf{a} \in \operatorname{Div}^0_{K^{\mathfrak{m}}}$, the distribution $\mathcal{D}_{\mathbf{a}}$ satisfies the following* shifting property *for all $\alpha \in K^{\mathfrak{m},1}$,*

$$\mathcal{D}_{\mathbf{a}+\llparenthesis \alpha \rrparenthesis}(- \cdot \alpha) = \mathcal{D}_{\mathbf{a}}(-)$$

Proof. Take an arbitrary $\mathbf{a} \in \operatorname{Div}^0_{K^{\mathfrak{m}}}$ and write $\mathbf{a} = d^0(\mathfrak{b}) + b \in \operatorname{Div}^0_{K^{\mathfrak{m}}}$ for an ideal $\mathfrak{b}$ and an $b = (b_\nu)_\nu \in H$ (where we see $b \in H \subseteq \operatorname{Div}^0_{K^{\mathfrak{m}}}$). To prove the statement it is enough to show for every $\beta \in K^{\mathfrak{m}}$

$$\mathcal{D}_{\mathbf{a}+\llparenthesis \alpha \rrparenthesis}(\beta \cdot \alpha) = \mathcal{D}_{\mathbf{a}}(\beta)$$

[9] This is interchangeable, as the ideal $\mathfrak{b} = \operatorname{Exp}(\mathbf{a}_{\mathfrak{f}})$ and element $b = \mathbf{a} - d^0(\mathfrak{b})$ are readily retrieved (where $b = \operatorname{Log}(y)$). It is important to note, however, that $\operatorname{Exp}(\mathbf{a}) \neq y \cdot \mathfrak{b}$. Instead, $\operatorname{Exp}(\mathbf{a}) \cdot \mathcal{N}(\mathfrak{b})^{1/n} = y \cdot \mathfrak{b}$. We chose for this convention to make $\mathbf{a}$ a degree-zero Arakelov divisor.

As can be proved in the same fashion as in Lemma 9.8, the distribution $\mathcal{D}_{\mathbf{a}}$ is the same as the distribution resulting from

$$\alpha \leftarrow \operatorname{Exp}_\infty(-(\mathbf{a}+\mathbf{b})_\infty) \cdot \mathcal{U}(\operatorname{Exp}(\mathbf{a}+\mathbf{b})_\tau \cap r\mathcal{B}_\infty) \text{ with } \mathbf{b} = d^0(\prod_j \mathfrak{p}_j) + a,$$

where $\mathfrak{p}_j$ are sampled according to step 2 and $a = \operatorname{Log}((A_\sigma)_\sigma)$ is sampled from a discrete Gaussian, as in step 4. Note that this a is slightly distorted, due to the computation of $(A_\sigma)_\sigma$, but that is not problematic; the property used here is that this a is distributed independently of the value of $\mathbf{a}$. We write $\mathbf{b} = d^0(\prod_j \mathfrak{p}_j) + a$. This divisor corresponds to the 'random walk part' of the algorithm.

We seek to compare the distributions $\mathcal{D}_{\mathbf{a}}$ and $\mathcal{D}_{\mathbf{a}+(\!|\alpha|\!)}$. By the law of total probability, since the sampling of $\mathbf{b} = d^0(\prod_j \mathfrak{p}_j) + a$ is independent of $\mathbf{a}$ and $(\!|\alpha|\!)$, it is sufficient to consider a fixed sample $\mathbf{b} = d^0(\prod_j \mathfrak{p}_j) + a$ and compare the resulting distributions. These resulting distributions, for a fixed $\mathbf{b}$, denoted $\mathcal{D}_{\mathbf{a}|\mathbf{b}}$ and $\mathcal{D}_{\mathbf{a}+(\!|\alpha|\!)|\mathbf{b}}$, are

$$\mathcal{D}_{\mathbf{a}|\mathbf{b}} = \operatorname{Exp}_\infty(-(\mathbf{b}+\mathbf{a})_\infty) \cdot \mathcal{U}(\operatorname{Exp}(\mathbf{b}+\mathbf{a})_\tau \cap r\mathcal{B}_\infty) \tag{82}$$

and

$$\begin{aligned}
\mathcal{D}_{\mathbf{a}+(\!|\alpha|\!)|\mathbf{b}} &= \operatorname{Exp}_\infty(-(\mathbf{b}+\mathbf{a}+(\!|\alpha|\!))_\infty) \cdot \mathcal{U}(\operatorname{Exp}(\mathbf{b}+\mathbf{a}+(\!|\alpha|\!))_\tau \cap r\mathcal{B}_\infty) & (83)\\
&= \operatorname{Exp}_\infty(-(\mathbf{b}+\mathbf{a})_\infty) \cdot \mathcal{U}(\operatorname{Exp}(\mathbf{b}+\mathbf{a}+(\!|\alpha|\!)_{\mathrm{f}})_\tau \cap \operatorname{Exp}_\infty(-(\!|\alpha|\!)_\infty) \cdot r\mathcal{B}_\infty) & (84)\\
&= \operatorname{Exp}_\infty(-(\mathbf{b}+\mathbf{a})_\infty) \cdot \mathcal{U}\Big((\sigma(\alpha))_\sigma \cdot (\operatorname{Exp}(\mathbf{b}+\mathbf{a})_\tau \cap r\mathcal{B}_\infty)\Big) & (85)\\
&= (\sigma(\alpha))_\sigma \cdot \mathcal{D}_{\mathbf{a}|\mathbf{b}} & (86)
\end{aligned}$$

where the first equality holds by definition, the second equality follows from pulling $\operatorname{Exp}_\infty(-(\!|\alpha|\!)_\infty)$ into the uniform distribution and using the relation $(\!|\alpha|\!) = (\!|\alpha|\!)_{\mathrm{f}} + (\!|\alpha|\!)_\infty$, the third equality follows from Lemma 10.2 and the last equality by Lemma 9.8.

The result quickly follows from the law of total probability and the fact that multiplying by $(\sigma(\alpha))_\sigma$ and α is the same in $K_\mathbb{R}$. □

Lemma 10.4 (Post-selection shifting lemma)**.** *Let $\mathcal{S}$ be a set of ideals coprime with $\mathfrak{m}_0$ and let $\mathcal{D}^{\mathcal{S}}_{\mathbf{a}}$ be the distribution obtained by running Algorithm 2 on input $\mathbf{a}$ and only outputting β if $\beta/\bar{\mathfrak{b}} \in \mathcal{S}$ and otherwise outputting $\perp$.*

Then, this distribution satisfies

$$\mathcal{D}^{\mathcal{S}}_{\mathbf{a}+(\!|\alpha|\!)}(- \cdot \alpha) = \mathcal{D}^{\mathcal{S}}_{\mathbf{a}}(-),$$

for all $\alpha \in K^{\mathfrak{m},1}$.

Proof. This uses the same proof structure as Lemma 10.3, again using Lemma 9.8. Reusing the notation $\mathcal{D}_{\mathbf{a}|\mathbf{b}}$ for $\mathbf{b} = d^0(\prod_j \mathfrak{p}_j) + a$ as in Lemma 10.3, we proved that, for a fixed sample $\mathbf{b}$, we have

$$\mathcal{D}_{\mathbf{a}|\mathbf{b}}(-) = \mathcal{D}_{\mathbf{a}+(\!|\alpha|\!)|\mathbf{b}}(\alpha \cdot -). \tag{87}$$

The ideal $\beta/\bar{\mathfrak{b}}$ (where $\beta, \bar{\mathfrak{b}}$ are from Algorithm 2 on input $\mathbf{a}$) is distributed as[10] $\operatorname{Exp}_{\mathrm{f}}(-\mathbf{a}_{\mathrm{f}}) \cdot \gamma$ with $\gamma \leftarrow \mathcal{D}_{\mathbf{a}|\mathbf{b}}$ (on input $\mathbf{a}$ with fixed $\mathbf{b}$).

[10] The distribution $\mathcal{D}_{\mathbf{a}|\mathbf{b}}$ concerns the right part $\alpha\operatorname{Exp}_\infty(-\mathbf{a}_\infty)$ of Lemma 9.8, whereas $\beta/\bar{\mathfrak{b}}$ concerns $\alpha \operatorname{Exp}(-\mathbf{a}) = \alpha\operatorname{Exp}_\infty(-\mathbf{a}_\infty)\operatorname{Exp}_{\mathrm{f}}(-\mathbf{a}_{\mathrm{f}})$.

But replacing $\mathbf{a}$ with $\mathbf{a} + \llparenthesis \alpha \rrparenthesis$ here yields that the distribution of $\beta/\bar{\mathfrak{b}}$ of Algorithm 2 on input $\mathbf{a} + \llparenthesis \alpha \rrparenthesis$ is equal to (for a fixed sample $\mathbf{b}$)

$$\operatorname{Exp}_{\mathrm{f}}(-(\mathbf{a} + \llparenthesis \alpha \rrparenthesis)_{\mathrm{f}}) \cdot \gamma \text{ with } \gamma \leftarrow \mathcal{D}_{\mathbf{a}+\llparenthesis \alpha \rrparenthesis|\mathbf{b}}$$

which is, by Equation (87), equal to

$$\operatorname{Exp}_{\mathrm{f}}(-(\mathbf{a} + \llparenthesis \alpha \rrparenthesis)_{\mathrm{f}}) \cdot \alpha \cdot \gamma \text{ with } \gamma \leftarrow \mathcal{D}_{\mathbf{a}|\mathbf{b}}$$

which is, by manipulation, equal to

$$\operatorname{Exp}_{\mathrm{f}}(-\mathbf{a}_{\mathrm{f}}) \cdot \gamma \text{ with } \gamma \leftarrow \mathcal{D}_{\mathbf{a}|\mathbf{b}}.$$

So, for a fixed sample of $\mathbf{b} = d^0(\prod_j \mathfrak{p}_j) + a$ the probabilities of obtaining $\beta/\bar{\mathfrak{b}}$ are the same, no matter whether the input was $\mathbf{a}$ or $\mathbf{a} + \llparenthesis \alpha \rrparenthesis$.

As a result, $\mathcal{D}^{\mathcal{S}}_{\mathbf{a}|\mathbf{b}} = \alpha \cdot \mathcal{D}^{\mathcal{S}}_{\mathbf{a}+\llparenthesis \alpha \rrparenthesis|\mathbf{b}}$, where $\mathcal{D}^{\mathcal{S}}_{\mathbf{a}|\mathbf{b}}$ denotes the output distribution of Algorithm 2 on input $\mathbf{a}$ with fixed sample $\mathbf{b} \leftarrow \mathcal{W}(N, B, s)$ and the additional post-selection (i.e., outputting β whenever $\beta/\bar{\mathfrak{b}} \in \mathcal{S}$ and $\perp$ otherwise).

By the law of total probability the distributions of $\beta/\bar{\mathfrak{b}}$ for $\mathbf{a}$ or $\mathbf{a} + \llparenthesis \alpha \rrparenthesis$ are the same; and, thus the rejection of $\beta/\bar{\mathfrak{b}}$ for not being in $\mathcal{S}$ happens for the same occurrences. Therefore, $\mathcal{D}^{\mathcal{S}}_{\mathbf{a}+\llparenthesis \alpha \rrparenthesis}(- \cdot \alpha) = \mathcal{D}^{\mathcal{S}}_{\mathbf{a}}(-)$ for all $\alpha \in K^{\mathfrak{m},1}$. □

10.4. **Boundedness Property.** In this section we show that the output distribution $\mathcal{D}_{\mathbf{a}}$ is bounded, i.e., no arbitrarily large inputs occur. This can be seen by the fact that the output is sampled from a distorted box, where the distortion is distributed as a truncated discrete Gaussian (a bounded distribution by itself). Thus, the output must be confined to a certain box, slightly larger than the dimensions of the original undistorted box. This is made more formal in the following text.

Recall the Euclidean distance notion on Div_K in Section 2.8.

Lemma 10.5. *Let $\mathbf{a} = d^0(\mathfrak{b}) + b \in \operatorname{Div}^0_{K^{\mathfrak{m}}}$ with $\mathfrak{b} \in \mathcal{I}_K$ and $b = \operatorname{Log}(y) \in H$ and let B, N, s, r, ε be as defined in Algorithm 2. Then, we have, for $\beta \leftarrow \mathcal{D}_{\mathbf{a}}$,*

$$\| \llparenthesis \beta \rrparenthesis \| \le 5 \log(B^N \cdot r^n) + \|\mathbf{a}\| + s \cdot \sqrt{n \log(8n^2/\varepsilon)}.$$

Proof.

Bound on the norm of β.

We start with a bound on $|\mathcal{N}(\beta)|$, which is of use for the rest of the proof. The output of $\beta \leftarrow \mathcal{D}_{\mathbf{a}}$ with $\mathbf{a} = d^0(\mathfrak{b}) + b$ lies in the box $\operatorname{Exp}(-a-b) \cdot r \cdot \mathcal{N}(\bar{\mathfrak{b}})^{1/n} \cdot \mathcal{B}_\infty \subseteq K_{\mathbb{R}}$ (see lines 6 and 7 in Algorithm 2), where $a = (a_\nu)_\nu = \operatorname{Log}((A_\sigma)_\sigma) \in H$ is an approximation a sample of a (truncated) Gaussian distribution (see line 4 and 5 of Algorithm 2); and $b = (b_\nu)_\nu = \operatorname{Log}(y)$. Hence, since $a, b \in H$ do not change the norm (we may assume this by Lemma 9.4), we obtain $|\mathcal{N}(\beta)| \le \mathcal{N}(\bar{\mathfrak{b}}) \cdot r^n \le B^N \cdot r^n \cdot \mathcal{N}(\mathfrak{b})$; a bound on the norm of β.

Split the bound on $\|\llparenthesis \beta \rrparenthesis\|$ into three parts.

We have, by the triangle inequality,

$$\|\llparenthesis \beta \rrparenthesis\| \le \|\llparenthesis \beta \rrparenthesis - \mathbf{a}\| + \|\mathbf{a}\|. \tag{88}$$

We seek to bound $\|\llparenthesis \beta \rrparenthesis\|$. To do so, we concentrate on $\|\llparenthesis \beta \rrparenthesis - \mathbf{a}\|$, which we can split in a 'finite part', a 'constant part' and an 'infinite part'.

$$\llparenthesis \beta \rrparenthesis - \mathbf{a} = \underbrace{\sum_{\mathfrak{p}} m_{\mathfrak{p}} \cdot \llparenthesis \mathfrak{p} \rrparenthesis}_{\text{finite part } \mathbf{f}} \underbrace{- \sum_{\nu} \frac{n_\nu \log \mathcal{N}(\prod_{\mathfrak{p}} \mathfrak{p}^{m_{\mathfrak{p}}})}{n} \llparenthesis \nu \rrparenthesis}_{\text{constant part } \mathbf{c}} + \underbrace{\sum_{\nu} \iota_\nu \llparenthesis \nu \rrparenthesis}_{\text{infinite part } \mathbf{i}} . \tag{89}$$

where $n_\nu = 2$ if ν is complex and 1 otherwise.

Bounding the finite part f and constant part c.

Since $\beta \in \bar{\mathfrak{b}} \subseteq \mathfrak{b}$, we see that $(\beta) = \mathfrak{a} \cdot \mathfrak{b}$ for an *integral ideal* $\mathfrak{a}$ (recall that $\mathfrak{b} = \operatorname{Exp}(\mathbf{a}_{\mathrm{f}})$). Hence, $\sum_{\mathfrak{p}} m_{\mathfrak{p}} \cdot d^0(\mathfrak{p}) = d^0(\mathfrak{a})$, has only positive coefficients $m_{\mathfrak{p}} \geq 0$. Since $\mathcal{N}(\mathfrak{a}) = |\mathcal{N}(\beta)|/\mathcal{N}(\mathfrak{b}) \leq B^N \cdot r^n$ by the norm bound on β in the very beginning of this proof, we quickly deduce

$$B^N \cdot r^n \geq \mathcal{N}(\mathfrak{a}) \geq \prod_{\mathfrak{p}} \mathcal{N}(\mathfrak{p})^{m_{\mathfrak{p}}} \geq 2^{\sum_{\mathfrak{p}} |m_{\mathfrak{p}}|} \geq 2^{(\sum_{\mathfrak{p}} |m_{\mathfrak{p}}|^2)^{1/2}},$$

which implies

$$\|\mathbf{f}\| \leq (\sum_{\mathfrak{p}} |m_{\mathfrak{p}}|^2)^{1/2} \leq \log(B^N \cdot r^n)/\log(2) \leq 2\log(B^N \cdot r^n) \tag{90}$$

Similarly, for the constant part $\mathbf{c}$, using $\mathfrak{a} = \prod_{\mathfrak{p}} \mathfrak{p}$, we have,

$$\|\mathbf{c}\| \leq \sqrt{\sum_{\nu} n_\nu^2} \cdot \frac{\log(\mathcal{N}(\mathfrak{a}))}{n} \leq \log(\mathcal{N}(\mathfrak{a})) \leq \log(B^N \cdot r^n) \tag{91}$$

Bounding the infinite part i.

By definition (see Equation (89)), and by the inclusion of $H \subseteq \operatorname{Div}^0_{K^{\mathfrak{m}}}$ we have

$$\iota_\nu = -n_\nu \log|\sigma_\nu(\beta)| + n_\nu \frac{\log(|\mathcal{N}(\beta)|)}{n} - b_\sigma \in H \tag{92}$$

since $\mathbf{a} = d^0(\mathfrak{b}) + b$. By the fact that β lies in the box $\operatorname{Exp}(-a-b) \cdot r \cdot \mathcal{N}(\bar{\mathfrak{b}})^{1/n} \cdot \mathcal{B}_\infty$, we have $\log|\sigma_\nu(\beta)| \leq -n_\nu^{-1} a_\sigma - n_\nu^{-1} b_\sigma + \log r + \log(\mathcal{N}(\bar{\mathfrak{b}})^{1/n})$. Therefore,

$$-\log|\sigma(\beta)| \geq n_\nu^{-1} a_\sigma + n_\nu^{-1} b_\sigma - \log r - \log(\mathcal{N}(\bar{\mathfrak{b}})^{1/n}) \tag{93}$$

Hence, substituting Equation (93) into Equation (92), and using that $\beta \in \bar{\mathfrak{b}}$ (and hence $|\mathcal{N}(\beta)| \geq \mathcal{N}(\bar{\mathfrak{b}})$), we obtain

$$\iota_\nu \geq a_\nu + n_\nu \log(|\mathcal{N}(\beta)|^{1/n} \cdot \mathcal{N}(\bar{\mathfrak{b}})^{-1/n} \cdot r^{-1}) \geq a_\nu - n_\nu \log(r).$$

So, writing $\iota - a \in H$, it satisfies $\sum_\nu (\iota - a)_\nu = 0$ and $\iota_\nu - a_\nu \geq -n_\nu \log(r)$ for all ν. Then $\|\iota - a\| = \sqrt{\sum_\nu |\iota_\nu - a_\nu|^2}$ is bounded by[11] $2\sqrt{n(n-1)}\log(r) \leq 2\log(r^n)$. Hence, writing $\mathbf{i} = (\iota_\nu)_\nu$, we have

$$\|\mathbf{i}\| = \|\iota\| \leq \|a\| + 2\log(r^n) \leq \|a\| + 2\log(B^N \cdot r^n). \tag{94}$$

[11] Suppose a sequence $(x_1, \ldots, x_n)$ satisfies $\sum_{j=1}^n x_j = 0$ and $x_j \geq -t$ for all $j \in \{1, \ldots, n\}$ for some $t \in \mathbb{R}_{>0}$. Then $\sum_{j=1}^n x_j^2 \leq n(n-1)t^2$. This maximum is attained by, for example, $x_1 = (n-1)t$ and $x_j = -t$ for $1 < j \leq n$.

We prove this by induction, where the case $n = 1$ is trivial (since then $x_1^2 = x_1 = 0 \leq n(n-1)t^2$). For $n > 1$, we can assume that a non-zero sequence $(x_1, \ldots, x_n)$ has a positive co-ordinate (by the zero-sum requirement). Moreover, we can assume it has only a single positive co-ordinate. Indeed, if there are multiple positive coordinates, say x_1, x_2, then the sequence $(y_1, \ldots, y_{n-1}) = (x_1 + x_2, x_3, \ldots, x_n)$ satisfies $\sum_{j=1}^{n-1} y_j = 0$ and $y_j \geq -t$. Hence, $\sum_{j=1}^n x_j^2 \leq \sum_{j=1}^{n-1} y_j^2 \leq (n-1)(n-2)t^2 \leq n(n-1)t^2$, by induction. For a single positive co-ordinate it is clear that $x_1 = (n-1)t$ and $x_j = -t$ for $1 < j \leq n$ is the optimal solution.

Applying this to the sequence $q_\sigma = n_\nu^{-1}(\iota_{\nu_\sigma} - a_{\nu_\sigma})$, we see that $\|\iota - a\| \leq 2\|(q_\sigma)_\sigma\| \leq 2\sqrt{n(n-1)}\log(r)$.

Combining the bounds on $\|\mathbf{i}\|$ and $\|\mathbf{f}\|$.
Concluding, by combining Equations (88) to (91) and (94), we obtain

$$\begin{aligned} \|(\!|\beta|\!)\| &\le \|(\!|\beta|\!) - \mathbf{a}\| + \|\mathbf{a}\| \le \|\mathbf{f}\| + \|\mathbf{c}\| + \|\mathbf{i}\| + \|\mathbf{a}\| && (95) \\ &\le 5\log(B^N \cdot r^n) + \|a\| + \|\mathbf{a}\|. && (96) \end{aligned}$$

The final bound then comes from the fact that $a = (a_\nu)_\nu \in H$ is (an approximation of a) discrete Gaussian distributed with deviation s computed according to the algorithm in Lemma 2.22 with $\varepsilon_{\mathcal{G}} := \varepsilon/4$ (see 4 of Algorithm 2). Hence, $\|a\| \le s \cdot \sqrt{n \log(8n^2/\varepsilon)}$. □

10.5. **(Almost)-Lipschitz Property.** The almost-Lipschitz-continuous property of the output distribution of Algorithm 2 implies that it does not matter too much if the input parameter $y = \mathrm{Exp}(b) \in K^0_{\mathbb{R}}$ is slightly inaccurate or rounded to a vector of rational numbers due to machine precision.

This is relevant for Part 2 of this paper, where we will feed Algorithm 2 $y = \mathrm{Exp}((b_\nu)_\nu)$ with a continuous Gaussian distributed $b \in H$. The Lipschitz-continuous property in this subsection essentially implies that using a discrete Gaussian distribution instead for $b \in H$ will only cause a very small error overhead.

Lemma 10.6. *The output distribution of the sampling algorithm (Algorithm 2) is ε-almost Lipschitz continuous in the input parameter $b \in H$ with $y = \mathrm{Exp}(b)$. That is, for $\mathbf{a} = d^0(\mathfrak{b}) + b$ and $\mathbf{a}' = d^0(\mathfrak{b}) + b' = \mathbf{a} + (b' - b)$ we have*

$$\tfrac{1}{2}\|\mathcal{D}_{\mathbf{a}} - \mathcal{D}_{\mathbf{a}'}\|_1 \le \frac{n^2}{2} \cdot \|b - b'\| + \varepsilon,$$

where $\frac{1}{2}\|\mathcal{D}_{\mathbf{a}} - \mathcal{D}_{\mathbf{a}'}\|_1$ is the total variation distance of the two distributions.

Proof. In this proof, like in the proof of Theorem 9.5, we will again assume that line 4 in Algorithm 2 is replaced by the line in Equation (63), and assume that $y \in K^0_{\mathbb{R}}$. Again we will refer to this as the 'continuous version of Algorithm 2'. This proof will be structured as follows. First we will show that this 'continuous version of Algorithm 2' is actually Lipschitz continuous with Lipschitz constant $n^2/2$. After that we will use Proposition 9.12 to show that the Lipschitz continuity also 'almost' holds for the ordinary version of Algorithm 2, with an additional error of ε.

In the following, we will reason about the 'continuous version of Algorithm 2'. The input $b \in H$ is only used in Algorithm 2 (as $b = \mathrm{Log}(y)$) in combination with the variable $a = (a_\nu)_\nu \in H$, which sampled according to a continuous Gaussian distribution with deviation $s = 1/n^2$. The sum $a + b$ is then used combined in the rest of the algorithm.

So, for two different inputs $b, b' \in H$, the only thing that differs in the algorithm are the distributions of $a + b$ and $a' + b'$, for which a, a' are (independently) sampled according to the same Gaussian distribution.

We denote the distributions respectively $\mathcal{G}_b$ and $\mathcal{G}_{b'}$; these are Gaussian distributions with deviation $s = 1/n^2$ centered at b and b' respectively (see Section 2.9).

Therefore, by the data processing inequality (Theorem 2.16), one immediately deduces that the statistical distance between the output distribution of the 'continuous version of Algorithm 2' on input $\mathfrak{b}, b$ and $\mathfrak{b}, b'$ is bounded above by

$$\tfrac{1}{2}\|\mathcal{G}_{H,b} - \mathcal{G}_{H,b'}\|_1.$$

The total variation distance of two continuous (multivariate) Gaussian distributions with different centers can be bounded by the Kullback-Leibler divergence [61, Ch. 1, Ex. 11],

$$\begin{aligned}\tfrac{1}{2}\|\mathcal{G}_{H,b} - \mathcal{G}_{H,b'}\|_1 &\leq \tfrac{1}{2}\sqrt{\tfrac{1}{2}D_{KL}(\mathcal{G}_{H,b} \parallel \mathcal{G}_{H,b'})} \\ &\leq \tfrac{1}{2}\sqrt{\tfrac{1}{4}\|b-b'\|^2/s^2} = \frac{\|b-b'\|}{4\cdot s} = \frac{n^2\cdot\|b-b'\|}{4}\end{aligned}$$

The last equality is obtained by instantiating $s = 1/n^2$.

As sketched in the introduction of this proof, we use that the ordinary version and the continuous version of Algorithm 2 only differ by $\varepsilon/2$, by the very same argument as in the proof of Theorem 9.5. Hence, the statistical distance between the ordinary version of Algorithm 2 input $\mathfrak{b}, b$ and $\mathfrak{b}, b'$ is bounded by $\frac{n^2}{2}\cdot\|b-b'\|+\varepsilon$, as required. □

In most use-cases, Algorithm 2 is *repeated until success*, for which the following lemma is more useful: it shows that the output distribution of the algorithm *conditioned on a successful outcome* is also almost-Lipschitz. This distribution conditioned on a successful outcome is exactly the same as the output distribution resulting from repeating the algorithm until success.

Lemma 10.7. *The output distribution $\mathcal{D}_{\mathbf{a}}^{suc}$, resulting from repeating the sampling algorithm (Algorithm 2) until success, is $\frac{\varepsilon}{\delta_{\mathcal{S}}[r^n]}$-almost Lipschitz continuous in the input parameter $b \in H$ with $y = \operatorname{Exp}(b)$. That is, for $\mathbf{a} = d^0(\mathfrak{b}) + b$ and $\mathbf{a}' = d^0(\mathfrak{b}) + b' = \mathbf{a} + (b' - b)$ we have*

$$\tfrac{1}{2}\|\mathcal{D}_{\mathbf{a}}^{suc} - \mathcal{D}_{\mathbf{a}'}^{suc}\|_1 \leq \frac{n^2}{2\delta_{\mathcal{S}}[r^n]}\cdot\|b-b'\| + \frac{\varepsilon}{\delta_{\mathcal{S}}[r^n]},$$

where $\frac{1}{2}\|\mathcal{D}_{\mathbf{a}}^{suc} - \mathcal{D}_{\mathbf{a}'}^{suc}\|_1$ is the total variation distance of the two distributions.

Proof. For this we apply the next Lemma 10.8 with the lower bound $\delta_{\mathcal{S}}[r^n]$ on the success probability. □

Lemma 10.8. *Let $\mathcal{A}$ be a randomized algorithm that can have input y and y'. Denote $\mathcal{A}_y$ respectively $\mathcal{A}_{y'}$ for the algorithm using y respectively y' as input. Assume that both $\mathcal{A}_y$ and $\mathcal{A}_{y'}$ have success probability at least $p \in (0,1]$.*

Now denote $\mathcal{B}_y$ for the algorithm running $\mathcal{A}_y$ until it outputs a successful output, and similarly denote $\mathcal{B}_{y'}$ for the algorithm running $\mathcal{A}_{y'}$ until success. By abuse of notation, denote $\mathcal{B}_y$, $\mathcal{B}_y$, $\mathcal{A}_y$, $\mathcal{A}_{y'}$ for the output distributions of these algorithms. Then the total variation distances are related by

$$\tfrac{1}{2}\|\mathcal{B}_y - \mathcal{B}_{y'}\|_1 \leq \frac{\frac{1}{2}\|\mathcal{A}_y - \mathcal{A}_{y'}\|_1}{p}.$$

Proof. Denote $X \cup \{\perp\}$ for the output space of $\mathcal{A}$, where $\perp$ denotes failure (unsuccessful output) and where $x \in X$ are all successful outputs.

The output distribution $\mathcal{B}_x$ is the output distribution of $\mathcal{A}_x$ *conditioned* on a successful output. Hence, denoting $\mathcal{A}_x^{\text{success}}$ (and similarly for x') for this conditional distribution, we have

$$\|\mathcal{B}_x - \mathcal{B}_{x'}\|_1 = \|\mathcal{A}_x^{\text{success}} - \mathcal{A}_{x'}^{\text{success}}\|_1 = \sum_{x\in X}|\mathcal{A}_y^{\text{success}}[x] - \mathcal{A}_{y'}^{\text{success}}[x]|,$$

where X is the probability space of the algorithm $\mathcal{A}$. The conditional probability of $\mathcal{A}$ outputting x *conditioned* on a successful output (which are all of $x \in X$) then must equal $\frac{\mathcal{A}_y[x]}{p_0}$ with $p_0 = \mathbb{P}[\mathcal{A}_y \text{ gives successful output}]$; and similarly for input y'. Note that $|p_0 - p_1| \leq \|\mathcal{A}_y - \mathcal{A}_{y'}\|_1$. Hence, writing $p = \min(p_0, p_1)$ and noting that $p_1 = \sum_{x \in X} \mathcal{A}_{y'}[x]$,

$$\begin{aligned}
&\sum_{x \in X} |\mathcal{A}_y^{\text{success}}[x] - \mathcal{A}_{y'}^{\text{success}}[x]| = \sum_{x \in X} \left| \frac{\mathcal{A}_y[x]}{p_0} - \frac{\mathcal{A}_{y'}[x]}{p_1} \right| \\
\leq &\sum_{x \in X} \left| \frac{\mathcal{A}_y[x]}{p_0} - \frac{\mathcal{A}_{y'}[x]}{p_0} \right| + \sum_{x \in X} \left| \frac{\mathcal{A}_{y'}[x]}{p_0} - \frac{\mathcal{A}_{y'}[x]}{p_1} \right| \\
\leq &\frac{1}{p_0} \sum_{x \in X} |\mathcal{A}_y[x] - \mathcal{A}_{y'}[x]| + \frac{1}{p_0}|p_1 - p_0| = \frac{1}{p_0} \sum_{x \in X \cup \{\perp\}} |\mathcal{A}_{y'}[x] - \mathcal{A}_{y'}[x]| \\
\leq &\frac{\|\mathcal{A}_y - \mathcal{A}_{y'}\|_1}{p}.
\end{aligned}$$

□

11. Estimation of the factor $\mathcal{N}(\mathfrak{m}_0)/\phi(\mathfrak{m}_0)$

Lemma 11.1. *Let K be a number field and let $\mathfrak{m}_0 \subseteq \mathcal{O}_K$ be an integral ideal. Then we have*

$$\frac{\mathcal{N}(\mathfrak{m}_0)}{\phi(\mathfrak{m}_0)} = \prod_{\mathfrak{p} | \mathfrak{m}_0} \frac{1}{1 - \mathcal{N}(\mathfrak{p})^{-1}}.$$

where $\phi(\mathfrak{m}_0) = |K^{\mathfrak{m}_0}/K^{\mathfrak{m}_0,1}|$ and where $\mathfrak{p}$ ranges over the prime ideals of K dividing $\mathfrak{m}_0$.

Proof. If $\mathfrak{m}_0 = \mathcal{O}_K$, we have $\phi(\mathfrak{m}_0) = 1$. Otherwise $\phi(\mathfrak{m}_0) = |(\mathcal{O}_K/\mathfrak{m}_0)^\times|$. Writing $\mathfrak{m}_0 = \prod_{\mathfrak{p}|\mathfrak{m}_0} \mathfrak{p}^{v_\mathfrak{p}(\mathfrak{m}_0)}$, we have, by multiplicativity of the norm $\mathcal{N}$ and the (generalized) Euler totient function ϕ (which follows from the Chinese remainder theorem),

$$\mathcal{N}(\mathfrak{m}_0) = \prod_{\mathfrak{p}|\mathfrak{m}_0} \mathcal{N}(\mathfrak{p})^{v_\mathfrak{p}(\mathfrak{m}_0)} \quad \text{and} \quad \phi(\mathfrak{m}_0) = \prod_{\mathfrak{p}|\mathfrak{m}_0} \mathcal{N}(\mathfrak{p})^{v_\mathfrak{p}(\mathfrak{m}_0)-1}(\mathcal{N}(\mathfrak{p}) - 1)$$

The quotient $\mathcal{N}(\mathfrak{m}_0)/\phi(\mathfrak{m}_0)$ then equals $\prod_{\mathfrak{p}|\mathfrak{m}_0} \frac{\mathcal{N}(\mathfrak{p})}{\mathcal{N}(\mathfrak{p})-1} = \prod_{\mathfrak{p}|\mathfrak{m}_0} \frac{1}{1-\mathcal{N}(\mathfrak{p})^{-1}}$. □

Proposition 11.2 (ERH)**.** *Let K be a number field and let $\mathfrak{m}_0 = \prod_{\mathcal{N}(\mathfrak{p})<x} \mathfrak{p}$ be the product of all prime ideals with norm below $x \geq 10$. Then there exists $c_0 \in [-8, 8]$ and $c_1 \in [0, 2]$ such that*

$$\frac{\mathcal{N}(\mathfrak{m}_0)}{\phi(\mathfrak{m}_0) \cdot \rho_K} = \log(x) \cdot \exp\left(c_1 + c_0 \cdot \frac{\log|\Delta_K| + n \log x}{\sqrt{x}}\right).$$

Proof. Write, as in Bach's paper [5, §6],

$$A(x) := \prod_{p<x} \frac{1 - p^{-1}}{\prod_{\substack{\mathfrak{p}|p \\ \mathcal{N}(\mathfrak{p})<x}} (1 - \mathcal{N}(\mathfrak{p})^{-1})}, \tag{97}$$

for which we have the bound[12] [5, Thm. 6.2 & Table 2]

$$|\log(\rho_K) - \log A(x)| \leq 8 \cdot \left(\frac{\log|\Delta| + n \log x}{\sqrt{x}} \right), \tag{98}$$

assuming the Riemann Hypothesis for the Dedekind zeta function of K.

By Lemma 11.1 and Equation (97), we have

$$\frac{\mathcal{N}(\mathfrak{m}_0)}{\phi(\mathfrak{m}_0)} = \prod_{\mathfrak{p}|\mathfrak{m}_0} \frac{1}{1 - \mathcal{N}(\mathfrak{p})^{-1}} = \prod_{p<x} \frac{1}{1-p^{-1}} \cdot A(x) \tag{99}$$

By taking logarithms of Equation (99) and subsequently applying Bach's bound (Equation (98)), we obtain

$$\log\left(\tfrac{\mathcal{N}(\mathfrak{m}_0)}{\phi(\mathfrak{m}_0)}\right) - \log(\rho_K) = \log(\prod_{p<x} \frac{1}{1-p^{-1}}) + \log A(x) - \log(\rho_K) \tag{100}$$

$$= \log(\prod_{p<x} \frac{1}{1-p^{-1}}) + c_0 \cdot \left(\frac{\log|\Delta| + n\log x}{\sqrt{x}} \right), \tag{101}$$

for some $c_0 \in [-8, 8]$, by Equation (98). Apply Lemma 11.3 for the left-hand side of Equation (101), to obtain $\prod_{p<x} \frac{1}{1-p^{-1}} \in [e^0, e^2] \cdot \log x$. Exponentiating the expression yields the final result. □

Lemma 11.3. *For $x \geq 2$, we have*

$$\log x \leq \prod_{p<x} \frac{1}{1-p^{-1}} \leq 6 \cdot \log x,$$

where the product ranges over all prime numbers up to x.

Proof. By the Euler product formula, we certainly have

$$\prod_{p<x} \frac{1}{1-p^{-1}} \geq \sum_{n<x} \frac{1}{n} \geq \log x.$$

For the upper bound, we invoke an explicit version of Mertens' third theorem of Rosser and Schoenfeld [67, Corollary 1, Equation 3.30],

$$\prod_{p<x} \frac{1}{1-p^{-1}} < \log(x) \cdot e^{\gamma} \cdot (1 + \frac{1}{\log(x)^2}).$$

Here, $\gamma \leq 0.578$ is the Euler-Mascheroni constant. Instantiating with $x = 2$ and explicitly computing e^{γ} yields a bound of $6 \cdot \log(x)$. □

The following result is a simplified version of a result by Grenié–Molteni [37, Cor. 1.4].

Lemma 11.4 (ERH, Grenié–Molteni)**.** *Let K be a number field, with degree n, discriminant Δ_K, and ring of integers $\mathcal{O}_K$. Let $\mathfrak{m}_0 = \prod_{\mathcal{N}(\mathfrak{p})<x} \mathfrak{p}$ be the product of all prime ideals with norm below $x > 100$. We have*

$$\begin{aligned}\log \mathcal{N}(\mathfrak{m}_0) &\leq x + \sqrt{x}\left(\left(\frac{\log x}{2\pi} + 2 \right) \log|\Delta_K| + \left(\frac{(\log x)^2}{8\pi} + 2 \right) n \right) \\ &= x + O\left(\sqrt{x}\log(x)\left(\log|\Delta_K| + n\log x\right)\right).\end{aligned}$$

[12] In his theorem, Bach writes $\frac{\zeta_K}{\zeta}(1)$ for the residue ρ_K at $s = 1$ of the Dedekind zeta function ζ_K [5, Thm. 6.2].

PART 2

Rigorous computation of class groups and unit groups

12. Introduction

In this second part of the article, we present the first algorithm for computing class groups and unit groups of arbitrary number fields that provably runs in probabilistic subexponential time, assuming the Extended Riemann Hypothesis. Let K be a number field of degree n and discriminant Δ_K. The determination of the structure of its class group $\mathrm{Cl}(K)$, together with a system of fundamental units, is one of the main problems of computational number theory [22, p. 217]. Previous subexponential algorithms were either restricted to imaginary quadratic fields, or relied on several heuristic assumptions that have long resisted rigorous analysis.

12.1. **The Log-$\mathbb{S}$-unit lattice.** While we stated our main result as an algorithm for computing units and class groups in Theorem 1.3, our algorithm actually does slightly more than that: it computes the so-called Log-$\mathbb{S}$-unit lattice for any finite set $\mathbb{S}$ of prime ideals. For such a set $\mathbb{S}$ of prime ideals of K, we consider the group $\mathrm{Div}_{K,\mathbb{S}} = \prod_{\mathfrak{p}\in\mathbb{S}} \mathbb{Z} \times \prod_{\nu} \mathbb{R}$, where ν ranges over all infinite places of K. The Log-$\mathbb{S}$-unit lattice $\mathrm{Log}_{\mathbb{S}}(\mathcal{O}_{K,\mathbb{S}}^{\times})$ is a lattice living in $\mathrm{Div}_{K,\mathbb{S}}$ generated by all vectors of the form $(v, -\mathrm{Log}(\alpha))$ with $\mathrm{Log} : K^{\times} \to \prod_{\nu} \mathbb{R}$ the logarithmic embedding of K, where $\alpha \in K$ and $v = (v_{\mathfrak{p}})_{\mathfrak{p}\in\mathbb{S}} \in \mathbb{Z}^{\mathbb{S}}$ are such that $\alpha \cdot \mathcal{O}_K = \prod_{\mathfrak{p}\in\mathbb{S}} \mathfrak{p}^{v_{\mathfrak{p}}}$. Such elements α are called $\mathbb{S}$-units, hence the name of the lattice. See Section 15 for more details. The main result of this part is the following theorem.

Theorem 12.1 (ERH)**.** *There is a probabilistic algorithm which, on input a number field K of degree n and discriminant Δ_K, an LLL-reduced basis of its ring of integers $\mathcal{O}_K$ and a finite set of prime ideals $\mathbb{S}$, computes the Log-$\mathbb{S}$-unit lattice and runs in expected time polynomial in the length of the input, in $L_{|\Delta_K|}(1/2)$, in $L_{n^n}(2/3)$, and in $\min(\rho_K, L_{|\Delta_K|}(2/3 + o(1)))$, where ρ_K is the residue at 1 of the Dedekind zeta function ζ_K.*

The Log-$\mathbb{S}$-unit lattice computed by the algorithm is represented exactly by elements $\alpha_i \in K$ and vectors $v_i \in \mathbb{Z}^{\mathbb{S}}$ such that the vectors $((v_i, -\mathrm{Log}\,\alpha_i))_i$ form a basis of $\mathrm{Log}_{\mathbb{S}}(\mathcal{O}_{K,\mathbb{S}}^{\times})$. These elements $\alpha_i \in K$ are represented by a so-called compact representation, i.e., as a product of smaller elements $\beta_j \in K$ (see Section 2.6). Even though this is called a compact representation (and it is indeed more compact than representing the α_i in a basis of $\mathcal{O}_K$), the bit-size of this representation of the α_i might be about as large as the running time of the algorithm.

12.2. **Previous work.** Shanks proposed in 1968 the first algorithm to compute the structure of the class group. His method, described for quadratic number fields, had an exponential running time $O_{\varepsilon}(|\Delta_K|^{1/4+\varepsilon})$ (or $O_{\varepsilon}(|\Delta_K|^{1/5+\varepsilon})$ assuming ERH). In

1989, Hafner and McCurley [38] proposed the first subexponential time probabilistic algorithm to compute the class group of imaginary quadratic number fields. Assuming ERH, they prove that the expected running time is $L_{|\Delta_K|}(1/2, \sqrt{2})$. The case of imaginary quadratic number fields distinguishes itself by the finiteness of the unit group, and the existence of a *reduced* ideal in each class, two properties that proved helpful for a rigorous analysis of the algorithm.

In 1990, Buchmann [17] generalized this algorithm to fields of arbitrary degree. The analysis proved much more delicate, requiring several heuristic assumptions, notably on the distribution of smooth ideals. From these assumptions, Buchmann argued that the expected running time is $L_{|\Delta_K|}(1/2, 1.7)$ when the degree of the field is constant. Practical improvements ensued, notably by Cohen, Diaz Y Diaz and Olivier [25]. In 2014, Biasse and Fieker [10] proposed an algorithm capable of dealing with fields of varying degree: under heuristic assumptions, they argue that the algorithm computes class groups in subexponential time $L_{|\Delta_K|}(2/3 + \varepsilon)$ for arbitrary families of number fields, even with growing degree (and $L_{|\Delta_K|}(1/2)$ whenever $n \leq (\log |\Delta_K|)^{3/4-\varepsilon}$, matching Buchmann's claimed complexity for fixed-degree fields). In the meantime, no progress was made towards a rigorous analysis.

On the quantum computing side, a non-heuristic (assuming ERH) quantum polynomial time algorithm for computing $\mathbb{S}$-units and class groups was developed in 2016 by Biasse and Song [11], for any number field. The structure of this algorithm is different from the classical algorithms mentioned above: it relies on a quantum polynomial time algorithm solving the Continuous Hidden Subgroup Problem over $\mathbb{R}^m$ [30, 13]. In the present article, we only consider classical algorithms.

12.3. **The blueprint of a 'classical' Log-$\mathbb{S}$-unit lattice algorithm.** For the sake of simplicity we limit ourselves in this overview to the computation of the class group of a number field. The situation for the Log-$\mathbb{S}$-unit lattice is, at this high level, very similar.

Classical algorithms computing the class group proceed by generating many class group relations of the shape $[\prod_{\mathfrak{p}} \mathfrak{p}^{a_\mathfrak{p}}] = 1$, where the prime factors $\mathfrak{p}$ in the ideal $\prod_{\mathfrak{p}} \mathfrak{p}^{a_\mathfrak{p}}$ have norm bounded by B, i.e., the product is a B-smooth ideal. Since there are only finitely many generators $\mathfrak{p}$, enough 'sufficiently independent' relations of this shape then yield a description of the (finite) class group.

Such a relation $[\prod_{\mathfrak{p}} \mathfrak{p}^{a_\mathfrak{p}}] = 1$ is generally found by taking a random smooth ideal $\mathfrak{s} = \prod_{\mathfrak{p}} \mathfrak{p}^{b_\mathfrak{p}}$, and repeatedly sampling random elements $\alpha \in \mathfrak{s}$ until $(\alpha)\mathfrak{s}^{-1} = \prod_{\mathfrak{p}} \mathfrak{p}^{c_\mathfrak{p}}$ is *also* a smooth ideal. This gives the relation $1 = [\alpha] = [\prod_{\mathfrak{p}} \mathfrak{p}^{a_\mathfrak{p}}]$ with $a_\mathfrak{p} = b_\mathfrak{p} + c_\mathfrak{p}$.

Heuristics in previous work. The usual heuristics for class group algorithms (and Log-$\mathbb{S}$-unit lattice algorithms) are twofold. The first kind of heuristic assumption concerns a lower bound on the probability of the random ideal $(\alpha)\mathfrak{s}^{-1}$ to be smooth, see for instance [9, Heuristic 1, 2]. This allows one to bound the expected running time of sampling a single relation of the shape $[\prod_{\mathfrak{p}} \mathfrak{p}^{a_\mathfrak{p}}] = 1$. This smoothness probability is typically assumed to match the *density* of smooth ideals in the set of all ideals, a somewhat well-understood quantity related to the Dickman function.

The second kind of heuristic assumption concerns the number of such relations required in order to get the class group (rather than a larger group, of which the class group would be a quotient), see for instance [9, Heuristic 3]. The heuristic running time is then obtained as the product of the running time per relation and the required number of relations.

12.4. **Why our algorithm is rigorous.** We would like to stress that the following overview of our proof makes substantial simplifications for expository purposes (in particular, we ignore complex embeddings, which in the full proof play a similar role to prime ideals).

To circumvent the first heuristic about the cost of finding one relation, we use a special *sampling algorithm* (Algorithm 2) from Part 1 of this paper. This algorithm samples an element $\alpha \in \mathfrak{a}$ in such a way that the success probability of $(\alpha)\mathfrak{a}^{-1}$ being smooth is *actually provably lower bounded* by the density of smooth ideals in the set of all ideals (up to a multiplicative constant). This yields an upper bound on the expected cost of finding one class group relation. The sampling algorithm is the object of Part 1; this algorithm, together with its properties is used here as a black-box.

The remaining challenge addressed in the current part is circumventing the *second heuristic*, concerning the number of sampled group relations $[\prod_{\mathfrak{p}} \mathfrak{p}^{a_{\mathfrak{p}}}] = 1$ required to compute the class group (rather than an extension). To achieve that goal, we investigate the lattice of class group relations, which consists of those vectors $(a_{\mathfrak{p}})_{\mathfrak{p}}$ for which $[\prod_{\mathfrak{p}} \mathfrak{p}^{a_{\mathfrak{p}}}] = 1$.

The class group lattice. Denoting $\mathbb{S}$ for the set of prime ideals with norm bounded by B, we can construct the class group lattice $L_{\mathrm{Cl}} = \{(a_{\mathfrak{p}})_{\mathfrak{p}\in\mathbb{S}} \in \mathbb{Z}^{\mathbb{S}} \mid [\prod_{\mathfrak{p}\in\mathbb{S}} \mathfrak{p}^{a_{\mathfrak{p}}}] = 1\} \subseteq \mathbb{Z}^{\mathbb{S}}$. In this paper we will construct (here treated quite informally) a randomized algorithm $\mathcal{A} : \mathbb{Z}^{\mathbb{S}} \to \mathbb{Z}^{\mathbb{S}}$ which has the following two properties.

(i) For $u \leftarrow \mathcal{A}(z)$, we have $u + z \in L_{\mathrm{Cl}}$ and $\|v\| < E$ for some bound E.
(ii) The distribution of $\mathcal{A}(z)$ only depends on the coset $z + L_{\mathrm{Cl}}$, i.e., $\mathcal{A}(z')$ and $\mathcal{A}(z)$ have the same distribution for any $z' \in z + L_{\mathrm{Cl}}$.

This algorithm $\mathcal{A}$ uses the sampling algorithm from Part 1 mentioned above. We will use algorithm $\mathcal{A}$ to sample vectors of L_{Cl} in such a way that we can prove that they will generate the full lattice with sufficiently good probability.

Gaussians over the class group lattice. Let $\mathcal{G}_{X,s}$ denote the discrete Gaussian distribution with parameter s over the discrete set $X \subseteq \mathbb{R}^{\mathbb{S}}$. We will make use of the following observation: when the deviation s is large enough, sampling from $\mathcal{G}_{\mathbb{Z}^{\mathbb{S}},s}$ is indistinguishable from sampling first a uniform $c \in \mathbb{Z}^{\mathbb{S}}/L_{\mathrm{Cl}}$, then sampling from $c + \mathcal{G}_{L_{\mathrm{Cl}},s}$.

For such $z \leftarrow c + \mathcal{G}_{L_{\mathrm{Cl}},s}$ for $c \in \mathbb{Z}^{\mathbb{S}}/L_{\mathrm{Cl}}$ uniformly distributed, we see by property (ii) of $\mathcal{A}$ that

$$\mathcal{A}(z) + z \sim \mathcal{A}(c) + (c + \mathcal{G}_{L_{\mathrm{Cl}},s}) = (\mathcal{A}(c) + c) + \mathcal{G}_{L_{\mathrm{Cl}},s}$$

By property (i) of $\mathcal{A}$, the distribution $\mathcal{A}(c)+c$ is *bounded*, i.e., for all $z \leftarrow \mathcal{A}(c)+c$ we have $\|z\| < E'$ with overwhelming probability for some bounded E' (which depends on E and on the covering radius of the lattice L_{Cl}). Then the variable $\mathcal{A}(z) + z$ is distributed as a Gaussian over L_{Cl} with some *independent 'noise'* of size E' caused by $\mathcal{A}(c)+c$. By taking the standard deviation s (of all discrete Gaussians involved) much larger than this 'noise' of size E', one can deduce that, for $z \leftarrow \mathcal{G}_{\mathbb{Z}^{\mathbb{S}},s}$,

$$\mathcal{A}(z) + z \approx \mathcal{G}_{L_{\mathrm{Cl}},s}.$$

Summarizing, sampling $z \leftarrow \mathcal{G}_{\mathbb{Z}^{\mathbb{S}},s}$ with large enough s and computing $\mathcal{A}(z) + z$ allows to compute *close to Gaussian samples* from the class group lattice L_{Cl}. There are well-known bounds for the number of samples required to generate the entire lattice L_{Cl} for such (close to) Gaussian samples.

The algorithm $\mathcal{A}$. The algorithm $\mathcal{A} : \mathbb{Z}^{\mathbb{S}} \to \mathbb{Z}^{\mathbb{S}}$ satisfying the properties (i) and (ii), is defined as follows. For $z \in \mathbb{Z}^{\mathbb{S}}$, compute $\mathfrak{a} = \prod_{\mathfrak{p} \in \mathbb{S}} \mathfrak{p}^{z_{\mathfrak{p}}}$. Sample $\alpha \in \mathfrak{a}$ according to a not-too-wide discrete Gaussian and put $\mathfrak{b} = (\alpha) \cdot \mathfrak{a}^{-1}$. Then, use the sampling algorithm from Part 1 to find a representative $\prod_{\mathfrak{p} \in \mathbb{S}} \mathfrak{p}^{u_{\mathfrak{p}}} \in [\mathfrak{b}] = [\mathfrak{a}]^{-1} = [\prod_{\mathfrak{p} \in \mathbb{S}} \mathfrak{p}^{-z_{\mathfrak{p}}}]$, and output $u \in \mathbb{Z}^{\mathbb{S}}$. We have that $u + z \in L_{\mathrm{Cl}}$ and that u is bounded (by properties of the sampling algorithm), so property (i) of algorithm $\mathcal{A}$ is true.

Property (ii) follows from the fact that the sample $\mathfrak{b} = (\alpha) \cdot \mathfrak{a}^{-1}$ only depends on the class $[\mathfrak{a}]$ of $\mathfrak{a}$, and not of the representation thereof[13]. Therefore, the output u only depends on the class, i.e., the coset $z + L_{\mathrm{Cl}}$ of the input z.

Technicalities involving the infinite places. In the general case of the Log-$\mathbb{S}$-unit lattice, there is also a *continuous* part involving the logarithmic unit lattice. Essentially the same proof structure applies, with additional technicalities introduced by the numerical approximation of the real numbers involved.

12.5. **More algorithmic problems.** Once we have an algorithm computing the Log-$\mathbb{S}$-unit lattice for any set of prime ideals $\mathbb{S}$, it is well known that one can use it to obtain other quantities, such as the units of the number field, or the structure of the class group. Combined with Theorem 17.7, which allows to decompose any integral ideal as an equivalent product of prime ideals in a sufficiently large set $\mathbb{S}$, this can also be used to solve other algorithmic problems, such as the principal ideal problem, or the class group discrete logarithm problem.

Units. To obtain the units of a number field K, one computes the Log-$\mathbb{S}$-unit lattice with an empty set $\mathbb{S}$. The output elements α_i of the algorithm is then a fundamental system of units of the number field. Note that these elements α_i, even when they are represented in compact representation (see Section 2.6), might have a large bit-size (as large as the running time of the algorithm). Note also that computing a fundamental system of units of K only describes the units modulo the roots of unity of K. Computing the roots of unity of K can be done efficiently using, e.g., [22, Section 4.9.4], and is not related to the algorithm we describe here.

Class group. In order to obtain the structure of the class group, one chooses a set $\mathbb{S}$ generating the class group. It is known that under the extended Riemann hypothesis, there are such sets $\mathbb{S}$ of size polynomial in $\log |\Delta_K|$ [4]. Given a basis (α_i, v_i) of the Log-$\mathbb{S}$-unit lattice, one then only keeps the rank $|\mathbb{S}|$ lattice L generated by the vectors v_i. The class group of K is then isomorphic to $\mathbb{Z}^{\mathbb{S}}/L$. Note that L is an integral lattice whose determinant is the class-number h_k of K. Hence, the HNF basis of L has bit-size polynomial in $|\mathbb{S}|$ and $\log h_k$, which is polynomial in $\log |\Delta_K|$. This means that the output size of the algorithm computing the class group is polynomially bounded in $\log |\Delta_K|$.

[13] This is actually false when considering the class group alone. To apply this 'independence of representation' technique, the infinite places (i.e., complex embeddings) must be included in the logarithmic embedding. The Log-$\mathbb{S}$-unit lattice accounts for the infinite places, so this technique applies.

Class group discrete logarithm. The class group discrete logarithm problem asks, given as input any integral ideal $\mathfrak{a}$ and a set of prime ideals $\mathbb{S}$, to compute $\alpha \in K$ and $v \in \mathbb{Z}^{\mathbb{S}}$ such that $\mathfrak{a} = \alpha \cdot \mathcal{O}_K \cdot \prod_{\mathfrak{p} \in \mathbb{S}} \mathfrak{p}^{-v_{\mathfrak{p}}}$. Solving the class group discrete logarithm problem is exactly what is done by Algorithm 3, provided that the set $\mathbb{S}$ contains sufficiently many ideals of small norm. In order to solve the class group discrete logarithm problem for smaller sets $\mathbb{S}$, one can set $\mathbb{S}'$ to be the minimal set containing $\mathbb{S}$ and sufficiently many small prime ideals so that Theorem 17.7 applies. One can then solve the class group discrete logarithm problem in $\mathbb{S}'$ and compute the log-$\mathbb{S}'$-unit lattice. Solving the class group discrete logarithm in the smaller set $\mathbb{S}$ can then be performed efficiently by linear algebra, using the knowledge of the log-$\mathbb{S}'$-unit lattice to replace primes of $\mathbb{S}' \setminus \mathbb{S}$ by a product of primes of $\mathbb{S}$.

Principal ideal problem. The principal ideal problem is the problem of computing a generator α of a principal integral ideal $\mathfrak{a}$ (represented by an HNF basis). Again, this can be solved by choosing a set $\mathbb{S}'$ containing sufficiently many ideals of small norm and solving the class group discrete logarithm problem with respect to $\mathfrak{a}$ and this set $\mathbb{S}'$ using Algorithm 3. Then, computing the log-$\mathbb{S}'$-unit lattice allows one to solve the principal ideal problem by linear algebra. The size of the computed generator α, even in compact representation (see Section 2.6), might be as large as the running time of the algorithm.

12.6. **Road map.** We start in Section 13 by specializing the ideal sampling theorem of Part 1 to our needs. Then, a lower bound is given on the density of smooth ideals, by means of a combinatorial technique, in Section 14. After that, some preliminaries on $\mathbb{S}$-units are treated in Section 15, and useful properties of $\mathbb{S}$-unit lattices are analyzed in Section 16.

Applying both the specialized sampling theorem and the density of the smooth ideals, we obtain an algorithm that samples a *single* $\mathbb{S}$-unit in Section 17.

In order to have an algorithm that outputs sufficiently 'random' $\mathbb{S}$-units, as to generate the entire $\mathbb{S}$-unit group, we combine two results. One is a bound on the 'generating radius' $\varrho(\mathrm{Log}_{\mathbb{S}}(\mathcal{O}_{K,\mathbb{S}}^{\times}))$ of the log-$\mathbb{S}$-unit lattice that is polynomial in $\log|\Delta_K|$, which is proven in Section 16. This is combined with the properties of the sampling algorithm (see Section 10 in Part 1), to build an algorithm that indeed outputs 'sufficiently random' $\mathbb{S}$-units, as proved in Section 18.

After gathering many such $\mathbb{S}$-units, we would like to compute a set of *fundamental* $\mathbb{S}$*-units*, which can be seen as the multiplicative analogue of a basis of the log-$\mathbb{S}$-unit lattice. This is done in Section 19 by 'post-processing' the many $\mathbb{S}$-units, which essentially consists in applying the Buchmann-Kessler-Pohst algorithm.

Part 2 is then concluded in Section 21 with the final theorem on the complexity of the $\mathbb{S}$-unit algorithm.

13. Specialized ideal sampling algorithm

In this section, we specialize the main result from Part 1 to $G = \mathrm{Pic}^0_{K^{\mathfrak{m}}}$.

Definition 13.1. *We denote by $\mathcal{S}_B$ the set of B-smooth ideals, i.e.,*

$$\mathcal{S}_B = \{\mathfrak{a} \text{ integral ideal of } \mathcal{O}_K \mid \text{ for any prime factor } \mathfrak{p} \text{ of } \mathfrak{a} \text{ we have } \mathcal{N}(\mathfrak{p}) \leq B\}$$

We also extend the notion of smoothness to sets $\mathbb{S}$ of prime ideals: we say that $\mathfrak{a} \subseteq \mathcal{O}_K$ is $\mathbb{S}$-smooth if all its prime factors belong to $\mathbb{S}$: $\mathcal{S}_{\mathbb{S}}$ is the set of $\mathbb{S}$-smooth ideals. Recall that the local density of a set of ideals $\mathcal{S}$ is defined (Definition 2.11) as

$$\delta_{\mathcal{S}}[x] = \min_{t\in[x/e^n,x]} \frac{|\mathcal{S}(t)|}{\rho_K \cdot t} = \min_{t\in[x/e^n,x]} \frac{|\{\mathfrak{b} \in \mathcal{S} \mid \mathcal{N}(\mathfrak{b}) \leq t\}|}{\rho_K \cdot t},$$

where, $\rho_K = \lim_{s\to 1}(s-1)\zeta_K(s)$ (see Equation (4)).

Theorem 13.2 below is a specialization of Theorem 9.5 from Part 1, together with the properties proved in Section 10. We will only use this specialization in this part.

Theorem 13.2 (ERH)**.** *There is a randomized algorithm* `Sample` *such that the following holds. Let K be a number field, with degree n, discriminant Δ_K, and ring of integers $\mathcal{O}_K$ of which an LLL-reduced basis is given. Let $\mathfrak{m}_0 \subseteq \mathcal{O}_K$ be an ideal modulus of which the prime ideal factorization is known, let $\mathfrak{a} \in \mathcal{I}_K$ be an ideal coprime to $\mathfrak{m}_0$, and let $y \in K_{\mathbb{R}}^{\times}$ be represented by rational numbers. Let $\mathsf{b} \geq 2$ be an integer, let $\omega \in \mathbb{Q}_{\geq 1}$, and let $r = 48 \cdot \omega \cdot \mathsf{b}^{2n/\mathsf{b}} \cdot n^{7/2} \cdot |\Delta_K|^{\frac{3}{2n}} \cdot \mathcal{N}(\mathfrak{m}_0)^{\frac{1}{n}}$.*
Given the above data, the algorithm `Sample`$(\mathfrak{a}, y, \mathfrak{m}_0, \mathsf{b}, \omega)$ *outputs $\beta \in \mathfrak{a}$ coprime to $\mathfrak{m}_0$ such that for any set $\mathcal{S}$ of ideals in $\mathcal{O}_K$ coprime to $\mathfrak{m}_0$, we have $(\beta)\cdot\mathfrak{a}^{-1} \in \mathcal{S}\cdot\mathcal{S}_{B_{\mathrm{rw}}}$ with probability at least*

$$\frac{\mathcal{N}(\mathfrak{m}_0)}{\phi(\mathfrak{m}_0)} \frac{\delta_{\mathcal{S}}[r^n]}{6},$$

for some smoothness bound $B_{\mathrm{rw}} = \mathsf{poly}(\log|\Delta_K|, \log\mathcal{N}(\mathfrak{m}_0))$, and where $\phi(\mathfrak{m}_0) = |(\mathcal{O}_K/\mathfrak{m}_0)^{\times}|$. Furthermore, the algorithm `Sample` *runs in time*

$$\mathsf{poly}(\mathrm{size}(\mathfrak{a}), \mathrm{size}(y), \log(\mathcal{N}(\mathfrak{m}_0)), \mathsf{b}^{\mathsf{b}}, \log|\Delta_K|, \mathrm{size}(\omega)).$$

Additionally, the output distribution $\mathcal{D}_{y,\mathfrak{a}}$ of `Sample` *on input $(y, \mathfrak{a}) \in (K_{\mathbb{R}}^{\times}, \mathcal{I}_K)$ (with $\mathfrak{m}_0, \mathsf{b}$ and ω implicit) satisfies the following properties.*

(1) (Almost Lipschitz-continuous). For all $y, y' \in K_{\mathbb{R}}^{\times}$,

$$\|\mathcal{D}_{y,\mathfrak{a}} - \mathcal{D}_{y',\mathfrak{a}}\|_1 \leq \frac{n^2}{2}\|\operatorname{Log}(y) - \operatorname{Log}(y')\| + \frac{1}{1200\cdot|\Delta_K|\cdot r^n}.$$

(2) (Shifting property). For all $\beta \in K$ and $\alpha \in K^{\mathfrak{m}_0,1}$, we have the following identity of distributions:

$$\mathcal{D}_{y,\mathfrak{a}}[-] = \mathcal{D}_{y\cdot(|\sigma(\alpha)^{-1}|)_{\sigma},\mathfrak{a}\cdot(\alpha)}[\alpha\cdot -]$$

(3) (Bounded). Writing $\llparenthesis\beta\rrparenthesis \in \operatorname{Div}_K^0$ and $\|\cdot\|$ for the norm on Div_K, we have

$$\|\llparenthesis\beta\rrparenthesis\| \leq \|d(\mathfrak{a}) + \operatorname{Log}(y)\| + O(\log^2|\Delta_K| + \log\mathcal{N}(\mathfrak{m}_0) + n\log(\omega))$$

(4) (Independent of $|\mathcal{N}(y)|$). $\mathcal{D}_{y,\mathfrak{a}} = \mathcal{D}_{y^0,\mathfrak{a}}$, where $y^0 = y/|\mathcal{N}(y)|^{1/n}$.

