Quantumly Computing S-unit Groups in Quantified Polynomial Time and Space

Koen de Boer and Joël Felderhoff[1]

[1] King's College London

Abstract. We present a novel analysis of a quantum algorithm computing the S-unit group for a number field from Eisenträger et al. [EHKS14a] and Biasse and Song [BS16]. We prove that this quantum algorithm runs within polynomial time, where we explicitly quantify the polynomials of the quantum gate and memory complexity (under GRH). We do so by carefully analyzing an implementation of a Continuous Hidden Subgroup Problem (CHSP) oracle function whose period is the (logarithm of the) S-unit group, and provide it to a CHSP-solving algorithm as in [dBDF20].
Our analysis is novel due to minimizing the use of the quantum memory-inefficient LLL-reduction, by resorting to strategically chosen precomputations of approximations of high powers of prime ideals. Additionally, we provide a novel analysis of a quantum algorithm computing a discrete Gaussian superposition analogue of the GPV algorithm by Gentry et al. [GPV08]. Lastly, we include a full and rigorous numerical analysis of all parts of the oracle-function computing algorithm, allowing to use fixed-point precision arithmetic and thus to precisely quantify the run-time and memory.

1 Introduction

Quantum algorithms in number theory

Out of all quantum algorithms, Shor's algorithm [Sho94] for factoring is often considered as the most ground-breaking. It is distinct in having theoretical implications, due to its large complexity gap between classical and quantum computing, as well as practical implications in the field of cryptography. Indeed, this algorithm can be seen as the early cause of the current standardization of quantum-resistant cryptographic schemes [NIS25].

The polynomial-time factoring algorithm from Shor [Sho94] can be divided into two parts: a period finding quantum algorithm solving the hidden subgroup problem [ME99] in commutative groups by means of a quantum Fourier transform [Cop02,Sho94,NC10], and the computation of an actual function of which one wants to find the period, which is often referred to as the *oracle function*. In the case of Shor's algorithm, the period of this oracle function found by the period finding quantum algorithm then gives information about the factorization of a given number.

This framework of quantumly finding the period of a certain computable oracle function has then been generalized to solve other number theoretic problems in polynomial time, like the discrete logarithm problem [Sho97], Pell's equation [Hal07] and even the problem of computing unit groups [EHKS14a] and class groups [BS16].

The oracle function

Shor's algorithm for factoring has quantum gate complexity $\tilde{O}(\ln^3 N)$ [Sho94] and may be further improved to $\tilde{O}(\ln^2 N)$ by fast multiplication techniques [HH21]. Remarkably, the bottleneck of Shor's algorithm is not the quantum period-finding algorithm, but rather the computation of the oracle function, which consists of taking powers of a number g modulo the to be factored number N [Sho94].

Such a precise quantum gate complexity and clear identification of the bottlenecks of the algorithms solving *other* number theoretic problems, like computing the unit group by Eisenträger, Hallgren, Kitaev and Song [EHKS14a], and computing the class group by Biasse and Song [BS16], is currently difficult to achieve. The cause for this is that no exponent of the polynomial run-time is presented in these works (including the submitted version of [EHKS14a] to STOC [EHKS14b]), and no clear division in complexity is made between the period-finding part and the oracle-part. Additionally, much of the intricacies involving handling real number operations by rounding or using finite precision numbers is overlooked. Efforts in the assessment of the complexity of the oracle function was made in concurrent work [BS25v2] but that work is not yet peer-reviewed to our knowledge and contains some gaps (see the end of this introduction for specifics).

In [dBDF20], a refined analysis of the *quantum period finding part* of these algorithms was presented, leading to a quantitative measure of the exponent of the polynomial complexity in terms of quantum gates, qubits and the number of queries to the oracle function. The actual computation of the oracle function is not treated in this refined analysis. This analysis was specialized in [BP23] to the case of computing unit groups in cyclotomic number fields, for which they prove that the number of qubits required is significantly lower.

S-units in cryptanalysis

In cryptanalysis studying quantum-safe protocols it is common to assume that an adversary has access to a quantum computer. In particular, by the works of Eisenträger et al. [EHKS14a] and Biasse and Song [BS16], in many papers, the S-unit group (e.g., class group and unit group) is assumed to be known by the adversary. The adversary knowing the S-unit group is in particulary relevant in the case of the attacks on IdealSVP and PIP [CDPR16,CDW17,PHS19]. Additionally, the knowledge of the S-unit group is also assumed in the module lattice reduction of [LPSW19].

Quantum computation being a scarce future resource, the precise complexity of computing S-unit groups is highly relevant for these cryptographic applications.

Our work

In this work, we present a refined analysis of (variants of) the *oracle functions* proposed by [EHKS14a,BS16], allowing for a quantified exponent of the polynomial complexity of the full algorithms. This allows as well to compare the weight on the running time of the period-finding part with that of the oracle-computing part. We present a variant of the CHSP oracle described in [BS16], and analyze this oracle function in Section 3 to prove that it satisfies the hypotheses of [dBDF20], needed to compute the S-unit group of a number field. Those hypotheses consist of the Lipschitz and separativity properties of the oracle function. We then present a description of the quantum algorithm, with an explicit qubit and gate complexity analysis focusing on the polynomial dependence on $d = [K : \mathbb{Q}]$, $s = |S|$ and $\ln(|\Delta_K|)$.

Our main result can be summarized by the following slightly informal statements.

Theorem 1.1 (Assuming GRH, informal). *There exists a quantum algorithm that computes an adequate Continuous Hidden Subgroup oracle for the S-unit group of a number field K of degree d and discriminant Δ_K, using*

$$O\left(d^4 \cdot (d+|S|)^{7.5+o(1)} \cdot \left(\ln\left(|\Delta_K|^{1/d}\right)+d\right)^{3+o(1)}\right) \text{ quantum memory}$$

and

$$O\left(d^7 \cdot (d+|S|)^{17.5+o(1)} \cdot \left(\ln\left(|\Delta_K|^{1/d}\right)+d\right)^{7+o(1)}\right) \text{ quantum gates,}$$

and polynomially many classical operations in the size of the input.

Corollary 1.1 (Assuming GRH). *There exists a quantum algorithm that computes (with constant success probability) a basis of the S-unit group of a number field K of degree d and discriminant Δ_K, using*

$$O\left(d^4 \cdot (d+|S|)^{7.5+o(1)} \cdot \left(\ln\left(|\Delta_K|^{1/d}\right)+d\right)^{3+o(1)}\right) \text{ quantum memory}$$

and

$$O\left(d^7 \cdot (d+|S|)^{18.5+o(1)} \cdot \left(\ln\left(|\Delta_K|^{1/d}\right)+d\right)^{8+o(1)}\right) \text{ quantum gates,}$$

and polynomially many classical operations in the size of the input.

This theorem and corollary are derived from Theorem 5.1 and Corollary 5.1, with the conservative estimates $\mathrm{LLLMem}(n, b) = O(n^4 b^{3/2})$, $\mathrm{LLLGates}(n, b) = O(n^7 \cdot b^{3.5})$ [TS19, Eq 7, 8 and p.15] and $\omega = 2.81$ for the polynomial complexity exponent for matrix multiplication[1] [Str69]. Retrieving a compact representation of all S-units from an approximate basis happens by the procedure in Appendix G. The GRH assumption is here only to give a polynomial-time procedure to complete a set S into a set S' that generates the class group of K. If S already generates Cl_K, the previous results hold without GRH.

Impact on cryptography. An attack algorithm's impact on cryptography is partially dictated by the time scale on which future quantum machines are actually able to run this algorithm. Usually, the minimum *size* of quantum machine is taken as proxy variable for this time scale, where the size is measured by the complexity or costs of the algorithms such a machine can execute.

What is considered a costly algorithm depends largely on the computational cost model, which can include gate count, memory count, circuit depth etc. (for a discussion, see [BBTV24, Appendix F]). As an example, the NIST proposes different [NIS16, Sec. 4.A.5] values for the maximal depth of a quantum circuit for attacks.

Our work lacks the analysis of the circuit depth of the oracle algorithm, which we left for future work. This makes the impact of this work's algorithm on cryptography hard to assess.

To nevertheless gain an idea of the size of a quantum machine required to run the algorithm presented in this work, we provide an example of the memory cost in the specific case of quadratic fields $K = \mathbb{Q}(\sqrt{D})$ (for square free D). Assuming GRH, putting $|S| = O(\log(\Delta_K)^2))$ and using the estimate of Corollary 1.1, the upper bound on the memory cost can be (roughly) lower bounded by $\log(D)^{15}$. This polynomial's large exponent makes that a practical quantum implementation of the oracle function (and hence an associated attack) seem currently unlikely to us.

Technical overview. Let K be a number field of degree $d = d_{\mathbb{R}} + 2d_{\mathbb{C}}$, let $S = \{\mathfrak{p}_1, \ldots, \mathfrak{p}_s\}$ be a factor basis and $\mathcal{O}_{K,S}^{\times}$ be the S-unit group of K. We define the (possibly infinite) distance between two ideals I_1 and I_2 as

$$\min_{x \in K_{\mathbb{R}}} \{\| \mathrm{LogEx}(x)\|_2 \ \mid \ x \cdot I_1 = I_2\},$$

i.e., the minimum ℓ_2 norm of the total logarithmic embedding of those $x \in K_{\mathbb{R}}$ that satisfy $x \cdot I_1 = I_2$. As was noted in [BS16], $\mathcal{O}_{K,S}^{\times}$ can be represented as the kernel of the morphism $f : (e_i, x) \mapsto c_{x,(e_i)} \cdot \mathrm{Exp}(x) \cdot \prod_i \mathfrak{p}_i^{e_i}$ where $(e_i) \in \mathbb{Z}^s$, $x \in \mathrm{Log}(K_{\mathbb{R}})$, and where $c_{x,(e_i)} \in \mathbb{R}_{>0}$ is a normalization constant. This fact allows to

[1] Though there has been much research and improvement in the complexity exponents of matrix multiplication (e.g., [CW81,WXXZ]), they do not improve the asymptotic running time of the specific algorithm in this paper. This is why we chose to use the well-known exponent of Strassen [Str69].

use a Continuous Hidden Subgroup Problem (CHSP) solver [EHKS14a,dBDF20] to efficiently find a basis of $\mathcal{O}_{K,S}^{\times}$, under some conditions on f. Those conditions are:

1. f must be Lipschitz (it should not vary too quickly);
2. f must be separative (it should not vary too slowly);
3. f must be efficiently implementable as a quantum algorithm.

The realization of the function $F_{R,\sigma}$ (see Definition 3.1) we build in this work is, as previously introduced in [EHKS14a,BS16], a "quantum fingerprint" of an ideal lattice $I = (x) \cdot \prod_i \mathfrak{p}_i^{e_i}$. This fingerprint is constructed as a tail-cut Gaussian superposition of the elements of I, encoded using the straddle encoding. In contrast to [BS16], we encode the extended logarithm of the elements of $K_{\mathbb{R}}$ instead of their complex embeddings.

Lipschitzianity. To our knowledge, the previous literature did not anticipate the tail-cut Gaussian superposition to be non-Lipschitz, though in generally it is not; as it can even be non-continuous. This problem is not raised by [BS25v2] neither. In order to overcome this problem, we introduce the notion of almost-Lipschitz continuity (i.e., functions that satisfy $\|f(x) - f(y)\| \leq a \cdot \|x - y\| + \varepsilon$) and show in Appendix H that any periodic and almost Lipschitz function is close (in maximum norm) to some (fully) Lipschitz function with the same periodicity. We thus prove that the oracle function f that we construct is almost-Lipschitz (Lemma 3.3).

Separativity. The separativity condition for an oracle function f can be phrased as follows: if x and y are two points that are far enough modulo the period of f, then $\langle f(x)|f(y)\rangle$ should be small. This property is proven in Section 3.4. We prove it, similar as in [EHKS14a,BS25v2], in two steps. First, we show that if the encodings of two ideals I_1 and I_2 have an inner product that is close to 1, then there exists a small distortion $g \in K_{\mathbb{R}}^0$ such that $g \cdot I_1$ shares a sublattice with I_2. As a second step, we then prove that if two different ideals share a sublattice, then the inner product of their Gaussian encoding cannot be too large, hence $g \cdot I_1 = I_2$.

Efficient Quantum Implementation. In previous analysis of the S-unit computation algorithm [BS16,Ng24], the quantum complexity of the CHSP oracle is either ignored or extended from the its classical description, which does not try to minimize the amount of quantum memory used. In order to fill this gap, we use several techniques. Since the CHSP-solving algorithm queries the oracle function on dyadic rationals, we can strategically precompute integral approximations of high powers of the prime ideals in S to minimize ideal multiplication and LLL-reduction. These integral approximations allow to use the Hermite Normal Form on products of those, which makes that we perform ideal multiplications reversely and hence minimize the quantum memory cost of the algorithm. Then a single quantum-LLL reduction of this HNF is performed (instead of one per single multiplication as in [BS25v2], which is much more). We aim to minimize

the number of quantum-LLL calls due to their heavy usage of quantum memory. This LLL-reduced basis allows us to compute a (tail-cut and punctured) Gaussian superposition, by using a quantum analogue of GPV [GPV08]. For this, we compute the $\boldsymbol{R}$, the R-factor of its QR-factorization using a quantum adaptation of Householder's algorithm.

Analysis of a quantum analogue of GPV. The (classical) GPV algorithm [GPV08] allows to sample from a discrete Gaussian over a lattice, with the special property that the parameter σ can be chosen almost as small as the largest Gram-Schmidt norm of the given basis of the lattice. In this paper, we propose an analysis (new, to our knowledge) of the algorithm constructing an approximate discrete Gaussian quantum superposition, by combining the original techniques of the classical algorithm by Gentry, Peikert and Vaikuntanathan [GPV08] and an algorithm of Kitaev and Webb [KW09] that computes a Gaussian superposition over $\mathbb{Z}$ efficiently. Much of the proof that the approximation is sufficiently close consists of numerical analysis. The algorithm described in Section 6 does not differ significantly from previous descriptions of Quantum GPV [Bra18,LMZ23], but our precise error and complexity analysis is new to our knowledge.

Parameters of the algorithm. In Section 5, we collect all relations and constraints related to the parameters of our algorithm, in order to fix values that minimize the exponents of the polynomials defining the quantum gate and memory costs.

Relation between our work and [BS25v2]. This paper heavily builds on an extended abstract published in SODA in 2016 [BS16]. Part of the motivation for the current work was that the *full version* of this extended abstract [BS16] has not been made public for at least 9 years after the initial SODA publication.

We contacted the authors 2 months prior to the first publication of the present work to ePrint and its submission to a conference, to let them know we plan to cite the full version of their extended abstract as independent of ours (since we could not access it, as it was not publicly available). They sent us a complete version and released it on arXiv [BS25v1] shortly before the submission of this work to a conference (respectively 3 and 1 day before the first release of this work).

In [BS25v1], an error in [BS25v1, Theorem 2] made that the running time of the oracle had polynomial growth in V, which is the upper bound on the Euclidean norm of its input. The analysis of [dBDF20] gives that this V has linear growth in $1/\tau$ (see Theorem K.1), where τ is the absolute error on the output basis of the CHSP solver. This dependency makes the oracle defined in [BS25v1] exponential in quantum space and gate count when one wants τ exponentially small.

After seeing this error, we added Appendix G to this paper to prove that the error term τ indeed needed to be exponentially small to compute a basis of the S-unit lattice, updated the preprint version of this work and emailed the authors of [BS25v1] about the error we found. Their work was updated 2

days later [BS25v2] fixing the error, and providing a more precise gate/memory-count for their oracle [BS25v2] in order to be compared to our work. As a side note, the running time of the CHSP oracle described in [BS25v2, Theorem 5] is not contained in [EHKS14b, Theorem 6.2] and seems to come from [dBDF20, Theorem 3.3].

We think the work [BS25v2] as concurrent to ours, and we will highlight the differences and similarities between our approach and [BS25v2] during the course of the paper. The purpose of this paper is not to overly optimize the overall and fine-grained complexity of the S-unit quantum algorithm (although we kept it polynomial), but rather to provide a modular and precise proof which will make further study of this line of work more accessible. We think that the precise and peer-reviewed analysis of the different parts of the S-unit oracle (gate and memory count, quasi-Lipchitznianity, error propagation) will be useful for future development of quantum lattice fingerprinting and possible improvements of the oracle.

Acknowledgments. During most of the time this research took place, Joël Felderhoff was working at the ENS de Lyon, LIP (UMR 5668) and funded by the Direction Générale de l'Armement (Pôle de Recherche CYBER). Joël Felderhoff's work is supported by UKRI grant EP/Y02432X/1. The authors want to thank Léo Ducas, Guillaume Hanrot, Vincent Lefèvre and Gilles Villard.

2 Preliminaries

2.1 Notations

We denote $\mathbb{N}, \mathbb{Z}, \mathbb{Q}, \mathbb{R}, \mathbb{C}$ for respectively the natural numbers, the integers, the rational numbers, the real numbers and the complex numbers. Vectors are considered column vectors and are, like matrices, denoted with bold letters. The inner product, except for the bra-ket notation, is denoted with a simple dot $\cdot$. We denote ln for the natural logarithm and $\log_2$ for the base-two logarithm. We make use of the Landau notation $O(\cdot)$, in which, in this work, the hidden constant is always absolute, meaning that it does not depend on any other quantity. For any $a, b \in \mathbb{R}$, we denote $[\![a, b]\!] := [a, b] \cap \mathbb{Z}$ and $[\![a]\!] := [\![0, a]\!]$ to simplify notation involving integer intervals. The proofs of the preliminary results presented in this section are available in Appendix B.

2.2 Quantum algorithms, complexity and analysis

In this paper, $\mathcal{H}$ will denote a Hilbert space, $\mathcal{S}(\mathcal{H})$ the unit sphere of $\mathcal{H}$ (with respect to the induced norm on $\mathcal{H}$). When $\mathcal{H}$ is defined as a qubit-space, $\mathcal{S}(\mathcal{H})$ is the set of all quantum states of these qubits. For any function $f : X \to \mathcal{H}$ and $k \in \mathbb{Z}_{>0}$ we define the function $f^{\otimes k} : X \to \mathcal{H}^{\otimes k}$ as $f^{\otimes k}(x) = f(x) \otimes \ldots \otimes f(x)$ for any $x \in X$. We make use of the following result, which describes how a classical algorithm can be simulated on a quantum computer.

Theorem 2.1 ([NC10, Sec. 1.4.1]). *Suppose $\mathcal{A}$ is an algorithm that uses N NAND operations, operates on M bits of memory and outputs O bits. Then it can be simulated by a quantum circuit $C_{\mathcal{A}}$, using $2N$ Toffoli gates, in the following way:*

$$C_{\mathcal{A}} \cdot |m\rangle|o\rangle|0^N\rangle := |m\rangle|o \oplus \mathcal{A}(m)\rangle|0^N\rangle \quad \text{for all } m \in \{0,1\}^M \text{ and } o \in \{0,1\}^O$$

2.3 Lipschitz continuity and separativity

Throughout this work, unless otherwise specified, we denote $\|\cdot\|$ as the Euclidean norm of vectors, and we denote $\|\boldsymbol{A}\| = \sup_{\|\boldsymbol{x}\|=1} \|\boldsymbol{A} \cdot \boldsymbol{x}\|$, as the spectral norm on matrices.

Definition 2.1. *Let (X, δ) denote a metric space. Let $f : X \to \mathcal{H}$. The function f is said to be (a, α)-almost Lipschitz continuous if, for all $x, y \in X$, $\|f(x) - f(y)\| < a \cdot \delta(x, y) + \alpha$.*

In this work, we study a function described by a tail-cut Gaussian superposition, which cannot be Lipschitz continuous since it is not even continuous in the ordinary sense. We tackle this issue by proving that this function is *almost*-Lipschitz continuous and is therefore close to a Lipschitz continuous function with respect to the uniform norm, see Theorem H.1. This contrasts with [BS25v2], where the functions are proven to be close to the infinite Gaussian sum.

Definition 2.2. *Let (X, δ) be a metric space, $\nu > 0$ and $\varepsilon \in [0, 1)$. We say that $f : X \to \mathcal{S}(\mathcal{H})$ is (ν, ε)-separative over (X, δ) if for any $x, y \in X$ it holds that*

$$\delta(x, y) > \nu \Rightarrow \ |\langle f(x)|f(y)\rangle| \leq \varepsilon$$

We say that f is ν-totally separative over (X, δ) if it is $(\nu, 0)$-separative.

This property is named "pseudo-injectivity" in [BS25v2]. Note that the separativity property depends heavily on the distance notion δ on X. In the use-case of this paper, $f : \mathbb{R}^n \to \mathcal{S}(\mathcal{H})$ will be a Λ-periodic function for some full-rank lattice $\Lambda \subset \mathbb{R}^n$. As can be readily verified, such a function cannot be separative for any parameter. But if we instead consider the induced function $\tilde{f}$ over $\mathbb{R}^n/\Lambda$, it might be (ν, ε)-separative over the quotient space $\mathbb{R}^n/\Lambda$ (equipped with the quotient metric).

2.4 Lattices

A lattice $\mathcal{L}$ is a discrete additive subgroup of $\mathbb{R}^n$. For any lattice $\mathcal{L}$ of $\mathbb{R}^n$, there exists a set of $\mathbb{R}$-linearly independent vectors $(\boldsymbol{b}_i)_{1 \leq i \leq m}$ such that $\mathcal{L} = \sum_{i=1}^{m} \mathbb{Z} \cdot \boldsymbol{b}_i$. The family $(\boldsymbol{b}_i)_{1 \leq i \leq m}$ is called a basis of $\mathcal{L}$, and it is unique up to the (right) action of $GL_m(\mathbb{Z})$. In this work we only consider full rank lattices, i.e., $m = n$. We define the ith minimum of $\mathcal{L}$ as $\lambda_i(\mathcal{L}) := \inf\{r > 0, \dim(\mathrm{span}(\mathcal{L} \cap B(0, r))) = i\}$ and the (co-)volume of $\mathcal{L}$ as $\mathrm{Vol}(\mathcal{L}) := \mathrm{Vol}(\mathbb{R}^n/\mathcal{L})$. If $\mathcal{L}$ is full-rank and has a basis $\boldsymbol{B}$, then we have $\mathrm{Vol}(\mathcal{L}) = |\det(\boldsymbol{B})|$. Note that the volume of $\mathcal{L}$ does not depend on the basis of $\mathcal{L}$.

Lemma 2.1 (Consequence of [MG02, Cor.7.2]). *Let $\mathcal{L}_1, \mathcal{L}_2$ two (full rank) lattices of $\mathbb{R}^d$. Let $R \geq \sqrt{d} \cdot \max(\lambda_d(\mathcal{L}_1), \lambda_d(\mathcal{L}_2))$. Then*

$$\mathcal{L}_1 = \mathcal{L}_2 \quad \textit{if and only if} \quad \mathcal{L}_1 \cap B(0,R) = \mathcal{L}_2 \cap B(0,R).$$

Gaussian measures The Gaussian function with parameter $\sigma > 0$ is defined by $\rho_\sigma(x) = \exp\left(-\pi \|x\|^2 / \sigma^2\right)$, and satisfies $\int_{\mathbb{R}^n} \rho_\sigma(x) dx = \sigma^n$. For any discrete $X \subset \mathbb{R}^n$, we define $\rho_\sigma(X) := \sum_{x \in X} \rho_\sigma(x)$. For $\varepsilon > 0$ and a lattice $\mathcal{L} \subset \mathbb{R}^d$, the smoothing parameter of $\mathcal{L}$ is defined as

$$\eta_\varepsilon(\mathcal{L}) := \inf\left\{\sigma > 0, \rho_{1/\sigma}(\mathcal{L}^\star \setminus \{0\}) \leq \varepsilon\right\},$$

where $\mathcal{L}^\star$ is the dual lattice of $\mathcal{L}$.

2.5 Number theory

We succinctly present the notions of algebraic number theory used in this work. The interested reader is referred to [Neu13]. In the current paper, K denotes a number field of degree d, $\mathcal{O}_K$ its ring of integers and Δ_K its discriminant. Let $\Phi = (\sigma_i)_{1 \leq i \leq d} : K \to \mathbb{C}^d$ be the canonical embedding of K. We order the embeddings σ_i in such a way that $\sigma_i(K) \subset \mathbb{R}$ for $1 \leq i \leq d_\mathbb{R}$ are the real embeddings, and $\sigma_{d_\mathbb{R}+i} = \overline{\sigma_{d_\mathbb{R}+d_\mathbb{C}+i}}$ for any $1 \leq i \leq d_\mathbb{C}$ are the complex embeddings. Note that $d = d_\mathbb{R} + 2d_\mathbb{C}$.

We denote $K_\mathbb{R} = K \otimes \mathbb{R}$ and extend the canonical embedding $\Phi = (\sigma_i)_i$ to the entirety of $K_\mathbb{R}$ by extension of scalars. This canonical embedding endows $K_\mathbb{R}$ with the structure of a normed space, where the norm of $x \in K_\mathbb{R}$ is $\|\Phi(x)\|$. For sake of brevity, we will write $\|x\| = \|\Phi(x)\|$ and $\|x\|_\infty = \|\Phi(x)\|_\infty$.

We define the algebraic norm of any element of $K_\mathbb{R}$ as $\mathcal{N}(x) = \left|\prod_{i=1}^d \sigma_i(x)\right|$, which extends the algebraic norm of K to $K_\mathbb{R}$ in the sense that $\mathcal{N}(x) = \left|\mathcal{N}_{K/\mathbb{Q}}(x)\right|$ for $x \in K$. We denote by $K_\mathbb{R}^\times := \{x \in K_\mathbb{R} \mid \sigma_i(x) \neq 0 \text{ for all } i\}$ the group of invertible elements of $K_\mathbb{R}$ and by $K_\mathbb{R}^0 \subset K_\mathbb{R}^\times$ the group of norm-1 elements of $K_\mathbb{R}^\times$. The unit group $\mathcal{O}_K^\times \subset \mathcal{O}_K$ is the set of integers of K whose inverses are also integers. We define the logarithmic embedding of $K_\mathbb{R}$,

$$\begin{aligned} \mathrm{Log} : (K_\mathbb{R}^\times, \cdot) &\longrightarrow (\mathbb{R}^{d_\mathbb{R}+d_\mathbb{C}}, +) \\ x &\longmapsto ((\ln(|\sigma_i(x)|))_{i=1,\ldots,d_\mathbb{R}}, (2\ln(|\sigma_{d_\mathbb{R}+i}(x)|))_{i=1,\ldots,d_\mathbb{C}}), \end{aligned}$$

which is a surjective group morphism. The logarithmic embedding is not injective, as its kernel is exactly the group of $x \in K_\mathbb{R}^\times$ such that $|\sigma_i(x)| = 1$ for all $i = 1, \ldots, d$ (in contains in particular the roots of unity of K). In order to turn Log into an injective group morphism, we introduce the *extended* logarithmic embedding.

Definition 2.3. *For any $z \in \mathbb{C} \setminus \{0\}$, recall that z can be written as $z = e^{i \arg(z)} \cdot |z|$ for $\arg(z) \in \mathbb{R}/(2\pi\mathbb{Z})$. If $z \in \mathbb{R} \setminus \{0\}$, we have $\arg(z) \in \{0, \pi\}$ depending on*

the sign of z. Let $\mathrm{Arg} : K_{\mathbb{R}}^{\times} \to \{0,\pi\}^{d_{\mathbb{R}}} \times (\mathbb{R}/(2\pi\mathbb{Z}))^{d_{\mathbb{C}}}$ *be the function defined by* $\arg(x) = (\arg(\sigma_i(x)))_{1\le i\le d_{\mathbb{R}}+d_{\mathbb{C}}}$. *We write* $\mathrm{Arg}_K = \mathrm{Arg}(K_{\mathbb{R}}^{\times})$ *and define the extended logarithmic embedding as follows:*

$$\begin{aligned} \mathrm{LogEx} : K_{\mathbb{R}}^{\times} &\longrightarrow \mathrm{Arg}_K \oplus \mathrm{Log}\, K_{\mathbb{R}} \\ x &\longmapsto (\mathrm{Arg}(x), \mathrm{Log}(x)) \end{aligned}$$

The function LogEx *is a group isomorphism, and we denote by* ExpEx *its inverse.*

Via the extended log embedding we can introduce a distance on $K_{\mathbb{R}}^{\times}$ by defining $\delta_{K_{\mathbb{R}}^{\times}}(x,y) := \delta(\mathrm{LogEx}(x), \mathrm{LogEx}(y))$ (where the latter distance is induced from the standard metric on $\mathbb{R}$ and $\mathbb{R}/(2\pi\mathbb{Z})$, and where we put the distance between 0 and π in $\{0,\pi\}$ to be ∞ in the case of real embeddings). This distance is translation invariant with respect to multiplication by $K_{\mathbb{R}}^{\times}$, i.e., for any $a,x,y \in K_{\mathbb{R}}^{\times}$, we have $\delta_{K_{\mathbb{R}}^{\times}}(ax,ay) = \delta_{K_{\mathbb{R}}^{\times}}(x,y)$. Additionally, if the sign of every real embedding of $x,y \in K_{\mathbb{R}}^{\times}$ coincides, we have $\delta_{K_{\mathbb{R}}^{\times}}(x,y) = \|\mathrm{LogEx}(x) - \mathrm{LogEx}(y)\|$.

Note that, since the definition of $\delta_{K_{\mathbb{R}}^{\times}}$ involves a discrete distance over $\{0,\pi\}$, it can take infinite values. For example, if $d_{\mathbb{R}} > 0$, we have $\delta_{K_{\mathbb{R}}^{\times}}(1,-1) = \infty$, a case in which $(K_{\mathbb{R}}^{\times}, \delta_{K_{\mathbb{R}}^{\times}})$ is a disconnected space.

2.6 Ideals

A fractional ideal of K is a non-zero discrete additive subgroup I of K that satisfies $\alpha \cdot I \subseteq I$ for all $\alpha \in \mathcal{O}_K$. A fractional ideal is integral if it is included in $\mathcal{O}_K$. A replete ideal is a subgroup of $K_{\mathbb{R}}$ of the form $x \cdot \mathfrak{a}$ for $\mathfrak{a}$ an integral ideal and $x \in K_{\mathbb{R}}^{\times}$.

The set of replete ideals forms a group: The product of two replete ideals is $I \cdot J := \left\{ \sum_{i=1}^{k} a_i \cdot b_i \mid k \in \mathbb{Z}_{>0}, a_i \in I, b_i \in J \right\}$ and the inverse of a replete ideal I is $I^{-1} = \{x \in K_{\mathbb{R}}, x \cdot I \subset \mathcal{O}_K\}$. The fractional ideals forms a group in the same fashion. The norm of a replete ideal is $\mathcal{N}(x \cdot \mathfrak{a}) := \mathcal{N}(x) \cdot \mathcal{N}(\mathfrak{a})$, which matches the usual ideal norm in the case of fractional ideals. A principal replete ideal is an ideal of the form $x \cdot \mathcal{O}_K$ with $x \in K_{\mathbb{R}}^{\times}$. Any fractional ideal I can be written in a unique way (up to ordering) as a finite product of prime ideals $I = \prod_{\mathfrak{p}} \mathfrak{p}^{v_{\mathfrak{p}}(I)}$ where $v_{\mathfrak{p}}(I) \in \mathbb{Z}$ is the $\mathfrak{p}$-adic valuation of I (if $x \in K$, we write $v_{\mathfrak{p}}(x) = v_{\mathfrak{p}}(x \cdot \mathcal{O}_K)$). The canonical embedding of any replete ideal is a full rank lattice in $\Phi(K_{\mathbb{R}})$; such a lattice is said to be an ideal lattice. In this work we identify replete ideals and their embeddings, and denote by IdLat_K the set of replete ideals of K and IdLat_K^0 the set of norm-1 replete ideals.

Definition 2.4. *We equip the set* IdLat_K^0 *with the distance*

$$\delta_{\mathrm{ideal}}(I,J) = \inf\{\|\mathrm{LogEx}(x)\| \mid x \in K_{\mathbb{R}} \text{ such that } x \cdot I = J\} \in \mathbb{R}_{\ge 0} \cup \{\infty\}.$$

Note that the distance between two ideals is finite if and only if the two ideals are in the same co-set of Cl_K. In [BS25v2], another distance is considered,

namely the "matrix distance", defined for any lattices $\mathcal{L}_1, \mathcal{L}_2$ as $\delta(\mathcal{L}_1, \mathcal{L}_2) = \inf\{\|\boldsymbol{M}\|, \mathcal{L}_1 = \exp(\boldsymbol{M}) \cdot \mathcal{L}_2\}$. This distance is more general, as it applies to all lattices, but in our case we choose to restrict it to "diagonal matrices" as it highlights the commutative nature of ideals, and leads to simpler computations.

S-units Let S be a finite set of prime ideals of K. We denote the set of fractional ideals generated by S as $\langle S \rangle$. The set of S-units is denoted $\mathcal{O}_{K,S}^{\times}$ and is the set of elements of K whose prime factorization only involves primes from S:

$$\mathcal{O}_{K,S}^{\times} = \{\alpha \in K \text{ such that } \alpha \cdot \mathcal{O}_K \in \langle S \rangle\},$$

Note that $\mathcal{O}_{K,\emptyset}^{\times} = \mathcal{O}_K^{\times}$, and $\mathcal{O}_{K,S}^{\times} \subset \mathcal{O}_{K,S'}^{\times}$ whenever $S \subseteq S'$.

Properties of ideal lattices Ideal lattices are not typical lattices. They have some properties that we present in the next lemmas. In the first of these lemmas is shown that ideal lattices have bounded smoothing parameters.

Lemma 2.2 ([PRSD17, Lemma 6.9]). *Let $I \in \mathrm{IdLat}_K$. It holds that $\eta_\varepsilon(I) \leq (\mathcal{N}(I) \cdot |\Delta_K|)^{1/d} \cdot \max\left(1, \sqrt{(\ln(1/\varepsilon))/d}\right)$.*

The successive minima of ideal lattices are constrained, unlike those of general lattices. This is shown in the next lemma.

Lemma 2.3 ([LPSW19, Le. 2.2], [dBDPW20, Le. 2.8] and [Boe22, Le. 2.22]). *For any $J \in \mathrm{IdLat}_K$ it holds that $\lambda_1(J) \geq \sqrt{d} \cdot \mathcal{N}(J)^{1/d}$, and $\lambda_d(J) \leq \sqrt{d} \cdot \lambda_d(\mathcal{O}_K) \cdot |\Delta_K|^{1/(2d)} \cdot \mathcal{N}(J)^{1/d}$, furthermore, $\lambda_d(\mathcal{O}_K) \leq \sqrt{d} \cdot |\Delta_K|^{1/d}$.*

For any $R > 0$ and $X \subseteq K_{\mathbb{R}}^{\times}$ we denote $X|_R = \{x \in X, \|x\| \leq R\}$.

Lemma 2.4. *Let $I \in \mathrm{IdLat}_K$ and $R \geq \mathcal{N}(I)^{1/d}$. Then for any $x \in I \setminus \{0\}$, $y \in I|_R \setminus \{0\}$ with $x \neq y$, it holds that $\delta_{K_{\mathbb{R}}^{\times}}(x, y) \geq \mathcal{N}(I)^{1/d}/2R$.*

Lemma 2.5. *Let $R \geq \sqrt{d}$, and let $I, J \in \mathrm{IdLat}_K^0$ such that there exists $x \in (I \cap J) \setminus \{0\}$ satisfying $\|x\|_\infty \leq R$. Then for any $(u, v) \in I|_R \times J|_R$ with $u \neq v$, we have $\delta_{K_{\mathbb{R}}^{\times}}(u, v) \geq 1/(2R^2 \cdot |\Delta_K|^{1/(2d)})$.*

In the following lemma, we show that the discrete Gaussian distribution over a lattice cannot have most of its weight on any strict sublattice.

Lemma 2.6 (see also [EHKS14b, Lemma E.5]). *Let $J \subsetneq I \in \mathrm{IdLat}_K$. Then, for any $\sigma \geq 3 \cdot d^{3/2} \cdot |\Delta_K|^{3/(2d)} \cdot \mathcal{N}(I)^{1/d}$ it holds that $\rho_\sigma(J)/\rho_\sigma(I) \leq 2/3$.*

Balancedness Multiplying an ideal I by some $x \in K_{\mathbb{R}}^{\times}$ changes the geometry of I. In order to measure the geometrical impact of the multiplication by such $x \in K_{\mathbb{R}}^{\times}$, we use the notion of *balancedness*.

Definition 2.5 ([FPSW23, Def. 2.4]). *Let $\eta \in \mathbb{R}_{>1}$. An element $x \in K_{\mathbb{R}}$ is said to be η-balanced if, for any $i \in [\![1, d_{\mathbb{R}} + d_{\mathbb{C}}]\!]$, it holds that $|\sigma_i(x)| \in |\mathcal{N}(x)|^{1/d} \cdot [1/\eta, \eta]$.*

Lemma 2.7. *Let $\eta_1, \eta_2 > 1$ and $x, y \in K_{\mathbb{R}}$ such that x is η_1-balanced and y is η_2-balanced. Then $x \cdot y$ is $(\eta_1 \cdot \eta_2)$-balanced.*

Lemma 2.8 (Derived of [FPSW23, Alg C.1]). *There exists a polynomial time algorithm* SampleBalanced *that, on input an ideal I with basis $\boldsymbol{B}_I$ of $\Phi(I) \subset \mathbb{R}^d$ and a balancedness parameter $\eta > 1$, outputs $x \in I \setminus \{0\}$ such that*

(i) $\|x\| \leq \frac{\eta}{\eta - 1} \cdot d^{3/2} \cdot \max_{1 \leq i \leq d} \|\boldsymbol{b}_i^{\star}\|$, *where $(\boldsymbol{b}_i^{\star})_{1 \leq i \leq d}$ is the Gram-Schmidt basis of $\boldsymbol{B}_I$,*

(ii) $\left|\sigma_i(x)/\mathcal{N}(x)^{1/d} - 1\right| \in [1 - \eta^{-1}, \eta - 1]$ *for all $i \in [\![1, d]\!]$. In particular, x is η-balanced.*

2.7 Computation of the S-units and the CHSP oracle

Recall that the purpose of this paper is to compute a fundamental set of S units of K represented in efficient compact representation. In order to compute such a fundamental set (see Appendix G), we need a sufficiently good approximation of a basis of the log-S-unit lattice. This section is about how to acquire such a good approximation of that log-S-unit basis.

Computing such a basis approximation is done by feeding an adequate oracle function to the CHSP algorithm described in [EHKS14a,dBDF20]. For our use-case, we will rephrase the main statement of [dBDF20] for the particular case of the Log-S unit lattice.

The CHSP algorithm for the Log-S-unit lattice. Let $S = \{\mathfrak{p}_1, \ldots, \mathfrak{p}_s\}$ be a set of prime ideals of K, let $N_S = \max_{\mathfrak{p} \in S}(\mathcal{N}(\mathfrak{p}))$ and $\boldsymbol{B}_{\mathrm{Log}} \in \mathbb{R}^{(d_{\mathbb{R}}+d_{\mathbb{C}}) \times (d_{\mathbb{R}}+d_{\mathbb{C}}-1)}$ be an orthonormal basis of $\mathrm{Log}\, K_{\mathbb{R}}^0$. We assume that the size of the ideals is polynomial in d and $\ln |\Delta_K|$, that is to say that $N_S = |\Delta_K|^{d^{O(1)}}$. We say that we compute the S-unit group if we compute a sufficiently close approximation of a basis of the following lattice[2], that we call Λ_S:

$$\left\{ (\boldsymbol{\theta}, \boldsymbol{x}, \boldsymbol{a}) \in \mathrm{Arg}_K \times \mathbb{R}^{d_{\mathbb{R}}+d_{\mathbb{C}}-1} \times \mathbb{Z}^s, \quad \mathrm{ExpEx}(\boldsymbol{\theta}, \boldsymbol{B}_{\mathrm{Log}} \boldsymbol{x}) \cdot \prod_{i=1}^{s} \left(\frac{\mathfrak{p}_i}{\mathcal{N}(\mathfrak{p}_i)^{1/d}} \right)^{a_i} = \mathcal{O}_K \right\}$$

$\subset \mathbb{R}^{2(d_{\mathbb{R}}+d_{\mathbb{C}})-1+s}$.

[2] Note that $\mathrm{Arg}_K = \mathrm{Arg}(K_{\mathbb{R}}^{\times})$ has a discrete part $\{0, \pi\}$. To make this group into a lattice, this discrete part is embedded into $\mathbb{R}$, meaning that Λ_S only has values in $\pi\mathbb{Z}$ on these real arguments and trivially includes vectors on which these real arguments are multiples of 2π and of which all the other entries are zero.

In Appendix J we show that Λ_S is full rank in $\mathbb{R}^{2(d_\mathbb{R}+d_\mathbb{C})-1+s}$, and we bound its minima and volume. Any $(\boldsymbol{\theta}, \boldsymbol{x}, \boldsymbol{a}) \in \Lambda_S$ is associated to an S-unit $\alpha = \mathrm{ExpEx}(\boldsymbol{\theta}, \boldsymbol{B}_{\mathrm{Log}}\boldsymbol{x}) \cdot \prod_{i=1}^{s} \mathcal{N}(\mathfrak{p}_i)^{a_i/d} \in \mathcal{O}_{K,S}^\times$ along with the relation: $[\prod_{i=1}^{s} \mathfrak{p}_i^{a_i}] = [\mathcal{O}_K] \in \mathrm{Cl}_K$. Conversely, any element of $\alpha \in \mathcal{O}_{K,S}^\times$ satisfying $(\alpha) = \prod_{i=1}^{s} \mathfrak{p}_i^{a_i}$ is associated with $(\mathrm{LogEx}(\alpha/\mathcal{N}(\alpha)^{1/d}), -\boldsymbol{a}) \in \Lambda_S$ so the knowledge of the full lattice Λ_S gives the full group $\mathcal{O}_{K,S}^\times$ as well as a matrix of relations of the ideals of S in the class group.
The set S can be any set of prime ideals, but in this paper, in order to simplify the analysis of Λ_S, we assume that S generates Cl_K. Our result can be amended to any S by extending it to a larger set S' that generates Cl_K (which can be done, using GRH, by including all prime ideals with norm bounded by $12\ln^2|\Delta_K|$ [Bac90]), and post-process the resulting S'-unit lattice into the S-unit lattice, by straightforward lattice algorithms like HNF.

Let $V > 0$, n and Q be integers. We define $V\mathbb{D}_Q^n = V2^{-Q} \cdot [\![-2^Q, 2^Q]\!]^n \subset [-V, V]^n$. The rephrasing of the main theorem of [dBDF20] for the use-case of Log-S-unit lattice can be formally stated as follows.

Theorem 2.2. *Let $\mathcal{H}$ be a qubit space of dimension 2^n, $\alpha \in (0, 1/32)$ and $\tau \in (0,1)$ be error parameters, $A \geq 1$ and $\nu \leq \mathrm{poly}(d)^{-1}$ two real numbers. Let $\mathbf{f} : \mathbb{R}^{2(d_\mathbb{R}+d_\mathbb{C})-1+s} \to \mathcal{H}$ be a quantum function that is (A, α)-almost-Lipschitz, $(\nu, 1/4 - 8\alpha)$-separative and Λ_S-periodic.*

Then there exist a quantum procedure

- *making k oracle calls to $\mathbf{f}$ over the set $V\mathbb{D}_Q^{2(d_\mathbb{R}+d_\mathbb{C})-1+s}$,*
- *using $O((d+s)Q + n)$ qubits,*
- *using $O\left(kQ(d+s) \cdot (\ln(kQ(d+s)))^2\right)$ quantum gates,*
- $\mathrm{poly}(s, \ln(a))$ *classical bit operations,*

which outputs with probability $\geq 1/2 - 4k\alpha$ a matrix $\tilde{\boldsymbol{B}}$ for which holds that $\left\|\boldsymbol{B} - \tilde{\boldsymbol{B}}\right\| \leq \tau$, where $\boldsymbol{B}$ is a basis of Λ_S satisfying $\|\boldsymbol{B}\| \leq (d+s)^{O(d+s)} \cdot |\Delta_K|^4$. Here, $Q, k \in \mathbb{Z}_{>0}$, $V \in \mathbb{R}_{>0}$ satisfy

- $Q = O\big((d+s)^2(\ln(d+s) + \ln(A)) + (d+s)\ln|\Delta_K| + \ln(1/\tau)\big)$
- $V \geq 1$ *with* $\ln(V) = O\big((d+s)\ln(d+s) + \ln(n) + \ln(|\Delta_K|) + \ln(1/\tau)\big)$,
- $k = O\big((d+s)[\ln(d+s) + \ln(A)] + \ln(|\Delta_K|)\big)$,

This theorem is a modified version of [dBDF20, Theorem 3.3], specialized for the lattice Λ_S. For the sake of completeness, we provide a proof in Appendix K.

3 A quantum encoding of ideal lattices

3.1 Introduction

The goal of this paper is to construct an oracle function for which the *period* tells us something about the S-unit group of a number field. Invoking a period-finding quantum algorithm then allows us to actually compute a representation of this S-unit group.

As already explained in Section 2.7, a crucial ingredient of this oracle function is a sound quantum encoding of ideal lattices. Sound, in the way that the encoding should be well-defined, has a periodicity that allows us to extract information about the S-unit group, and it should be almost-Lipschitz continuous and separative. This is respectively shown in Sections 3.2 to 3.4

In this work, a quantum encoding of ideal lattices $I \in \mathrm{IdLat}_K^0$, i.e., an injective function $F : \mathrm{IdLat}_K^0 \to \mathcal{S}(\mathcal{H})$, is constructed by a (tail-cut) Gaussian superposition over the logarithmic embedded elements of I. This map is similar to the one presented in [BS16,BS25v2], which generalize the one first presented in [EHKS14a]. It differs in the sense that [EHKS14a,BS16] encode elements of I directly, whereas we encode their logarithmic embedding. We deemed this more natural, as it maps the multiplicative distance between ideal lattices to an additive one in the log space. Our encoding also does not encode 0, which is present in every ideal, as its logarithm is not defined.