Remark 13.3. *In the case where $\mathfrak{m}_0 = \mathcal{O}_K$, we get $\mathcal{N}(\mathfrak{m}_0) = \phi(\mathfrak{m}_0) = 1$, yielding a lower bound $\delta_{\mathbb{S}}[r^n]/6$ on the success probability.*

Proof. The algorithm `Sample` is Algorithm 2 from Part 1 instantiated with modulus $\mathfrak{m} := \mathfrak{m}_0$, subgroup $G := \operatorname{Pic}^0_{K^{\mathfrak{m}}}$ (yielding $[\operatorname{Pic}^0_{K^{\mathfrak{m}}} : G] = 1$), ideal $\mathfrak{b} := \mathfrak{a}$, $y := y \in K_{\mathbb{R}}^{\times}$, block size $\mathsf{b} := \mathsf{b}$ and element $\tau \in (\mathcal{O}_K/\mathfrak{m}_0)^{\times}$ uniformly random[14] if $\mathfrak{m}_0 \neq \mathcal{O}_K$ and $\tau = 1$ if $\mathfrak{m}_0 = \mathcal{O}_K$ (note that $\tau K^{\mathfrak{m},1} = K^{\mathfrak{m},1} = K^*$ if $\mathfrak{m} = \mathfrak{m}_0 = \mathcal{O}_K$). We put

[14] Uniform sampling of $\tau \in (\mathcal{O}_K/\mathfrak{m}_0)^{\times}$ can be done in time $\mathsf{poly}(\log\mathcal{N}(\mathfrak{m}_0), \log|\Delta_K|)$ by sampling uniformly in $(\mathcal{O}_K/\mathfrak{q})^{\times}$ for all prime power divisors $\mathfrak{q} \mid \mathfrak{m}_0$ and using the Chinese Remainder theorem. This uses that the factorization of $\mathfrak{m}_0$ is assumed to be known.

the error parameter $\varepsilon := \frac{1}{1200 \cdot |\Delta_K| r^n} \leq \frac{1}{6 \cdot |\Delta_K| \cdot r^n} \leq \frac{1}{6 \cdot \rho_K \cdot r^n}$ (see Equation (5)). We will use Theorem 9.5.

Note that we may assume that $\delta_{\mathcal{S}}[r^n] = \min_{t \in [r^n/e^n, r^n]} \frac{|\{\mathfrak{b} \in \mathcal{S} \mid \mathcal{N}(\mathfrak{b}) \leq t\}|}{\rho_K \cdot t} > 0$, since otherwise the statement of the theorem is inherently true, as any probability is at least zero. Hence, the set $\{\mathfrak{b} \in \mathcal{S} \mid \mathcal{N}(\mathfrak{b}) \leq r^n/e^n\}$ is nonempty, and we can immediately deduce

$$\delta_{\mathcal{S}}[r^n] = \min_{t \in [r^n/e^n, r^n]} \frac{|\{\mathfrak{b} \in \mathcal{S} \mid \mathcal{N}(\mathfrak{b}) \leq t\}|}{\rho_K \cdot t} \geq \frac{1}{\rho_K \cdot r^n} \geq 6 \cdot \varepsilon. \tag{102}$$

By Theorem 9.5, the success probability of Algorithm 2 with these parameters is at least

$$\frac{\mathcal{N}(\mathfrak{m}_0)}{\phi(\mathfrak{m}_0)} \frac{\delta_{\mathcal{S}}[r^n]}{3} - \varepsilon \geq \frac{\mathcal{N}(\mathfrak{m}_0)}{\phi(\mathfrak{m}_0)} \frac{\delta_{\mathcal{S}}[r^n]}{3} - \frac{\delta_{\mathcal{S}}[r^n]}{6} \geq \frac{\mathcal{N}(\mathfrak{m}_0)}{\phi(\mathfrak{m}_0)} \frac{\delta_{\mathcal{S}}[r^n]}{6}.$$

where the first inequality holds by Equation (102) and the second one by the fact that $\mathcal{N}(\mathfrak{m}_0) \geq \phi(\mathfrak{m}_0)$. This proves the lower bound on the success probability of algorithm $\mathcal{A}$ in the theorem.

It remains to show the bound on B_{rw} and the running time. Note that $[\mathrm{Pic}^0_{K^{\mathfrak{m}}} : G] = 1$, $\mathrm{size}(\tau) = 1$, and that

$$\log(1/\varepsilon) = O(\log(|\Delta_K| r^n)) = O(\log^2 |\Delta_K| + \log \mathcal{N}(\mathfrak{m}_0) + \log \omega)$$

by direct computation. Hence

$$B_{\mathrm{rw}} = \mathsf{poly}(\log |\Delta_K|, \log \mathcal{N}(\mathfrak{m}_0), \log \omega)$$

and we have a running time of

$$T = \mathsf{poly}(\mathsf{b}^{\mathsf{b}}, \mathrm{size}(\mathfrak{a}), \mathrm{size}(y), \log(\mathcal{N}(\mathfrak{m}_0)), \log |\Delta_K|, \mathrm{size}(\omega)),$$

with no queries to any oracle, since $G = \mathrm{Pic}^0_{K^{\mathfrak{m}}}$.

The properties (1)-(3), directly follow from the almost-Lipschitz continuity (Lemma 10.6), the shifting property (Lemma 10.3) and the boundedness property (Lemma 10.5) of Algorithm 2, with the following extra remarks. For the almost-Lipschitz continuity, we use that $\| \mathrm{Log}(y^0) - \mathrm{Log}(z^0) \| \leq \| \mathrm{Log}(y) - \mathrm{Log}(z) \|$ for $y^0 = \frac{y}{|\mathcal{N}(y)|^{1/n}}$ and $z^0 = \frac{z}{|\mathcal{N}(z)|^{1/n}} \in K^0_{\mathbb{R}}$. For the Lipschitz error, we just substitute $\varepsilon = \frac{1}{1200 \cdot |\Delta_K| r^n}$. Here, $\mathrm{Log}(y), \mathrm{Log}(z) \in \mathrm{Log}(K_{\mathbb{R}}^{\times}) \hookrightarrow \mathrm{Div}_K$ inherit their 'natural' norm from the norm on Div_K (see Section 2.8).

For the boundedness property we use that

$$\log(B^N r^n) = O(\log^2 |\Delta_K| + \log \mathcal{N}(\mathfrak{m}_0) + n \log(\omega))$$

and $s = 1/n^2$ for the parameters used in Algorithm 2. Additionally, with the instantiation $\varepsilon = \frac{1}{1200 |\Delta_K| r^n}$ we have

$$s\sqrt{n \log(8n^2/\varepsilon)} \leq O(\log |\Delta_K| + \log(r)) \leq O(\log^2 |\Delta_K| + \log \mathcal{N}(\mathfrak{m}_0) + \log \omega).$$

With these simplifications, Lemma 10.5 tells us that $\|(\!|\beta|\!)\| \leq \|d^0(\mathfrak{a}) + \mathrm{Log}(y)\| + O(\log^2 |\Delta_K| + \log \mathcal{N}(\mathfrak{m}_0) + n \log(\omega))$. In order to obtain the desired bound (i.e., with $d(\mathfrak{a})$ instead of $d^0(\mathfrak{a})$), we use property (4) and we replace the input $(\mathfrak{a}, y)$ by $(\mathfrak{b}, z) := (\mathfrak{a}, y \cdot \mathcal{N}(\mathfrak{a})^{1/n})$, so that $d^0(\mathfrak{b}) + \mathrm{Log}(z) = d(\mathfrak{a}) + \mathrm{Log}(y)$.

For the shifting property, we use the fact that $d(\mathfrak{a}) + \mathrm{Log}(y) + (\!|\alpha|\!) = d(\mathfrak{a} \cdot (\alpha)) + \mathrm{Log}(y \cdot (|\sigma(\alpha)|^{-1})_\sigma) \in \mathrm{Div}_K$. For the last property, item (4), note that `Sample` does not depend on $|\mathcal{N}(y)|$, as follows from Lemma 9.4. □

Remark 13.4. *In the $\mathbb{S}$-unit computation algorithm in this paper, we assume that we are given an LLL-reduced basis of the ring of integers $\mathcal{O}_K$. In other words, we take such a basis of $\mathcal{O}_K$ as part of the input. We refer the reader to the discussion in Section 2.6.*

Remark 13.5. *Looking into the details of Part 1, the part b^{b} in the running time of Theorem 9.5 is caused by a b-BKZ reduction on an ideal lattice $\mathfrak{a}$ (see Section 9). One may wonder whether, in the context of this section, using an ideal-SVP algorithm specifically designed for ideals (e.g., [29, 63]), instead of the BKZ algorithm, might lead to better results. This is unfortunately not the case at the moment, since all current algorithms exploiting the special structure of ideal lattices are heuristic. Additionally, they all use an algorithm computing the class group and the unit group of K as a building block, which would be a circular requirement here.*

Remark 13.6. *Theorem 13.2 can also be made to work for arbitrary $\tau \in K^{\mathfrak{m}}$ (with an additional $\mathrm{size}(\tau)$ in the running time). Even a random τ is possible (which might be useful in certain use-cases), though then the factorization of $\mathfrak{m}_0$ must be known. Indeed, using this factorization, uniform sampling of $\tau \in (\mathcal{O}_K/\mathfrak{m}_0)^\times$ can be done in time $\mathsf{poly}(\log \mathcal{N}(\mathfrak{m}_0), \log|\Delta_K|)$ by sampling uniformly in $(\mathcal{O}_K/\mathfrak{q})^\times$ for all prime power divisors $\mathfrak{q} \mid \mathfrak{m}_0$ and using the Chinese Remainder theorem.*

14. Density of smooth ideals

From the early work of Hafner and McCurley [38] for imaginary quadratic fields, to the generalizations for larger degree fields [17, 10], the core of the strategy to compute class groups as been to find random relations between classes involving smooth ideals. The probability of a somewhat random ideal to be smooth thus plays a key role in the analysis of such algorithms.

For any $0 < B \leq x$, let $\Psi_K(x, B)$ be the number of B-smooth integral ideals of K of norm at most x. It is well known [75] that whenever $(\log\log x)^{5/3+\varepsilon} \leq \log B \leq \log x$, we have

$$\Psi_K(x,B) = x \cdot \rho_K \cdot \rho(u)\left(1 + O_{\varepsilon,K}\left(\frac{\log(u+1)}{\log B}\right)\right), \tag{103}$$

where ρ is the Dickman function, and $u = \frac{\log x}{\log B}$. One then needs to estimate the dependence in K hidden in the big O error. Unfortunately, scrutinizing the proof of [75], and using the best known uniform estimates of the ideal counting function [73], one finds that the hidden constant is exponential in the degree n of K. This means that for the estimate of Equation (103) to be non-vacuous, one needs the smoothness bound B to be doubly exponential in n.

It is not clear at the time whether this issue is unavoidable (i.e., the estimate $\Psi_K(x,B) \sim x \cdot \rho_K \cdot \rho(u)$ is truly only valid when B is large compared to the degree), or is an artifact of the techniques not being precise enough. Therefore, instead of this estimate, we rely on the following weaker but simpler combinatorial bound.

Lemma 14.1 (ERH)**.** *For any $\varepsilon > 0$, there exists a constant C such that the following holds. Let $B \geq B_{\mathrm{sm}} := C(n + \log|\Delta_K|)^{2+\varepsilon}$, and let $A \leq B/(4\log B)$. Let $\mathcal{S}_{A,B}$ be the set of integral ideals of $\mathcal{O}_K$ whose prime divisors have norm in $(A, B]$. Let $x \geq B \cdot e^n$, and $u = \frac{\log x}{\log B}$. Then the density at x of $\mathcal{S}_{A,B}$ satisfies*

$$\delta_{\mathcal{S}_{A,B}}[x] \geq \frac{(4\log B)^{1-u}}{\rho_K B} u^{-u}.$$

Proof. Let $y \in [x/e^n, x]$. Note that, by the assumption $x \geq B \cdot e^n$, we have $y \geq B$. Let $(\mathfrak{p}_i)_{i=1}^N$ be the list of all prime ideals in $\mathcal{O}_K$ of norm in $(A, B]$. We have

$$|\mathcal{S}_{A,B}(y)| = \left| \left\{ \prod_{i=1}^N \mathfrak{p}_i^{e_i} \,\middle|\, \sum_{i=1}^N e_i \log(\mathcal{N}(\mathfrak{p}_i)) \leq \log y \right\} \right| \geq \left| \left\{ (e_i) \in \mathbb{Z}_{\geq 0}^N \,\middle|\, \sum_{i=1}^N e_i \leq \frac{\log y}{\log B} \right\} \right|.$$

Writing $u_y = \frac{\log y}{\log B}$ and $v = \lfloor u_y \rfloor \geq 1$ (since $y \geq B$), we deduce

$$|\mathcal{S}_{A,B}(y)| \geq \binom{v+N-1}{v} \geq \left(\frac{v+N-1}{v}\right)^v \geq N^v v^{-v} \geq N^v u_y^{-u_y}.$$

There exists $D > 0$ depending only on ε such that for any $X > D$, we have $X^{1/(2+\varepsilon/2)} < X^{1/2}/(\log X)^2$. Then, by a result of Lagarias and Odlyzko [47], there is a constant[15] E (absolute) such that for any $X \geq \max(D, E(n + \log|\Delta_K|)^{2+\varepsilon/2}) =: F$, we have $\frac{X}{2\log X} \leq \pi_K(X) \leq X$. Let $A' = \max(A, F)$. Recall that $A \leq B/(4\log B)$. If furthermore $F \leq B/(4\log B)$, then, $A' \leq B/(4\log B)$, and we get

$$N = \pi_K(B) - \pi_K(A) \geq \frac{B}{2\log B} - \pi_K(A') \geq \frac{B}{2\log B} - A' \geq \frac{B}{4\log B}.$$

We now show that we can choose the constant C such that indeed $F \leq B/(4\log B)$ (hence the conditions of the lemma do imply $N \geq \frac{B}{4\log B}$). There exists $G > 0$ depending only on ε such that for any $X > G$, we have $X^{\frac{2+\varepsilon/2}{2+\varepsilon}} < X/(4\log X)$. Therefore, for any $\delta > 0$ and $B \geq \max(G, F^\delta)$, we have

$$B/(4\log B) \geq B^{\frac{2+\varepsilon/2}{2+\varepsilon}} \geq F^{\delta\frac{2+\varepsilon/2}{2+\varepsilon}}.$$

Choosing $\delta = \frac{2+\varepsilon}{2+\varepsilon/2}$, we get that $B/(4\log B) \geq F$ (as desired). Now, we have $F^\delta = \max(D^\delta, E^\delta(n + \log|\Delta_K|)^{2+\varepsilon})$. One can therefore choose the constant $C = \max(G, D^\delta, E^\delta)$, to enforce $B/(4\log B) \leq F$.

We have just proved that $N \geq \frac{B}{4\log B}$. Then, we have

$$N^v \geq N^{u_y - 1} \geq (y/B)(4\log B)^{1-u_y}.$$

We obtain $|\mathcal{S}_{A,B}(y)| \geq (y/B)(4\log B)^{1-u_y} u_y^{-u_y}$, hence, writing $u = u_x$, we have

$$\frac{|\mathcal{S}_{A,B}(y)|}{\rho_K \cdot y} \geq \frac{(4\log B)^{1-u_y}}{\rho_K B} u_y^{-u_y} \geq \frac{(4\log B)^{1-u}}{\rho_K B} u^{-u}.$$

□

15. Background on $\mathbb{S}$-unit lattices

15.1. **$\mathbb{S}$-units.** In this section, we formally define $\mathbb{S}$-units and the Log-$\mathbb{S}$-unit lattice, and recall some of its properties.

Definition 15.1 ($\mathbb{S}$-units)**.** *Let $\mathbb{S}$ be a set of prime ideals of $\mathcal{O}_K$. An element $\alpha \in K$ is an $\mathbb{S}$-unit if and only if $\alpha \cdot \mathcal{O}_K = \prod_{\mathfrak{p} \in \mathbb{S}} \mathfrak{p}^{v_\mathfrak{p}}$ for some $(v_\mathfrak{p})_{\mathfrak{p} \in \mathbb{S}} \in \mathbb{Z}^{\mathbb{S}}$. We write $\mathcal{O}_{K,\mathbb{S}}^\times \subset K$ for the set of $\mathbb{S}$-units.*

We will use the language of divisors introduced in Section 2.8.

[15] The original statement [47] (putting $L = K$ and $G = \{1\}$) reads $|\pi_K(X) - \mathrm{Li}(X)| = O\left(X^{1/2}\log(|\Delta_K|X^n) + \log|\Delta_K|\right)$, where the constants in the big-O are absolute, and Li is the logarithmic integral.

Definition 15.2 ($\mathbb{S}$-Divisor group)**.** *Let $\mathbb{S}$ be a (finite) set of prime ideals of $\mathcal{O}_K$. We define the $\mathbb{S}$-divisor group $\mathrm{Div}_{K,\mathbb{S}} \subseteq \mathrm{Div}_K$*

$$\mathrm{Div}_{K,\mathbb{S}} := \bigoplus_{\mathfrak{p}\in\mathbb{S}} \mathbb{Z} \times \bigoplus_{\nu} \mathbb{R},$$

where ν ranges over the set of all infinite places (embeddings into the complex numbers up to possible conjugation). Additionally, we define the degree-zero $\mathbb{S}$-divisor group $\mathrm{Div}^0_{K,\mathbb{S}} = \mathrm{Div}_{K,\mathbb{S}} \cap \mathrm{Div}^0_K$.

In the present part of the article, we often write elements of $\mathrm{Div}_{K,\mathbb{S}}$ as a vector $((a_\mathfrak{p})_{\mathfrak{p}\in\mathbb{S}}, (a_\nu)_\nu)$ to emphasize that it has finite dimension and that these elements are computationally represented as lists. We have the inclusion $H \subseteq \mathrm{Log}(K_\mathbb{R}^\times) \subseteq \mathrm{Div}_{K,\mathbb{S}}$. This degree-zero $\mathbb{S}$-divisor group serves as an ambient space for the Log-$\mathbb{S}$-unit lattice, which is defined as follows.

Definition 15.3 (Log-$\mathbb{S}$-unit lattice)**.** *Let $\mathbb{S}$ be a set of prime ideals of $\mathcal{O}_K$. The map $\mathrm{Log}_\mathbb{S}$ is defined over $\mathcal{O}_{K,\mathbb{S}}^\times$ by*

$$\begin{aligned} \mathrm{Log}_\mathbb{S} : \mathcal{O}_{K,\mathbb{S}}^\times &\longrightarrow \mathrm{Div}^0_{K,\mathbb{S}} \\ \alpha &\longmapsto ((-v_\mathfrak{p})_{\mathfrak{p}\in\mathbb{S}}, \mathrm{Log}(\alpha)) = -(\!|\alpha|\!). \end{aligned}$$

where the $v_\mathfrak{p}$ are such that $\alpha\cdot\mathcal{O}_K = \prod_{\mathfrak{p}\in\mathbb{S}} \mathfrak{p}^{v_\mathfrak{p}}$, and where $\mathrm{Log}(\alpha) = (n_\nu \log|\sigma_\nu(\alpha)|)_\nu \in \bigoplus_\nu \mathbb{R}$, with $n_\nu = 2$ if ν is complex and 1 otherwise, see Section 2.7.
The Log-$\mathbb{S}$-unit lattice is defined as

$$\mathrm{Log}_\mathbb{S}(\mathcal{O}_{K,\mathbb{S}}^\times) = \{\mathrm{Log}_\mathbb{S}(\alpha) \,|\, \alpha \in \mathcal{O}_{K,\mathbb{S}}^\times\} \subset \mathrm{Div}^0_{K,\mathbb{S}}\,.$$

Remark 15.4. *Note that the image of $\mathrm{Log}_\mathbb{S}$ on $\mathcal{O}_{K,\mathbb{S}}^\times$ is indeed in $\mathrm{Div}^0_{K,\mathbb{S}}$, since this map $\mathrm{Log}_\mathbb{S}$ is just the* negative *of the 'principal divisor' map $(\!|\cdot|\!) : K \to \mathrm{Div}_K$ restricted to $\mathcal{O}_{K,\mathbb{S}}^\times$, and principal divisors have degree zero (see Section 5.1).*

The Log-$\mathbb{S}$-unit lattice is a lattice in the real vector space

$$\mathrm{Div}^0_{K,\mathbb{S}}(\mathbb{R}) := \Big\{((x_\mathfrak{p})_{\mathfrak{p}\in\mathbb{S}}, (x_\nu)_\nu) \in \bigoplus_{\mathfrak{p}\in\mathbb{S}} \mathbb{R} \times \bigoplus_{\nu} \mathbb{R} \;\Big|\; \sum_{\mathfrak{p}\in\mathbb{S}} x_\mathfrak{p} \cdot \log \mathcal{N}(\mathfrak{p}) + \sum_\nu x_\nu = 0\Big\}. \tag{104}$$

Here the sum formula in above definition is a generalization of the degree function to real $x_\mathfrak{p} \in \mathbb{R}$. This lattice $\mathrm{Log}_\mathbb{S}(\mathcal{O}_{K,\mathbb{S}}^\times)$ has rank $\mathbb{r} + |\mathbb{S}|$ where $\mathbb{r} = n_\mathbb{R} + n_\mathbb{C} - 1$, i.e., it is full rank in $\mathrm{Div}^0_{K,\mathbb{S}}(\mathbb{R})$.

Note that $\mathrm{Log}_\mathbb{S}(\mathcal{O}_{K,\mathbb{S}}^\times) \subseteq \mathrm{Div}^0_{K,\mathbb{S}} \subseteq \mathrm{Div}_K$ inherits a Euclidean length notion via that of Div_K (see Section 2.8). When $\mathbb{S} = \emptyset$, the lattice $\mathrm{Log}_\mathbb{S}(\mathcal{O}_{K,\mathbb{S}}^\times) \subset \mathbb{R}^n$ coincides with the Log-unit lattice $\mathrm{Log}(\mathcal{O}_K^\times)$.

Lemma 15.5. *If $\mathbb{S}$ is a set of prime ideals generating the class group, then it holds that* [16] *$\mathrm{Vol}(\mathrm{Log}_\mathbb{S}(\mathcal{O}_{K,\mathbb{S}}^\times)) = h_K \cdot \mathrm{Vol}(\mathrm{Log}(\mathcal{O}_K^\times)) = h_K \cdot R_K \cdot \sqrt{n_\mathbb{R} + n_\mathbb{C}}$, where h_K is the class number of K, R_K is the regulator, $n_\mathbb{R}$ is the number of real embedding and $n_\mathbb{C}$ is the number of complex pairs of embeddings. In particular, $\mathrm{Vol}(\mathrm{Log}_\mathbb{S}(\mathcal{O}_{K,\mathbb{S}}^\times))$ does not depend on the choice of $\mathbb{S}$ (as long as it generates the class group). Moreover, we have*

$$\log(\mathrm{Vol}(\mathrm{Log}_\mathbb{S}(\mathcal{O}_{K,\mathbb{S}}^\times))) \leq \log|\Delta_K|.$$

[16] We understand $\mathrm{Vol}(\mathrm{Log}_\mathbb{S}(\mathcal{O}_{K,\mathbb{S}}^\times))$ as the volume of the quotient $\mathrm{Div}^0_{K,\mathbb{S}}(\mathbb{R}) / \mathrm{Log}_\mathbb{S}(\mathcal{O}_{K,\mathbb{S}}^\times)$.

Proof. The lemma follows from Lemma 5.1 and the observation that the group Pic^0_K in [14] is isomorphic to $\mathrm{Div}^0_{K,\mathbb{S}} \,/\, \mathrm{Log}_{\mathbb{S}}(\mathcal{O}^\times_{K,\mathbb{S}})$, whenever $\mathbb{S}$ generates the class group. □

A lower bound on the first minimum of the lattice $\mathrm{Log}_{\mathbb{S}}(\mathcal{O}^\times_{K,\mathbb{S}})$ can be derived from Kessler's lower bound on the first minimum of the Log-unit lattice (Lemma 2.14).

Lemma 15.6. *For any set of prime ideals* $\mathbb{S}$*, it holds that* $\lambda_1(\mathrm{Log}_{\mathbb{S}}(\mathcal{O}^\times_{K,\mathbb{S}})) \geq \frac{1}{1000\cdot\sqrt{n}\cdot\log(n)^3}$.

Proof. Let $\alpha \in \mathcal{O}^\times_{K,\mathbb{S}}$ be such that $\mathrm{Log}_{\mathbb{S}}(\alpha)$ reaches the first minimum of $\mathrm{Log}_{\mathbb{S}}(\mathcal{O}^\times_{K,\mathbb{S}})$, i.e., $\|\mathrm{Log}_{\mathbb{S}}(\alpha)\| = \lambda_1(\mathrm{Log}_{\mathbb{S}}(\mathcal{O}^\times_{K,\mathbb{S}}))$. Let $v_{\mathfrak{p}} \in \mathbb{Z}$ be such that $\alpha\cdot\mathcal{O}_K = \prod_{\mathfrak{p}\in\mathbb{S}} \mathfrak{p}^{v_{\mathfrak{p}}}$ (we know that they exist since α is an $\mathbb{S}$-unit). Assume first that one of the $v_{\mathfrak{p}}$ is non-zero (say $v_{\mathfrak{p}_0}$), then $\|\mathrm{Log}_{\mathbb{S}}(\alpha)\| \geq |v_{\mathfrak{p}_0}| \geq 1$ since $v_{\mathfrak{p}_0}$ is an integer. This gives us $\lambda_1(\mathrm{Log}_{\mathbb{S}}(\mathcal{O}^\times_{K,\mathbb{S}})) \geq 1 \geq \frac{1}{1000\cdot\sqrt{n}\cdot\log(n)^3}$ as desired.

Suppose now that all $v_{\mathfrak{p}}$'s are zero. Then $\alpha \in \mathcal{O}^\times_K$ is a unit and $\|\mathrm{Log}_{\mathbb{S}}(\alpha)\| = \|\mathrm{Log}(\alpha)\|$, and the bound follows from Lemma 2.14. □

Representation of the log-$\mathbb{S}$*-unit lattice.* In the present part of this article, we will compute a generating set of the Log-$\mathbb{S}$-unit lattice by a collection of pairs $((v^{(j)}_{\mathfrak{p}})_{\mathfrak{p}\in\mathbb{S}}, \alpha^{(j)}) \in \mathbb{Z}^{\mathbb{S}} \times K$, which generate $\mathrm{Log}_{\mathbb{S}}(\mathcal{O}^\times_{K,\mathbb{S}})$ as a $\mathbb{Z}$-module. How this is done is explained more precisely in Section 18. Up to and including that section, the elements $\alpha^{(j)}$ in these pairs are represented by their coordinates with respect to the basis of $\mathcal{O}_K$ (as explained in Section 2.6).

In the sections after that, Section 19 and Section 21, instead of a generating set, a *basis* of $\mathrm{Log}_{\mathbb{S}}(\mathcal{O}^\times_{K,\mathbb{S}})$ is computed. This will again lead to a collection of pairs $((w_{\mathfrak{p}})_{\mathfrak{p}\in\mathbb{S}}, \beta) \in \mathrm{Log}_{\mathbb{S}}(\mathcal{O}^\times_{K,\mathbb{S}})$ but this time the $\beta \in \mathrm{Log}_{\mathbb{S}}(\mathcal{O}^\times_{K,\mathbb{S}})$ are written in a so-called *compact representation*, see Section 2.6. Indeed, the elements β will be of the form $\beta = \prod_j (\alpha^{(j)})^{N_j}$, where $\alpha^{(j)} \in K$ come from the generating set, and $N_j \in \mathbb{Z}$. The product is *not* evaluated but rather stored as $(\alpha^{(j)}, N_j)_j$. Even in this compact representation, the bit size of the final N_j can be as large as the running time of the entire algorithm. For more details, see Section 2.6 and Section 21.

16. The generating and covering radius of the log-$\mathbb{S}$-unit lattice

In this section, we give an upper bound on the generating radius $\varrho(\mathrm{Log}_{\mathbb{S}}(\mathcal{O}^\times_{K,\mathbb{S}}))$, which is related to the covering radius of the Log-$\mathbb{S}$-unit lattice (see Definition 2.2 and Lemma 2.3). Recall that $\varrho(\mathrm{Log}_{\mathbb{S}}(\mathcal{O}^\times_{K,\mathbb{S}}))$ is the smallest real number $r > 0$ such that $\mathrm{Log}_{\mathbb{S}}(\mathcal{O}^\times_{K,\mathbb{S}})$ can be generated by vectors of Euclidean norm $\leq r$. Having an upper bound on this quantity r will be useful in Section 18. Indeed, we will first see in Section 17 how to compute a single vector in $\mathrm{Log}_{\mathbb{S}}(\mathcal{O}^\times_{K,\mathbb{S}})$, and we will then want, in Section 18, to compute a generating set of $\mathrm{Log}_{\mathbb{S}}(\mathcal{O}^\times_{K,\mathbb{S}})$. In order to do so, we will need to know how short we can expect the vectors of this generating set to be, which is exactly the generating radius $\varrho(\mathrm{Log}_{\mathbb{S}}(\mathcal{O}^\times_{K,\mathbb{S}}))$.

The first part of this section (until and including Lemma 16.8) is devoted to computing an upper bound on the covering radius[17] $\mathrm{Log}_{\mathbb{S}'}(\mathcal{O}^\times_{K,\mathbb{S}'})$ of the Log-$\mathbb{S}'$-unit lattice for a specific set $\mathbb{S}'$ (of relatively small cardinality). This upper bound

[17] This covering radius makes sense when seeing $\mathrm{Log}_{\mathbb{S}'}(\mathcal{O}^\times_{K,\mathbb{S}'})$ as a lattice in the real vector space $\mathrm{Div}_{K,\mathbb{S}}(\mathbb{R})$, see Equation (104).

is obtained due to the random walk result from [14, Theorem 3.3]. When rephrased with our formalism, and with a bit of work, the random walk theorem from [14] states that there exists a distribution $\mathcal{W}^T$ outputting somewhat short vectors of $\mathrm{Div}^0_{K,\mathbb{S}'}$ (the ambient space of $\mathrm{Log}_{\mathbb{S}'}(\mathcal{O}^\times_{K,\mathbb{S}'})$) and such that $\mathcal{W}^T \bmod \mathrm{Log}_{\mathbb{S}'}(\mathcal{O}^\times_{K,\mathbb{S}'})$ is close to uniform in $\mathrm{Div}^0_{K,\mathbb{S}'} \bmod \mathrm{Log}_{\mathbb{S}'}(\mathcal{O}^\times_{K,\mathbb{S}'})$. The shortness bound on the output vectors of $\mathcal{W}^T$ is proven in Lemma 16.5, and the close to uniformity modulo $\mathrm{Log}_{\mathbb{S}'}(\mathcal{O}^\times_{K,\mathbb{S}'})$ is obtained by combining Theorem 16.2 and Lemma 16.4. Intuitively, this means that any vector from $\mathrm{Div}^0_{K,\mathbb{S}'} \bmod \mathrm{Log}_{\mathbb{S}'}(\mathcal{O}^\times_{K,\mathbb{S}'})$ is somewhat close to a vector in the support of the distribution $\mathcal{W}^T$, which are all somewhat short. Hence, the covering radius of $\mathrm{Log}_{\mathbb{S}'}(\mathcal{O}^\times_{K,\mathbb{S}'})$ is somewhat small (see Lemma 16.8). In order to formalize this intuition, and because our distribution $\mathcal{W}^T$ is not perfectly uniform but only statistically close to uniform, we need to show that the volume of a small ball in $\mathrm{Div}^0_{K,\mathbb{S}'} \bmod \mathrm{Log}_{\mathbb{S}'}(\mathcal{O}^\times_{K,\mathbb{S}'})$ is not too small (it could happen that the ball folded modulo $\mathrm{Log}_{\mathbb{S}'}(\mathcal{O}^\times_{K,\mathbb{S}'})$ becomes much smaller than the original unfolded ball). This is what is done in Lemma 16.6.

Once we have an upper bound on $\mathrm{cov}(\mathrm{Log}_{\mathbb{S}'}(\mathcal{O}^\times_{K,\mathbb{S}'}))$, we also have an upper bound on $\varrho(\mathrm{Log}_{\mathbb{S}'}(\mathcal{O}^\times_{K,\mathbb{S}'}))$ by Lemma 2.3. We then need to extend this to larger sets $\mathbb{S} \supseteq \mathbb{S}'$, which is done in Lemma 16.9. We could also extend directly the bound on $\mathrm{cov}(\mathrm{Log}_{\mathbb{S}}(\mathcal{O}^\times_{K,\mathbb{S}}))$ to larger sets $\mathbb{S}$, but this would grow as $\sqrt{|\mathbb{S}|}$, whereas our sharper bound on the generating radius $\varrho(\mathrm{Log}_{\mathbb{S}}(\mathcal{O}^\times_{K,\mathbb{S}}))$ only grows as $\max_{\mathfrak{p}\in\mathbb{S}} \log \mathcal{N}(\mathfrak{p})$. Using the larger bound on $\mathrm{cov}(\mathrm{Log}_{\mathbb{S}}(\mathcal{O}^\times_{K,\mathbb{S}}))$ would have no impact on the asymptotic complexity of our algorithm, but we prefer to keep the intermediate results as tight as possible, to facilitate reusability in other works. Overall, our upper bound on $\varrho(\mathrm{Log}_{\mathbb{S}}(\mathcal{O}^\times_{K,\mathbb{S}}))$, which is the main result of this section, is stated in Proposition 16.10.

We start by rephrasing the random walk theorem from [14, Theorem 3.3] (see also Definition 6.1) in $\mathbb{S}$-unit terminology. As already mentioned in the preliminaries, the quotient group $\mathrm{Div}^0_{K,\mathbb{S}} / \mathrm{Log}_{\mathbb{S}}(\mathcal{O}^\times_{K,\mathbb{S}})$ is isomorphic to the Arakelov class group Pic^0_K, provided that $\mathbb{S}$ generates the class group.

Definition 16.1 (Random walk distribution, rephrased from [14, Definition 3.1])**.** *Let $\mathbb{S}$ be a set of prime ideals of K, $N \in \mathbb{Z}_{>0}$ and $s > 0$. The distribution $\mathcal{W}(\mathbb{S}, N, s)$ is a probability distribution over $\mathrm{Div}^0_{K,\mathbb{S}}$ obtained by the following procedure.*

1. *Set $y = 0 \in \mathrm{Div}_{K,\mathbb{S}}$. The first $|\mathbb{S}|$ coordinates of y are denoted by $y_\mathfrak{p}$ (with $\mathfrak{p} \in \mathbb{S}$) and the last $n_\mathbb{R} + n_\mathbb{C}$ coordinates of y are denoted by y_ν (with ν the places associated with the (pairs of) embeddings $\sigma : K \to \mathbb{C}$).*
2. *Sample $x \in H$ from a continuous centered Gaussian distribution with standard deviation s, where $H = \mathrm{span}_\mathbb{R}(\mathrm{Log}(\mathcal{O}^\times_K))$ (i.e., $H = \mathrm{Log}(K^0_\mathbb{R}) = \{(x_\nu)_\nu \in \bigoplus_\nu \mathbb{R} \mid \sum_\nu x_\nu = 0\}$). Set $y_\nu = x_\nu$ for all places ν.*
3. *For j from 1 to N, sample $\mathfrak{p}$ uniformly at random in $\mathbb{S}$ and update $y_\mathfrak{p} = y_\mathfrak{p} + 1$ and $y_\nu = y_\nu - \frac{n_\nu \log \mathcal{N}(\mathfrak{p})}{n}$ for all places ν. Here $n_\nu = 2$ if ν is complex and 1 otherwise.*
4. *Return y*

Observe that the distribution $\mathcal{W}(\mathbb{S}, N, s)$ produces vectors in $\mathrm{Div}^0_{K,\mathbb{S}}$, (i.e., vectors $y \in \bigoplus_{\mathfrak{p}\in\mathbb{S}} \mathbb{Z} \times \bigoplus_\nu \mathbb{R}$ such that $\sum_\nu y_\nu + \sum_{\mathfrak{p}\in\mathbb{S}} y_\mathfrak{p} \cdot \log \mathcal{N}(\mathfrak{p}) = 0$.

Theorem 16.2 ([14, Theorem 3.3], ERH)**.** *Let $\varepsilon > 0$ and $s > 0$ be positive real numbers and $k \in \mathbb{Z}_{>0}$. Let $s' = \min(\sqrt{2} \cdot s, 1/\eta_1(\mathrm{Log}(\mathcal{O}_K^\times)^\vee))$, where $\eta_1(\mathrm{Log}(\mathcal{O}_K^\times)^\vee)$ is the smoothing parameter of the dual lattice of the log-unit lattice $\mathrm{Log}(\mathcal{O}_K^\times)$. Then there exists a bound*

$$B = \widetilde{O}(n^{2k}[n^2(\log\log(1/\varepsilon))^2 + n^2(\log(1/s'))^2 + (\log|\Delta_K|)^2])$$

such that for any

$$N \geq \frac{\frac{n_\mathbb{R}+n_\mathbb{C}-1}{2} \cdot \log(1/s') + \frac{1}{2}\log(\mathrm{Vol}(\mathrm{Log}_\mathbb{S}(\mathcal{O}_{K,\mathbb{S}}^\times))) + \log(1/\varepsilon) + 1}{k \log n},$$

the random walk distribution $\mathcal{W}(\mathbb{S}, N, s) \bmod \mathrm{Log}_\mathbb{S}(\mathcal{O}_{K,\mathbb{S}}^\times)$ is within statistical distance at most $\varepsilon/2$ from uniform in $\mathrm{Div}_{K,\mathbb{S}}^0 \,/\, \mathrm{Log}_\mathbb{S}(\mathcal{O}_{K,\mathbb{S}}^\times)$, where $\mathbb{S} = \{\mathfrak{p}$ prime ideal $\mid \mathcal{N}(\mathfrak{p}) \leq B\}$.

Proof. This theorem is a direct translation of [14, Theorem 3.3] (or Theorem 6.4) with this section's $\mathbb{S}$-unit terminology: recall that if $\mathbb{S}$ generates the class group, then Pic_K^0 is isomorphic to $\mathrm{Div}_{K,\mathbb{S}}^0 \,/\, \mathrm{Log}_\mathbb{S}(\mathcal{O}_{K,\mathbb{S}}^\times)$. By a result of Bach [4], $\mathbb{S}$ indeed generates the class group for $B \geq 12\log^2|\Delta_K|$, hence indeed the bound in the theorem statement suffices. □

In order to prove Proposition 16.10, we will use a tail-cut random walk distribution.

Definition 16.3 (Tail-cut random walk distribution)**.** *Let $\mathbb{S}$ be a set of prime ideals of K, $N \in \mathbb{Z}_{>0}$, $s > 0$ and $\varepsilon \in (0, 1]$. The distribution $\mathcal{W}^T(\mathbb{S}, N, s, \varepsilon)$ is a probability distribution over $\mathrm{Div}_{K,\mathbb{S}}^0$ obtained as in Definition 16.1, except that the Gaussian distribution used to sample x is tail-cut at $t := s \cdot \sqrt{2 \cdot n \cdot \log\left(\frac{4 \cdot n}{\varepsilon}\right)}$. In other words, x is sampled from $\mathcal{G}_{H,s}$ conditioned on $\|x\| \leq t$.*

Lemma 16.4. *For any set of prime ideals $\mathbb{S}$, $N \in \mathbb{Z}_{>0}$, $s > 0$ and $\varepsilon \in (0, 1]$, the statistical distance between $\mathcal{W}(\mathbb{S}, N, s)$ and $\mathcal{W}^T(\mathbb{S}, N, s, \varepsilon)$ is upper bounded by $\varepsilon/2$.*

Proof. By the data processing inequality (Theorem 2.16), it suffices to prove that the statistical distance between $\mathcal{G}_{H,s}$ and its tail-cut variant is bounded by $\varepsilon/2$. This directly follows from Lemma 2.19. □

Lemma 16.5. *Let $\mathbb{S}$ be a set of prime ideals, $N \in \mathbb{Z}_{>0}$, $s > 0$ and $\varepsilon \in (0, 1]$. Then the support of the distribution $\mathcal{W}^T(\mathbb{S}, N, s, \varepsilon)$ is included in $\{x \in \mathrm{Div}_{K,\mathbb{S}}^0 \mid \|x\| \leq R\}$, where*

$$R = s \cdot \sqrt{2 \cdot n \cdot \log\left(\frac{4 \cdot n}{\varepsilon}\right)} + N \cdot \left(1 + \max_{\mathfrak{p} \in \mathbb{S}} \log \mathcal{N}(\mathfrak{p})\right).$$

Proof. Let y be output by $\mathcal{W}^T(\mathbb{S}, N, s, \varepsilon)$. By the definition of this distribution, we have $y \in \mathrm{Div}_{K,\mathbb{S}}^0$. We want to show that $\|y\| \leq R$.

Let us consider the components of $y = ((y_\mathfrak{p})_{\mathfrak{p}\in\mathbb{S}}, (y_\nu)_\nu)$ in $\bigoplus_{\mathfrak{p}\in\mathbb{S}} \mathbb{Z}$ and $\bigoplus_\nu \mathbb{R}$ separately. The vector $(y_\mathfrak{p})_{\mathfrak{p}\in\mathbb{S}}$ satisfies $\|(y_\mathfrak{p})_{\mathfrak{p}\in\mathbb{S}}\| \leq \|(y_\mathfrak{p})_{\mathfrak{p}\in\mathbb{S}}\|_1 \leq N$, by definition. From the tail-cut definition of $\mathcal{W}^T$, we also know that $(y_\nu)_\nu$ is the sum of a vector of norm $\leq t$ (where t is defined in Definition 16.3) and of N vectors of norm at most $\frac{\sqrt{2n} \cdot \max_{\mathfrak{p}\in\mathbb{S}} \log \mathcal{N}(\mathfrak{p})}{n} \leq \max_{\mathfrak{p}\in\mathbb{S}} \log \mathcal{N}(\mathfrak{p})$. Summing all these terms provides the desired upper bound on $\|y\|$. □

We will also need the following auxiliary lemma, which lower bounds the probability that a uniform element modulo $\mathrm{Log}_{\mathbb{S}}(\mathcal{O}_{K,\mathbb{S}}^{\times})$ falls in a small ball. We denote

$$t\mathcal{B}_{\infty} := \{((x_{\mathfrak{p}})_{\mathfrak{p}},(x_{\nu})_{\nu}) \in \mathrm{Div}^0_{K,\mathbb{S}}(\mathbb{R}) \mid \|((x_{\mathfrak{p}})_{\mathfrak{p}},(x_{\nu})_{\nu})\|_{\infty} \leq t\},$$

where $\mathrm{Div}^0_{K,\mathbb{S}}(\mathbb{R})$ was defined in Equation (104) (note that this is the $\mathbb{R}$-span, even for the coordinates in $\mathbb{S}$). Recall that $\mathcal{U}(\mathrm{Div}^0_{K,\mathbb{S}} / \mathrm{Log}_{\mathbb{S}}(\mathcal{O}_{K,\mathbb{S}}^{\times}))$ is the uniform distribution over the compact group $\mathrm{Div}^0_{K,\mathbb{S}} / \mathrm{Log}_{\mathbb{S}}(\mathcal{O}_{K,\mathbb{S}}^{\times})$.

Lemma 16.6. *Let $\mathbb{S}$ be any set of prime ideals generating the class group and let $x \in \mathrm{Div}^0_{K,\mathbb{S}}(\mathbb{R})$ be fixed. Then, for $t = 1 + \frac{1}{n} \cdot \sum_{\mathfrak{p}\in\mathbb{S}} \log \mathcal{N}(\mathfrak{p})$,*

$$\begin{aligned}&\Pr_{y \leftarrow \mathcal{U}(\mathrm{Div}^0_{K,\mathbb{S}} / \mathrm{Log}_{\mathbb{S}}(\mathcal{O}_{K,\mathbb{S}}^{\times}))}\Big(\exists z \in t\mathcal{B}_{\infty} \text{ so that } y = x + z \bmod \mathrm{Log}_{\mathbb{S}}(\mathcal{O}_{K,\mathbb{S}}^{\times})\Big)\\ &\geq \frac{1}{(1000 \cdot n \cdot \log(n)^3)^n \cdot h_K \cdot \mathrm{Vol}(\mathrm{Log}(\mathcal{O}_K^{\times}))}.\end{aligned}$$

Remark 16.7. *Note that in this lemma, the element y belongs to $\mathrm{Div}^0_{K,\mathbb{S}}$, hence has its first coordinates in $\bigoplus_{\mathfrak{p}\in\mathbb{S}} \mathbb{Z}$, but x and z are only required to live in the larger space $\mathrm{span}_{\mathbb{R}}(\mathrm{Log}_{\mathbb{S}}(\mathcal{O}_{K,\mathbb{S}}^{\times})) = \mathrm{Div}^0_{K,\mathbb{S}}(\mathbb{R})$, which means that their coordinates $x_{\mathfrak{p}}, z_{\mathfrak{p}}$ might be real.*

Proof. We will start with writing the probability in terms of volumes in the quotient $X_{\mathbb{S}} = \mathrm{Div}^0_{K,\mathbb{S}} / \mathrm{Log}_{\mathbb{S}}(\mathcal{O}_{K,\mathbb{S}}^{\times})$. For convenience, we will use the notation

$$\mathrm{Fold}_{\mathbb{S}} : \mathrm{Div}^0_{K,\mathbb{S}} \longrightarrow X_{\mathbb{S}}.$$

We have

$$\begin{aligned}&\Pr_{y \leftarrow \mathcal{U}(X)}\Big(\exists z \in t\mathcal{B}_{\infty} \text{ so that } y = x + z \bmod \mathrm{Log}_{\mathbb{S}}(\mathcal{O}_{K,\mathbb{S}}^{\times})\Big)\\ &= \frac{\mathrm{Vol}(\mathrm{Fold}_{\mathbb{S}}((t\mathcal{B}_{\infty} + x) \cap \mathrm{Div}^0_{K,\mathbb{S}}))}{\mathrm{Vol}(X_{\mathbb{S}})}\end{aligned}$$

To address the semi-continuous nature of $\mathrm{Div}^0_{K,\mathbb{S}}$ (as opposed to $\mathrm{Div}^0_{K,\mathbb{S}}(\mathbb{R})$), we define $\bar{x} \in \mathrm{Div}^0_{K,\mathbb{S}}$ by the rule $\bar{x}_{\mathfrak{p}} := \lceil x_{\mathfrak{p}} \rfloor$ and $\bar{x}_{\nu} = x_{\nu} - \frac{n_{\nu}}{n} \sum_{\mathfrak{p}} (\bar{x}_{\mathfrak{p}} - x_{\mathfrak{p}}) \log \mathcal{N}(\mathfrak{p})$ (for all $\mathfrak{p} \in \mathbb{S}$ and places ν; here $n_{\nu} = 2$ if ν is complex and 1 otherwise). Then $\bar{x} \in \mathrm{Div}^0_{K,\mathbb{S}}$ and

$$\|\bar{x} - x\|_{\infty} \leq \frac{1}{2} + \frac{1}{n} \sum_{\mathfrak{p}\in\mathbb{S}} \log \mathcal{N}(\mathfrak{p}). \tag{105}$$

In particular, $\frac{1}{2}\mathcal{B}_{\infty} + \bar{x} \subset t\mathcal{B}_{\infty} + x$. Since volumes in $X_{\mathbb{S}}$ are invariant by translation (by elements of $\mathrm{Div}^0_{K,\mathbb{S}}$), we get

$$\begin{aligned}\mathrm{Vol}(\mathrm{Fold}_{\mathbb{S}}((t\mathcal{B}_{\infty} + x) \cap \mathrm{Div}^0_{K,\mathbb{S}})) &\geq \mathrm{Vol}(\mathrm{Fold}_{\mathbb{S}}((\tfrac{1}{2}\mathcal{B}_{\infty} + \bar{x}) \cap \mathrm{Div}^0_{K,\mathbb{S}}))\\ &= \mathrm{Vol}(\mathrm{Fold}_{\mathbb{S}}(\tfrac{1}{2}\mathcal{B}_{\infty} \cap \mathrm{Div}^0_{K,\mathbb{S}}))\end{aligned}$$

We have $\frac{1}{2}\mathcal{B}_{\infty} \cap \mathrm{Div}^0_{K,\mathbb{S}} = \{0\}^{\mathbb{S}} \times (\frac{1}{2}\mathcal{B}_{\infty}^{\emptyset})$, where $\mathcal{B}_{\infty}^{\emptyset}$ is the unit ∞-ball in the $\mathbb{R}$-vector space $\mathrm{Div}^0_{K,\emptyset}(\mathbb{R})$. We deduce

$$\mathrm{Vol}(\mathrm{Fold}_{\mathbb{S}}(((\tfrac{1}{2}\mathcal{B}_{\infty}) \cap \mathrm{Div}^0_{K,\mathbb{S}})) = \mathrm{Vol}(\mathrm{Fold}_{\emptyset}(\tfrac{1}{2}\mathcal{B}_{\infty}^{\emptyset})) \geq \mathrm{Vol}(\mathrm{Fold}_{\emptyset}(r\mathcal{B}_{\infty}^{\emptyset}))$$

for any $0 < r \leq 1/2$. If furthermore $n^{1/2}r \leq \frac{1}{2}\lambda_1(\mathrm{Log}(\mathcal{O}_K^\times))$, the restriction of $\mathrm{Fold}_\emptyset$ to $r\mathcal{B}_\infty^\emptyset \subset n^{1/2}r\mathcal{B}_2^\emptyset$ is injective, and we get

$$\mathrm{Vol}(\mathrm{Fold}_\emptyset(r\mathcal{B}_\infty^\emptyset)) = \mathrm{Vol}(r\mathcal{B}_\infty^\emptyset) = (2r)^{n_\mathbb{R}+n_\mathbb{C}-1} \geq (2r)^n.$$

Composing all our volume (in)equalities, for $r \leq \frac{1}{2}\min(1, \lambda_1(\mathrm{Log}(\mathcal{O}_K^\times))/\sqrt{n})$ we have

$$\mathrm{Vol}(\mathrm{Fold}_\mathbb{S}((t\mathcal{B}_\infty + x) \cap \mathrm{Div}_{K,\mathbb{S}}^0)) \geq (2r)^n.$$

From Lemma 2.14, we know that $\lambda_1(\mathrm{Log}(\mathcal{O}_K^\times)) \geq (1000 \cdot \sqrt{n} \cdot \log(n)^3)^{-1}$. Hence, taking $r = \frac{1}{2} \cdot \frac{1}{1000n(\log n)^3}$ satisfies the above condition. Together with the identity $\mathrm{Vol}(X_\mathbb{S}) = h_K \cdot \mathrm{Vol}(\mathrm{Log}(\mathcal{O}_K^\times))$ (see Lemma 15.5), we obtain the desired bound. □

From this, we can prove the following upper bound on the covering radius of $\mathrm{Log}_\mathbb{S}(\mathcal{O}_{K,\mathbb{S}}^\times)$ when $|\mathbb{S}|$ is sufficiently large.

Lemma 16.8 (ERH). *There exist $B_1 = \mathsf{poly}(\log|\Delta_K|)$ such that if $\mathbb{S}'$ is the set of all prime ideals of norm $\leq B_1$, then $\mathbb{S}'$ generates the class group of K and we have*

$$\mathrm{cov}(\mathrm{Log}_{\mathbb{S}'}(\mathcal{O}_{K,\mathbb{S}'}^\times)) \leq \mathsf{poly}(\log|\Delta_K|).$$

Proof. Let us take $k = 1$, $\varepsilon = \frac{1}{2\cdot(1000\cdot n\cdot\log(n)^3)^n\cdot h_K\cdot\mathrm{Vol}(\mathrm{Log}(\mathcal{O}_K^\times))}$ and $s = 1/n$. Let B and N be as in Theorem 16.2, for these choices of k, ε and s, with N minimal.

We will prove that the lemma holds for $B_1 = B$. We have already seen that the B from Theorem 16.2 is such that all prime ideals of norm $\leq B$ generates the class group, which proves the first part of the lemma.

Let us now prove the second part of the lemma. Let $x \in \mathrm{span}_\mathbb{R}(\mathrm{Log}_{\mathbb{S}'}(\mathcal{O}_{K,\mathbb{S}'}^\times)) = \mathrm{Div}_{K,\mathbb{S}'}^0(\mathbb{R})$. We would like to show that there exists a vector y in the support of the tail-cut random walk distribution $\mathcal{W}^T(\mathbb{S}', N, s, \varepsilon)$ that is not too far away from $x \bmod \mathrm{Log}_{\mathbb{S}'}(\mathcal{O}_{K,\mathbb{S}'}^\times)$.

From Lemma 16.6 and by definition of ε, we know that

$$\Pr_{y \leftarrow \mathcal{U}(\mathrm{Div}_{K,\mathbb{S}'}^0 / \mathrm{Log}_{\mathbb{S}'}(\mathcal{O}_{K,\mathbb{S}'}^\times))}\Big(\exists z \in t\mathcal{B}_\infty \text{ such that } y = x + z \bmod \mathrm{Log}_{\mathbb{S}'}(\mathcal{O}_{K,\mathbb{S}'}^\times)\Big) \geq 2\varepsilon,$$

where $\mathcal{B}_\infty := \{((x_\mathfrak{p})_\mathfrak{p}, (x_\nu)_\nu) \in \mathrm{Div}_{K,\mathbb{S}}^0(\mathbb{R}) \mid \|((x_\mathfrak{p})_\mathfrak{p}, (x_\nu)_\nu)\|_\infty \leq t\}$ and $t = 1 + \sum_{\mathfrak{p}\in\mathbb{S}'} \log \mathcal{N}(\mathfrak{p})$.

Moreover, combining Lemma 16.4 and Theorem 16.2, we get that the statistical distance between $\mathcal{W}^T(\mathbb{S}', N, s, \varepsilon)$ and the uniform distribution over $\mathrm{Div}_{K,\mathbb{S}}^0 / \mathrm{Log}_{\mathbb{S}'}(\mathcal{O}_{K,\mathbb{S}'}^\times)$ is at most ε. Hence, we obtain that

$$\Pr_{y \leftarrow \mathcal{W}^T(\mathbb{S}', N, s, \varepsilon)}\Big(\exists z \in t\mathcal{B}_\infty \text{ such that } y = x + z \bmod \mathrm{Log}_{\mathbb{S}'}(\mathcal{O}_{K,\mathbb{S}'}^\times)\Big) \geq 2\varepsilon - \varepsilon > 0.$$

This proves the existence of some y in the support of $\mathcal{W}^T(\mathbb{S}', N, s, \varepsilon)$ and some $z \in \mathrm{Div}_{K,\mathbb{S}}^0(\mathbb{R}) = \mathrm{span}_\mathbb{R}(\mathrm{Log}_\mathbb{S}(\mathcal{O}_{K,\mathbb{S}}^\times))$ with $\|z\|_\infty \leq 1 + \sum_{\mathfrak{p}\in\mathbb{S}'} \log \mathcal{N}(\mathfrak{p})$ such that $x + z - y \in \mathrm{Log}_{\mathbb{S}'}(\mathcal{O}_{K,\mathbb{S}'}^\times)$. This means in particular that

$$\min_{v \in \mathrm{Log}_{\mathbb{S}'}(\mathcal{O}_{K,\mathbb{S}'}^\times)} \|x - v\| \leq \|z\| + \|y\|.$$

We know that the number of prime ideals of norm bounded by B is at most $B \cdot n$ (since there are at most n prime ideals above any prime integer in $\mathbb{Z}$). Hence, $\sum_{\mathfrak{p}\in\mathbb{S}'} \log \mathcal{N}(\mathfrak{p}) \leq B \cdot n \cdot \log B$ and we have $\|z\| \leq \sqrt{\mathbb{r} + 1 + |\mathbb{S}'|} \cdot (1 + nB\log B) \leq \mathsf{poly}(B, \log|\Delta_K|)$.

Since y belongs to the support of $\mathcal{W}^T(\mathbb{S}', N, s, \varepsilon)$, we know from Lemma 16.5 that $\|y\| \leq s \cdot \sqrt{2 \cdot n \cdot \log\left(\frac{4 \cdot n}{\varepsilon}\right)} + N \cdot \left(1 + \log B\right)$.

Let us upper bound the terms $\log(1/\varepsilon)$, B and N appearing in the two upper bounds. Recall from Lemma 15.5 that $\log(h_K \cdot \mathrm{Vol}(\mathrm{Log}(\mathcal{O}_K^\times))) \leq \log|\Delta_K|$. From this, we see that $\log(1/\varepsilon) = \mathsf{poly}(\log|\Delta_K|)$. It was shown in [14] (in the proof of Corollary 3.4) that $\eta_1(\mathrm{Log}(\mathcal{O}_K^\times)^\vee)) \leq \mathsf{poly}(n)$. Hence, by choice of s, we have $1/s' = \mathsf{poly}(n)$. From this, Lemma 15.5, and the choice of $k = 1$, we obtain that $B = \mathsf{poly}(\log|\Delta_K|)$ and $N = \mathsf{poly}(\log|\Delta_K|)$.

We conclude that for any $x \in \mathrm{span}_{\mathbb{R}}(\mathrm{Log}_{\mathbb{S}'}(\mathcal{O}_{K,\mathbb{S}'}^\times))$, we have

$$\min_{v \in \mathrm{Log}_{\mathbb{S}'}(\mathcal{O}_{K,\mathbb{S}'}^\times)} \|x - v\| \leq \mathsf{poly}(\log|\Delta_K|),$$

hence $\mathrm{cov}(\mathrm{Log}_{\mathbb{S}'}(\mathcal{O}_{K,\mathbb{S}'}^\times)) \leq \mathsf{poly}(\log|\Delta_K|)$. □

Only one last lemma remains to be proven before being able to prove Proposition 16.10. This lemma relates the generating radii $\varrho(\mathrm{Log}_{\mathbb{S}}(\mathcal{O}_{K,\mathbb{S}}^\times))$ and $\varrho(\mathrm{Log}_{\mathbb{S}'}(\mathcal{O}_{K,\mathbb{S}'}^\times))$ when $\mathbb{S}'$ is a subset of $\mathbb{S}$.

Lemma 16.9. *Let $\mathbb{S}$ and $\mathbb{S}'$ be finite sets of prime ideals in $\mathcal{O}_K$ satisfying $\mathbb{S}' \subseteq \mathbb{S}$. Let $\mathbb{S}'$ generate the class group. Then we have*

$$\varrho(\mathrm{Log}_{\mathbb{S}}(\mathcal{O}_{K,\mathbb{S}}^\times)) \leq (n + |\mathbb{S}'|) \cdot \varrho(\mathrm{Log}_{\mathbb{S}'}(\mathcal{O}_{K,\mathbb{S}'}^\times)) + \max_{\mathfrak{p} \in \mathbb{S}} \log \mathcal{N}(\mathfrak{p}) + 1.$$

Proof. In order to show an upper bound on the generating radius $\varrho(\mathrm{Log}_{\mathbb{S}}(\mathcal{O}_{K,\mathbb{S}}^\times))$, we will construct short vectors that generate the lattice $\mathrm{Log}_{\mathbb{S}}(\mathcal{O}_{K,\mathbb{S}}^\times)$, *using* the short vectors that generate the lattice $\mathrm{Log}_{\mathbb{S}'}(\mathcal{O}_{K,\mathbb{S}'}^\times)$. In this way, we relate the generating radii of $\mathrm{Log}_{\mathbb{S}}(\mathcal{O}_{K,\mathbb{S}}^\times)$ and $\mathrm{Log}_{\mathbb{S}'}(\mathcal{O}_{K,\mathbb{S}'}^\times)$. We will first show that these vectors that we construct span a lattice of rank $n_{\mathbb{R}} + n_{\mathbb{C}} + |\mathbb{S}| - 1$ (the same as the rank of $\mathrm{Log}_{\mathbb{S}}(\mathcal{O}_{K,\mathbb{S}}^\times)$) included in $\mathrm{Log}_{\mathbb{S}}(\mathcal{O}_{K,\mathbb{S}}^\times)$. We then conclude that both lattices are equal by a volumetric argument.

Let us order the elements of $\mathbb{S} = \{\mathfrak{p}_1, \ldots, \mathfrak{p}_{|\mathbb{S}|}\}$ such that $\mathbb{S}' = \{\mathfrak{p}_1, \ldots, \mathfrak{p}_{|\mathbb{S}'|}\}$. By definition of $\mathrm{Log}_{\mathbb{S}}(\mathcal{O}_{K,\mathbb{S}}^\times)$ and $\mathrm{Log}_{\mathbb{S}'}(\mathcal{O}_{K,\mathbb{S}'}^\times)$, we know that every vector $v \in \mathrm{Log}_{\mathbb{S}'}(\mathcal{O}_{K,\mathbb{S}'}^\times)$ padded with $|\mathbb{S}| - |\mathbb{S}'|$ zeros (at the 'prime places') belongs to $\mathrm{Log}_{\mathbb{S}}(\mathcal{O}_{K,\mathbb{S}}^\times)$.