The quantum encoding The quantum encoding used in the present work is defined as follows.

Definition 3.1. *Let $a, \sigma, \nu \in \mathbb{R}_{>0}$ and $R \geq \sqrt{d}$, and let* Enc *be an a-Lipschitz (for some $a \in \mathbb{R}_{>0}$) and ν-totally separative map $K_{\mathbb{R}}^{\times}|_R \mapsto \mathcal{S}(\mathcal{H})$. For any $J \in \mathrm{IdLat}_K^0$, the punctured tail-cut Gaussian distribution with parameter σ and radius R on J is the distribution on $J \setminus \{0\}\,|_R$ defined by the following rule.*

$$p_{R,\sigma}(J, x) := \frac{\rho_\sigma(x)}{\rho_\sigma(J \setminus \{0\}\,|_R)}$$

The tail-cut Gaussian superposition with parameter $\sigma > 0$ and radius $R > 0$ is defined by

$$\begin{aligned} F_{R,\sigma} : \mathrm{IdLat}_K^0 &\longrightarrow \mathcal{S}(\mathcal{H}) \\ J &\longmapsto |J\rangle := \sum_{x \in J \setminus \{0\}|_R} \sqrt{p_{R,\sigma}(J,x)}|\mathrm{Enc}(x)\rangle \cdot \end{aligned}$$

Theorem 3.1. *Let σ and R such that $\sigma \geq 3 \cdot d^{3/2} \cdot |\Delta_K|^{3/(2d)}$; $R \geq 2 \cdot \sigma \sqrt{d \ln(32\sigma)}$. Let $\nu = 1/(4R)$, $\nu' \leq (30 \cdot (5d + 2a))^{-1}$, $\varepsilon' \in [1/60, 1/30]$ and $\mathrm{Enc} : K_{\mathbb{R}}^{\times} \to \mathcal{S}(\mathcal{H})$ a map that is injective, a-Lipschitz for some $a > 0$, totally ν-separative and $(\nu', 1 - \varepsilon')$-separative.*
Then $F_{R,\sigma}$ is well-defined, $[(5d + 2a), 4e^{-(R/\sigma)^2/2}]$-almost Lipschitz continuous and $(\nu', 1 - \varepsilon')$-separative.

Proof. The conditions on σ, R and ν match the conditions of Lemmas 3.1 and 3.3, hence $F_{R,\sigma}$ is well defined and $[(5d + 2a), 4e^{-(R/\sigma)^2/2}]$-almost Lipschitz continuous. Note that for our value of R and σ, it holds that $4e^{-(R/\sigma)^2/2} \leq 1/30$. We can then apply Lemma 3.6 and the result follows. □

Remark 3.1. The two types of separativity of Enc capture different properties of Enc and serve different goals. The total separativity captures the moment at

which $\langle \mathrm{Enc}(x)|\mathrm{Enc}(y)\rangle$ is zero; it is indispensable for the Lipschitz analysis, as it allows to control the interference between different elements $x, y \in K_{\mathbb{R}}^{\times}$. In contrast, the $(\nu', 1-\varepsilon')$-separativity captures the moment at which $\langle \mathrm{Enc}(x)|\mathrm{Enc}(y)\rangle$ is just a small bit away from 1, and therefore concerns $x, y \in K_{\mathbb{R}}^{\times}$ that are much closer to each other; it is this latter separativity that is inherited by the total fingerprint $F_{R,\sigma}$.

3.2 Well-definedness of $F_{R,\sigma}$

The following lemma shows that the quantum encoding $F_{R,\sigma}$, for adequate parameters, is well-defined (indeed takes values in $\mathcal{S}(\mathcal{H})$) and injective.

Lemma 3.1. *Assume that* $0 < \nu < 1/(2R)$ *and* $R \geq d^{3/2} \cdot |\Delta_K|^{1/2+1/(2d)}$. *Let* Enc *be an injective, a-Lipschitz (for some* $a \in \mathbb{R}_{>0}$*) and* ν*-totally separative map* $K_{\mathbb{R}}^{\times}|_R \mapsto \mathcal{S}(\mathcal{H})$. *Then for any* $I \in \mathrm{IdLat}_K^0$, *we have* $F_{R,\sigma}(I) \in \mathcal{S}(\mathcal{H})$. *Furthermore, the function* $F_{R,\sigma}$ *is injective.*

Proof. Let $I \in \mathrm{IdLat}_K^0$. We first show that $F_{R,\sigma}(I) \in \mathcal{S}(\mathcal{H})$. Lemma 2.4 implies that the distance between two distinct non-zero points of I is always greater than ν, so $\langle \mathrm{Enc}(x)|\mathrm{Enc}(y)\rangle = 0$ for any $x \neq y \in I \setminus \{0\}\,|_R$. Hence $\|F_{R,\sigma}(I)\|^2 = \sum_{x \in I\setminus\{0\}|_R} p_{R,\sigma}(I,x) = 1$.

We conclude with the injectivity of $F_{R,\sigma}$. Let $I, J \in \mathrm{IdLat}_K^0$ satisfy $F_{R,\sigma}(I) = F_{R,\sigma}(J)$. Then, by injectivity of Enc, I and J coincide on the ball of radius R. Then, by Lemmas 2.1 and 2.3, we can conclude that $I = J$. □

3.3 Almost Lipschitz continuity

In this lemma, we give a Lipschitz constant on $F_{\infty,\sigma}$. The proof is similar to the one of [EHKS14b, Section D], but adapted with our metric $\delta_{\mathrm{ideal}}(\cdot,\cdot)$, and the embedding which is Lipschitz over $K_{\mathbb{R}}^{\times}$ and not over $K_{\mathbb{R}}$. We also use differential analysis instead of infinitely close lattices.

Lemma 3.2. *Let* $\sigma \geq 2|\Delta_K|^{1/d}$, *let* $\nu \leq (8\sigma\sqrt{d\ln(32\sigma)})^{-1}$ *and let* Enc *be an a-Lipschitz (for some* $a \in \mathbb{R}_{>0}$*),* ν*-totally separative and injective map. Then the function* $F_{\infty,\sigma}$ *is* $(5d+2a)$*-Lipschitz.*

Proof. We may without loss of generality assume that $I, J \in \mathrm{IdLat}_K^0$ satisfy $\delta_{\mathrm{ideal}}(I,J) < \infty$, since the statement follows trivially otherwise. So there exists $u \in K_{\mathbb{R}}^0$ satisfying $\|\mathrm{LogEx}(u)\| = \delta_{\mathrm{ideal}}(I,J)$ with $J = u \cdot I$.
Writing $|\Psi\rangle = F_{\infty,\sigma}(I) = \sum_{x \in I\setminus\{0\}} q(x)|\mathrm{Enc}(x)\rangle$ and $|\Psi_u\rangle = F_{\infty,\sigma}(uI)$ (where $q(x) = \sqrt{p_{\infty,\sigma}(I,x)}$ and $q_u(ux) = \sqrt{p_{\infty,\sigma}(uI,ux)}$), we have, by the triangle inequality

$$\begin{aligned}&\|\Psi - \Psi_u\| \\ &\leq \underbrace{\|\sum_{x\in I\setminus\{0\}} (q(x) - q_u(ux))|\mathrm{Enc}(x)\rangle\|}_{|\Psi_A\rangle} + \underbrace{\|\sum_{x\in I\setminus\{0\}} q_u(ux)(|\mathrm{Enc}(x)\rangle - |\mathrm{Enc}(ux)\rangle)\|}_{|\Psi_B\rangle}\end{aligned}$$

Let $R = 1/(4\nu)$. We have, by Lemma 2.4 and ν-total separativity,

$$\begin{aligned}
\||\Psi_A\rangle\|^2 &= \sum_{x,y\in I\setminus\{0\}} (q(x) - q_u(ux))(q(y) - q_u(uy)) \langle \mathrm{Enc}(x)|\mathrm{Enc}(y)\rangle \\
&\leq \sum_{x\in I|_R\setminus\{0\}} (q(x) - q_u(ux))^2 + \sum_{x,y\in I\setminus I|_R} (q(x) - q_u(ux))(q(y) - q_u(uy)) \\
&\leq \left(\sum_{x\in I|_R\setminus\{0\}} L_x^2 + \sum_{x,y\in I\setminus I|_R} L_x L_y \right) \|\operatorname{LogEx}(u)\|^2
\end{aligned}$$

where L_x is the Lipschitz-constant of the function $u \mapsto q_u(ux)$ at $u = 1$ with respect to the variable $z = \operatorname{LogEx}(u)$ (i.e., at $z = 0$) and the metric on the image of LogEx is discussed in the text directly after Definition 2.3. The Lipschitzianity and a bound on its Lipschitz constant of the function $u \mapsto q_u(ux)$ is proven in Lemma C.2 and Lemma C.3.

By Lemma C.2, we have $\sum_{x\in I|_R\setminus\{0\}} L_x^2 \leq d^2\pi^2$ and by Lemma C.3 we have $\sum_{x\in I\setminus I|_R} L_x \leq 1$, so $\sum_{x,y\in I\setminus I|_R} L_x L_y = (\sum_{x\in I\setminus I|_R} L_x)^2 \leq 1$. Hence,

$$\||\Psi_A\rangle\| \leq \sqrt{d^2\pi^2 + 1}\|\operatorname{LogEx}(u)\|.$$

We finish by bounding $\||\Psi_B\rangle\|$. We may assume $\|\operatorname{LogEx}(u)\| \leq \nu$. For $x \in I|_R, y \in I$ with $x \neq y$, by Lemma 2.4 and the triangle inequality, it holds that $\delta_{K_{\mathbb{R}}^\times}(x, y), \delta_{K_{\mathbb{R}}^\times}(x, u \cdot y), \delta_{K_{\mathbb{R}}^\times}(u \cdot x, y)$ and $\delta_{K_{\mathbb{R}}^\times}(u \cdot x, u \cdot y)$ are all greater than $1/(4R) = \nu$. Hence, by ν-total separativity of Enc, for $x \in I|_R \setminus \{0\}$ and $y \in I\setminus\{0\}$, the inner product between $|\mathrm{Enc}(x)\rangle - |\mathrm{Enc}(ux)\rangle$ and $|\mathrm{Enc}(y)\rangle - |\mathrm{Enc}(uy)\rangle$ equals zero if $x \neq y$. Hence, by using the Cauchy-Schwarz inequality for the inner products of $|\mathrm{Enc}(x)\rangle - |\mathrm{Enc}(ux)\rangle$ and $|\mathrm{Enc}(y)\rangle - |\mathrm{Enc}(uy)\rangle$, and their a-Lipschitz continuity, we obtain, by Lemma C.3,

$$\begin{aligned}
\||\Psi_B\rangle\|^2 &= a^2 \cdot \|\operatorname{LogEx}(u)\|^2 \left(\sum_{x\in I|_R\setminus\{0\}} q_u(ux)^2 + \sum_{x,y\in I\setminus I|_R} q_u(ux) q_u(uy) \right) \\
&\leq 2 \cdot a^2 \cdot \|\operatorname{LogEx}(u)\|^2
\end{aligned}$$

Hence $\|\Psi - \Psi_u\| \leq (\sqrt{d^2\pi^2 + 1} + \sqrt{2}a)\|\operatorname{LogEx}(u)\| \leq (5d + 2a)\|\operatorname{LogEx}(u)\|$, which finishes the proof. □

We now give the almost-Lipschitz parameters for $F_{R,\sigma}$, which allows to take into account that this map is not continuous due to the tail-cut, which is novel compared to [EHKS14a,BS25v2].

Lemma 3.3. *Let σ, ν and* Enc *satisfy the conditions of Lemma 3.2. Let $R \geq 2\sigma$. Then $F_{R,\sigma}$ is $[(5d + 2a), 4e^{-(R/\sigma)^2/2}]$-almost Lipschitz continuous.*

The proof of this lemma can be found in Appendix B

3.4 Separativity

In this section, we describe the separativity condition for the function $F_{R,\sigma}$. We show that if $\langle F_{R,\sigma}(I_1)|F_{R,\sigma}(I_2)\rangle$ is sufficiently close to 1, then there exists a small distortion $u \in K_{\mathbb{R}}^0$ such that $u \cdot I_1 = I_2$. The proof technique is similar to the one of [EHKS14b,BS25v2], but the details are different as we make use of the $(\nu', 1-\varepsilon')$-separativity of Enc. This and the fact that Enc encodes over $K_{\mathbb{R}}^\times$ and not $K_{\mathbb{R}}$ makes that we do not derive the same conditions on the parameters σ, R and Enc. The proof of those statements can be found in Appendix B.

Lemma 3.4. *Let* Enc *be* $(\nu', 1-\varepsilon')$*-separative for some* $\nu', \varepsilon' \in (0,1)$*, and let* $I, J \in \mathrm{IdLat}_K^0$ *satisfy* $\langle F_{R,\sigma}(I)|F_{R,\sigma}(J)\rangle \geq 1-\varepsilon'$*. Then there exists* $I' \in \mathrm{IdLat}_K$ *with* $\delta_{\mathrm{ideal}}(I, I') \leq \nu'$ *such that* $I' \cap J \neq \{0\}$*.*

The next lemma is adapted from [EHKS14b, Lemma E.7].

Lemma 3.5. *Let* $\sigma \geq 3 \cdot d^{3/2} \cdot |\Delta_K|^{3/(2d)}$*,* $R \geq \sqrt{d} \cdot \sigma$ *and* $\nu \leq 1/(2R)$*. Let* $I, J \in \mathrm{IdLat}_K^0$ *satisfying* $(I \cap J)|_R \neq \{0\}$*. Then either* $I = J$ *or*

$$\langle I|J\rangle < \frac{4}{5}.$$

We now state the separativity of $F_{R,\sigma}$.

Lemma 3.6. *Let* $\sigma \geq 3 \cdot d^{3/2} \cdot |\Delta_K|^{3/(2d)}$*,* $\nu \leq 1/(2R)$ *and* Enc *an injective map which is* ν*-totally separative over* $(K_{\mathbb{R}}^\times, \delta_{K_{\mathbb{R}}^\times})$*. Furthermore, assume that* σ, ν, R *and* Enc *are such that* $F_{R,\sigma}$ *is* (A, α)*-almost Lipschitz for some* $A \in \mathbb{R}$ *with* $\alpha \leq 1/30$*, and that* Enc *is* $(\nu', 1-\varepsilon')$*-separative for some* $\varepsilon' \in (0, 1/30]$ *and* $\nu' \leq 1/(30A)$*. Then the function* $F_{R,\sigma}$ *is* $(\nu', 1-\varepsilon')$*-separative.*

4 Implementation of the Quantum algorithm

4.1 Introduction

The goal of the current section is to show how to construct a quantum circuit evaluating a quantum oracle $G_{R,\sigma}$ (built from $F_{R,\sigma}$) and to precisely analyze its quantum complexity. In particular, we focus on deriving a precise polynomial bound on the number of quantum gates and qubits required to build the quantum circuit. For the classical operations that might occur in-between, we are less precise and merely show their polynomial running time.

We start with defining the quantum oracle $G_{R,\sigma}$ in Section 4.2, which consists of a composition of two functions: one that defines the replete ideal I from exponents of the prime ideals from S (for which $|S| = s$), the embeddings and the phases. This oracle function is very similar to that of [EHKS14a,BS16].

$$\begin{array}{ccccccccc} \underbrace{\mathbb{R}^{d_{\mathbb{R}}+d_{\mathbb{C}}-1}}_{\text{embeddings}} & \times \underbrace{\mathbb{Z}^{d_{\mathbb{R}}} \times \mathbb{R}^{d_{\mathbb{C}}}}_{\text{`phases'}} & \times & \underbrace{\mathbb{Z}^s}_{\text{prime ideals}} & \longrightarrow \mathrm{IdLat}_K & \longrightarrow & \{\text{quantum states}\} \\ \boldsymbol{x}, & \boldsymbol{s}, \quad \boldsymbol{\theta}, & & \boldsymbol{a} & \mapsto I & \mapsto & F_{R,\sigma}(I). \end{array}$$

We show that a repetition of this oracle function (meaning, a manifold tensor product) is sufficient for this function to satisfy the conditions of Theorem 2.2.

In Section 4.3, we make the important observation that the CHSP algorithm (see Theorem 2.2) as in [dBDF20] only queries the oracle functions on vectors consisting of dyadic rationals (i.e., rationals with a power of two as a denominator); in other words, it queries the oracle function on a specific grid. This allows for the precomputation of the ideals, embeddings and phases in a specific power-of-two way, that vastly diminishes the number of expensive ideal multiplications and lattice reductions.

These precomputations of these ideals involve exponentially large powers of prime ideals, for which extra care needs to be taken to represent them in such a way that they can be handled and computed with efficiently. This is the objective of Sections 4.4 and 4.5.

The rest of this section is devoted to making precise the computations in Algorithm 4.1 (that computes the quantum function $G_{R,\sigma}$) and carefully determine their quantum gate and memory complexity. We note that a special algorithm is devised (see Section 4.7) to make ideal multiplication more memory-efficient in the quantum setting. Additionally, a new analysis of a quantum variant of the GPV algorithm [GPV08] (see also [Bra18,LMZ23]) is used in Algorithm 4.1, of which the algorithm definition, its complexity and numerical analysis is presented in Section 6.

4.2 From ideal lattice encoding to CHSP Oracle

Recall that $S = \{\mathfrak{p}_1, \dots, \mathfrak{p}_s\}$ be a set of prime ideals of $\mathcal{O}_K$ of size s generating Cl_K, $\boldsymbol{V} \in \mathbb{R}^{(d_\mathbb{R}+d_\mathbb{C})\times(d_\mathbb{R}+d_\mathbb{C}-1)}$ a fixed orthonormal basis of $\mathrm{Log}(K^0_\mathbb{R})$, we define the function

$$G_{R,\sigma} : \mathbb{R}^{d_\mathbb{R}+d_\mathbb{C}-1} \times \mathbb{R}^{d_\mathbb{C}} \times \mathbb{Z}^{d_\mathbb{R}} \times \mathbb{Z}^s \to \mathcal{S}(\mathcal{H}),$$

by

$$G_{R,\sigma}(\boldsymbol{x}, \boldsymbol{\theta}, \boldsymbol{s}, \boldsymbol{a}) = F_{R,\sigma}\left(\mathrm{ExpEx}\left((\boldsymbol{s}', \boldsymbol{\theta}', \boldsymbol{B}_{\mathrm{Log}} \cdot \boldsymbol{x})\right) \cdot \prod_{i=1}^{s} (\mathfrak{p}_i/\mathcal{N}(\mathfrak{p}_i)^{1/d})^{a_i}\right), \quad (1)$$

where $\boldsymbol{\theta'} = \boldsymbol{\theta} \bmod 2\pi$ and $s'_i = \pi$ if s_i is even, 0 else. Apart from the differences on $F_{R,\sigma}$ addressed in the previous sections, this function is essentially the same as the one presented in [BS25v2, Section 4].

Lemma 4.1. *Assume that $R, \sigma, \nu, \nu', \varepsilon'$, and* Enc *satisfy the hypotheses of Theorem 3.1, then there exists $l = O(1) \in \mathbb{Z}$ such that $G^{\otimes l}_{R,\sigma}$ is $(\nu', 1/5)$-separative.*

Proof. This follows directly from Theorem 3.1 and the fact that if a function f is $(\nu', 1-\varepsilon')$-separative, then the function $f^{\otimes l}$ is $(\nu', (1-\varepsilon')^l)$-separative. Per assumption, ε' is bounded from below by a positive constant (namely $1/60$), and hence $l = \lceil \frac{\ln(1/5)}{\ln(1-\varepsilon')} \rceil \leq 96$ is bounded from above by a constant. □

Lemma 4.2. *Assume that R, σ, ν, ν', a, and* Enc *satisfy the hypotheses of Theorem 3.1 and let l be as in Lemma 4.1. Then, $G^{\otimes l}_{R,\sigma}$ is $(O(d+a), O(\exp(-R^2/\sigma^2)))$-almost Lipschitz.*

4.3 From continuous to polynomial-size input space

In [BS25v2], the product $\mathrm{ExpEx}\left((\boldsymbol{s'}, \boldsymbol{\theta'}, \boldsymbol{B}_{\mathrm{Log}} \cdot \boldsymbol{x})\right) \cdot \prod_{i=1}^{s} (\mathfrak{p}_i / \mathcal{N}(\mathfrak{p}_i)^{1/d})^{a_i}$ is performed using E-ideal arithmetic [BS25v2, Section 4.2] and approximation of the exponential function. Overall, this technique consists of representing high power of ideals $\mathfrak{p}^a$ as a product of small-norm elements and a small-norm ideal. The algorithm is presented in a classic setting but it is implied that it is transformed into a quantum algorithm by means such as Theorem 2.1. Here, we twist this approach by noting that in Theorem 2.2, the oracle $G_{R,\sigma}$ is only called on a finite set. This set is equal to $V\mathbb{D}_Q^{2(d_{\mathbb{R}}+d_{\mathbb{C}})+s-1} = \frac{V}{2^Q} \cdot [\![-2^Q, 2^Q]\!]^{2(d_{\mathbb{R}}+d_{\mathbb{C}})+s-1}$, meaning that if V and Q can be computed beforehand, our algorithm's parameter set can be restricted to $V\mathbb{D}_Q^{2(d_{\mathbb{R}}+d_{\mathbb{C}})+s-1}$, allowing for a lot of classical precomputation.