Let us fix some $b_1, \cdots, b_r$ (for some $r > 0$) generating $\mathrm{Log}_{\mathbb{S}'}(\mathcal{O}_{K,\mathbb{S}'}^\times)$ with $\|b_i\| \leq \varrho(\mathrm{Log}_{\mathbb{S}'}(\mathcal{O}_{K,\mathbb{S}'}^\times))$ for all i's, and let us consider the associated vectors $\bar{b}_1, \cdots, \bar{b}_r \in \mathrm{Log}_{\mathbb{S}}(\mathcal{O}_{K,\mathbb{S}}^\times)$ obtained by padding them with zeros. Since padding with zeros does not change the euclidean norm, we still have $\|\bar{b}_i\| \leq \varrho(\mathrm{Log}_{\mathbb{S}'}(\mathcal{O}_{K,\mathbb{S}'}^\times))$ for all i's. Moreover, the vectors $\bar{b}_i$ span a lattice of rank $n_{\mathbb{R}} + n_{\mathbb{C}} + |\mathbb{S}'| - 1$, included in $\mathrm{Div}_{K,\mathbb{S}}^0$.

Let us now construct $|\mathbb{S}| - |\mathbb{S}'|$ other vectors $c_{|\mathbb{S}'+1|}, \cdots, c_{|\mathbb{S}|}$ in $\mathrm{Log}_{\mathbb{S}'}(\mathcal{O}_{K,\mathbb{S}'}^\times)$. Let us fix some i with $|\mathbb{S}'| < i \leq |\mathbb{S}|$ and consider the prime ideal $\mathfrak{p}_i \in \mathbb{S} \setminus \mathbb{S}'$. Since $\mathbb{S}'$ generates the class group, we know that there exists $\alpha_i \in K$ and $v^{(i)} \in \mathbb{Z}^{\mathbb{S}'}$ such that $\alpha_i \cdot \mathcal{O}_K = \mathfrak{p}_i \cdot \prod_{j \leq |\mathbb{S}'|} \mathfrak{p}_j^{v_j^{(i)}}$. For $j > |\mathbb{S}'|$, let us define $v_j^{(i)} = 1$ if $j = i$ and 0 otherwise. Then, the vector $c_i := \left((v_j^{(i)})_{j \leq |\mathbb{S}|}, -\mathrm{Log}(\alpha_i)\right)$ belongs to $\mathrm{Log}_{\mathbb{S}}(\mathcal{O}_{K,\mathbb{S}}^\times)$.

By definition, we know that we can decompose $c_i = d_i + e_i$ where d_i belongs to $\mathrm{span}_{\mathbb{R}}(\bar{b}_1, \cdots, \bar{b}_r)$ and $e_i = ((0, \ldots, 1, \ldots, 0), (-\frac{n_\nu \log \mathcal{N}(\mathfrak{p}_i)}{n})_\nu) \in \mathrm{Div}_{K,\mathbb{S}}$ (with the last coefficients (associated with the infinite places) equal to $\frac{n_\nu \log \mathcal{N}(\mathfrak{p}_i)}{n}$ and the 1 in position of the prime $\mathfrak{p}_i$). We can reduce the vector d_i by using the vectors $\bar{b}_1, \cdots, \bar{b}_r$ to ensure that $\|d_i\| \leq (n+|\mathbb{S}'|)\cdot \max_i \|\bar{b}_i\|$ (for example, by taking $n_{\mathbb{R}}+n_{\mathbb{C}}+|\mathbb{S}'|-1$ linearly independent vectors among the $\bar{b}_i$'s and performing Babai's round-off algorithm).

Hence, without loss of generality, we can assume that

$$\|c_i\| \leq (n+|\mathbb{S}'|)\cdot \varrho(\mathrm{Log}_{\mathbb{S}'}(\mathcal{O}_{K,\mathbb{S}'}^{\times})) + \max_{\mathfrak{p}\in\mathbb{S}} \log \mathcal{N}(\mathfrak{p}) + 1.$$

So if we can show that the $\bar{b}_i$'s and c_i's generate $\mathrm{Log}_{\mathbb{S}}(\mathcal{O}_{K,\mathbb{S}}^{\times})$, we have $\varrho(\mathrm{Log}_{\mathbb{S}}(\mathcal{O}_{K,\mathbb{S}}^{\times})) \leq (n+|\mathbb{S}'|)\cdot \varrho(\mathrm{Log}_{\mathbb{S}'}(\mathcal{O}_{K,\mathbb{S}'}^{\times})) + \max_{\mathfrak{p}\in\mathbb{S}} \log \mathcal{N}(\mathfrak{p}) + 1$.

So, it remains to show that the vectors $\bar{b}_i$'s and c_i's generate $\mathrm{Log}_{\mathbb{S}}(\mathcal{O}_{K,\mathbb{S}}^{\times})$. Let L be the lattice generated by the $\bar{b}_i$'s and the c_i's. By construction, $L \subseteq \mathrm{Log}_{\mathbb{S}}(\mathcal{O}_{K,\mathbb{S}}^{\times})$. Moreover, from the structure of the c_i's, we know that L has rank $n_{\mathbb{R}}+n_{\mathbb{C}}+|\mathbb{S}'|-1+(|\mathbb{S}|-|\mathbb{S}'|)$, i.e., the same rank as $\mathrm{Log}_{\mathbb{S}}(\mathcal{O}_{K,\mathbb{S}}^{\times})$. Finally, by choice of the c_i's, we know that $\mathrm{Vol}(L) = \mathrm{Vol}(\mathrm{Log}_{\mathbb{S}'}(\mathcal{O}_{K,\mathbb{S}'}^{\times})) = \mathrm{Vol}(\mathrm{Log}_{\mathbb{S}}(\mathcal{O}_{K,\mathbb{S}}^{\times}))$, where the last equality comes from Lemma 15.5. From this, we conclude that $L = \mathrm{Log}_{\mathbb{S}}(\mathcal{O}_{K,\mathbb{S}}^{\times})$, as desired. □

Combining everything, we can now prove Proposition 16.10.

Proposition 16.10 (ERH)**.** *There exists some $B_1 = \mathsf{poly}(\log|\Delta_K|)$ such that for any set $\mathbb{S}$ containing all prime ideals of $\mathcal{O}_K$ of norm $\leq B_1$, it holds that*

$$\varrho(\mathrm{Log}_{\mathbb{S}}(\mathcal{O}_{K,\mathbb{S}}^{\times})) \leq \mathsf{poly}(\log|\Delta_K|, \max_{\mathfrak{p}\in\mathbb{S}} \log \mathcal{N}(\mathfrak{p})).$$

Moreover, B_1 is such that the prime ideals of norm $\leq B_1$ generate the class group of K.

Proof of Proposition 16.10. We prove that the lemma holds for the same bound B_1 as in Lemma 16.8. We already know from Lemma 16.8 that this bound B_1 is such that all prime ideals of norm $\leq B_1$ generates the class group.

Let us call $\mathbb{S}'$ the set of prime ideals of norm $\leq B_1$. From Lemma 16.8 and Lemma 2.3, we know that $\varrho(\mathrm{Log}_{\mathbb{S}'}(\mathcal{O}_{K,\mathbb{S}'}^{\times})) \leq 2\cdot \mathrm{cov}(\mathrm{Log}_{\mathbb{S}'}(\mathcal{O}_{K,\mathbb{S}'}^{\times})) \leq \mathsf{poly}(\log|\Delta_K|)$.

From Lemma 16.9, we conclude that

$$\begin{aligned}\varrho(\mathrm{Log}_{\mathbb{S}}(\mathcal{O}_{K,\mathbb{S}}^{\times})) &\leq \sqrt{n+|\mathbb{S}'|}\cdot \varrho(\mathrm{Log}_{\mathbb{S}'}(\mathcal{O}_{K,\mathbb{S}'}^{\times})) + \max_{\mathfrak{p}\in\mathbb{S}} \log \mathcal{N}(\mathfrak{p}) + 1 \\ &\leq \mathsf{poly}\left(\log|\Delta_K|, \max_{\mathfrak{p}\in\mathbb{S}} \log \mathcal{N}(\mathfrak{p})\right),\end{aligned}$$

where we used the fact that $|\mathbb{S}'| \leq nB_1 = \mathsf{poly}(B_1, \log|\Delta_K|)$ (see the proof of Lemma 16.8). □

17. Generating a single $\mathbb{S}$-unit

In this section, we describe an algorithm that computes in subexponential time a vector in the Log-$\mathbb{S}$-unit lattice when the set $\mathbb{S}$ is sufficiently large. Our algorithm even does slightly more than that, namely given as input an integral ideal $\mathfrak{a}$, the algorithm computes a relation between the ideal $\mathfrak{a}$ and the ideals of $\mathbb{S}$, that is, it outputs $\alpha \in K$ and $(v_{\mathfrak{p}})_{\mathfrak{p}\in\mathbb{S}}$ such that $\mathfrak{a} = \alpha \cdot \mathcal{O}_K \cdot \prod_{\mathfrak{p}\in\mathbb{S}} \mathfrak{p}^{-v_{\mathfrak{p}}}$.

Running the algorithm with $\mathfrak{a} = \mathcal{O}_K$ provides a vector in the Log-$\mathbb{S}$-unit lattice. However, in Section 18 we will use the fact that the algorithm can take as input any integral ideal $\mathfrak{a}$ and not just $\mathcal{O}_K$. Indeed, in Section 18, we want to generate many independent vectors in the Log-$\mathbb{S}$-unit lattice (so that they generate the full lattice with good probability once we have enough of them). In order to ensure independence of the vectors, we will crucially rely on the fact that we can choose as input any ideal $\mathfrak{a}$.

Another place where we need to take as input an ideal $\mathfrak{a}$ is when we want to recover a generator of a principal ideal or compute a class group discrete logarithm (as explained in the introduction). In these cases, which correspond to the descent phase of the sieving algorithm, we need to find a relation between any input ideal $\mathfrak{a}$ and the prime ideals of $\mathbb{S}$ for which we computed the Log-$\mathbb{S}$-unit lattice.

Given as input an integral ideal $\mathfrak{a}$, Algorithm 3 uses Theorem 1.1 to sample elements $\alpha \in \mathfrak{a}$ until the relative ideal $\alpha \cdot \mathfrak{a}^{-1}$ is $\mathbb{S}$-smooth. Any such smooth relative ideal provides a relation between $\mathfrak{a}$ and the ideals of $\mathbb{S}$. By Theorem 1.1, the success probability of this procedure is related to the local density of smooth ideals, which we computed in Section 14. As will be discussed later (see Section 21.1) there are certain regimes, depending intricately on ρ_K, n and $\log |\Delta_K|$ where the sampling algorithm's success probability increases when the first small primes are omitted from the computation, thus introducing a modulus $\mathfrak{m}_0$ consisting of the product of small primes. This is the main reason why Algorithm 3 starts with computing an approximation of ρ_K and chooses different avenues depending on the value of ρ_K, leading to either a modulus $\mathfrak{m}_0$ consisting of small primes (if ρ_K is large) or an empty modulus $\mathfrak{m}_0 = \mathcal{O}_K$ (if ρ_K is small).

17.1. **The algorithm that generates a single $\mathbb{S}$-unit relation.** The following definition of ω will be used in Algorithm 3.

Definition 17.1. *For a given number field K of degree n, a modulus $\mathfrak{m}_0$, and real numbers $x, \mathsf{b} \geq 1$, we define $\omega \in \mathbb{Z}_{>0}$ as the smallest positive integer such that $r = 48 \cdot \omega \cdot \mathsf{b}^{2n/\mathsf{b}} \cdot n^{7/2} \cdot |\Delta_K|^{\frac{3}{2n}} \cdot \mathcal{N}(\mathfrak{m}_0)^{\frac{1}{n}}$ satisfies $r^n \geq e^n \max(B_{\mathrm{sm}}, B_{\mathrm{rw}}, 10x^2)$, where B_{sm} is defined in Lemma 14.1 (with $\varepsilon := 1$ loc. cit.), and B_{rw} is defined in Theorem 13.2.*

Note that using the fact that $B_{\mathrm{sm}} = \mathsf{poly}(\log |\Delta_K|)$ and $B_{\mathrm{rw}} := \mathsf{poly}(\log |\Delta_K|, \log \mathcal{N}(\mathfrak{m}_0))$, we can see that $\omega = O(1)$ is a constant (and so in particular $\mathrm{size}(\omega) = O(1)$ since it is an integer).

We can now describe Algorithm 3, which computes one relation.

Notation 17.2. *Throughout the following lemmas, we will use the following notation:*

- *We put $B_{\max} = \max(\exp(\sqrt{\log(r^n) \log\log(r^n)}), B_{\mathrm{sm}}, B_{\mathrm{rw}}, 10x^2)$. Note that this instantiation implies $x < \frac{10x^2}{4\log(10x^2)} < B_{\max}/(4 \log B_{\max})$ (in the first inequality we used $x \geq 1$ and for the second one we used the fact that $y \mapsto y/(4\log(y))$ is increasing for $y > 3$).*
- $u = \frac{\log(r^n)}{\log(B_{\max})}$;

Note that, since $r = 48 \cdot \omega \cdot \mathsf{b}^{2n/\mathsf{b}} \cdot n^{7/2} \cdot |\Delta_K|^{\frac{3}{2n}} \cdot \mathcal{N}(\mathfrak{m}_0)^{\frac{1}{n}} \geq 48$, we certainly have $r^n \geq e^n \cdot \exp(\sqrt{\log(r^n) \log\log(r^n)})$ and hence $r^n \geq e^n B_{\max}$.

In order to analyze the running time of Algorithm 3, we need to get a lower bound on the success probability p_{success} that the `repeat...until` loop from lines 10-12 terminate. This is what we compute in the next four lemmas. In order to improve readability, we decomposed the computation into small lemmas, the most interesting one being the last one (Lemma 17.6).

Algorithm 3 Computing one relation

```
Require:
    (i) An LLL-reduced basis of $\mathcal{O}_K$,
    (ii) an ideal $\mathfrak{a} \in \mathcal{I}_K$,
    (iii) a set of prime ideals $\mathbb{S}$ of $K$,
    (iv) $y \in K_{\mathbb{R}}^{\times}$.
Ensure: $\alpha \in \mathfrak{a}$ and $(v_{\mathfrak{p}})_{\mathfrak{p} \in \mathbb{S}} \in \mathbb{Z}_{\geq 0}^{\mathbb{S}}$ such that $\mathfrak{a} = \alpha \mathcal{O}_K \cdot \prod_{\mathfrak{p} \in \mathbb{S}} \mathfrak{p}^{-v_{\mathfrak{p}}}$
1: define $\mathsf{b} = n^{\frac{2}{3}}$
2: compute an approximation $\tilde{\rho}_K$ of $\rho_K$ such that $\tilde{\rho}_K \leq \rho_K < 2\tilde{\rho}_K$ (see Proposition 2.9)
3: define $x = \max\left( \frac{\log^{2/3} |\Delta_K|}{\log^{4/3} \log |\Delta_K|}, \frac{n^{2/3}}{\log^{2/3}(n)} \right)$
4: if $\tilde{\rho}_K \leq \exp(x \log^2(x))$ then
5:     define $\mathfrak{m}_0 = (1)$
6: else
7:     define $\mathfrak{m}_0 = \prod_{\mathcal{N}(\mathfrak{p})<x} \mathfrak{p}$
8: end if
9: Define $\omega$ as in Definition 17.1
10: repeat
11:     $\alpha \leftarrow \texttt{Sample}(\mathfrak{a}, y, \mathfrak{m}_0, \mathsf{b}, \omega)$ (see Theorem 13.2)
12: until $\alpha \mathcal{O}_K \cdot \mathfrak{a}^{-1}$ is $\mathbb{S}$-smooth
13: compute $(v_{\mathfrak{p}})_{\mathfrak{p} \in \mathbb{S}} \in \mathbb{Z}_{\geq 0}^{\mathbb{S}}$ such that $\alpha \mathcal{O}_K \cdot \mathfrak{a}^{-1} = \prod_{\mathfrak{p} \in \mathbb{S}} \mathfrak{p}^{v_{\mathfrak{p}}}$
14: return $(\alpha, (v_{\mathfrak{p}})_{\mathfrak{p} \in \mathbb{S}})$.
```

Lemma 17.3 (ERH)**.** *Put $r, B_{\max}, u \in \mathbb{R}_{>0}$ as in Notation 17.2. Assume that $\mathbb{S}$ contains all prime ideals not dividing $\mathfrak{m}_0$ of norm $\leq B_{\max}$, and that $\mathfrak{a}$ is coprime with $\mathfrak{m}_0$. Then the success probability p_{success} of $\alpha \mathcal{O}_K \mathfrak{a}^{-1}$ being $\mathbb{S}$-smooth in the repeat-loop (lines 10-12) of Algorithm 3 satisfies*

$$p_{\text{success}} \geq \frac{1}{6} \frac{\mathcal{N}(\mathfrak{m}_0)}{\phi(\mathfrak{m}_0) \cdot \rho_K} \frac{(4 \log(B_{\max}))^{1-u} \cdot u^{-u}}{B_{\max}}. \tag{106}$$

Proof. We denote $\mathcal{S}_{\mathbb{S}}$ for the $\mathbb{S}$-smooth integral ideals. Since $\mathbb{S}$ contains all prime ideals (coprime to $\mathfrak{m}_0$) of norm $\leq B_{\max}$ per assumption, and $B_{\max} \geq B_{\text{rw}}$ by definition, we have $\mathcal{S}_{\mathbb{S}} \mathcal{S}_{B_{\text{rw}}} = \mathcal{S}_{\mathbb{S}}$ (where we considered only prime ideals coprime to $\mathfrak{m}_0$ in $\mathcal{S}_{B_{\text{rw}}}$).

In line 11 of Algorithm 3, we use Sample from Theorem 13.2. According to that theorem,[18] the probability p_{success} of $\alpha \mathcal{O}_K \cdot \mathfrak{a}^{-1}$ lying in $\mathcal{S}_{\mathbb{S}} \cdot \mathcal{S}_{B_{\text{rw}}} = \mathcal{S}_{\mathbb{S}}$ is at least

$$p_{\text{success}} \geq \frac{\mathcal{N}(\mathfrak{m}_0)}{\phi(\mathfrak{m}_0)} \frac{\delta_{\mathcal{S}_{\mathbb{S}}}[r^n]}{6} \geq \frac{1}{6} \frac{\mathcal{N}(\mathfrak{m}_0)}{\phi(\mathfrak{m}_0) \cdot \rho_K} \frac{(4 \log(B_{\max}))^{1-u} \cdot u^{-u}}{B_{\max}}.$$

Here, the last inequality comes from computing the density of smooth numbers as in Lemma 14.1. We instantiate that lemma with $A := x$ if $\mathfrak{m}_0 \neq \mathcal{O}_K$ and $A := 1$ otherwise (this is to take into account that the prime dividing $\mathfrak{m}_0$ are omitted, see lines 7 and 5), $\varepsilon := 1$, and $B := B_{\max}$. These parameters indeed satisfy the constraints of Lemma 14.1 since $B_{\max} \geq B_{\text{sm}}$, $r^n \geq e^n B_{\max}$, and $A \leq B_{\max}/(4 \log B_{\max})$ (see Notation 17.2). □

[18] We can apply the theorem because $\mathfrak{a}$ is assumed to be coprime with $\mathfrak{m}_0$.

Lemma 17.4. *Keeping the notations from Lemma 17.3, it holds that*

$$\frac{(4\log(B_{\max}))^{1-u}\cdot u^{-u}}{B_{\max}} \geq \frac{1}{(B_{\max})^3}. \tag{107}$$

Proof. Taking negative logarithms, and writing $R = \log(r^n)$, $\beta = \log(B_{\max})$ and $u = R/\beta$, we obtain

$$-\log\left(\frac{(4\log(B_{\max}))^{1-u}\cdot u^{-u}}{B_{\max}}\right) = \beta + \left(\frac{R}{\beta} - 1\right)\log(4\beta) + \frac{R}{\beta}\log(R/\beta). \tag{108}$$

It suffices to show that the two right-most summands of Equation (108) are upper bounded by β. Note that, by Notation 17.2, we have that $\beta \geq \sqrt{R\log R}$. Hence, we immediately see $R/\beta \cdot \log(R/\beta) \leq \sqrt{R\log R} \leq \beta$. For the middle summand, note that $(\frac{R}{\beta} - 1)\log(4\beta) \leq \beta$ is equivalent to $R \leq \beta^2/\log(4\beta) + \beta$. The latter inequality is true because $\beta \mapsto \beta^2/\log(4\beta) + \beta$ is an increasing function for $\beta \geq 1/2$ and, for $R \geq 3$ (which is clearly satisfied since $r \geq 16$),

$$R \leq \frac{(\sqrt{R\log R})^2}{\log(4\sqrt{R\log R})} + \sqrt{R\log R} \leq \beta^2/\log(4\beta) + \beta,$$

where the first inequality follows by graphical inspection and the second by the monotonicity of the function $\beta \mapsto \beta^2/\log(4\beta) + \beta$. □

Lemma 17.5. *Keeping the notations from Lemma 17.3, it holds that*

$$B_{\max} \in \textsf{\textit{poly}}(L_{|\Delta_K|}(\tfrac{1}{2}), L_{\mathcal{N}(\mathfrak{m}_0)}(\tfrac{1}{2}), L_{n^n}(\tfrac{2}{3})).$$

Proof. From the definition of $B_{\max}$ in Notation 17.2, we have that

$$B_{\max} \in \mathsf{poly}(\exp(\sqrt{\log(r^n)\log\log(r^n)}), \log|\Delta_K|, \log\mathcal{N}(\mathfrak{m}_0)) \tag{109}$$
$$= \mathsf{poly}(L_{|\Delta_K|}(\tfrac{1}{2}), L_{\mathcal{N}(\mathfrak{m}_0)}(\tfrac{1}{2}), L_{n^n}(\tfrac{2}{3})). \tag{110}$$

Indeed, by the definition of r and the definition of b, and using that $\omega = O(1)$

$$\log(r^n) \leq O\Big(\frac{n^2\log\mathsf{b}}{\mathsf{b}} + \log|\Delta_K| + \log\mathcal{N}(\mathfrak{m}_0)\Big) = O(n^{4/3}\log n, \log|\Delta_K|, \log\mathcal{N}(\mathfrak{m}_0)).$$

And hence

$$\exp(\sqrt{\log(r^n)\log\log(r^n)}) \leq \mathsf{poly}(L_{|\Delta_K|}(\tfrac{1}{2}), L_{\mathcal{N}(\mathfrak{m}_0)}(\tfrac{1}{2}), L_{n^n}(\tfrac{2}{3})). \tag{111}$$

□

Lemma 17.6. *Keep the notations from Lemma 17.3 and define*

$$\rho_K^{\mathrm{cut}} = \min\left(\rho_K, \max(e^{\log^{\frac{2}{3}}|\Delta_K|\cdot\log^{\frac{2}{3}}(\log|\Delta_K|)}, e^{n^{\frac{2}{3}}\log^{\frac{4}{3}}(n)})\right).$$

Then it holds that $B_{\max}$ and $p_{\mathrm{success}}^{-1}$ are both in $\textsf{\textit{poly}}(L_{|\Delta_K|}(\tfrac{1}{2}), L_{n^n}(\tfrac{2}{3}), \rho_K^{\mathrm{cut}})$.

Proof. Combining Lemma 17.3, Lemma 17.4 we obtain that

$$p_{\mathrm{success}}^{-1} \in \mathsf{poly}(\tfrac{\phi(\mathfrak{m}_0)\cdot\rho_K}{\mathcal{N}(\mathfrak{m}_0)}, B_{\max}).$$

Moreover, Lemma 17.5 gives us $B_{\max} \in \mathsf{poly}(L_{|\Delta_K|}(\tfrac{1}{2}), L_{n^n}(\tfrac{2}{3}), L_{\mathcal{N}(\mathfrak{m}_0)}(\tfrac{1}{2}))$. To prove the result, it then suffices to show that

$$\mathsf{poly}(L_{\mathcal{N}(\mathfrak{m}_0)}(\tfrac{1}{2}), \tfrac{\phi(\mathfrak{m}_0)\cdot\rho_K}{\mathcal{N}(\mathfrak{m}_0)}) = \mathsf{poly}(\rho_K^{\mathrm{cut}}).$$

We write $x = \max\left(\frac{\log^{2/3}|\Delta_K|}{\log^{4/3}\log|\Delta_K|}, \frac{n^{2/3}}{\log^{2/3}(n)}\right)$ as in line 3 of Algorithm 3. Note that $\rho_K^{\mathrm{cut}} = \min\left(\rho_K, \exp(\Theta(x\log^2(x)))\right)$. We distinguish two cases.
Case 1: $\tilde{\rho}_K \leq \exp(x\log^2(x))$**.**
By line 4 and 5, we then have $\rho_K < 2\tilde{\rho}_K \leq 2\cdot\exp(x\log^2(x))$ (in particular, $\rho_K = \mathsf{poly}(\rho_K^{\mathrm{cut}})$) and $\mathfrak{m}_0 = (1)$. Hence $\mathcal{N}(\mathfrak{m}_0) = \phi(\mathfrak{m}_0) = 1$ and thus

$$\mathsf{poly}(L_{\mathcal{N}(\mathfrak{m}_0)}(\tfrac{1}{2}), \tfrac{\phi(\mathfrak{m}_0)\cdot\rho_K}{\mathcal{N}(\mathfrak{m}_0)}) = \mathsf{poly}(\rho_K) = \mathsf{poly}(\rho_K^{\mathrm{cut}}).$$

Case 2: $\tilde{\rho}_K > \exp(x\log^2(x))$**.**
In this case, it holds that $x\log(x)^2 = O(\log\rho_K)$ and have $\mathfrak{m}_0 = \prod_{\mathcal{N}(\mathfrak{p})<x}\mathfrak{p}$ (by lines 4 and 7). On one hand, by Proposition 11.2 we have

$$\begin{aligned} -\log\left(\frac{\mathcal{N}(\mathfrak{m}_0)}{\phi(\mathfrak{m}_0)\cdot\rho_K}\right) &\leq \frac{8\log|\Delta_K|}{\sqrt{x}} + \frac{8n\log x}{\sqrt{x}} \\ &\leq O(x\log^2(x)) = O(\log\rho_K^{\mathrm{cut}}). \end{aligned} \tag{112}$$

This follows from the fact that $\log|\Delta_K| \leq O(x^{\frac{3}{2}}\cdot\log^2(x))$ and $n \leq O(x^{\frac{3}{2}}\log x)$, since $x = \max\left(\left(\frac{\log|\Delta_K|}{\log^2\log|\Delta_K|}\right)^{\frac{2}{3}}, \left(\frac{n}{\log n}\right)^{\frac{2}{3}}\right)$. On the other hand, using the same bounds on $\log|\Delta_K|$ and n, we have, by Lemma 11.4

$$\begin{aligned} \log\mathcal{N}(\mathfrak{m}_0) &\leq O(x + \sqrt{x}\log(x)\log|\Delta_K| + \sqrt{x}\log^2(x)\cdot n) \\ &\leq O\left(x^2\log^3(x)\right). \end{aligned} \tag{113}$$

We deduce that

$$\log L_{\mathcal{N}(\mathfrak{m}_0)}(\tfrac{1}{2}) \leq O(\sqrt{\log\mathcal{N}(\mathfrak{m}_0)\log\log\mathcal{N}(\mathfrak{m}_0)}) \leq O(x\log^2(x)) = O(\log\rho_K^{\mathrm{cut}}). \tag{114}$$

Combining Equations (112) and (114), we thus obtain

$$\mathsf{poly}(L_{\mathcal{N}(\mathfrak{m}_0)}(\tfrac{1}{2}), \tfrac{\phi(\mathfrak{m}_0)\cdot\rho_K}{\mathcal{N}(\mathfrak{m}_0)}) = \mathsf{poly}(\rho_K^{\mathrm{cut}}).$$

□

Theorem 17.7 (ERH)**.** *There exists some* $B_{\max} = \textsf{\textit{poly}}(L_{|\Delta_K|}(1/2), L_{n^n}(2/3), \rho_K^{\mathrm{cut}})$, *where*

$$\rho_K^{\mathrm{cut}} = \min\left(\rho_K, \max(e^{\log^{\frac{2}{3}}|\Delta_K|\cdot\log^{\frac{2}{3}}(\log|\Delta_K|)}, e^{n^{\frac{2}{3}}\log^{\frac{4}{3}}(n)})\right),$$

such that the following holds. Assume that $\mathbb{S}$ *contains all prime ideals coprime to* $\mathfrak{m}_0$ *of norm* $\leq B_{\max}$. *Then, on input an integral ideal* $\mathfrak{a}$ *coprime with* $\mathfrak{m}_0$, *the set* $\mathbb{S}$ *and* $y \in H = \mathrm{Log}(K_{\mathbb{R}}^0)$, *Algorithm 3 outputs* $(\alpha, (v_{\mathfrak{p}})_{\mathfrak{p}\in\mathbb{S}}) \in \mathfrak{a}\times\mathbb{Z}_{\geq 0}^{|\mathbb{S}|}$ *such that*

$$\mathfrak{a} = \alpha\mathcal{O}_K \cdot \prod_{\mathfrak{p}\in\mathbb{S}}\mathfrak{p}^{-v_{\mathfrak{p}}}.$$

Furthermore, Algorithm 3 runs in expected time

$$\textsf{\textit{poly}}(L_{|\Delta_K|}(\tfrac{1}{2}), L_{n^n}(\tfrac{2}{3}), \mathrm{size}(\mathbb{S}), \rho_K^{\mathrm{cut}}, \mathrm{size}(\mathfrak{a}), \mathrm{size}(y), \log|\Delta_K|).$$

Proof. Thanks to the `repeat...until` loop from lines 10 to 12, the output of Algorithm 3 is correct whenever it terminates. The fact that $B_{\max} = \mathsf{poly}(L_{|\Delta_K|}(1/2), L_{n^n}(2/3), \rho_K^{\mathrm{cut}})$ follows from Lemma 17.6. Hence, the main focus of this proof is the expected running time of the algorithm.

Since lines 1 up to 10 do not significantly contribute to the running time, it suffices to concentrate on the expected running time of lines 11 up to 14.

For this time analysis we will use that $\log \mathcal{N}(\mathfrak{m}_0) \leq \mathsf{poly}(\log|\Delta_K|)$, by Lemma 11.4 and the definition of x on line 3. Also, we will use that $\mathsf{b}^{\mathsf{b}} \leq L_{n^n}(\frac{2}{3})$, from the definition of $\mathsf{b} = n^{2/3}$.

- (Line 11) The sampling algorithm `Sample` takes (by Theorem 13.2) time $\mathsf{poly}(\log|\Delta_K|, \mathrm{size}(\mathfrak{a}), \log(\mathcal{N}(\mathfrak{m}_0)), \mathrm{size}(y), \mathsf{b}^{\mathsf{b}}) = \mathsf{poly}(\log|\Delta_K|, \mathrm{size}(\mathfrak{a}), \mathrm{size}(y), L_{n^n}(\frac{2}{3}))$. Here we use that $\mathrm{size}(\omega) = O(1)$ (see Definition 17.1).
- (Line 12 and 13) Checking for $\alpha\mathcal{O}_K \cdot \mathfrak{a}^{-1}$ being $\mathbb{S}$-smooth (line 12), as well as writing down the decomposition (line 13) if the ideal indeed is smooth can be done with $\mathsf{poly}(\mathrm{size}(\mathbb{S}))$ division of ideals.
- (Line 14) This is the return statement.

Hence, denoting p_{success} for the probability of $\alpha\mathcal{O}_K \cdot \mathfrak{a}^{-1}$ being $\mathbb{S}$-smooth in line 12, we obtain an expected running time of

$$p_{\text{success}}^{-1} \cdot \mathsf{poly}(L_{n^n}(\tfrac{2}{3}), \mathrm{size}(\mathbb{S}), \mathrm{size}(\mathfrak{a}), \mathrm{size}(y), \log|\Delta_K|). \tag{115}$$

Using Lemma 17.6 provides an upper bound on p_{success}^{-1} and concludes the proof. □

17.2. **Properties of the output distribution of Algorithm 3.** From now on, we assume that the input ideal $\mathfrak{a}$ is $\mathbb{S}$-smooth, since this will be the case when we will call Algorithm 3 in the next section (this ensures in particular that $\mathfrak{a}$ is coprime with $\mathfrak{m}_0$, which is needed to apply Theorem 17.7). We describe the input of Algorithm 3 as a divisor $\mathbf{a} = d(\mathfrak{a}) + \mathrm{Log}(y) \in \mathrm{Div}_{K,\mathbb{S}}$ (we recover $\mathfrak{a} = \mathrm{Exp}_{\mathrm{f}}(\mathbf{a}_{\mathrm{f}})$ and $y = \mathrm{Exp}_{\infty}(\mathbf{a}_{\infty}) \in K_{\mathbb{R}}^{\times}$). The following properties of Algorithm 3 are required in the later proof that Algorithm 4 (which is essentially Algorithm 3 with a *Gaussian distributed input*) has an evenly distributed and concentrated output distribution (see Lemmas 18.10 and 18.11). These two properties of this last output distribution are indispensable to show that sufficiently many samples from this algorithm eventually generate the log-$\mathbb{S}$-unit lattice $\mathrm{Log}_{\mathbb{S}}(\mathcal{O}_{K,\mathbb{S}}^{\times})$.

We denote the output distribution of $\alpha \in K$ in Algorithm 3 with input $\mathbf{a} \in \mathrm{Div}_{K,\mathbb{S}}$ by[19] $\bar{\mathcal{D}}_{\mathbf{a}}$. By abuse of notation, we denote by the same symbol $\bar{\mathcal{D}}_{\mathbf{a}}$ the distribution over $\mathrm{Log}_{\mathbb{S}}(\mathcal{O}_{K,\mathbb{S}}^{\times})$ defined by sampling $\alpha \leftarrow \bar{\mathcal{D}}_{\mathbf{a}}$ and taking the log-$\mathbb{S}$-unit map[20] $\mathrm{Log}_{\mathbb{S}}(\alpha)$. Note that this map is well-defined, as Algorithm 3 only outputs $\alpha \in K^*$ whose prime divisors lie in $\mathbb{S}$, i.e., $\mathbb{S}$-units (by line 12 and because $\mathfrak{a}$ is $\mathbb{S}$-smooth). This yields a distribution $\bar{\mathcal{D}}_{\mathbf{a}} \in L^1(\mathrm{Log}_{\mathbb{S}}(\mathcal{O}_{K,\mathbb{S}}^{\times}))$ for each $\mathbf{a} \in \mathrm{Div}_{K,\mathbb{S}}$. Note that Algorithm 3 and hence the distribution $\bar{\mathcal{D}}_{\mathbf{a}}$ is independent on the degree of $\mathbf{a}$ (as in Equation (10)), as `Sample` from Theorem 13.2 is independent on the norm.

Lemma 17.8. *For all $\mathbf{a} \in \mathrm{Div}_{K,\mathbb{S}}$, the output distribution $\bar{\mathcal{D}}_{\mathbf{a}}$ of Algorithm 3 (where we consider the divisor $\mathbf{a} = d(\mathfrak{a}) + \mathrm{Log}(y)$ as input, which is equivalent to the input $(\mathfrak{a}, y)$) satisfies the following three properties.*

(i) We have $\bar{\mathcal{D}}_{\mathbf{a}}[\mathrm{Log}_{\mathbb{S}}(\mathcal{O}_{K,\mathbb{S}}^{\times})] = 1$.

(ii) For all $\alpha \in K^{\mathfrak{m}_0,1}$ and all $x \in \mathrm{Log}_{\mathbb{S}}(\mathcal{O}_{K,\mathbb{S}}^{\times})$

$$\bar{\mathcal{D}}_{\mathbf{a}}[x] = \bar{\mathcal{D}}_{\mathbf{a}-\mathrm{Log}_{\mathbb{S}}(\alpha)}[x + \mathrm{Log}_{\mathbb{S}}(\alpha)].$$

[19] Note the difference in notation compared to $\mathcal{D}$ in Section 10; the bar signifies that the distribution is from Algorithm 3.

[20] This sends the roots of unity to the zero-vector, but that is not troublesome, as elements $\alpha, \zeta\alpha \in K^*$ for a root of unity ζ have equal probability to be sampled.

(iii) There exists some $R = O(\log^2|\Delta_K|) + \|\mathbf{a}\|$ *such that*

$$\bar{\mathcal{D}}_{\mathbf{a}}\left[\operatorname{Log}_{\mathbb{S}}(\mathcal{O}_{K,\mathbb{S}}^{\times})\backslash(R\cdot\mathcal{B}_2)\right] = 0$$

where $R\cdot\mathcal{B}_2 = \{\mathbf{a}\in\operatorname{Div}_{K,\mathbb{S}} \mid \|\mathbf{a}\|\le R\}$*.*

(iv) For $\mathbf{a},\mathbf{a}'\in\operatorname{Div}_{K,\mathbb{S}}$ *with* $\mathbf{a}' = \mathbf{a} + \sum_\nu b_\nu (\!|\nu|\!)$*, we have*

$$\|\bar{\mathcal{D}}_{\mathbf{a}} - \bar{\mathcal{D}}_{\mathbf{a}'}\|_1 \le 3\cdot|\Delta_K|\cdot r^n\cdot n^2\cdot\|b\| + 1/200.$$

with $r = 48\cdot\omega\cdot\mathsf{b}^{2n/\mathsf{b}}\cdot n^{7/2}\cdot|\Delta_K|^{\frac{3}{2n}}\cdot\mathcal{N}(\mathfrak{m}_0)^{\frac{1}{n}}$*.*

Proof. Statement (i) follows from the fact that Algorithm 3 outputs α only if $\alpha\mathcal{O}_K\cdot\mathfrak{a}^{-1}$ is $\mathbb{S}$-smooth. Together with the fact that $\mathfrak{a}$ itself is assumed to be $\mathbb{S}$-smooth (it is made out of $\mathbf{a}\in\operatorname{Div}_{K,\mathbb{S}}$), we deduce that α is $\mathbb{S}$-smooth, thus an $\mathbb{S}$-unit. Therefore, $\bar{\mathcal{D}}_{\mathbf{a}}$ only has support on $\operatorname{Log}_{\mathbb{S}}(\mathcal{O}_{K,\mathbb{S}}^{\times})\subseteq\operatorname{Div}_{K,\mathbb{S}}^0$.

Statement (ii) follows from property (2) of Theorem 13.2. Let $\mathbf{a} = d(\mathfrak{a}) + \operatorname{Log}(y)$ for $\mathfrak{a} = \operatorname{Exp}_{\mathrm{f}}(\mathbf{a}_{\mathrm{f}})$ and $y = \operatorname{Exp}_\infty(\mathbf{a}_\infty)$. By definition of $\operatorname{Log}_{\mathbb{S}}(\alpha)$, we have that $\mathbf{a} - \operatorname{Log}_{\mathbb{S}}(\alpha) = d(\mathfrak{a}\cdot(\alpha)) + \operatorname{Log}(y\cdot\alpha^{-1})$. Property (2) of Theorem 13.2 tells us that $\mathcal{D}_{y,\mathfrak{a}}[-] = \mathcal{D}_{y\cdot\alpha^{-1},\mathfrak{a}\cdot(\alpha)}[\alpha\cdot -]$ (note the different distribution, this is the distribution from Theorem 13.2). As the exit criterion of the repeat loop (line 12 of Algorithm 3) happens at the same occurrences, the output distribution function $\bar{\mathcal{D}}_{\mathbf{a}}$ satisfies $\bar{\mathcal{D}}_{\mathbf{a}}[x] = \bar{\mathcal{D}}_{\mathbf{a}-\operatorname{Log}_{\mathbb{S}}(\alpha)}[x + \operatorname{Log}_{\mathbb{S}}(\alpha)]$ for all $x\in\operatorname{Log}_{\mathbb{S}}(\mathcal{O}_{K,\mathbb{S}}^{\times})$.

The statement (iii) follows from property (3) of Theorem 13.2, $\omega = O(1)$ (see Definition 17.1) and the fact that $\log\mathcal{N}(\mathfrak{m}_0) = O(x\cdot\log^2(x)\cdot\log|\Delta_K|) = O(\log^2|\Delta_K|)$ (see Equation (113) in the proof of Theorem 17.7, and see the definition of x in line 3 of Algorithm 3). Note that here the correspondence between $(y,\mathfrak{a})\in K_{\mathbb{R}}^{\times}\times\mathcal{I}_K$ and $\mathbf{a} = d(\mathfrak{a}) + \operatorname{Log}(y)$ is used.

Statement (iv) follows from Lemma 10.8 and Theorem 13.2 (property (1), Lipschitz-property), with the fact that the success probability of a single loop of Theorem 17.7 is lower bounded by $\delta_{\mathcal{S}}[r^n]/6 \ge \frac{1}{6\cdot|\Delta_K|\cdot r^n}$ (see Equation (102) in the proof of Theorem 13.2). Hence we have (applying Lemma 10.8 with $p^{-1}\le 6\cdot|\Delta_K|\cdot r^n$)

$$\|\bar{\mathcal{D}}_{\mathbf{a}} - \bar{\mathcal{D}}_{\mathbf{a}'}\|_1 \le 3\cdot|\Delta_K|\cdot r^n\cdot n^2\cdot\|\operatorname{Log}(y)\| + 1/200.$$

□

18. Obtaining a generating set of $\mathbb{S}$-units

18.1. **Introduction.** In Section 17 a probabilistic algorithm is described that computes a *single* $\mathbb{S}$-unit (or a logarithmic $\mathbb{S}$-unit lattice point). But to actually obtain the entire $\mathbb{S}$-unit group $\mathcal{O}_{K,\mathbb{S}}^{\times}$, one needs a *generating set* of $\mathbb{S}$-units. Or, equivalently, a $\mathbb{Z}$-generating set of lattice points for the Log-$\mathbb{S}$-unit lattice $\operatorname{Log}_{\mathbb{S}}(\mathcal{O}_{K,\mathbb{S}}^{\times})$.

A 'good' output distribution. One could hope that such a generating set will eventually be formed by *repeating* the probabilistic algorithm (Algorithm 3) many times and joining the outputs. The chance for this approach to succeed strongly depends on the *output distribution* of the mentioned probabilistic algorithm. Indeed, one could imagine cases where the output distribution is, for example, only being supported on a strict *sublattice* of the logarithmic $\mathbb{S}$-unit lattice. In such instances one never obtains a generating set of the full logarithmic $\mathbb{S}$-unit lattice.

It turns out that, for the repeated sampling approach to succeed, it is sufficient for the output distribution to have the following two properties: 'evenly distributed', meaning not having too much weight on a strict sublattice, and 'concentrated', meaning that most of the weight is reasonably centered around the origin. These two properties are made more precise in Section 18.2, as well as the consequences of these properties to the number of samples needed to generate the whole lattice.

The output distribution of the sampling algorithm. It remains to be shown that the output distribution of the sampling algorithm (Algorithm 3) is indeed actually 'good', i.e., evenly distributed and concentrated.

It turns out that the output distribution of Algorithm 3 for a *fixed* input $(\mathfrak{a}, y) \in \mathcal{I}_K \times K_{\mathbb{R}}^{\times}$ is hard to analyze; and it seems very difficult to show that it satisfies these properties. Fortunately, one can feed the algorithm different inputs, causing to gain more control of the output distribution.

More specifically, in our approach we randomly sample a Gaussian distributed point from the space $\mathrm{Div}_{K,\mathbb{S}}$, which can be considered as an 'ambient space' of the logarithmic $\mathbb{S}$-units. This randomly sampled point is then transformed into an element $(\mathfrak{a}, y) \in \mathcal{I}_K \times K_{\mathbb{R}}^{\times}$ ($\mathfrak{a}$ from the finite places and y from the infinite places), which is then given as an input to the $\mathbb{S}$-unit sampling algorithm (Algorithm 3).

We show that this 'compound' distribution, taking into account both the randomness of the Gaussian over $\mathrm{Div}_{K,\mathbb{S}}$ and the randomness of the $\mathbb{S}$-unit sampling algorithm, indeed satisfies the 'evenly distributed' property (see Lemma 18.11) and 'concentrated' property (see Lemma 18.10). Thus, repeating the $\mathbb{S}$-unit sampling algorithm with Gaussian distributed inputs lets us indeed obtain a generating set of the logarithmic $\mathbb{S}$-unit lattice with high probability; this final statement is the object of Theorem 18.14.

Intuition. The intuition why this approach (of letting the input be Gaussian distributed) works is already sketched in Section 12.4. Another way of looking at it is by seeing Algorithm 3 as a specific CVP-algorithm for the logarithmic $\mathbb{S}$-unit lattice $\mathrm{Log}_{\mathbb{S}}(\mathcal{O}_{K,\mathbb{S}}^{\times})$, given an input vector $\mathbf{a}$ in $\mathrm{Div}_{K,\mathbb{S}}$.

As a CVP-algorithm, Algorithm 3 outputs a $\mathrm{Log}_{\mathbb{S}}(\alpha)$ 'close' to $\mathbf{a} \in \mathrm{Div}_{K,\mathbb{S}}$, so one can imagine that the output only differs slightly from the input. If then the input element $\mathbf{a} \in \mathrm{Div}_{K,\mathbb{S}}$ is distributed according to a sufficiently wide Gaussian, and this specific CVP output of Algorithm 3 is sufficiently close to $\mathbf{a}$, this CVP output distribution must be close to a Gaussian as well, but instead its support is on $\mathrm{Log}_{\mathbb{S}}(\mathcal{O}_{K,\mathbb{S}}^{\times})$. For such Gaussian distributions over lattices one can show that sufficiently many samples thereof generate the entire lattice (see Proposition 18.2).

18.2. **Obtaining a generating set of a lattice.** Given a certain distribution ψ on a lattice Λ, this section is about how many samples one has to draw from ψ to obtain a generating set of the lattice Λ. Of course, this depends on the distribution at hand: for example, the distribution that always outputs $0 \in \Lambda$ would never yield a generating set of the lattice Λ, no matter how many samples one takes.

For a distribution on a lattice Λ it it is sufficient to consider two particular properties to analyze the number of samples required to draw in order to obtain a generating set of Λ. Those two properties are *evenly distributedness*, which measures the maximum weight of the distribution ψ on strict sublattices of Λ, and *concentratedness*, which measures how much weight of the distribution ψ is (relatively) close to the origin. This is formalized in the following definition and proposition.

Definition 18.1. *Let $\Lambda \subset \mathbb{R}^n$ be a full-rank lattice, and ψ be a distribution on Λ. The distribution ψ is called* p-evenly distributed *if $\Pr_{x \leftarrow \psi}[x \in \Lambda'] \leq p$ for any proper sublattice $\Lambda' \subsetneq \Lambda$. It is called* (R, q)-concentrated *if $\Pr_{x \leftarrow \psi}[\|v\| \geq R] \leq q$.*

Proposition 18.2 ([13, Lemma 5])**.** *Let $\Lambda \subset \mathbb{R}^n$ be a full-rank lattice, and ψ be a distribution on Λ. Suppose that ψ is p-evenly distributed and (R, q)-concentrated for $R \geq \mathrm{Vol}(\Lambda)^{1/n}$. Denote by S the random variable defined as the number of samples drawn from ψ until these samples generate Λ. Then, for any $\alpha > 0$,*

$$\Pr\left[S > (2+\alpha)\frac{t+n}{1-p-q}\right] \leq e^{-\alpha(t+n)/2},$$

where $t = n\log_2(R) - \log_2(\mathrm{Vol}\,\Lambda) \geq 0$.

18.3. **The discrete Gaussian over a lattice is evenly distributed.** In the following two lemmas, we show that a Gaussian distribution over a lattice with an arbitrary center point is p-evenly distributed. This fact is used to show that the output of the sampling algorithm with a (continuous) Gaussian input is p-evenly distributed as well (see Lemma 18.11).

Recall that $g_\varsigma(x) := e^{-\pi\|x\|^2/\varsigma^2}$ is the Gaussian function, whereas $\mathcal{G}_{X,\varsigma}$ is the Gaussian distribution over X (where the definition depends on whether X is discrete or continuous, see Section 2.9).

Lemma 18.3. *For any lattice $\Lambda \subseteq V$ (where V is a Euclidean space) any $\varsigma > 0$ and any $t, w \in V$, we have*

$$g_\varsigma(\Lambda + t + w) + g_\varsigma(\Lambda + t - w) \geq 2g_\varsigma(w)g_\varsigma(\Lambda + t).$$

Proof. This lemma is a simple generalization of [41, Claim 2.10], and we follow the same strategy:

$$\begin{aligned} g_\varsigma(\Lambda + t + w) + g_\varsigma(\Lambda + t - w) &= \sum_{x \in \Lambda + t}\left(e^{-\pi\|x+w\|^2/\varsigma^2} + e^{-\pi\|x-w\|^2/\varsigma^2}\right) \\ &= 2e^{-\pi\|w\|^2/\varsigma^2}\sum_{x \in \Lambda + t}\left(e^{-\pi\|x\|^2/\varsigma^2}\cosh(2\pi\langle x, w\rangle/\varsigma^2)\right) \\ &\geq 2g_\varsigma(w)g_\varsigma(\Lambda + t), \end{aligned}$$

where the last inequality follows from $\cosh(\alpha) \geq 1$ for any real α. □

Lemma 18.4. *Let $\Lambda \subseteq V$ be a lattice and V an Euclidean space, $t \in V$ and $\varsigma \geq c \cdot \varrho(\Lambda)$ for some $c > 0$ (see Definition 2.2). Then the discrete Gaussian distribution on Λ with parameter ς and centered at t is p-evenly distributed with $p = \frac{1}{1+e^{-\pi c^{-2}}}$. Concretely, for any strict sublattice $\Lambda' \subsetneq \Lambda$*

$$\frac{g_\varsigma(\Lambda' + t)}{g_\varsigma(\Lambda + t)} \leq \frac{1}{1 + e^{-\pi c^{-2}}}.$$

Proof. Let $\Lambda' \subsetneq \Lambda$ be a sub-lattice of Λ and let $w \in \Lambda \setminus \Lambda'$. Then, by Lemma 18.3,

$$\begin{aligned} g_\varsigma(\Lambda + t) &\geq g_\varsigma(\Lambda' + t) + \frac{g_\varsigma(\Lambda' + t + w) + g_\varsigma(\Lambda' + t - w)}{2} \\ &\geq (1 + g_\varsigma(w))g_\varsigma(\Lambda' + t). \end{aligned}$$

Writing $\mathcal{G}_{\Lambda+t,\varsigma}$ for the Gaussian distribution on $\Lambda + t$ with parameter ς, we have,

$$\Pr_{x \leftarrow \mathcal{G}_{\Lambda+t,\varsigma}}[x \in \Lambda' + t] = \frac{g_\varsigma(\Lambda' + t)}{g_\varsigma(\Lambda + t)} \leq \frac{1}{1 + g_\varsigma(w)}.$$

Since $\Lambda' \subsetneq \Lambda$, we know from the definition of $\varrho(\Lambda)$ that there exists $w \in \Lambda \setminus \Lambda'$ such that $\|w\| \leq \varrho(\Lambda) \leq \varsigma/c$, hence $g_\varsigma(w) \geq \exp\left(-\pi c^{-2}\right)$, proving the lemma. □

18.4. **The (semi-)discrete Gaussian over** $\mathrm{Div}_{K,\mathbb{S}}$**.** We introduce the following notation for the discretization of the $\mathbb{S}$-divisor group as in Definition 15.2. It is needed in Algorithm 4, as this algorithm samples a Gaussian over $\mathrm{Div}_{K,\mathbb{S}}$ and feeds it to Algorithm 3. As $\mathrm{Div}_{K,\mathbb{S}}$ is partially continuous (for which a finite computer cannot sample), we thus need this discretized $\mathrm{Div}_{K,\mathbb{S},N}$ version of $\mathrm{Div}_{K,\mathbb{S}}$.

Notation 18.5 (Discretized $\mathbb{S}$-divisor group)**.** *For a set of prime ideals $\mathbb{S}$, we denote by $\mathrm{Div}_{K,\mathbb{S},N} \subseteq \mathrm{Div}_{K,\mathbb{S}}$ the* discrete *subgroup of the restricted divisor group, for which $x_\nu \in \frac{1}{N}\mathbb{Z}$ for all places ν. In other words, any element of $\mathrm{Div}_{K,\mathbb{S},N}$ can be written as*

$$\mathbf{a} = \sum_{\mathfrak{p} \in \mathbb{S}} a_\mathfrak{p} (\!|\mathfrak{p}|\!) + \sum_\nu x_\nu (\!|\nu|\!) \qquad \textit{(with } x_\nu \in \tfrac{1}{N}\mathbb{Z}\textit{)}$$

Remark 18.6. *In Algorithm 4, we need to compute an approximation of the discrete Gaussian over the group $\mathrm{Div}_{K,\mathbb{S},N}$ (see Lemma 2.22). For this we need a $\mathbb{Z}$-basis of $\mathrm{Div}_{K,\mathbb{S},N}$, which is naturally given by $\{\frac{\mathbf{e}_\nu}{N} \mid \nu$ places of $K\} \cup \{\mathbf{e}_\mathfrak{p} \mid \mathfrak{p} \in \mathbb{S}\}$, where $\mathbf{e}_x$ is one on the coordinate x and zero elsewhere (where x can be a place or a prime). We call this basis $\mathbf{B}(\mathrm{Div}_{K,\mathbb{S},N})$.*

In order to define a Gaussian distribution over $\mathrm{Div}_{K,\mathbb{S}}$, we recall the distance notion on this space (see Section 2.8). Since the group $\mathrm{Div}_{K,\mathbb{S}}$ is partially discrete, namely, at the finite places, the Gaussian over this group is semi-discrete as well.

Definition 18.7 (The semi-discrete Gaussian over $\mathrm{Div}_{K,\mathbb{S}}$)**.** *The semi-discrete Gaussian distribution $\mathcal{G}_{\mathrm{Div}_{K,\mathbb{S}},\varsigma} \in L_1(\mathrm{Div}_{K,\mathbb{S}})$ is the distribution defined by*

$$\mathcal{G}_{\mathrm{Div}_{K,\mathbb{S}},\varsigma}[\mathbf{a}] := g_\varsigma(\mathbf{a}) \cdot \left(\int_{\mathbf{b} \in \mathrm{Div}_{K,\mathbb{S}}} g_\varsigma(\mathbf{b}) d\mathbf{b} \right)^{-1}$$

where $g_\varsigma(\mathbf{a}) = \exp(-\pi\|\mathbf{a}\|^2/\varsigma^2)$ (with the distance notion from Section 2.8) and where we use the shorthand notation

$$\int_{\mathbf{b} \in \mathrm{Div}_{K,\mathbb{S}}} g_\varsigma(\mathbf{b}) d\mathbf{b} := \sum_{(a_\mathfrak{p}) \in \mathbb{Z}^\mathbb{S}} \int_{(b_\nu)_\nu \in \mathrm{Log}(K_\mathbb{R}^\times)} g_\varsigma\Big(\sum_{\mathfrak{p} \in \mathbb{S}} a_\mathfrak{p} (\!|\mathfrak{p}|\!) + \sum_\nu b_\nu (\!|\nu|\!) \Big) db$$

Definition 18.8 (The discrete Gaussian over $\mathrm{Div}_{K,\mathbb{S},N}$)**.** *For $N \in \mathbb{N}_{>0}$ we define the discrete Gaussian over $\mathrm{Div}_{K,\mathbb{S},N}$ by the rule*

$$\mathcal{G}_{\mathrm{Div}_{K,\mathbb{S},N},\varsigma} := g_\varsigma(\mathbf{a}) \cdot \left(\sum_{\mathbf{b} \in \mathrm{Div}_{K,\mathbb{S},N}} g_\varsigma(\mathbf{b}) \right)^{-1}$$

where $g_\varsigma(\mathbf{a}) = \exp(-\pi\|\mathbf{a}\|^2/\varsigma^2)$ (with the distance notion from Section 2.8).

18.5. **Sampling a set of generators for the Log-$\mathbb{S}$-unit lattice.**

Algorithm 4 Computing a random relation of $\mathbb{S}$-units

```
Require: A number field $K$ , an LLL-reduced basis of $\mathcal{O}_K$, and a set $\mathbb{S}$ of prime ideals of $K$.
Ensure: An random element of $\mathrm{Log}_{\mathbb{S}}(\mathcal{O}_{K,\mathbb{S}}^{\times})$.
1: Let $\omega \in \mathbb{Z}_{\geq 1}$ be as in Definition 17.1 (with $\mathfrak{m}_0$, $\mathsf{b}$, and $x$ as in Algorithm 3).
2: Let $\tilde{\varrho} = \mathsf{poly}(\log |\Delta_K|, \max_{\mathfrak{p} \in \mathbb{S}} \log \mathcal{N}(\mathfrak{p}))$ the bound from Proposition 16.10, such that $\varrho(\mathrm{Log}_{\mathbb{S}}(\mathcal{O}_{K,\mathbb{S}}^{\times})) \leq \tilde{\varrho}$.
3: Put $\varsigma = 3 \cdot \max(\sqrt{\log(n_{\mathbb{R}} + n_{\mathbb{C}} + |\mathbb{S}|)}, \tilde{\varrho})$, and put $N := \omega^n \cdot \exp(11 + 16 \log^2 |\Delta_K| + 9n^2)$.
4: Sample, using Lemma 2.22 with $\varepsilon_{\mathcal{G}} := 1/100$, and the basis $\mathbf{B}(\mathrm{Div}_{K,\mathbb{S},N})$ described in Remark 18.6,
       $$\mathbf{a} = \sum_{\mathfrak{p} \in \mathbb{S}} a_{\mathfrak{p}} (\!|\mathfrak{p}|\!) + \sum_{\nu} b_{\nu} (\!|\nu|\!) \leftarrow \widehat{\mathcal{G}}_{\mathbf{B}(\mathrm{Div}_{K,\mathbb{S},N}),\varepsilon_{\mathcal{G}},\varsigma,0}, \qquad (116)$$
   put $z = \mathrm{Exp}_{\infty}(\mathbf{a}_{\infty})$ and compute a rational $y \in K_{\mathbb{R}}^{\times}$ such that $\|y - z\|_{\infty} \leq \min_{\nu} z_{\nu}/N$.
5: Put $\mathfrak{a} = \prod_{\mathfrak{p} \in \mathbb{S}} \mathfrak{p}^{a_{\mathfrak{p}}} = \mathrm{Exp}_{\mathrm{f}}(\mathbf{a}_{\mathrm{f}})$.
6: Apply Algorithm 3 with $\mathfrak{a} \in \mathcal{I}_K$, $y \in K_{\mathbb{R}}^{\times}$, and $\mathbb{S}$, yielding $(\alpha, (v_{\mathfrak{p}})_{\mathfrak{p} \in \mathbb{S}})$.
7: return $(-(v_{\mathfrak{p}} + a_{\mathfrak{p}})_{\mathfrak{p} \in \mathbb{S}}, \mathrm{Log}(\alpha))$ where the $a_{\mathfrak{p}}$ are from line 4.
```

18.5.1. *Algorithm 4: Algorithm 3 with Gaussian input.* In this section, we prove that the output distribution of Algorithm 4 (which is essentially Algorithm 3 with input $\mathbf{a} \in \mathrm{Div}_{K,\mathbb{S},N}$ following a (discrete) Gaussian distribution), is evenly distributed and concentrated on the Log-$\mathbb{S}$-unit lattice (for certain adequate parameters). Hence, by repeating Algorithm 4 sufficiently many times, one obtains a generating set of the Log-$\mathbb{S}$-unit lattice, which is made precise in the later Theorem 18.14.

Lemma 18.9. *Assume that $\mathbb{S}$ contains all prime ideals coprime to $\mathfrak{m}_0$ of norm $\leq B_{\max}$ (where $B_{\max}$ is defined as in Theorem 17.7). Then, on input $\mathbb{S}$, Algorithm 4 runs in expected time*

$$\mathsf{poly}(\log |\Delta_K|, \varsigma, \mathrm{size}(\mathbb{S})) + T,$$

where T is an upper bound on the expected time of Algorithm 3 on input $\mathfrak{a}$ and y of size $\mathrm{size}(\mathfrak{a}) \leq \mathsf{poly}(\varsigma, \mathrm{size}(\mathbb{S}), \log |\Delta_K|)$ and $\mathrm{size}(y) \leq O(\varsigma n \cdot (\mathrm{size}(\mathbb{S}) + \log |\Delta_K|))$.

Proof. Apart from the line 6, in which Algorithm 3 is called, Algorithm 4 only requires computation power during the sampling of the approximate discrete Gaussian and the computation of $y \in K_{\mathbb{R}}^{\times}$. By Lemma 2.22, and denoting $n_0 = |\mathbb{S}| + \mathfrak{r} + 1 = |\mathbb{S}| + n_{\mathbb{R}} + n_{\mathbb{C}} = \dim(\mathrm{Div}_{K,\mathbb{S},N})$, the sampling of the approximate discrete Gaussian takes time (since $\varepsilon_{\mathcal{G}} := 1/100$) polynomial in the size of the input, which can be given by the size of the lattice basis of $\mathrm{Div}_{K,\mathbb{S},N}$, which is $\mathsf{poly}(n_0)$. Note that $\max_{\mathbf{b} \in \mathbf{B}(\mathrm{Div}_{K,\mathbb{S},N})} \|\mathbf{b}\| = 1$, so, by choice of ς, we have $\varsigma \geq \sqrt{\frac{\log(1/\varepsilon_{\mathcal{G}}) + 2\log(n_0) + 3}{\pi}} \cdot \max_{\mathbf{b} \in \mathbf{B}(\mathrm{Div}_{K,\mathbb{S},N})} \|\mathbf{b}\|$ as required to apply Lemma 2.22. The run time of the computation of $y \in K_{\mathbb{R}}^{\times}$ depends on the minimum of z_{ν}/N and the maximum of z_{ν}. We have that $|b_{\nu}| \leq \varsigma \sqrt{n_0 \log(2n_0^2/\varepsilon_{\mathcal{G}})}$ by Lemma 2.22 (with $\varepsilon_{\mathcal{G}} := 1/100$). Hence, in the worst case, we must approximate $y \approx z = \mathrm{Exp}_{\infty}(\mathbf{a}_{\infty}) = (e^{b_{\nu}})_{\nu}$ up to precision

$\exp(-\varsigma \cdot \sqrt{n_0 \cdot \log(2n_0^2/\varepsilon_{\mathcal{G}})})/N$. For the maximum of z_ν, a similar reasoning tells us that $\max_\nu z_\nu \leq \exp(\varsigma \cdot \sqrt{n_0 \cdot \log(2n_0^2/\varepsilon_{\mathcal{G}})})$.

Since we have $\log N = O(\log|\Delta_K|)$ and $\sqrt{n_0 \log(2n_0^2/\varepsilon_{\mathcal{G}})} = O(\text{size}(\mathbb{S}) + \log|\Delta_K|)$, the approximation $y \approx z$ required in line 4 can be computed within time $\mathsf{poly}(\varsigma, \text{size}(\mathbb{S}), \log|\Delta_K|)$. This reasoning also shows that $\text{size}(y) \leq O(\varsigma n \cdot (\text{size}(\mathbb{S}) + \log|\Delta_K|))$. Additionally, using the same bound of Lemma 2.22, the size of $\mathfrak{a}$ must also be bounded by $\mathsf{poly}(\varsigma, \text{size}(\mathbb{S}), \log|\Delta_K|)$. □

18.5.2. *The output distribution of Algorithm 4 is concentrated.* In the next lemma, we prove that the output distribution of Algorithm 4 is concentrated.

Lemma 18.10. *The output distribution $\bar{\mathcal{D}}$ of Algorithm 4 is $(R_0, 0)$-concentrated for some*

$$R_0 = O(\log^2|\Delta_K|) + 3 \cdot \varsigma \cdot (|\mathbb{S}| + n_{\mathbb{R}} + n_{\mathbb{C}}).$$

Proof. The output distribution of Algorithm 4 depends on an approximate discrete Gaussian from line 4, which is computed using Lemma 2.22. From that lemma, denoting $n_0 = |\mathbb{S}| + n_{\mathbb{R}} + n_{\mathbb{C}} = \dim(\text{Div}_{K,\mathbb{S},N})$ follows that

$$\|\mathbf{a}\| \leq \varsigma \cdot \sqrt{n_0 \log(2n_0^2/\varepsilon_{\mathcal{G}}))} \leq 3\varsigma n_0$$

with ς from line 3 and $\varepsilon_{\mathcal{G}} = 1/100$ (see line 4 of Algorithm 4). Here we use that $\sqrt{x \log(2x^2/\varepsilon_{\mathcal{G}})} \leq 3x$ for $x \geq 1$ and $\varepsilon_{\mathcal{G}} = 1/100$.

But since Algorithm 3 satisfies property (iii) from Lemma 17.8, we know that

$$\|\text{Log}_{\mathbb{S}}(\alpha)\| \leq O(\log^2|\Delta_K|) + \|\mathbf{a}\| \leq O(\log^2|\Delta_K|) + 3 \cdot \varsigma \cdot n_0.$$

Hence, there exists a $R_0 = O(\log^2|\Delta_K|) + 3 \cdot \varsigma \cdot (|\mathbb{S}| + n_{\mathbb{R}} + n_{\mathbb{C}})$ for which no α with $\|\text{Log}_{\mathbb{S}}(\alpha)\| > R_0$ can be the output of Algorithm 4. □

18.5.3. *The output distribution of Algorithm 4 is evenly distributed.* We now prove that the output distribution of Algorithm 4 is evenly distributed on the logarithmic $\mathbb{S}$-unit lattice.

The strategy to show evenly distributedness is more convoluted than was concentratedness. We start with defining the 'continuous version of Algorithm 4', in which in line 4 is used a *semi-discrete Gaussian* (as in Definition 18.7) instead of a fully discrete Gaussian (as in Definition 18.8). More precisely, the 'continuous version of Algorithm 4' is Algorithm 4 in which line 4 is replaced with

$$\mathbf{a} = \sum_{\mathfrak{p} \in \mathbb{S}} a_{\mathfrak{p}} \llparenthesis \mathfrak{p} \rrparenthesis + \sum_{\nu} b_\nu \llparenthesis \nu \rrparenthesis \leftarrow \mathcal{G}_{\text{Div}_{K,\mathbb{S}},\varsigma}, \text{ and put } y = \text{Exp}_\infty(\mathbf{a}_\infty).$$

We will show in Lemma 18.13 that the output distribution of the 'continuous version' and the ordinary version of Algorithm 4 are close in statistical distance.

In the present section, we focus on the following step: proving that the output distribution of 'continuous version of Algorithm 4' is evenly distributed for certain parameters. By the closeness of the 'continuous version' and the ordinary version of Algorithm 4, the ordinary must then also be evenly distributed.

Lemma 18.11. *Assume that $\mathbb{S}$ generates the ray class group $\text{Cl}_K^{\mathfrak{m}_0}$ of K and does not contain any prime dividing $\mathfrak{m}_0$. Then the output distribution of Algorithm 4 is $2/3$-evenly distributed on $\text{Log}_{\mathbb{S}}(\mathcal{O}_{K,\mathbb{S}}^\times)$.*

Proof. We will later show, in Lemma 18.13, that Algorithm 4 is $1/50$-close in statistical distance to the 'continuous version of Algorithm 4'. We show in the current proof that this 'continuous version' is $6/10$-evenly distributed on $\mathrm{Log}_{\mathbb{S}}(\mathcal{O}_{K,\mathbb{S}}^{\times})$, and hence the ordinary version is then $2/3$-evenly distributed on $\mathrm{Log}_{\mathbb{S}}(\mathcal{O}_{K,\mathbb{S}}^{\times})$, since $6/10 + 1/50 < 2/3$.

The remainder of this proof is then devoted to showing that the 'continuous version of Algorithm 4' is $6/10$-evenly distributed on $\mathrm{Log}_{\mathbb{S}}(\mathcal{O}_{K,\mathbb{S}}^{\times})$.

The output distribution $\bar{\mathcal{D}}$ of the 'continuous version of Algorithm 4' is supported on $\mathrm{Log}_{\mathbb{S}}(\mathcal{O}_{K,\mathbb{S}}^{\times})$ and given by the rule, for $\alpha \in \mathcal{O}_{K,\mathbb{S}}^{\times}$,

$$\bar{\mathcal{D}}[\mathrm{Log}_{\mathbb{S}}(\alpha)] = \int_{\mathbf{a} \in \mathrm{Div}_{K,\mathbb{S}}} \bar{\mathcal{D}}_{\mathbf{a}}[\mathrm{Log}_{\mathbb{S}}(\alpha)] \cdot \mathcal{G}_{\varsigma}(\mathbf{a}) d\mathbf{a}.$$

To prove that this distribution is p-evenly distributed (with $p = 6/10$), we need to show that $\bar{\mathcal{D}}[M] = \sum_{m \in M} \bar{\mathcal{D}}[m] \leq p$ for every strict sublattice $M \subsetneq \mathrm{Log}_{\mathbb{S}}(\mathcal{O}_{K,\mathbb{S}}^{\times})$.

We first prove this for the simpler case where the modulus is trivial, i.e., $\mathfrak{m}_0 = \mathcal{O}_K$; in which case we have $K^{\mathfrak{m}_0,1} = K^*$, and the shifting property (Property (ii) of Lemma 17.8) holds for any $\alpha \in K^*$, so in particular for $\alpha \in \mathcal{O}_{K,\mathbb{S}}^{\times}$.

So, pick any strict sublattice $M \subsetneq \mathrm{Log}_{\mathbb{S}}(\mathcal{O}_{K,\mathbb{S}}^{\times})$. Then, by Property (ii) of Lemma 17.8, and, subsequently, by change of variables, we have

$$\begin{aligned}
\sum_{\mathrm{Log}_{\mathbb{S}}(\mu) \in M} \bar{\mathcal{D}}[\mathrm{Log}_{\mathbb{S}}(\mu)] &= \int_{\mathbf{a} \in \mathrm{Div}_{K,\mathbb{S}}} \sum_{\mathrm{Log}_{\mathbb{S}}(\mu) \in M} \bar{\mathcal{D}}_{\mathbf{a}}[\mathrm{Log}_{\mathbb{S}}(\mu)] \cdot \mathcal{G}_{\varsigma}(\mathbf{a}) d\mathbf{a} \\
&= \int_{\mathbf{a} \in \mathrm{Div}_{K,\mathbb{S}}} \sum_{\mathrm{Log}_{\mathbb{S}}(\mu) \in M} \bar{\mathcal{D}}_{\mathbf{a} + \mathrm{Log}_{\mathbb{S}}(\mu)}[0] \cdot \mathcal{G}_{\varsigma}(\mathbf{a}) d\mathbf{a} \\
&= \int_{\mathbf{a} \in \mathrm{Div}_{K,\mathbb{S}}} \sum_{\mathrm{Log}_{\mathbb{S}}(\mu) \in M} \bar{\mathcal{D}}_{\mathbf{a}}[0] \cdot \mathcal{G}_{\varsigma}(\mathbf{a} - \mathrm{Log}_{\mathbb{S}}(\mu)) d\mathbf{a} \\
&= \int_{\mathbf{a} \in \mathrm{Div}_{K,\mathbb{S}}} \bar{\mathcal{D}}_{\mathbf{a}}[0] \cdot \mathcal{G}_{\varsigma}(\mathbf{a} + M) d\mathbf{a} \qquad (117)
\end{aligned}$$

Using the assumption that $\mathbb{S}$ generates the full class group, we take a fundamental domain F of $\mathrm{Log}_{\mathbb{S}}(\mathcal{O}_{K,\mathbb{S}}^{\times})$ in $\mathrm{Div}_{K,\mathbb{S}}^{0}$, in order to decompose $\mathrm{Div}_{K,\mathbb{S}} = \mathbb{R} \cdot \mathbf{1} + \mathrm{Log}_{\mathbb{S}}(\mathcal{O}_{K,\mathbb{S}}^{\times}) + F$, where $\mathbf{1} = \sum_{\nu} n_{\nu} \cdot (\!|\nu|\!) \in \mathrm{Log}(K_{\mathbb{R}}^{\times})$ is the vector consisting n_{ν}, which equals 2 if ν is complex and 1 otherwise. This vector corresponds to the freedom of the norm (or rather, degree) in the space $\mathrm{Div}_{K,\mathbb{S}}$, compared to $\mathrm{Div}_{K,\mathbb{S}}^{0}$.

We can then rewrite any integral over $\mathrm{Div}_{K,\mathbb{S}}$ with integrand $h(\mathbf{a})$ as

$$\int_{\mathbf{a} \in \mathrm{Div}_{K,\mathbb{S}}} h(\mathbf{a}) d\mathbf{a} = \int_{t \in \mathbb{R}} \int_{\mathbf{a} \in F} \sum_{\mathrm{Log}_{\mathbb{S}}(\beta) \in \mathrm{Log}_{\mathbb{S}}(\mathcal{O}_{K,\mathbb{S}}^{\times})} h(\mathbf{a} + \mathrm{Log}_{\mathbb{S}}(\beta) + t \cdot \mathbf{1}) d\mathbf{a} dt$$

We apply this to Equation (117), we use again Lemma 17.8(ii) to write $\bar{\mathcal{D}}_{\mathbf{a} + \mathrm{Log}_{\mathbb{S}}(\beta)}[0] = \bar{\mathcal{D}}_{\mathbf{a}}[\mathrm{Log}_{\mathbb{S}}(\beta)]$ and we use the fact that $\bar{\mathcal{D}}_{\mathbf{a} + t\mathbf{1}} = \bar{\mathcal{D}}_{\mathbf{a}}$, since Algorithm 3 does not depend on the norm of the input (since `Sample` does not, see Theorem 13.2). Then, Equation (117) equals

$$= \int_{t \in \mathbb{R}} \int_{\mathbf{a} \in F} \sum_{\mathrm{Log}_{\mathbb{S}}(\beta) \in \mathrm{Log}_{\mathbb{S}}(\mathcal{O}_{K,\mathbb{S}}^{\times})} \bar{\mathcal{D}}_{\mathbf{a}}[\mathrm{Log}_{\mathbb{S}}(\beta)] \cdot \mathcal{G}_{\varsigma}(\mathbf{a} + t\mathbf{1} + \mathrm{Log}_{\mathbb{S}}(\beta) + M) d\mathbf{a} dt \qquad (118)$$

By Hölder's inequality, the positivity of all arguments and the fact that (by Lemma 17.8(i))

$$\sum_{\mathrm{Log}_{\mathbb{S}}(\beta)\in\mathrm{Log}_{\mathbb{S}}(\mathcal{O}_{K,\mathbb{S}}^{\times})} \bar{\mathcal{D}}_{\mathbf{a}}[\mathrm{Log}_{\mathbb{S}}(\beta)] = 1,$$

we can bound Equation (118) by

$$\leq \int_{t\in\mathbb{R}}\int_{\mathbf{a}\in F} \max_{\mathrm{Log}_{\mathbb{S}}(\beta)\in\mathrm{Log}_{\mathbb{S}}(\mathcal{O}_{K,\mathbb{S}}^{\times})} \mathcal{G}_{\varsigma}(\mathbf{a}+t\mathbf{1}+\mathrm{Log}_{\mathbb{S}}(\beta)+M)d\mathbf{a}dt \tag{119}$$

$$\leq \int_{t\in\mathbb{R}}\int_{\mathbf{a}\in F} \frac{\mathcal{G}_{\varsigma}(\mathbf{a}+t\mathbf{1}+\mathrm{Log}_{\mathbb{S}}(\mathcal{O}_{K,\mathbb{S}}^{\times}))}{1+e^{-\pi/c^2}} d\mathbf{a}dt \leq \frac{1}{1+e^{-\pi/c^2}}. \tag{120}$$

whenever $\varsigma \geq c\cdot\varrho(\mathrm{Log}_{\mathbb{S}}(\mathcal{O}_{K,\mathbb{S}}^{\times}))$. Here we apply Lemma 18.4, which gives us

$$\mathcal{G}_{\varsigma}(\mathbf{a}+t\mathbf{1}+\mathrm{Log}_{\mathbb{S}}(\beta)+M) \leq \frac{\mathcal{G}_{\varsigma}(\mathbf{a}+t\mathbf{1}+\mathrm{Log}_{\mathbb{S}}(\beta)+\mathrm{Log}_{\mathbb{S}}(\mathcal{O}_{K,\mathbb{S}}^{\times}))}{1+e^{-\pi/c^2}} = \frac{\mathcal{G}_{\varsigma}(\mathbf{a}+t\mathbf{1}+\mathrm{Log}_{\mathbb{S}}(\mathcal{O}_{K,\mathbb{S}}^{\times}))}{1+e^{-\pi/c^2}}.$$

By picking $c = 3$, it holds that $\varsigma \geq c\cdot\varrho(\mathrm{Log}_{\mathbb{S}}(\mathcal{O}_{K,\mathbb{S}}^{\times}))$ (see line 3 of Algorithm 4) and that $(1+e^{-\pi/c^2})^{-1} \leq 6/10$.

We now explain how to amend the proof for the case where $\mathfrak{m}_0 \neq \mathcal{O}_K$, i.e., a non-trivial modulus. In this case, the shifting property (Property (ii) of Lemma 17.8) holds for $\alpha \in K^{\mathfrak{m}_0,1}$ only.

Write $M_{\mathfrak{m}_0} = \mathrm{Log}_{\mathbb{S}}(K^{\mathfrak{m}_0,1}) \cap M$ and put $\{\mathrm{Log}_{\mathbb{S}}(\mu_1),\ldots,\mathrm{Log}_{\mathbb{S}}(\mu_w)\} \subseteq M$ for a set of representatives of $M/M_{\mathfrak{m}_0}$ (which is finite by the fact that $K/K^{\mathfrak{m}_0,1}$ is finite). By similar computations as in Equation (117), we obtain

$$\begin{aligned}\sum_{\mathrm{Log}_{\mathbb{S}}(\mu)\in M} \bar{\mathcal{D}}[\mathrm{Log}_{\mathbb{S}}(\mu)] &= \int_{\mathbf{a}\in\mathrm{Div}_{K,\mathbb{S}}} \sum_{\mathrm{Log}_{\mathbb{S}}(\mu)\in M_{\mathfrak{m}_0}} \sum_{i=1}^{w} \bar{\mathcal{D}}_{\mathbf{a}}[\mathrm{Log}_{\mathbb{S}}(\mu_i)+\mathrm{Log}_{\mathbb{S}}(\mu)]\cdot\mathcal{G}_{\varsigma}(\mathbf{a})d\mathbf{a}\\ &= \int_{\mathbf{a}\in\mathrm{Div}_{K,\mathbb{S}}} \sum_{\mathrm{Log}_{\mathbb{S}}(\mu)\in M_{\mathfrak{m}_0}} \sum_{i=1}^{w} \bar{\mathcal{D}}_{\mathbf{a}+\mathrm{Log}_{\mathbb{S}}(\mu)}[\mathrm{Log}_{\mathbb{S}}(\mu_i)]\cdot\mathcal{G}_{\varsigma}(\mathbf{a})d\mathbf{a}\\ &= \int_{\mathbf{a}\in\mathrm{Div}_{K,\mathbb{S}}} \sum_{\mathrm{Log}_{\mathbb{S}}(\mu)\in M_{\mathfrak{m}_0}} \sum_{i=1}^{w} \bar{\mathcal{D}}_{\mathbf{a}}[\mathrm{Log}_{\mathbb{S}}(\mu_i)]\cdot\mathcal{G}_{\varsigma}(\mathbf{a}-\mathrm{Log}_{\mathbb{S}}(\mu))d\mathbf{a}\\ &= \int_{\mathbf{a}\in\mathrm{Div}_{K,\mathbb{S}}} \sum_{i=1}^{w} \bar{\mathcal{D}}_{\mathbf{a}}[\mathrm{Log}_{\mathbb{S}}(\mu_i)]\cdot\mathcal{G}_{\varsigma}(\mathbf{a}+M_{\mathfrak{m}_0})d\mathbf{a} \qquad (121)\end{aligned}$$

Now we use that $\mathbb{S}$ and $\mathfrak{m}_0$ do not share primes, and that $\mathbb{S}$ generates the ray class group $\mathrm{Cl}_K^{\mathfrak{m}_0}$, so that we can split up the space

$$\mathrm{Div}_{K,\mathbb{S}} = \mathbb{R}\cdot\mathbf{1} + \mathrm{Log}_{\mathbb{S}}(\mathcal{O}_{K,\mathbb{S}}^{\times}) \cap \mathrm{Log}_{\mathbb{S}}(K^{\mathfrak{m}_0,1}) + F_{\mathfrak{m}_0}, \tag{122}$$

with a compact $F_{\mathfrak{m}_0}$. The set $\mathbb{S}$ and $\mathfrak{m}_0$ not sharing primes is essential for the fundamental domain $F_{\mathfrak{m}_0}$ to be compact (which is required for the current proof); indeed, if $\mathbb{S}$ and $\mathfrak{m}_0$ *would* share a prime, $\mathrm{Log}_{\mathbb{S}}(\mathcal{O}_{K,\mathbb{S}}^{\times}) \cap \mathrm{Log}_{\mathbb{S}}(K^{\mathfrak{m}_0,1})$ would be of rank stictly less than that of $\mathrm{Div}_{K,\mathbb{S}}$, resulting in a $F_{\mathfrak{m}_0}$ in Equation (122) of infinite volume.

For $\mathbb{S}$ and $\mathfrak{m}_0$ not sharing a prime divisor, one could construct $F_{\mathfrak{m}_0}$ as follows. Pick a set of ideals $A = \{\mathfrak{a}_1,\ldots,\mathfrak{a}_w\}$ (with all prime factors in $\mathbb{S}$) that are representatives of the ray class group $\mathrm{Cl}_K^{\mathfrak{m}_0}$. Next, define V to be the Voronoi fundamental domain of $\mathrm{Log}(K^{\mathfrak{m}_0,1} \cap \mathcal{O}_K^{\times}) \subseteq H$. Then $F_{\mathfrak{m}_0} = \{(\!|\mathfrak{a}|\!) + x \mid \mathfrak{a} \in A \text{ and } x \in V\}$ is an example of a fundamental domain satisfying Equation (122).

Similar computations as in Equation (118) then show that Equation (121) equals (where $\text{Log}_{\mathbb{S}}(\beta)$ sums over $\text{Log}_{\mathbb{S}}(\mathcal{O}_{K,\mathbb{S}}^{\times} \cap K^{\mathfrak{m}_0,1})$)

$$= \int_{t\in\mathbb{R}} \int_{\mathbf{a}\in F_{\mathfrak{m}_0}} \sum_{\text{Log}_{\mathbb{S}}(\beta)} \sum_{i=1}^{w} \bar{\mathcal{D}}_{\mathbf{a}}[\text{Log}_{\mathbb{S}}(\beta) + \text{Log}_{\mathbb{S}}(\mu_i)] \cdot \mathcal{G}_{\varsigma}(\mathbf{a} + t\mathbf{1} + \text{Log}_{\mathbb{S}}(\beta) + M_{\mathfrak{m}_0}) d\mathbf{a}dt \tag{123}$$

Since $\{\text{Log}_{\mathbb{S}}(\mu_1), \ldots, \text{Log}_{\mathbb{S}}(\mu_w)\}$ are different representatives in $\text{Log}_{\mathbb{S}}(\mathcal{O}_{K,\mathbb{S}}^{\times}) / \text{Log}_{\mathbb{S}}(\mathcal{O}_{K,\mathbb{S}}^{\times} \cap K^{\mathfrak{m}_0,1})$ (indeed, if $\text{Log}_{\mathbb{S}}(\mu_1) - \text{Log}_{\mathbb{S}}(\mu_2) \in \text{Log}_{\mathbb{S}}(K^{\mathfrak{m}_0,1})$, we have $\text{Log}_{\mathbb{S}}(\mu_1) - \text{Log}_{\mathbb{S}}(\mu_2) \in \text{Log}_{\mathbb{S}}(K^{\mathfrak{m}_0,1}) \cap M$, contradiction), we can apply Hölders inequality again, yielding an upper bound of

$$\int_{t\in\mathbb{R}} \int_{\mathbf{a}\in F} \max_{\text{Log}_{\mathbb{S}}(\beta) \in \text{Log}_{\mathbb{S}}(\mathcal{O}_{K,\mathbb{S}}^{\times} \cap K^{\mathfrak{m}_0,1})} \mathcal{G}_{\varsigma}(\mathbf{a} + t\mathbf{1} + \text{Log}_{\mathbb{S}}(\beta) + M_{\mathfrak{m}_0}) d\mathbf{a}dt \leq \frac{1}{1 + e^{\pi/c^2}}$$

whenever $\varsigma \geq c \cdot \varrho(\text{Log}_{\mathbb{S}}(\mathcal{O}_{K,\mathbb{S}}^{\times}))$, since $M_{\mathfrak{m}_0} + \text{Log}_{\mathbb{S}}(\mathcal{O}_{K,\mathbb{S}}^{\times} \cap K^{\mathfrak{m}_0,1})$ is a lattice in $\text{Log}_{\mathbb{S}}(\mathcal{O}_{K,\mathbb{S}}^{\times})$. □

18.5.4. *About the closeness of the 'continuous version of Algorithm 4'.* In this part, we show that the 'continuous version of Algorithm 4' and the ordinary one are close in statistical distance. Before we do that, we first need a lemma that shows that N, the discretization parameter in $\text{Div}_{K,\mathbb{S},N}$ in Algorithm 4 satisfies some bound.

Lemma 18.12. *Let* $r = 48 \cdot \omega \cdot \mathsf{b}^{2n/\mathsf{b}} \cdot n^{7/2} \cdot |\Delta_K|^{\frac{3}{2n}} \cdot \mathcal{N}(\mathfrak{m}_0)^{\frac{1}{n}}$ *as used in Algorithm 3. Then we have*

$$40000 \cdot |\Delta_K| \cdot r^n \cdot n^{5/2} \leq \omega^n \cdot \exp(11 + 16 \log^2 |\Delta_K| + 9n^2) = N,$$

where N *is as in line 3 in Algorithm 4.*

Proof. Let $r = 48 \cdot \omega \cdot \mathsf{b}^{2n/\mathsf{b}} \cdot n^{7/2} \cdot |\Delta_K|^{\frac{3}{2n}} \cdot \mathcal{N}(\mathfrak{m}_0)^{\frac{1}{n}}$, where $\mathfrak{m}_0$ and b are as chosen in Algorithm 3. The largest value $\mathcal{N}(\mathfrak{m}_0)$ can have is whenever $\mathfrak{m}_0$ is the product of the prime ideals of norm smaller than $x \leq \frac{3}{2} \log |\Delta_K|$ (by line 3 of Algorithm 3 and Minkowski's theorem, which tells that $\log |\Delta_K| \geq \log(\pi/2)n \geq 0.4n$). Hence, using Lemma 11.4 with $x \leq \frac{3}{2} \log |\Delta_K|$, using that $\sqrt{x}(\log(x)/(2\pi) + 2) \leq 3x$ and $\sqrt{x}(\log(x)^2/(8\pi) + 2 \leq 3x$ (for $x > 0.5$),

$$\log \mathcal{N}(\mathfrak{m}_0) \leq x + \sqrt{x} \left(\left(\frac{\log x}{2\pi} + 2 \right) \log |\Delta_K| + \left(\frac{(\log x)^2}{8\pi} + 2 \right) n \right) \tag{124}$$

$$\leq \underbrace{x}_{\leq \frac{3}{2} \log |\Delta_K|} + \underbrace{3x \log |\Delta_K|}_{\leq \frac{9}{2} \log^2 |\Delta_K|} + \underbrace{3xn}_{\leq \frac{9}{2} \log |\Delta_K| \cdot \frac{3}{2} \log |\Delta_K|} \leq 13 \log^2 |\Delta_K|. \tag{125}$$

By taking logarithms and using that $\log \mathcal{N}(\mathfrak{m}_0) \leq 13 \log^2 |\Delta_K|$, that $\log(\mathsf{b}^{2n^2/\mathsf{b}}) \leq n^2$, that $n^{7n/2} \leq e^{3n^2}$, and that $48^n \leq e^{4n^2}$, we have that

$$\begin{aligned} r^n &= (48 \cdot \omega \cdot \mathsf{b}^{2n/\mathsf{b}} \cdot n^{7/2} \cdot |\Delta_K|^{\frac{3}{2n}} \cdot \mathcal{N}(\mathfrak{m}_0)^{\frac{1}{n}})^n \\ &\leq \omega^n \cdot e^{4n^2 + n^2 + 3n^2} \cdot e^{\frac{3}{2} \log |\Delta_K| + 13 \log^2 |\Delta_K|} \leq \omega^n \cdot e^{8n^2} e^{15 \log^2 |\Delta_K|}. \end{aligned}$$

Hence, using $40000 \leq e^{11}$ and $n^{5/2} \leq e^{n^2}$, we obtain

$$40000 \cdot |\Delta_K| \cdot r^n \cdot n^{5/2} \leq \omega^n \cdot \exp(11 + 16 \log^2 |\Delta_K| + 9n^2) = N.$$

□

Recall that the 'continuous version of Algorithm 4' means: Algorithm 4 in which line 4 is replaced by

$$\mathbf{a} = \sum_{\mathfrak{p}\in\mathbb{S}} a_{\mathfrak{p}} (\!|\mathfrak{p}|\!) + \sum_{\nu} b_\nu (\!|\nu|\!) \leftarrow \mathcal{G}_{\mathrm{Div}_{K,\mathbb{S}},\varsigma}, \text{ and put } y = \mathrm{Exp}_\infty(\mathbf{a}_\infty).$$

Lemma 18.13. *The statistical distance of the output distributions of Algorithm 4 and the 'continuous version of Algorithm 4' is at most* $1/50$*.*

Proof. The output distribution of the 'continuous version of Algorithm 4' can be written as

$$\int_{\mathbf{a}\in\mathrm{Div}_{K,\mathbb{S}}} \bar{\mathcal{D}}_{\mathbf{a}}[-]\mathcal{G}_{\mathrm{Div}_{K,\mathbb{S}},\varsigma}(\mathbf{a})d\mathbf{a} \tag{126}$$

whereas the output distribution of the ordinary one can be written as

$$\sum_{\ddot{\mathbf{a}}\in\mathrm{Div}_{K,\mathbb{S},N}} \bar{\mathcal{D}}_{\tilde{\ddot{\mathbf{a}}}}[-]\hat{\mathcal{G}}_{\mathrm{Div}_{K,\mathbb{S},N},\varsigma}(\ddot{\mathbf{a}}),$$

which is $\varepsilon_{\mathcal{G}}$-close (see Lemma 2.22, with $\varepsilon_{\mathcal{G}} = 1/100$) to

$$\sum_{\ddot{\mathbf{a}}\in\mathrm{Div}_{K,\mathbb{S},N}} \bar{\mathcal{D}}_{\tilde{\ddot{\mathbf{a}}}}[-]\mathcal{G}_{\mathrm{Div}_{K,\mathbb{S},N},\varsigma}(\ddot{\mathbf{a}}), \tag{127}$$

where the approximation $\tilde{\ddot{\mathbf{a}}} \approx \ddot{\mathbf{a}}$ is caused by the computation of y in line 4. Writing F for a fundamental domain of $\mathrm{Div}_{K,\mathbb{S},N} \subseteq \mathrm{Div}_{K,\mathbb{S}}$ (i.e., $F + \mathrm{Div}_{K,\mathbb{S},N} = \mathrm{Div}_{K,\mathbb{S}}$ is a tiling), and taking into account the error of $\varepsilon_{\mathcal{G}} = 1/100$ caused by the Gaussian approximation (Lemma 2.22), the statistical distance between the continuous and ordinary variant of the algorithm can be estimated by (writing $|F|$ for the volume of F)

$$\varepsilon_{\mathcal{G}} + \frac{1}{2}\cdot \left\| \int_{\mathbf{a}\in F} \sum_{\ddot{\mathbf{a}}\in\mathrm{Div}_{K,\mathbb{S},N}} \bar{\mathcal{D}}_{\ddot{\mathbf{a}}+\mathbf{a}}[-]\mathcal{G}_{\mathrm{Div}_{K,\mathbb{S}},\varsigma}(\ddot{\mathbf{a}}+\mathbf{a}) - |F|^{-1}\bar{\mathcal{D}}_{\tilde{\ddot{\mathbf{a}}}}[-]\mathcal{G}_{\mathrm{Div}_{K,\mathbb{S},N},\varsigma}(\ddot{\mathbf{a}})d\mathbf{a} \right\|_1$$

By the trick $ab - a'b' = b(a-a') - a'(b'-b)$, the triangle inequality, and the fact that the ℓ_1 norm of a distribution is equal to 1, the quantity above can be bounded by

$$\varepsilon_{\mathcal{G}} + \underbrace{\frac{1}{2}\cdot \int_{\mathbf{a}\in F} \sum_{\ddot{\mathbf{a}}\in\mathrm{Div}_{K,\mathbb{S},N}} \|\bar{\mathcal{D}}_{\ddot{\mathbf{a}}+\mathbf{a}} - \bar{\mathcal{D}}_{\tilde{\ddot{\mathbf{a}}}}\|_1 \cdot \mathcal{G}_{\mathrm{Div}_{K,\mathbb{S}},\varsigma}(\ddot{\mathbf{a}}+\mathbf{a})d\mathbf{a}}_{(A)} \tag{128}$$

$$+ \underbrace{\frac{1}{2}\cdot \int_{\mathbf{a}\in F} \sum_{\ddot{\mathbf{a}}\in\mathrm{Div}_{K,\mathbb{S},N}} |\mathcal{G}_{\mathrm{Div}_{K,\mathbb{S}},\varsigma}(\ddot{\mathbf{a}}+\mathbf{a}) - |F|^{-1}\mathcal{G}_{\mathrm{Div}_{K,\mathbb{S},N},\varsigma}(\ddot{\mathbf{a}})|d\mathbf{a}}_{(B)} \tag{129}$$

By property Lemma 17.8(iv), we have that $\bar{\mathcal{D}}_{\mathbf{a}}$ is almost Lipschitz-continuous in $\mathbf{a}$,

$$\|\bar{\mathcal{D}}_{\ddot{\mathbf{a}}+\mathbf{a}} - \bar{\mathcal{D}}_{\tilde{\ddot{\mathbf{a}}}}\|_1 \leq 3\cdot|\Delta_K|\cdot r^n\cdot n^2\cdot\|\ddot{\mathbf{a}}+\mathbf{a}-\tilde{\ddot{\mathbf{a}}}\| + \tfrac{1}{200},$$

and $\|\ddot{\mathbf{a}}+\mathbf{a}-\tilde{\ddot{\mathbf{a}}}\| \leq \|\tilde{\ddot{\mathbf{a}}}-\ddot{\mathbf{a}}\| + \|\mathbf{a}\| \leq 2\sqrt{n}/N$ by construction[21] of $\tilde{\ddot{\mathbf{a}}} = \mathrm{Log}(y/z) + \ddot{\mathbf{a}}$ on line 4 and since $\max_{\mathbf{a}\in F}\|\mathbf{a}\| = \sqrt{n}/N$ for $F = \{\sum_\nu x_\nu (\!|\nu|\!) \in \mathrm{Div}_{K,\mathbb{S}} \mid x_\nu \in$

[21] We have $\|\mathrm{Log}(y\cdot z^{-1})\| \leq \sqrt{n}\max_\nu \log(y_\nu z_\nu^{-1}) \leq \sqrt{n}\max_\nu |y_\nu z_\nu^{-1} - 1| \leq \sqrt{n}(\min_\nu z_\nu)^{-1}\cdot \max_\nu |y_\nu - z_\nu| \leq \sqrt{n}/N$, by line 4 of Algorithm 4.

$[-1/2, 1/2)\}$. Therefore, we can bound part (A) of Equation (128) by (using Lemma 18.12)

$$(A) \leq \underbrace{\frac{6 \cdot |\Delta_K| \cdot r^n \cdot n^{5/2}}{N}}_{\leq 1/200} + 1/200 < 1/100.$$

For the bound on Equation (129), note that the sole continuity is on the $\mathrm{Log}(K_{\mathbb{R}}^\times)$-part of $\mathrm{Div}_{K,\mathbb{S}}$. Hence, part (B) in Equation (129) is equal to

$$(B) = \int_{\mathbf{a} \in F} \sum_{\ddot{\mathbf{a}} \in \frac{1}{N} \prod_\nu \mathbb{Z}} |\mathcal{G}_{\mathrm{Log}(K_{\mathbb{R}}^\times), \varsigma}(\ddot{\mathbf{a}} + \mathbf{a}) - |F|^{-1} \mathcal{G}_{\frac{1}{N} \prod_\nu \mathbb{Z}, \varsigma}(\ddot{\mathbf{a}})| d\mathbf{a}, \tag{130}$$

where we understand the inclusion $\frac{1}{N} \prod_\nu \mathbb{Z} \hookrightarrow \mathrm{Log}(K_{\mathbb{R}})$. Instantiating Lemma 9.13 with $\varepsilon = 1/700$ (loc. cit.), we can bound (B) via Equation (130) by $7\varepsilon = 1/100$. Here we use that $\varsigma > 1$, that $\lambda_n(\frac{1}{N} \prod_\nu \mathbb{Z}) = \frac{1}{N}$, and that (by Lemma 18.12) $N \geq 40000 n^2 \geq 5\pi \sqrt{n \log(4n/\varepsilon)} \varepsilon^{-1} n$, hence the assumptions of Lemma 9.13 are satisfied.

Concluding, the statistical distance between the continuous and ordinary variant of Algorithm 4, by using that $\varepsilon_{\mathcal{G}} = 1/100$, is at most $1/100 + 1/200 + 1/200 = 1/50$, since the respective summands in Equations (128) and (129) are bounded by $1/100, 1/200, 1/200$ respectively. □

18.6. **Conclusion: the number of samples that generate the full $\mathbb{S}$-unit lattice.** The concluding theorem of this section quantifies how many samples of the output distribution $\bar{\mathcal{D}}$ of Algorithm 4 one needs to draw in order to get a generating set of the logarithmic unit lattice with high probability. The proof of the theorem uses the fact that this output distribution is evenly distributed and concentrated and applies Proposition 18.2 on distributions over lattices.

Theorem 18.14. *There exists an absolute constant $C > 0$ and some $B_{\max} = \mathsf{poly}(L_{|\Delta_K|}(1/2), L_{n^n}(2/3), \rho_K^{\mathrm{cut}})$, where*

$$\rho_K^{\mathrm{cut}} = \min\left(\rho_K, \max(e^{\log^{\frac{2}{3}} |\Delta_K| \cdot \log^{\frac{2}{3}}(\log |\Delta_K|)}, e^{n^{\frac{2}{3}} \log^{\frac{4}{3}}(n)})\right),$$

such that the following holds. For every number field K, for every $k \in \mathbb{Z}_{>0}$, and for every set of primes $\mathbb{S}$ that generates the ray class group $\mathrm{Cl}_K^{\mathfrak{m}_0}$ of K, does not contain primes dividing $\mathfrak{m}_0$ and contains all primes (coprime with $\mathfrak{m}_0$) of norm $\leq B_{\max}$, the probability that

$$6 \cdot k + 6 \cdot (|\mathbb{S}| + \mathbb{r}) \cdot \Big[\log((|\mathbb{S}| + \mathbb{r}) \cdot \varsigma) + C \log \log |\Delta_K|\Big]$$

samples from Algorithm 4 generate the entire logarithmic $\mathbb{S}$-unit lattice $\mathrm{Log}_{\mathbb{S}}(\mathcal{O}_{K,\mathbb{S}}^\times)$ is at least $1 - e^{-k}$. Here, $|\mathbb{S}| + \mathbb{r} = \dim(\mathrm{Log}_{\mathbb{S}}(\mathcal{O}_{K,\mathbb{S}}^\times))$ (with $\mathbb{r} = \dim(\mathrm{Log}(\mathcal{O}_K^\times))$) and ς is from line 3 of Algorithm 4.

Proof. By Lemma 18.11 and Lemma 18.10 the output distribution of Algorithm 4 on $\mathrm{Log}_{\mathbb{S}}(\mathcal{O}_{K,\mathbb{S}}^\times)$ is 2/3-evenly distributed and $(R_0, 0)$-concentrated for $R_0 = O(\log^2 |\Delta_K|) + 3\varsigma \cdot n_0$, where ς is the deviation of the Gaussian distribution as in Algorithm 4, and where $n_0 = |\mathbb{S}| + n_{\mathbb{R}} + n_{\mathbb{C}} = \dim(\mathrm{Log}_{\mathbb{S}}(\mathcal{O}_{K,\mathbb{S}}^\times)) + 1$.

So, one can apply Proposition 18.2 with $p = \frac{2}{3}$ and $q = 0$ (using $(1-p-q)^{-1} = 3$) to deduce that the number of samples S required from the output distribution of

Algorithm 4 to generate the entire log-$\mathbb{S}$-unit lattice satisfies, for any $\alpha > 0$,

$$\Pr\left[S > 3 \cdot (2+\alpha)(t+n_1)\right] \leq e^{-\alpha(t+n_1)/2}, \tag{131}$$

where $n_1 = |\mathbb{S}| + \mathbb{r} = \dim(\mathrm{Log}_{\mathbb{S}}(\mathcal{O}_{K,\mathbb{S}}^{\times})) = n_0 - 1$ and[22]

$$\begin{aligned} t &= n_1 \cdot \log_2(R_0) - \log_2(\mathrm{Vol}(\mathrm{Log}_{\mathbb{S}}(\mathcal{O}_{K,\mathbb{S}}^{\times}))) \\ &\leq n_1 \cdot \log_2(R_0) + 2 \\ &\leq n_1[\log(n_0\varsigma) + O(\log\log|\Delta_K|)]. \end{aligned}$$

Taking $\alpha = 2k/(t+n_1)$, we have $\alpha(t+n_1)/2 = k$, and

$$3 \cdot (2+\alpha)(t+n_1) = 6(t+n_1) + 3\alpha(t+n_1) = 6(t+n_1) + 6k.$$

Hence, replacing the above formula into Equation (131), we obtain

$$\Pr\left[S > 6(t+n_1) + 6k\right] \leq e^{-k}.$$

Therefore, the probability that $6(t+n_1)+6k$ samples from the output distribution of Algorithm 4 generate the entire logarithmic $\mathbb{S}$-unit lattice $\mathrm{Log}_{\mathbb{S}}(\mathcal{O}_{K,\mathbb{S}}^{\times})$ is at least $1 - e^{-k}$. Replacing $t + n_1$ by the larger $n_1[\log(n_1\varsigma) + C(\log\log|\Delta_K|)]$ (for some absolute constant, and hiding n_1 under this constant by increasing the constant slightly if needed), yields the final claim. □

This theorem roughly states that the number of samples drawn from $\bar{\mathcal{D}}$ (the output distribution of Algorithm 4) for the sampled vectors to generate the log-$\mathbb{S}$-unit lattice $\mathrm{Log}_{\mathbb{S}}(\mathcal{O}_{K,\mathbb{S}}^{\times})$ (with high probability) only needs to exceed a small multiple of the dimension of $\mathrm{Log}_{\mathbb{S}}(\mathcal{O}_{K,\mathbb{S}}^{\times})$. In other words, if one assembles sufficiently many of such samples (quasi-linearly in the dimension), they generate the log-$\mathbb{S}$-unit lattice except for an exponentially small probability.

In almost all applications involving $\mathbb{S}$-units, a generating set with high probability is not sufficient. Instead, often a *basis* of the logarithmic $\mathbb{S}$-units is demanded (or, equivalently, a *fundamental set of* $\mathbb{S}$*-units*, see Section 19.1); and, additionally, no probabilistic error is allowed (i.e., one wants to be sure that the basis at hand truly is a basis of the full log-$\mathbb{S}$-unit lattice).

To resolve these issues, one needs to post-process the set of generators by lattice reduction techniques. This is the subject of Section 19.

19. Post-processing phase

19.1. Introduction.

Up to now, the algorithm of this paper allows to provably sample $\mathbb{S}$-units in $\mathcal{O}_{K,\mathbb{S}}^{\times}$ with a certain probability. Furthermore, a sufficient number of these $\mathbb{S}$-units samples will generate the entire $\mathcal{O}_{K,\mathbb{S}}^{\times}$ with overwhelming probability. To complete the entire algorithm, two tasks remain to be done.

(I) An algorithm to *verify* that the sampled elements truly generate all of $\mathcal{O}_{K,\mathbb{S}}^{\times}$ and not a subgroup thereof[23].

[22] We use here that $\mathrm{Vol}(\mathrm{Log}_{\mathbb{S}}(\mathcal{O}_{K,\mathbb{S}}^{\times})) = h_K \cdot R_K \cdot \sqrt{n_{\mathbb{R}} + n_{\mathbb{C}}} \geq R_K \geq 0.206$, since $R_K \geq 0.206$ uniformly for all number fields [32, Theorem B].

[23] One could say that this algorithm changes the probabilistic algorithm of this paper from a Monte Carlo algorithm (a randomized algorithm whose output may be incorrect with a small probability, but with bounded running time) to a Las Vegas algorithm (a randomized algorithm that always gives correct results, but whose running time is a random variable)

(II) An algorithm to compute a *fundamental system of* $\mathbb{S}$*-units* of $\mathcal{O}_{K,\mathbb{S}}^{\times}$ out of these generators, i.e., a sequence of elements $(\eta_j)_j \in \mathcal{O}_{K,\mathbb{S}}^{\times}$ with $j \in \{1,\ldots,\mathrm{rank}(\mathcal{O}_{K,\mathbb{S}}^{\times})\}$ such that every element $\eta \in \mathcal{O}_{K,\mathbb{S}}^{\times}$ is a power product of these elements (and a possible root of unity): $\eta = \tau_K \prod_{j=1}^{\mathrm{rank}(\mathcal{O}_{K,\mathbb{S}}^{\times})} \eta_j^{m_j}$ for some $m_j \in \mathbb{Z}$; where $\tau_K \in \mathcal{O}_K$ is some root of unity[24].

19.2. **Solving both (I) and (II) by computing bases.** Let $G \subseteq \mathcal{O}_{K,\mathbb{S}}^{\times}$ be a finite set of $\mathbb{S}$-units. Then its associated lattice

$$\mathcal{L}_G := \sum_{\eta \in G} \mathrm{Log}_{\mathbb{S}}(\eta) \cdot \mathbb{Z} = \{\mathrm{Log}_{\mathbb{S}}(\eta') \mid \eta' \in \langle G \rangle\}$$

satisfies $\mathcal{L}_G \subseteq \mathrm{Log}_{\mathbb{S}}(\mathcal{O}_{K,\mathbb{S}}^{\times})$, i.e., it is a *sublattice* of $\mathrm{Log}_{\mathbb{S}}(\mathcal{O}_{K,\mathbb{S}}^{\times})$.

An immediate observation is now that both the tasks (I) and (II) can be solved[25] if one is able to compute a *basis* of the lattice $\mathcal{L}_G$. Indeed, such a basis allows to retrieve the rank and the determinant of $\mathcal{L}_G$. If $\mathrm{rank}(\mathcal{L}_G) = \mathrm{rank}(\mathrm{Log}_{\mathbb{S}}(\mathcal{O}_{K,\mathbb{S}}^{\times})) = n_{\mathbb{C}} + n_{\mathbb{R}} + |\mathbb{S}| - 1$ and $\mathrm{Vol}(\mathcal{L}_G) = \mathrm{Vol}(\mathrm{Log}_{\mathbb{S}}(\mathcal{O}_{K,\mathbb{S}}^{\times})) = R_K \cdot h_K \cdot \sqrt{n_{\mathbb{R}} + n_{\mathbb{C}}}$ (see Equation (9)), we deduce that $\mathcal{L}_G = \mathrm{Log}_{\mathbb{S}}(\mathcal{O}_{K,\mathbb{S}}^{\times})$ (solving (I)) and that the basis of $\mathcal{L}_G$ is a basis for $\mathrm{Log}_{\mathbb{S}}(\mathcal{O}_{K,\mathbb{S}}^{\times})$ (solving (II)).

So, the naive approach would be to create the matrix $\mathbf{G}$ whose rows consist of $\mathrm{Log}_{\mathbb{S}}(\eta)$ with $\eta \in G$; the rows of this matrix then generate $\mathrm{Log}_{\mathbb{S}}(\mathcal{L}_G)$. Apply LLL-reduction to $\mathbf{G}$ to obtain a basis of $\mathrm{Log}_{\mathbb{S}}(\mathcal{L}_G)$ and compute its rank and determinant. Unfortunately, this approach is not directly applicable.

19.3. **The challenge of approximate matrices.** The challenge lies in the fact that the infinite places of $\mathrm{Log}_{\mathbb{S}}(\eta)$ for $\eta \in \mathcal{O}_{K,\mathbb{S}}^{\times}$ can only be *approximated*. Indeed, these infinite places consists of logarithms of algebraic numbers, whose can only be computed with a certain precision.

This has as a consequence that the row-oriented matrix $\mathbf{G}$ consisting of the (exact) elements $\mathrm{Log}_{\mathbb{S}}(\eta)$ for $\eta \in G$ is computationally out of reach. Instead one is forced to work with $\tilde{\mathbf{G}}$, an *approximation* of $\mathbf{G}$. This rational matrix $\tilde{\mathbf{G}}$ consists of elements $\mathrm{Log}_{\mathbb{S}}(\eta) + \epsilon_\eta$ for $\eta \in G$ and $\|\epsilon_\eta\|_1 < \epsilon$. Given the set G, we can compute $\tilde{\mathbf{G}}$ in time $|G| \cdot \mathsf{poly}(\log(\epsilon))$. Of course, we have

$$\|\mathbf{G} - \tilde{\mathbf{G}}\|_\infty \leq \epsilon.$$

where $\|\cdot\|_\infty$ is the induced ∞-norm on matrices[26].

19.4. **The Buchmann-Kessler-Pohst algorithm.** For sufficiently (but still feasibly) small $\epsilon \in (0,1)$, an algorithm by Buchmann, Pohst and Kessler [18, 19] allows to compute an integer matrix $\mathbf{M}$ such that $\tilde{\mathbf{B}} = \mathbf{M}\tilde{\mathbf{G}}$ is an approximate basis that satisfies

$$\|\tilde{\mathbf{B}} - \mathbf{B}\|_\infty \leq C \cdot \epsilon \quad \text{and} \quad \|\mathbf{M}\|_\infty \leq C$$

for some very large, but sufficiently bounded C (depending on $\|\mathbf{G}\|_\infty$ and invariants of the lattice $\mathrm{Log}_{\mathbb{S}}(\mathcal{O}_{K,\mathbb{S}}^{\times})$) and where $\mathbf{B} = \mathbf{M}\mathbf{G}$ is a well-conditioned basis for the lattice $\mathcal{L}_G$ generated by $\mathbf{G}$.

[24] As roots of unity vanish under the Logarithmic embedding into H and are easily computed, we omit them often in discussions.

[25] Note that we assume that the primes in $\mathbb{S}$ generate the ideal class group.

[26] We have $\|\mathbf{A}\|_\infty = \max_j \|\mathbf{a}_j\|_1$ where $\mathbf{a}_j$ are the rows of $\mathbf{A}$.

Though this algorithm is analyzed well [18], its analysis can only be applied whenever the rank of the lattice $\mathcal{L}_G$ is known. A slight variation of this analysis, in which the rank of $\mathcal{L}_G$ is not required to be known beforehand, is described in Section 23. Note that this analysis proves that the Buchmann-Kessler-Pohst algorithm actually *computes* the rank of $\mathcal{L}_G$.

19.5. **Computing the determinant.** So, the Buchmann-Kessler-Pohst algorithm allows to compute a $\tilde{\mathbf{B}} = \mathbf{B} + \epsilon' X$ close to a well-conditioned basis $\mathbf{B}$; say, $\|X\|_\infty \leq 1$ and $\epsilon' < 1$ is small.

Then, by a bound of Ipsen and Rehman [43, Corollary 2.14], for sufficiently (but feasibly) small $\epsilon' < 1$, one has

$$\det(\tilde{\mathbf{B}}^\top \tilde{\mathbf{B}}) \in [\tfrac{7}{8}, \tfrac{9}{8}] \cdot \det(\mathbf{B}^\top \mathbf{B}).$$

The determinant $\mathrm{Vol}(\mathrm{Log}_{\mathbb{S}}(\mathcal{O}_{K,\mathbb{S}}^\times))$ of the *full* log-$\mathbb{S}$-unit lattice can be (multiplicatively) approximated by means of Euler products (e.g., [5] or Proposition 2.9). In other words, we can efficiently compute a $D \in \mathbb{R}_{>0}$ such that

$$D \in [\tfrac{3}{4}, \tfrac{5}{4}] \cdot \mathrm{Vol}(\mathrm{Log}_{\mathbb{S}}(\mathcal{O}_{K,\mathbb{S}}^\times)).$$

The index $[\mathrm{Log}_{\mathbb{S}}(\mathcal{O}_{K,\mathbb{S}}^\times) : \mathcal{L}_G]$ can then be approximated by computing $\sqrt{\det(\tilde{\mathbf{B}}^\top \tilde{\mathbf{B}})/D}$, and satisfies[27]

$$\sqrt{\det(\tilde{\mathbf{B}}^\top \tilde{\mathbf{B}})/D} \in (0.74, 1.42) \cdot [\mathrm{Log}_{\mathbb{S}}(\mathcal{O}_{K,\mathbb{S}}^\times) : \mathcal{L}_G].$$

One deduces that if $[\mathrm{Log}_{\mathbb{S}}(\mathcal{O}_{K,\mathbb{S}}^\times) : \mathcal{L}_G] \geq 2$, then $\sqrt{\det(\tilde{\mathbf{B}}^\top \tilde{\mathbf{B}})/D} > 1.48$. Contrarily, if $[\mathrm{Log}_{\mathbb{S}}(\mathcal{O}_{K,\mathbb{S}}^\times) : \mathcal{L}_G] = 1$, then $\sqrt{\det(\tilde{\mathbf{B}}^\top \tilde{\mathbf{B}})/D} < 1.42$, which allows us to distinguish the two cases. Additionally, in the second case $\tilde{\mathbf{B}}$ is an approximate basis of $\mathrm{Log}_{\mathbb{S}}(\mathcal{O}_{K,\mathbb{S}}^\times)$.

19.6. **Assembling a fundamental set of $\mathbb{S}$-units.** Recall that $G = \{\eta_1, \ldots, \eta_k\} \subseteq \mathcal{O}_{K,\mathbb{S}}^\times$ is a set of $\mathbb{S}$-units and $\mathbf{G} = (\mathrm{Log}_{\mathbb{S}}(\eta_j))_{j \in \{1,\ldots,k\}}$ its associated matrix of logarithmic images. Assume that (by means of the algorithm in the text above) one has deduced that G indeed generates $\mathcal{O}_{K,\mathbb{S}}^\times$, i.e., $\langle G \rangle = \mathcal{O}_{K,\mathbb{S}}^\times$.

By the algorithm of Buchmann-Kessler-Pohst in Section 19.4, we can find an $\mathbf{M} = (m_{ij})_{ij}$ for which $\mathbf{B} = \mathbf{M}\mathbf{G}$ is a basis of $\mathrm{Log}_{\mathbb{S}}(\mathcal{O}_{K,\mathbb{S}}^\times)$. In other words, for $j = 1, \ldots, \mathrm{rank}(\mathcal{O}_{K,\mathbb{S}}^\times)$,

$$\mathbf{b}_j = \sum_{i=1}^k m_{ji} \,\mathrm{Log}_{\mathbb{S}}(\eta_i) = \mathrm{Log}_{\mathbb{S}}(\beta_j) \quad \text{ form a basis of } \mathrm{Log}_{\mathbb{S}}(\mathcal{O}_{K,\mathbb{S}}^\times).$$

As a consequence, by taking exponentials, for $j = 1, \ldots, \mathrm{rank}(\mathcal{O}_{K,\mathbb{S}}^\times)$,

$$\beta_j = \prod_{i=1}^k \eta_i^{m_{ji}} \quad \text{ form a system of fundamental units of } \mathcal{O}_{K,\mathbb{S}}^\times. \tag{132}$$

[27] We have $\sqrt{\det(\mathbf{B}^\top \mathbf{B})} \cdot c_1 = \sqrt{\det(\tilde{\mathbf{B}}^\top \tilde{\mathbf{B}})}$ for $c_1 \in [\sqrt{\frac{7}{8}}, \sqrt{\frac{9}{8}}]$ and $D = c_2 \cdot \mathrm{Vol}(\mathrm{Log}_{\mathbb{S}}(\mathcal{O}_{K,\mathbb{S}}^\times))$ with $c_2 \in [\frac{3}{4}, \frac{5}{4}]$. Therefore, $\sqrt{\det(\tilde{\mathbf{B}}^\top \tilde{\mathbf{B}})/D} = (c_1/c_2) \cdot \sqrt{\det(\mathbf{B}^\top \mathbf{B})/\mathrm{Vol}(\mathrm{Log}_{\mathbb{S}}(\mathcal{O}_{K,\mathbb{S}}^\times))}$, with $(c_1/c_2) \in [\sqrt{\frac{7}{8}} \cdot \frac{4}{5}, \sqrt{\frac{9}{8}} \cdot \frac{4}{3}] \subseteq (0.74, 1.42)$.

Note that the expression in Equation (132) is in exact arithmetic (i.e., no precision issues). Namely, the elements $\eta_i \in G$ are written in the basis of the number ring and the exponents m_{ji} are integers from the integer matrix $\mathbf{M}$.

19.7. **A fundamental set of $\mathbb{S}$-units by means of compact representation.** Note that actually representing β_j in terms of basis elements of $\mathcal{O}_K$ or K is generally computationally out of reach. Indeed, since the coefficients m_{ij} are rather large, the computation of β_j (i.e., expanding the product in Equation (132)) suffers from coefficient explosion.

This is the reason why the computation of a system of fundamental ($\mathbb{S}$)-units is generally given in 'compact representation' (e.g., [10], [26, §5.8.3]). That just means that, instead of giving the β_j in Equation (132) by expanding its product and writing β_j in the basis of the number ring, one represents β_j by the formal product consisting of m_{ji}-th powers of η_i.

In other words, a compact representation of $(\beta_j)_{j=1,\ldots,\operatorname{rank}(\mathcal{O}_{K,\mathbb{S}}^\times)}$ is just given by the pair $(\mathbf{M}, G)$ with $\mathbf{M} = (m_{ij})_{ij} \in \mathbb{Z}^{k\times(\operatorname{rank}(\mathcal{O}_{K,\mathbb{S}}^\times))}$ and $G = \{\eta_1, \ldots, \eta_k\}$, by which we represent, for $j \in \{1, \ldots, \operatorname{rank}(\mathcal{O}_{K,\mathbb{S}}^\times)\}$,

$$\beta_j = \prod_{i=1}^{k} \eta_i^{m_{ij}}.$$

The final output of the log-$\mathbb{S}$-unit algorithm of this paper is thus the pair $(\mathbf{M}, G)$.

19.8. **Final post-processing theorem.** A rigorous treatment of the post-processing phase can be found in Sections 23 to 25, as well as the proof of the following theorem. The techniques used in that treatment generally involve well-known lattice reductions and computations, though for their rigorousness there is a heavy emphasis on numerical stability. For now, to stay close to the current subject manner, those numerical stability computations are thus deferred to Part 3.

Theorem 19.1. *Let K be a number field with degree n, and let $G = \{\eta_1, \ldots, \eta_k\} \subseteq \mathcal{O}_{K,\mathbb{S}}^\times$ a finite set of $\mathbb{S}$-units. There exists an algorithm that takes as input the set G*[28] *and computes a system of fundamental $\mathbb{S}$-units for the subgroup $\langle G\rangle \subseteq \mathcal{O}_{K,\mathbb{S}}^\times$ generated by G in compact representation, i.e., a pair $(\mathbf{N}, G)$ with $\mathbf{N} \in \mathbb{Z}^{r\times k}$ such that $\upsilon_j = \prod_{i=1}^k \eta_i^{\mathbf{N}_{ji}}$ for $j \in \{1, \ldots, r\}$ form a system of fundamental $\mathbb{S}$-units for $\langle G\rangle$.*

Additionally, this algorithm decides whether $\langle G\rangle = \mathcal{O}_{K,\mathbb{S}}^\times$ or not, and runs in time polynomial in k, $\log|\Delta_K|$, $|\mathbb{S}|$, $\log\big(\max_j \|\operatorname{Log}_{\mathbb{S}}(\eta_j)\|\big)$ and $\max_j(\operatorname{size}(\eta_j))$.

20. Generating the exceptional $\mathbb{S}$-units

20.1. **Introduction.** In Theorem 18.14, the main result of Section 18, the set of primes $\mathbb{S}$ is assumed to contain no primes dividing the modulus $\mathfrak{m}_0$. This modulus $\mathfrak{m}_0$ is defined in lines 2 - 8 of Algorithm 3, and only depends on the Dedekind residue ρ_K, the degree $n = [K : \mathbb{Q}]$ and the absolute discriminant $|\Delta_K|$.

In our end result, we would like to be able to compute $\mathbb{S}'$-units for *any* set of primes $\mathbb{S}'$, without any restrictions. Indeed, we do want to allow $\mathbb{S}'$ to contain primes dividing $\mathfrak{m}_0$. Hence, in this section we aim, for any $\mathfrak{p} \mid \mathfrak{m}_0$, for computing

[28] The elements $\eta_i \in K$ are represented as vectors of rational coordinates in a fixed $\mathcal{O}_K$-basis, as explained in Section 2.6.

an $(\mathbb{S} \cup \{\mathfrak{p}\})$-unit $\eta_\mathfrak{p} \in \mathcal{O}^\times_{K,\mathbb{S}\cup\{\mathfrak{p}\}}$ that additionally satisfies $\operatorname{ord}_\mathfrak{p}(\eta_\mathfrak{p}) = 1$. Such an $(\mathbb{S} \cup \{\mathfrak{p}\})$-unit we will then call an *exceptional* $\mathbb{S}$*-unit*.

By adding these exceptional $\mathbb{S}$-units $\eta_\mathfrak{p}$ (for $\mathfrak{p} \mid \mathfrak{m}_0$) to a fundamental system of $\mathbb{S}$-units, one gets a fundamental system of $(\mathbb{S} \cup \{\mathfrak{p} \mid \mathfrak{p} \text{ divides } \mathfrak{m}_0\})$-units. Hence computing these $\eta_\mathfrak{p}$ allows for computing a $\mathbb{S}'$ unit group without any restrictions on the prime ideals. Note that, if $\mathfrak{m}_0 = (1)$ is trivial, no exceptional $\mathbb{S}$-units exist, and the algorithm of this section can be omitted. Hence, throughout this section we will assume that $\mathfrak{m}_0 \neq (1)$.

20.2. **Algorithm for generating exceptional $\mathbb{S}$-units.** In Algorithm 5, we describe how to compute an exceptional $\mathbb{S}$-unit in the case of $\mathfrak{m}_0 \neq (1)$. The following lemma shows that the running time of this algorithm is the same as the running time of sampling a single $\mathbb{S}$-unit as in Theorem 17.7.

Algorithm 5 Computing an exceptional $\mathbb{S}$-unit

Require:

(i) An LLL-reduced basis of $\mathcal{O}_K$,
(ii) a prime ideal $\mathfrak{q} \mid \mathfrak{m}_0$, (we assume $\mathfrak{m}_0 \neq (1)$)
(iii) a set of prime ideals $\mathbb{S}$ of K, that does not contain the primes dividing $\mathfrak{m}_0$.

Ensure: $\alpha \in \mathfrak{q}$ and $(v_\mathfrak{p})_{\mathfrak{p}\in\mathbb{S}} \in \mathbb{Z}^{|\mathbb{S}|}_{\geq 0}$ such that $\alpha\mathcal{O}_K = \mathfrak{q} \cdot \prod_{\mathfrak{p}\in\mathbb{S}} \mathfrak{p}^{v_\mathfrak{p}}$

```
define $\mathsf{b}, \omega$ as in Algorithm 3.
repeat
    $\alpha \leftarrow \texttt{Sample}(\mathfrak{q}, y = 1, \mathfrak{m}_0 \cdot \mathfrak{q}^{-1}, \mathsf{b}, \omega)$ (see Theorem 13.2)
until $\alpha\mathcal{O}_K \cdot \mathfrak{q}^{-1}$ is $\mathbb{S}$-smooth
compute $(v_\mathfrak{p})_{\mathfrak{p}\in\mathbb{S}} \in \mathbb{Z}^{\mathbb{S}}_{\geq 0}$ such that $\alpha\mathcal{O}_K \cdot \mathfrak{q}^{-1} = \prod_{\mathfrak{p}\in\mathbb{S}} \mathfrak{p}^{v_\mathfrak{p}}$
return $(\alpha, (v_\mathfrak{p})_{\mathfrak{p}\in\mathbb{S}})$.
```

Proposition 20.1. *There exists some $B_{\max} = \mathsf{poly}(L_{|\Delta_K|}(1/2), L_{n^n}(2/3), \rho_K^{\mathrm{cut}})$, where*

$$\rho_K^{\mathrm{cut}} = \min\left(\rho_K, \max(e^{\log^{\frac{2}{3}}|\Delta_K| \cdot \log^{\frac{2}{3}}(\log|\Delta_K|)}, e^{n^{\frac{2}{3}} \log^{\frac{4}{3}}(n)})\right),$$

such that the following holds. Assume that $\mathbb{S}$ contains all prime ideals coprime to $\mathfrak{m}_0$ of norm $\leq B_{\max}$ (where $\mathfrak{m}_0$ is defined as in lines 2 - 8 of Algorithm 3).