Given V and Q, the strategy we choose is to pre-compute classically a two-element representation of a finite (polynomial size) set of integral ideals and to transform the calls to $G_{R,\sigma}$ into a (polynomial size) multiplication of ideals, which can be done exactly with a quantum algorithm. Recall that $\boldsymbol{B}_{\mathrm{Log}} = [\boldsymbol{b}_1, \ldots, \boldsymbol{b}_{d_{\mathbb{R}}+d_{\mathbb{C}}-1}]$ is an orthonormal basis of $\mathrm{Log}(K_{\mathbb{R}}^0)$; we have that for any $\boldsymbol{\theta'}$, $\boldsymbol{s'}$, $\boldsymbol{x} = \boldsymbol{y} \cdot V/2^Q$ with $\boldsymbol{y} \in [\![-2^Q, 2^Q]\!]^{d_{\mathbb{R}}+d_{\mathbb{C}}-1}$ and $\boldsymbol{a} \in [\![-2^Q, 2^Q]\!]^s$,

$$
\begin{aligned}
&\mathrm{ExpEx}\left((\boldsymbol{s'}, \boldsymbol{\theta'}, \boldsymbol{B}_{\mathrm{Log}} \cdot \boldsymbol{x})\right) \cdot \prod_{i=1}^{s} (\mathfrak{p}_i / \mathcal{N}(\mathfrak{p}_i)^{1/d})^{a_i} \\
&= \mathrm{Exp}(i(\boldsymbol{s'}, \boldsymbol{\theta'})) \cdot \prod_{i=1}^{d_{\mathbb{R}}+d_{\mathbb{C}}-1} \mathrm{Exp}(\boldsymbol{b}_i \cdot x_i) \cdot \prod_{i=1}^{s} (\mathfrak{p}_i / \mathcal{N}(\mathfrak{p}_i)^{1/d})^{a_i} \\
&= \mathrm{Exp}(i(\boldsymbol{s'}, \boldsymbol{\theta'})) \cdot \prod_{i=1}^{d_{\mathbb{R}}+d_{\mathbb{C}}-1} \prod_{\substack{j=0 \\ \mathrm{bit}_j(|y_i|)=1}}^{Q} \mathrm{Exp}(\mathrm{sign}(y_i) \frac{V}{2^Q} \boldsymbol{b}_i \cdot 2^j) \cdot \prod_{i=1}^{s} \prod_{\substack{j=0 \\ \mathrm{bit}_j(a_i)=1}}^{Q} (\mathfrak{p}_i / \mathcal{N}(\mathfrak{p}_i)^{1/d})^{2^j},
\end{aligned}
$$

where $\mathrm{bit}_j(x)$ is the jth bit in the binary decomposition of $x \in \mathbb{Z}_{\geq 0}$ and $\mathrm{Exp}(\boldsymbol{a})$ (for $\boldsymbol{a} \in (\mathbb{R} \cup \{i \cdot k\pi, k \in \mathbb{Z}\})^{d_{\mathbb{R}}} \times \mathbb{C}^{d_{\mathbb{C}}}$) is the element of $K_{\mathbb{R}}$ satisfying $\sigma_i(\mathrm{Exp}(\boldsymbol{a})) = \exp(a_i)$ for $1 \leq i \leq d$. This equality implies that if a $\mathbb{Z}$-basis of the ideals

$$
I_{\pm,i,j} = \mathrm{Exp}(\pm \frac{V}{2^Q} \boldsymbol{b}_i)^{2^j} \cdot \mathcal{O}_K \text{ and } F_{\pm,k,j} = \mathfrak{p}_k^{\pm 2^j}
$$

has been pre-computed, then – up to rescaling and multiplying by a phase – computing the ideal involved in $G_{R,\sigma}(\cdot)$ is reduced to computing a polynomially large ideal product. This technique is similar to the one presented in [BS25v2, Section 4.1], but with classical precomputation and without needing E-ideal arithmetic (described in [BS25v2, Section 4.2]), since we precompute integral ideals.

4.4 Interlude: small approximations of ideals

As said before, the ideals we are considering have exponential size and would lead to exponential gate and memory complexity. In order to represent and manipulate those large ideals, the usual method (see for example [BS25v2, Algorithm

2]) is to use a compact representation of ideals, namely a representation of the form $((\alpha_i)_{0\le i\le k}, \mathfrak{a})$ such that $I^{2^k} = \prod_{i=0}^{k} \alpha_i^{2^i} \cdot \mathfrak{a}$ with $\mathfrak{a}$ and the α_i of polynomial size. We propose a modification of this method and show that up to taking a larger (but still polynomially-sized) $\mathfrak{a}$, we can have almost all the geometric information of the ideal I^{2^k} represented in the ideal $\mathfrak{a}$.

Lemma 4.3. *Let $p, m \ge 1$, and $k \ge 0$. For any replete ideal $I \subset K_{\mathbb{R}}$ there exists an integral ideal $\mathfrak{a}$ and an element $\alpha \in K_{\mathbb{R}}$ such that and $I^{2^k} = \alpha \cdot \mathfrak{a}$. with $\mathfrak{a}$ satisfying $\mathcal{N}(\mathfrak{a}) \le B_{K,p,m,k}$ where*

$$\ln(B_{K,p,m,k}) \le C \cdot d \cdot (\ln(m) + d + k + p + \ln(|\Delta_K|)) .$$

for some absolute $C > 1$, and

$$\delta_{\text{ideal}}(I^{2^k}/\mathcal{N}(I^{2^k})^{1/d}, \mathfrak{a}/\mathcal{N}(\mathfrak{a})^{1/d}) \le 2^{-p}/m$$

Furthermore, if I is a fractional ideal, then there exists a polynomial time algorithm in size$(I), m, p, \ln(|\Delta_K|), k$ and d computing a basis of $\mathfrak{a}$ and a polynomial-size representation of α.

The proof of this lemma is available in Appendix D.

4.5 Classical precomputations

We precompute classically in polynomial-time the integral ideals $(\widetilde{I_{\pm,i,j}})_{i,j}$ and $(\widetilde{F_{\pm,k,j}})_{k,j}$ which are the integral approximation of the ideals $(I_{\pm,i,j})_{i,j}$ and $(F_{\pm,k,j})_{k,j}$ as described in Lemma 4.3 for $m = Q \cdot (d_{\mathbb{R}} + d_{\mathbb{C}} + s)$. These ideals are represented by a two element representation (see Appendix I for a presentation of our computation model), which have Euclidean norm $(2^d \mathcal{N}(I)^{1/d} \cdot |\Delta_K|^{1/(2d)})^{O(1)}$ for an ideal I. We also pre-compute a two-element representation of the inverses of the $\widetilde{I_{\pm,i,j}}$ and $\widetilde{F_{\pm,k,j}}$, which will be used to quantumly compute the product in an invertible way (see Algorithm 4.2).

These pre-computations allow us to describe the quantum algorithm computing the function $G_{R,\sigma}$ in Algorithm 4.1. We study its complexity and the required precision in the next few subsections. The outline of the algorithm is essentially the same as the one described in [BS25v2], where our presentation differs by having additional explicit Gaussian superposition computations and classical precomputations. In the next subsection, we detail Algorithm 4.1 step by step.

4.6 Determination of the set of ideals $\mathcal{A}$ in Line 2

Let $\boldsymbol{x} \in (V/2^Q) \cdot [\![-2^Q, 2^Q]\!]^{d_{\mathbb{R}}+d_{\mathbb{C}}-1}, \boldsymbol{\theta} \in \mathbb{R}^{d_{\mathbb{C}}}, \boldsymbol{s} \in \mathbb{Z}^{d_{\mathbb{R}}}, \boldsymbol{a} \in [\![-2^Q, 2^Q]\!]^s$ be the input of the algorithm. We approximate the ideal (up to the norm and phase factors)

$$I = \prod_{i=1}^{d_{\mathbb{R}}+d_{\mathbb{C}}-1} \prod_{\substack{j=0 \\ \text{bit}_j(|x_i|)=1}}^{Q} \text{Exp}(\pm \frac{V}{2^Q} \boldsymbol{b}_i \cdot 2^j) \cdot \prod_{i=1}^{s} \prod_{\substack{j=0 \\ \text{bit}_j(|a_i|)=1}}^{Q} \mathfrak{p}_i^{2^j} .$$

Algorithm 4.1 Overview of $\mathrm{GCompute}_{R,\sigma,q,p,\varepsilon}$

Input: $\boldsymbol{x} \in (V/2^Q) \cdot [\![-2^Q, 2^Q]\!]^{d_{\mathbb{R}}+d_{\mathbb{C}}-1}, \boldsymbol{\theta} \in \mathbb{R}^{d_{\mathbb{C}}}, \boldsymbol{s} \in \mathbb{Z}^{d_{\mathbb{R}}}, \boldsymbol{a} \in [\![-2^Q, 2^Q]\!]^s$
Output: $|\psi'\rangle$ ε-close to $G_{R,\sigma}(\boldsymbol{x}, \boldsymbol{\theta}, \boldsymbol{s}, \boldsymbol{a})$.

1: Compute $|\boldsymbol{s'}\rangle$, $|\boldsymbol{\theta'}\rangle$ and $|\boldsymbol{y}\rangle = 2^Q/V \cdot \boldsymbol{x}$ (see Eq. (1).
2: Compute the list $\mathcal{A}$ of ideals associated with $\boldsymbol{y}$ and $\boldsymbol{a}$ (see Eq. (2)).
3: Define $\mathfrak{b} = \prod_{\mathfrak{a}\in\mathcal{A}} \mathfrak{a}$, compute $\boldsymbol{H}_{\mathfrak{b}} \leftarrow \mathrm{HNF}(\prod_{\mathfrak{a}\in\mathcal{A}} \mathfrak{a})$.
4: Let $\boldsymbol{M}_{\mathfrak{b}} \leftarrow \mathrm{QuantumLLL}_{2^d}(\boldsymbol{H}_{\mathfrak{b}})$.
5: Compute $\boldsymbol{B}'_{\mathfrak{b}} = \tilde{\boldsymbol{B}}_{\mathcal{O}_K} \cdot \boldsymbol{M}_{\mathfrak{b}}/\mathcal{N}(\mathfrak{b})^{1/d}$ an approximation of a basis of the canonical embedding of $\mathfrak{b}/\mathcal{N}(\mathfrak{b})^{1/d}$.
6: Compute $\boldsymbol{R}'_{\mathfrak{b}} = \mathrm{Householder}(\boldsymbol{B}'_{\mathfrak{b}})$ an approximation of the R-factor of the QR-factorization of $\boldsymbol{B}'_{\mathfrak{b}}$.
7: Compute $|\phi'\rangle = \mathrm{QGaussian}\ (\boldsymbol{R}'_{\mathfrak{b}})$, an approximation of the Gaussian superposition using Algorithm 6.1 with error parameter $\varepsilon/2$, periodization parameter q, deviation $2^p \cdot \sigma$, matrix $2^p \cdot \boldsymbol{R}'_{\mathfrak{b}} \in \mathbb{Z}^{d\times d}$.
8: Apply multiplication by $\mathrm{diag}(\boldsymbol{\theta}, \boldsymbol{s'}) \cdot \boldsymbol{B}'_{\mathfrak{b}}$ to the coordinate qubits of $|\phi'\rangle$ to get $|\varphi'\rangle$.
9: Compute $|\psi'\rangle = \mathrm{Enc}'(|\varphi'\rangle)$ an approximation of Enc with precision 2^{-p} over the state $|\phi'\rangle$.
10: Un-compute $\boldsymbol{R}'_{\mathfrak{b}}, \boldsymbol{B}'_{\mathfrak{b}}, \boldsymbol{M}_{\mathfrak{b}}, \boldsymbol{H}_{\mathfrak{b}}, \mathcal{A}, \boldsymbol{y}, \boldsymbol{\theta'}$ and $\boldsymbol{s'}$.
11: **Return** $|\psi'\rangle$.

Recall that we have pre-computed polynomial-size approximations of all of the ideals of this product (see Section 4.5 for the definitions of $\widetilde{F}$ and $\widetilde{I}$). We can then define the set

$$\mathcal{A} = \left\{ \widetilde{I_{\mathrm{sign}(x_i),i,j}},\ \ \mathrm{bit}_j(|x_i|) = 1 \right\} \bigcup \left\{ \widetilde{F_{\mathrm{sign}(a_k),k,j}},\ \mathrm{bit}_j(|a_k|) = 1 \right\}, \qquad (2)$$

where $i \in [\![1, d_{\mathbb{R}}+d_{\mathbb{R}}]\!]$, $j \in [\![Q-1]\!]$ and $k \in [\![1, s]\!]$. Note that $|\mathcal{A}| \leq Q\cdot(d_{\mathbb{R}}+d_{\mathbb{C}}+s) = m$. Finally, $\mathcal{A}$ can be computed with a number of quantum gates and memory negligible compared to the rest of the algorithm.

4.7 Computation of the product of ideals in Line 3

A quantum algorithm to compute the product of two ideals in place. The classical algorithm to multiply two ideals is described in [Coh93, §4.7.1], and works as follows: if $I = (x_I) + (y_I)$ and J is given by its HNF $(b_1, \ldots, b_d)$ with $x_I, y_I, (b_i) \in K$, then the lattice spanned by $\boldsymbol{B} = [x_I \cdot b_1, y_I \cdot b_1, \ldots, x_I \cdot b_d, y_I \cdot b_d]$ is $I \cdot J$ and a basis of it can be extracted by computing the HNF of $\boldsymbol{B}$. By Theorem 2.1, there exists a quantum circuit QIdMult such that

$$\mathrm{QIdMult} \cdot |x_I, y_I\rangle|\boldsymbol{H}_J\rangle|\boldsymbol{C} \in \{0,1\}^{d^2\cdot\lceil \ln(B)\rceil}\rangle = |x_I, y_I\rangle|\boldsymbol{H}_J\rangle|\boldsymbol{C} \oplus \boldsymbol{H}_{I\cdot J}\rangle,$$

where the $\oplus$ stands for bit-wise xor and where B is an upper bound of the maximal size of the entries of $\boldsymbol{C}$. In order to adapt this algorithm to the quantum context and minimize the quantum memory used, we use an "in-place" version of it that we call QIdMultInPlace, described in Algorithm 4.2.

Algorithm 4.2 QIdMultInPlace

Input: $|x_{\mathfrak{a}}, y_{\mathfrak{a}}\rangle$ a two-element representation of $\mathfrak{a}$, $|x_{\mathfrak{a}^{-1}}, y_{\mathfrak{a}^{-1}}\rangle$ a two-element representation of $\mathfrak{a}^{-1}$, $|\boldsymbol{H}_{\mathfrak{b}}\rangle$ the HNF of an ideal $\mathfrak{b}$.

Output: $|x_{\mathfrak{a}}, y_{\mathfrak{a}}\rangle|x_{\mathfrak{a}^{-1}}, y_{\mathfrak{a}^{-1}}\rangle|\boldsymbol{H}_{\mathfrak{a}\cdot\mathfrak{b}}\rangle$.

1: Initialize the state $\underbrace{|x_{\mathfrak{a}}, y_{\mathfrak{a}}\rangle}_{R_1}\underbrace{|x_{\mathfrak{a}^{-1}}, y_{\mathfrak{a}^{-1}}\rangle}_{R_2}\underbrace{|\boldsymbol{H}_{\mathfrak{b}}\rangle}_{R_3}\underbrace{|0^{d\times d}\rangle}_{R_4}$

2: Apply QIdMult on registers R_1, R_3, R_4 yielding $|x_{\mathfrak{a}}, y_{\mathfrak{a}}\rangle|x_{\mathfrak{a}^{-1}}, y_{\mathfrak{a}^{-1}}\rangle|\boldsymbol{H}_{\mathfrak{b}}\rangle|\boldsymbol{H}_{\mathfrak{a}\mathfrak{b}}\rangle$

3: Uncompute the state $|\boldsymbol{H}_{\mathfrak{b}}\rangle$ in R_3 by applying QIdMult on R_2, R_4, R_3, yielding (by deletion with bit-wise XOR) $|x_{\mathfrak{a}}, y_{\mathfrak{a}}\rangle|x_{\mathfrak{a}^{-1}}, y_{\mathfrak{a}^{-1}}\rangle|0^{d\times d}\rangle|\boldsymbol{H}_{\mathfrak{a}\mathfrak{b}}\rangle$

4: Swap R_3 and R_4 and **Return** R_1, R_2, R_4, R_3

As described in Appendix I.2, the classical algorithm computing the product of two ideals runs in time

$$O\left((d^{\omega+1}\left(\text{size}(x_{\mathfrak{a}}) + \text{size}(y_{\mathfrak{a}}) + \ln(\mathcal{N}(\mathfrak{a}\cdot\mathfrak{b}))\right))^{1+o(1)}\right).$$

In our context, all $\mathfrak{a}$ are obtained from Lemma 4.3 so that their two element representation have size $O(\ln\left(B_{K,p,m,Q}^{1/d}\right))$. The size of $\mathfrak{b}$ will change during the algorithm. The overall amount of quantum gates and memory for one call to QIdMultInPlace is then

$$O\left((d^{\omega+1}\left(\ln(B_{K,p,m,Q}) + \ln(\mathcal{N}(\mathfrak{b}))\right))^{1+o(1)}\right).$$

The product algorithm of Line 3. Let $\mathcal{A} = \{\mathfrak{a}_1, \ldots, \mathfrak{a}_k\}$. We start by $\mathfrak{b}_0 = \mathcal{O}_K$, represented by the identity matrix, and let $\mathfrak{b}_{i+1} = \mathfrak{a}_i \cdot \mathfrak{b}_i$. We then repeatedly use the two-element representation of any ideal $\mathfrak{a}_i$ and their inverse to multiply $\mathfrak{b}_{i-1}$ by $\mathfrak{a}_i$ using QIdMultInPlace.

Complexity. The norm of every $\mathfrak{a}_i$ is bounded by $B_{K,p,m,Q}$, the norm of $\mathfrak{b}_i = \prod_{j\leq i}\mathfrak{a}_j$ then satisfies $\mathcal{N}(\mathfrak{b}_i) \leq B_{K,p,m,Q}^i$. The fact that QIdMultInPlace is in place implies that the quantum memory is re-used from the computation of one product to the next one. There are at most m terms in the product, the number of memory qubits used in order to compute the product is then

$$O\left(\left(d^{\omega+1}\cdot m\cdot\ln(B_{K,p,m,Q})\right)^{1+o(1)}\right),$$

and the required number of quantum gates is

$$O\left(\left(d^{\omega+1}\cdot m^2\cdot\ln(B_{K,p,m,Q})\right)^{1+o(1)}\right).$$

The final norm of the ideal $\mathfrak{b}$ is then bounded by $B_{K,p,m,Q}^m$.

Note that the approach using the HNF in order to limit the number of calls to QuantumLLL is not the one chosen in [BS25v2]. There, the authors use QuantumLLL in order to compute the product. We think our method leads to an easier error analysis (since a lot of the computation are made over the integers), but it might not be the optimal one (as for the complexity claimed in [BS25v2, Theorem 2]).

Error analysis. Let $I = \mathrm{Exp}(\boldsymbol{x}) \prod_{i=1}^{s} (\mathfrak{p}_i/\mathcal{N}(\mathfrak{p}_i)^{1/d})^{a_i}$ be the ideal whose superposition is computed. The ideal distance between $\mathfrak{b}/\mathcal{N}(\mathfrak{b})^{1/d}$ and I is, by Lemma 4.3, bounded by $|\mathcal{A}| \cdot 2^{-p}/m \leq 2^{-p}$.

4.8 Matrix reduction of Step 4

We implement the 2^d-reduction of the integral matrix $\boldsymbol{H}_{\mathfrak{b}}$ by using the Quantum LLL procedure. Assessing the complexity of this procedure is not an easy task. Our first idea would be to simulate the classical LLL algorithm (either the textbook version [LLL82] or the quadratic complexity version [NS09]) using Theorem 2.1, but doing so requires a very large amount of quantum memory. Tiepelt and Szepieniec [TS19] proposed a version of the implementation of textbook LLL with a new technique to achieve a trade-off between quantum memory and gates, leading to a better memory efficiency. They claim that their techniques apply to the quadratic-complexity version of LLL, but do not estimate the number of quantum gates needed in this case. To retain full generality, we will denote by $\mathrm{LLLGates}(n, b)$ (resp. $\mathrm{LLLMem}(n, b)$) the number of quantum gates (resp. of quantum memory) needed to compute the LLL reduction of a full rank integral matrix of size $n \times n$ whose entries are bounded by 2^b. For example, for [TS19] version of textbook LLL, we have (see [TS19, Eq 7, 8 and p.15])

$$\mathrm{LLLGates}(n, b) = O\left(n^7 \cdot b^{3.5}\right) \text{ and } \mathrm{LLLMem}(n, b) = O\left(n^4 b^{3/2}\right)$$

Step 4 is then realized using $\mathrm{LLLGates}(d, m \ln(B_{K,p,Q}))$ quantum gates and a quantum memory of $\mathrm{LLLMem}(d, m \ln(B_{K,p,Q}))$ memory qubits.

Error analysis. These computations are made over integral matrices, hence no error analysis is required for this step.

4.9 From the exact representation to canonical embedding in Step 5

We recall that the real computations are done within absolute precision 2^{-p} (see Appendix I.2). As mentioned in Appendix I.2, the integral ideal $\mathfrak{b}$ is represented as its basis $\boldsymbol{M}_{\mathfrak{b}}$ over $\boldsymbol{B}_{\mathcal{O}_K}$, we now compute the embedding into $K_{\mathbb{R}}$ of the basis of $\mathfrak{b}$. This is done by multiplying $\boldsymbol{M}_{\mathfrak{b}}$ by $\boldsymbol{B}_{\mathcal{O}_K}$. Note that $\boldsymbol{B}_{\mathcal{O}_K}$ is represented in fixed point representation with an error of 2^{-p} by the matrix $\widetilde{\boldsymbol{B}_{\mathcal{O}_K}}$.

Complexity analysis The bit-size of $\widetilde{\boldsymbol{B}_{\mathcal{O}_K}}$ is $O\left(d^2 \cdot \left(p + \ln\left(|\Delta_K|^{1/d}\right)\right)\right)$ since $\boldsymbol{B}_{\mathcal{O}_K}$ is LLL-reduced. The product $\boldsymbol{B}'_{\mathfrak{b}} = \tilde{\boldsymbol{B}}_{\mathcal{O}_K} \cdot \boldsymbol{M}_{\mathfrak{b}}/\mathcal{N}(\mathfrak{b})^{1/d}$ is computed in two steps. First, computing $\boldsymbol{M}_{\mathfrak{b}}/\mathcal{N}(\mathfrak{b})^{1/d}$ within precision 2^{-p}, which is done in time $O(d^2(\ln\left(B_{K,p,m,Q}^{m/d}\right))^{1+o(1)})$. Then the product of the two matrices is computed in time

$$O\left(d^{\omega} \cdot \ln\left(p + \ln\left(|\Delta_K|^{1/d}\right)\right)^{1+o(1)}\right)$$

Error analysis Now that we are working with real values, we track the error propagation in our computations step by step. We define $\boldsymbol{B}_{\mathfrak{b}} = \boldsymbol{B}_{\mathcal{O}_K} \cdot \boldsymbol{M}_{\mathfrak{b}}/\mathcal{N}(\mathfrak{b})^{1/d}$ an exact basis of $\mathfrak{b}/\mathcal{N}(\mathfrak{b})^{1/d}$, let $\boldsymbol{M} = \boldsymbol{M}_{\mathfrak{b}}/\mathcal{N}(\mathfrak{b})^{1/d}$ and $\boldsymbol{M'}$ the 2^{-p} approximation of $\boldsymbol{M}$. We have $\boldsymbol{B}'_{\mathfrak{b}} = \tilde{\boldsymbol{B}}_{\mathcal{O}_K} \cdot \boldsymbol{M'}$, and hence

$$\begin{aligned}\|\boldsymbol{B}'_{\mathfrak{b}} - \boldsymbol{B}_{\mathfrak{b}}\| \leq & \left\|\tilde{\boldsymbol{B}}_{\mathcal{O}_K} \cdot \boldsymbol{M'} - \boldsymbol{B}_{\mathcal{O}_K} \cdot \boldsymbol{M'}\right\| + \|\boldsymbol{B}_{\mathcal{O}_K} \cdot \boldsymbol{M'} - \boldsymbol{B}_{\mathcal{O}_K} \cdot \boldsymbol{M}\| \\ \leq & 2^{-p} \cdot \|\boldsymbol{M'}\| + 2^{-p} \cdot \|\boldsymbol{B}_{\mathcal{O}_K}\| \\ = & \operatorname{poly}(d) \cdot 2^{d-p} \cdot |\Delta_K|^{1/(2d)} = 2^{O(d)} \cdot 2^{-p} \cdot |\Delta_K|^{1/(2d)}. \end{aligned} \tag{3}$$

4.10 QR factorization of Step 6

The R-part of the QR decomposition of $\boldsymbol{B}'_{\mathfrak{b}}$ (which we denote $\boldsymbol{R}'_{\mathfrak{b}}$) is computed using the Householder algorithm on $\boldsymbol{B}'_{\mathfrak{b}}$ with fixed point precision 2^{-p} in black box.