Then, on input $\mathfrak{q} \mid \mathfrak{m}_0$ and the set $\mathbb{S}$, Algorithm 5 outputs $(\alpha, (v_\mathfrak{p})_{\mathfrak{p}\in\mathbb{S}}) \in \mathfrak{q} \times \mathbb{Z}^{|\mathbb{S}|}_{\geq 0}$ such that

$$\alpha\mathcal{O}_K = \mathfrak{q} \prod_{\mathfrak{p}\in\mathbb{S}} \mathfrak{p}^{v_\mathfrak{p}}.$$

Furthermore, Algorithm 3 runs in expected time

$$\mathsf{poly}(L_{|\Delta_K|}(\tfrac{1}{2}), L_{n^n}(\tfrac{2}{3}), \mathrm{size}(\mathbb{S}), \rho_K^{\mathrm{cut}}, \log|\Delta_K|).$$

Proof. The proof can be copied from that of Theorem 17.7. The sole differences are the instantiation $\mathfrak{m}_0 \cdot \mathfrak{q}^{-1}$ instead of $\mathfrak{m}_0$, and the instantiations $\mathfrak{a} = \mathfrak{q}$ and $y = 1$. The latter instantiations do not impact the proof, and since $\mathrm{size}(\mathfrak{q}) \leq \mathsf{poly}(\log|\Delta_K|) \cdot x \leq \mathsf{poly}(\log|\Delta_K|)$ by the definition of x in line 3 of Algorithm 3, the size of $\mathfrak{q}$ can be omitted in the running time.

Note that in Theorem 17.7, the input ideal $\mathfrak{a}$ is required to be coprime to $\mathfrak{m}_0$. In the context of the current theorem, $\mathfrak{q}$ is coprime to $\mathfrak{m}_0 \cdot \mathfrak{q}^{-1}$ by construction, and hence the same reasoning as in Theorem 17.7 applies.

The instantiation $\mathfrak{m}_0 \cdot \mathfrak{q}^{-1}$ instead of $\mathfrak{m}_0$ does impact the value of $N(\mathfrak{m}_0)/\phi(\mathfrak{m}_0)$, but only by a factor $\frac{N(\mathfrak{q})}{N(\mathfrak{q})-1} \in [1,2]$, which only impacts the success probability by a factor in $[1,2]$ and hence does not impact the final running time. Therefore, the running time of Proposition 20.1 equals that of Theorem 17.7, with $\text{size}(\mathfrak{a})$ and $\text{size}(y)$ deleted. □

21. Full algorithm, final theorem and discussion

Algorithm 6 Computing a fundamental system of $\mathbb{S}$-units

Require: A number field K , an LLL-reduced basis of $\mathcal{O}_K$, and a set $\mathbb{S}$ of prime ideals of K.

Ensure: A pair $(\mathbf{M}, G)$ with $\mathbf{M} = (m_{ij})_{ij} \in \mathbb{Z}^{k \times \text{rank}(\mathcal{O}_{K,\mathbb{S}}^\times)}$ and $G = \{\eta_1, \dots \eta_k\} \subseteq \mathcal{O}_{K,\mathbb{S}}^\times$ such that $\{\beta_1, \dots, \beta_{\text{rank}(\mathcal{O}_{K,\mathbb{S}}^\times)}\}$ is a fundamental set of $\mathbb{S}$-units, where

$$\beta_j = \prod_{i=1}^{k} \eta_i^{m_{ji}}.$$

```
Put $G = \emptyset$
Define $\mathfrak{m}_0$ as in lines 2 - 8 of Algorithm 3.
Define $\mathbb{T} = \mathbb{S} \backslash \{\mathfrak{p} \mid \mathfrak{p} \mid \mathfrak{m}_0\}$.
repeat
    Apply Algorithm 4 with input $\mathbb{T}$, yielding $(\alpha, (a_\mathfrak{p})_{\mathfrak{p} \in \mathbb{T}})$.
    Add the output $\alpha$ (and its factorization) to $G'$.
until the Buchmann-Kessler-Pohst algorithm as in Section 19 finds that $G'$ generates $\mathcal{O}_{K,\mathbb{T}}^\times$.
Let $(\mathbf{M}', G')$ be the output of the Buchmann-Kessler-Pohst algorithm, as in Theorem 19.1. Write $G' = \{\eta_1, \dots, \eta_{k'}\}$ and $\mathbf{M}' = (m'_{ij})_{ij} \in \mathbb{Z}^{k' \times \text{rank}(\mathcal{O}_{K,\mathbb{S}}^\times)}$.
Compute for any $\mathfrak{q} \mid \mathfrak{m}_0$, an exceptional $\mathbb{T}$-unit $\eta_\mathfrak{q} \in \mathcal{O}_{K,\mathbb{T}\cup\{\mathfrak{q}\}}^\times$, using Algorithm 5.
Order $\{\eta_\mathfrak{q} \mid \mathfrak{q} \mid \mathfrak{m}_0\} = \{\eta_{k'+1}, \dots, \eta_k\}$ and extend the set $G'$ to $G = \{\eta_1, \dots, \eta_{k'}, \eta_{k'+1}, \dots, \eta_k\}$. Put $m_{ji} = m'_{ji}$ for $i \in \{1, \dots, k'\}$ and $j \in \{1, \dots, \text{rank}(\mathcal{O}_{K,\mathbb{T}}^\times)\}$, put $m_{ji} = \delta_{ji}$ (Kronecker delta) for $i \in \{k'+1, \dots, k\}$. Set $\mathbf{M} = (m_{ij})_{ij} \in \mathbb{Z}^{k \times \text{rank}(\mathcal{O}_{K,\mathbb{S}}^\times)}$
return $(\mathbf{M}, G)$.
```

Proposition 21.1 (ERH)**.** *There is a randomized algorithm $\mathcal{A}$ (Algorithm 6) for which the following holds. Let K be a number field and let*

$$\rho_K^{\text{cut}} = \min(\rho_K, \max(e^{\log^{\frac{2}{3}} |\Delta_K| \cdot \log^{\frac{2}{3}} (\log |\Delta_K|)}, e^{n^{\frac{2}{3}} \cdot \log^{\frac{4}{3}} (n)})).$$

Note that $\rho_K^{\text{cut}} \leq L_{|\Delta_K|}(2/3 + o(1))$.
Then there exists a bound

$$B = \mathit{poly}(L_{|\Delta_K|}(1/2), L_{n^n}(2/3), \rho_K^{\text{cut}}).$$

such that if the algorithm $\mathcal{A}$ is given a set of primes $\mathbb{S}$ of K containing all primes with norm bounded by B and generating the ray class group $\mathrm{Cl}_K^{\mathfrak{m}_0}$ of K, the algorithm $\mathcal{A}$ computes a fundamental system of $\mathbb{S}$-units of $\mathcal{O}_{K,\mathbb{S}}^\times$ in compact representation, in expected time

$$\mathsf{poly}(L_{|\Delta_K|}(1/2), L_{n^n}(2/3), \rho_K^{\mathrm{cut}}, \mathrm{size}(\mathbb{S})).$$

Proof. The correctness of the algorithm follows from the following two arguments. By the 'until' criterion in step 7, that requires G' to generate the $\mathbb{T}$-unit group, where $\mathbb{T} = \mathbb{S}\backslash\{\mathfrak{p} \mid \mathfrak{p} \mid \mathfrak{m}_0\}$. Since the exceptional $\mathbb{T}$-units $\eta_\mathfrak{q}$ satisfy $\mathrm{ord}_\mathfrak{q}(\eta_\mathfrak{q}) = 1$ and $\eta_\mathfrak{q} \in \mathcal{O}_{K,\mathbb{S}}^\times$, adding these indeed result in a fundamental set of $\mathbb{S}$-units.

The running time of the algorithm is dominated by the repeat loop and the computation of the exceptional units. We will use that $\mathrm{size}(\mathbb{T}) \leq \mathrm{size}(\mathbb{S})$.

In the repeat loop, Algorithm 4 and the 'post-processing' Buchmann-Kessler-Pohst algorithm take expected time $\mathsf{poly}(L_{|\Delta_K|}(1/2), L_{n^n}(2/3), \rho_K^{\mathrm{cut}}, \mathrm{size}(\mathbb{S}))$ (see Lemma 18.9, Theorem 17.7, Theorem 19.1, and the bound

$$\varsigma \leq 3\max(\sqrt{\log(n_\mathbb{R} + n_\mathbb{C} + |\mathbb{S}|)}, \tilde{\varrho}) \leq \mathsf{poly}(\log|\Delta_K|, \mathrm{size}(\mathbb{S}))$$

from Proposition 16.10). The number of repetitions is expected to be within $\widetilde{O}(|\mathbb{S}|\log\log|\Delta_K|) = \mathsf{poly}(\mathrm{size}(\mathbb{S}))$ (see Theorem 18.14).

The run time of computing all exceptional units is, by Proposition 20.1, at most $\mathsf{poly}(L_{|\Delta_K|}(\frac{1}{2}), L_{n^n}(\frac{2}{3}), \mathrm{size}(\mathbb{S}), \rho_K^{\mathrm{cut}}, \log|\Delta_K|)$, where we use that the number of prime ideals dividing $\mathfrak{m}_0$ can be at most $O(\log(\mathcal{N}(\mathfrak{m}_0))) = \mathsf{poly}(\log|\Delta_K|)$ (see Equation (113)).

Therefore, the expected running time of the overall computation is $\mathsf{poly}(L_{|\Delta_K|}(1/2), L_{n^n}(2/3), \rho_K^{\mathrm{cut}}, \mathrm{size}(\mathbb{S}))$ as well. □

The following theorem is the same as Proposition 21.1 except that the set of primes $\mathbb{S}$ can now be arbitrary.

Theorem 21.2 (ERH)**.** *There is a probabilistic algorithm which, on input a number field K and a set of primes $\mathbb{S}$ of K, computes a fundamental system of $\mathbb{S}$-units of $\mathcal{O}_{K,\mathbb{S}}^\times$ in compact representation, in expected time*

$$\mathsf{poly}(L_{|\Delta_K|}(1/2), L_{n^n}(2/3), \rho_K^{\mathrm{cut}}, \mathrm{size}(\mathbb{S})).$$

where $\rho_K^{\mathrm{cut}} = \min(\rho_K, \max(e^{\log^{\frac{2}{3}}|\Delta_K| \cdot \log^{\frac{2}{3}}(\log|\Delta_K|)}, e^{n^{\frac{2}{3}} \cdot \log^{\frac{4}{3}}(n)}))$. *Note that* $\rho_K^{\mathrm{cut}} \leq L_{|\Delta_K|}(2/3 + o(1))$.

Proof. Extend $\mathbb{S}$ to a set $\mathbb{S}' \supset \mathbb{S}$ of primes, containing all primes with norm below $B = \mathsf{poly}(\max(L_{|\Delta_K|}(1/2, 1), L_{n^n}(2/3, 1), \rho_K^{\mathrm{cut}}))$ and generating the ray class group $\mathrm{Cl}_K^{\mathfrak{m}_0}$ of K, as in Proposition 21.1. Note that, since $\log(\mathcal{N}(\mathfrak{m}_0)) = O(\log^2|\Delta_K|)$ (see Equation (113)) and by Bach's bound $O(\log(|\Delta_K|^2 N(\mathfrak{m}_0)))$ for ray class groups [4] this is asymptotically clearly satisfied.

Then, apply Algorithm 6 to get the pair $(\mathbf{M}, G)$ representing a fundamental system of $\mathbb{S}'$-units. This costs expected time $\mathsf{poly}(L_{|\Delta_K|}(1/2), L_{n^n}(2/3), \rho_K^{\mathrm{cut}}, \mathrm{size}(\mathbb{S}))$, since $\mathrm{size}(\mathbb{S}') = \mathsf{poly}(B, \mathrm{size}(\mathbb{S}))$.

As a preparation for the Hermite normal form algorithm, we put the infinite primes and the primes of $\mathbb{S}$ to the very left in the representation of $\mathrm{Log}_{\mathbb{S}'}(\gamma)$, i.e.,

$$\mathrm{Log}_{\mathbb{S}'}(\gamma) := \Big(\underbrace{\mathrm{Log}(\gamma)}_{\text{infinite places}}, \underbrace{(-\mathrm{ord}_\mathfrak{p}(\gamma))_{\mathfrak{p}\in\mathbb{S}}}_{\text{places in } \mathbb{S}}, (-\mathrm{ord}_\mathfrak{p}(\gamma))_{\mathfrak{p}\in\mathbb{S}'\backslash\mathbb{S}} \Big)$$

Put $\tilde{\mathbf{G}}$ for the row-matrix consisting of the approximations of $\operatorname{Log}_{\mathbb{S}'}(\gamma_j)$ for $j = 1, \ldots, |G|$, where $G = \{\gamma_1, \ldots, \gamma_{|G|}\}$. We then can write $\tilde{\mathbf{B}} = \mathbf{M}\tilde{\mathbf{G}}$ for an approximate basis for the log-$\mathbb{S}'$-unit lattice (with the infinite places and the places of $\mathbb{S}$ at the very left).

We now apply a Hermite normal form algorithm to $\tilde{\mathbf{B}}$, leading to $\tilde{\mathbf{H}} = \mathbf{U}\tilde{\mathbf{B}}$ that is in a lower-triangular shape. Hence, the first $|\mathbb{S}| + n_{\mathbb{R}} + n_{\mathbb{C}} - 1$ rows must then represent the Log-$\mathbb{S}$-units, as it is the dimension of $\mathbb{S}$ plus the dimension of the logarithmic unit lattice $\operatorname{Log}(\mathcal{O}_K^\times)$. Hence, the $\mathbb{S}$-units can be given in compact representation $(\mathbf{N}, G)$ with $\mathbf{N}$ being the first $|\mathbb{S}| + n_{\mathbb{R}} + n_{\mathbb{C}} - 1$ rows of $\mathbf{UM}$. I.e., a fundamental set of $\mathbb{S}$-units is

$$\eta_k = \prod_{j=1}^{|G|} \gamma_j^{\mathbf{N}_{kj}} \qquad \text{for } k \in \{1, \ldots, |\mathbb{S}| + n_{\mathbb{R}} + n_{\mathbb{C}} - 1\}.$$

Since the running time of the Hermite normal form is polynomial in the dimensions of the input matrix and the maximum $\operatorname{size}(c_{ij})$ of the coefficients[29] [72], we simply deduce that computing this Hermite normal form also takes time $\mathsf{poly}(L_{|\Delta_K|}(1/2), L_{n^n}(2/3), \rho_K^{\text{cut}}, \operatorname{size}(\mathbb{S}))$. This concludes the proof. □

21.1. **On the complexity of the $\mathbb{S}$-unit computation algorithm.** Now that we have proved the main result of this part of the article, we turn to its meta-analysis, comparing it with former heuristic claims. Both the provable and the heuristic running time of the $\mathbb{S}$-unit computation algorithm depend heavily and quite intricately on both the parameters $\log(|\Delta_K|)$ and the degree n of the number field K at hand (see Figure 2). We will explain here the origin of the various changes in the running times, starting with that of the heuristic algorithm in Section 21.3 and ending with the running time of the provable algorithm of this paper in Section 21.4.

In this explanation of the running times, we will only discuss the the running time in terms of their 'dominant part', which are of the shape $\exp(\widetilde{O}(\log^\delta |\Delta_K|))$ or $\exp(\widetilde{O}(n^\delta))$ for some $\delta \in [0,1]$. Also, in Figure 2 only this 'dominant part' is shown. It is useful to think of, for example, $\exp(\widetilde{O}(\log^\delta |\Delta_K|))$ of being of 'approximately the same magnitude' as $L_{|\Delta_K|}(\delta, 1)$, while keeping in mind that $L_{|\Delta_K|}(\delta, 1)$ is a much more precise way of estimating complexities. For the the more intricate analysis, we refer to the proof of Theorem 17.7.

21.2. **A general formula for the complexity of the $\mathbb{S}$-unit computation algorithm.** At the very core, most $\mathbb{S}$-unit computation algorithms consist of the following two steps (in which $\mathbb{S} = \{\mathfrak{p} \text{ prime ideal} \mid \mathcal{N}(\mathfrak{p}) \leq B\}$):

(I) Assemble R 'relations' in the shape of B-smooth elements of norm $\leq M$. Those are just elements of $\mathcal{O}_{K,\mathbb{S}}^\times$.

(II) Post-process these R 'relations' to get a system of fundamental $\mathbb{S}$-units using lattice reduction techniques. Such a system of fundamental $\mathbb{S}$-units can be seen as a (multiplicative) 'basis' of $\mathcal{O}_{K,\mathbb{S}}^\times$.

[29] We might have to scale up the coefficients of the infinite places to integers (column-wise), but that does not impact significantly the size of the coefficients.

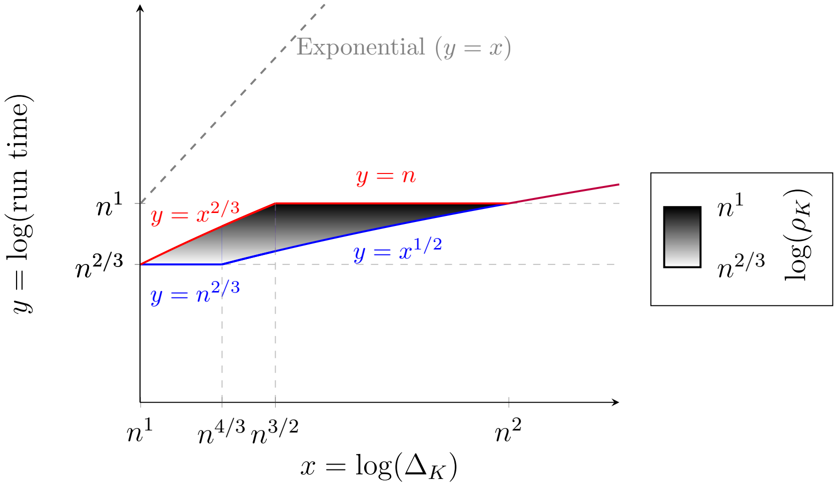

FIGURE 2. A depiction of the (dominant part of the) provable running time compared to the (dominant part of the) heuristic running time for number fields K, depending on $\log(\rho_K)$ and the value of $\log(|\Delta_K|)$ compared to the degree n.
The blue line corresponds to the heuristic running time claimed in [10].
The provable running time of the $\mathbb{S}$-unit algorithm of the present work varies between the blue and red lines, depending on $\log(\rho_K)$. The running time can be found in this graph by first searching the point $\log|\Delta_K|$ (in terms of powers of the degree n) on the x-axis, and then read off, depending on the size of $\log(\rho_K)$, where between the blue and red line the (logarithm of the) run time must be. For number fields with $\log(\rho_K) \leq n^{2/3}$, the running time is dictated by the blue line, and when $\log(\rho_K) \geq n$, the complexity is dictated by the red line.

21.2.1. *Part (I).* The B-smooth elements are sampled in a probabilistic way, by 'picking an element of norm $\leq M$ at random and hoping it will be smooth'. One such attempt costs 1 sampling of a norm $\leq M$ element (denoted S_M) and 1 check for being B-smooth (denoted C_B). Let us denote $p_{M,B}$ for the probability of success, namely that such sampling indeed yields a B-smooth element. Then, the running time for sampling a single B-smooth element (a 'relation') with constant success probability takes time $(S_M + C_B) \cdot p_{M,B}^{-1}$. To obtain R such relations, one obtains for part (I) an expected complexity of

$$\mathsf{poly}\left(R(S_M + C_B) \cdot p_{M,B}^{-1}\right).$$

21.2.2. *Part (II).* The post-processing step consists of lattice reduction on a sort-of 'valuation matrix' of the gathered relations (elements in $\mathcal{O}_{K,\mathbb{S}}^{\times}$). Namely, each of these R elements $\eta \in \mathcal{O}_{K,\mathbb{S}}^\times$ are decomposed as

$$\big[\operatorname{Log}(\eta), (v_{\mathfrak{p}}(\eta))_{\mathfrak{p}\in\mathbb{S}}\big] \in \mathbb{R}^{n_{\mathbb{R}}+n_{\mathbb{C}}-1} \times \mathbb{Z}^{\mathbb{S}},$$

where each vector entry is bounded by[30] $\mathsf{poly}(\log(M))$, where M is the maximum norm of the elements. Reducing an $R \times (\mathbb{S} + n_{\mathbb{R}} + n_{\mathbb{C}} - 1)$-matrix like this costs, roughly said, time at most $\mathsf{poly}(R \cdot B \cdot \log(M))$ (where we use $B \approx |\mathbb{S}|$). Here we assume that the decomposition of the elements into prime ideals has already be done in the 'check for being smooth' step in Part (I).

21.2.3. *The general formula for the complexity.* Adding the complexities, using that $R \geq B$ and assuming that $\log(M)$ is negligible compared to B, we obtain a general formula for the complexity, by effectively ignoring the costs of Part (II):

$$\mathsf{poly}\left(R \cdot (S_M + C_B) \cdot p_{M,B}^{-1}\right), \tag{133}$$

where S_M is the cost of sampling a random element in $\mathcal{O}_K$ of norm $\leq M$, where C_B is the costs of checking whether an element is B-smooth, where $p_{M,B}$ is the probability that a randomly sampled norm $\leq M$ element in $\mathcal{O}_K$ is B-smooth, and where R is the (average) number of relations (elements in $\mathcal{O}_{K,\mathbb{S}}^\times$) needed to generate $\mathcal{O}_{K,\mathbb{S}}^\times$.

21.3. **Heuristic running time of Biasse & Fieker** [10]**.** For the heuristic running time [10], there are essentially two 'regimes' of number fields. Namely, 'regime A', the fields for which $n \leq \log|\Delta_K| \leq n^{4/3}$ and 'regime B' the fields for which $n^{4/3} \leq \log|\Delta_K| < \infty$ (see the blue 'Heuristic' line in Figure 2). We will explain, with the general formula of Equation (133) and the heuristic assumptions of [10], why these regimes pop up.

Heuristic 1 (and 2) of [10] roughly state that they assume that the probability $p_{M,B}$ of a random element of norm $\leq M$ being B-smooth equals

$$p_{M,B} = \exp\Big(-\widetilde{O}\Big(\frac{\log(M)}{\log(B)}\Big)\Big) \quad \text{(heuristic)}$$

Additionally, Heuristic 3 of [10], roughly states that the number of elements ('relations') R required to generate the full $\mathbb{S}$-unit group $\mathcal{O}_{K,\mathbb{S}}^\times$, is only slightly larger than B, i.e.,

$$R = \widetilde{O}(B) \quad \text{(heuristic)}$$

Using trivial trial division for B-smooth checking yields $C_B = O(B)$. One of the main improvements of [10] is the use of stronger lattice reduction techniques (BKZ-b) to sample small norm elements. More precisely, in their paper, they roughly take $M = e^{n^2/\mathsf{b}} \cdot |\Delta_K|$ and $S_M = \widetilde{O}(\mathsf{b}^{\mathsf{b}})$; where b is the block size parameter of BKZ.

Filling in these parameters into the formula of Equation (133) and simplifying adequately, we obtain

$$\mathsf{poly}\left(\mathsf{b}^{\mathsf{b}} \cdot B \cdot p_{M,B}\right) = \exp\Big(\widetilde{O}\Big(\mathsf{b} + \log(B) + \frac{n^2/\mathsf{b} + \log|\Delta_K|}{\log(B)}\Big)\Big)$$

[30] This is not the case per se, a fixed norm element can have arbitrarily large entries of the logarithmic embedding. Though, here, in this quick explanation, we assume that this is the case.

Optimizing the parameters B and b, we quickly arrive at a complexity[31] of $\exp(\widetilde{O}(\log^{1/2}|\Delta_K|))$ whenever $n^{4/3} \leq \log|\Delta_K|$ and $\exp(\widetilde{O}(n^{2/3}))$ whenever $n^{4/3} \geq \log|\Delta_K|$. This explains the differences in complexity in the two regimes.

Remark 21.3. *Intuitively, these two regimes can be explained by the hardness of sampling small-norm elements in $\mathcal{O}_K$. Recall that for a sample of an element of maximum norm $M = e^{n^2/\mathsf{b}}|\Delta_K|$ one has to pay b^{b}.*

For $|\Delta_K|$ large compared to the degree (regime B), the extra factor $e^{n^2/\mathsf{b}}$ in M does not impact the asymptotic size of M. Contrarily, if the discriminant is small (regime A), the factor $e^{n^2/\mathsf{b}}$ in M becomes dominant. Therefore, the block size b needs to be increased, resulting in a larger complexity.

To summarize, for large discriminants, it is the probability of sampling smooth elements that is the bottleneck of the algorithm, whereas for small discriminants it is the run time of BKZ.

21.4. **Provable running time of this paper.** We revisit the running time of the $\mathbb{S}$-unit group computing algorithm, but now without heuristics and only assuming the Generalized Riemann hypothesis. The provable running time depends again on the parameters n and $\log|\Delta_K|$ and fall into three regimes. Regime A are the number fields for which $n \leq \log|\Delta_K| \leq n^{3/2}$, regime B the number fields satisfying $n^{3/2} \leq \log|\Delta_K| \leq n^2$ and regime C the number fields for which $n^2 \leq \log|\Delta_K| < \infty$ (see the red 'Provable' line in Figure 2). In the following we show how these different regimes arise.

In the algorithm of this paper the sampling probability for smooth elements has a provable lower bound from Part 1 (see Theorem 9.5)

$$p_{M,B} \geq \frac{1}{\rho_K} \cdot e^{-\widetilde{O}\left(\frac{\log M}{\log B}\right)} \quad \text{(provable)}$$

where $\rho_K = \lim_{s\to 1}(s-1)\zeta_K(s)$ is the residue of the Dedekind zeta function at $s = 1$. Additionally, in Part 2, we prove that taking

$$R = \widetilde{O}(B) \quad \text{(provable)}$$

is indeed sufficient to generate the full $\mathbb{S}$-unit group. Instantiating the rest of the parameters like in the heuristic version (Section 21.3), we obtain the same running time, with an extra factor ρ_K.

$$\exp\left(\widetilde{O}\Big(\mathsf{b} + \log(B) + \log(\rho_K) + \frac{n^2/\mathsf{b} + \log|\Delta_K|}{\log(B)}\Big)\right)$$

Regime C. As we have $\log(\rho_K) \leq n\log(\log(|\Delta_K|)/n)$ [53], we can deduce, with the same reasoning as in Section 21.3 that the run time is bounded by $\exp(\log^{1/2}|\Delta_K|)$ whenever $\log|\Delta_K| \geq n^2$. Then, namely, $\log\rho_K \leq \widetilde{O}(\log^{1/2}|\Delta_K|)$ and therefore ρ_K has essentially no significant influence on the running time. This explains the complexity of $\exp(\widetilde{O}(\log^{1/2}|\Delta_K|))$ for regime C in Figure 2.

[31] A lower bound of this complexity is $\exp(\widetilde{O}(\log^{1/2}|\Delta_K|))$, which is achieved when $\log B = \mathsf{b} = \log^{1/2}|\Delta_K|$ and $n^2/\mathsf{b} \leq \log|\Delta_K|$, i.e., $n^2 \leq \log^{3/2}|\Delta_K|$. This explains the running time for 'regime B', in which $n^{4/3} \leq \log|\Delta_K|$.

In 'regime A', we have $n^{4/3} \geq \log|\Delta_K|$, for which a lower bound for the complexity is $\exp(\widetilde{O}(n^{2/3}))$ which is achieved by taking $\log(B) = \mathsf{b} = n^{2/3}$.

Regime B. When $\log|\Delta_K| \leq n^2$, the running time of the provable algorithm possibly depends on ρ_K. As ρ_K can be exponentially large in n, the worst-case running time for this regime is $\exp(O(n))$. Note, however, that for this regime, depending on the magnitude of ρ_K, the running time can lie anywhere in the gray area of Figure 2.

Regime A. Whenever the discriminant is sufficiently small, namely, $\log|\Delta_K| \leq n^{3/2}$, one can apply a trick to significantly diminish the influence of the residue ρ_K on the running time.

Instead of sampling random elements in $\mathcal{O}_K$, one samples random elements *coprime with some modulus ideal* $\mathfrak{m}_0 \subseteq \mathcal{O}_K$ that is the product of all primes below some bound X. This sampling is done in a specific way, namely, by first sampling $\tau \in (\mathcal{O}_K/\mathfrak{m}_0)^\times$ and then sampling a random element equivalent to τ modulo $\mathfrak{m}_0$. This causes the sampling probability of smooth elements (coprime with $\mathfrak{m}_0$) to *increase* by a factor $\mathcal{N}(\mathfrak{m}_0)/\phi(\mathfrak{m}_0)$, but it also causes the maximum norm M of the sampled elements to increase[32] by a factor $\mathcal{N}(\mathfrak{m}_0)$, i.e.,

$$p_{M,B} \geq \frac{\mathcal{N}(\mathfrak{m}_0)}{\phi(\mathfrak{m}_0)\cdot\rho_K} \cdot e^{-\widetilde{O}\left(\frac{\log M + \log \mathcal{N}(\mathfrak{m}_0)}{\log B}\right)} \quad \text{(provable)} \tag{134}$$

with $M = 2^{n^2/\mathsf{b}}|\Delta_K|$. Using the fact that $\frac{\mathcal{N}(\mathfrak{m}_0)}{\phi(\mathfrak{m}_0)\cdot\rho_K} \geq \exp(-\widetilde{O}(\log|\Delta_K|/\sqrt{X}))$ (see Proposition 11.2), and $\mathcal{N}(\mathfrak{m}_0) \approx \exp(\widetilde{O}(X + \log|\Delta_K| \cdot \sqrt{X}))$ (see Lemma 11.4), we can rewrite Equation (134):

$$p_{M,B} \geq e^{-\widetilde{O}\left(\frac{\log|\Delta_K|}{\sqrt{X}} + \frac{n^2/\mathsf{b} + \log|\Delta_K| + X + \log|\Delta_K|\cdot\sqrt{X}}{\log B}\right)} \quad \text{(provable).} \tag{135}$$

Plugging in this probability lower bound into the general formula, the logarithm of the running time becomes, asymptotically,

$$\widetilde{O}\left(\mathsf{b} + \log(B) + \frac{n^2/\mathsf{b} + \log|\Delta_K|\cdot\sqrt{X}}{\log(B)} + \frac{\log|\Delta_K|}{\sqrt{X}}\right). \tag{136}$$

This has as asymptotic lower bound $\widetilde{O}(\log^{2/3}|\Delta_K|)$, attained when $X = \log^{2/3}|\Delta_K|$ and $\log(B) = \log^{2/3}|\Delta_K|$. Note that this trick is only useful whenever $\log^{2/3}|\Delta_K| \leq \log\rho_K \leq n\log(\log|\Delta_K|/n)$, otherwise the approach as in regime B (whose running time depends on ρ_K) would be faster. So, surely whenever $\log^{2/3}|\Delta_K| \leq n$, i.e., $\log|\Delta_K| \leq n^{3/2}$, this trick could outperform regime B (if ρ_K is large), which explains regime A of the provable running time (see Figure 2).

21.5. Discussion.

Is the dependency of the running time on ρ_K innate? The (dominant parts of the) provable expected running time of our $\mathbb{S}$-unit computation algorithm is depicted in Figure 2 depending on the parameters $\log(\rho_K)$ and $\log|\Delta_K|$ compared to the degree n. For a fixed number field with $\log|\Delta_K| = n^\delta$ for $\delta \in [1,2]$ the provable expected running time can be anywhere where the line $x = n^\delta$ intersects the shaded area, depending on the size of ρ_K. For small ρ_K, this running time can be equal to the heuristic one (the bottom blue line), whereas for large ρ_K, it might be equal to

[32] Because, instead of sampling in $\mathcal{O}_K$, we now have to sample in the shifted lattice $\tau + \mathfrak{m}_0$, which has covolume $\mathcal{N}(\mathfrak{m}_0)$ times the covolume of $\mathcal{O}_K$.

the worst-case provable one (the top red line). As this quantity ρ_K is computable by Euler products.

The main question in this discussion is: *does ρ_K really have this influence on the running time* or is it merely an artifact of the lack of tight bounds on smooth ideal densities?

Namely, there are asymptotic formulae [75] (see also Equation (103)) indicating that, for $u = \log(x)/\log(B)$,

$$\delta_{\mathcal{S}_B}[x] \geq u^{-u}$$

but, generally, nothing is said about what the smallest $x \in \mathbb{R}_{>0}$ is for which this holds. For our purposes, we would like this smallest x to be at most $O(|\Delta_K|)$ (and $B = \exp(O(\log^{1/2}(|\Delta_K|))))$. This is not something we could deduce from asymptotic results. Therefore, we chose to use a combinatorial result (Lemma 14.1), which shows that, for every $x > \log(B) \geq 2\log\log|\Delta_K|$,

$$\delta_{\mathcal{S}_B}[x] \geq \frac{1}{\rho_K \cdot B \cdot (4\log(B))^u} u^{-u}.$$

This lower bound has as an advantage that it is explicit and holds for small values of x (in particular the range around $|\Delta_K|$ we need). The disadvantage is the extra factor $\frac{1}{\rho_K}$, which increases, for large ρ_K, the provable running time of our algorithm drastically.

We could ask the main question of this discussion differently:

Are there bounds $B = \exp(O(\log^{1/2}(|\Delta_K|)))$, $X = O(|\Delta_K|)$, and $C^{-1} = \mathsf{poly}(B, \exp(\log^{1/2}|\Delta_K|))$ such that for all $x \geq X$, and for $u = \log(x)/\log(B)$, the inequality

$$\delta_{\mathcal{S}_B}[x] = \frac{|\{\mathfrak{a} \in \mathcal{S}_B \mid \mathcal{N}(\mathfrak{a}) \leq x\}|}{\rho_K \cdot x} \geq C \cdot u^{-u}$$

holds?

If the answer is *yes*, our algorithm would reach the heuristic running time proposed by [10]. If the answer is *no*, it is likely that the heuristic algorithm of [10] does not achieve its claimed running time, and instead has the same dependence on ρ_K as our algorithm.

The use of variants of the Riemann Hypothesis. An interesting follow-up question would be how the algorithms in this paper would perform without assuming variants of the Riemann Hypothesis. To aid such research we now identify which type of Riemann Hypothesis is used for which task and discuss the possible impact of letting go of these hypotheses.

(1) **Random Walk Theorem.** In Theorem 6.4, the Riemann Hypothesis for all Hecke L-functions associated with a Hecke-character over a number field K (with fixed modulus $\mathfrak{m}$) is assumed. The Riemann Hypothesis in this form here seems indispensable for this proof, due to its influence on the bound on the Hecke eigenvalues (see Equation (171)).

Without the Riemann Hypothesis, one might have to resort to an alternative, weaker bound, optimistically maybe of the following form, inspired by bounds [74, 31] for the prime counting function:

$$\lambda_\chi = O\left(c_{K,\mathfrak{m}} \cdot e^{-\sqrt{\log(B)}}\right). \tag{137}$$

Here, $c_{K,\mathfrak{m}}$ is some constant depending on K and $\mathfrak{m}$, and the big-O is, as it is throughout the entire present work, absolute. A bound as Equation (137) for all eigenvalues of Hecke characters under Hecke operators, of which we do not know a proof and what the constant $c_{K,\mathfrak{m}}$ then might be, implies then that we are required to choose

$$B = O(e^{\log^2(c_{K,\mathfrak{m}})}),$$

which might or might not be useful, depending on the value of $c_{K,\mathfrak{m}}$.

(2) **Estimation of the factor $\mathcal{N}(\mathfrak{m}_0)/\phi(\mathfrak{m}_0)$.** In Section 11, the Riemann hypothesis for the Dedekind zeta function of K is assumed to estimate the factor $\mathcal{N}(\mathfrak{m}_0)/\phi(\mathfrak{m}_0)$. This estimation is used to diminish the impact of ρ_K on the running time of the $\mathbb{S}$-unit computation algorithm. To be more precise, this estimate on this factor allows for making the running time depend on ρ_K^{cut} instead of ρ_K (see Theorem 21.2, also the left-hand down-sloping part of the red graph in Figure 2).

Not assuming the Riemann hypothesis for the Dedekind zeta function of K might impact the bound in Proposition 11.2 as well as the bound in Lemma 11.4. Hence, the precise implications of letting go of the Riemann hypothesis is not totally clear, but might heavily impact the dependency of the provable running time on ρ_K.

(3) **Density of smooth ideals.** In Section 14, a combinatorial bound is given for the density of smooth ideals, in which bounds on the number of prime ideals in a number field are used. Letting go of the Riemann Hypothesis here might have as a consequence that heavier restrictions on the numbers B_{sm} and A are required.

(4) **Computation of an approximation of ρ_K.** In multiple parts of the $\mathbb{S}$-unit computation algorithm an approximation of the Dedekind zeta function residue ρ_K is required. In Algorithm 3, one uses this approximation whenever ρ_K is large, to apply the 'modulus trick' in order to diminish the influence of ρ_K on the running time. In the full algorithm, Algorithm 6, an approximation of ρ_K (or equivalently, of $R_K \cdot h_K$) is needed to decide whether the algorithm indeed found the full logarithmic $\mathbb{S}$-unit lattice (see Section 19.5). The knowledge of an approximation of ρ_K thus allows to change the full algorithm from a Las Vegas algorithm (i.e., the run time is a random variable but the answer of the algorithm is always correct) into a Monte Carlo algorithm (i.e., the run time is bounded but the answer might be wrong or inconclusive with a certain fixed small probability).

If the Riemann hypothesis for the Dedekind zeta function of K is not assumed, longer Euler products are required to approximate ρ_K. Depending on the length of these Euler products compared to the running time of the rest of the entire algorithm (which is already at least $L_{|\Delta_K|}(1/2)$), letting go of the Riemann hypothesis might or might not influence the running time of the algorithm.

PART 3

Provable lattice reduction techniques

22. Introduction

Roughly, this last part of the current paper can be divided into two main subjects: The post-processing part, which is about processing the generators of the log-$\mathbb{S}$-unit lattice into a basis thereof (Sections 23 to 25); and the BKZ-algorithm part on approximate bases, which is needed to find short vectors in ideal lattices (Sections 26 to 28).

22.1. **Post-processing.**

The first subject is the post-processing of the generating matrix of the log-$\mathbb{S}$-unit lattice in order to get a basis (and hence a fundamental set of $\mathbb{S}$-units) of this log-$\mathbb{S}$-unit lattice, see also Section 19.

In this post-processing stage, one consecutively computes the *rank*, the *determinant* and a well-conditioned *basis* of the lattice generated by the (approximate) generating matrix. The rank and determinant are computed first in order to check if the generating matrix indeed generates the entire log-$\mathbb{S}$-unit lattice (and not a sublattice thereof).

This post-processing makes use of the Buchmann-Kessler-Pohst algorithm, an algorithm that, roughly said, LLL-reduces a real-valued basis of one which only knows a sufficiently good approximation of. Though, in the work of Buchmann, Kessler and Pohst [18], it is assumed that the rank and the determinant of the generated lattice is *known*. In our application, though, these quantities are not known but instead need to be computed.

Hence, a slight modification of the analysis of Buchmann, Kessler and Pohst is treated in Section 23 in order to show that one can indeed *compute* the rank and a ‘relatively well conditioned’ approximate basis of the generated lattice with use of the Buchmann-Kessler-Pohst algorithm. This relatively well conditioned approximate basis can then be used to compute an approximation of the determinant (see Section 24)

The entire compound result with the exact precision needed, applied to our $\mathbb{S}$-unit computation algorithm of Part 2, is then treated in Section 25.

22.2. **The BKZ-algorithm on ideal lattices.**

As we aim for a provable running time in this paper, we use a variant of the BKZ-algorithm [39] with a proven upper bound on the run time. The challenge with the lattices occurring in this paper, is that only an approximate basis of them are known. Hence, to make use of this variant of the BKZ-algorithm, we have to prove that this variant behaves well on small changes of the input.

In order to do so, a preliminary section about lattices is required (Section 26). After that, the running time of the variant of the BKZ-algorithm [39] on *integer* bases is analyzed, following a sketch of [39, Section 3, Cost of BKZ’] (Section 27). Lastly, this BKZ-variant is applied on an approximate basis, with use of the Buchmann-Kessler-Pohst algorithm (Section 28).

23. Buchmann-Kessler-Pohst algorithm

23.1. **Introduction.** In this section, we will treat a slightly modified analysis of the algorithm of Buchmann, Pohst and Kessler [18, 19]. The original algorithm allows to compute an approximate basis of a lattice Λ from an approximate generating matrix of a lattice (i.e., a matrix whose rows generate the lattice), provided that the approximation of this generating matrix is good enough. In this original algorithm the rank and the determinant of the lattice generated by the generating matrix are assumed to be known.

In our case, computing (a fundamental system of) $\mathbb{S}$-units, the rank and the determinant of the lattice of the exact generating matrix are *not* known. Rather, we would like to approximately compute these quantities, in order to deduce whether the generating matrix generates the entire logarithmic $\mathbb{S}$-unit lattice or not.

Exactly this lack of knowing the rank and determinant of the lattice (spanned by the exact generating matrix) requires us to slightly modify the analysis of Buchmann, Pohst and Kessler. In this modified analysis, the algorithm of Buchmann, Pohst and Kessler is actually applied twice (this was already suggested in [18, p. 9, remark after Theorem 4.2]). These two applications of the same algorithm serve two different goals.

During the first application of the Buchmann-Kessler-Pohst algorithm, one computes an (approximate) *basis* of the lattice at hand. From this approximate basis one can directly obtain the *rank* of the lattice. This first (approximate) basis, though, might have bad conditioning properties and too long vectors. It is exactly the purpose of the second application of the Buchmann-Kessler-Pohst algorithm to transform this approximate basis in a *well-conditioned* approximate basis, with reasonably short vectors (close to what LLL can achieve).

This well-conditioning of the (approximate) basis allows to approximate the *determinant* of this basis within reasonable precision. This will be the subject of Section 24. Knowing the determinant and the rank of the resulting basis allows to verify whether the basis is an actual basis of the logarithmic $\mathbb{S}$-unit lattice or not (as there are formulas to compute the rank and the covolume of the logarithmic $\mathbb{S}$-unit lattice).

23.2. Preliminaries.

23.2.1. *On the quasi-norm* $\|\mathbf{A}\|_{2,\infty}$. In order to phrase the results of [18] more succinctly, we introduce the following norm-like function on matrices. It is just the maximum over the 2-norms of the rows of the matrix.

Definition 23.1. *For any $k \times m$-matrix $\mathbf{A}$, we denote*

$$\|\mathbf{A}\|_{2,\infty} := \max_{1\le j\le k} \|\mathbf{a}_j\|,$$

where $\mathbf{a}_j$ are the rows *of $\mathbf{A}$, and where $\|\cdot\|$ is the Euclidean norm on the row vectors.*

This is not a matrix norm as it is not submultiplicative, but it is almost submultiplicative:

Lemma 23.2. *For a real $m \times \ell$-matrix $\mathbf{M}'$ and a real $\ell \times k$-matrix $\mathbf{M}$, we have*

$$\|\mathbf{M}'\mathbf{M}\|_{2,\infty} \leq \sqrt{\ell} \cdot \|\mathbf{M}'\|_{2,\infty}\|\mathbf{M}\|_{2,\infty}$$

Proof. Take the 2-norm of the t-th row in $\mathbf{M}'\mathbf{M}$, which equals

$$\|\sum_{j=1}^{\ell} m'_{tj}\mathbf{M}_j\| \leq \sum_{j=1}^{\ell} |m'_{tj}|\|\mathbf{M}_j\| \leq \|\mathbf{M}'_t\| \cdot \Big(\sum_{j=1}^{\ell} \|\mathbf{M}_j\|^2\Big)^{1/2} \leq \|\mathbf{M}'_t\| \cdot \sqrt{\ell} \cdot \|\mathbf{M}\|_{2,\infty}$$

where m'_{tj} is the tj-th entry in $\mathbf{M}'$, $\mathbf{M}_j$ is the j-th row of $\mathbf{M}$ and $\mathbf{M}'_t$ is the t-th row of $\mathbf{M}'$. The first inequality is the triangle inequality, the second inequality follows from the Cauchy-Schwarz inequality and the third from the inequality between 2-norms and infinity norms. As this holds for any row in $\mathbf{M}'\mathbf{M}$, we conclude that

$$\|\mathbf{M}'\mathbf{M}\|_{2,\infty} \leq \sqrt{\ell} \cdot \|\mathbf{M}'\|_{2,\infty}\|\mathbf{M}\|_{2,\infty}.$$

□

23.2.2. *Rounding in the Buchmann-Kessler-Pohst algorithm.* In the original analysis of the Buchmann-Kessler-Pohst algorithm [18], the input approximate generating matrix $\mathbf{A} \in \mathbb{R}^{k\times n}$ (consisting of rows $\mathbf{a}_j \in \mathbb{R}^n$) is scaled up by 2^q and rounded entry-wise to the nearest integer, thus obtaining an integer matrix $\hat{\mathbf{A}} \in \mathbb{Z}^{k\times n}$ (consisting of rows $\hat{\mathbf{a}}_j \in \mathbb{Z}^n$) [18, Section 2]. This is done this way because then the LLL-algorithm can be applied to $\hat{\mathbf{A}}$ (the LLL-algorithm is generally only applied on integer (or rational) matrices). As a result, such an integer matrix $\hat{\mathbf{A}}$ satisfies $\|\hat{\mathbf{A}} - 2^q\mathbf{A}\|_{2,\infty} \leq \sqrt{n}/2$ [18, Equation (1)].

In our analysis, we deviate from this. Going over the result of Buchmann and Kessler [18], one can verify that the only requirements on $\hat{\mathbf{A}}$ for their algorithm to work is:

- Closeness to $2^q\mathbf{A}$, i.e., $\|\hat{\mathbf{A}} - 2^q\mathbf{A}\|_{2,\infty} \leq \sqrt{n}/2$ for some sufficiently large $q \in \mathbb{N}$, see [18, Equation (1)].
- Being able to call LLL on $\hat{\mathbf{A}}$; so, in fact $\hat{\mathbf{A}}$ is allowed be a rational matrix, as long as the denominators not too large.

By choosing, for example, $\hat{\mathbf{A}} \in \frac{1}{2}\mathbb{Z}^{k\times n}$, the inequality $\|\hat{\mathbf{A}} - 2^q\mathbf{A}\|_{2,\infty} \leq \sqrt{n}/2$ can be easily achieved, using a sufficiently good approximation of $\mathbf{A}$ to construct $\hat{\mathbf{A}}$.

Summarizing, for our purposes it is not required that $\hat{\mathbf{A}}$ is integral (rationals with small common denominators will suffice, too) nor that $\hat{\mathbf{A}}$ is obtained by rounding $2^q\mathbf{A}$. For this reason, the following notation turns out to be useful.

Notation 23.3 (Binary approximation)**.** *A matrix $\tilde{\mathbf{A}} \in \mathbb{Q}^{k\times n}$ is called a binary approximation of $\mathbf{A} \in \mathbb{R}^{k\times n}$ with precision ε_0 if $2^q \cdot \tilde{\mathbf{A}} \in \mathbb{Z}^{k\times n}$ for some $q \in \mathbb{N}$ and $\|\mathbf{A} - \tilde{\mathbf{A}}\|_{2,\infty} \leq \varepsilon_0$.*

Note that we can choose $q = \log_2(\varepsilon_0 \cdot n)$ in above notation, as that precision is sufficient to obtain the approximation $\|\mathbf{A} - \tilde{\mathbf{A}}\|_{2,\infty} \leq \varepsilon_0$. As a result, there always exists such a matrix $\tilde{\mathbf{A}} \in \mathbb{Q}^{k\times n}$ whose bit-size is bounded by $q^2 \cdot k \cdot n \cdot \log(\|\mathbf{A}\|_{2,\infty}) = \log_2(\varepsilon_0 n)^2 \cdot k \cdot n \cdot \log(\|\mathbf{A}\|_{2,\infty})$ (each coefficient of $\tilde{\mathbf{A}}$ is a rational number with numerator bounded by $2^q \cdot \|\mathbf{A}\|_{2,\infty}$ and denominator bounded by 2^q). Moreover, from any rational matrix $\tilde{\mathbf{A}}'$ satisfying $\|\mathbf{A} - \tilde{\mathbf{A}}'\|_{2,\infty} \leq \varepsilon_0/2$, one can compute in polynomial time a matrix $\tilde{\mathbf{A}}$ with bit-size as above (polynomial in n, k, $\log(1/\varepsilon_0)$ and such that $\log\|\mathbf{A}\|_{2,\infty}$) and $\|\mathbf{A} - \tilde{\mathbf{A}}\|_{2,\infty} \leq \varepsilon_0$.

23.3. **The algorithm of Buchmann, Pohst and Kessler.** In the following lemma, we start by updating slightly the analysis of [18, Theorem 4.1], to make it compatible with our matrix norm, and also take into account the fact that the rank and determinant of our lattice are not known. In the later theorem (Theorem 23.5) we will apply this lemma twice, first to obtain just any basis of the lattice, and the second time to obtain a well-conditioned basis consisting of reasonably short vectors.

Lemma 23.4. *Let $\Lambda \subseteq \mathbb{Z}^{n_1} \times \mathbb{R}^{n_2}$ be a lattice with $\lambda_1(\Lambda) \geq \mu$ and $\operatorname{rank}(\Lambda) \leq r_0$. Let $\mathbf{A}$ be a $k \times (n_1 + n_2)$-matrix whose rows generate the lattice Λ with $k \geq n_2 \geq 1$, let*

$$C := 2^{4k} \cdot \left(\frac{r_0 \cdot \|\mathbf{A}\|_{2,\infty}}{\mu} \right)^{r_0+1},$$

and let $\tilde{\mathbf{A}}$ be a binary approximation of $\mathbf{A}$ satisfying $\|\tilde{\mathbf{A}} - \mathbf{A}\|_{2,\infty} < \frac{1}{4}\mu \cdot C^{-1}$.

Then there exists an algorithm that, on input $\tilde{\mathbf{A}}, \mu, r_0$, computes a matrix $\mathbf{M} \in \mathbb{Z}^{k \times r}$ with $r = rank(\Lambda)$ in time polynomial in k, n_1, n_2, $\log\|\mathbf{A}\|_{2,\infty}, \log(1/\mu)$ and the bit-size of its input, and such that

(i) $\mathbf{MA} = \mathbf{B}$ *is a basis of* Λ.
(ii) $\|\mathbf{M}\|_{2,\infty} \leq C/\sqrt{k} \leq C$,
(iii) $\|\mathbf{B}\|_{2,\infty} \leq 4^k \cdot \|\mathbf{A}\|_{2,\infty}$.
(iv) Writing $\tilde{\mathbf{B}} = \mathbf{M}\tilde{\mathbf{A}}$, *we have* $\|\tilde{\mathbf{B}} - \mathbf{B}\|_{2,\infty} \leq C \cdot \|\tilde{\mathbf{A}} - \mathbf{A}\|_{2,\infty}$.

Additionally, if $\mathbf{A}$ happened itself to be a basis of Λ, i.e., $k = r = \operatorname{rank}(\Lambda)$, we have, for all $j \in \{1, \ldots, r\}$,

$$\|\mathbf{b}_j\| \leq (\sqrt{rn_2} + 2)2^{\frac{r-1}{2}} \cdot \lambda_j(\Lambda),$$

where $\mathbf{b}_j$ are the rows of $\mathbf{B} = (\mathbf{b}_1, \ldots, \mathbf{b}_r)$.

Proof. **Constructing the matrix $\hat{\mathbf{A}}$ from $\tilde{\mathbf{A}}$.** Using the discussion after Notation 23.3, we can also replace the matrix $\tilde{\mathbf{A}}$ by a matrix with bounded bit-size (in time polynomial in the bit-size of $\tilde{\mathbf{A}}$). Hence, in the rest of the proof, we will assume without loss of generality that the bit-size of $\tilde{\mathbf{A}}$ is polynomial in k, n_1, n_2, $\log\|\mathbf{A}\|_{2,\infty}$, $\log(1/\mu)$, and $\tilde{\mathbf{A}}$ satisfies $\|\mathbf{A} - \tilde{\mathbf{A}}\|_{2,\infty} \leq 1/4 \cdot \mu \cdot C^{-1}$.

The algorithm first creates the matrix $\hat{\mathbf{A}}$ defined by $\hat{\mathbf{A}}_{ij} = \frac{1}{2}\lfloor 2^{q+1}\tilde{\mathbf{A}}_{ij} \rceil$ with $q = \lfloor \log_2(T) \rfloor$, where $\lfloor\cdot\rceil$ denotes rounding to the nearest integer, $\lfloor\cdot\rfloor$ denotes rounding down, and

$$T = \frac{2^{3k}}{\mu} \cdot \left(\frac{r_0 \cdot \|\mathbf{A}\|_{2,\infty}}{\mu} \right)^{r_0} = \frac{2^{-k}}{\mu} \left(\frac{r_0 \cdot \|\mathbf{A}\|_{2,\infty}}{\mu} \right)^{-1} \cdot C \leq C/(2\mu). \tag{138}$$

Note that computing $\hat{\mathbf{A}}$ can be done in time polynomial in q and in the bit-size of $\tilde{\mathbf{A}}$. The entries of $\hat{\mathbf{A}}$ lie in $\frac{1}{2}\mathbb{Z}$, and the matrix $\hat{\mathbf{A}}$ satisfies

$$\|\hat{\mathbf{A}} - 2^q\mathbf{A}\|_{2,\infty} \leq \underbrace{\|\hat{\mathbf{A}} - 2^q\tilde{\mathbf{A}}\|_{2,\infty}}_{\leq \sqrt{n_2}/4} + \underbrace{\|2^q\tilde{\mathbf{A}} - 2^q\mathbf{A}\|_{2,\infty}}_{\leq 2^q \cdot \frac{1}{2} \cdot \mu/C \leq 1/4} \leq \sqrt{n_2}/2.$$

According to the discussion in Section 23.2.2 this is sufficient to apply the Buchmann-Kessler-Pohst algorithm.

Applying Buchmann-Kessler-Pohst to $[\mathbf{I} \mid \hat{\mathbf{A}}]$. This Buchmann-Kessler-Pohst algorithm [18, Theorem 4.1] essentially consists of applying LLL-reduction to the matrix $[\mathbf{I} \mid \hat{\mathbf{A}}]$ consisting of the horizontal concatenation of $\hat{\mathbf{A}}$ and $\mathbf{I}$, the $k \times k$ identity matrix.

Put,[33] in order to satisfy the prerequisites of [18, Proposition 3.2],

$$\lambda := 2^k \cdot \left(\frac{r_0\|\mathbf{A}\|_{2,\infty}}{\mu}\right)^{r_0} \geq (\tfrac{1}{2}k\sqrt{n_2} + \sqrt{k}) \cdot \frac{\|\mathbf{A}\|_{2,\infty}^r}{\mathrm{Vol}(\Lambda)}. \tag{139}$$

where $r = \mathrm{rank}(\Lambda) \leq r_0$; in the inequality we made use of Minkowski's inequality[34] and the fact that $2^k \geq \frac{1}{2}k\sqrt{n_2} + \sqrt{k}$ (since we assumed $k \geq n_2$).

Note that $q = \lfloor \log_2(T) \rfloor$ satisfies [18, Equation 7], since we have

$$\begin{aligned} 2^q &\geq \tfrac{1}{2} \cdot \frac{2^{3k}}{\mu} \cdot \left(\frac{r_0 \cdot \|\mathbf{A}\|_{2,\infty}}{\mu}\right)^{r_0} = \tfrac{1}{2} \cdot \frac{2^{2k}}{\mu} \cdot \lambda \\ &> (\sqrt{n_2 k} + 2) \cdot 2^{\frac{k-3}{2}} \cdot \frac{\lambda}{\mu}, \end{aligned}$$

by Equation (139) and because $\frac{1}{2} \cdot 2^{2k} > (\sqrt{n_2 k} + 2)2^{\frac{k-3}{2}}$ for all $k \geq n_2 \geq 1$.

With these parameters λ and q, one can apply [18, Theorem 4.1]: LLL-reducing $[\mathbf{I} \mid \hat{\mathbf{A}}]$ yields an integral matrix $(\mathbf{R}^\top, \mathbf{M}^\top)^\top$ such that

$$\begin{bmatrix} \mathbf{R} \\ \mathbf{M} \end{bmatrix} \cdot [\mathbf{I} \quad \hat{\mathbf{A}}] = \begin{bmatrix} \mathbf{R} & \mathbf{R}\hat{\mathbf{A}} \\ \mathbf{M} & \mathbf{M}\hat{\mathbf{A}} \end{bmatrix} = \begin{bmatrix} \mathbf{R} & \hat{\mathbf{E}} \\ \mathbf{M} & \hat{\mathbf{B}} \end{bmatrix} \tag{140}$$

for which holds $\mathbf{RA} = \mathbf{0}$, and $\mathbf{MA} = \mathbf{B}$, a basis of Λ. Furthermore, every row $\hat{\mathbf{e}}$ of $\hat{\mathbf{E}}$ satisfies ([18, Theorem 4.1, first equation on page 8])

$$\|\hat{\mathbf{e}}\| \leq 2^{\frac{k-1}{2}}\lambda$$

and every row $\hat{\mathbf{b}}$ of $\hat{\mathbf{B}}$ satisfies ([18, Proposition 3.1])

$$\|\hat{\mathbf{b}}\| > 2^{\frac{k-1}{2}}\lambda.$$

Identifying M and $\hat{\mathbf{B}}$, proving (i). This gives means to distinguish where in the output matrix of Equation (140) the approximate basis elements $\hat{\mathbf{B}}$ are. Also, this gives us a way to identify $\mathbf{M}$, as it shares the same rows as $\hat{\mathbf{B}}$. Therefore, we can compute an $\mathbf{M}$ that satisfies part (i) of the lemma.

[33] We replaced α in [18, Proposition 3.2] here by $\|\mathbf{A}\|_{2,\infty}$. As $\alpha \in \mathbb{R}_{>0}$ is required to be an upper bound on the 2-norms of the rows of $\mathbf{A}$, $\|\mathbf{A}\|_{2,\infty} = \max_j \|\mathbf{g}_j\|$ clearly suffices.

[34] Minkowski's inequality states that $1/\mathrm{Vol}(\Lambda) \leq (r/\lambda_1(\Lambda))^r \leq (r/\mu)^r$ with $r = \mathrm{rank}(\Lambda)$. By the fact that $\|\mathbf{A}\|_{2,\infty}/\mu \geq 1$ and $r_0 \geq r$, we can conclude that

$$\left(\frac{r_0\|\mathbf{A}\|_{2,\infty}}{\mu}\right)^{r_0} \geq \left(\frac{r\|\mathbf{A}\|_{2,\infty}}{\mu}\right)^{r} \geq \frac{\|\mathbf{A}\|_{2,\infty}^r}{\mathrm{Vol}(\Lambda)}$$

Bounds on M, B and $\|\mathbf{B}-\tilde{\mathbf{B}}\|_{2,\infty}$. By [18, Theorem 4.1, Equation 9, bottom inequality], we have,[35] since $2^q \le T$ (see Equation (138)),

$$\|\mathbf{m}\| \le 2^{\frac{k-1}{2}+q+1} \cdot \|\mathbf{A}\|_{2,\infty} = 2^{\frac{k+1}{2}} \cdot \|\mathbf{A}\|_{2,\infty} \cdot T \tag{141}$$

$$\le 2^{\frac{k+1}{2}} \cdot \|\mathbf{A}\|_{2,\infty} \cdot \frac{2^{3k}}{\mu} \cdot \left(\frac{r_0 \cdot \|\mathbf{A}\|_{2,\infty}}{\mu}\right)^{r_0} \tag{142}$$

$$\le \frac{2^{4k}}{\sqrt{k}} \cdot \left(\frac{r_0 \cdot \|\mathbf{A}\|_{2,\infty}}{\mu}\right)^{r_0+1} \tag{143}$$

$$\le \frac{C}{\sqrt{k}} \le C \quad \text{for all rows } \mathbf{m} \text{ of } \mathbf{M}, \tag{144}$$

where we used $2^{\frac{k+1}{2}} \le 2^k/\sqrt{k}$ for all $k \in \mathbb{Z}_{>0}$. Additionally, by [18, Theorem 4.1, Equation 10], we have,

$$\|\mathbf{b}\| \le (\sqrt{kn_2}+2)2^{\frac{k-1}{2}} \cdot \|\mathbf{A}\|_{2,\infty} \tag{145}$$

$$\le 2^{2k} \cdot \|\mathbf{A}\|_{2,\infty} \quad \text{for all rows } \mathbf{b} \text{ of } \mathbf{B}, \tag{146}$$

which prove (ii) and (iii), as $\|\mathbf{M}\|_{2,\infty}$ and $\|\mathbf{B}\|_{2,\infty}$ are just the maximum 2-norm of the rows of the respective matrices.

For part (iv) use the 'almost submultiplicativity' of $\|\cdot\|_{2,\infty}$ (Lemma 23.2) and Equation (144), to obtain

$$\|\tilde{\mathbf{B}}-\mathbf{B}\|_{2,\infty} = \|\mathbf{M}(\tilde{\mathbf{A}}-\mathbf{A})\|_{2,\infty} \le \sqrt{k} \cdot \|\mathbf{M}\|_{2,\infty} \cdot \|\tilde{\mathbf{A}}-\mathbf{A}\|_{2,\infty} \le C \cdot \|\tilde{\mathbf{A}}-\mathbf{A}\|_{2,\infty}.$$

LLL-alike bounds for B when A is itself a basis of Λ. In the case that $\mathbf{A}$ itself happens to be a basis, sharper bounds on $\mathbf{B}$ can be established, as can be deduced[36] from [18, Theorem 4.1, Equation 10]. Applying this bound yields, for all $j \in \{1,\dots,r\}$,

$$\|\mathbf{b}_j\| \le (\sqrt{kn_2}+2)2^{\frac{k-1}{2}} \cdot \lambda_j(\Lambda)$$

where $\mathbf{b}_j$ is the j-th row of $\mathbf{B}$.

Running time. For the running time, note that the most costly part of the algorithm is to run the LLL algorithm on the matrix $[\mathbf{I} \mid \hat{\mathbf{A}}]$. This can be done in polynomial time in the bit-size of the matrix, which is polynomial in k, n_1, n_2, $\log\|\mathbf{A}\|_{2,\infty}$ and $\log(1/\mu)$, as desired. □

Theorem 23.5 (Buchmann-Kessler-Pohst). *Let $\Lambda \subseteq \mathbb{Z}^{n_1} \times \mathbb{R}^{n_2}$ be a lattice with $\lambda_1(\Lambda) \ge \mu$ and $\operatorname{rank}(\Lambda) \le r_0$. Let $\mathbf{A}$ be a $k \times (n_1+n_2)$-matrix whose rows generate the lattice Λ with $k \ge n_2 \ge 1$, let*

$$C_0 = 2^{8k} \cdot \left(\frac{r_0 \cdot 4^k \cdot \|\mathbf{A}\|_{2,\infty}}{\mu}\right)^{2(r_0+1)},$$

and let $\tilde{\mathbf{A}}$ be a binary approximation of $\mathbf{A}$ that satisfies $\|\tilde{\mathbf{A}}-\mathbf{A}\|_{2,\infty} < \frac{1}{4} \cdot \mu \cdot C_0^{-1}$.

[35] In [18, Eq. 10], the lattice successive minima $\lambda_j(L_r)$ of the lattice L_r are used; this lattice L_r is any lattice spanned by r rows of $\mathbf{A}$. So these successive minima $\lambda_j(L_r)$ are trivially bounded by α, the uniform bound on the lengths of the rows of $\mathbf{A}$. Additionally, we replaced α by $\|\mathbf{A}\|_{2,\infty}$ again.

[36] We substitute $k = r$ (as $\mathbf{B}$ consists of r rows) and $L_r = \Lambda$. Here L_r is defined as any lattice defined by any r row vectors from $\mathbf{B}$. As $\mathbf{B}$ consists of only r row vectors and is a basis of Λ, we have $L_r = \lambda$.

Then, there exists an algorithm that, on input $\tilde{\mathbf{A}}, \mu, r_0$*, computes in time polynomial in* k*,* n_1*,* n_2*,* $\log\|\mathbf{A}\|_{2,\infty}$*,* $\log(1/\mu)$ *and the bit-size of its input, a matrix* $\mathbf{N} \in \mathbb{Z}^{r\times k}$ *with* $r = \operatorname{rank}(\Lambda)$ *such that*

(i) $\|\mathbf{N}\|_{2,\infty} \leq C_0$

(ii) $\mathbf{B} = \mathbf{N}\mathbf{A}$ *is a basis of* Λ*, whose rows satisfy, for* $j \in \{1, \ldots, r\}$*,*

$$\|\mathbf{b}_j\| \leq (\sqrt{rn_2} + 2) \cdot 2^{\frac{r-1}{2}} \cdot \lambda_j(\Lambda)$$

(iii) The approximate basis $\tilde{\mathbf{B}} = \mathbf{N}\tilde{\mathbf{A}}$ *satisfies* $\|\tilde{\mathbf{B}} - \mathbf{B}\|_{2,\infty} \leq C_0 \cdot \|\tilde{\mathbf{A}} - \mathbf{A}\|_{2,\infty}$*.*

Proof. We will apply the Buchmann-Kessler-Pohst algorithm as in Lemma 23.4 twice: the first time we apply it to $\tilde{\mathbf{A}}$ to obtain an approximate basis $\tilde{\mathbf{B}}_0$ of Λ, and the second time we apply it to $\tilde{\mathbf{B}}_0$ to obtain an approximate *good* basis $\tilde{\mathbf{B}}$ of Λ. This basis $\tilde{\mathbf{B}}$ is called 'good' because it is an approximation of a basis $\mathbf{B}$ with a guarantee on the shortness of its basis vectors.

First application of Lemma 23.4.

We have that $C_0 > 2^{4k} \cdot \left(\frac{r_0 \cdot \|\mathbf{A}\|_{2,\infty}}{\mu}\right)^{r_0+1} = C$ from Lemma 23.4. Therefore, the approximation $\tilde{\mathbf{A}}$ of $\mathbf{A}$ is sufficiently good to apply Lemma 23.4, and compute $\mathbf{M} \in \mathbb{Z}^{k\times r}$ such that $\mathbf{M}\mathbf{A} = \mathbf{B}_0$ is a basis of Λ. Additionally, this $\mathbf{M}$ and $\mathbf{B}_0$ satisfy

$$\|\mathbf{M}\|_{2,\infty} \leq C \quad \text{and} \quad \|\mathbf{B}_0\|_{2,\infty} \leq 4^k \cdot \|\mathbf{A}\|_{2,\infty}, \tag{147}$$

and $\tilde{\mathbf{B}}_0 := \mathbf{M}\tilde{\mathbf{A}}$ satisfies

$$\|\tilde{\mathbf{B}}_0 - \mathbf{B}_0\|_{2,\infty} \leq C \cdot \|\tilde{\mathbf{A}} - \mathbf{A}\|_{2,\infty} \leq C \cdot \mu \cdot C_0^{-1} \tag{148}$$

Moreover, the bit-size of $\tilde{\mathbf{B}}_0$ is polynomially bounded by the bit-size of $\tilde{\mathbf{A}}$, and the bit-size of $\mathbf{M}$, the later being polynomially bounded by k, n_1, n_2, $\log\|\mathbf{A}\|_{2,\infty}$ and $\log(1/\mu)$.

Second application of Lemma 23.4.

By Equation (148) we deduce that $\|\tilde{\mathbf{B}}_0 - \mathbf{B}_0\|_{2,\infty} < \mu(C_0/C)^{-1}$; since $\tilde{\mathbf{B}}_0 = \mathbf{M}\tilde{\mathbf{A}}$, this matrix $\tilde{\mathbf{B}}_0$ is a binary approximation of $\mathbf{B}_0$. In order to apply Lemma 23.4, we need to show that $C_0/C \geq 2^{4r} \cdot \left(\frac{r \cdot \|\mathbf{B}_0\|_{2,\infty}}{\mu}\right)^{r+1} =: C'$, i.e., that the precision of $\tilde{\mathbf{B}}_0$ is good enough . By the choice of C_0, the bound $4^k \cdot \|\mathbf{A}\|_{2,\infty} \geq \|\mathbf{B}_0\|_{2,\infty}$ (Equation (147)), $r_0 \geq r$ and $k \geq r$, we have

$$\frac{C_0}{C} = \frac{2^{8k} \cdot \left(\frac{r_0 \cdot 4^k \cdot \|\mathbf{A}\|_{2,\infty}}{\mu}\right)^{2(r_0+1)}}{2^{4k} \cdot \left(\frac{r_0 \cdot \|\mathbf{A}\|_{2,\infty}}{\mu}\right)^{r_0+1}} \geq 2^{4k} \cdot \left(\frac{r_0 \cdot 4^k \cdot \|\mathbf{A}\|_{2,\infty}}{\mu}\right)^{r_0+1} \tag{149}$$

$$\geq 2^{4k} \cdot \left(\frac{r_0 \cdot \|\mathbf{B}_0\|_{2,\infty}}{\mu}\right)^{r_0+1} \geq 2^{4r} \cdot \left(\frac{r \cdot \|\mathbf{B}_0\|_{2,\infty}}{\mu}\right)^{r+1} =: C' \tag{150}$$

Therefore, we can apply Lemma 23.4 again, but now with $k = r$ and C' as in Equation (150), and with input $(\tilde{\mathbf{B}}_0, \mu, r)$ (note that r is now known, and we still have $\mu \leq \lambda_1(\Lambda)$). This yields $\mathbf{M}'$ such that $\mathbf{M}'\mathbf{B}_0 = \mathbf{B}$ is a basis of Λ,

$$\|\mathbf{M}'\|_{2,\infty} \leq C'/\sqrt{r} \tag{151}$$

and for all $j \in \{1, \ldots, r\}$ we have

$$\|\mathbf{b}'_j\| \leq (\sqrt{rn_2} + 2) \cdot 2^{\frac{r-1}{2}} \cdot \lambda_j(\Lambda),$$

which proves (ii), putting $\mathbf{N} = \mathbf{M}'\mathbf{M}$.

For (i), we apply the 'almost submultiplicativity' (Lemma 23.2) to $\mathbf{N} = \mathbf{M}'\mathbf{M}$, which yields, by Equation (151) and Equation (150) ($C_0/C \geq C'$),

$$\|\mathbf{N}\|_{2,\infty} \leq \sqrt{r}\|\mathbf{M}'\|_{2,\infty}\|\mathbf{M}\|_{2,\infty} \leq C' \cdot C \leq C_0.$$

For (iii), we note that, by the results of Lemma 23.4,

$$\|\tilde{\mathbf{B}} - \mathbf{B}\|_{2,\infty} \leq C' \cdot \|\tilde{\mathbf{B}}_0 - \mathbf{B}_0\|_{2,\infty} \leq C' \cdot C \cdot \|\tilde{\mathbf{A}} - \mathbf{A}\|_{2,\infty} \leq C_0 \cdot \|\tilde{\mathbf{A}} - \mathbf{A}\|_{2,\infty}$$

For the running time of the algorithm, note that we applied the algorithm of Lemma 23.4 twice, once with $(\mathbf{A}, \mu, r_0)$ and once with $(\mathbf{B}_0, \mu, r)$. Recall that the bit-size of $\mathbf{B}_0$ is polynomially bounded by k, n_1, n_2, $\log\|\mathbf{A}\|_{2,\infty}$ and $\log(1/\mu)$ and by the bit-size of $\tilde{\mathbf{A}}$. Using Lemma 23.4 gives the desired upper bound on the total running time of the algorithm. □

24. Computing the determinant of an approximate basis

24.1. **Preliminaries.** In this section we will use the induced 2-norm on matrices, which is defined (for $m \times n$-matrices $\mathbf{A}$) by the rule,

$$\|\mathbf{A}\|_2 := \sup\{\|\mathbf{A}\mathbf{x}\| \mid \mathbf{x} \in \mathbb{C}^n \text{ with } \|\mathbf{x}\| = 1\}.$$

It satisfies the following inequalities

$$\|\mathbf{A}\|_2 \leq \|\mathbf{A}\|_F := (\sum_{i,j} |A_{ij}|^2)^{1/2},$$

where $\|\mathbf{A}\|_F$ is the Frobenius norm. As a result, a $m \times n$ matrix $\mathbf{E}$ with $E_{ij} \leq \varepsilon$ for all i, j, satisfies

$$\|\mathbf{E}\|_2 \leq \|\mathbf{E}\|_F \leq \sqrt{nm} \cdot \varepsilon.$$

We will use the fact that $\|\mathbf{A}\|_2$ corresponds to the largest singular value of $\mathbf{A}$, which gives us

$$\|\mathbf{A}^\top\|_2 = \|\mathbf{A}\|_2. \tag{152}$$

24.2. **The main result of this section.** This section is about the following problem. Suppose we have a rank-n lattice $\Lambda \subseteq \mathbb{R}^m$ of which we can approximate the determinant, say $D \approx \mathrm{Vol}(\Lambda)$. Suppose furthermore that we have an *approximate basis* $\tilde{\mathbf{B}}' \approx \mathbf{B}'$ of a *sub*lattice $\Lambda' \subseteq \Lambda$ (with the same rank n). The task is to decide whether $\Lambda' = \Lambda$, using only the approximations $\tilde{\mathbf{B}}'$ and D.

A straightforward way is to compute $\sqrt{\det((\tilde{\mathbf{B}}')^\top\tilde{\mathbf{B}}')} \approx \mathrm{Vol}(\Lambda')$ and compare it with $D \approx \mathrm{Vol}(\Lambda)$. If the approximations are sufficiently good and D and $\sqrt{\det((\tilde{\mathbf{B}}')^\top\tilde{\mathbf{B}}')}$ are sufficiently close, we can conclude $\Lambda' = \Lambda$. If they are not close, we can conclude $\Lambda' \neq \Lambda$.

This section treats how well $\tilde{\mathbf{B}}'$ needs to approximate $\mathbf{B}'$ (the exact basis of Λ') and how well D needs to approximate $\mathrm{Vol}(\Lambda)$ in order to make this reasoning correct. The main result of this section is the following statement.

Theorem 24.1. *Let $\Lambda \subset \mathbb{R}^m$ be a rank-n lattice (for some $m \geq n \geq 1$). Let $\mathbf{B}' \in \mathbb{R}^{m\times n}$ be a basis of a rank-n sublattice $\Lambda' \subseteq \Lambda \subseteq \mathbb{R}^m$ and let $\tilde{\mathbf{B}}' \in \mathbb{Q}^{m\times n}$ be* *an approximation of* $\mathbf{B}'$ *where the entry-wise error is less than*

$$\varepsilon = 2^{-6} \cdot n^{-(n+4)} \cdot m^{-1} \cdot \left(\prod_j \frac{\|\mathbf{b}'_j\|}{\lambda_j(\Lambda')} \right)^{-2} \cdot \lambda_1(\Lambda')^2 / \|\mathbf{B}'\|_2,$$

and let $D \in [\frac{3}{4}, \frac{5}{4}] \operatorname{Vol}(\Lambda) \cap \mathbb{Q}$.

Then we can decide whether $\Lambda' = \Lambda$ *or not, using only* $\tilde{\mathbf{B}}'$ *and* D*, in time polynomial in the bit-sizes of* $\tilde{\mathbf{B}}'$ *and* D.

24.3. **Some preliminary lemmas.** Before proving Theorem 24.1, we will prove some intermediary lemmas. The first lemma gives an upper bound on the size of the inverse of a square matrix $\mathbf{A}$ that is the basis of a lattice (the upper bound depends on some quantities related to $\mathbf{A}$, which quantify how well-conditioned $\mathbf{A}$ is). This will be used in the second lemma to prove that if two rectangular matrices $\mathbf{B}$ and $\tilde{\mathbf{B}}$ are close coefficient-wise, then the determinants of the square matrices $\mathbf{B}^\top \mathbf{B}$ and $\tilde{\mathbf{B}}^\top \tilde{\mathbf{B}}$ are close too. This second lemma will be central in the proof of Theorem 24.1.

Lemma 24.2. *Suppose* $\mathbf{A} = (\mathbf{a}_1, \ldots, \mathbf{a}_n) \in \mathbb{R}^{n \times n}$ *is a square real matrix and a basis of a lattice* $\Lambda \subset \mathbb{R}^n$. *Then*

$$\|\mathbf{A}^{-1}\|_2 \leq n^{n/2+1} \cdot \lambda_1(\Lambda)^{-1} \cdot \left(\prod_{j=1}^n \frac{\|\mathbf{a}_j\|}{\lambda_j(\Lambda)} \right).$$

Proof. For $j \in \{1, \ldots, n\}$, define $C_j = \frac{\|\mathbf{a}_j\|}{\lambda_j(\Lambda)}$. We have that $\mathbf{A}^{-1} = \frac{1}{\det \mathbf{A}} \operatorname{adj}(\mathbf{A})$. We have that $\operatorname{adj}(\mathbf{A})_{ij}$ is defined by the determinant of the minor of $\mathbf{A}$ where the i-th row and j-th column are deleted. By the Hadamard bound and subsequently by Minkowski's second theorem (see, e.g., [54, Theorem 1.5]),

$$\begin{aligned} |\operatorname{adj}(\mathbf{A})_{ij}| &\leq \prod_{k \neq i} \|\mathbf{a}_k\| = \prod_{k \neq i} C_k \cdot \lambda_k(\Lambda) \\ &\leq n^{n/2} \cdot (\prod_{k \neq i} C_k) \cdot \operatorname{Vol}(\Lambda)/\lambda_i(\Lambda) \leq n^{n/2} (\prod_{k=1}^n C_k) \cdot \operatorname{Vol}(\Lambda)/\lambda_1(\Lambda). \end{aligned}$$

Therefore, by observing that $\det(\mathbf{A}) = \operatorname{Vol}(\Lambda)$, we obtain

$$\|\mathbf{A}^{-1}\|_2 \leq \|\mathbf{A}^{-1}\|_F \leq \frac{1}{\det(\mathbf{A})} \cdot n \cdot \max_{ij} |\operatorname{adj}(\mathbf{A})_{ij}| \leq n^{n/2+1} \cdot (\prod_{k=1}^n C_k)/\lambda_1(\Lambda)$$

□

Lemma 24.3. *Let* $\mathbf{B} \in \mathbb{R}^{m \times n}$ *be a basis of a rank-*n *lattice* $\Lambda \subset \mathbb{R}^m$ *(for some* $m \geq n \geq 1$*), and let* $\tilde{\mathbf{B}}$ *be an approximation of* $\mathbf{B}$ *where the entry-wise error satisfies s* $|\mathbf{B}_{ij} - \tilde{\mathbf{B}}_{ij}| \leq \varepsilon$ *for every* $i \in \{1, \ldots, m\}, j \in \{1, \ldots, n\}$*, where*

$$\varepsilon = 2^{-6} \cdot n^{-(n+4)} \cdot m^{-1} \cdot \left(\prod_j \frac{\|\mathbf{b}_j\|}{\lambda_j(\Lambda)} \right)^{-2} \cdot \lambda_1(\Lambda)^2 / \|\mathbf{B}\|_2. \tag{153}$$

Then

$$\det(\tilde{\mathbf{B}}^\top \tilde{\mathbf{B}}) \in [\tfrac{7}{8}, \tfrac{9}{8}] \cdot \det(\mathbf{B}^\top \mathbf{B}).$$

Proof. We have, by [43, Corollary 2.14], for $n \times n$ complex matrices $\mathbf{A}, \mathbf{E}$, where $\mathbf{A}$ is non-singular,

$$\begin{aligned} \frac{|\det(\mathbf{A}+\mathbf{E}) - \det(\mathbf{A})|}{|\det(\mathbf{A})|} &\leq (\|\mathbf{A}^{-1}\|_2 \|\mathbf{E}\|_2 + 1)^n - 1 \\ &\leq 2n \|\mathbf{A}^{-1}\|_2 \|\mathbf{E}\|_2 \quad \text{ if } \|\mathbf{A}^{-1}\|_2 \|\mathbf{E}\|_2 \leq 1/n. \end{aligned} \tag{154}$$

The last inequality here easily follows from $(x+1)^n - 1 \leq (e^x)^n - 1 \leq 2nx$ for $nx \leq 1$.

We instantiate this inequality with $\mathbf{A} = \mathbf{B}^\top \mathbf{B}$ and $\mathbf{E} = \tilde{\mathbf{B}}^\top \tilde{\mathbf{B}} - \mathbf{B}^\top \mathbf{B}$, so that $\mathbf{A} + \mathbf{E} = \tilde{\mathbf{B}}^\top \tilde{\mathbf{B}}$ (note that $\mathbf{A}$ is non-singular since $\mathbf{B}$ is of rank n, so the instantiation is valid). In order to obtain a meaningful upper bound, let us compute bounds on the two quantities $\|\mathbf{A}^{-1}\|_2$ and $\|\mathbf{E}\|_2$.

We can rewrite $\mathbf{E}$ as

$$\begin{aligned} \mathbf{E} &= \big(\mathbf{B} + (\tilde{\mathbf{B}} - \mathbf{B})\big)^\top \cdot \big(\mathbf{B} + (\tilde{\mathbf{B}} - \mathbf{B})\big) - \mathbf{B}^\top \mathbf{B} \\ &= \mathbf{B}^\top \cdot (\tilde{\mathbf{B}} - \mathbf{B}) + (\tilde{\mathbf{B}} - \mathbf{B})^\top \cdot \mathbf{B} + (\tilde{\mathbf{B}} - \mathbf{B})^\top \cdot (\tilde{\mathbf{B}} - \mathbf{B}). \end{aligned}$$

Using the sub-multiplicativity of the induced 2-norm and Equation (152), we obtain

$$\|\mathbf{E}\|_2 \leq 2 \cdot \|\mathbf{B}\|_2 \cdot \|\tilde{\mathbf{B}} - \mathbf{B}\|_2 + \|\tilde{\mathbf{B}} - \mathbf{B}\|_2^2.$$

By assumption on the entry-wise error, we know that $\|\tilde{\mathbf{B}} - \mathbf{B}\|_2 \leq \|\tilde{\mathbf{B}} - \mathbf{B}\|_F \leq \sqrt{nm} \cdot \varepsilon$. Moreover, it holds that $\|\mathbf{B}\|_2 \geq \|\mathbf{b}_1\| \geq \lambda_1(\Lambda) \geq \lambda_1(\Lambda)^2 / \|\mathbf{B}\|_2 \geq \varepsilon$ (where $\mathbf{b}_1$ is the first column vector of $\mathbf{B}$, and see Equation (153)). This yields

$$\|\mathbf{E}\|_2 \leq 3 \cdot n \cdot m \cdot \varepsilon \cdot \|\mathbf{B}\|_2. \tag{155}$$

Let us now compute an upper bound on $\|\mathbf{A}^{-1}\|_2 = \|(\mathbf{B}^\top \mathbf{B})^{-1}\|_2$. Let $(\mathbf{Q}, \mathbf{R})$ be the QR-decomposition of $\mathbf{B}$, with $\mathbf{Q} \in \mathbb{R}^{m \times m}$ orthogonal and $\mathbf{R} \in \mathbb{R}^{m \times n}$ upper triangular, i.e., $\mathbf{B} = \mathbf{QR}$. Since $m \geq n$, we have $\mathbf{R} = \begin{pmatrix} \mathbf{C} \\ 0 \end{pmatrix}$ for some invertible matrix $\mathbf{C} \in \mathbb{R}^{n \times n}$. It then holds, by orthogonality of $\mathbf{Q}$, that $\mathbf{C}^\top \mathbf{C} = \mathbf{B}^\top \mathbf{B}$; that $\|\mathbf{b}_i\| = \|\mathbf{c}_i\|$ for all $1 \leq i \leq n$; and that $\lambda_i(\Lambda) = \lambda_i(\Lambda_{\mathbf{C}})$ for all i's, where $\Lambda_{\mathbf{C}}$ is the lattice spanned by the columns of $\mathbf{C}$. This gives us

$$\begin{aligned} \|\mathbf{A}^{-1}\|_2 = \|(\mathbf{B}^\top \mathbf{B})^{-1}\|_2 &= \|(\mathbf{C}^\top \mathbf{C})^{-1}\|_2 \\ &\leq \|(\mathbf{C}^{-1})^\top\|_2 \cdot \|\mathbf{C}^{-1}\|_2 \\ &= \|\mathbf{C}^{-1}\|_2^2 \\ &\leq \left(n^{n/2+1} \cdot \lambda_1(\Lambda_{\mathbf{C}})^{-1} \cdot \left(\prod_{j=1}^n \frac{\|\mathbf{c}_j\|}{\lambda_j(\Lambda_{\mathbf{C}})} \right) \right)^2 \\ &= n^{n+2} \cdot \lambda_1(\Lambda)^{-2} \cdot \left(\prod_{j=1}^n \frac{\|\mathbf{b}_j\|}{\lambda_j(\Lambda)} \right)^2, \end{aligned} \tag{156}$$

where we used Equation (152) for the third line and Lemma 24.2 on $\mathbf{C}$ (which is n by n as required) for the fourth line.

Combining the bounds Equations (154) to (156) and observing that $\|\mathbf{A}^{-1}\|_2 \cdot \|\mathbf{E}\|_2 \leq 1/n$ by choice of ε, we obtain

$$\begin{aligned}\frac{|\det(\tilde{\mathbf{B}}^\top \tilde{\mathbf{B}}) - \det(\mathbf{B}^\top \mathbf{B})|}{|\det(\mathbf{B}^\top \mathbf{B})|} &\leq 2n \cdot \|\mathbf{A}^{-1}\|_2 \cdot \|\mathbf{E}\|_2 \\ &\leq 2n \cdot \underbrace{n^{n+2} \cdot \lambda_1(\Lambda)^{-2} \cdot \left(\prod_{j=1}^{n} \frac{\|\mathbf{b}_j\|}{\lambda_j(\Lambda)} \right)^2}_{\|\mathbf{A}^{-1}\|_2} \cdot \underbrace{3 \cdot n \cdot m \cdot \varepsilon \cdot \|\mathbf{B}\|_2}_{\|\mathbf{E}\|_2} \\ &\leq 1/8,\end{aligned}$$

which precisely means that $\det(\tilde{\mathbf{B}}^\top \tilde{\mathbf{B}}) \in [\frac{7}{8}, \frac{9}{8}] \cdot \det(\mathbf{B}^\top \mathbf{B})$. □

24.4. **Proving the main result.** We are now equipped to prove Theorem 24.1. The proof is quite short once we have Lemma 24.3.

Proof of Theorem 24.1. Let $\ell = [\Lambda : \Lambda']$ be the index of Λ' in Λ. Then $\ell \geq 1$ is an integer, and we know that $\det\left((\mathbf{B}')^\top \mathbf{B}'\right) / \operatorname{Vol}(\Lambda)^2 = \ell^2$. Our objective is to determine whether $\ell^2 = 1$ (i.e., $\Lambda' = \Lambda$) or $\ell^2 \geq 4$ (i.e., $\Lambda' \subsetneq \Lambda$).

By the choice of ε, we can apply Lemma 24.3 to $\tilde{\mathbf{B}}'$, which proves that $\det\left((\tilde{\mathbf{B}}')^\top \tilde{\mathbf{B}}'\right) \in [\frac{7}{8}, \frac{9}{8}] \cdot \det\left((\mathbf{B}')^\top \mathbf{B}'\right)$. Recall that by assumption, $D \in [\frac{3}{4}, \frac{5}{4}] \cdot \operatorname{Vol}(\Lambda)$. From these two inequalities, we obtain that

$$\frac{\det\left((\tilde{\mathbf{B}}')^\top \tilde{\mathbf{B}}'\right)}{D^2} \in [\tfrac{16}{25} \cdot \tfrac{7}{8}, \tfrac{16}{9} \cdot \tfrac{9}{8}] \cdot \frac{\det\left((\mathbf{B}')^\top \mathbf{B}'\right)}{\operatorname{Vol}(\Lambda)^2} = [\tfrac{14}{25}, 2] \cdot \ell^2.$$

If $\Lambda = \Lambda'$, then $\ell = 1$ and we have $\frac{\det\left((\tilde{\mathbf{B}}')^\top \tilde{\mathbf{B}}'\right)}{D^2} \leq 2$. Otherwise, $\ell \geq 2$ and we have $\frac{\det\left((\tilde{\mathbf{B}}')^\top \tilde{\mathbf{B}}'\right)}{D^2} > 2.2$. We can then distinguish the two cases by computing the quantity $\det\left((\tilde{\mathbf{B}}')^\top \tilde{\mathbf{B}}'\right)$ and checking whether is it $\leq 2 \cdot D^2$ or $> 2 \cdot D^2$.

For the running time, we only need to compute the determinant of the rational matrix $(\tilde{\mathbf{B}}')^\top \tilde{\mathbf{B}}'$, and then compare it with the rational number $2 \cdot D^2$. This can be done in polynomial time in the bit-sizes of the $\tilde{\mathbf{B}}'$ and D (see, e.g., [45] for an algorithm that computes the determinant of integer matrices in polynomial time). □

25. Post-processing in the $\mathbb{S}$-unit computation

Theorem 19.1. *Let K be a number field with degree n, and let $G = \{\eta_1, \ldots, \eta_k\} \subseteq \mathcal{O}_{K,\mathbb{S}}^\times$ a finite set of $\mathbb{S}$-units. There exists an algorithm that takes as input the set G*[37] *and computes a system of fundamental $\mathbb{S}$-units for the subgroup $\langle G \rangle \subseteq \mathcal{O}_{K,\mathbb{S}}^\times$ generated by G in compact representation, i.e., a pair $(\mathbf{N}, G)$ with $\mathbf{N} \in \mathbb{Z}^{r \times k}$ such that $\upsilon_j = \prod_{i=1}^k \eta_i^{\mathbf{N}_{ji}}$ for $j \in \{1, \ldots, r\}$ form a system of fundamental $\mathbb{S}$-units for $\langle G \rangle$.*

Additionally, this algorithm decides whether $\langle G \rangle = \mathcal{O}_{K,\mathbb{S}}^\times$ or not, and runs in time polynomial in k, $\log|\Delta_K|$, $|\mathbb{S}|$, $\log\left(\max_j \|\operatorname{Log}_{\mathbb{S}}(\eta_j)\|\right)$ and $\max_j(\operatorname{size}(\eta_j))$.

[37] The elements $\eta_i \in K$ are represented as vectors of rational coordinates in a fixed $\mathcal{O}_K$-basis, as explained in Section 2.6.

Proof. **Part I: Setting up the log-matrix and assembling properties of the log-S-unit lattice.** Define $A = \max_j \|\mathrm{Log}_{\mathbb{S}}(\eta_j)\|$ and $\mu = (1000 \cdot \sqrt{n} \cdot \log(n)^3)^{-1}$. We know from Lemma 15.6 that $\mu \leq \lambda_1(\mathrm{Log}_{\mathbb{S}}(\mathcal{O}_{K,\mathbb{S}}^{\times}))$. Note that this lower bound is automatically also a lower bound for the first minimum of any sublattice of $\mathrm{Log}_{\mathbb{S}}(\mathcal{O}_{K,\mathbb{S}}^{\times})$.

Consider the exact matrix $\mathbf{G}$ where the rows consist of $\mathrm{Log}_{\mathbb{S}}(\eta_j)$ with $j \in \{1, \ldots, k\}$, and compute an approximation $\tilde{\mathbf{G}} \in \mathbb{Q}^{k \times (n+|\mathbb{S}|)}$, with $\|\tilde{\mathbf{G}}_j - \mathbf{G}_j\| \leq \varepsilon$ for each row, where we put

$$\varepsilon = \mu \cdot \underbrace{2^{-8k} \cdot \left(\frac{k \cdot 4^k \cdot A}{\mu}\right)^{-2(k+1)}}_{\substack{\leq C_0^{-1} \text{ as in Theorem 23.5} \\ \text{(Buchmann-Kessler-Pohst)}}} \cdot \underbrace{2^{-6} \cdot (n + |\mathbb{S}|)^{-(k+6)} \cdot 2^{-(2k+1)(k+2)} \cdot A^{-1} \cdot \mu}_{\substack{\text{Determinant} \\ \text{approximation}}}$$

$$\leq \tfrac{1}{4} \cdot \mu \cdot C_0^{-1} \tag{157}$$

Note that from the exact representation of the η_i's, one can compute approximations of the $\mathrm{Log}_{\mathbb{S}}(\eta_i)$'s with arbitrary precision ε in time polynomial in $\mathrm{size}(\eta_i)$, in $\log(\|\mathrm{Log}_{\mathbb{S}}(\eta_i)\|)$ and in $\log(1/\varepsilon)$. From our choice of ε, this is polynomial in k, $\log|\Delta_K|$, $\log\big(\max_j \|\mathrm{Log}_{\mathbb{S}}(\eta_j)\|\big) = \log(A)$, $|\mathbb{S}|$, and $\max_i(\mathrm{size}(\eta_i))$ as expected. Moreover, the bit-size of the resulting matrix $\tilde{\mathbf{G}}$ is also polynomial in these quantities.

The quantity ε is chosen such that the left part of the product in Equation (157) counteracts the loss in precision due to the Buchmann-Kessler-Pohst algorithm (Theorem 23.5), whereas the right part is the minimum precision required to make the determinant approximation work (as in Theorem 24.1).

Part II: Applying the Buchmann-Kessler-Pohst algorithm. We apply Theorem 23.5 to $\tilde{\mathbf{G}}$ with $r_0 = k$, $\mu \leq \lambda_1(\mathrm{Log}_{\mathbb{S}}(\langle G \rangle))$ (as defined above) and we note that, by definition, $\|\mathbf{G}\|_{2,\infty} \leq A$ which implies $\varepsilon < \frac{1}{4} \cdot \mu \cdot C_0^{-1}$ (for C_0 as in Theorem 23.5), so the prerequisites of the theorem are satisfied.

So, the algorithm of Theorem 23.5 outputs an $\mathbf{N}$ such that $\mathbf{B} := \mathbf{N}\mathbf{G}$ is a basis of $\mathrm{Log}_{\mathbb{S}}(\langle G \rangle)$ and $\tilde{\mathbf{B}} := \mathbf{N}\tilde{\mathbf{G}}$ satisfies

$$\|\mathbf{B} - \tilde{\mathbf{B}}\|_{2,\infty} \leq C_0 \cdot \varepsilon \leq 2^{-6} \cdot (n + |\mathbb{S}|)^{-(k+6)} \cdot 2^{-(2k+1)(k+2)} \cdot A^{-1} \cdot \mu^2 \tag{158}$$

Additionally, the rows $\mathbf{b}_j$ of $\mathbf{B}$ satisfy

$$\|\mathbf{b}_j\| \leq (\sqrt{rn} + 2) \cdot 2^{\frac{r-1}{2}} \cdot \lambda_j(\mathrm{Log}_{\mathbb{S}}(\langle G \rangle)), \tag{159}$$

where r is the rank of the lattice $\mathrm{Log}_{\mathbb{S}}(\langle G \rangle)$.

Note that the pair $(\mathbf{N}, G)$ encodes a fundamental system of units of $\mathrm{Log}_{\mathbb{S}}(\langle G \rangle)$ in compact representation.

The Buchmann-Kessler-Pohst algorithm computes the rank r of $\mathrm{Log}_{\mathbb{S}}(\langle G \rangle)$; we proceed only to Part III if this rank equals the rank of $\mathrm{Log}_{\mathbb{S}}(\mathcal{O}_{K,\mathbb{S}}^{\times})$, namely $n_{\mathbb{R}} + n_{\mathbb{C}} + |\mathbb{S}| - 1$. It thus remains to decide whether $\mathrm{Log}_{\mathbb{S}}(\langle G \rangle) = \mathrm{Log}_{\mathbb{S}}(\mathcal{O}_{K,\mathbb{S}}^{\times})$ or not. This is done by approximating determinants.

Part III: Computing the determinant. In order to apply Theorem 24.1 to the matrix $\tilde{\mathbf{B}}$ obtained, we need upper bounds on the quantities $\prod_{j=1}^{r} \frac{\|\mathbf{b}_j\|}{\lambda_j(\mathrm{Log}_{\mathbb{S}}(\langle G \rangle))}$ and $\|\mathbf{B}\|_2$, and a lower bound on the quantity $\lambda_1(\mathrm{Log}_{\mathbb{S}}(\langle G \rangle))$. Since $\mathrm{Log}_{\mathbb{S}}(\langle G \rangle)$ is a sublattice of $\mathrm{Log}_{\mathbb{S}}(\mathcal{O}_{K,\mathbb{S}}^{\times})$, we have already seen that $\lambda_1(\mathrm{Log}_{\mathbb{S}}(\langle G \rangle)) \geq \mu$.

By Equation (159), the output basis of the Buchmann-Kessler-Pohst algorithm satisfies

$$\frac{\|\mathbf{b}_j\|}{\lambda_j(\mathrm{Log}_{\mathbb{S}}(\langle G\rangle))} \leq (\sqrt{rn}+2)\cdot 2^{\frac{r-1}{2}} \leq 2^{k+2},$$

where the last inequality holds because $k \geq r$ and $r = n_{\mathbb{R}} + n_{\mathbb{C}} - 1 \geq n/2 - 1$ (since we proceeded to Part III). So, $\prod_{j=1}^{r} \frac{\|\mathbf{b}_j\|}{\lambda_j(\mathrm{Log}_{\mathbb{S}}(\langle G\rangle))} \leq 2^{k(k+2)}$.

Finally, we observe that

$$\|\mathbf{B}\|_2 \leq r \cdot \max_j \|\mathbf{b}_j\| \leq r \cdot \max_j \left(\frac{\|\mathbf{b}_j\|}{\lambda_j(\mathrm{Log}_{\mathbb{S}}(\langle G\rangle))}\right) \cdot \lambda_r(\mathrm{Log}_{\mathbb{S}}(\langle G\rangle)) \leq r \cdot 2^{k+2} \cdot A,$$

where for the last inequality we used the bound on $\frac{\|\mathbf{b}_j\|}{\lambda_j(\mathrm{Log}_{\mathbb{S}}(\langle G\rangle))}$ computed above, and the fact that $A = \max_j \|\mathrm{Log}_{\mathbb{S}}(\eta_j)\|$ is an upper bound on the last minimum of the lattice generated by the η_j's.

Combining these bounds with Equation (158) (and using the fact that $k \geq r$ and $n + |\mathbb{S}| \geq r$), we obtain, using $(n+|\mathbb{S}|)^{-(k+6)} \cdot r \leq (n+|\mathbb{S}|)^{-1} \cdot (n+|\mathbb{S}|)^{-(k+5)} \cdot r \leq (n+|\mathbb{S}|)^{-1} \cdot r^{-(r+4)}$,

$$\begin{aligned}
&\|\mathbf{B} - \tilde{\mathbf{B}}\|_{2,\infty} \\
&\quad \leq 2^{-6} \cdot (n+|\mathbb{S}|)^{-(k+6)} \cdot 2^{-(2k+1)(k+2)} \cdot A^{-1} \cdot \mu^2 \\
&\quad \leq 2^{-6} \cdot (n+|\mathbb{S}|)^{-(k+6)} \cdot \underbrace{2^{-2k(k+2)}}_{\leq (\prod_j \|\mathbf{b}_j\|/\lambda_j)^{-2}} \cdot r \cdot \underbrace{r^{-1} \cdot 2^{-(k+2)} \cdot A^{-1}}_{\leq \|\mathbf{B}\|_2^{-1}} \cdot \underbrace{\mu^2}_{\leq \lambda_1^2} \\
&\quad \leq 2^{-6} \cdot r^{-(r+4)} \cdot (n+|\mathbb{S}|)^{-1} \cdot \left(\prod_j \frac{\|\mathbf{b}_j\|}{\lambda_j(\mathrm{Log}_{\mathbb{S}}(\langle G\rangle))}\right)^{-2} \cdot \lambda_1(\mathrm{Log}_{\mathbb{S}}(\langle G\rangle))^2 / \|\mathbf{B}\|_2,
\end{aligned}$$

as needed to apply Theorem 24.1 to the matrix $\tilde{\mathbf{B}}$. We conclude that from the knowledge of $\tilde{\mathbf{B}}$ and a rational number $D \in [\frac{3}{4}, \frac{5}{4}] \cdot \mathrm{Vol}(\mathrm{Log}_{\mathbb{S}}(\mathcal{O}_{K,\mathbb{S}}^{\times}))$, we can decide whether $\mathrm{Log}_{\mathbb{S}}(\langle G\rangle) = \mathrm{Log}_{\mathbb{S}}(\mathcal{O}_{K,\mathbb{S}}^{\times})$ or not.

Part IV: The running time. For the running time, we go through each part. For Part I, we have already seen that $\tilde{\mathbf{G}}$ can be computed in time polynomial in the desired quantities, and that its bit-size is also polynomial. For Part II, note that we applied the Buchmann-Kessler-Pohst algorithm as in Theorem 23.5, which takes time polynomial in k, $(n+|\mathbb{S}|)$, $\log(\|\mathbf{G}\|_{2,\infty})$, $\log(1/\mu)$ and the bit-size of $\tilde{\mathbf{G}}$. By definition of μ, this is polynomial in the desired quantities. Moreover, the bit-size of the output matrix $\tilde{\mathbf{B}}$ is also polynomial in those same quantities.

For Part III, we first compute a rational $D \in [\frac{3}{4}, \frac{5}{4}] \cdot \mathrm{Vol}(\mathrm{Log}_{\mathbb{S}}(\mathcal{O}_{K,\mathbb{S}}^{\times}))$, which can be done in time polynomial in $\log|\Delta_K|$ from Proposition 2.9 and Lemma 15.5. We then apply Theorem 24.1 to the matrix $\tilde{\mathbf{B}}$ and the rational number D. This takes time polynomial in the bit-sizes of $\tilde{\mathbf{B}}$ and D, which is polynomial in all the desired quantities. This concludes the proof. □

26. Lattice preliminaries

In this section, we succinctly treat lattice preliminaries required for understanding the next two sections (Sections 27 and 28) about a BKZ-variant by Hanrot, Stehlé and Pujol. This includes the Gram-Schmidt orthogonalization of a basis, the potential of a basis, HKZ-reduced bases, Banaszczyk's transference theorem and the definition of a dually exponentially reduced basis.

26.1. Gram-Schmidt orthogonalization in $\mathbb{Z}$-bases.

Notation 26.1. *For a basis $\mathbf{B} \in \mathbb{R}^{n\times n}$ we denote its Gram-Schmidt orthogonalization by $\mathbf{B}^\star = (\mathbf{b}_1^\star, \ldots, \mathbf{b}_n^\star)$. We denote the dual basis of $\mathbf{B}$ by $\mathbf{D} = (\mathbf{B}^{-1})^\top =: \mathbf{B}^{-\top} = (\mathbf{d}_1, \ldots, \mathbf{d}_n) = (\mathbf{b}_1^\vee, \ldots, \mathbf{b}_n^\vee)$. If $\mathbf{B}$ is a basis of Λ, then $\mathbf{D}$ is a basis of the dual lattice $\Lambda^\vee$ of Λ.*

Notation 26.2. *By a tilde, we generally denote an* approximation. *In this section, often a real-valued $\mathbf{B}$ is approximated by a basis $\tilde{\mathbf{B}}$ with rational coefficients. Likewise, an approximation of the dual basis $\mathbf{D}$ is denoted by $\tilde{\mathbf{D}}$.*

Notation 26.3. *For a basis $\mathbf{B} = (\mathbf{b}_1, \ldots, \mathbf{b}_n) \in \mathbb{R}^{m\times n}$ we denote by $\mathbf{B}_{[j:k]}$ the basis $(\pi_{j-1}(\mathbf{b}_j), \ldots, \pi_{j-1}(\mathbf{b}_k)) \in \mathbb{R}^{m\times(k+1-j)}$, where $\pi_{j-1} = \pi_{(\mathbf{b}_1,\ldots,\mathbf{b}_{j-1})^\perp}$ is the projection orthogonal to the linear subspace generated by $(\mathbf{b}_1, \ldots, \mathbf{b}_{j-1})$. Note that $\mathbf{B}_{[1:k]}$ is just $(\mathbf{b}_1, \ldots, \mathbf{b}_k)$.*

Definition 26.4 (Potential of a basis)**.** *For a basis $\mathbf{B} \in \mathbb{R}^{m\times n}$ we denote the potential by*

$$P(\mathbf{B}) = \prod_{j=1}^{n} \|\mathbf{b}_j^\star\|^{n+1-j} = \prod_{j=1}^{n} \mathrm{Vol}(\mathcal{L}(\mathbf{b}_1, \ldots, \mathbf{b}_j)),$$

where $\mathcal{L}(\mathbf{b}_1, \ldots, \mathbf{b}_j)$ is the lattice generated by $(\mathbf{b}_1, \ldots, \mathbf{b}_j)$.

Lemma 26.5. *Let $\mathbf{B} = (\mathbf{b}_1, \ldots, \mathbf{b}_n) \in \mathbb{Z}^{m\times n}$ be a basis of the lattice Λ, then $\mathrm{Vol}(\mathcal{L}(\mathbf{b}_1, \ldots, \mathbf{b}_{j-1}))^2 \cdot \mathbf{B}_{[j:n]} \in \mathbb{Z}^m$. In particular, $\mathrm{Vol}(\mathcal{L}(\mathbf{b}_1, \ldots, \mathbf{b}_{j-1}))^2 \cdot \mathbf{b}_j^\star \in \mathbb{Z}^m$ and hence $P(\mathbf{B})^2 \cdot \mathbf{B}^\star \in \mathbb{Z}^{m\times n}$.*

Proof. By the determinant formula for orthonormal projection [34, Vol. 1, Chapter IX, §4, Equation (21)] we have that, for $D_j := \det(\mathbf{B}_{[1:j-1]}^\top \mathbf{B}_{[1:j-1]})$, $D_j \mathbf{B}_{[j:n]} \in \mathbb{Z}^m$. In particular $D_j \mathbf{b}_j^\star \in \mathbb{Z}^m$. Note that $D_j = \mathrm{Vol}(\mathcal{L}(\mathbf{b}_1, \ldots, \mathbf{b}_{j-1}))^2$ by definition.

As a result, we certainly deduce that for $P(\mathbf{B})^2 = \left(\prod_{j=2}^n D_j\right) \cdot \mathrm{Vol}(\mathcal{L}(\mathbf{B}))^2$ we have $P(\mathbf{B})^2 \cdot \mathbf{b}_k^\star \in \mathbb{Z}^m$ for any $k \in \{1, \ldots, n\}$ (here, we use that $D_j \in \mathbb{Z}$ since $\mathbf{B} \in \mathbb{Z}^{m\times n}$). That is, $P(\mathbf{B})^2 \cdot \mathbf{B}^\star \in \mathbb{Z}^{m\times n}$. □

Lemma 26.6. *Let $\mathbf{B} = (\mathbf{b}_1, \ldots, \mathbf{b}_n) \in \mathbb{Z}^{m\times n}$ be a basis of the lattice Λ, then $\|\mathbf{b}_j^\star\| \le \mathrm{Vol}(\mathcal{L}(\mathbf{b}_1, \ldots, \mathbf{b}_j)) \le P(\mathbf{B})$.*

Proof. We know that $\mathrm{Vol}(\mathcal{L}(\mathbf{b}_1, \ldots, \mathbf{b}_j)) = \mathrm{Vol}(\mathcal{L}(\mathbf{b}_1, \ldots, \mathbf{b}_{j-1})) \cdot \|\mathbf{b}_j^\star\|$. Moreover, since $\mathbf{B}$ is integral, then $\mathrm{Vol}(\mathcal{L}(\mathbf{b}_1, \ldots, \mathbf{b}_{j-1}))^2 \in \mathbb{Z}$, and so it is ≥ 1 (because $\mathbf{B}$ is a basis, so its vectors are linearly independent). The inequality $\|\mathbf{b}_j^\star\| \le \mathrm{Vol}(\mathcal{L}(\mathbf{b}_1, \ldots, \mathbf{b}_j))$ follows. □

Definition 26.7. *We call a basis $\mathbf{B} = (\mathbf{b}_1, \ldots, \mathbf{b}_n) \in \mathbb{R}^{m\times n}$ size-reduced if for all $0 < i < j \le n$*

$$|\langle \mathbf{b}_i^\star, \mathbf{b}_j \rangle| \le \tfrac{1}{2}\|\mathbf{b}_i^\star\|^2.$$

Lemma 26.8. *Let $\mathbf{B} \in \mathbb{Z}^{m\times n}$ be a size reduced basis of a lattice Λ. Then, for all $j \in \{1, \ldots, n\}$,*

$$\|\mathbf{b}_j\| \le \tfrac{\sqrt{n}}{2} \max_i \|\mathbf{b}_i^\star\|.$$

Proof. Size reduced means that any basis vector $\mathbf{b}_j$ can be written as $\sum_{i=1}^n c_{ij}\mathbf{b}_i^\star$ for $|c_{ij}| \leq 1/2$. Hence, by the Pythagorean theorem, we have

$$\|\mathbf{b}_j\|^2 = \sum_{i=1}^n c_{ij}^2 \|\mathbf{b}_i^\star\|^2 \leq \tfrac{n}{4} \max_j \|\mathbf{b}_i^\star\|^2,$$

from which the claim follows. □

Corollary 26.9. *Let $\mathbf{B} \in \mathbb{Z}^{m\times n}$ be a size reduced basis of a lattice Λ. Then each coefficient in $\mathbf{B}$, $\mathbf{B}^\star$ and $\mathbf{B}_{[j:k]}$ (for $k > j \in \mathbb{N}$) can be represented in $5\log_2(n \cdot P(\mathbf{B})) + 3$ bits.*

Proof. Each vector in $\mathbf{B}$ and $\mathbf{B}^\star$ is bounded in Euclidean norm by $\sqrt{n}\cdot P(\mathbf{B})/2$, which is also an upper bound on the absolute value of each coefficient. Moreover, $\mathbf{B}$ is integral and the denominators occurring in $\mathbf{B}^\star$ can be maximally $P(\mathbf{B})^2$. Every coefficient of $\mathbf{B}$ and $\mathbf{B}^\star$ is then a rational number which can be represented by a fraction a/b with a, b integers, $|a| \leq \sqrt{n}\cdot P(\mathbf{B})^3/2$ and $0 < b \leq P(\mathbf{B})^2$. They can be represented by respectively by $\lceil \log_2(\sqrt{n}\cdot P(\mathbf{B})^3 + 1)\rceil$ and $\lceil \log_2(P(\mathbf{B})^2)\rceil$ bits.

Precisely the same reasoning can be used for $\mathbf{B}_{[i:j]}$, since $\mathrm{Vol}(\mathcal{L}(\mathbf{b}_1,\ldots,\mathbf{b}_{j-1}))^2 \cdot \mathbf{B}_{[j:n]} \in \mathbb{Z}^m$ by Lemma 26.5 (and $\mathbf{B}_{[j:k]}$ consists of just the first $k-j+1$ vectors of $\mathbf{B}_{[j:n]}$). □

26.2. HKZ reduction.

Definition 26.10 (HKZ-reduced basis)**.** *A basis $\mathbf{B} \in \mathbb{R}^{m\times n}$ of a lattice Λ is called HKZ-reduced if it is size reduced and if*

$$\|\mathbf{b}_j^\star\| = \lambda_1(\mathbf{B}_{[j:n]}) \text{ for all } j \in \{1,\ldots,n\},$$

i.e., if the Gram-Schmidt vector $\mathbf{b}_j^\star$ attains the minimum of the projected lattice $\pi_{(\mathbf{b}_1,\ldots,\mathbf{b}_{j-1})^\perp}(\Lambda)$ for every $j \in \{1,\ldots,n\}$.

Lemma 26.11. *Let $\mathbf{B}$ be a HKZ-reduced basis of Λ, then*

$$P(\mathbf{B}) \leq n^{2n^2} \min_{\mathbf{C}} P(\mathbf{C}),$$

where the minimum over the potential $P(\cdot)$ is over all bases $\mathbf{C}$ of Λ.

Proof. The HKZ reduced basis $\mathbf{B}$ satisfies $\frac{4}{i+3}\lambda_i(\Lambda)^2 \leq \|\mathbf{b}_i\|^2 \leq \frac{i+3}{4}\cdot\lambda_i(\Lambda)^2$ [48], hence (using $\frac{i+3}{4} \leq i$ for $i \geq 1$),

$$\begin{aligned} P(\mathbf{B}) = \prod_{j=1}^n (\|\mathbf{b}_1^\star\|\cdots\|\mathbf{b}_j^\star\|) \leq \prod_{j=1}^n (\|\mathbf{b}_1\|\cdots\|\mathbf{b}_j\|) \leq \prod_{j=1}^n j^j(\lambda_1(\Lambda)\cdots\lambda_j(\Lambda)) \\ \leq \prod_{j=1}^n j^j\cdot j^j \min_{\Lambda_j}\mathrm{Vol}(\Lambda_j) \leq n^{2n^2}\prod_{j=1}^n \min_{\Lambda_j}\mathrm{Vol}(\Lambda_j) \leq n^{2n^2}\min_{\mathbf{C}} P(\mathbf{C}). \end{aligned} \tag{160}$$

where the minimum $\min_{\Lambda_j}$ is over all j-dimensional sublattices $\Lambda_j \subseteq \Lambda$ and where the minimum $\min_{\mathbf{C}}$ is over all bases of Λ.

The third inequality of Equation (160) might require some explanation. Any i-dimensional sublattice $\Lambda_i \subseteq \Lambda$ satisfies $\lambda_j(\Lambda) \leq \lambda_j(\Lambda_i)$ for all $j \in \{1,\ldots,i\}$. Therefore, by Minkowski's second inequality,

$$\lambda_1(\Lambda)\cdots\lambda_i(\Lambda) \leq \lambda_1(\Lambda_i)\cdots\lambda_i(\Lambda_i) \leq i^i\,\mathrm{Vol}(\Lambda_i).$$

□

This lemma has as a consequence that applying the HKZ algorithm can only increase the potential of a basis by a factor n^{2n^2}.

Corollary 26.12. *Let $\mathbf{B} \in \mathbb{R}^{m\times n}$ be a basis of the n-dimensional lattice Λ and let $\mathbf{B}' = \mathrm{HKZ}(\mathbf{B})$ be a HKZ-reduced basis of Λ. Then*

$$P(\mathbf{B}') \leq n^{2n^2} P(\mathbf{B}).$$

Proof. We have $P(\mathbf{B}') \leq n^{2n^2} \cdot \min_{\mathbf{C}} P(\mathbf{C}) \leq n^{2n^2} P(\mathbf{B})$. □

The following lemma is for when a *block* of $\mathbf{B}$ is being HKZ-reduced, which happens in the BKZ-algorithm.

Corollary 26.13. *Let $\mathbf{B} \in \mathbb{R}^{m\times n}$ be a basis of a lattice Λ, let $\mathsf{b} \in \{2, \dots, n\}$, and let $\mathbf{C} = \mathbf{B}_{[k:k+\mathsf{b}-1]}$ be a projected sub-block of $\mathbf{B}$. Suppose $\mathbf{U} \in GL_{\mathsf{b}}(\mathbb{Z})$ is such that $\mathbf{C}' = \mathbf{CU}$ is HKZ-reduced and write $\mathbf{B}' = \mathbf{B}\bar{\mathbf{U}}$, where $\bar{\mathbf{U}}$ acts as $\mathbf{U}$ on the basis elements $(\mathbf{b}_k, \dots, \mathbf{b}_{k+\mathsf{b}-1})$ and leaves the rest intact. Then*

$$P(\mathbf{B}') \leq \mathsf{b}^{2\mathsf{b}^2} P(\mathbf{B}),$$

Proof. The action of $\mathbf{B}' = \mathbf{B}\bar{\mathbf{U}}$ only changes the vectors $(\mathbf{b}_k, \dots, \mathbf{b}_{k+\mathsf{b}-1})$, but does *not* change the *space* spanned by $(\mathbf{b}_k, \dots, \mathbf{b}_{k+\mathsf{b}-1})$. Hence, the Gram-Schmidt vectors $\mathbf{b}_j^\star$ for $j \notin \{k, \dots, k+\mathsf{b}-1\}$ also remain intact, as well as the volume $\mathrm{Vol}(\mathcal{L}(\mathbf{b}_k, \dots, \mathbf{b}_{k+\mathsf{b}-1})) = \prod_{j=k}^{k+\mathsf{b}-1} \|\mathbf{b}_j^\star\| = \prod_{j=k}^{k+\mathsf{b}-1} \|(\mathbf{b}_j')^\star\|$. So,

$$\begin{aligned}\frac{P(\mathbf{B}')}{P(\mathbf{B})} &= \frac{\prod_{j=1}^{n} \|(\mathbf{b}_j')^\star\|^{n+1-j}}{\prod_{j=1}^{n} \|\mathbf{b}_j^\star\|^{n+1-j}} = \frac{\prod_{j=k}^{k+\mathsf{b}-1} \|(\mathbf{b}_j')^\star\|^{n+1-j}}{\prod_{j=k}^{k+\mathsf{b}-1} \|\mathbf{b}_j^\star\|^{n+1-j}} = \frac{\prod_{j=0}^{\mathsf{b}-1} \|(\mathbf{b}_j')^\star\|^{\mathsf{b}-j}}{\prod_{j=0}^{\mathsf{b}-1} \|\mathbf{b}_j^\star\|^{\mathsf{b}-j}} \\ &= \frac{P(\mathbf{C}')}{P(\mathbf{C})} \leq \mathsf{b}^{2\mathsf{b}^2}.\end{aligned}$$

where the last inequality follows from Corollary 26.12 and where the third equality follows from dividing by $\prod_{j=k}^{k+\mathsf{b}-1} \|\mathbf{b}_j^\star\|^{(n+1)-(\mathsf{b}+k)}$ so that

$$\prod_{j=k}^{k+\mathsf{b}-1} \|\mathbf{b}_j^\star\|^{n+1-j} = \prod_{j=k}^{k+\mathsf{b}-1} \|\mathbf{b}_j^\star\|^{(\mathsf{b}+k)-j} = \prod_{j=0}^{\mathsf{b}-1} \|\mathbf{b}_j^\star\|^{j},$$

and similarly for $(\mathbf{b}_j')^\star$, where we use that $\prod_{j=k}^{k+\mathsf{b}-1} \|\mathbf{b}_j^\star\| = \prod_{j=k}^{k+\mathsf{b}-1} \|(\mathbf{b}_j')^\star\|$. □

26.3. Reduction of dual bases.

Definition 26.14 (A dually exponentially reduced basis)**.** *A square basis $\mathbf{B} \in \mathbb{R}^{n\times n}$ of a lattice Λ is T-dually exponentially reduced (for $T \geq 1$) if the dual basis $\mathbf{D} = \mathbf{B}^{-\top}$ satisfies $\|\mathbf{d}_j\| \leq 2^{Tn}\lambda_j(\Lambda^\vee)$, where $\Lambda^\vee$ denotes the dual lattice of Λ.*

26.4. **Transference theorems.** Banaszczyk proved the following transference theorem, relating the successive minima of a lattice Λ with the ones of its dual lattice $\Lambda^\vee$.

Theorem 26.15 ([6, Theorem 2.1])**.** *Let Λ be an arbitrary lattice of rank n in $\mathbb{R}^m$, for some $m \geq n \geq 1$. Then, for all $1 \leq i \leq n$ it holds that*

$$1 \leq \lambda_i(\Lambda) \cdot \lambda_{n-i+1}(\Lambda^\vee) \leq n.$$

Proof. The upper bound is exactly [6, Theorem 2.1]. The lower bound comes from the following standard argument. Let $v_1, \dots, v_i$ be i linearly independent vectors of Λ with euclidean norm $\leq \lambda_i(\Lambda)$, and $w_1, \dots, w_{n-i+1}$ be $(n-i+1)$ linearly independent vectors of $\Lambda^\vee$ with euclidean norm $\leq \lambda_{n-i+1}(\Lambda^\vee)$. Let V be the vector space spanned by the $(v_j)_j$ and W be the vector space spanned by the $(w_j)_j$. Both spaces live in $\mathrm{span}(\Lambda)$, which has dimension n. Moreover, $\dim(V) + \dim(W) = n+1$. Hence, V and W cannot be orthogonal, which means that there should exist v_k and w_ℓ such that $\langle v_k, w_\ell \rangle \neq 0$. By definition of the dual, this implies that $|\langle v_k, w_\ell \rangle| \geq 1$ (because the inner product must be an integer). And so, $1 \leq |\langle v_k, w_\ell \rangle| \leq \|v_k\| \cdot \|w_\ell\| \leq \lambda_i(\Lambda) \cdot \lambda_{n-i+1}(\Lambda^\vee)$. □

27. The BKZ algorithm on integer bases

27.1. **Introduction.** In this section, we show that the variant of BKZ from Hanrot, Pujol and Stehlé [39, Algorithm 2] (which we call BKZ' in the rest of this section) does not cause coefficient explosion in integer basis matrices. Hence, BKZ' is suitable for integer lattice bases and one can obtain a bound on the bit-complexity of the algorithm, and not only the number of tours needed. This was already discussed in [39, Section 3, Cost of BKZ'], where the authors explained the big lines of the reasoning. In this section, we make this discussion fully formal.

Concretely, applying the b-BKZ' algorithm of Hanrot, Pujol and Stehlé on an integer basis $\mathbf{B}$ of a lattice Λ yields a new basis $\mathbf{C} = (\mathbf{c}_1, \dots, \mathbf{c}_n)$ of Λ satisfying $\|\mathbf{c}_1\| \leq 2 \cdot \mathsf{b}^{\frac{n-1}{2(\mathsf{b}-1)} + \frac{3}{2}} \cdot \mathrm{Vol}(\Lambda)^{1/n}$. Furthermore, this algorithm runs in time $\mathsf{poly}(n, \log \max_j \|\mathbf{b}_j\|) \cdot \mathsf{b}^{\mathsf{b}}$.

Approach. In [39], the authors give an upper bound on the *number of tours* needed for the BKZ' algorithm, in order to have a provable upper bound on the short vector output by the algorithm. To obtain a total and provable bit-complexity for the time of the b-BKZ' variant for integer matrices, we need to show that no coefficient explosion occurs and that the b-HKZ algorithm on the sub-blocks runs in time about b^{b}.

To show that no coefficient explosion occurs, we rely on standard techniques using the potential $P(\mathbf{B})$ of a basis. Namely, it is a fact that the bit sizes of the coefficients occurring in a size-reduced basis $\mathbf{B}$ and its Gram-Schmidt basis $\mathbf{B}^\star$ are bounded by $O(\log_2(nP(\mathbf{B})))$ (see Corollary 26.9). Hence, it is enough to sufficiently bound the potential.

In the BKZ' algorithm of Hanrot, Pujol and Stehlé, only size-reduction and HKZ-reduction in dimension b occur (see Algorithm 7). As size-reduction does not change the potential, only the influence of HKZ-reduction on the potential needs to be examined. It can be shown that b-HKZ-reduction can only increase the potential of a basis by a factor $\mathsf{b}^{2\mathsf{b}^2}$ (see Corollary 26.12). Hence, all coefficients remain sufficiently bounded, whenever the number of HKZ-reductions is polynomially bounded. But the latter is true by the fact that the number of 'BKZ tours' (essentially n times a b-HKZ-reduction) is bounded; this is what Hanrot, Pujol and Stehlé show in their work [39, Theorem 1].

For the HKZ-reduction algorithm in dimension b, that is used as a subroutine in the BKZ-algorithm, we use the provable Kannan-algorithm [40] on integer matrices with a run time of about b^{b} (disregarding the size of the basis matrix). Note that the projected sub-blocks of the matrix are rational, but can be scaled up to be integral. This does not significantly increase the bit-size, because the denominators can be shown to be bounded by the potential.

27.2. **The BKZ' algorithm of Hanrot, Pujol and Stehlé.** We restate the algorithm of Hanrot, Pujol and Stehlé [39, Algorithm 2] and their result on the upper bound on the number of tours required to obtain a sufficiently short non-zero vector of the input lattice.