Complexity Classically, computing the R-factor of a matrix of size $n \times n$ using Householder algorithm takes $O(d^3)$ real multiplications and additions, leading to a classical complexity of $O\left(d^3 \cdot \left(p + \ln\left(|\Delta_K|^{1/d}\right)\right)^{1+o(1)}\right)$, the number of quantum memory and gates is then the same as the classical complexity, up to a constant factor.

Error analysis In Appendix E.1, we prove that there exists an absolute constant $C_1 > 0$ such that $\left\|\boldsymbol{R}'_{\mathfrak{b}} \cdot \boldsymbol{R}_{\mathfrak{b}}^{-1} - \boldsymbol{I}\right\| \leq 2^{C_1 d} \cdot 2^{-p} \cdot |\Delta_K|^{1/(2d)}$, where $\boldsymbol{R}_{\mathfrak{b}} = \mathrm{QR}(\boldsymbol{B}_{\mathfrak{b}})$.

4.11 Computing the Gaussian superposition of Step 7

We propose a quantum implementation of the GPV [GPV08] algorithm. In Appendix L, we summarize the results needed for our analysis. The Gaussian superposition is computed by Algorithm 6.1 with error parameter $\varepsilon/2$, periodization parameter q, deviation $2^p \cdot \sigma$, matrix $2^p \cdot \boldsymbol{R}'_{\mathfrak{b}} \in \mathbb{Z}^{d \times d}$. At the end of this step, the main state of the algorithm is approximates (with C a normalization factor)

$$C^{-1} \sum_{\substack{\mathbf{z} \in \mathbb{Z}^d \setminus \{0\}, \\ \|\boldsymbol{B}_{\mathfrak{b}} \cdot \boldsymbol{z}\| \leq R}} \rho_\sigma(\boldsymbol{B}_{\mathfrak{b}} \cdot \boldsymbol{z})|\boldsymbol{z}\rangle \tag{4}$$

Complexity analysis By Lemma L.5 and the fact that $\|\boldsymbol{R}'_{\mathfrak{b}}\| \leq 2^{O(d)} \cdot |\Delta_K|^{1/(2d)}$, the implementation uses

$$\tilde{O}\left(d^2 \cdot \left(\ln(q) + d + \ln\left(|\Delta_K|^{1/d}\right) + p + \operatorname{size}(\sigma)\right)^{1+o(1)} + d \cdot \log_2(q) \cdot (\ln(1/\varepsilon))^{3/2}\right)$$

quantum gates and

$$O\left(d \cdot \left(\ln(q) + \ln\left(|\Delta_K|^{1/d}\right) + p + \mathrm{size}(\sigma)\right)^{1+o(1)} + \ln(1/\varepsilon)\right)$$

memory qubits.

Error analysis Theorem 6.1 and the fact that $\mathrm{cond}(\boldsymbol{R}_\mathfrak{b}) = 2^{O(d)}$ imply that as long as ε, σ, R and q satisfy

- $2^{C_1 d} \cdot 2^{-p} \cdot |\Delta_K|^{1/(2d)} \leq \varepsilon^2/(64d)$,
- $\sigma \geq \mathrm{poly}(d) \cdot \varepsilon^{-4/d} \cdot 2^d \cdot |\Delta_K|^{1/(2d)}$,
- $R = \sqrt{\ln(2/\varepsilon) \cdot d} \cdot \sigma$,
- q is a power-of-two larger than $\varepsilon^{-4/d} \cdot 2^{O(d)}$,

then the trace distance between Eq. (4) and $|\phi'\rangle$ is less than $\varepsilon/2$. From now-on, we fix q to be the smallest power of two satisfying this condition.

4.12 Update of the coordinates in Step 8

We apply the linear transformation in order to bring the output state close to

$$|\psi\rangle = C'^{-1} \sum_{\substack{\boldsymbol{z} \in \mathbb{Z}^d \setminus \{0\}, \\ \|\boldsymbol{B}_\mathfrak{b} \cdot \boldsymbol{z}\| \leq R}} \rho_\sigma(\boldsymbol{B}_\mathfrak{b} \cdot \boldsymbol{z}) |\boldsymbol{B}'_\mathfrak{b} \cdot \boldsymbol{z}\rangle, \tag{5}$$

where $|x\rangle$ is the representation of $x \in K_\mathbb{R}$ as a d-dimensional vector of complex numbers represented in fixed points with precision 2^{-p}. This register has size $O(d(p + \ln(R)))$. The complexity of this step is negligible compared to the rest of the computations.

Error analysis Since the update is a trace-preserving operation over the qubits, the error between $|\psi\rangle$ and $|\psi'\rangle$ is carried from the previous computation, so it is less than $\varepsilon/2$.

Encoding the elements of $\boldsymbol{K}_\mathbb{R}$ in Step 9 Let $\mathfrak{b}' = \mathrm{ExpEx}(\boldsymbol{s}, \boldsymbol{\theta}) \cdot \mathfrak{b}/\mathcal{N}(\mathfrak{b})^{1/d}$. By multiplying by the phases and applying Enc' on the elements register ($|\boldsymbol{B}'_\mathfrak{b} \cdot \boldsymbol{z}\rangle$ in Eq. (5)), the state approximates $F_{R,\sigma}(\mathfrak{b}')$, where $F_{R,\sigma}$ is defined in Section 3.

Complexity analysis The complexity in terms of quantum gates and memory of this step is exactly the same as the one to compute Enc with precision 2^{-p}, it is negligible compared to the rest of the computation.

Error analysis The error analysis is described in Appendix E.2. We give here the final result: the distance between $|\psi'\rangle$ and $F_{R,\sigma}(\mathfrak{b}')$ is less than $2^{-p} + \varepsilon/2$.

4.13 Final distance to $G_{R,\sigma}$

We have that $G_{R,\sigma}(\boldsymbol{x},\boldsymbol{\theta},\boldsymbol{s},\boldsymbol{a}) = F_{R,\sigma}(I')$, with $I' = \mathrm{ExpEx}(\boldsymbol{s},\boldsymbol{\theta})\cdot I$ with the notation of the previous subsection. The state computed by Algorithm 4.1 is an approximation of $F_{R,\sigma}(\mathfrak{b}')$, the distance between I' and $\mathfrak{b}'$ is bounded by 2^{-p} so we have that as long R,σ,ν and Enc follow the hypothesis of Theorem 3.1 for some a,ν',ε', then $F_{R,\sigma}$ is

$$\||\psi'\rangle - G_{R,\sigma}(\boldsymbol{x},\boldsymbol{\theta},\boldsymbol{s},\boldsymbol{a})\| \leq 2^{-p} + \varepsilon/2 + (5d+10a)\cdot 2^{-p} + 4\exp\bigl(-(R/\sigma)^2/2\bigr). \tag{6}$$

5 Parameters and final complexity

In order to simplify computations, we assume that $s = |S| = \mathrm{poly}(d, \ln(|\Delta_K|))$ (else, the number of considered ideals is superpolynomial), and $N_S \leq |\Delta_K|^{O(1)} \cdot 2^{d^{O(1)}}$ in order to have polynomial-sized ideals[3]. Note that this is not needed for the algorithm to work and terminate and for the polynomial complexity.

5.1 Choosing σ, R and p

Let σ, R, ν, p and ε the parameters of Algorithm 4.1. For this analysis, we instantiate $\mathrm{Enc} = \mathrm{Enc}_{R,t}$ where $\mathrm{Enc}_{R,t}$ is defined in Definition M.3. We will fix the parameter t in this section. Note that by Lemma M.3, it is $2\sqrt{d}\cdot t$ totally separative and $a = \sqrt{d}\cdot\pi/(2t)$-Lipschitz. By Corollary M.1, it is also $(2\sqrt{d}\cdot t/(\sqrt{10}\cdot\pi), 1-\varepsilon')$ separative with $\varepsilon' = 1/30$. The parameters need to satisfy several hypotheses. First, we fix σ, it needs to satisfy $\sigma \geq 3\cdot d^{3/2}\cdot|\Delta_K|^{3/(2d)}$ for Theorem 3.1 and $\sigma \geq 2^{O(d)}\cdot\varepsilon^{-4/d}|\Delta_K|^{1/2d}$ for Corollary L.2. There exists an absolute constant $c_\sigma \geq 1$ such that fixing

$$\sigma = (2^d\cdot\varepsilon^{-1/d})^{c_\sigma}\cdot|\Delta_K|^{3/(2d)}$$

satisfies all these conditions. We now fix R. It needs to satisfy

- $R \geq 2\cdot\sigma\sqrt{d\ln(32\sigma)}$ for Theorem 3.1,
- $R \geq O(\sqrt{\ln(1/\varepsilon)\cdot d}/\pi)\cdot\sigma$ for the error in the almost-Lipschitz continuity of Theorem 3.1, equal to $4\exp\bigl(-(R/\sigma)^2/2\bigr)$ to be smaller than $\varepsilon/4$,
- $R \geq \sqrt{\ln(2/\varepsilon)\cdot d}\cdot\sigma$ for Theorem 6.1 with error parameter $\varepsilon/2$.

Note that for any $d \geq 2, \varepsilon < 1$, we have[4] that $\ln(1/\varepsilon) \leq d/e\cdot\varepsilon^{-1/d}$. There exists an absolute constant $c_R \geq 1$ such that fixing

$$R = \left(2^d\cdot\varepsilon^{-1/d}\right)^{c_R}\cdot|\Delta_K|^{1/d}$$

satisfies all these conditions. Now we fix t. Let $\nu = 2\sqrt{d}\cdot t$ and $\nu' = 2\sqrt{d}\cdot t/(\sqrt{10}\cdot\pi)$, they need to satisfy

[3] Note that in order to compute the whole class group, assuming GRH [Bac90], it suffices to take S to be the set of all prime ideals of norm $\leq 12(\ln|\Delta_K|)^2$.

[4] By $\ln(A)/d = \ln\left(A^{1/d}\right) \leq A^{1/d}/e$ for all $A \in \mathbb{R}_{>0}$ and $d \in \mathbb{Z}_{>0}$ and putting $A = 1/\varepsilon$.

– $\nu \leq 1/(2R)$ for Theorem 3.1 (where $\nu = 2\sqrt{d} \cdot t$);
– $\nu' \leq \mathrm{poly}(d) \cdot s \cdot \ln(N_S))^{-1}$ to satisfy Theorem 2.2.

There exists an absolute constant $c_t \geq 1$ such that fixing

$$t = \left(\varepsilon^{1/d} \cdot 2^{-d} \cdot |\Delta_K|^{-1/(2d)}\right)^{c_t}$$

satisfies all these conditions. Finally, we fix the precision parameter p of Algorithm 4.1. The conditions it needs to satisfy are

– $2^{-p} \leq \varepsilon^2 \cdot \mathrm{poly}(d)^{-1} \cdot 2^{-O(d)} \cdot |\Delta_K|^{-1/d}$ for Theorem 6.1,
– $2^{-p}+(5d+10a)\cdot 2^{-p}+3\varepsilon/4 \leq \varepsilon$, where $a = \sqrt{d}\cdot\pi/(2t)$ for the right-hand-side of Eq. (6) to be less than ε.

So there exists an absolute constant $c_p \geq 1$ such that fixing

$$p = c_p \cdot \left(d + \ln\left(|\Delta_K|^{1/d}\right) + \ln(1/\varepsilon)\right)$$

satisfy those conditions.

We do amplify a constant number of time the function $G_{R,\sigma}$, as noted in Lemmas 4.1 and 4.2. This has no impact on the parameter choice and the complexity, up to a constant factor.

5.2 Final complexity of the quantum oracle

In this subsection, we compute the final complexity of Algorithm 4.1 for the previously computed values of σ, R, t, p, and the value of Q of Theorem 2.2. We also take our error parameter to be $\varepsilon = 2^{-\Theta(d)} \cdot |\Delta_K|^{\Theta(1)}$. This implies that we take $\sigma = (2^d \cdot |\Delta_K|^{1/d})^{O(1)}$, $R = (2^d \cdot |\Delta_K|^{1/d})^{O(1)}$, $t = (2^{-d} \cdot |\Delta_K|^{-1/d})^{\Omega(1)}$, $p = O\left(d + \ln(|\Delta_K|)\right)$, $q = O\left(d + \ln\left(|\Delta_K|^{1/d}\right)\right)$. Now, using Proposition G.1, we fix $\tau = 2^{-\Theta((d+s)^2 \ln(d+s))} \cdot |\Delta_K|^{-\Theta(d+s)}$ to be the error parameter of Theorem 2.2, which gives

$$Q = O\left((d+s)^{2+o(1)} \left(\ln\left(|\Delta_K|^{1/d}\right) + d\right)\right)$$

We now bound the complexity of each step of Algorithm 4.1 for the parameter values we computed. For readability we will omit the O notation. We have that

$$m = Q \cdot (d_{\mathbb{R}} + d_{\mathbb{C}} + s) = (d+s)^{3+o(1)} \left(\ln\left(|\Delta_K|^{1/d}\right) + d\right),$$

and

$$\ln(B_{K,p,m,Q}) = d(d+s)^{2+o(1)} \left(\ln\left(|\Delta_K|^{1/d}\right) + d\right).$$

For the sake of brevity, we defer the step-by-step description of the costs to Appendix F.

Overall complexity The overall complexity of computing G within precision $|\Delta_K|^{-\Theta(1)} \cdot 2^{-\Theta(d)}$ is dominated by Steps 3 and 4.

Theorem 5.1 (Overall complexity of the oracle computation). *With all the parameters fixed in Section 5, it holds that the complexity of Algorithm 4.1 is:*
Quantum memory:

$$\begin{aligned}&\mathrm{LLLMem}\left(d, (d+s)^{5+o(1)} \cdot \left(\ln\left(|\Delta_K|^{1/d}\right) + d\right)^{2+o(1)}\right)\\ &+O\left(d^{\omega+2+o(1)} \cdot (d+s)^{5+o(1)} \cdot \left(\ln\left(|\Delta_K|^{1/d}\right) + d\right)^{2+o(1)}\right).\end{aligned}$$

Gate count:

$$\begin{aligned}&\mathrm{LLLGates}\left(d, (d+s)^{5+o(1)} \cdot \left(\ln\left(|\Delta_K|^{1/d}\right) + d\right)^{2+o(1)}\right)\\ &+O\left(d^{\omega+2+o(1)} \cdot (d+s)^{8+o(1)} \cdot \left(\ln\left(|\Delta_K|^{1/d}\right) + d\right)^{3+o(1)}\right).\end{aligned}$$

Omitting big-O notation again, the output space is of size (see Eq. (40))

$$\ln\big(\dim(\mathcal{H}_{\nu/2})\big) = d \cdot (d + \ln\left(|\Delta_K|^{1/d}\right)),$$

and the Lipschitz constant A of G satisfies

$$\ln(A) = d + \ln\left(|\Delta_K|^{1/d}\right).$$

Corollary 5.1. *The complexity of the quantum procedure of Theorem 2.2 running with this paper's quantum implementation of the oracle G is:*
Quantum memory:

$$\begin{aligned}&\mathrm{LLLMem}\left(d, (d+s)^{5+o(1)} \cdot \left(\ln\left(|\Delta_K|^{1/d}\right) + d\right)^{2+o(1)}\right)\\ &+O\left(d^{\omega+2+o(1)} \cdot (d+s)^{5+o(1)} \cdot \left(\ln\left(|\Delta_K|^{1/d}\right) + d\right)^{2+o(1)}\right).\end{aligned}$$

Gate count:

$$\begin{aligned}&\mathrm{LLLGates}\left(d, (d+s)^{5+o(1)} \cdot \left(\ln\left(|\Delta_K|^{1/d}\right) + d\right)^{2+o(1)}\right)\\ &\cdot O\left((d+s)^{1+o(1)}\left(\ln\left(|\Delta_K|^{1/d}\right) + d\right)\right)\\ &+O\left(d^{\omega+2+o(1)} \cdot (d+s)^{9+o(1)} \cdot \left(\ln\left(|\Delta_K|^{1/d}\right) + d\right)^{4+o(1)}\right).\end{aligned}$$

The proof of this corollary can be found in Appendix B.

6 An efficient and numerically stable quantum circuit for the GPV algorithm, computing a Gaussian lattice superposition

In this section, we will present an efficient algorithm, with a novel analysis, to compute an approximation of the Gaussian quantum state

$$c_{\Lambda,\sigma}^{-1} \sum_{\mathbf{z}\in\mathbb{G}_Q} \rho_{\sigma,\mathbf{c}}(\boldsymbol{B}\mathbf{z})|\boldsymbol{z}\rangle,$$

where $\mathbb{G}_Q = \{-2^Q/2, \ldots, 0, \ldots, (2^Q-1)/2\}^n \setminus \mathbf{0} \subseteq \mathbb{Z}^n$, a centered set of representatives of $(\mathbb{Z}/2^Q\mathbb{Z})^n$, with $Q \in \mathbb{Z}_{>0}$. This will be done by combining two techniques; one of Kitaev and Webb [KW09], which allows to compute a Gaussian superposition over $\mathbb{Z}$ efficiently; and one of Gentry, Peikert and Vaikuntanathan [GPV08], which is a classical technique computing general discrete Gaussian distributions from Gaussian distributions over $\mathbb{Z}$. Though this latter technique is classical, it is here amended for our quantum setting. As said in the introduction, Algorithm 6.1 does not differ significantly from previous descriptions of quantum GPV [Bra18,LMZ23], but its precise error and complexity analysis is new to our knowledge.

Recall the definition of the Gaussian function from Section 2.4. For $\boldsymbol{c}, \boldsymbol{x} \in \mathbb{R}^m$, we denote $\rho_{\sigma,\boldsymbol{c}}(\boldsymbol{x}) = e^{-\pi\|\boldsymbol{x}-\boldsymbol{c}\|^2/\sigma^2}$.

Theorem 6.1. *For any $\varepsilon \in (0, 2^{-n})$, and any non-singular upper triangular matrices $\boldsymbol{R}, \boldsymbol{R}' \in \mathbb{Z}^{n\times n}$ with positive diagonal satisfying $\left\|\boldsymbol{R}'\boldsymbol{R}^{-1} - \boldsymbol{I}\right\| \leq \varepsilon^2/(16n) \leq 1$, and $\sigma \geq \sqrt{2\cdot\ln(64n^3/\varepsilon^2)}\cdot\|\boldsymbol{R}\|$, then the output of Algorithm 6.1 on input $(\boldsymbol{R}', \sigma, R, q, \boldsymbol{c} = \mathbf{0})$ is ε-close to the state*

$$C'^{-1} \sum_{\substack{\mathbf{z}\in\mathbb{Z}^n,\\ \|\boldsymbol{R}\boldsymbol{z}\|\leq R}} \rho_\sigma(\boldsymbol{R}\cdot\boldsymbol{z})|\boldsymbol{z}\rangle, \tag{7}$$

where $R = \sqrt{\ln(1/\varepsilon)\cdot n}\cdot\sigma$ and q is the smallest power of two such that

$$q \geq \sqrt{2n\cdot\ln(1/\varepsilon)\cdot\ln(64n^3/\varepsilon^2)}\cdot\|\boldsymbol{R}\|\cdot\mathrm{cond}(\boldsymbol{R}).$$

Moreover, Algorithm 6.1 uses $\tilde{O}\left(n^2\cdot\beta^{1+o(1)} + n\cdot\log_2(q)\cdot(\ln(1/\varepsilon))^{3/2}\right)$ quantum gates and $O\left(n\cdot\beta^{1+o(1)} + \ln(1/\varepsilon)\right)$ ancillary qubits, where $\beta = \ln(n\cdot q\cdot\|\boldsymbol{R}\|) + \max_i(\mathit{size}(\boldsymbol{c}_i)) + \mathit{size}(\sigma)$.

Algorithm 6.1 QuantumGaussian

Input: An upper triangular invertible matrix $\boldsymbol{R} = [\boldsymbol{r}_1, \ldots, \boldsymbol{r}_n] \in \mathbb{Z}^{n \times n}$ with positive diagonal, a center $\boldsymbol{c} \in \mathbb{Z}^n$, a deviation $\sigma \in \mathbb{Q}_{>0}$ and a window parameter $q \in \mathbb{Z}_{>0}$ that is a power of 2.
Output: A quantum state ε-close in the trace distance to

$$C^{-1} \sum_{\boldsymbol{z} \in (\mathbb{Z}/q\mathbb{Z})^n} \xi_{\sigma,\boldsymbol{c},q}(\boldsymbol{R}, \boldsymbol{z}) |\boldsymbol{z}\rangle,$$

where $C \in \mathbb{R}_{>0}$ satisfies $C^2 = \sum_{\boldsymbol{z} \in \mathbb{Z}^n} \rho_\sigma(\boldsymbol{R}\boldsymbol{z})^2$.

1: Initialize $|\boldsymbol{R}\rangle|\boldsymbol{c}\rangle|\sigma\rangle|0\rangle|\boldsymbol{0}\rangle$.

2: Compute $c' = \frac{\boldsymbol{c}_n}{\boldsymbol{r}_{n,n}}$ and $(\sigma')^2 = \sigma^2/\boldsymbol{r}_{n,n}^2$. Put it in the state

$$|\boldsymbol{B}\rangle|\boldsymbol{c}\rangle|\sigma\rangle|c', (\sigma')^2\rangle|\boldsymbol{0}\rangle$$

3: Use $\mathrm{QGauss}_{\mathbb{Z}}^{(q,\varepsilon/(2n))}$ the periodized discrete Gaussian over $\mathbb{Z}$ with center c' and deviation σ' within trace distance $\varepsilon/(2n)$.

$$|\boldsymbol{R}\rangle|\boldsymbol{c}\rangle|\sigma\rangle \left(|c', (\sigma')^2\rangle \cdot C_0^{-1} \sum_{z \in \mathbb{Z}/q\mathbb{Z}} \tilde{\xi}_{\sigma',c',q}(1, z)|z\rangle \right) |\boldsymbol{0}\rangle,$$

where $C_0 \in \mathbb{R}_{>0}$ satisfies $C_0^2 = \sum_{z \in \mathbb{Z}} \rho_{\sigma',c'}(z)^2$ and $\tilde{\xi}$ signifies that it is a close approximation of ξ.