Algorithm 7 The BKZ' algorithm of Hanrot, Pujol and Stehlé, for integer bases

```
Require:
    • A basis $\mathbf{B} = (\mathbf{b}_1, \dots, \mathbf{b}_n) \in \mathbb{Z}^{m \times n}$
    • A block size $\mathsf{b} \in \{1, \dots, n\}$.
Ensure: A basis of $\mathcal{L}(\mathbf{B})$.
repeat
    for $k$ from 1 to $n - \mathsf{b} + 1$ do
        Modify $(\mathbf{b}_i)_{k \le i < k+\mathsf{b}}$ so that $(\pi_{k-1}(\mathbf{b}_i))_{k \le i < k+\mathsf{b}}$ is HKZ-reduced;
        Size-reduce $(\mathbf{b}_1, \dots, \mathbf{b}_n)$.
    end for
until no changes occur or termination is requested.
return $(\mathbf{b}_1, \dots, \mathbf{b}_n)$.
```

Theorem 27.1 (Theorem 1 of [39])**.** *There exists an absolute constant $C > 0$ such that the following holds for all n and b. Let $\mathbf{B} = (\mathbf{b}_1, \dots, \mathbf{b}_n)$ be a basis of a lattice Λ, given as input to the modified BKZ algorithm (Algorithm 7) with block-size b. If terminated after*

$$\tau_{\mathrm{BKZ}} = C\frac{n^3}{\mathsf{b}^2}(\log n + \log\log \max_i \frac{\|\mathbf{b}_i\|}{\mathrm{Vol}(\Lambda)^{1/n}})$$

calls to an HKZ-reduction in dimension b, the output $\mathbf{C} = (\mathbf{c}_1, \dots, \mathbf{c}_n)$ is a basis of Λ that satisfies

$$\|\mathbf{c}_1\| \le 2\mathsf{b}^{\frac{n-1}{2(\mathsf{b}-1)}+\frac{3}{2}} \cdot \mathrm{Vol}(\Lambda)^{1/n}.$$

Remark 27.2. *In the original theorem statement* [39, Theorem 1]*, the last line reads: "If $\Lambda \subseteq \mathbb{Q}^n$, then the overall cost*[38] *is $\le \mathit{poly}(n, \mathrm{size}(\mathbf{B})) \cdot \mathrm{Cost}_{\mathrm{HKZ}}(\mathsf{b})$."*

It is precisely this statement in the last line that we prove in this section. This, because the authors [39] *only gave a sketch of how to show this (*[39, Section 3, Cost of BKZ']*). In this section we will follow their sketch and prove this statement rigorously.*

27.3. **Applying BKZ' to integral bases.**

Proposition 27.3. *Let $\mathbf{B} \in \mathbb{Z}^{m \times n}$ be a basis of a lattice Λ, with $m \ge n \ge 2$. Let $\mathsf{b} \in \{2, \cdots, n\}$ be a block-size parameter. Then the BKZ' algorithm of Hanrot, Pujol and Stehlé on input $\mathbf{B}$ and b has a bit-complexity*

$$\mathit{poly}(\mathrm{size}(\mathbf{B}), m) \cdot \mathsf{b}^{\mathsf{b}}$$

[38] Here, $\mathrm{size}(\mathbf{B}) = \sum_{ij} \mathrm{size}(\mathbf{B}_{ij})$ and $\mathrm{size}(a/b) = \log|a| + \log|b|$ for reduced fractions a/b. The number $\mathrm{Cost}_{\mathrm{HKZ}}(\mathsf{b})$ is the cost of applying HKZ in dimension b.

and outputs a basis $\mathbf{C} = (\mathbf{c}_1, \dots, \mathbf{c}_n)$ *satisfying*

$$\|\mathbf{c}_1\| \leq 2 \cdot \mathsf{b}^{\frac{n-1}{2(\mathsf{b}-1)}+\frac{3}{2}} \cdot \mathrm{Vol}(\Lambda)^{1/n} \leq 2 \cdot \mathsf{b}^{2n/\mathsf{b}} \cdot \lambda_n(\Lambda). \tag{161}$$

Proof. We instantiate Theorem 27.1 with the integral basis $\mathbf{B}$, yielding the shortness bound[39] in Equation (161) and an upper bound on the number of HKZ-reductions τ_{BKZ} in dimension b. Note that τ_{BKZ} is $\mathsf{poly}(n, \mathrm{size}(\mathbf{B}))$, according to Theorem 27.1.

For the b-HKZ reduction algorithm we use the provable Kannan-algorithm [40, Theorem 2]: on input an integral basis $\mathbf{M}$ of a rank-b lattice $L \subseteq \mathbb{Z}^m$, this algorithm runs in time $\mathsf{poly}(\mathrm{size}(\mathbf{M}), m) \cdot \mathsf{b}^{\mathsf{b}/(2e)+o(\mathsf{b})} = \mathsf{poly}(\mathrm{size}(\mathbf{M}), m) \cdot \mathsf{b}^{\mathsf{b}}$ and outputs an HKZ-reduced basis of L. We will apply this algorithm to the projected bases $\mathbf{B}_{[k:k+\mathsf{b}-1]}$ occurring in the BKZ' algorithm. Even though these bases are rational, this is still possible just by multiplying out the denominators of the coefficients (and revert that process after the lattice reduction). For this to be feasible, we will of course need to show that these coefficients are sufficiently small in size (i.e., have sufficiently small numerator and denominator).

Concluding, if we can show that all sizes of the coefficients occurring in the sub-blocks of all intermediately computed matrices in Algorithm 7 remain sufficiently bounded, say $\leq \mathsf{poly}(\mathrm{size}(\mathbf{B}).m)$, we can indeed conclude that Algorithm 7 has bit-complexity $\mathsf{poly}(\mathrm{size}(\mathbf{B}), m)$.

So, indeed, it remains to show that all (rational) coefficients of the occurring bases, their Gram-Schmidt bases and the sub-blocks $\mathbf{B}_{[j:j+\mathsf{b}-1]}$ remain bounded in bit-size by $\mathsf{poly}(\mathrm{size}(\mathbf{B}), m)$. We distinguish some cases

- The very first time that a HKZ-reduction is applied is just on the basis $\mathbf{B}_{[1:\mathsf{b}]}$, whose coefficient sizes are clearly bounded by $\mathsf{poly}(\log P(\mathbf{B}))$.
- *After* this very first time, we can, by line 4 always assume that the basis at hand, which we call $\mathbf{B}^{(0)}$ for now, is size-reduced right before line 3. Hence, the coefficients of $(\pi_{k-1}((\mathbf{b}_i^{(0)}))_{k \leq i < k+\mathsf{b}}$ are bounded in size by $\mathsf{poly}(\log P(\mathbf{B}^{(0)}))$, by Corollary 26.9. Hence Kannan's HKZ algorithm can reasonably applied in time $\mathsf{poly}(\log(P(\mathbf{B}^{(0)})), m) \cdot \mathsf{b}^{\mathsf{b}}$.
- Right after line 3 but *before* line 4, the new matrix $\mathbf{B}^{(1)} = \mathbf{B}^{(0)}\mathbf{U}$ (where $\mathbf{U}$ only changes $(\mathbf{b}_i^{(0)})_{k \leq i < k+\mathsf{b}})$ is not size-reduced anymore. But we do know, by Corollary 26.12, that $P(\mathbf{B}^{(1)}) \leq \mathsf{b}^{2\mathsf{b}^2} P(\mathbf{B}^{(0)})$. Since $(\pi_{k-1}(\mathbf{b}_i^{(1)}))_{k \leq i < k+\mathsf{b}}$ is HKZ-reduced (and thus size-reduced by definition), we know that the coefficients of $\mathbf{B}^{(1)}_{[k:k+\mathsf{b}-1]}$ are bounded by $\mathsf{poly}(\log(P(\mathbf{B}^{(1)})), m)$, by Corollary 26.9. Since for $i \notin \{k, \dots, k+\mathsf{b}-1\}$, $\mathbf{b}_i^{(0)}$ are unaffected by $\mathbf{U}$, we know that these still satisfy $\mathrm{size}(\mathbf{b}_i^{(0)}) \leq \mathsf{poly}(\log(P(\mathbf{B}^{(0)})), m)$. Hence, we can certainly find a lift (by using Babai's nearest plane algorithm) of $\mathbf{B}^{(1)}_{[k:k+\mathsf{b}-1]}$ to $\mathbf{B}^{(1)}$ for which the coefficient sizes of $(\mathbf{b}_i^{(1)}))_{k \leq i < k+\mathsf{b}}$ are bounded by
$$\mathsf{poly}(\log(P(\mathbf{B}^{(0)})), \log(P(\mathbf{B}^{(1)})), m) = \mathsf{poly}(\log(P(\mathbf{B}^{(0)})) + \mathsf{b}, m).$$
(Note that this lift does *not* affect the potential of $\mathbf{B}^{(1)}$.

Note that in the previous reasoning we see that in each HKZ-reduction, the potential of the basis the algorithm is working on is maximally increased by a factor $\mathsf{b}^{2\mathsf{b}^2}$.

[39] The additional shortness bound in terms of $\lambda_n(\Lambda)$ comes from the fact that $\lambda_n(\Lambda)^n \geq \prod_{j=1}^n \lambda_j(\Lambda) \geq \mathrm{Vol}(\Lambda)$. For the simplification into $\mathsf{b}^{n/\mathsf{b}}$, one can observe that $(n-1)b + 3(\mathsf{b}-1)b \leq 4n(\mathsf{b}-1)$ for all b satisfying $2 \leq \mathsf{b} \leq n$. Dividing by $2(\mathsf{b}-1)\mathsf{b}$ yields the simplification.

But we know that the maximum number of HKZ reductions in the entire algorithm is at most $\tau_{\text{BKZ}} \cdot n = \mathsf{poly}(\text{size}(\mathbf{B}), m)$ (where τ_{BKZ} is the number of tours). Hence, the bases occurring in the algorithm can maximally have log potential of

$$\log(\mathbf{B}_{\text{occurring}}) \leq \log(\mathsf{b}^{2\mathsf{b}^2}) \cdot \tau_{\text{BKZ}} \cdot n + \log(P(\mathbf{B})) \leq \mathsf{poly}(\text{size}(\mathbf{B}), m).$$

Hence all occurring bases $\mathbf{B}_{\text{occurring}}$ have a logarithmic potential bounded by $\mathsf{poly}(\text{size}(\mathbf{B}), m)$, which has as an immediate consequence that all bases, Gram-Schmidt bases and projected bases have coefficient sizes bounded by $\mathsf{poly}(\text{size}(\mathbf{B}), m)$, which finishes the proof. □

Corollary 27.4. *Let $\mathbf{B} \in \mathbb{Z}^{m \times n}$ be a basis of a lattice Λ, with $m \geq n \geq 2$ and let $\mathsf{b} \in \{2, \cdots, n\}$. Then there exists an algorithm that takes as input $\mathbf{B}$ and b, has bit-complexity*

$$\mathsf{poly}(m, \text{size}(\mathbf{B})) \cdot \mathsf{b}^{\mathsf{b}}$$

and outputs a basis $\mathbf{C} = (\mathbf{c}_1, \dots, \mathbf{c}_n)$ of Λ satisfying

$$\|\mathbf{c}_j\| \leq n \cdot \mathsf{b}^{2n/\mathsf{b}} \cdot \lambda_n(\Lambda) \quad \text{for all } j \in \{1, \dots, n\}. \tag{162}$$

Proof. We apply the provable BKZ' algorithm of Hanrot, Pujol and Stehlé as in Proposition 27.3, recursively to

$$\mathbf{B},\ \det(\mathbf{B}_{[1:1]})^2 \cdot \mathbf{B}_{2:n},\ \dots, \det(\mathbf{B}_{[1:j]})^2 \cdot \mathbf{B}_{j+1:n}\ , \dots,\ \det(\mathbf{B}_{[1:n-\mathsf{b}]})^2 \cdot \mathbf{B}_{n-\mathsf{b}+1:n}.$$

Here, we keep using the notation $\mathbf{B}$ after an application of BKZ'. After these applications, we backwards recursively lift this sequence into a size-reduced basis of $\mathcal{L}(\mathbf{B})$.

More precisely we apply the following steps:

- Apply BKZ' to $\mathbf{B}$, call the result $\mathbf{B}$.
- Construct the basis $\det(\mathbf{B}_{[1:1]})\mathbf{B}_{[2:n]}$ (integral, by Lemma 26.5) and apply BKZ' to it, call the result again $\det(\mathbf{B}_{[1:1]})\mathbf{B}_{[2:n]}$.
- ...
- Construct the basis $\det(\mathbf{B}_{[1:j]})\mathbf{B}_{[j+1:n]}$ (integral) and apply BKZ' to it, call the result again $\det(\mathbf{B}_{[1:j]})\mathbf{B}_{[j+1:n]}$.
- ...
- Construct the basis $\det(\mathbf{B}_{[1:n-\mathsf{b}]})^2 \cdot \mathbf{B}_{n-\mathsf{b}+1:n}$ (integral) and apply BKZ' to it, call the result again $\det(\mathbf{B}_{[1:j]})\mathbf{B}_{[j+1:n]}$.
- Going backwards, add $c_k \mathbf{b}^\star_{n-\mathsf{b}}$ with $c_k \in [-1/2, 1/2)$ to each vector in $\mathbf{B}_{n-\mathsf{b}+1:n}$ to reconstruct a short basis $\mathbf{B}_{n-\mathsf{b}:n}$.
- ...
- Add $c_k \mathbf{b}^\star_j$ with $c_k \in [-1/2, 1/2)$ to each vector in $\mathbf{B}_{j+1:n}$ to reconstruct a short basis $\mathbf{B}_{j:n}$.
- ...
- Add $c_k \mathbf{b}^\star_j$ with $c_k \in [-1/2, 1/2)$ to each vector in $\mathbf{B}_{2:n}$ to reconstruct a short basis $\mathbf{B}$.

Since a BKZ'-reduction as in Proposition 27.3 consists of τ_{BKZ} HKZ-reductions, each BKZ' reduction can maximally increase the potential by a factor $\mathsf{b}^{2\mathsf{b}^2 \cdot \tau_{\text{BKZ}}}$ (see Corollary 26.12). Hence the $n - \mathsf{b}$ BKZ'-reductions in above procedure can at most result in intermediate potentials of size

$$\mathsf{b}^{2\mathsf{b}^2 \cdot \tau_{\text{BKZ}} \cdot n} \cdot P(\mathbf{B}_{\text{init}}),$$

where $\mathbf{B}_{\text{init}}$ is the starting basis. Since $\det(\mathbf{B}_{[1:k]})$ divide $P(\mathbf{B})$ (the potential of the same basis), and since $\mathbf{B}_{[k+1:n]}$ has size bounded in $O(\log(n \cdot P(\mathbf{B})))$ (see Corollary 26.9), we can immediately deduce that in above procedure, all bit sizes remain within (for the bound on τ_{BKZ} see Theorem 27.1)

$$\log\left(n\mathsf{b}^{2\mathsf{b}^2\cdot\tau_{\text{BKZ}}\cdot n} \cdot P(\mathbf{B}_{\text{init}})\right) = \mathsf{poly}(m, \mathsf{size}(\mathbf{B}_{\text{init}})).$$

In the resulting basis, we have, by construction,

$$\|\mathbf{b}_j^\star\| \leq 2 \cdot \mathsf{b}^{2n/\mathsf{b}} \cdot \lambda_n(\Lambda) \text{ for all } j \in \{1, \ldots, n\},$$

where we use that $\lambda_{n-j}(\mathcal{L}(\mathbf{B}_{j+1:n})) \leq \lambda_n(\Lambda)$ and $\mathsf{b}^{2(n-j)/\mathsf{b}} \leq \mathsf{b}^{2n/\mathsf{b}}$. For the last b vectors, we have this shortness guarantee from the b-HKZ reduction (see [48] or the proof of Lemma 26.11).

Hence, by the size-reduced-like way the basis $\mathbf{B}$ is constructed, we have $\mathbf{b}_j = \mathbf{b}_j^\star + \sum_{i<j} c_i \mathbf{b}_i^\star$ with $c_i \in [-1/2, 1/2)$, and we obtain

$$\|\mathbf{b}_j\| \leq 2n \cdot \mathsf{b}^{2n/\mathsf{b}} \cdot \lambda_n(\Lambda) \text{ for all } j \in \{1, \ldots, n\},$$

as required. □

28. The BKZ algorithm on approximate bases

28.1. **Introduction.** In this section, we will show that there exists a provable algorithm that runs essentially in time $O(\mathsf{b}^{\mathsf{b}})$ that computes a basis of an ideal lattice $x\mathfrak{a}$ whose Gram-Schmidt lengths are at most $\mathsf{b}^{O(n/\mathsf{b})}\lambda_n(x\mathfrak{a})$, which is sufficient for the needs of this paper (see Sections 8.3 and 8.5). We assume throughout this entire section that an LLL-reduced basis of $\mathcal{O}_K$ is given; in the end result, Lemma 8.5, this assumption is again explicitly mentioned.

In all this section, the ideal lattices $x\mathfrak{a}$ we considered will be represented exactly. To do so, we will restrict ourselves to elements $x \in K_{\mathbb{R}}^\times$ that have all their complex coordinates of the form $a + ib$, with $a, b \in \mathbb{Q}$. In other words, we consider the intersection of $K_{\mathbb{R}}$ with the subset $(\mathbb{Q} + i\mathbb{Q})^n$ of $\mathbb{C}^n$. The ideal $\mathfrak{a}$ is a fractional ideal, and will be represented by a $\mathbb{Z}$-basis, consisting of n elements $(\alpha_1, \ldots, \alpha_n)$ of K (which can be exactly represented in a fixed basis of $\mathcal{O}_K$).

Even though we have a way to represent the ideal lattice $x\mathfrak{a}$ exactly, we can only compute an *approximate basis* of the corresponding lattice, due to the (irrational) Minkowski embedding of $x\mathfrak{a}$ into $K_{\mathbb{R}}$ (even whenever $x \in K_{\mathbb{R}}^\times$ itself is rational). In this section we settle these precision issues by applying the basis reduction algorithms (LLL, BKZ') on rational approximations of the bases of $x\mathfrak{a}$.

The algorithm and its analysis consists of the following steps. These steps (i) - (v) will correspond with Sections 28.2 to 28.6

(i) **Computing an (approximate) dual basis of the basis of $x\mathfrak{a}$.** For the algorithm to succeed, the basis that we start with is required to be *dually* reduced. For the purpose of reducing it, we therefore first compute the dual basis of the input basis of $x\mathfrak{a}$ (which we may assume to be in some Hermite Normal form).

(ii) **Apply the the Buchmann-Pohst-Kessler algorithm to this approximate dual basis.** To make the dual basis reduced, we apply the Buchmann-Pohst-Kessler algorithm [18, 19] (see also Section 23). We apply the inverse-transposed transformation to the original basis of $x\mathfrak{a}$, to obtain a dually exponentially reduced basis (see Definition 26.14). This part is treated in Section 28.3.

(iii) **Apply the variant of BKZ by Hanrot, Pujol and Stehlé [39].** Applying a variant of BKZ with provable running time on this dually exponentially reduced (approximate) basis of $x\mathfrak{a}$ yields an approximate basis of $x\mathfrak{a}$ with BKZ-like shortness guarantees on the basis vectors.

(iv) **Apply the closeness-lemma.** By computing an initial approximate basis with sufficiently good precision, the same unimodular transformation that makes this approximate basis have BKZ-like qualities, will also make the exact basis of $x\mathfrak{a}$ satisfy these shortness properties. Note that the unimodular transform can be actually exactly (that is, not approximately) applied to a $\mathbb{Z}$-basis of $x\mathfrak{a}$ which consists of elements $(x \cdot \alpha_1, \dots, x \cdot \alpha_n) \in K_{\mathbb{R}}$, where $\alpha_j \in K$.

(v) **Concluding.** Applying the compositions of the unimodular transformations in each step to the initial basis $(x \cdot \alpha_1, \dots, x \cdot \alpha_n)$ of $x\mathfrak{a}$ yields a new basis of $x\mathfrak{a}$ that has BKZ-like shortness guarantees.

28.2. **Computing an (approximate) dual basis of the basis of $x\mathfrak{a}$.**

Introduction. Computing the dual basis of an approximate basis $\tilde{\mathbf{B}}_{x\mathfrak{a}} \approx \mathbf{B}_{x\mathfrak{a}}$ of $x\mathfrak{a}$ is done by inverting and transposing the matrix: $\tilde{\mathbf{D}} = \tilde{\mathbf{B}}_{x\mathfrak{a}}^{-\top}$. The precision loss in numerical approximation of inversion is proportional to $\|\mathbf{B}_{x\mathfrak{a}}^{-1}\|$, the matrix norm of the inverse of the basis. Hence, we would like to upper bound this matrix norm.

We do that by applying Lemma 24.2, which bounds this matrix norm in terms of how far the basis vectors $\mathbf{B}_{x\mathfrak{a}}$ are from attaining the successive minima $\lambda_j(x\mathfrak{a})$. Since we generally assume that ideals are given in a Hermite Normal Form (HNF) in coordinates of an LLL-reduced $\mathbb{Z}$-basis of $\mathcal{O}_K$ (see Section 2.6), a reasonable upper bound on the basis vectors $\mathbf{B}_{x\mathfrak{a}}$ can be achieved. On the other hand, a lower bound on $\lambda_j(x\mathfrak{a})$ can be achieved by using ideal lattice properties from Lemma 2.13.

To obtain these results, the following lemma will turn out useful.

Lemma 28.1. *Let $x \in K_{\mathbb{R}}^{\times}$ and $\mathfrak{a} \in \mathcal{I}_K$. Let $\mathbf{B}_{x\mathfrak{a}} = (x\alpha_1, \dots, x\alpha_n) \in K_{\mathbb{R}}^n$ a basis of $x\mathfrak{a}$, where the basis $(\alpha_1, \dots, \alpha_n)$ is assumed to be in HNF form whenever written in terms of an LLL-reduced basis of $\mathcal{O}_K$ (see Section 2.6). Then*

(i) For all $j \in \{1, \dots, n\}$, we have $\|x\alpha_j\| \leq 2^{\operatorname{size}(x\mathfrak{a})} \cdot n^{3/2} \cdot 2^n \cdot |\Delta_K|^{1/n}$.
(ii) $N(x\mathfrak{a}) \leq 2^{n \operatorname{size}(x\mathfrak{a})}$.
(iii) $N(x\mathfrak{a})^{-1} \leq 2^{n \operatorname{size}(x\mathfrak{a})}$.
(iv) $\lambda_1(x\mathfrak{a})^{-1} \leq 2^{\operatorname{size}(x\mathfrak{a})}$.

where size *is defined in Section 2.6.*

Proof. For (i), we make use of the assumption that $\mathfrak{a}$ is represented by a rational HNF basis in terms of an LLL-reduced $\mathbb{Z}$-basis $(\beta_1, \dots, \beta_n)$ of $\mathcal{O}_K$ (for which thus holds $\|\beta_j\| \leq 2^n \cdot \sqrt{n} \cdot |\Delta_K|^{1/n}$ by Lemma 2.13). More precisely, $\alpha_j = \sum_{i=1}^n a_{ij}\beta_i$, where $(a_{ij})_{ij} \in \mathbb{Q}^{n \times n}$ is an HNF-reduced basis. By definition of the size of an ideal (see Section 2.6), we have that $\operatorname{size}(\mathfrak{a}) \geq \operatorname{size}(a_{ij})$. Recall that the size of rational numbers a/b with $\gcd(a,b) = 1$ is equal to $\log_2|a| + \log_2|b|$. Hence, using size$(x\mathfrak{a})$ = size(x) + size$(\mathfrak{a})$

$$\|x\alpha_j\| \leq n \cdot \max_j |x_j| \cdot \max_{ij} 2^{\text{size}(a_{ij})} \cdot \underbrace{\max_j \|\beta_j\|}_{\leq 2^n \cdot \sqrt{n} \cdot |\Delta_K|^{1/n}} \leq 2^{\text{size}(x\mathfrak{a})} \cdot n^{3/2} \cdot 2^n \cdot |\Delta_K|^{1/n}.$$

For (ii) note that $N(x\mathfrak{a})$ is equal to $\det((a_{ij})_{ij}) \cdot \prod_j x_j$. By Hadamard's bound, $\det((a_{ij})_{ij}) \leq \prod_j \|(a_{ij})_i\|_2 \leq \prod_j (\sum_i |a_{ij}|)$ (by the inequality between 1 and 2-norms) and the latter can be bounded by $(\sum_i |a_{ij}|)^n$ and hence $\log_2 N(x\mathfrak{a}) \leq n\,\text{size}(x) + n \log_2(\sum_i |a_{ij}|) \leq n\,\text{size}(x) + n\,\text{size}(\mathfrak{a}) = n\,\text{size}(x\mathfrak{a})$.

Item (iii) can be proven as follows. Because $\mathcal{N}(x\mathfrak{a}) = |\mathcal{N}(x)| \cdot \mathcal{N}(\mathfrak{a})$ and $|\mathcal{N}(x)|^{-1} = \prod_{j=1}^n |x_j|^{-1} \leq 2^{\text{size}(x)}$, we only need to show that $\mathcal{N}(\mathfrak{a})^{-1} \leq 2^{n\,\text{size}(\mathfrak{a})}$. Write $\mathfrak{a} = \frac{1}{d} \cdot \mathfrak{b}$ with d a minimal positive integer and $\mathfrak{b}$ an integral ideal. Since d minimally clears the denominators of $\mathfrak{a}$, we certainly have size$(d) \leq$ size$(\mathfrak{a})$. Now $\mathcal{N}(\mathfrak{a})^{-1} = \mathcal{N}(\mathfrak{a}^{-1}) = |\mathcal{N}(d)| \cdot \mathcal{N}(\mathfrak{b})^{-1} \leq |\mathcal{N}(d)| = d^n \leq 2^{n\,\text{size}(d)} \leq 2^{n\,\text{size}(\mathfrak{a})}$.

Item (iv) holds because $x\mathfrak{a}$ is an ideal lattice, which implies $\lambda_1(x\mathfrak{a}) \geq \mathcal{N}(x\mathfrak{a})^{1/n}$ due to item (i) from Lemma 2.13. Hence $\lambda_1(x\mathfrak{a})^{-1} \leq \mathcal{N}(x\mathfrak{a})^{-1/n}$ and the result follows from item (iii) □

Bounding the matrix norm of $\mathbf{B}_{x\mathfrak{a}}^{-1}$. We start by bounding the matrix norm of the exact inverse of $\mathbf{B}_{x\mathfrak{a}}$.

Lemma 28.2. *Let $x \in K_{\mathbb{R}}^{\times}$ and $\mathfrak{a} \in \mathcal{I}_K$. Let $\mathbf{B}_{x\mathfrak{a}} = (x\alpha_1, \ldots, x\alpha_n) \in K_{\mathbb{R}}^n$ a basis of $x\mathfrak{a}$, where the basis $(\alpha_1, \ldots, \alpha_n)$ is assumed to be in HNF form whenever written in terms of an LLL-reduced basis of $\mathcal{O}_K$ (see Section 2.6). Then*

$$\|\mathbf{B}_{x\mathfrak{a}}^{-1}\|_2 \leq n^{2n+1} \cdot 2^{n^2} \cdot |\Delta_K|^{1/2} \cdot 2^{(2n+1)\,\text{size}(x\mathfrak{a})}.$$

Proof. We apply Lemma 24.2 to obtain

$$\begin{aligned}\|\mathbf{B}_{x\mathfrak{a}}^{-1}\|_2 &\leq n^{n/2+1} \cdot \lambda_1(x\mathfrak{a})^{-1} \cdot \left(\prod_{j=1}^n \frac{\|x\alpha_j\|}{\lambda_j(x\mathfrak{a})} \right) \\ &\leq n^{n/2+1} \cdot \lambda_1(x\mathfrak{a})^{-1} \cdot |\Delta_K|^{-1/2} \cdot \mathcal{N}(x\mathfrak{a})^{-1} \cdot \prod_{j=1}^n \|x\alpha_j\|. \qquad (163)\end{aligned}$$

where we used $\prod_{j=1}^n \lambda_j(x\mathfrak{a}) \geq \text{Vol}(x\mathfrak{a}) = \mathcal{N}(x\mathfrak{a})|\Delta_K|^{1/2}$ in the last inequality.

Using that $\mathcal{N}(x\mathfrak{a})^{-1} \leq 2^{n\,\text{size}(x\mathfrak{a})}$, $\lambda_1(x\mathfrak{a})^{-1} \leq 2^{\text{size}(x\mathfrak{a})}$ and $\|x\alpha_j\| \leq 2^{\text{size}(x\mathfrak{a})} \cdot n^{3/2} \cdot 2^n \cdot |\Delta_K|^{1/n}$ for all $j \in \{1, \ldots, n\}$ (see Lemma 28.1(iii), (iv) and (i)), and applying it to Equation (163), we obtain

$$\begin{aligned}\|\mathbf{B}_{x\mathfrak{a}}^{-1}\|_2 &\leq n^{n/2+1} \cdot \lambda_1(x\mathfrak{a})^{-1} \cdot |\Delta_K|^{-1/2} \cdot \mathcal{N}(x\mathfrak{a})^{-1} \cdot \prod_{j=1}^n \|x\alpha_j\| \\ &\leq n^{n/2+1} \cdot 2^{\text{size}(x\mathfrak{a})} \cdot |\Delta_K|^{-1/2} \cdot 2^{n\,\text{size}(x\mathfrak{a})} \cdot \left(2^{\text{size}(x\mathfrak{a})} \cdot n^{3/2} \cdot 2^n \cdot |\Delta_K|^{1/n}\right)^n \\ &\leq n^{2n+1} \cdot 2^{n^2} \cdot |\Delta_K|^{1/2} \cdot 2^{(2n+1)\,\text{size}(x\mathfrak{a})}.\end{aligned}$$

□

We can therefore conclude that the loss in precision by inverting an approximate basis of $x\mathfrak{a}$ can be reasonably upper bounded.

Lemma 28.3. *Let* $\tilde{\mathbf{B}}_{x\mathfrak{a}} \in \mathbb{Q}^{n\times n}$ *be an approximation of* $\mathbf{B}_{x\mathfrak{a}}$*, then*

$$\log_2\|\tilde{\mathbf{B}}_{x\mathfrak{a}}^{-1} - \mathbf{B}_{x\mathfrak{a}}^{-1}\|_2 \leq \log_2\|\tilde{\mathbf{B}}_{x\mathfrak{a}} - \mathbf{B}_{x\mathfrak{a}}\|_2 + 5n^2 + \tfrac{1}{2}\log_2|\Delta_K| + 3n\,\mathrm{size}(x\mathfrak{a}),$$

under the assumption that $\log_2\|\mathbf{B}_{x\mathfrak{a}} - \tilde{\mathbf{B}}_{x\mathfrak{a}}\|_2 \leq -(5n^2 + \frac{1}{2}\cdot\log_2|\Delta_K| + 3n\,\mathrm{size}(x\mathfrak{a}))$*.*

Proof. By [7, Cor. 7.2, Eq. 7.46], we have that

$$\|\mathbf{B}^{-1} - \tilde{\mathbf{B}}^{-1}\|_2 \leq 2\|\mathbf{B}^{-1}\|_2^2 \cdot \|\mathbf{B} - \tilde{\mathbf{B}}\|_2,$$

as long as $2\|\mathbf{B}^{-1}\|_2 \cdot \|\mathbf{B} - \tilde{\mathbf{B}}\|_2 \leq 1$. Instantiating with $\mathbf{B} = \mathbf{B}_{x\mathfrak{a}}$, $\tilde{\mathbf{B}} = \tilde{\mathbf{B}}_{x\mathfrak{a}}$ and the bound of Lemma 28.2, we obtain the claim, using the following computation:

$$\begin{aligned}\log_2(2\|\mathbf{B}_{x\mathfrak{a}}^{-1}\|) &\leq 1 + (2n+1)\log_2(n) + n^2 + \tfrac{1}{2}\log|\Delta_K| + (2n+1)\,\mathrm{size}(x\mathfrak{a}) \\ &\leq 2n + 2n^2 + n^2 + \tfrac{1}{2}\log|\Delta_K| + (2n+1)\,\mathrm{size}(x\mathfrak{a}) \\ &\leq 5n^2 + \tfrac{1}{2}\log|\Delta_K| + 3n\,\mathrm{size}(x\mathfrak{a})\end{aligned}$$

using the fact that for any positive integer n it holds that $2n \leq 2n^2$, $\log_2(n) \leq n$ and $\log_2(n) + 1 \leq 2n$. The condition $2\|\mathbf{B}^{-1}\|_2 \cdot \|\mathbf{B} - \tilde{\mathbf{B}}\|_2 \leq 1$ holds by the assumption in the lemma's statement. □

Hence, the inversion of the approximate basis $\tilde{\mathbf{B}}_{x\mathfrak{a}}$ only yields a polynomial (in the size of the input) loss of bit precision.

Lemma 28.4. *Let* $\tilde{\mathbf{B}}_{x\mathfrak{a}}$ *be an approximation of a T-dually exponentially reduced basis* $\mathbf{B}_{x\mathfrak{a}}$ *with* $\log\|\tilde{\mathbf{B}}_{x\mathfrak{a}} - \mathbf{B}_{x\mathfrak{a}}\|_2 \leq -(9n^2 + \frac{5}{2}\log_2|\Delta_K| + 5n\,\mathrm{size}(x\mathfrak{a}) + n(T+2))$*. Then* $\tilde{\mathbf{B}}_{x\mathfrak{a}}$ *is* $(T+3)$*-dually exponentially reduced.*

Proof. Let us write $(\mathbf{d}_1, \ldots, \mathbf{d}_n) = \mathbf{D} = \mathbf{B}_{x\mathfrak{a}}^{-\top}$ and $(\tilde{\mathbf{d}}_1, \ldots, \tilde{\mathbf{d}}_n) = \tilde{\mathbf{D}} = \tilde{\mathbf{B}}_{x\mathfrak{a}}^{-\top}$. Let us also define ε such that $\log_2(1/\varepsilon) = 4n^2 + 2\log_2|\Delta_K| + 2n\,\mathrm{size}(x\mathfrak{a})$. By Lemma 28.3 (and since transposing does not change the 2-norm of matrices), we have

$$\begin{aligned}\log_2\|\mathbf{D} - \tilde{\mathbf{D}}\|_2 &\leq \log_2\|\tilde{\mathbf{B}}_{x\mathfrak{a}} - \mathbf{B}_{x\mathfrak{a}}\|_2 + 5n^2 + \tfrac{1}{2}\log_2|\Delta_K| + 3n\,\mathrm{size}(x\mathfrak{a}) \\ &\leq -(4n^2 + 2\log_2|\Delta_K| + 2n\,\mathrm{size}(x\mathfrak{a})) \leq \log(\varepsilon).\end{aligned}$$

Then, we have, for all j that $\|\tilde{\mathbf{d}_j} - \mathbf{d}_j\| = \|(\mathbf{D} - \tilde{\mathbf{D}})\mathbf{e}_j\| \leq \|\mathbf{D} - \tilde{\mathbf{D}}\|_2 \leq \varepsilon$, which implies that

$$\|\tilde{\mathbf{d}_j}\| = \|\mathbf{d}_j\| + \varepsilon \leq 2^{Tn}\lambda_j((x\mathfrak{a})^\vee) + \varepsilon \leq (2^{Tn}+1)\lambda_j((x\mathfrak{a})^\vee) \leq 2^{(T+1)n}\lambda_j((x\mathfrak{a})^\vee), \tag{164}$$

where, in the second inequality, we used the fact that $\varepsilon \leq \lambda_1((x\mathfrak{a})^\vee)$. This inequality comes from combining the transference bound (Theorem 26.15) with items (ii) and (iv) from Lemma 2.13 and item (ii) of Lemma 28.1

$$\begin{aligned}\lambda_1((x\mathfrak{a})^\vee) &\geq \lambda_n(x\mathfrak{a})^{-1} \geq (\sqrt{n}\cdot|\Delta_K|^{3/2}\cdot\mathcal{N}(x\mathfrak{a})^{1/n})^{-1} \\ &\geq (\sqrt{n}\cdot|\Delta_K|^{3/2}\cdot 2^{\mathrm{size}(x\mathfrak{a})})^{-1} \geq \varepsilon\end{aligned}$$

Writing $\tilde{\Lambda}$ for the lattice generated by the approximation $\tilde{\mathbf{B}}_{x\mathfrak{a}}$, it suffices to show that $\lambda_j((x\mathfrak{a})^\vee) \leq 2n\lambda_j(\tilde{\Lambda}^\vee)$ for all j, to show that $\tilde{\mathbf{B}}_{x\mathfrak{a}}$ is $(T+2)$-dually exponentially reduced. Indeed, then, by Equation (164), we have $\|\tilde{\mathbf{d}_j}\| \leq 2^{(T+1)n}\lambda_j((x\mathfrak{a})^\vee) \leq 2^{(T+3)n}\lambda_j(\tilde{\Lambda}^\vee)$ for all j.

By transference (see Theorem 26.15), we have

$$\frac{1}{n} \le \frac{\lambda_j(\Lambda^\vee)\cdot\lambda_{n-j}(\Lambda)}{\lambda_j(\tilde{\Lambda}^\vee)\cdot\lambda_{n-j}(\tilde{\Lambda})} \le n$$

hence $\lambda_j(\Lambda^\vee) \le n\frac{\lambda_{n-j}(\tilde{\Lambda})}{\lambda_{n-j}(\Lambda)}\cdot\lambda_j(\tilde{\Lambda}^\vee) \le 2n\cdot\lambda_j(\tilde{\Lambda}^\vee)$, if one shows that $\frac{\lambda_{n-j}(\tilde{\Lambda})}{\lambda_{n-j}(\Lambda)} \le 2$, i.e., $\lambda_j(\tilde{\Lambda}) \le 2\lambda_j(\Lambda)$ for all j. This latter statement is what we finish the proof with.

Write $\mathbf{x}_j \in x\mathfrak{a} = \Lambda$ for the vector attaining $\lambda_j(x\mathfrak{a})$. Then $\mathbf{x}_j = \mathbf{B}_{x\mathfrak{a}}\mathbf{u}_j$ for some $\mathbf{u}_j \in \mathbb{Z}^n$. By the closeness lemma (Lemma 28.6), we must have $\|\mathbf{u}_j\| \le 2^{(T+2)n}\cdot\lambda_j(x\mathfrak{a})/\lambda_1(x\mathfrak{a}) \le 2^{(T+2)n}\cdot|\Delta_K|^{1/n}$ (by Lemma 2.13). Hence $\tilde{\mathbf{x}}_j = \tilde{\mathbf{B}}_{x\mathfrak{a}}\mathbf{u}_j = \mathbf{x}_j + (\mathbf{B}_{x\mathfrak{a}} - \tilde{\mathbf{B}}_{x\mathfrak{a}})\mathbf{u}_j$ satisfy the property that $\{\tilde{\mathbf{x}}_1,\ldots,\tilde{\mathbf{x}}_j\}$ span a j-dimensional space, and therefore surely

$$\begin{aligned}\lambda_j(\tilde{\Lambda}) &\le \lambda_j(x\mathfrak{a}) + \|\mathbf{B}_{x\mathfrak{a}} - \tilde{\mathbf{B}}_{x\mathfrak{a}}\|_2\cdot 2^{(T+2)n}\cdot|\Delta_K|^{1/n}\\ &\le \lambda_j(x\mathfrak{a}) + \lambda_1(x\mathfrak{a}) \le 2\cdot\lambda_j(x\mathfrak{a}),\end{aligned}$$

since $\|\mathbf{B}_{x\mathfrak{a}} - \tilde{\mathbf{B}}_{x\mathfrak{a}}\|_2 \le 2^{-n\,\mathrm{size}(x\mathfrak{a}) - \log|\Delta_K| - (T+2)n} \le \mathcal{N}(x\mathfrak{a})^{1/n}\cdot|\Delta_K|^{-1/n}\cdot 2^{-(T+2)n} \le \lambda_1(x\mathfrak{a})\cdot 2^{-(T+2)n}\cdot|\Delta_K|^{-1/n}$ (where we used item (i) from Lemma 2.13). □

28.3. **Applying Buchmann-Pohst-Kessler to the approximate dual basis.** We apply Theorem 23.5 with $k = r_0 = n_2 = [K:\mathbb{Q}] =: n$, $n_1 = 0$, $\mathbf{A} = \mathbf{B}_{x\mathfrak{a}}^{-\top}$ and $\tilde{\mathbf{A}} = \tilde{\mathbf{B}}_{x\mathfrak{a}}^{-\top}$. To satisfy the initial conditions, we must have $\log_2\|\mathbf{B}_{x\mathfrak{a}}^{-\top} - \tilde{\mathbf{B}}_{x\mathfrak{a}}^{-\top}\|_{2,\infty} < \log_2(1/4\cdot\mu\cdot C_0^{-1})$, where[40] $\mu = \mathcal{N}(x\mathfrak{a})^{-1/n}\cdot n^{-1/2}\cdot|\Delta_K|^{-\frac{3}{2n}} \le \lambda_n(x\mathfrak{a})^{-1} \le \lambda_1((x\mathfrak{a})^\vee)$. And

$$C_0 = 2^{8n}\cdot\left(\frac{n\cdot 4^n\cdot\|\mathbf{B}_{x\mathfrak{a}}^{-\top}\|_{2,\infty}}{\mu}\right)^{2(n+1)}$$

By Lemma 28.2, we have that $\|\mathbf{B}_{x\mathfrak{a}}^{-\top}\|_{2,\infty} \le \|\mathbf{B}_{x\mathfrak{a}}^{-\top}\|_2 \le n^{2n+1}2^{n^2}|\Delta_K|^{1/2}\cdot 2^{2n\,\mathrm{size}(x\mathfrak{a})}$. Hence, $-\log_2(1/4\cdot\mu\cdot C_0^{-1}) = O(n^3 + n\log|\Delta_K| + n^2\,\mathrm{size}(x\mathfrak{a}))$, and a precision polynomial in $n, \log|\Delta_K|$ and $\mathrm{size}(x\mathfrak{a})$ is sufficient to apply Buchmann-Pohst-Kessler. Hence we obtain a unimodular transformation $\mathbf{U}$ such that the basis elements of $\mathbf{B}_{x\mathfrak{a}}^{-\top}\cdot\mathbf{U}$ have an LLL-like shortness quality compared to the successive minima of $(x\mathfrak{a})^\vee$. More precisely, the basis elements $(\mathbf{c}_1,\ldots,\mathbf{c}_n) = \mathbf{B}_{x\mathfrak{a}}^{-\top}\cdot\mathbf{U}$ satisfy

$$\|\mathbf{c}_j\| \le (n+2)2^{\frac{n-1}{2}}\cdot\lambda_j(x\mathfrak{a}^\vee) \le 2^{3n}\cdot\lambda_j(x\mathfrak{a}^\vee),$$

hence, the basis $\mathbf{C}_{x\mathfrak{a}} := (\mathbf{B}_{x\mathfrak{a}}^{-\top}\cdot\mathbf{U})^{-\top} = \mathbf{B}_{x\mathfrak{a}}\mathbf{U}^{-\top}$ is T-dually exponentially reduced with $T := 3$ (see Definition 26.14). For this last computation, note that both inverse and transpose switch positions, hence doing them both keeps the positions of the matrices.

We can conclude with the following proposition.

Proposition 28.5. *There exists an algorithm that computes a 3-dually exponentially reduced $\mathbb{Z}$-basis $(x\alpha_1,\ldots,x\alpha_n)$ of an ideal lattice $x\mathfrak{a}$ in time polynomial in $n, \log|\Delta_K|$ and $\mathrm{size}(x\mathfrak{a})$.*

Proof. Computing $\tilde{\mathbf{B}}_{x\mathfrak{a}}$ with sufficient precision of order $O(n^3 + n\log|\Delta_K| + n^2\,\mathrm{size}(x\mathfrak{a}))$ allows for computing the dual basis $\tilde{\mathbf{B}}_{x\mathfrak{a}}^{-\top} = (\tilde{\mathbf{B}}_{x\mathfrak{a}}^{-1})^\top$ and applying the Buchmann-Kessler-Pohst algorithm (see Lemma 28.3 and the discussion above). Hence, by applying the inverse transpose $\mathbf{U}^{-\top}$ of the unimodular matrix $\mathbf{U}$ that is the output

[40] We have $\lambda_n(x\mathfrak{a}) \le \sqrt{n}\cdot|\Delta_K|^{\frac{3}{2n}}\cdot\mathcal{N}(x\mathfrak{a})^{1/n}$ by items (ii) and (iv) of Lemma 2.13.

of the Buchmann-Kessler-Pohst algorithm to the initial $\mathbb{Z}$-basis $(x\alpha_1, \ldots, x\alpha_n)$ of $x\mathfrak{a}$ yields a 3-dually exponentially reduced basis of $x\mathfrak{a}$. □

28.4. **Applying the variant of BKZ by Hanrot, Pujol and Stehlé.** By clearing the denominators in the (new, dually exponentially reduced) matrix $\tilde{\mathbf{B}}_{x\mathfrak{a}}$, we obtain an integer basis on which we can apply Algorithm 7, a variant of the BKZ algorithm from Hanrot, Pujol and Stehlé. By Corollary 27.4, this algorithm outputs in time[41] $\mathsf{poly}(n, \text{size}(\tilde{\mathbf{B}}_{x\mathfrak{a}})) \cdot \mathsf{b}^{\mathsf{b}}$ a basis $\tilde{\mathbf{C}}_{x\mathfrak{a}} = (\mathbf{c}_1, \ldots, \mathbf{c}_n)$ of $x\mathfrak{a}$ that satisfies

$$\|\mathbf{c}_j\| \leq n \cdot \mathsf{b}^{2n/\mathsf{b}} \cdot \lambda_n(x\mathfrak{a}).$$

28.5. **The closeness lemma.**

Lemma 28.6. *Let $\mathbf{B} \in \mathbb{R}^{m \times n}$ be basis of a rank-n lattice Λ that is T-dually exponentially reduced as in Definition 26.14 and let $\mathbf{v} \in \Lambda$ be any lattice vector, with $\mathbf{v} = \mathbf{B}\mathbf{u}$ for some integral vector $\mathbf{u}$. Then*

$$\|\mathbf{u}\|^2 \leq n^3 \cdot 2^{nT} \cdot \|\mathbf{v}\|^2/\lambda_1(\Lambda)^2 \leq 2^{n(T+2)} \cdot \|\mathbf{v}\|^2/\lambda_1(\Lambda)^2.$$

Proof. Write $\mathbf{Q} = \mathbf{B}^\top\mathbf{B}$ for the quadratic form associated to $\mathbf{B}$; then $\mathbf{Q}^{-1} = (\mathbf{B}^\top\mathbf{B})^{-1} = \mathbf{B}^{-1}\mathbf{B}^{-\top} = \mathbf{D}^\top\mathbf{D}$ is the quadratic form associated to the dual basis $\mathbf{D} = \mathbf{B}^{-\top} = (\mathbf{d}_1, \ldots, \mathbf{d}_n)$ of $\mathbf{B}$.

Denote $\Lambda^\vee$ for the dual lattice of Λ. By $\mathbf{B}$ being T-dually exponentially reduced, we have $(\mathbf{Q}^{-1})_{jj} = \|\mathbf{d}_j\|^2 \leq 2^{Tn} \cdot \lambda_j(\Lambda^\vee)^2 \leq n^2 \cdot 2^{T \cdot n}/\lambda_{n-j+1}(\Lambda)^2$ by Banaszczyk's transference theorem (Theorem 26.15). Therefore

$$\text{Tr}(\mathbf{Q}^{-1}) = \sum_{i=1}^{n} (\mathbf{Q}^{-1})_{ii} \leq n^2 \cdot 2^{Tn} \sum_{i=1}^{n} \frac{1}{\lambda_i(\Lambda)^2} \leq n^3 \cdot 2^{Tn}/\lambda_1(\Lambda)^2.$$

Write $\mu_1, \ldots, \mu_n > 0$ for the eigenvalues of the positive definite matrix $\mathbf{Q}$. Then, for all $i \in \{1, \ldots, n\}$, $\frac{1}{\mu_i} \leq \text{Tr}(\mathbf{Q}^{-1}) \leq n^3 \cdot 2^{Tn}/\lambda_1(\Lambda)^2$, since all eigenvalues of $\mathbf{Q}$ are positive. Therefore, the inverse of any eigenvalue of $\mathbf{Q}$ is upper bounded by $n^3 \cdot 2^{Tn}/\lambda_1(\Lambda)^2$,

Write $\mathbf{v} = \mathbf{B}\mathbf{u}$, then

$$\|\mathbf{u}\|^2 = \mathbf{u}^\top\mathbf{u} \leq \frac{\mathbf{u}^\top\mathbf{Q}\mathbf{u}}{\min_i \mu_i} \leq \|\mathbf{v}\|^2 \cdot \frac{n^3 \cdot 2^{Tn}}{\lambda_1(\Lambda)^2}$$

which, together with the fact that $n^3 \leq 2^{2n}$, proves the claim. □

Corollary 28.7. *Let $\tilde{\mathbf{B}}$ and $\mathbf{B} \in \mathbb{R}^{m \times n}$ be T-dually exponentially reduced bases of rank-n lattices $\tilde{\Lambda}$ and Λ respectively.*

Then

$$\left(1 - \frac{2^{(T+2)n} \cdot \|\tilde{\mathbf{B}} - \mathbf{B}\|_2}{\min(\lambda_1(\Lambda), \lambda_1(\tilde{\Lambda}))}\right) \leq \frac{\lambda_j(\Lambda)}{\lambda_j(\tilde{\Lambda})} \leq \left(1 + \frac{2^{(T+2)n} \cdot \|\tilde{\mathbf{B}} - \mathbf{B}\|_2}{\min(\lambda_1(\Lambda), \lambda_1(\tilde{\Lambda}))}\right)$$

and $|\lambda_1(\tilde{\Lambda}) - \lambda_1(\Lambda)| \leq 2^{(T+2)n} \cdot \|\tilde{\mathbf{B}} - \mathbf{B}\|_2$.

[41] Clearing denominators in a basis $\mathbf{B} \in \mathbb{Q}^{m \times n}$ at most increases the size of that basis by $2 \cdot m \cdot n$. Since $d\mathbf{B}$ for $d = \prod_{ij} d_{ij}$ (where d_{ij} is the denominator of $\mathbf{B}_{ij}$) has size $mn\,\text{size}(d) + \text{size}(\mathbf{B}) \leq (mn+1)\,\text{size}(\mathbf{B}) \leq 2mn\,\text{size}(\mathbf{B})$

Proof. Write $\mathbf{U} \in \mathbb{Z}^{n\times n} \cap \mathrm{GL}_n(\mathbb{R})$ for the (not necessarily unimodular) transformation such that $\mathbf{C} = \mathbf{BU}$ attains the successive minima, i.e., $\|\mathbf{c}_j\| = \lambda_j(\Lambda)$, and the vectors $\mathbf{c}_j$ are linearly independent. By Lemma 28.6, we know that $\|\mathbf{u}_j\| \leq 2^{(T+2)n} \cdot \lambda_j(\Lambda)/\lambda_1(\Lambda)$.

Since $\mathbf{U}$ is non-singular, then $\tilde{\mathbf{C}} := \tilde{\mathbf{B}}\mathbf{U} = \tilde{\mathbf{B}}(\mathbf{u}_1, \dots, \mathbf{u}_n)$ consists of n independent vectors. Hence $\lambda_j(\tilde{\Lambda}) \leq \max_{1\leq k\leq j} \|\tilde{\mathbf{c}}_k\|$, but we have

$$\|\tilde{\mathbf{c}}_k\| \leq \|\tilde{\mathbf{c}}_k - \mathbf{c}_k\| + \|\mathbf{c}_k\| \leq \|\tilde{\mathbf{B}} - \mathbf{B}\|_2 \cdot \|\mathbf{u}_k\| + \lambda_k(\Lambda) \leq \lambda_k(\Lambda)\left(1 + \frac{2^{(T+2)n} \cdot \|\tilde{\mathbf{B}} - \mathbf{B}\|_2}{\lambda_1(\Lambda)}\right),$$

hence, by applying the same reasoning but interchanging Λ and $\tilde{\Lambda}$,

$$\lambda_j(\tilde{\Lambda}) \leq \lambda_j(\Lambda)\left(1 + \frac{2^{(T+2)n} \cdot \|\tilde{\mathbf{B}} - \mathbf{B}\|_2}{\lambda_1(\Lambda)}\right) \text{ and } \lambda_j(\Lambda) \leq \lambda_j(\tilde{\Lambda})\left(1 + \frac{2^{(T+2)n} \cdot \|\tilde{\mathbf{B}} - \mathbf{B}\|_2}{\lambda_1(\tilde{\Lambda})}\right).$$

Instantiating this with $j = 1$, shows $|\lambda_1(\tilde{\Lambda}) - \lambda_1(\Lambda)| \leq 2^{(T+2)n} \cdot \|\tilde{\mathbf{B}} - \mathbf{B}\|_2$. Applying the inequality $1 - x \leq 1/(1+x)$ (which holds for all $x \geq 0$) and replacing $\lambda_1(\Lambda)$ and $\lambda_1(\tilde{\Lambda})$ by their minimum, we obtain the final claim

$$\left(1 - \frac{2^{(T+2)n} \cdot \|\tilde{\mathbf{B}} - \mathbf{B}\|_2}{\min(\lambda_1(\Lambda), \lambda_1(\tilde{\Lambda}))}\right) \leq \frac{\lambda_j(\Lambda)}{\lambda_j(\tilde{\Lambda})} \leq \left(1 + \frac{2^{(T+2)n} \cdot \|\tilde{\mathbf{B}} - \mathbf{B}\|_2}{\min(\lambda_1(\Lambda), \lambda_1(\tilde{\Lambda}))}\right).$$

□

Corollary 28.8. *Let $\tilde{\mathbf{B}}$ and $\mathbf{B} \in \mathbb{R}^{m\times n}$ be T-dually exponentially reduced bases of rank-n lattices $\tilde{\Lambda}$ and Λ respectively. Let $\tilde{\mathbf{C}} = \tilde{\mathbf{B}}\mathbf{U}$ be a basis of $\tilde{\Lambda}$ satisfying*

$$\|\tilde{\mathbf{c}}_j\| \leq q_j \cdot \lambda_n(\tilde{\Lambda}) \quad \textit{for all } j \in \{1, \dots, n\},$$

for $q_j \in \mathbb{R}_{>0}$. Suppose $\|\tilde{\mathbf{B}} - \mathbf{B}\|_2 \leq \frac{1}{4} \cdot 2^{-(T+2)n} \cdot \min(\lambda_1(\Lambda), \lambda_1(\tilde{\Lambda}))$ and write $\mathbf{C} = \mathbf{BU}$. Then,

$$\|\mathbf{c}_j\| \leq 2 \cdot q_j \cdot \lambda_n(\Lambda) \quad \textit{for all } j \in \{1, \dots, n\}.$$

Proof. We have $\|\mathbf{u}_j\| \leq 2^{(T+2)n}\|\tilde{\mathbf{c}}_j\|/\lambda_1(\tilde{\Lambda})$ by Lemma 28.6, and hence

$$\begin{aligned}
\|\mathbf{c}_j\| &\leq \|\mathbf{c}_j - \tilde{\mathbf{c}}_j\| + \|\tilde{\mathbf{c}}_j\| \leq \|\tilde{\mathbf{B}} - \mathbf{B}\|_2 \cdot \|\mathbf{u}_j\| + \|\tilde{\mathbf{c}}_j\| \\
&\leq \|\tilde{\mathbf{c}}_j\|\left(1 + \frac{\|\tilde{\mathbf{B}} - \mathbf{B}\|_2 \cdot 2^{(T+2)n}}{\lambda_1(\tilde{\Lambda})}\right) \leq q_j \cdot \lambda_n(\tilde{\Lambda}) \cdot \left(1 + \frac{\|\tilde{\mathbf{B}} - \mathbf{B}\|_2 \cdot 2^{(T+2)n}}{\lambda_1(\tilde{\Lambda})}\right) \\
&\leq q_j \cdot \lambda_n(\Lambda) \cdot \left(1 + \frac{\|\tilde{\mathbf{B}} - \mathbf{B}\|_2 \cdot 2^{(T+2)n}}{\min(\lambda_1(\tilde{\Lambda}), \lambda_1(\Lambda))}\right)^2 \leq 2 \cdot q_j \cdot \lambda_n(\Lambda),
\end{aligned}$$

where in the penultimate inequality, we used Corollary 28.7 and in the last inequality we used the assumed upper bound on $\|\tilde{\mathbf{B}} - \mathbf{B}\|_2$. □

28.6. Conclusion.

Lemma 8.5. *Let K be a number field of degree n. Let $x\mathfrak{a}$ be an ideal lattice where $x \in K_{\mathbb{R}}^{\times}$ is represented by rational numbers, and where $\mathfrak{a} \in \mathcal{I}_K$ is represented by a rational Hermite Normal Form matrix $M_{\mathfrak{a}}$ with respect to a given LLL-reduced basis of $\mathcal{O}_K$.*

Then there exists an algorithm that computes a $\mathbb{Z}$-basis $(x\alpha_1, \dots, x\alpha_n)$ of $x\mathfrak{a}$ with $\alpha_i \in K$ such that

$$\|x \cdot \alpha_i\| \leq 2n \cdot \mathsf{b}^{2n/\mathsf{b}} \cdot \lambda_n(x \cdot \mathfrak{a}),$$

using time at most $T = \mathsf{poly}(\mathsf{b}^{\mathsf{b}}, \mathrm{size}(\mathfrak{a}), \log|\Delta_K|, \mathrm{size}(x))$.

Proof. Using the HNF rational matrix $M_{\mathfrak{a}}$, we have a sequence of elements $(\alpha_1^{(0)}, \ldots, \alpha_n^{(0)})$ generating $\mathfrak{a}$. The exact Minkowski basis is given by $\mathbf{B}_{x\mathfrak{a}} = (\sigma(\alpha_j^{(0)}))_{\sigma,j}$, i.e., the columns of $\mathbf{B}_{x\mathfrak{a}}$ are the Minkowski embeddings of $\alpha_j^{(0)}$ into $K_{\mathbb{R}}$.

Due to the fact that these numbers are irrational in general, the basis $\mathbf{B}_{x\mathfrak{a}}$ can only be approximated; we call this approximation $\tilde{\mathbf{B}}_{x\mathfrak{a}}$. Note that the values of $\sigma(\alpha_j^{(0)})$ can reasonably be approximated by computing the roots of the defining polynomial of the number field K. This can be done efficiently by, for example, Newton iteration; this costs time polynomial in the desired bit precision. Hence, if the final required bit precision is polynomial in the size of the input (which we indeed will show to be the case), approximation is not going to have a significant influence on the running time.

Computing the dual. So, suppose we have a rational approximation $\tilde{\mathbf{B}}_{x\mathfrak{a}}$ of $\mathbf{B}_{x\mathfrak{a}}$ within polynomial bits of precision. By Lemma 28.3, computing the dual basis of $\tilde{\mathbf{B}}_{x\mathfrak{a}}$ only looses $O(n^2 + \log|\Delta_K| + (n+2)\,\mathrm{size}(x\mathfrak{a}))$ bits of precision. Therefore we can assume that the dual basis of $\tilde{\mathbf{B}}_{x\mathfrak{a}}$ approximates the dual basis of $\mathbf{B}_{x\mathfrak{a}}$ with polynomial bits of precision.

Applying Buchmann-Kessler-Pohst on the approximate dual basis. Using the Buchmann-Kessler-Pohst algorithm on the approximate dual basis, we have Proposition 28.5, which states we can compute efficiently a $\mathbb{Z}$-basis $(x\alpha_1^{(1)}, \ldots, x\alpha_n^{(1)})$ that is '3-dually exponentially reduced' (see Definition 26.14). That means the dual basis $(\mathbf{d}_1, \ldots, \mathbf{d}_n)$ of this new basis satisfies $\|\mathbf{d}_j\| \leq 2^{3n} \cdot \lambda_j((x\mathfrak{a})^\vee)$, where $(x\mathfrak{a})^\vee$ is the dual lattice of $x\mathfrak{a}$.

This new basis $(x\alpha_1^{(1)}, \ldots, x\alpha_n^{(1)})$ is exact and we can re-approximate the associated basis $\mathbf{B}_{x\mathfrak{a}}^{(1)}$ by $\tilde{\mathbf{B}}_{x\mathfrak{a}}^{(1)}$. By Lemma 28.4, a sufficient (polynomial) approximation $\tilde{\mathbf{B}}_{x\mathfrak{a}}^{(1)}$ can be shown to be 6-dually exponentially reduced as well.

Applying Hanrot-Pujol-Stehlé. Without loss of generality, we can clear the denominators in $\tilde{\mathbf{B}}_{x\mathfrak{a}}^{(1)}$. Then, we apply the BKZ version of Hanrot-Pujol-Stehlé with provable bounds on the number of tours. Using Corollary 27.4, we conclude that there exists an algorithm using time $\mathsf{poly}(n, \mathrm{size}(\tilde{\mathbf{B}}_{x\mathfrak{a}}^{(1)}), \mathsf{b}^{\mathsf{b}}) = \mathsf{poly}(\log|\Delta_K|, \mathrm{size}(x\mathfrak{a}), \mathsf{b}^{\mathsf{b}})$ that outputs a new basis $(\tilde{\mathbf{b}}_1^{(2)}, \ldots, \tilde{\mathbf{b}}_n^{(2)}) = \tilde{\mathbf{B}}_{x\mathfrak{a}}^{(2)} = \tilde{\mathbf{B}}_{x\mathfrak{a}}^{(1)} \cdot \mathbf{U}$ of the lattice $\mathcal{L}(\tilde{\mathbf{B}}_{x\mathfrak{a}}^{(1)})$ that satisfies

$$\|\tilde{\mathbf{b}}_j^{(2)}\| \leq n \cdot \mathsf{b}^{2n/\mathsf{b}} \cdot \lambda_n(\mathcal{L}(\tilde{\mathbf{B}}_{x\mathfrak{a}}^{(1)}))$$

But we would like the new *exact basis* $(x\alpha_1^{(2)}, \ldots, x\alpha_n^{(2)})$, obtained by applying $\mathbf{U}$ to $(x\alpha_1^{(1)}, \ldots, x\alpha_n^{(1)})$ to satisfy above claim. For this we need the *closeness lemma*.

Applying the closeness lemma. Note that $\tilde{\mathbf{B}}_{x\mathfrak{a}}^{(2)} = \tilde{\mathbf{B}}_{x\mathfrak{a}}^{(1)}\mathbf{U}$ is an approximation of $\mathbf{B}_{x\mathfrak{a}}^{(2)} = \mathbf{B}_{x\mathfrak{a}}^{(1)}\mathbf{U}$, the exact Minkowski-basis of the $\mathbb{Z}$-basis $(x\alpha_1^{(2)}, \ldots, x\alpha_n^{(2)})$. Since $\tilde{\mathbf{B}}_{x\mathfrak{a}}^{(1)}$ and $\mathbf{B}_{x\mathfrak{a}}^{(1)}$ are both T-dually exponentially reduced (with $T = 6$ and 3 respectively), we can apply Corollary 28.8. A sufficient (polynomial bit precision) approximation $\tilde{\mathbf{B}}_{x\mathfrak{a}}^{(1)}$ of $\mathbf{B}_{x\mathfrak{a}}^{(1)}$ then suffices to deduce that the vectors of $\mathbf{B}_{x\mathfrak{a}}^{(2)}$ satisfy the same BKZ-like bounds (with a blow-up factor of 2), hence,

$$\|x\alpha_j^{(2)}\| \leq 2n \cdot \mathsf{b}^{2n/\mathsf{b}} \cdot \lambda_n(x\mathfrak{a}) \text{ for all } j \in \{1, \ldots, n\}.$$

□

References

[1] D. Aggarwal, J. Li, P. Q. Nguyen, and N. Stephens-Davidowitz. Slide reduction, revisited - filling the gaps in SVP approximation. In *CRYPTO*, 2020.

[2] T. Apostol. *Introduction to Analytic Number Theory*. Undergraduate Texts in Mathematics. Springer New York, 1998.

[3] M. F. Atiyah and I. G. MacDonald. *Introduction to commutative algebra*. Addison-Wesley-Longman, 1969.

[4] E. Bach. Explicit bounds for primality testing and related problems. *Mathematics of Computation*, 55(191):355–380, 1990.

[5] E. Bach. Improved approximations for Euler products. *Number Theory: Fourth Conference of the Canadian Number Theory Association, July 2-8, 1994, Dalhousie University, Halifax, Nova Scotia, Canada*, 15, 1995.

[6] W. Banaszczyk. New bounds in some transference theorems in the geometry of numbers. *Mathematische Annalen*, 296(4):625–636, 1993.

[7] L. Beilina, E. Karchevskii, and M. Karchevskii. *Numerical Linear Algebra: Theory and Applications*. Springer, 09 2017.

[8] M. Bhargava, A. Shankar, T. Taniguchi, F. Thorne, J. Tsimerman, and Y. Zhao. Bounds on 2-torsion in class groups of number fields and integral points on elliptic curves. *Journal of the American Mathematical Society*, 33(4):1087–1099, Oct. 2020.

[9] J.-F. Biasse. Subexponential time relations in the class group of large degree number fields. *Adv. Math. Commun.*, 8(4):407–425, 2014.

[10] J.-F. Biasse and C. Fieker. Subexponential class group and unit group computation in large degree number fields. *LMS Journal of Computation and Mathematics*, 17:385–403, 1 2014.

[11] J.-F. Biasse and F. Song. A polynomial time quantum algorithm for computing class groups and solving the principal ideal problem in arbitrary degree number fields. In *SODA*, 2016.

[12] K. de Boer. *Random Walks on Arakelov Class Groups*. PhD thesis, Mathematical Institute (MI) , Faculty of Science , Leiden University, 2022.

[13] K. de Boer, L. Ducas, and S. Fehr. On the quantum complexity of the continuous hidden subgroup problem. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 341–370. Springer, 2020.

[14] K. de Boer, L. Ducas, A. Pellet-Mary, and B. Wesolowski. Random self-reducibility of ideal-svp via arakelov random walks. In D. Micciancio and T. Ristenpart, editors, *Advances in Cryptology – CRYPTO 2020*, pages 243–273, Cham, 2020. Springer International Publishing.

[15] K. de Boer, L. Ducas, A. Pellet-Mary, and B. Wesolowski. Random self-reducibility of ideal-SVP via Arakelov random walks. Cryptology ePrint Archive, Paper 2020/297, 2020. https://eprint.iacr.org/2020/297.

[16] K. de Boer and C. Pagano. Calculating the power residue symbol and ibeta. In *ISSAC*, volume 68, pages 923–934, 2017.

[17] J. Buchmann. A subexponential algorithm for the determination of class groups and regulators of algebraic number fields. *Séminaire de théorie des nombres, Paris*, 1989:28–41, 1988.

[18] J. Buchmann and V. Kessler. Computing a reduced lattice basis from a generating system. *Unpublished Manuscript*, 08 1996.

[19] J. Buchmann and M. Pohst. Computing a lattice basis from a system of generating vectors. In *Proceedings of the European Conference on Computer Algebra*, EUROCAL '87, pages 54–63, London, UK, UK, 1989. Springer-Verlag.

[20] J. A. Buchmann and H. W. Lenstra. Approximating rings of integers in number fields. *Journal de Théorie des Nombres de Bordeaux*, 6(2):221–260, 1994.

[21] J. Cassels. *An Introduction to the Geometry of Numbers*. Classics in Mathematics. Springer Berlin Heidelberg, 2012.

[22] H. Cohen. *A course in computational algebraic number theory*, volume 8. Springer-Verlag Berlin, 1993.

[23] H. Cohen. *Advanced Topics in Computational Number Theory*. Graduate Texts in Mathematics. Springer New York, 1999.

[24] H. Cohen and F. Diaz Y Diaz. A polynomial reduction algorithm. *Journal de théorie des nombres de Bordeaux*, 3(2):351–360, 1991.

[25] H. Cohen, F. Diaz Y Diaz, and M. Olivier. Subexponential algorithms for class group and unit computations. *Journal of Symbolic Computation*, 24(3-4):433–441, 1997.

[26] H. Cohen and P. Stevenhagen. Computational class field theory, 2008.

[27] S. D. Conte and C. W. De Boor. *Elementary Numerical Analysis: an Algorithmic Approach*. International Series in Pure and Applied Mathematics. McGraw-Hill, New York, NY, third edition, 1980.

[28] T. M. Cover and J. A. Thomas. *Elements of Information Theory 2nd Edition (Wiley Series in Telecommunications and Signal Processing)*. Wiley-Interscience, 7 2006.

[29] R. Cramer, L. Ducas, and B. Wesolowski. Mildly short vectors in cyclotomic ideal lattices in quantum polynomial time. *J ACM*, 2021.

[30] K. Eisenträger, S. Hallgren, A. Kitaev, and F. Song. A quantum algorithm for computing the unit group of an arbitrary degree number field. In *STOC*, pages 293–302. ACM, 2014.

[31] K. Ford. Vinogradov's integral and bounds for the riemann zeta function. *Proceedings of the London Mathematical Society*, 85(3):565–633, 2002.

[32] E. Friedman. Analytic formulas for the regulator of a number field. *Inventiones mathematicae*, 98:599–622, 1989.

[33] N. Gama and P. Q. Nguyen. Finding short lattice vectors within mordell's inequality. In *Proceedings of the fortieth annual ACM symposium on Theory of computing*, pages 207–216. ACM, 2008.

[34] F. R. Gantmacher. *The Theory of Matrices, Vol. I, II.* Chelsea Publishing Comp., New York, 1959.

[35] A. Gélin and A. Joux. Reducing number field defining polynomials: an application to class group computations. *LMS Journal of Computation and Mathematics*, 19(A):315–331, 2016.

[36] C. Gentry, C. Peikert, and V. Vaikuntanathan. Trapdoors for hard lattices and new cryptographic constructions. In *STOC*, pages 197–206, 2008.

[37] L. Grenié and G. Molteni. Explicit versions of the prime ideal theorem for Dedekind zeta functions under GRH. *Mathematics of Computation*, 85(298):889–906, Oct 2015.

[38] J. L. Hafner and K. S. McCurley. A rigorous subexponential algorithm for computation of class groups. *Journal of the American mathematical society*, 2(4):837–850, 1989.

[39] G. Hanrot, X. Pujol, and D. Stehlé. Analyzing blockwise lattice algorithms using dynamical systems. In *CRYPTO*, 2011.

[40] G. Hanrot and D. Stehlé. Improved analysis of Kannan's shortest lattice vector algorithm. pages 170–186, 08 2007.

[41] I. Haviv and O. Regev. *On the Lattice Isomorphism Problem*, pages 391–404.

[42] R. Horn and C. Johnson. *Matrix Analysis*. Cambridge University Press, 2012.

[43] I. C. F. Ipsen and R. Rehman. Perturbation bounds for determinants and characteristic polynomials. *SIAM Journal on Matrix Analysis and Applications*, 30(2):762–776, 2008.

[44] H. Iwaniec and E. Kowalski. *Analytic Number Theory*. American Mathematical Society, 2004.

[45] E. Kaltofen and G. Villard. On the complexity of computing determinants. *Computational complexity*, 13(3-4):91–130, 2005.

[46] V. Kessler. On the minimum of the unit lattice. *Séminaire de Théorie des Nombres de Bordeaux*, 3(2):377–380, 1991.

[47] J. Lagarias and A. Odlyzko. Effective versions of the Chebotarev density theorem. In *Algebraic number fields: L-functions and Galois properties (Proc. Sympos., Univ. Durham, Durham, 1975)*, pages 409–464. Academic Press, London, 1977.

[48] J. C. Lagarias, H. W. Lenstra Jr., and C.-P. Schnorr. Korkin-Zolotarev bases and successive minima of a lattice and its reciprocal lattice. *Combinatorica*, 10(4):333–348, 1990.

[49] S. Lang. *Algebraic Number Theory*. Graduate Texts in Mathematics. Springer, 1994.

[50] S. Lang. *Algebra*. Graduate Texts in Mathematics. Springer New York, 2005.

[51] A. K. Lenstra, H. W. Lenstra Jr., et al. *The development of the number field sieve*, volume 1554. Springer Science & Business Media, 1993.

[52] A. K. Lenstra, H. W. Lenstra Jr., and L. Lovász. Factoring polynomials with rational coefficients. *Mathematische Annalen*, 261(4):515–534, December 1982.

[53] S. Louboutin. Explicit bounds for residues of Dedekind zeta functions, values of L-functions at s=1, and relative class numbers. *Journal of Number Theory*, 2000.

[54] D. Micciancio and S. Goldwasser. *Complexity of Lattice Problems: a cryptographic perspective*, volume 671 of *The Kluwer International Series in Engineering and Computer Science*. Kluwer Academic Publishers, Boston, Massachusetts, 2002.

[55] D. Micciancio and O. Regev. Worst-case to average-case reductions based on gaussian measures. *SIAM J. Comput.*, 37(1):267–302, Apr. 2007.

[56] H. Minkowski. *Gesammelte Abhandlungen*. Chelsea, New York, 1967.

[57] T. Miyake and Y. Maeda. *Modular Forms*. Springer Monographs in Mathematics. Springer Berlin Heidelberg, 2006.

[58] J. Neukirch and N. Schappacher. *Algebraic Number Theory*. Grundlehren der mathematischen Wissenschaften. Springer Berlin Heidelberg, 2013.

[59] P. Q. Nguyen and D. Stehlé. An LLL algorithm with quadratic complexity. *SIAM Journal on Computing*, 39(3):874–903, 2009.

[60] M. Overholt. *A Course in Analytic Number Theory*. Graduate Studies in Mathematics. American Mathematical Society, 2014.

[61] L. Pardo. *Statistical Inference Based on Divergence Measures*. CRC Press, Abingdon, 2005.

[62] C. Peikert and A. Rosen. Lattices that admit logarithmic worst-case to average-case connection factors. Cryptology ePrint Archive, Paper 2006/444, 2006.

[63] A. Pellet-Mary, G. Hanrot, and D. Stehlé. Approx-SVP in ideal lattices with pre-processing. In *Eurocrypt*, pages 685–716. Springer, 2019.

[64] M. Plançon and T. Prest. Exact lattice sampling from non-Gaussian distributions. In J. A. Garay, editor, *Public-Key Cryptography – PKC 2021*, pages 573–595, Cham, 2021. Springer International Publishing.

[65] C. Pomerance. Fast, rigorous factorization and discrete logarithm algorithms. In D. S. Johnson, T. Nishizeki, A. Nozaki, and H. S. Wilf, editors, *Discrete Algorithms and Complexity*, pages 119–143. Academic Press, 1987.

[66] S. Rabinowitz. The volume of an n-simplex with many equal edges. *Missouri Journal of Mathematical Sciences*, 1, 01 1989.

[67] J. B. Rosser and L. Schoenfeld. Approximate formulas for some functions of prime numbers. *Illinois Journal of Mathematics*, 6(1):64–94, 1962.

[68] C.-P. Schnorr. A hierarchy of polynomial time lattice basis reduction algorithms. *Theor. Comput. Sci.*, 53:201–224, 1987.

[69] R. Schoof. Computing Arakelov class groups. In *Algorithmic Number Theory: Lattices, Number Fields, Curves and Cryptography*, pages 447–495. Cambridge University Press, 2008.

[70] M. Seysen. A probabilistic factorization algorithm with quadratic forms of negative discriminant. *Mathematics of computation*, 48(178):757–780, 1987.

[71] V. Shoup. A new polynomial factorization algorithm and its implementation. *Journal of Symbolic Computation*, 20(4):363 – 397, 1995.

[72] A. Storjohann and G. Labahn. Asymptotically fast computation of Hermite normal forms of integer matrices. In *Proceedings of the 1996 International Symposium on Symbolic and Algebraic Computation*, ISSAC '96, page 259–266, New York, NY, USA, 1996. Association for Computing Machinery.

[73] J. E. Sunley. On the class numbers of totally imaginary quadratic extensions of totally real fields. *Bulletin of the American Mathematical Society*, 78(1):74–76, 1972.

[74] T. Trudgian. Updating the error term in the prime number theorem. *The Ramanujan Journal*, 39(2):225–234, Feb 2016.

[75] U. von Krause et al. Abschätzungen für die funktion ωk (x, y) in algebraischen zahlkörpern. *Manuscripta mathematica*, 69(1):319–331, 1990.

[76] J. von zur Gathen and J. Gerhard. *Modern Computer Algebra*. Cambridge University Press, 1999.

[77] J. von zur Gathen and D. Panario. Factoring polynomials over finite fields: A survey. *Journal of Symbolic Computation*, 31(1):3 – 17, 2001.

[78] B. P. Wesolowski. *Arithmetic and geometric structures in cryptography*. PhD thesis, École Polytechnique Fédérale de Lausanne, 11 2018.

PART 4

Appendix

A.1. Proof of Theorem 6.4, the random walk theorem

This following proof of Theorem 6.4 is put in the appendix because it is very similar to proofs in other work [14, Section 3] [12, Chapter 4], with the difference that it is here generalized to arbitrary moduli $\mathfrak{m}$ and finite-index subgroups of $\mathrm{Pic}^0_{K^{\mathfrak{m}}}$, and a slight change of the definition of the log-space H.

Theorem 6.4 (Random Walks in finite-index subgroups of the Arakelov Ray Class Group, ERH)**.** *Let $\varepsilon > 0$ and $s > 0$ be any positive real numbers and let $k \in \mathbb{R}_{>0}$ be a positive real number as well. Let $G \subseteq \mathrm{Pic}^0_{K^{\mathfrak{m}}}$ be a finite-index subgroup of the Arakelov ray class group. Putting*[42] *$\tilde{s} = \min(\sqrt{2} \cdot s, 1/\eta_1(\mathrm{Log}(\mathcal{O}^{\times}_{K^{\mathfrak{m}},1})^{\vee}))$, there exists a bound*

$$B = \widetilde{O}\Big([\mathrm{Pic}^0_{K^{\mathfrak{m}}} : G]^2 \cdot n^{2k} \big[n^2(\log\log(1/\varepsilon))^2 + n^2(\log(1/\tilde{s}))^2 + (\log(|\Delta_K|\mathcal{N}(\mathfrak{m})))^2\big]\Big)$$

such that for any integer

$$N \geq \left\lceil \frac{1}{2k\log n} \cdot (\mathbb{r} \cdot \log(1/\tilde{s}) + 2\log(1/\varepsilon) + \log|\mathrm{Pic}^0_{K^{\mathfrak{m}}}| - \log[\mathrm{Pic}^0_{K^{\mathfrak{m}}} : G] + 2)\right\rceil, \tag{24}$$

the random walk distribution $[\mathcal{W}_G(B, N, s)]$ is ε-close to uniform in $L_1(G)$, i.e.,

$$\|[\mathcal{W}_G(B,N,s)] - \mathcal{U}(G)\|_1 \leq \varepsilon,$$

where $\mathcal{N}(\mathfrak{m}) = \mathcal{N}(\mathfrak{m}_0) \cdot 2^{|\mathfrak{m}_{\mathbb{R}}|}$, $\mathbb{r} = \dim(H) = n_{\mathbb{R}} + n_{\mathbb{C}} - 1$ and $|\mathfrak{m}_{\mathbb{R}}|$ is the number of different real places dividing $\mathfrak{m}$ (see Section 2.4).

We follow the proof of [14, Section 3], with small adaptations.

The Hecke operator, characters and eigenvalues

Recall that $G \subseteq \mathrm{Pic}^0_{K^{\mathfrak{m}}}$ is a finite-index subgroup of $\mathrm{Pic}^0_{K^{\mathfrak{m}}}$ where we try to randomize over. Putting $\mathcal{P} = \{\mathfrak{p} \text{ prime ideal of } K \mid \mathcal{N}(\mathfrak{p}) \leq B, [d^0(\mathfrak{p})] \in G \text{ and } \mathfrak{p} \nmid \mathfrak{m}\}$, we define the Hecke operator $\mathcal{H} : L^2(G) \to L^2(G)$ by the following rule, for $f \in L^2(G)$.

$$\mathcal{H}(f)(x) := \frac{1}{|\mathcal{P}|}\sum_{\mathfrak{p}\in\mathcal{P}} f(x - [d^0(\mathfrak{p})]).$$

This Hecke operator has the *characters* $\chi \in \widehat{G}$ as eigen functions, with eigenvalues $\lambda_\chi \in \mathbb{C}$ satisfying $|\lambda_\chi| \leq 1$ [14, Section 3.2]. The trivial character $\mathbf{1} \in \widehat{G}$ can be

[42] Recall that for any lattice Λ, we write $\Lambda^{\vee}$ for its dual, and $\eta_1(\Lambda)$ for its smoothing parameter (see page 18).

shown to have eigenvalue $\lambda_{\mathbf{1}} = 1$.

$$\mathcal{H}(\chi)(\cdot) := \frac{1}{|\mathcal{P}|}\sum_{\mathfrak{p}\in\mathcal{P}} \chi(\cdot - [d^0(\mathfrak{p})]) = \chi(\cdot)\cdot \underbrace{\frac{1}{|\mathcal{P}|}\sum_{\mathfrak{p}\in\mathcal{P}}\chi(-[d^0(\mathfrak{p})])}_{\lambda_\chi}$$

For a fixed $\chi \in \widehat{G}$, there are $[\mathrm{Pic}^0_{K^\mathfrak{m}} : G]$ extensions of this character to $\theta \in \widehat{\mathrm{Pic}^0_{K^\mathfrak{m}}}$. These characters satisfy the following identity, which can be proved by using standard character orthogonality techniques.

$$[\mathrm{Pic}^0_{K^\mathfrak{m}} : G]^{-1} \sum_{\substack{\theta\in\widehat{\mathrm{Pic}^0_{K^\mathfrak{m}}}\\ \theta|_G=\chi}} \theta(-[d^0(\mathfrak{p})]) = \begin{cases}\chi(-[d^0(\mathfrak{p})]) & \text{if } [d^0(\mathfrak{p})]\in G\\ 0 & \text{otherwise}\end{cases} \tag{165}$$

Putting $\overline{\mathcal{P}} = \{\mathfrak{p} \text{ prime ideal of } K \mid \mathcal{N}(\mathfrak{p}) \leq B \text{ and } \mathfrak{p} \nmid \mathfrak{m}\}$ (note the absence of the class condition on $\mathfrak{p}$), we thus have, for any fixed $\chi \in \widehat{G}$, using Equation (165),

$$\lambda_\chi = \frac{1}{|\mathcal{P}|}\sum_{\mathfrak{p}\in\mathcal{P}}\chi(-[d^0(\mathfrak{p})]) = \frac{1}{|\mathcal{P}|\cdot[\mathrm{Pic}^0_{K^\mathfrak{m}} : G]}\sum_{\mathfrak{p}\in\overline{\mathcal{P}}}\sum_{\substack{\theta\in\widehat{\mathrm{Pic}^0_{K^\mathfrak{m}}}\\ \theta|_G=\chi}}\theta(-[d^0(\mathfrak{p})]) \tag{166}$$

$$= \frac{|\overline{\mathcal{P}}|}{|\mathcal{P}|}\cdot\frac{1}{[\mathrm{Pic}^0_{K^\mathfrak{m}} : G]}\sum_{\substack{\theta\in\widehat{\mathrm{Pic}^0_{K^\mathfrak{m}}}\\ \theta|_G=\chi}}\frac{1}{|\overline{\mathcal{P}}|}\sum_{\mathfrak{p}\in\overline{\mathcal{P}}}\theta(-[d^0(\mathfrak{p})]) \tag{167}$$

Assuming the Extended Riemann Hypothesis (as formulated in [44, §5.7]), one can use results from analytic number theory [44, Theorem 5.15] to derive the following asymptotic bound for non-constant characters $\theta \in \widehat{\mathrm{Pic}^0_{K^\mathfrak{m}}}$ (see [14, Section 3.3]).

$$\frac{1}{|\overline{\mathcal{P}}|}\sum_{\mathfrak{p}\in\overline{\mathcal{P}}}\theta(-[d^0(\mathfrak{p})]) = O\left(\frac{\log(B)\log(B^n\cdot|\Delta_K|\cdot\mathcal{N}(\mathfrak{m})\cdot\mathfrak{q}_\infty(\theta))}{B^{1/2}}\right), \tag{168}$$

where $\mathfrak{q}_\infty(\chi)$ is the infinite part of the analytic conductor of the character χ (cf. [44, Eq. (5.6)]), The proof of this result uses the Abel summation formula and is almost the same as in [14, Section 3.3] or [78, Corollary 2.3.5] (see also [12, Section 4.3.3]), with the sole difference that the analytic conductor gets an extra factor $\mathcal{N}(\mathfrak{m}) = \mathcal{N}(\mathfrak{m}_0)\cdot 2^{|\mathfrak{m}_\mathbb{R}|}$.

By applying Equation (166) for $\chi = 1 \in \widehat{G}$, the trivial character, one can get an upper bound on the fraction $|\overline{\mathcal{P}}|/|\mathcal{P}|$. By the fact that $\lambda_1 = 1$, we obtain, from Equation (168)

$$\frac{|\mathcal{P}|\cdot[\mathrm{Pic}^0_{K^\mathfrak{m}} : G]}{|\overline{\mathcal{P}}|} = 1 + O\left(\frac{[\mathrm{Pic}^0_{K^\mathfrak{m}} : G]\cdot\log(B)\log(B^n\cdot|\Delta_K|\cdot\mathcal{N}(\mathfrak{m})\cdot\mathfrak{q}_\infty(1))}{B^{1/2}}\right). \tag{169}$$

Hence, choosing B such that the value of the big-O of Equation (169) is strictly bounded by a half[43], we obtain

$$\frac{|\overline{\mathcal{P}}|}{|\mathcal{P}|} \cdot \frac{1}{[\mathrm{Pic}^0_{K^{\mathfrak{m}}} : G]} \leq 2 \tag{170}$$

Combining Equations (167), (168) and (170), we obtain that the eigen value λ_χ for non-constant characters $\chi \in \widehat{G}$ under the Hecke operator $\mathcal{H}$ satisfies

$$\lambda_\chi = O\left(\frac{[\mathrm{Pic}^0_{K^{\mathfrak{m}}} : G] \cdot \log(B)\log(B^n \cdot |\Delta_K| \cdot \mathcal{N}(\mathfrak{m}) \cdot \mathfrak{q}_\infty(\chi))}{B^{1/2}}\right), \tag{171}$$

The infinite analytic conductor $\mathfrak{q}_\infty$

The infinite analytic conductor $\mathfrak{q}_\infty(\chi) \in \mathbb{R}_{>0}$ is a number quantifying the amount of oscillation of the character $\chi \in \widehat{G}$. By restricting a character $\chi \in \widehat{G}$ to the ray unit torus[44] $T^{\mathfrak{m}} = H/\operatorname{Log}(\mathcal{O}_K^\times \cap K^{\mathfrak{m},1}) \subseteq G$, one gets a character on $T^{\mathfrak{m}}$, which can be uniquely associated with a dual lattice point $\ell^* \in \operatorname{Log}(\mathcal{O}^\times_{K^{\mathfrak{m}},1})^\vee$, where $\operatorname{Log}(\mathcal{O}^\times_{K^{\mathfrak{m}},1}) = \operatorname{Log}(\mathcal{O}_K^\times \cap K^{\mathfrak{m},1})$. The infinite part of the analytic conductor is in that case equal to[45] (see [44, Equation (5.3), (5.4), (5.7)] and [57, Equation (3.3.6), (3.3.12)] or [14, Section 3.4] or [12, Section 4.3.4])

$$\mathfrak{q}_\infty(\chi) = \prod_{\nu \text{ real}} (3 + |2\pi\ell^*_\nu|) \prod_{\nu \text{ complex}} (3 + |2\pi\ell^*_\nu|)\big(3 + |i2\pi\ell^*_\nu + 1|\big) \tag{172}$$

Here, the components of the dual lattice point in the ray logarithmic unit lattice $\operatorname{Log}(\mathcal{O}^\times_{K^{\mathfrak{m}},1})^\vee$ are parametrized by the places ν. As a result, by applying the geometric and arithmetic inequality for vector norms, we obtain the following bound on the infinite analytic conductor.

$$\mathfrak{q}_\infty(\chi) \leq \left(4 + 4\pi \|\ell^*\| / \sqrt{n}\right)^n,$$

where $\ell^* \in \operatorname{Log}(\mathcal{O}^\times_{K^{\mathfrak{m}},1})^\vee$ is the unique dual logarithmic ray unit lattice point associated with the character $\chi|_{T^{\mathfrak{m}}} \in \widehat{T^{\mathfrak{m}}}$.

Fourier analysis of the Gaussian

Since the initial distribution of the random walk is a Gaussian $g_s : H \to \mathbb{R}, x \mapsto e^{-\pi\|x\|^2/s^2}$ over the hyperplane $H = \log K^0_{\mathbb{R}}$, we are particularly interested in the behavior of the distribution resulting from applying the Hecke operator N times, i.e., $\mathcal{H}^N(g_s)$. By standard Fourier computations, one can prove that periodized

[43] Notice that, in order the bound Equation (171) to be non-trivial (i.e., smaller than 1), we already need B to satisfy this condition, so it does not add an extra requirement on B. In Equation (168), the analytic conductor satisfies $\mathfrak{q}_\infty(\theta) \leq 4^n$ for extensions θ of the unit character on G, as $T^{\mathfrak{m}} \subseteq G$. This follows from the definition of the analytic conductor [44, Eq. (5.7)] (denoted $\mathfrak{q}(f) = \mathfrak{q}(f, 0)$ there), and the fact that the local parameters (denoted κ_j in [44]) are zero for characters trivial on $T^{\mathfrak{m}}$.

[44] Due to the fact that $[\mathrm{Pic}^0_{K^{\mathfrak{m}}} : G]$ is finite and the ray unit torus satisfies $T^{\mathfrak{m}} \subseteq \mathrm{Pic}^0_{K^{\mathfrak{m}}}$, it is also included in G.

[45] This definition of the infinite part of the analytic conductor is slightly different to that of Kowalski & Iwaniec [44, Equation (5.7)] (with $s = 1$) which is due to the difference in definitions of the real and the complex L-functions (see also [12, Remark 4.13]). This difference is solved by putting $L_{\mathbb{R}}(s)L_{\mathbb{R}}(s+1) = L_{\mathbb{C}}(s)$, (see [58, Chapter 7, Proposition 4.3(iv)]).

Gaussian function[46] $s^{-\mathfrak{r}} g_s|^{T^{\mathfrak{m}}} \in L^2(T^{\mathfrak{m}})$ satisfies

$$s^{-\mathfrak{r}} g_s|^{T^{\mathfrak{m}}} = \sum_{\ell^* \in \mathrm{Log}(\mathcal{O}^{\times}_{K^{\mathfrak{m}},1})^{\vee}} a_{\ell^*} \chi_{\ell^*} \tag{173}$$

where $a_{\ell^*} = \frac{1}{\mathrm{Vol}(T^{\mathfrak{m}})} g_{1/s}(\ell^*)$, where $\mathrm{Log}(\mathcal{O}^{\times}_{K^{\mathfrak{m}},1})^{\vee}$ is the dual lattice of the log ray unit lattice $\mathrm{Log}(\mathcal{O}^{\times}_{K^{\mathfrak{m}},1})$, and where $\chi_{\ell^*}(x) = e^{-2\pi i \langle x, \ell^* \rangle}$ is a function on $T^{\mathfrak{m}}$, i.e., $\chi_{\ell^*} \in \widehat{T^{\mathfrak{m}}}$.

By standard character arguments, one can simply write each $\chi_{\ell^*} \in \widehat{G}$ (which is zero everywhere except on $T^{\mathfrak{m}} \subseteq G$) as an average of all characters in $\widehat{G}$ that restrict to χ_{ℓ^*} on $T^{\mathfrak{m}}$. This results in the following identity (see also [14, §3.6]), where $\chi' \in \widehat{G}$ ranges over the characters of $G \subseteq \mathrm{Pic}^0_{K^{\mathfrak{m}}}$.

$$s^{-\mathfrak{r}} g_s|^{T^{\mathfrak{m}}} = \frac{1}{|G|} \sum_{\chi_{\ell^*} \in \widehat{T^{\mathfrak{m}}}} g_{1/s}(\ell^*) \sum_{\chi'|_{T^{\mathfrak{m}}} = \chi_{\ell^*}} \chi'. \tag{174}$$

Splitting up the sum

By splitting up the sum of Equation (174) into a 'unit part', a 'low frequency part' where $\|\ell^*\| < r$ and a 'high frequency part' where $\|\ell^*\| \geq r$, we obtain the following decomposition, for any $r > 0$, where $\chi' \in \widehat{G}$ ranges over the characters of $G \subseteq \mathrm{Pic}^0_{K^{\mathfrak{m}}}$.

$$\begin{aligned} |G| \cdot s^{-\mathfrak{r}} \cdot g_s|^{T^{\mathfrak{m}}} = {} & \underbrace{\mathbf{1}_G}_{\text{Unit character}} \\ & + \underbrace{\sum_{\substack{\chi_{\ell^*} \in \widehat{T^{\mathfrak{m}}} \\ \|\ell^*\| < r}} g_{1/s}(\ell^*) \sum_{\substack{\chi'|_{T^{\mathfrak{m}}} = \chi_{\ell^*} \\ \chi' \neq \mathbf{1}}} \chi'}_{\text{Low frequency characters}} + \underbrace{\sum_{\substack{\chi_{\ell^*} \in \widehat{T^{\mathfrak{m}}} \\ \|\ell^*\| \geq r}} g_{1/s}(\ell^*) \sum_{\chi'|_{T^{\mathfrak{m}}} = \chi_{\ell^*}} \chi'}_{\text{High frequency characters}}, \end{aligned}$$

Now, applying the Hecke operator $\mathcal{H}^N$ times to this equation, and taking the absolute value of $|G| \cdot \mathcal{H}^N(g_s|^{T^{\mathfrak{m}}}) - \mathbf{1}_G$, we obtain, by the Pythagorean theorem (using that $\|\chi' \overline{\chi'}\|_2^2 = |G|$),

$$\left\| |G| \cdot \mathcal{H}^N(s^{-\mathfrak{r}} g_s|^{T^{\mathfrak{m}}}) - \mathbf{1}_G \right\|^2 = \underbrace{|G| \sum_{\substack{\chi_{\ell^*} \in \widehat{T^{\mathfrak{m}}} \\ \|\ell^*\| < r}} g^2_{1/s}(\ell^*) \sum_{\substack{\chi'|_{T^{\mathfrak{m}}} = \chi_{\ell^*} \\ \chi' \neq \mathbf{1}}} |\lambda_{\chi'}|^{2N}}_{\text{Low frequency}} \tag{175}$$

$$+ \underbrace{|G| \sum_{\substack{\chi_{\ell^*} \in \widehat{T^{\mathfrak{m}}} \\ \|\ell^*\| \geq r}} g^2_{1/s}(\ell^*) \sum_{\chi'|_{T^{\mathfrak{m}}} = \chi_{\ell^*}} |\lambda_{\chi'}|^{2N}}_{\text{High frequency}}. \tag{176}$$

[46] The periodization is defined as follows: $g_s|^{T^{\mathfrak{m}}}(x) := \sum_{\ell \in \mathrm{Log}(\mathcal{O}^{\times}_{K^{\mathfrak{m}},1})} g_s(x + \ell)$.

We will bound the parts Equation (175) and Equation (176) separately. For the latter, we have

$$|G| \sum_{\substack{\chi_{\ell^*} \in \widehat{T^{\mathfrak{m}}} \\ \|\ell^*\| \geq r}} g_{1/s}^2(\ell^*) \sum_{\chi'|_{T^{\mathfrak{m}}} = \chi_{\ell^*}} |\lambda_{\chi'}|^{2N} \leq |G| \cdot |G/T^{\mathfrak{m}}| \sum_{\substack{\ell^* \in \mathrm{Log}(\mathcal{O}_{K^{\mathfrak{m}},1}^{\times})^{\vee} \\ \|\ell^*\| \geq r}} g_{\frac{1}{\sqrt{2}s}}(\ell^*) \tag{177}$$

$$\leq |G| \cdot |G/T^{\mathfrak{m}}| \cdot \beta_{\sqrt{2}rs}^{(\mathbb{r})} \cdot g_{\frac{1}{\sqrt{2}s}}(\mathrm{Log}(\mathcal{O}_{K^{\mathfrak{m}},1}^{\times})^{\vee}), \tag{178}$$

where the last inequality follows from Banaszczyk's tail bound [6] and the assumption that $rs > \sqrt{\mathbb{r}/(4\pi)}$, where $\beta_z^{(n)} := \left(\frac{2\pi e z^2}{n}\right)^{n/2} e^{-\pi z^2}$, for which holds $\beta_t^{(n)} \leq e^{-t^2}$ for all $t \geq \sqrt{n}$ (see also [14, §3.6]).

To bound the share of the low-frequency characters, we use the bound from Equation (171), $|\lambda_{\chi'}| \leq O\left(\frac{[\mathrm{Pic}_{K^{\mathfrak{m}}}^0 : G] \cdot \log(B) \log(B^n \cdot |\Delta_K| \cdot \mathcal{N}(\mathfrak{m}) \cdot \mathfrak{q}_\infty(\chi'))}{B^{1/2}}\right)$. Since these characters have a 'low frequency', their analytic conductor $\mathfrak{q}_\infty(\chi')$ is bounded, namely, $\mathfrak{q}_\infty(\chi') \leq (4 + 2\pi r/\sqrt{n})^n$ for any $\chi' \in \widehat{G}$ such that $\chi'|_{T^{\mathfrak{m}}} = \chi_{\ell^*}$ for some $\ell^* \in \mathrm{Log}(\mathcal{O}_{K^{\mathfrak{m}},1}^{\times})^{\vee}$ with $\|\ell^*\| < r$. Therefore,

$$|\lambda_{\chi'}| \leq c = O\left(\frac{[\mathrm{Pic}_{K^{\mathfrak{m}}}^0 : G] \cdot \log(B) \log(B^n \cdot |\Delta_K| \cdot \mathcal{N}(\mathfrak{m}) \cdot (4 + 4\pi r/\sqrt{n})^n)}{B^{1/2}}\right).$$

We then obtain

$$|G| \sum_{\|\ell^*\| \leq r} g_{1/s}^2(\ell^*) \underbrace{\sum_{\chi'|_{T^{\mathfrak{m}}} = \chi_{\ell^*}} |\lambda_{\chi'}|^{2N}}_{\leq |G/T^{\mathfrak{m}}| \cdot c^{2N}} \leq |G| \cdot |G/T^{\mathfrak{m}}| \cdot c^{2N} \cdot g_{\frac{1}{\sqrt{2}s}}(\mathrm{Log}(\mathcal{O}_{K^{\mathfrak{m}},1}^{\times})^{\vee}) \tag{179}$$

We obtain the following bound by combining Equations (177) and (179), dividing by $|G|$ (which accounts to dividing by $|G|^2$ under the square 2-norm) and using the identity $|G| = |G/T^{\mathfrak{m}}| \cdot \mathrm{Vol}(T^{\mathfrak{m}})$. Assuming the Extended Riemann Hypothesis for Hecke L-functions (e.g., [44, §5.7]), and for all $r, s > 0$ with $rs > \sqrt{\frac{\mathbb{r}}{4\pi}}$, we have

$$\left\| \mathcal{H}^N(s^{-n} g_s|^{T^{\mathfrak{m}}}) - \frac{1}{|G|} \mathbf{1}_G \right\|^2 \leq \frac{g_{\frac{1}{\sqrt{2}s}}(\mathrm{Log}(\mathcal{O}_{K^{\mathfrak{m}},1}^{\times})^{\vee})}{\mathrm{Vol}(T^{\mathfrak{m}})} \left(c^{2N} + \beta_{\sqrt{2}rs}^{(\mathbb{r})}\right) \tag{180}$$

with $c = O\left(\frac{[\mathrm{Pic}_{K^{\mathfrak{m}}}^0 : G] \cdot \log(B) \log(B^n \cdot |\Delta_K| \cdot \mathcal{N}(\mathfrak{m}) \cdot (4+4\pi r/\sqrt{n})^n)}{B^{1/2}}\right)$.

Tuning parameters

Let $1 > \varepsilon > 0$, $s > 0$ and $k \in \mathbb{R}_{>0}$ be given. We have $g_{\frac{1}{\sqrt{2}s}}(\mathrm{Log}(\mathcal{O}_{K^{\mathfrak{m}},1}^{\times})^{\vee}) \leq g_{1/\tilde{s}}(\mathrm{Log}(\mathcal{O}_{K^{\mathfrak{m}},1}^{\times})^{\vee}) \leq 2\,\mathrm{Vol}(\mathrm{Log}(\mathcal{O}_{K^{\mathfrak{m}},1}^{\times}))/\tilde{s}^{\mathbb{r}}$ by smoothing arguments (see Lemma 2.20). Using that inequality and Hölder's inequality (i.e., $\|f \cdot 1\|_1 \leq \|f\|_2 \|1\|_2$), noting that

$\|1_G\|_2^2 = |G|$, we obtain, for each $r > \sqrt{\mathbb{r}}/(\sqrt{2}s)$,

$$\|\mathcal{W}_G(B,N,s) - \mathcal{U}(G)\|_1^2 \tag{181}$$

$$\leq |G| \cdot \left\| \mathcal{H}^N(s^{-\mathbb{r}} g_s|^{T^{\mathfrak{m}}}) - \frac{1}{|G|}\mathbf{1}_G \right\|_2^2 \tag{182}$$

$$\leq |G/T^{\mathfrak{m}}| \cdot g_{\frac{1}{\sqrt{2}s}}(\mathrm{Log}(\mathcal{O}_{K^{\mathfrak{m}},1}^{\times})^{\vee}) \cdot (c^{2N} + \beta_{\sqrt{2}rs}^{(\mathbb{r})}) \tag{183}$$

$$\leq 2 \cdot |G| \cdot \tilde{s}^{-\mathbb{r}} \cdot (c^{2N} + \beta_{\sqrt{2}rs}^{(\mathbb{r})}) \tag{184}$$

Here, $c = O\left(\frac{[\mathrm{Pic}^0_{K^{\mathfrak{m}}}:G]\cdot\log(B)\log(B^n\cdot|\Delta_K|\cdot\mathcal{N}(\mathfrak{m})\cdot(4+2\pi r/\sqrt{n})^n)}{B^{1/2}}\right)$. We proceed by bounding the two summands in Equation (184) separately.

- By putting[47] r equal to
$$\frac{1}{\sqrt{2}s} \cdot \max\left(\sqrt{\mathbb{r}}, \sqrt{2 + \mathbb{r}\log(1/\tilde{s}) + 2\log(1/\varepsilon) + \log|G|}\right)$$
we deduce that $2 \cdot |G| \cdot \tilde{s}^{-\mathbb{r}} \cdot \beta_{\sqrt{2}rs}^{(\mathbb{r})} \leq \varepsilon^2/2$.
- Subsequently, choose[48] a $B = \tilde{O}([\mathrm{Pic}^0_{K^{\mathfrak{m}}} : G]^2 \cdot n^{2k}[\log(|\Delta_K|\mathcal{N}(\mathfrak{m}))^2 + n^2\log(r)^2])$, i.e.,
$$B = \tilde{O}\Big([\mathrm{Pic}^0_{K^{\mathfrak{m}}} : G]^2 \cdot n^{2k} \cdot \big[\log(|\Delta_K|\mathcal{N}(\mathfrak{m}))^2 + n^2\log(1/\tilde{s})^2 + n^2\log(\log(1/\varepsilon))^2\big]\Big)$$
such that $c \leq 1/n^k$, where $c = O\left(\frac{[\mathrm{Pic}^0_{K^{\mathfrak{m}}}:G]\cdot\log(B)\log(B^n\cdot|\Delta_K|\cdot\mathcal{N}(\mathfrak{m})\cdot(4+4\pi r/\sqrt{n})^n)}{B^{1/2}}\right)$. Lastly, taking any integer $N \geq \frac{1}{2k\log n}\cdot(\mathbb{r}\cdot\log(1/\tilde{s}) + 2\log(1/\varepsilon) + \log|G| + 2)$ and noting that $c^{\frac{1}{k\log n}} \leq 1/e$, we deduce that $2 \cdot |G| \cdot \tilde{s}^{-\mathbb{r}} \cdot c^{2N} \leq \varepsilon^2/2$.

Combining, we can bound the right-hand side of Equation (184) by ε^2. Taking square roots gives the final result. □

A.2. Sampling of random prime ideals

In this section we show how to sample prime ideals $\mathfrak{p}$ that are coprime with $\mathfrak{m}$ and whose class $[d^0(\mathfrak{p})]$ fall into $G \subseteq \mathrm{Pic}^0_{K^{\mathfrak{m}}}$, using an *oracle* that can check whether an ideal $\mathfrak{a}$ satisfies $[d^0(\mathfrak{p})] \in G$ or not. The algorithm doing this is a slight generalization of the algorithm sampling prime ideals in [14, Lemma 2.2]. Before giving this algorithm (together with its properties), we first need the following result on the number of prime ideals that are coprime with $\mathfrak{m}$.

Lemma A.2.1 (ERH). *Let $\mathfrak{m} \subseteq \mathcal{O}_K$ be an ideal modulus and denote*

$$\pi_K^{\mathfrak{m}}(x) = |\{\mathfrak{p} \in \mathcal{I}_K^{\mathfrak{m}} \mid \mathfrak{p} \textit{ prime and } \mathcal{N}(\mathfrak{p}) \leq x\}|$$

for the number of prime ideals not dividing $\mathfrak{m}$ and having norm bounded by $x \in \mathbb{R}$. Let $\omega(\mathfrak{m})$ denote the number of different prime ideal divisors of $\mathfrak{m}$.

[47] We use the bound $\beta_\alpha^{(\mathbb{r})} \leq e^{-\alpha^2}$ for $\alpha \geq \sqrt{\mathbb{r}}$

[48] In this bound on B one would expect an additional $\log\log|G| \leq \log\log|\mathrm{Pic}^0_{K^{\mathfrak{m}}}|$. But as it is bounded by $\log(\log(|\Delta_K|\mathcal{N}(\mathfrak{m})))$ (see Lemma 5.1), it can be put in the hidden polylogarithmic factors.

Then, there exists $x_0 \in \widetilde{O}(\log^2 |\Delta_K| + \omega(\mathfrak{m}))$ such that for all $x \geq x_0$ we have

$$\pi_K^{\mathfrak{m}}(x) \geq \frac{x}{4 \log x}.$$

Proof. Denote $\pi_K(x) = |\{\mathfrak{p} \in \mathcal{I}_K \mid \mathfrak{p} \text{ prime and } \mathcal{N}(\mathfrak{p}) \leq x\}|$, i.e., whenever $\mathfrak{m} = \mathcal{O}_K$. We will prove the statement for this specific case first. By simplifying an explicit result of Grenié and Molteni [37, Corollary 1.4], we obtain, under the Extended Riemann Hypothesis[49],

$$\left| \pi_K(x) - \pi_K(3) - \int_3^x \frac{du}{\log u} \right| \leq \sqrt{x}[6 \log |\Delta_K| + 4n \log x + 14].$$

Therefore, we have

$$\begin{aligned} \pi_K(x) &\geq \int_3^x \frac{du}{\log u} - \sqrt{x}[6 \log |\Delta_K| + 4n \log x + 14] \\ &\geq \frac{x}{\log x} - \sqrt{x} \log(x)[6 \log |\Delta_K| + 4n + 14] \\ &= \frac{x}{\log x} \left(1 - \frac{\log(x)^2 (6 \log |\Delta_K| + 4n + 14)}{\sqrt{x}} \right) \geq \frac{x}{2 \log x} \end{aligned}$$

where the first inequality follows from omitting $\pi_K(3)$ and the second inequality from $\int_3^x \frac{du}{\log u} \geq \frac{x}{\log x}$ and from the assumption that $x > x_0$, where $x_0 \in \widetilde{O}(\log^2 |\Delta_K|)$ such that $\frac{\log(x_0)^2(6 \log |\Delta_K| + 4n + 14)}{\sqrt{x_0}} < 1/2$.

For the general case of $\mathfrak{m} \neq \mathcal{O}_K$, we need to avoid $\mathfrak{m}$; so writing $\omega(\mathfrak{m})$ for the number of different prime ideals dividing $\mathfrak{m}$, we obtain

$$\pi_K^{\mathfrak{m}}(x) \geq \pi_K(x) - \omega(\mathfrak{m}) \geq \frac{x}{2 \log x} \left(1 - \frac{2 \cdot \omega(\mathfrak{m}) \cdot \log x}{x} \right) \geq \frac{x}{4 \log x}.$$

Where the last inequality can be deduced from the fact that $x > x_0$ and where $x_0 = \widetilde{O}(\log^2 |\Delta_K| + \omega(\mathfrak{m}))$ is chosen such that $\frac{2 \cdot \omega(\mathfrak{m}) \cdot \log x_0}{x_0} < 1/2$. □

The following lemma is a slight generalization of [14, Lemma 2.2], where we demand the Arakelov ray class of the sampled prime ideals to be lying in G, some subgroup of $\mathrm{Pic}^0_{K^{\mathfrak{m}}}$.

Lemma 5.4 (Uniform sampling of prime ideals, ERH)**.** *Let a basis of $\mathcal{O}_K$ be known, and let $\mathfrak{m}_0 \subseteq \mathcal{O}_K$ be a modulus. Let $G \subseteq \mathrm{Pic}^0_{K^{\mathfrak{m}}}$ a finite-index subgroup and let $\mathbf{O}_G$ be an oracle that on input an ideal $\mathfrak{c}$ returns whether $[d^0(\mathfrak{c})] \in G$ or not. Let $\mathcal{P}_B = \{\mathfrak{p}$ prime ideal of $K \mid \mathcal{N}(\mathfrak{p}) \leq B, \mathfrak{p} \nmid \mathfrak{m}_0$ and $[d^0(\mathfrak{p})] \in G\}$.*

There exists a bound

$$B_0 = \widetilde{O}\Big([\mathrm{Pic}^0_{K^{\mathfrak{m}}} : G]^2 \cdot \big[n^2 (\log\log(1/\varepsilon))^2 + n^2 (\log(1/\tilde{s}))^2 + (\log(|\Delta_K| \mathcal{N}(\mathfrak{m})))^2 \big] \Big)$$

such that for all $B \geq B_0$, one can sample uniformly from $\mathcal{P}_B$ in expected time $O([\mathrm{Pic}^0_{K^{\mathfrak{m}}} : G] \cdot n^3 \log^2 B)$ and using $O([\mathrm{Pic}^0_{K^{\mathfrak{m}}} : G] \cdot n \log B)$ queries to $\mathbf{O}_G$.

Proof. The algorithm can be described as follows. Sample a uniform integer from $[0, B]$ and check if it is prime (if not, output 'failure'). If it is, name the prime p, factor $p\mathcal{O}_K$ into prime ideals of $\mathcal{O}_K$ and list the different prime factors $\{\mathfrak{p}_1, \ldots, \mathfrak{p}_k\}$

[49] In the paper of Grenié and Molteni [37, Corollary 1.4], only the Dedekind zeta function $\zeta_K(s) = \sum_{\mathfrak{a}} \mathcal{N}(\mathfrak{a})^{-s}$ needs to satisfy the condition that all of its non-trivial zeroes lie at the vertical line $\Re(s) = 1/2$.

that have norm bounded by B, do not divide $\mathfrak{m}_0$ and satisfy $[d^0(\mathfrak{p}_j)] \in G$. If this set is empty, output 'failure'; otherwise, choose one $\mathfrak{p}_j$ uniformly at random in $\{\mathfrak{p}_1, \ldots, \mathfrak{p}_k\}$ and output it with probability k/n. Otherwise, output 'failure'.

Let $\mathfrak{q} \in \mathcal{P}_B$ be arbitrary, and let $p = \mathfrak{q} \cap \mathbb{Z}$ the prime 'below' $\mathfrak{q}$. Then the probability of sampling this $\mathfrak{q}$ equals $\frac{1}{nB}$, namely, $\frac{1}{n}$ times the probability of sampling p. So, the probability of sampling successfully (that is, no failure) equals

$$\frac{|\mathcal{P}_B|}{nB} \geq \frac{1}{8 \cdot [\mathrm{Pic}^0_{K^{\mathfrak{m}}} : G] \cdot n \log B}$$

since, by Equation (169), denoting $\overline{\mathcal{P}}_B = \{\mathfrak{p} \text{ prime ideal of K } \mid \mathcal{N}(\mathfrak{p}) \leq B, \mathfrak{p} \nmid \mathfrak{m}_0\}$,

$$\frac{|\mathcal{P}_B|}{|\overline{\mathcal{P}}_B|} \geq \frac{1}{2 \cdot [\mathrm{Pic}^0_{K^{\mathfrak{m}}} : G]},$$

and by Lemma A.2.1, we have $|\overline{\mathcal{P}}_B| \geq \frac{B}{4 \log B}$ (by adequately increasing B_0 so that $B_0 \geq x_0$ from Lemma A.2.1).

The most costly part of the algorithm is the factorization of a prime $p \leq B$ in $\mathcal{O}_K$. This can be performed using Kummer-Dedekind algorithm, which essentially amounts to factoring a degree n polynomial modulo p. Using Shoup's algorithm [71] (which has complexity $O(n^2 + n \log p)$ [77, §4.1]) and the fact that the algorithm needs to repeat $[\mathrm{Pic}^0_{K^{\mathfrak{m}}} : G] \cdot n \log B$ times to get constant success probability, yields the complexity claim and the query complexity for $\mathbf{O}_G$. □

A.3. Upper bound on the nth successive minimum of $\mathcal{O}_K$ by Bhargava et al.

The following theorem and its proof is a copy of that of Bhargava et al. [8, Theorem 3.1], with the difference that it is applied to the infinity norm and has explicit constants everywhere. A similar result, but also without explicit constants, can be found in an article by Peikert and Rosen [62, Lemma 5.4].

Theorem A.3.1 (Bhargava et al. [8])**.** *Let K be any number field of degree n and let $\mathcal{O}_K$ be its ring of integers. Let $\mathcal{O}_K \subseteq K_{\mathbb{R}}$ have the structure of a lattice via the Minkowski embedding (see Section 2), and denote $\lambda_j^{\infty}(\mathcal{O}_K)$ for the j-th successive minimum with respect to the infinity norm in $K_{\mathbb{R}}$. Then*

$$\lambda_n^{\infty}(\mathcal{O}_K) \leq |\Delta_K|^{1/n}.$$

Proof. Let $\alpha_j \in \mathcal{O}_K$ attain the successive minima for the infinity norm $\lambda_j^{\infty}(\mathcal{O}_K)$ for $j \in \{1, \ldots, n\}$, with $\alpha_1 = 1$. For any element $\beta \in \mathcal{O}_K$, we write $\beta = \sum_{j=1}^{n} [\beta]_j \alpha_j$, i.e., $[\beta]_j$ are the coordinates of β with respect to $(\alpha_1, \ldots, \alpha_n)$.

For $2 \leq k, \ell \leq n-1$ consider the $(n-2) \times (n-2)$-matrix $C = ([\alpha_k \alpha_\ell]_n)$, i.e., the matrix consisting of the coordinates of $\alpha_k \alpha_\ell$ with respect to α_n. We will show at the end of this proof that this is a non-degenerate matrix, implying that there are no zero rows or columns. In other words, there exists a permutation $\pi : \{2, \ldots, n-1\} \to \{2, \ldots, n-1\}$ such that $[\alpha_k \alpha_{\pi(k)}]_n \neq 0$ for all $k \in \{2, \ldots, n-1\}$.

So, the product $\alpha_k \alpha_{\pi(k)} \in \mathcal{O}_K$ extends $\{\alpha_1, \ldots, \alpha_{n-1}\}$ to a n-dimensional lattice; therefore we have $\|\alpha_k\|_\infty \|\alpha_{\pi(k)}\|_\infty \geq \|\alpha_k \alpha_{\pi(k)}\|_\infty \geq \lambda_n^{\infty}(\mathcal{O}_K)$. Taking products over all $k \in \{2, \ldots, n-1\}$ we obtain

$$\prod_{k=2}^{n-1} \|\alpha_k\|_\infty^2 = \prod_{k=2}^{n-1} \|\alpha_k\|_\infty \|\alpha_{\pi(k)}\|_\infty \geq \left(\lambda_n^{\infty}(\mathcal{O}_K)\right)^{n-2}.$$

Multiplying above equation by $\|\alpha_1\|_\infty^2 = 1$ and $\|\alpha_n\|_\infty^2 = \lambda_n^\infty(\mathcal{O}_K)^2$, and using Minkowski's second inequality[50] [21, Chap. VIII, Thm. 5] $\prod_{k=1}^n \lambda_k^\infty(\Lambda) \leq \mathrm{Vol}(\Lambda)$, we obtain

$$|\Delta_K| \geq \prod_{k=1}^n \|\alpha_k\|_\infty^2 \geq \big(\lambda_n^\infty(\mathcal{O}_K)\big)^n.$$

It remains to prove that $C = ([\alpha_k \alpha_\ell]_n)$ is non-degenerate. Suppose it is not, and there exists d_ℓ for $\ell \in \{2, \ldots, n-1\}$ (not all zero) such that

$$\Big[\sum_{\ell=2}^{n-1} d_\ell \alpha_k \alpha_\ell\Big]_n = \sum_{\ell=2}^{n-1} d_\ell [\alpha_k \alpha_\ell]_n = 0 \text{ for all } k \in \{2, \ldots, n-1\}$$

Writing $\beta = \sum_{\ell=2}^{n-1} d_\ell \alpha_\ell$, this means that, for all $k \in \{1, \ldots, n-1\}$, $\alpha_k \beta$ lies in the span of $(\alpha_1, \ldots, \alpha_{n-1})$. In other words, $L = \mathbb{Q}\alpha_1 + \ldots + \mathbb{Q}\alpha_{n-1}$ is $\mathbb{Q}(\beta)$-invariant, i.e., a $\mathbb{Q}(\beta)$-vector (strict) subspace of K. That is, $\dim_{\mathbb{Q}(\beta)}(L) \leq \dim_{\mathbb{Q}(\beta)}(K) - 1$. But then

$$\begin{aligned} n - 1 = \dim_{\mathbb{Q}}(L) &= \dim_{\mathbb{Q}(\beta)}(L) \cdot [\mathbb{Q}(\beta) : \mathbb{Q}] \\ &\leq (\dim_{\mathbb{Q}(\beta)}(K) - 1) \cdot [\mathbb{Q}(\beta) : \mathbb{Q}] = n - [\mathbb{Q}(\beta) : \mathbb{Q}], \end{aligned}$$

yielding $[\mathbb{Q}(\beta) : \mathbb{Q}] = 1$, i.e., $\beta \in \mathbb{Q}$, which is impossible by the fact that $\beta = \sum_{\ell=2}^{n-1} d_\ell \alpha_\ell$ is assumed to be non-zero and has no $\alpha_1 = 1$ part.

We conclude that C is non-degenerate, which finishes the proof. □

A.4. Upper bound on a defining polynomial of a number field

The following two lemmas show together that for any number field K there exists a polynomial $f \in \mathbb{Z}[x]$ such that $K = \mathbb{Q}[x]/(f(x))$ and so that $\mathrm{size}(f) \leq O(\log^2 |\Delta_K| \cdot \log^2(\log |\Delta_K|))$. That is, each field has a 'small' defining polynomial. The first lemma is just a variant of the primitive element theorem [50, V.§4, Theorem 4.6], whereas the second lemma uses this primitive element theorem inductively to bound such a primitive element.

Lemma A.4.1 (Primitive Element Theorem)**.** *Let $K = \mathbb{Q}(\alpha, \beta)$ be a separable extension. Then there exist an integer $\lambda \in \{0, 1, \ldots, d_\alpha \cdot d_\beta\}$ such that $K = \mathbb{Q}(\alpha + \lambda\beta)$, where $d_\alpha = [\mathbb{Q}(\alpha) : \mathbb{Q}]$ and $d_\beta = [\mathbb{Q}(\beta) : \mathbb{Q}]$.*

Proof. We write $\gamma = \alpha + \lambda\beta$ with $\lambda \in \mathbb{Q}$. Our aim is to show that the number of $\lambda \in \mathbb{Q}$ for which $\mathbb{Q}(\alpha + \lambda\beta) \neq \mathbb{Q}(\alpha, \beta)$ is bounded above by $d_\alpha \cdot d_\beta$, hence, for a certain $\lambda \in \{0, 1, \ldots, d_\alpha \cdot d_\beta\}$, it holds that $\mathbb{Q}(\alpha + \lambda\beta) = \mathbb{Q}(\alpha, \beta) = K$; by the pigeon-hole principle.

Write f_α, f_β for the minimum polynomial of α respectively β. Write $\gamma = \alpha + \lambda\beta$. Then we have $f_\alpha(\gamma - \lambda\beta) = 0$, hence β is a zero of the polynomial $g = f_\alpha(\gamma - \lambda x) \in \mathbb{Q}(\gamma)[x]$.

[50] Note that Minkowski's second inequality as stated in [21, Chap. VIII, Thm. 5] only states that $\prod_{k=1}^n \lambda_k^\infty(\Lambda) \cdot \mathrm{Vol}(B) \leq 2^n \cdot \mathrm{Vol}(\Lambda)$, where $B = \{x \in K_\mathbb{R} \mid \|x\|_\infty \leq 1\}$ (B is a ball of dimension n, living in a real vector space of dimension $2n$ since our lattice is not full rank). To compute the volume of B, observe that B consists in $n_\mathbb{R}$ orthogonal copies of $B_\mathbb{R} = \{x \in \mathbb{C} \mid x \in \mathbb{R}, |x| \leq 1\}$ and $n_\mathbb{C}$ orthogonal copies of $B_\mathbb{C} = \{(x, \bar{x}) \in \mathbb{C}^2 \mid |x| \leq 1\}$. One can check that $\mathrm{Vol}(B_\mathbb{R}) = 2$ and $\mathrm{Vol}(B_\mathbb{C}) = 2\pi$, leading to $\mathrm{Vol}(B) = 2^{r_\mathbb{R} + r_\mathbb{C}} \cdot \pi^{r_\mathbb{C}}$. Minkowski's second theorem then implies that $\prod_{k=1}^n \lambda_k^\infty(\Lambda) \leq \mathrm{Vol}(\Lambda)$ as desired.

So, the minimum polynomial h^γ_β of β over $\mathbb{Q}(\gamma)$ divides $g \in \mathbb{Q}(\gamma)[x]$, but also $f_\beta \in \mathbb{Q}[x]$. So,

$$h^\gamma_\beta \mid \gcd(g, f_\beta).$$

Suppose $\beta \notin \mathbb{Q}(\gamma)$, then the minimum polynomial h^γ_β has degree $d \geq 2$, and divides both g and f_β. By separability, the polynomial h^γ_β has d distinct roots (in the algebraic closure of $\mathbb{Q}(\gamma)$), so in particular, there exists $\beta_2 \neq \beta$ such that $h^\gamma_\beta(\beta_2) = 0$. This β_2 then satisfies $f_\beta(\beta_2) = g(\beta_2) = 0$. Putting $\alpha_2 = \gamma - \lambda\beta_2$ and using the definition of g, one can see that α_2 is a root of f_α.

Finally, combining the equations $\alpha = \gamma - \lambda\beta$ and $\alpha_2 = \gamma - \lambda\beta_2$ and using the fact that $\beta \neq \beta_2$, we obtain that $\lambda = \frac{\alpha - \alpha_2}{\beta_2 - \beta}$, where β_2 is a root of f_β and α_2 is a root of f_α. There are at most $\deg(f_\alpha) \cdot \deg(f_\beta) = d_\alpha d_\beta$ possible λ with this shape. Hence, $\mathbb{Q}(\gamma) = \mathbb{Q}(\alpha + \lambda\beta) = \mathbb{Q}(\alpha, \beta)$ for all but $d_\alpha \cdot d_\beta$ values $\lambda \in \mathbb{Q}$. This was what was remained to be proven. □

Lemma A.4.2. *Let K be a number field of degree n. Then there exists a monic integral irreducible polynomial $g = \sum_{i=0}^n g_i x^i \in \mathbb{Z}[x]$ such that $K \simeq \mathbb{Q}[x]/g(x)$ with*

$$\max_i \log |g_i| = O(\log |\Delta_K| \cdot \log\log |\Delta_K|)$$

and hence

$$\mathrm{size}(g) := \sum_i \log |g_i| = O(\log^2 |\Delta_K| \cdot \log\log |\Delta_K|).$$

Proof. Let $\mathcal{O}_K$ be the ring of integers of K. By definition of the last minimum of a lattice, we know that there exist n linearly independent elements $\beta_1, \dots, \beta_n \in \mathcal{O}_K$ satisfying $\|\beta_i\|_\infty \leq \lambda_i^{(\infty)}(\mathcal{O}_K) \leq |\Delta_K|^{1/n}$, where we used Theorem A.3.1 for the last inequality.

Since $\beta_1, \dots, \beta_n$ span a n-dimensional space, then $K = \mathbb{Q}(\beta_1, \dots, \beta_n)$. Applying the primitive element theorem (in the shape of Lemma A.4.1) inductively, we can then deduce that there exists

$$\beta = \sum_{i=1}^n n_i \beta_i$$

with $n_i \in \{0, 1, \dots, n^2\}$ for all i, and such that $K = \mathbb{Q}(\beta)$. Indeed, assume by induction that $K_j := \mathbb{Q}(\beta_1, \dots, \beta_j) = \mathbb{Q}(\gamma_j)$ for some $\gamma_j = \sum_{i=1}^j n_i \beta_i$ as above. Then by Lemma A.4.1, it holds that $K_{j+1} = \mathbb{Q}(\gamma_j, \beta_{j+1}) = \mathbb{Q}(\gamma_j + n_{j+1}\beta_{j+1})$ for some $n_{j+1} \leq n^2$ (since all the intermediate fields appearing have degree at most n).

Therefore,

$$\|\beta\|_\infty \leq n^3 |\Delta_K|^{1/n}.$$

By definition of $\|\beta\|_\infty$, this implies that for all embeddings $\sigma_j : K \to \mathbb{C}$, it holds that $|\sigma_j(\beta)| \leq n^3 |\Delta_K|^{1/n}$.

Let $g = \sum_{i=0}^n g_i x^i$ be the minimal polynomial of β. Since $\beta \in \mathcal{O}_K$ (because the β_i's are in $\mathcal{O}_K$ and the n_i's are in $\mathbb{Z}$), then g has integer coefficients. Moreover, we have

$$\begin{aligned}|g_j| &= \Big|\sum_{\substack{J\subseteq\{1,\ldots,n\}\\|J|=n-j}}\prod_{i\in J}\sigma_i(\beta)\Big| \leq \binom{n}{j}(n^3|\Delta_K|^{1/n})^{n-j}\\ &\leq \sum_{j=0}^{n}\binom{n}{j}(n^3|\Delta_K|^{1/n})^{n-j} = (1+n^3|\Delta_K|^{1/n})^n\\ &= n^{3n}|\Delta_K|\left(1+\frac{1}{n^3|\Delta_K|^{1/n}}\right)^n \leq e\cdot n^{3n}|\Delta_K|,\end{aligned}$$

since $n^3|\Delta_K|^{1/n} \geq n$.

Hence, $\max_i \log|g_i| = O(n\log(n)+\log|\Delta_K|)$. Using the fact that $n = O(\log|\Delta_K|)$ concludes the proof. □