4: Then uncompute c' and $(\sigma')^2$ to obtain

$$|\boldsymbol{R}\rangle|\boldsymbol{c}\rangle|\sigma\rangle|0\rangle \left(C_0^{-1} \sum_{z \in \mathbb{Z}/q\mathbb{Z}} \tilde{\xi}_{\sigma',c',q}(1, z)|z\rangle \right) |\boldsymbol{0}\rangle$$

5: Compute $|\boldsymbol{c}\rangle|\sigma\rangle|0\rangle|z\rangle \mapsto |\boldsymbol{c}_z\rangle|\sigma\rangle|0\rangle|z\rangle$, where $\boldsymbol{c}_z := \boldsymbol{c} - z\boldsymbol{r}_n$. Then we obtain

$$|\boldsymbol{R}\rangle \left(C_0^{-1} \sum_{z \in \mathbb{Z}/q\mathbb{Z}} \tilde{\xi}_{\sigma',c',q}(1, z)|\boldsymbol{c}_z\rangle|\sigma\rangle|0\rangle|z\rangle \right) |\boldsymbol{0}\rangle$$

6: Recursively, use the first $n-1$ basis vectors of $\boldsymbol{R}$, the center $\boldsymbol{c'}_z = \pi_{n-1}(\boldsymbol{c} - z\boldsymbol{r}_n)$ (where π_{n-1} is the projection to the first $n-1$ coordinates) and deviation σ (using the ancilla space of σ', c') to obtain the periodized discrete Gaussian over $\boldsymbol{R}_\circ = (\boldsymbol{r}_1, \ldots, \boldsymbol{r}_{n-1})$ within trace distance $\frac{(n-1)\varepsilon}{n}$, yielding

$$|\boldsymbol{R}\rangle \cdot C_0^{-1} \sum_{z \in \mathbb{Z}/q\mathbb{Z}} \tilde{\xi}_{\sigma',c',q}(1, z)|\boldsymbol{c}_z\rangle|\sigma\rangle|0\rangle|z\rangle \cdot C_z^{-1} \cdot \sum_{\boldsymbol{z}_\circ \in (\mathbb{Z}/q\mathbb{Z})^{n-1}} \tilde{\xi}_{\sigma,\boldsymbol{c}_z,q}(\boldsymbol{R}_\circ, \boldsymbol{z}_\circ)|\boldsymbol{z}_\circ\rangle,$$

where $C_z \in \mathbb{R}_{>0}$ satisfies $C_z^2 = \sum_{\boldsymbol{z}_\circ \in \mathbb{Z}^{n-1}} \rho_{\sigma,\boldsymbol{c}_z}(\boldsymbol{R}_\circ \boldsymbol{z}_\circ)^2$, and $\tilde{\xi}$ signifies that it is a close approximation of ξ.

7: Uncompute the shifts of the center $\boldsymbol{c}$, i.e., $|\boldsymbol{c}_z\rangle|\sigma\rangle|0\rangle|z\rangle \mapsto |\boldsymbol{c}\rangle|\sigma\rangle|0\rangle|z\rangle$, to obtain

$$|\boldsymbol{R}\rangle|\boldsymbol{c}\rangle|\sigma\rangle|0\rangle C_0^{-1} \sum_{z \in \mathbb{Z}/q\mathbb{Z}} \tilde{\xi}_{\sigma',c',q}(1, z)|z\rangle C_z^{-1} \sum_{\boldsymbol{z}_\circ \in (\mathbb{Z}/q\mathbb{Z})^{n-1}} \tilde{\xi}_{\sigma,\boldsymbol{c}_z,q}(\boldsymbol{R}_\circ, \boldsymbol{z}_\circ)|\boldsymbol{z}_\circ\rangle.$$

8: Output the resulting state.

References

Bab85. L. Babai. On Lovász' lattice reduction and the nearest lattice point problem. In *STACS*, 1985.

Bac90. E. Bach. Explicit bounds for primality testing and related problems. *Mathematics of Computation*, 55(191):355–380, 1990.

Ban93. W. Banaszczyk. New bounds in some transference theorems in the geometry of numbers. *Mathematische Annalen*, 296, 1993.

BBTV24. N. Bindel, X. Bonnetain, M. Tiepelt, and F. Virdia. Quantum lattice enumeration in limited depth. In *CRYPTO*, 2024.

BK96. J. Buchmann and V. Kessler. Computing a reduced lattice basis from a generating system. *Unpublished Manuscript*, 08 1996.

Boe22. K. Boer. *Random walks on Arakelov class groups*. PhD thesis, Leiden University, 2022.

BP23. R. Barbulescu and A. Poulalion. The special case of cyclotomic fields in quantum algorithms for unit groups. In *AFRICACRYPT*, Lecture notes in computer science, 2023.

BPW25. K. Boer, A. Pellet-Mary, and B. Wesolowski. Rigorous methods for computational number theory. Cryptology ePrint Archive, Paper 2025/1514, 2025. URL https://eprint.iacr.org/2025/1514.

Bra18. Z. Brakerski. Quantum fhe (almost) as secure as classical. In *CRYPTO*, 2018.

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

BS25v1. J.-F. Biasse and F. Song. An efficient quantum algorithm for computing S-units and its applications. (arXiv:2510.02280v1), Oct. 2025. https://doi.org/10.48550/arXiv.2510.02280v1. arXiv:2510.02280v1 [cs].

BS25v2. J.-F. Biasse and F. Song. An efficient quantum algorithm for computing S-units and its applications. (arXiv:2510.02280v2), Nov. 2025. https://doi.org/10.48550/arXiv.2510.02280v2. arXiv:2510.02280v2 [cs].

CDPR16. R. Cramer, L. Ducas, C. Peikert, and O. Regev. Recovering short generators of principal ideals in cyclotomic rings. In *EUROCRYPT*, 2016.

CDW17. R. Cramer, L. Ducas, and B. Wesolowski. Short Stickelberger class relations and application to Ideal-SVP. In *EUROCRYPT*, 2017.

Coh93. H. Cohen. *Algorithms for Algebraic Number Theory II*. Graduate Texts in Mathematics. Springer, 1993.

Cop02. D. Coppersmith. An approximate fourier transform useful in quantum factoring, 2002, quant-ph/0201067. URL https://arxiv.org/abs/quant-ph/0201067.

CSV12. X.-W. Chang, D. Stehlé, and G. Villard. Perturbation analysis of the QR factor R in the context of LLL lattice basis reduction. *Mathematics of Computation*, 81(279):1487–1511, 2012.

CW81. D. Coppersmith and S. Winograd. On the asymptotic complexity of matrix multiplication. In *22nd Annual Symposium on Foundations of Computer Science (sfcs 1981)*, pages 82–90, 1981. https://doi.org/10.1109/SFCS.1981.27.

dBDF20. K. de Boer, L. Ducas, and S. Fehr. On the quantum complexity of the continuous hidden subgroup problem. In *EUROCRYPT*, 2020.

dBDPW20. K. de Boer, L. Ducas, A. Pellet-Mary, and B. Wesolowski. Random self-reducibility of Ideal-SVP via Arakelov random walks. In *CRYPTO*, 2020.

EHKS14a. K. Eisenträger, S. Hallgren, A. Kitaev, and F. Song. A quantum algorithm for computing the unit group of an arbitrary degree number field. In *Proceedings of the forty-sixth annual ACM symposium on Theory of computing*. ACM, 2014.

EHKS14b. K. Eisenträger, S. Hallgren, A. Kitaev, and F. Song. A quantum algorithm for computing the unit group of an arbitrary degree number field. Available at: https://www.cse.psu.edu/~sjh26/units-stoc-submission.pdf, 2014. Full version, Accessed: September 9th, 2025.

FPSW23. J. Felderhoff, A. Pellet-Mary, D. Stehlé, and B. Wesolowski. Ideal-SVP is hard for small-norm uniform prime ideals. In *TCC*, 2023.

GPV08. C. Gentry, C. Peikert, and V. Vaikuntanathan. Trapdoors for hard lattices and new cryptographic constructions. In *STOC*, 2008.

Hal07. S. Hallgren. Polynomial-time quantum algorithms for pell's equation and the principal ideal problem. *J. ACM*, 54(1), Mar. 2007. https://doi.org/10.1145/1206035.1206039.

HH21. D. Harvey and J. v. d. Hoeven. Integer multiplication in time $O(n \log n)$. *Annals of Mathematics*, 193(2):563–617, 2021.

HR14. I. Haviv and O. Regev. On the lattice isomorphism problem. In *SODA*, 2014.

Kes91. V. Kessler. On the minimum of the unit lattice. *Séminaire de Théorie des Nombres de Bordeaux*, 3(2):377–380, 1991. URL http://www.jstor.org/stable/44165664.

KW09. A. Kitaev and W. A. Webb. Wavefunction preparation and resampling using a quantum computer, 2009, 0801.0342.

LJS90. J. C. Lagarias, H. W. L. Jr., and C.-P. Schnorr. Korkin-Zolotarev bases and successive minima of a lattice and its reciprocal lattice. *Combinatorica*, 10(4):333–348, 1990.

LLL82. A. K. Lenstra, H. W. Lenstra, and L. Lovász. Factoring polynomials with rational coefficients. *Mathematische Annalen*, 1982.

LMZ23. J. Liu, H. Montgomery, and M. Zhandry. Another round of breaking and making quantum money: How to not build it from lattices, and more. In *QIP*, 2023.

Lou00. 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.

LPSW19. C. Lee, A. Pellet-Mary, D. Stehlé, and A. Wallet. An LLL algorithm for module lattices. In *ASIACRYPT*, 2019.

ME99. M. Mosca and A. Ekert. The hidden subgroup problem and eigenvalue estimation on a quantum computer. In C. P. Williams, editor, *Quantum Computing and Quantum Communications*, pages 174–188, Berlin, Heidelberg, 1999. Springer Berlin Heidelberg.

MG02. D. Micciancio and S. Goldwasser. *Complexity of lattice problems: a cryptographic perspective*, volume 671. Springer Science & Business Media, 2002.

MR07. D. Micciancio and O. Regev. Worst-case to average-case reductions based on gaussian measures. *SIAM Journal on Computing*, 37(1):267–302, 2007.

NC10. M. A. Nielsen and I. L. Chuang. *Quantum Computation and Quantum Information: 10th Anniversary Edition*. Cambridge University Press, 2010.

Neu13. J. Neukirch. *Algebraic number theory*, volume 322. Springer Science & Business Media, 2013.

Ng24. I.-I. Ng. Upper bounding the quantum space complexity for computing class group and principal ideal problem. (arXiv:2405.12508), May 2024. https://doi.org/10.48550/arXiv.2405.12508. arXiv:2405.12508 [quant-ph].

NIS16. Submission requirements and evaluation criteria for the post-quantum cryptography standardization process, 2016. https://csrc.nist.gov/CSRC/media/Projects/Post-Quantum-Cryptography/documents/call-for-proposals-final-dec-2016.pdf.

NIS25. National institute of standards and technology: Post-quantum cryptography standardization, 2025. URL https://csrc.nist.gov/Projects/Post-Quantum-Cryptography/.

NS09. P. Q. Nguyen and D. Stehlé. An LLL algorithm with quadratic complexity. *SIAM Journal on Computing*, 2009.

NS13. J. Neukirch and N. Schappacher. *Algebraic Number Theory*. Grundlehren der mathematischen Wissenschaften. Springer Berlin Heidelberg, 2013. URL https://books.google.nl/books?id=hS3qCAAAQBAJ.

NV09. P. Q. Nguyen and B. Valle. *The LLL Algorithm: Survey and Applications*. Springer Publishing Company, Incorporated, 1st edition, 2009.

PHS19. A. Pellet-Mary, G. Hanrot, and D. Stehlé. Approx-SVP in ideal lattices with pre-processing. In *EUROCRYPT*, 2019.

PMS21. A. Pellet-Mary and D. Stehlé. On the hardness of the ntru problem. In *ASIACRYPT*, 2021.

PRSD17. C. Peikert, O. Regev, and N. Stephens-Davidowitz. Pseudorandomness of ring-lwe for any ring and modulus. In *Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing*, pages 461–473, 2017.

Sho94. P. Shor. Algorithms for quantum computation: discrete logarithms and factoring. In *Proceedings 35th Annual Symposium on Foundations of Computer Science*, 1994.

Sho97. P. W. Shor. Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. *SIAM Journal on Computing*, 1997.

SL96. 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. Association for Computing Machinery, 1996.

Str69. V. Strassen. Gaussian elimination is not optimal. *Numer. Math.*, 1969.

TS19. M. Tiepelt and A. Szepieniec. Quantum LLL with an application to mersenne number cryptosystems. In *LATINCRYPT*, 2019.

Wil17. M. Wilde. *Quantum Information Theory*. Cambridge University Press, 2017.

WXXZ. V. V. Williams, Y. Xu, Z. Xu, and R. Zhou. *New Bounds for Matrix Multiplication: from Alpha to Omega*, pages 3792–3835. https://epubs.siam.org/doi/pdf/10.1137/1.9781611977912.134. https://doi.org/10.1137/1.9781611977912.134.

A Extended preliminaries

Lemma A.1 (Smoothing lemma, see the proof of [MR07, Lemma 4.4]). *Let $\mathcal{L} \subset \mathbb{R}^d$ be lattice. Then, for any $\varepsilon > 0$ and $\sigma > \eta_\varepsilon(\mathcal{L})$ we have*

$$(1-\varepsilon)\frac{\sigma^d}{\det(\mathcal{L})} \leq \rho_\sigma(\mathcal{L}) \leq (1+\varepsilon)\frac{\sigma^d}{\det(\mathcal{L})}$$

Lemma A.2 (Banaszczyk's tail bound). *Let* $\beta_d(\kappa) := \left(\frac{2\pi e\kappa^2}{d}\right)^{d/2} \exp\left(-\pi\kappa^2\right)$. *Let* $\sigma > 0$ *and* $R \geq \sigma \cdot \sqrt{d}$. *Then for all lattices* $\mathcal{L} \subset \mathbb{R}^d$,

$$\frac{\rho_\sigma(\mathcal{L} \setminus B(0,R))}{\rho_\sigma(\mathcal{L})} \leq \beta_d(R/\sigma) \leq e^{-(R/\sigma)^2}. \tag{8}$$

Proof. The proof of the left-most inequality of Equation (8) is by Banaszczyk [Ban93, Lemma 1.5]. The right-most inequality follows from the following computation for $c > 1$.

$$\beta_d(\sqrt{d}c) = (\sqrt{2\pi e} \cdot c \cdot e^{-\pi c^2})^d \leq (e^{-c^2})^d = e^{-c^2 d},$$

and hence $\beta_d(R/\sigma) \leq e^{-(R/\sigma)^2}$ whenever $R/\sigma \geq \sqrt{d}$.

B Technical proofs

Lemma 2.4. *Let* $I \in \mathrm{IdLat}_K$ *and* $R \geq \mathcal{N}(I)^{1/d}$. *Then for any* $x \in I \setminus \{0\}$, $y \in I|_R \setminus \{0\}$ *with* $x \neq y$, *it holds that* $\delta_{K_\mathbb{R}^\times}(x,y) \geq \mathcal{N}(I)^{1/d}/2R$.

Proof. Let i maximizing $\ln|\sigma_i(x)/\sigma_i(y)|$. Then we have $\delta_{K_\mathbb{R}^\times}(x,y) \geq \|\mathrm{Log}(x) - \mathrm{Log}(y)\|_2 \geq \|\mathrm{Log}(x) - \mathrm{Log}(y)\|_\infty \geq \ln|\sigma_i(x)/\sigma_i(y)|$. Let $s \in \{-1,1\}$ be the sign of $|\sigma_i(x)| - |\sigma_i(y)|$. Then, by Lemma 2.3 and the fact that $s(|\sigma_i(x)| - |\sigma_i(y)|) \geq \lambda_1(I)/\sqrt{d}$ (by the inequality between infinity norms and Euclidean norms),

$$\begin{aligned}\delta_{K_\mathbb{R}^\times}(x,y) &\geq s\ln\left|1 + \frac{|\sigma_i(x)| - |\sigma_i(y)|}{|\sigma_i(y)|}\right| \geq \ln\left|1 + \frac{\lambda_1(I)}{\sqrt{d}|\sigma_i(y)|}\right| \geq \ln\left(1 + \frac{\mathcal{N}(I)^{1/d}}{R}\right) \\ &\geq \frac{\mathcal{N}(I)^{1/d}}{2R},\end{aligned}$$

where the last inequality holds since for $x \in (0,1), \ln(1+x) \geq x/2$. □

Lemma 2.5. *Let* $R \geq \sqrt{d}$, *and let* $I, J \in \mathrm{IdLat}_K^0$ *such that there exists* $x \in (I \cap J) \setminus \{0\}$ *satisfying* $\|x\|_\infty \leq R$. *Then for any* $(u,v) \in I|_R \times J|_R$ *with* $u \neq v$, *we have* $\delta_{K_\mathbb{R}^\times}(u,v) \geq 1/(2R^2 \cdot |\Delta_K|^{1/(2d)})$.

Proof. We write $I = x\cdot\mathfrak{a}$ and $J = y\cdot\mathfrak{b}$ for integral ideals $\mathfrak{a}$ and $\mathfrak{b}$, note that $\mathcal{N}(x) = \mathcal{N}(\mathfrak{a})^{-1}$. The condition on $I \cap J$ implies that $\mathcal{N}(I \cap J) \leq R^d$.

By Minkowski's theorem, there exists $a \in x^{-1} \cdot (I \cap J)$ such that

$$\begin{aligned}\|a\|_\infty \leq& |\Delta_K|^{1/(2d)} \cdot \mathcal{N}\left(x^{-1} \cdot (I \cap J)\right) \\ \leq& |\Delta_K|^{1/(2d)} \cdot R \cdot \mathcal{N}(\mathfrak{a})^{1/d}.\end{aligned}$$

In particular this implies that $\mathcal{N}(a/\mathfrak{a}) \leq |\Delta_K|^{1/2} \cdot R^d$. Since $a \in x^{-1} \cdot (I \cap J)$, there exists $b \in \mathfrak{b}$ such that $x \cdot a = y \cdot b$, which implies that $x/y \in (1/a) \cdot \mathfrak{b}$.

Since $a \in (x^{-1}) \cdot x \cdot \mathfrak{a} = \mathfrak{a}$, we have that $\mathcal{O}_K \subseteq (1/a) \cdot \mathfrak{a}$, this implies that $J \subset (1/a) \cdot \mathfrak{a} \cdot J$. On the other hand we have

$$I = (x/y) \cdot y \cdot \mathfrak{a} \subseteq (1/a) \cdot \mathfrak{b} \cdot y \cdot \mathfrak{a} = (1/a) \cdot \mathfrak{a} \cdot J.$$

We proved that $I, J \subset (1/a) \cdot \mathfrak{a} \cdot J$. Let $u, v \in I|_R \times J|_R$ with $u \neq v$. The fact that $J \subset (1/a) \cdot \mathfrak{a} \cdot J$ implies that $\mathcal{N}((1/a) \cdot \mathfrak{a} \cdot J) \leq 1$, so the condition on R allows to apply Lemma 2.4:

$$\delta_{K_{\mathbb{R}}^\times}(u, v) \geq \frac{\mathcal{N}((1/a) \cdot \mathfrak{a} \cdot J)^{1/d}}{2R}.$$

The fact that $\mathcal{N}(\mathfrak{a}/a) \geq |\Delta_K|^{-1/2} \cdot R^{-d}$ then concludes the proof. □

Lemma 2.6 (see also [EHKS14b, Lemma E.5]). *Let $J \subsetneq I \in \mathrm{IdLat}_K$. Then, for any $\sigma \geq 3 \cdot d^{3/2} \cdot |\Delta_K|^{3/(2d)} \cdot \mathcal{N}(I)^{1/d}$ it holds that $\rho_\sigma(J)/\rho_\sigma(I) \leq 2/3$.*

Proof. We prove this fact for arbitrary rank d lattices $\Lambda' \subsetneq \Lambda$ and $\sigma \geq 3\sqrt{d} \cdot \lambda_d(\Lambda)$, which we will instantiate at the end of this proof with $\Lambda' = J$ and $\Lambda = I$.

Let $\Lambda' \subsetneq \Lambda$ be a sub-lattice of Λ and let $w \in \Lambda \setminus \Lambda'$. Then we have, by a technique from [HR14, Claim 2.10],

$$\begin{aligned}
\rho_\sigma(\Lambda' + w) + \rho_\sigma(\Lambda' - w) &= \sum_{x \in \Lambda'} \left(e^{-\pi \|x+w\|^2/\sigma^2} + e^{-\pi \|x-w\|^2/\sigma^2} \right) \\
&= 2e^{-\pi \|w\|^2/\sigma^2} \sum_{x \in \Lambda'} \left(e^{-\pi \|x\|^2/\sigma^2} \cosh\left(2\pi \langle x, w \rangle / \sigma^2\right) \right) \\
&\geq 2\rho_\sigma(w)\rho_\sigma(\Lambda'),
\end{aligned}$$

by applying the fact that $\cosh(\alpha)$ is bounded by 1 for real α. Hence,

$$\begin{aligned}
\rho_\sigma(\Lambda) &\geq \rho_\sigma(\Lambda') + \frac{\rho_\sigma(\Lambda' + w) + \rho_\sigma(\Lambda' - w)}{2} \\
&\geq (1 + \rho_\sigma(w))\rho_\sigma(\Lambda').
\end{aligned}$$

Therefore,

$$\frac{\rho_\sigma(\Lambda')}{\rho_\sigma(\Lambda)} \leq \frac{1}{1 + \rho_\sigma(w)}.$$

The set $\{\ell \in \Lambda \mid \|\ell\| \leq \sqrt{d}\lambda_d(\Lambda)\}$ must contain a HKZ-basis of Λ [LJS90]. So, for any $\Lambda' \subsetneq \Lambda$ there exists $w \in \Lambda \setminus \Lambda'$ with $\|w\| \leq \sqrt{d} \cdot \lambda_d(\Lambda)$. So there exists $w \in \Lambda \setminus \Lambda'$ such that $\|w\| \leq \sqrt{d} \cdot \lambda_d(\Lambda) < \sigma/3$, hence $\rho_\sigma(w) \geq \exp\left(-\pi 3^{-2}\right) \geq 0.7$, and thus $\frac{1}{1+\rho_\sigma(w)} \leq 1/1.7 \leq 2/3$. Instantiating this for $\Lambda = I$ we use the inequality

$$3 \cdot d^{3/2} \cdot |\Delta_K|^{3/(2d)} \cdot \mathcal{N}(I)^{1/d} \geq 3 \cdot \sqrt{d} \cdot \lambda_d(I),$$

by Lemma 2.3. This concludes the proof.

Lemma 2.8 (Derived of [FPSW23, Alg C.1]). *There exists a polynomial time algorithm* SampleBalanced *that, on input an ideal I with basis $\boldsymbol{B}_I$ of $\Phi(I) \subset \mathbb{R}^d$ and a balancedness parameter $\eta > 1$, outputs $x \in I \setminus \{0\}$ such that*

(i) $\|x\| \leq \frac{\eta}{\eta-1} \cdot d^{3/2} \cdot \max_{1 \leq i \leq d} \|\boldsymbol{b}_i^\star\|$, *where $(\boldsymbol{b}_i^\star)_{1 \leq i \leq d}$ is the Gram-Schmidt basis of $\boldsymbol{B}_I$,*

(ii) $\left|\sigma_i(x)/\mathcal{N}(x)^{1/d} - 1\right| \in [1 - \eta^{-1}, \eta - 1]$ *for all $i \in [\![1, d]\!]$. In particular, x is η-balanced.*

Proof. The algorithm consists in running the nearest-plane algorithm with basis $\boldsymbol{B}_I$ with target $\boldsymbol{t} = d \cdot \|\boldsymbol{B}_I^\star\| \cdot \eta/(\eta-1) \cdot \mathbf{1}$. Let $\boldsymbol{y} \in \Phi(I)$ be the output of the algorithm. We have that $\|\boldsymbol{y} - \boldsymbol{t}\|_\infty \leq (\sqrt{d}/2) \cdot \|\boldsymbol{B}_I^\star\|$. This implies that $y \neq 0$, and the bound on $\|y\|$. The balancedness of $y = \Phi^{-1}(\boldsymbol{y})$ comes from exactly the same computations as in the proof of [FPSW23, Lemma C.2] (where we take $M = 2/(\eta-1)$ in that lemma).

Lemma 3.4. *Let* Enc *be $(\nu', 1-\varepsilon')$-separative for some $\nu', \varepsilon' \in (0,1)$, and let $I, J \in \mathrm{IdLat}_K^0$ satisfy $\langle F_{R,\sigma}(I) | F_{R,\sigma}(J) \rangle \geq 1 - \varepsilon'$. Then there exists $I' \in \mathrm{IdLat}_K$ with $\delta_{\mathrm{ideal}}(I, I') \leq \nu'$ such that $I' \cap J \neq \{0\}$.*

Proof. Write $|I\rangle = F_{R,\sigma}(I)$ and $|J\rangle = F_{R,\sigma}(J)$. Then, by the Cauchy-Schwarz inequality,

$$\begin{aligned}
|\langle I|J\rangle| &\leq \sum_{x,y \in I|_R \times J|_R} \sqrt{p_{R,\sigma}(I,x) p_{R,\sigma}(J,y)} |\langle \mathrm{Enc}(x) | \mathrm{Enc}(y)\rangle| \\
&\leq \max_{x,y \in I|_R \times J|_R} (|\langle \mathrm{Enc}(x) | \mathrm{Enc}(y)\rangle|) \cdot \sum_{x,y \in I|_R \times J|_R} \sqrt{p_{R,\sigma}(I,x) p_{R,\sigma}(J,y)} \\
&\leq \max_{x,y \in I|_R \times J|_R} |\langle \mathrm{Enc}(x) | \mathrm{Enc}(y)\rangle|.
\end{aligned}$$

Thus, there must exist $x \in I|_R$ and $y \in J|_R$ with $|\langle \mathrm{Enc}(x) | \mathrm{Enc}(y)\rangle| \geq |\langle I|J\rangle| \geq 1 - \varepsilon'$. Hence, by separativity, we must have $\delta_{K_{\mathbb{R}}^\times}(x, y) \leq \nu'$, i.e., $y/x = \mathrm{ExpEx}(\boldsymbol{\theta})$ with $\|\boldsymbol{\theta}\| \leq \nu'$. Putting $I' := (y/x) \cdot I$, we have $y \in (\mathrm{ExpEx}(\boldsymbol{\theta}) \cdot I) = (y/x) \cdot I = I'$ but also $y \in J$. We can therefore conclude that $y \in I' \cap J$ with $\delta_{\mathrm{ideal}}(I, I') \leq \nu'$. □

Lemma 3.5. *Let $\sigma \geq 3 \cdot d^{3/2} \cdot |\Delta_K|^{3/(2d)}$, $R \geq \sqrt{d} \cdot \sigma$ and $\nu \leq 1/(2R)$. Let $I, J \in \mathrm{IdLat}_K^0$ satisfying $(I \cap J)|_R \neq \{0\}$. Then either $I = J$ or*

$$\langle I|J\rangle < \frac{4}{5}.$$

Proof. If $I = J$, there is nothing to prove. Hence, let us assume that $I \neq J$. Without loss of generality, we assume that $\rho_\sigma(J|_R) \leq \rho_\sigma(I|_R)$. Lemmas 2.5, 2.6,

A.1 and A.2 and the condition on ν implies that

$$\begin{aligned}
\langle I|J\rangle &= \sum_{x\in(I\cap J)|_R\setminus\{0\}} \sqrt{p_R(I,x)\cdot p_R(J,x)} = \frac{\rho_\sigma\left((I\cap J)\setminus\{0\}\,|_R\right)}{\sqrt{\rho_\sigma(I\setminus\{0\}\,|_R)\rho_\sigma(J\setminus\{0\}\,|_R)}} \\
&\leq \frac{\rho_\sigma\left((I\cap J)\setminus\{0\}\,|_R\right)}{\rho_\sigma(I\setminus\{0\}\,|_R)} = \frac{\rho_\sigma\left((I\cap J)|_R\right)-1}{\rho_\sigma(I|_R)-1} \leq \frac{\rho_\sigma\left(I\cap J\right)-1}{(1-e^{-d^2})\cdot\rho_\sigma(I)-1} \\
&= \frac{\rho_\sigma(I\cap J)}{\rho_\sigma(I)}\cdot\frac{1-\rho_\sigma(I\cap J)^{-1}}{1-e^{-d^2}-\rho_\sigma(I)^{-1}} \leq \frac{2}{3}\cdot\frac{1}{1-e^{-d^2}-\rho_\sigma(I)^{-1}} \\
&\leq \frac{2}{3}\cdot\frac{1}{1-e^{-d^2}-(1-e^{-d})^{-1}\cdot\sqrt{|\Delta_K|}/\sigma^d} \\
&\leq \frac{2}{3}\cdot\frac{1}{1-e^{-d^2}-(1-e^{-d})^{-1}\cdot(3\sqrt{2}\cdot d^{1.5})^{-d}}.
\end{aligned}$$

An analysis of this expression shows that for $d\geq 2$, it is less than $4/5$, hence the result. □

Lemma 3.6. *Let $\sigma\geq 3\cdot d^{3/2}\cdot|\Delta_K|^{3/(2d)}$, $\nu\leq 1/(2R)$ and* Enc *an injective map which is ν-totally separative over $(K_{\mathbb{R}}^\times,\delta_{K_{\mathbb{R}}^\times})$. Furthermore, assume that σ,ν,R and* Enc *are such that $F_{R,\sigma}$ is (A,α)-almost Lipschitz for some $A\in\mathbb{R}$ with $\alpha\leq 1/30$, and that* Enc *is $(\nu',1-\varepsilon')$-separative for some $\varepsilon'\in(0,1/30]$ and $\nu'\leq 1/(30A)$. Then the function $F_{R,\sigma}$ is $(\nu',1-\varepsilon')$-separative.*

Proof. Let $I,J\in\mathrm{IdLat}_K^0$ such that $\langle F_{R,\sigma}(I)|F_{R,\sigma}(J)\rangle\geq 1-\varepsilon'$. By Lemma 3.4 we can deduce that there exists $I'\in\mathrm{IdLat}_K$ with $\delta_{\mathrm{ideal}}(I,I')\leq\nu'$ with $I'\cap J\neq\{0\}$.

Our next aim is to show that $I'=J$, and hence $\delta_{\mathrm{ideal}}(I,J)=\delta_{\mathrm{ideal}}(I,I')\leq\nu'$, which then finishes the proof. Writing $|J\rangle := F_{R,\sigma}(J)$ (and similarly for I',I), we have

$$\begin{aligned}
\langle I'|J\rangle &= \langle I|J\rangle - (\langle I|-\langle I'|)|J\rangle \\
&\geq 1-\varepsilon'-(A\cdot\nu'+\alpha) \qquad \text{by } (A,\alpha)\text{-almost Lipschitz continuity} \\
&\geq 9/10.
\end{aligned}$$

Now, by Lemma 3.5, this implies that $I'=J$, which finishes the proof. □

Lemma 3.3. *Let σ, ν and* Enc *satisfy the conditions of Lemma 3.2. Let $R\geq 2\sigma$. Then $F_{R,\sigma}$ is $[(5d+2a),4e^{-(R/\sigma)^2/2}]$-almost Lipschitz continuous.*

Proof. Using that $\||\Psi_R\rangle - |\Psi_\infty\rangle\|^2 = 2 - 2\,\mathrm{Re}(\langle\Psi_R|\Psi_\infty\rangle)$ we concentrate on this latter inner product. We have

$$\begin{aligned}
\langle\Psi_R|\Psi_\infty\rangle &= \sum_{x\in I\setminus\{0\}|_R, y\in I\setminus\{0\}} \sqrt{p_{R,\sigma}(I,x)}\sqrt{p_{\infty,\sigma}(I,y)}\,\langle \mathrm{Enc}(x)|\mathrm{Enc}(y)\rangle \\
&= \sum_{x\in I\setminus\{0\}|_R} \sqrt{p_{R,\sigma}(I,x)}\sqrt{p_{\infty,\sigma}(I,x)} = \sum_{x\in I\setminus\{0\}|_R} \frac{\rho_\sigma(x)}{\sqrt{\rho_\sigma(I\setminus\{0\}\,|_R)\rho_\sigma(I\setminus\{0\})}} \\
&= \sqrt{\frac{\rho_\sigma(I\setminus\{0\}\,|_R)}{\rho_\sigma(I\setminus\{0\})}} = \sqrt{\frac{\rho_\sigma(I|_R)-1}{\rho_\sigma(I)-1}} = \sqrt{\frac{-e^{-R^2/\sigma^2}+1-\rho_\sigma(I)^{-1}}{1-\rho_\sigma(I)^{-1}}} \\
&\geq \sqrt{1-2e^{-(R/\sigma)^2}} \geq 1-2e^{-(R/\sigma)^2}.
\end{aligned}$$

where the second equality follows from Lemma 2.4, and the last equality from Banaszczyk's tail bound (Lemma A.2). In the second last inequality we use that $\rho_\sigma(I) \geq \frac{1}{2}\sigma^d/\sqrt{|\Delta_K|} \geq 2$ by smoothing arguments (Lemma A.1), and in the last inequality we use $\sqrt{1-2x} \geq 1-2x$ for $x \in [0,1/2]$. Clearly $e^{-(R/\sigma)^2} \leq 1/2$ since $R \geq 2\sigma$. Hence

$$\begin{aligned}
&\||F_{R,\sigma}\rangle(uI) - |F_{R,\sigma}\rangle(I)\| \\
&\leq \||F_{R,\sigma}\rangle(uI) - |F_{\infty,\sigma}\rangle(uI)\| + \||F_{\infty,\sigma}\rangle(uI) - |F_{\infty,\sigma}\rangle(I)\| + \||F_{\infty,\sigma}\rangle(I) - |F_{R,\sigma}\rangle(I)\| \\
&\leq \sqrt{4e^{(R/\sigma)^2}} + (5d+2a)\|\,\mathrm{LogEx}(u)\| + \sqrt{4e^{(R/\sigma)^2}}.
\end{aligned}$$

Hence $F_{R,\sigma}$ is $[(5d+2a), 4e^{-(R/\sigma)^2/2}]$-almost Lipschitz continuous. □

Corollary 5.1. *The complexity of the quantum procedure of Theorem 2.2 running with this paper's quantum implementation of the oracle G is:*
Quantum memory:

$$\begin{aligned}
&\mathrm{LLLMem}\left(d, (d+s)^{5+o(1)}\cdot\left(\ln\left(|\Delta_K|^{1/d}\right)+d\right)^{2+o(1)}\right) \\
+&O\left(d^{\omega+2+o(1)}\cdot(d+s)^{5+o(1)}\cdot\left(\ln\left(|\Delta_K|^{1/d}\right)+d\right)^{2+o(1)}\right).
\end{aligned}$$

Gate count:

$$\begin{aligned}
&\mathrm{LLLGates}\left(d, (d+s)^{5+o(1)}\cdot\left(\ln\left(|\Delta_K|^{1/d}\right)+d\right)^{2+o(1)}\right) \\
&\cdot O\left((d+s)^{1+o(1)}\left(\ln\left(|\Delta_K|^{1/d}\right)+d\right)\right) \\
+&O\left(d^{\omega+2+o(1)}\cdot(d+s)^{9+o(1)}\cdot\left(\ln\left(|\Delta_K|^{1/d}\right)+d\right)^{4+o(1)}\right).
\end{aligned}$$

Proof. Note that the quantum complexity, both time and space, of the whole procedure to compute a basis of Λ_S is dominated by the computation of G. By Theorem 2.2 we need to compute the oracle $k = O((d+s)^{1+o(1)}\ln(A)) =$

$O((d+s)^{1+o(1)}(\ln\left(|\Delta_K|^{1/d}\right)+d))$ times. By re-using memory, the quantum memory count is the same as in Theorem 5.1, whereas the quantum gate count is multiplied by $k = O((d+s)^{1+o(1)}(\ln\left(|\Delta_K|^{1/d}\right)+d))$, which yields the result.

C Lipschitz bound on the difference of Gaussian sums

Lemma C.1. *Let Λ a full rank lattice in $\mathbb{R}^n$, and let $1 > \varepsilon > 0$ and $\sigma > 2\eta_\varepsilon(\Lambda)$. Let X be a random variable sampled from the discrete Gaussian over Λ with parameter σ. Let $i, j \in \{1, \ldots, n\}$.*

Then

- $\left|\mathbb{E}(X_i^2) - \frac{\sigma^2}{2\pi}\right| < \frac{\sigma^2\varepsilon}{1-\varepsilon} \cdot \frac{1}{4\pi^2}$.
- $\left|\mathbb{E}(X_i^2 X_j^2) - \frac{3\sigma^4}{4\pi^2}\right| < \frac{\sigma^4\varepsilon}{1-\varepsilon} \cdot \left(\frac{3}{4\pi^3} + \frac{1}{16\pi^4}\right)$.

Proof. This is a generalization of the proof of [MR07, Lemma 4.2]. □

Corollary C.1. *Let $I \in \mathrm{IdLat}_K^0$ and $\sigma \geq |\Delta_K|^{1/d}$. Let $\widetilde{X}$ be a random variable sampled from the discrete Gaussian distribution over $I \setminus \{0\}$ with parameter σ. Then for any $i, j \in \{1, \ldots, n\}$,*

- $\mathbb{E}\left(\widetilde{X}_i^2\right) \leq 0.4 \cdot \sigma^2$.
- $\mathbb{E}\left(\left\|\widetilde{X}\right\|^2\right) \leq 0.4 \cdot d \cdot \sigma^2$
- $\mathbb{E}\left(\widetilde{X}_i^2 \widetilde{X}_j^2\right) \leq 0.2 \cdot \sigma^4$.
- $\mathbb{E}\left(\left\|\widetilde{X}\right\|^4\right) \leq 0.2 \cdot d^2 \cdot \sigma^4$.

Proof. One can prove that if we set $\delta = \rho_\sigma(I \setminus \{0\})^{-1}$, then for any function $f : K_\mathbb{R} \to \mathbb{R}_{\geq 0}$ satisfying $f(0) = 0$ it holds that

$$\mathbb{E}(f(\widetilde{X})) = (1+\delta) \cdot \mathbb{E}(f(X)),$$

where X is sampled from the Gaussian distribution over I with parameter σ. By Lemma 2.2, it holds that $\sigma \geq \eta_{2^{-d}}(I)$, which implies that $\rho_\sigma(I) \geq (1-2^{-d}) \cdot \sigma^d/|\Delta_K|^{1/2} \geq \sqrt{|\Delta_K|} \geq 2^{d/2}$ where the last bound follows from Minkowski's theorem. In particular, it holds that $\delta \leq (2^{d/2}-1)^{-1}$. The result follows from applying Lemma C.1 with $\varepsilon = 2^{-d}$ and multiplying by $1+\delta$. □

In this subsection, for any $I \in \mathrm{IdLat}_K^0$, $x \in I$ and $u \in K_\mathbb{R}^0$ we write $h_\sigma(I, x) = \rho_\sigma(x)/\rho_\sigma(I)$ and $\boldsymbol{v}_{\sigma,I}(\boldsymbol{\theta}) = (\sqrt{h_\sigma(e^{\boldsymbol{\theta}} I, e^{\boldsymbol{\theta}} x)})_{x \in I}$.

Lemma C.2. *Let $\sigma \geq |\Delta_K|^{1/d}$. Then the function $\boldsymbol{\theta} \mapsto \boldsymbol{v}_{\sigma,I}(\boldsymbol{\theta})$ is $(\pi \cdot d)$-Lipschitz.*

Proof. Note that the derivative with respect to the imaginary parts of $\boldsymbol{\theta}$ is equal to zero, due to the fact that the Gaussian only depends on the norm. Hence, we only consider $\boldsymbol{\theta} \in \mathbb{R}^{d_{\mathbb{R}}+d_{\mathbb{C}}}$. Then, by applying standard derivative rules, we obtain

$$\partial_{\theta_i} \sqrt{h_\sigma(e^{\boldsymbol{\theta}} I, e^{\boldsymbol{\theta}} x)} = \frac{-\pi}{\sigma^2} \cdot \sqrt{h_\sigma(e^{\boldsymbol{\theta}} I, e^{\boldsymbol{\theta}} x)} \cdot \left(\left|\sigma_i(e^{\boldsymbol{\theta}} x)\right|^2 - \mathbb{E}(|\sigma_i(Y)|^2) \right)$$

where Y is a discrete Gaussian sampled over $e^{\boldsymbol{\theta}} I$ with parameter σ. For any i, the sequence $(\boldsymbol{\theta} \mapsto \partial_{\theta_i} \sqrt{h_\sigma(e^{\boldsymbol{\theta}} \cdot I, e^{\boldsymbol{\theta}} \cdot x)})_{x \in I|R}$ converges uniformly to $(\boldsymbol{\theta} \mapsto \partial_{\theta_i} \sqrt{h_\sigma(e^{\boldsymbol{\theta}} \cdot I, e^{\boldsymbol{\theta}} \cdot x)})_{x \in I}$ when $R \to \infty$. This implies that

$$\partial_{\theta_i} \left[\boldsymbol{\theta} \mapsto (\sqrt{h_\sigma(e^{\boldsymbol{\theta}} \cdot I, e^{\boldsymbol{\theta}} \cdot x)})_{x \in I} \right] = \boldsymbol{\theta} \mapsto (\partial_{\theta_i} \sqrt{h_\sigma(e^{\boldsymbol{\theta}} \cdot I, e^{\boldsymbol{\theta}} \cdot x)})_{x \in I}.$$

In particular, if we denote $\mathrm{D}\boldsymbol{v}_{\sigma,I}|_{\boldsymbol{\theta}}$ the differential operator of $\boldsymbol{v}_{\sigma,I}$ at $\boldsymbol{\theta}$, we have for all $\alpha \in \mathbb{R}^{d_{\mathbb{R}}+d_{\mathbb{C}}}$,

$$\mathrm{D}\boldsymbol{v}_{\sigma,I}|_{\boldsymbol{\theta}}(\alpha) = \frac{-\pi}{\sigma^2} \cdot \sum_{i=1}^{d} \alpha_i \cdot \left(\sqrt{h_\sigma(e^{\boldsymbol{\theta}} I, e^{\boldsymbol{\theta}} x)} \cdot \left(\left|\sigma_i(e^{\boldsymbol{\theta}} x)\right|^2 - \mathbb{E}(|\sigma_i(Y)|^2) \right) \right)_{x \in I} \tag{9}$$

We then have

$$\begin{aligned}
\left\|\mathrm{D}\boldsymbol{v}_{\sigma,I}|_{\boldsymbol{\theta}}(\alpha)\right\|^2 &= \frac{\pi^2}{\sigma^4} \sum_{x \in I} \sum_{i=1}^{d} \alpha_i^2 \cdot h_\sigma(e^{\boldsymbol{\theta}} I, e^{\boldsymbol{\theta}} x) \cdot \left(\left|\sigma_i(e^{\boldsymbol{\theta}} x)\right|^2 - \mathbb{E}(|\sigma_i(Y)|^2) \right)^2 \\
&\leq \frac{\|\alpha\|^2 \cdot \pi^2}{\sigma^4} \sum_{x \in I} h_\sigma(e^{\boldsymbol{\theta}} I, e^{\boldsymbol{\theta}} x) \cdot \sum_{i=1}^{d} \left(\left|\sigma_i(e^{\boldsymbol{\theta}} x)\right|^2 - \mathbb{E}(|\sigma_i(Y)|^2) \right)^2 \\
&\leq \frac{\|\alpha\|^2 \cdot \pi^2}{\sigma^4} \sum_{x \in I} h_\sigma(e^{\boldsymbol{\theta}} I, e^{\boldsymbol{\theta}} x) \cdot \sum_{i=1}^{d} \left(\left|\sigma_i(e^{\boldsymbol{\theta}} x)\right|^4 + \mathbb{E}(|\sigma_i(Y)|^2)^2 \right) \\
&\leq \frac{\|\alpha\|^2 \cdot \pi^2}{\sigma^4} \sum_{x \in I} h_\sigma(e^{\boldsymbol{\theta}} I, e^{\boldsymbol{\theta}} x) \cdot \sum_{i=1}^{d} \left(\left|\sigma_i(e^{\boldsymbol{\theta}} x)\right|^4 + \mathbb{E}(|\sigma_i(Y)|^2)^2 \right) \\
&\leq \frac{\|\alpha\|^2 \cdot \pi^2}{\sigma^4} \cdot \left(\mathbb{E}\left(\|Y\|^2 \right)^2 + \sum_{x \in I} h_\sigma(e^{\boldsymbol{\theta}} I, e^{\boldsymbol{\theta}} x) \cdot \left\|e^{\boldsymbol{\theta}} x\right\|^4 \right) \\
&\leq \frac{\|\alpha\|^2 \cdot \pi^2}{\sigma^4} \cdot \left(\mathbb{E}\left(\|Y\|^2 \right)^2 + \mathbb{E}\left(\|Y\|^4 \right) \right) \leq d^2 \cdot \pi^2 \cdot \|\alpha\|^2,
\end{aligned}$$

where the last inequality comes from Corollary C.1. This concludes the proof. □

Lemma C.3. *Let $\sigma \geq 2|\Delta_K|^{1/d}$, let $R \geq \max(\sqrt{d}\sigma, 4\sigma \ln \sigma)$ and let $I \in \mathrm{IdLat}_K^0$. Write*

$$g_\sigma(I, x) = \rho_\sigma(x)/\rho_\sigma(I \setminus \{0\})$$

Define $L_x := \left(\sum_{i=1}^{d_{\mathbb{R}}+d_{\mathbb{C}}} (\partial_{\theta_i} \sqrt{g_\sigma(e^{\boldsymbol{\theta}} I, e^{\boldsymbol{\theta}} x)}|_{\theta_i=0})^2\right)^{1/2}$*, which are Lipschitz constants of the components of the function* $\boldsymbol{\theta} \mapsto \boldsymbol{v}_{\sigma,I}(\boldsymbol{\theta}) = (\sqrt{g_\sigma(e^{\boldsymbol{\theta}} I, e^{\boldsymbol{\theta}} x)})_{x \in I}$*. We have*

$$\sum_{x \in I \setminus B(0,R)} L_x \leq (1 + \pi\sqrt{d}) \cdot 4 \cdot 2^d \sigma^{d/2} |\Delta_K|^{-1/4} e^{-R^2/(2\sigma)^2}$$

and

$$\sum_{x \in I \setminus B(0,R)} \sqrt{g_\sigma(I, x)} \leq 4 \cdot 2^d \sigma^{d/2} |\Delta_K|^{-1/4} e^{-R^2/(2\sigma)^2}.$$

Additionally, if $R \geq 2\sigma\sqrt{d \ln(32\sigma)}$*, both sums are bounded by* 1*.*

Proof. By the chain rule, we have

$$\partial_{\theta_i} \sqrt{g_\sigma(e^{\boldsymbol{\theta}} I, e^{\boldsymbol{\theta}} x)} = \frac{1}{2} \cdot g_\sigma(e^{\boldsymbol{\theta}} I, e^{\boldsymbol{\theta}} x)^{-1/2} \cdot \partial_{\theta_i} g_\sigma(e^{\boldsymbol{\theta}} I, e^{\boldsymbol{\theta}}). \tag{10}$$

By the quotient rule, we have

$$\begin{aligned}
\partial_{\theta_i} g_\sigma(e^{\boldsymbol{\theta}} I, e^{\boldsymbol{\theta}}) &= \frac{\rho_\sigma(e^{\boldsymbol{\theta}} I \setminus \{0\}) \partial_{\theta_i} \rho_\sigma(e^{\boldsymbol{\theta}} x) - \rho_\sigma(e^{\boldsymbol{\theta}} x) \partial_{\theta_i} \rho_\sigma(e^{\boldsymbol{\theta}} I \setminus \{0\})}{\rho_\sigma(e^{\boldsymbol{\theta}} I \setminus \{0\})^2} \\
&= \frac{-2\pi \cdot \rho_\sigma(e^{\boldsymbol{\theta}} x)}{\sigma^2} \cdot \frac{e^{2\theta_i} |\sigma_i(x)|^2 \rho_\sigma(e^{\boldsymbol{\theta}} I \setminus \{0\}) - \sum_{x \in I \setminus \{0\}} e^{2\theta_i} |\sigma_i(x)|^2 \rho_\sigma(e^{\boldsymbol{\theta}} x)}{\rho_\sigma(e^{\boldsymbol{\theta}} I \setminus \{0\})^2} \\
&= \frac{-2\pi \cdot \rho_\sigma(e^{\boldsymbol{\theta}} x)}{\sigma^2 \cdot \rho_\sigma(e^{\boldsymbol{\theta}} I \setminus \{0\})} \cdot \left(e^{2\theta_i} |\sigma_i(x)|^2 - \mathbb{E}[\tilde{X}_i^2]\right) \\
&= \frac{-2\pi}{\sigma^2} \cdot g_\sigma(e^{\boldsymbol{\theta}} I, e^{\boldsymbol{\theta}} x) \cdot \left(e^{2\theta_i} |\sigma_i(x)|^2 - \mathbb{E}[\tilde{X}_i^2]\right),
\end{aligned} \tag{11}$$

where $\tilde{X}_i$ is the σ_i-th component of the random variable over the replete ideal $e^{\boldsymbol{\theta}} I$ defined by the probability distribution $e^{\boldsymbol{\theta}} x \mapsto g_\sigma(e^{\boldsymbol{\theta}} I, e^{\boldsymbol{\theta}} x)$.

Combining Equations (10) and (11), we obtain

$$\partial_{\theta_i} \sqrt{g_\sigma(e^{\boldsymbol{\theta}} I, e^{\boldsymbol{\theta}} x)} = \frac{-\pi}{\sigma^2} \sqrt{g_\sigma(e^{\boldsymbol{\theta}} I, e^{\boldsymbol{\theta}} x)} \left(e^{2\theta_i} |\sigma_i(x)|^2 - \mathbb{E}[\tilde{X}_i^2]\right).$$

Hence, evaluating the derivative at $\boldsymbol{\theta} = 0$ and taking the square of the Euclidean norm,

$$\begin{aligned}
L_x^2 &= \frac{\pi^2}{\sigma^4} \cdot g_\sigma(I, x) \sum_{i=1}^{d_{\mathbb{R}}+d_{\mathbb{C}}} (|\sigma_i(x)|^2 - \mathbb{E}[\tilde{X}_i^2])^2 \leq \frac{\pi^2}{\sigma^4} \cdot g_\sigma(I, x) \left(\|x\|^4 + \sum_{i=1}^{d_{\mathbb{R}}+d_{\mathbb{C}}} \mathbb{E}[\tilde{X}_i^2]^2\right) \\
&\leq \frac{\pi^2}{\sigma^4} \cdot g_\sigma(I, x) \left(\|x\|^4 + d\sigma^4\right)
\end{aligned}$$

where the last inequality follows from Corollary C.1. Therefore, by using $\sqrt{a+b} \leq \sqrt{a} + \sqrt{b}$ for $a, b \in \mathbb{R}_{>0}$,

$$\sum_{x \in I \setminus B(0,R)} L_x \leq \frac{\pi}{\sigma^2} \sum_{x \in I \setminus B(0,R)} \sqrt{g_\sigma(I, x)} \|x\|^2 + \pi\sqrt{d} \sum_{x \in I \setminus B(0,R)} \sqrt{g_\sigma(I, x)}. \tag{12}$$

We concentrate on the left-hand summand, for which we have

$$\begin{aligned}\sum_{x\in I\setminus B(0,R)} \sqrt{g_\sigma(I,x)}\|x\|^2 &\le \frac{1}{\rho_\sigma(I\setminus\{0\})^{1/2}} \sum_{x\in I\setminus B(0,R)} e^{-\frac{\pi}{2\sigma^2}\|x\|^2+2\ln\|x\|}\\ &\le \frac{1}{\rho_\sigma(I\setminus\{0\})^{1/2}} \sum_{x\in I\setminus B(0,R)} e^{-\frac{\pi}{4\sigma^2}\|x\|^2}\\ &\le \frac{\rho_{2\sigma}(I\setminus B(0,R))}{\rho_\sigma(I\setminus\{0\})^{1/2}} \le e^{-R^2/(2\sigma)^2}\cdot\frac{\rho_{2\sigma}(I)}{\rho_\sigma(I\setminus\{0\})^{1/2}}\\ &\le e^{-R^2/(2\sigma)^2}\frac{2\cdot(2\sigma)^d|\Delta_K|^{-1/2}}{\sqrt{0.5\cdot\sigma^d\cdot|\Delta_K|^{-1/2}-1}}\\ &\le 4\cdot 2^d\sigma^{d/2}|\Delta_K|^{-1/4}e^{-R^2/(2\sigma)^2}.\end{aligned}$$

where the second inequality holds since $\frac{1}{2}\le 1-\frac{2\sigma^2}{\pi}\frac{\ln(\|x\|^2)}{\|x\|^2}$, whenever $\|x\|\ge R\ge 4\sigma\ln\sigma$; the fourth inequality by the fact that $R\ge\sqrt{d}\sigma$ and Banaszczyk's bound (Lemma A.2); the fifth inequality by smoothing arguments (Lemma A.1) (which gives an error of e^{-d} for $\sigma\ge|\Delta_K|^{1/d}$); the sixth inequality by the fact that $\sigma^d\ge 4|\Delta_K|^{1/2}$.

The right-hand summand of Equation (12) can be similarly bounded by the fact that

$$\begin{aligned}\sum_{x\in I\setminus B(0,R)} \sqrt{g_\sigma(I,x)} &\le \frac{\rho_{2\sigma}(I\setminus B(0,R))}{\rho_\sigma(I\setminus\{0\})^{1/2}} \le e^{-R^2/(2\sigma)^2}\cdot\frac{\rho_{2\sigma}(I)}{\rho_\sigma(I\setminus\{0\})^{1/2}}\\ &\le 4\cdot 2^d\sigma^{d/2}|\Delta_K|^{-1/4}e^{-R^2/(2\sigma)^2}.\end{aligned}$$

By combining these two bounds, we obtain the bound on the sum over L_x, whereas just the last computation yields the bound on the sum over $\sqrt{g_\sigma(I,x)}$.

For the last statement, we only have to prove the bound of 1 of the first sum, as the last sum easily follows. If $R\ge 2\sigma\sqrt{d\ln(32\sigma)}$, we have that $e^{-R^2/(2\sigma)^2}\le(32\sigma)^{-d}$ and hence

$$(1+\pi\sqrt{d})\cdot 4\cdot 2^d\sigma^{d/2}|\Delta_K|^{-1/4}\cdot e^{-R^2/(2\sigma)^2}\le(1+\pi\sqrt{d})\cdot 4\cdot 16^{-d}\sigma^{-d/2}<1.$$

D Compact representation of high powers of an ideal

Lemma D.1. *Algorithm D.1 is correct and runs in polynomial time in* $\ln(k)$, d *and in the size of its input. Additionally,* $\mathfrak{a}$ *is an integral ideal satisfying* $\mathcal{N}(\mathfrak{a})\le C_\eta^d\cdot|\Delta_K|^{1/2}$ *for* $C_\eta=2^d\cdot d^3\cdot(\eta-1)^{-1}$, *and the* α_i *are* η*-balanced elements of* K *with size polynomial in* $\ln(C_\eta),\ln(|\Delta_K|)$ *and size*(I). *Furthermore, we always have* $t\le\lceil\log_2(k)\rceil$ *for* $k>0$.

Proof. We prove the correctness by induction on k, where the ground case $k=0$ is trivial. We assume $k>0$ and k is even. Then, by induction, $(\mathfrak{a}_0,\beta_0,\ldots,\beta_{t-1})=$

Algorithm D.1 CompRep

Input: An ideal lattice I represented as its basis over $\boldsymbol{B}_{\mathcal{O}_K}$, a power $k \in \mathbb{N}$, and a real $\eta \in (1,2)$

Output: $(\mathfrak{a}, \alpha_0, \ldots, \alpha_t)$ such that $\mathfrak{a} \cdot \prod_{j=0}^{t} \alpha_j^{2^j} = I^k$.

1: If $k = 0$, return $\mathfrak{a} = (\mathcal{O}_K)$.
2: Compute $(\mathfrak{a}_0, \beta_0, \ldots, \beta_{t-1}) = \mathrm{CompRep}(I, \lfloor k/2 \rfloor)$.
3: Put $\mathfrak{a}' = \mathfrak{a}_0^2$ if k is even, $\mathfrak{a}' = I \cdot \mathfrak{a}_0^2$ else.
4: Compute a LLL-reduced basis $\boldsymbol{B}$ of $\Phi(\mathfrak{a}'^{-1})$.
5: Compute $\alpha' = \mathrm{SampleBalanced}(\boldsymbol{B}, \eta)$, and put $\mathfrak{a} = \alpha' \cdot \mathfrak{a}'$.
6: Put $\alpha_j = \beta_{j-1}$ for $j \in \{1, \ldots, t\}$ and $\alpha_0 = 1/\alpha'$.
7: Output $(\mathfrak{a}, \alpha_0, \ldots, \alpha_t)$

$\mathrm{CompRep}(I, k/2)$ satisfies $(I)^{k/2} = \mathfrak{a}_0 \cdot \prod_{j=0}^{t-1} \beta_j^{2^j}$. By definition, we have $\mathfrak{a}_0^2 = \mathfrak{a} \cdot (\alpha')^{-1}$, hence

$$(I)^k = \left(\mathfrak{a}_0 \cdot \prod_{j=0}^{t-1} \beta_j^{2^j} \right)^2 = \mathfrak{a}_0^2 \cdot \prod_{j=0}^{t-1} \beta_j^{2^{j+1}} = \mathfrak{a} \cdot (\alpha')^{-1} \cdot \prod_{j=1}^{t} \alpha_j^{2^j} = \mathfrak{a} \cdot \prod_{j=0}^{t} \alpha_j^{2^j}.$$

For k is odd we have, by induction $(I)^{(k-1)/2} = \mathfrak{a}_0 \cdot \prod_{j=0}^{t-1} \beta_j^{2^j}$. Since, by definition we have $\mathfrak{a} \cdot (\alpha')^{-1} = I \cdot \mathfrak{a}_0^2$, we obtain

$$(I)^k = I \cdot \left(\mathfrak{a}_0 \cdot \prod_{j=0}^{t-1} \beta_j^{2^j} \right)^2 = I \cdot \mathfrak{a}_0^2 \prod_{j=1}^{t} \alpha_j^{2^j} = \mathfrak{a} \cdot \prod_{j=0}^{t} \alpha_j^{2^j},$$

which finishes the proof.

We will now show the bounds on $\mathfrak{a}$ and α_i of the output. In all cases, $\mathfrak{a}$ is either $\mathcal{O}_K$ or $\mathfrak{a}' \cdot \alpha'$ where α' is output of SampleBalancedwith a LLL-short basis of $\mathfrak{a}'^{-1}$. This implies that $\mathfrak{a} \subseteq \mathcal{O}_K$, that $\|\alpha'\| \leq C_\eta \cdot \det(\mathfrak{a}')^{-1/d}$ and that α' (and hence also $\alpha_0 = 1/\alpha'$) is η-balanced and that its size is polynomial in $\ln(C_\eta), \ln(|\Delta_K|)$ and $\mathrm{size}(I)$. By the arithmetic-geometric mean inequality we obtain that

$$\mathcal{N}(\alpha') \leq (\|\alpha'\|/\sqrt{d})^d \leq C_\eta^d \cdot \det(\mathfrak{a}')^{-1} = C_\eta^d \cdot \mathcal{N}(\mathfrak{a}')^{-1} \cdot |\Delta_K|^{1/2},$$

hence $\mathcal{N}(\mathfrak{a}) = \mathcal{N}(\mathfrak{a}') \cdot \mathcal{N}(\alpha') \leq C_\eta^d \cdot |\Delta_K|^{1/2}$.

For the bound on t, use induction: for $k = 1$ note that only β_0 is defined, hence $t = 0$. For $k > 1$, a new α_i added to a list of, by induction, of $\leq \lceil \log_2(k/2) \rceil \leq \lceil \log_2(k) \rceil - 1$ elements, which proves the claim.

The fact that k is at least divided by 2 at each recursive call to CompRepimplies that $\lceil \log_2(k)$ recursive call are made on input k. The LLL algorithm and ideal multiplications run in polynomial time on the size of their input, which implies the claimed running time. □

Note that when computing a compact representation for I^{2^k}, a compact representation for I^{2^j} is computed along the way for all $j = 1, \ldots, k-1$.

Lemma D.2. *Let $\mathfrak{a}$ output from Algorithm D.1 on input I, k, η. Then we have that*

$$\delta_{\mathrm{ideal}}(\mathfrak{a}/\mathcal{N}(\mathfrak{a})^{1/d}, I^k/\mathcal{N}(I^k)^{1/d}) \leq 2^{k'+1} \cdot \sqrt{d} \cdot (\eta - 1)$$

for $k' = \lceil \log_2(k) \rceil$.

Proof. One can check that for any $I, J \in \mathrm{IdLat}_K^0$, one has $\delta_{\mathrm{ideal}}(I^2, J^2) \leq 2\delta_{\mathrm{ideal}}(I, J)$. We now show the result for $k = 1$. We have that $\mathfrak{a}' = I$, and $\mathfrak{a} = \alpha' \cdot I$ for α' satisfying, by Lemma 2.8, that

$$\left| \sigma_i(\alpha')/\mathcal{N}(\alpha')^{1/d} - 1 \right| \in [1 - \eta^{-1}, \eta - 1].$$

We denote $\tilde{\mathfrak{a}} = \mathfrak{a}/\mathcal{N}(\mathfrak{a})^{1/d}$, $\tilde{I} = I/\mathcal{N}(I)^{1/d}$ and $\tilde{\alpha'} = \alpha'/\mathcal{N}(\alpha')^{1/d}$. We have that $\tilde{I} = (\tilde{\alpha}) \cdot \tilde{\mathfrak{a}}$, and hence $\delta_{\mathrm{ideal}}(\tilde{I}, \tilde{\mathfrak{a}}) \leq \left\| \mathrm{LogEx}(\tilde{\alpha'}) \right\|$. Now, for every $i = 1, \ldots, d$, we have that $\ln\left(\left|\sigma_i(\tilde{\alpha'})\right|\right) \leq \left|\sigma_i(\tilde{\alpha'})\right| - 1 \leq \eta - 1$. It is also true for any $i = d_{\mathbb{R}} + 1, \ldots, d_{\mathbb{R}} + 2d_{\mathbb{C}}$ that $\arg(\sigma_i(\tilde{\alpha'})) \leq \tan^{-1}(|\sigma_i(\alpha') - 1|) \leq \eta - 1$. Finally, it holds that

$$\|\mathrm{LogEx}(\alpha')\| \leq \sqrt{d} \cdot (\eta - 1).$$

For $k \geq 2$, by induction we have that the distance between the normalized target ideal and $\mathfrak{a}'$ is multiplied by two at Step 3 and then by the same argument as for $k = 1$, the distance is increased by at most $\sqrt{d} \cdot (\eta - 1)$. A bound is then given by a sequence $d_1 = \sqrt{d}(\eta - 1)$, $d_k = 2d_{k/2} + \sqrt{d}(\eta - 1)$. We bound d_k by $d_{2^{k'}}$ for $k' = \lceil \log_2(k) \rceil$, and get $d_{2^{k'}} = (2^{k'+1} - 1) \cdot \sqrt{d} \cdot (\eta - 1)$, which gives us the desired result. □

Lemma 4.3. *Let $p, m \geq 1$, and $k \geq 0$. For any replete ideal $I \subset K_{\mathbb{R}}$ there exists an integral ideal $\mathfrak{a}$ and an element $\alpha \in K_{\mathbb{R}}$ such that and $I^{2^k} = \alpha \cdot \mathfrak{a}$. with $\mathfrak{a}$ satisfying $\mathcal{N}(\mathfrak{a}) \leq B_{K,p,m,k}$ where*

$$\ln(B_{K,p,m,k}) \leq C \cdot d \cdot (\ln(m) + d + k + p + \ln(|\Delta_K|)).$$

for some absolute $C > 1$, and

$$\delta_{\mathrm{ideal}}(I^{2^k}/\mathcal{N}(I^{2^k})^{1/d}, \mathfrak{a}/\mathcal{N}(\mathfrak{a})^{1/d}) \leq 2^{-p}/m$$

Furthermore, if I is a fractional ideal, then there exists a polynomial time algorithm in size$(I), m, p, \ln(|\Delta_K|), k$ and d computing a basis of $\mathfrak{a}$ and a polynomial-size representation of α.

Proof. Let $\eta = 1 + 2^{-k-1-p}/(\sqrt{d} \cdot m)$. By Lemma D.1, $(\mathfrak{a}, (\alpha_i))$ can be computed (in polynomial time if I is fractional), with α_i η-balanced and

$$I^{2^k} = \mathfrak{a} \cdot \prod_{j=0}^{k} \alpha_j^{2^j}$$

We have $(\eta-1)^{-1} = 2^{k+1+p}\cdot\sqrt{d}\cdot m$, the bound on $\mathcal{N}(\mathfrak{a})$ then becomes

$$\begin{aligned}\mathcal{N}(\mathfrak{a}) &\le (2^{k+1+p+d}\cdot d^{3.5}\cdot m)^d\sqrt{|\Delta_K|}\\ &= m^d\cdot d^{3.5d}\cdot 2^{d^2}\cdot 2^{d(p+k+1)}\sqrt{|\Delta_K|}\\ &\le 2^{O\left(d\ln(m)+d^2+dk+dp\right)}\sqrt{|\Delta_K|}.\end{aligned}$$

Now we have, by Lemma D.2:

$$\begin{aligned}\delta_{\mathrm{ideal}}(I^{2^k}/\mathcal{N}(I^{2^k})^{1/d}, \mathfrak{a}/\mathcal{N}(\mathfrak{a})^{1/d}) &\le 2^{k+1}\cdot\sqrt{d}\cdot\left(2^{k+1+p}\cdot\sqrt{d}\cdot m\right)^{-1}\\ &= 2^{-p}/m,\end{aligned}$$

which allows to conclude the proof. □

E Error analysis of Section 5

E.1 Error analysis of Step 6 of Algorithm 4.1

We use the notations defined in Section 4. We define $\boldsymbol{R}_{\mathfrak{b}} = \mathrm{QR}(\boldsymbol{B}_{\mathfrak{b}})$. Our goal here is to bound the relative difference between $\boldsymbol{R}_{\mathfrak{b}}$ and $\boldsymbol{R}'_{\mathfrak{b}}$. By [CSV12, Theorem 6.4], there exists a matrix $\boldsymbol{B}''_{\mathfrak{b}}$ within distance $\mathrm{poly}(d)\cdot 2^{-p}\cdot\|\boldsymbol{B}'_{\mathfrak{b}}\| \le 2^{O(d)}\cdot 2^{-p}\cdot|\Delta_K|^{1/(2d)}$ of $\boldsymbol{B}'_{\mathfrak{b}}$ such that $\boldsymbol{R}'_{\mathfrak{b}} = \mathrm{QR}(\boldsymbol{B}''_{\mathfrak{b}})$, combining with Eq. (3) we have that

$$\|\boldsymbol{B}''_{\mathfrak{b}} - \boldsymbol{B}_{\mathfrak{b}}\| \le 2^{O(d)}\cdot 2^{-p}\cdot|\Delta_K|^{1/(2d)}.$$

Now, by [CSV12, Theorem 2.3] we have that as long as p is big enough ($p=\Omega(d)$ is enough),

$$\|\boldsymbol{R}'_{\mathfrak{b}} - \boldsymbol{R}_{\mathfrak{b}}\| \le \mathrm{poly}(d)\cdot\|\boldsymbol{B}_{\mathfrak{b}}\|\cdot\mathrm{cond}(\boldsymbol{B}_{\mathfrak{b}})\cdot\|\boldsymbol{B}''_{\mathfrak{b}} - \boldsymbol{B}_{\mathfrak{b}}\|$$

The fact that $\boldsymbol{B}_{\mathcal{O}_K}$ is LLL-reduced and $\boldsymbol{M}_{\mathfrak{b}}$ too, this implies by [CSV12, Lemma 5.5] that $\mathrm{cond}(\boldsymbol{B}_{\mathfrak{b}}) \le 2^{O(d)}$, which finally gives

$$\begin{aligned}\left\|\boldsymbol{R}'_{\mathfrak{b}}\cdot\boldsymbol{R}_{\mathfrak{b}}^{-1} - \boldsymbol{I}\right\| &\le \mathrm{poly}(d)\cdot 2^{-p}\cdot\mathrm{cond}(\boldsymbol{B}_{\mathfrak{b}})^2\cdot\|\boldsymbol{B}_{\mathfrak{b}}\|\\ &\le 2^{O(d)}\cdot 2^{-p}\cdot|\Delta_K|^{1/(2d)}.\end{aligned}$$

E.2 Error analysis of Step 9 of Algorithm 4.1

We use the notations defined in Section 4. In this section we are bounding the distance in trace norm between the state output by Step 9 of Algorithm 4.1 (that we denote $|\psi'\rangle$) and the state $F_{R,\sigma}(\mathfrak{b}')$. We denote by $\boldsymbol{B}_{\mathfrak{b}'} = \mathrm{diag}(\boldsymbol{\theta}, \boldsymbol{s'})\cdot\boldsymbol{B}_{\mathfrak{b}}$ a basis of $\mathfrak{b}'$.By the previous computations and the fact that a multiplication by $\mathrm{ExpEx}(\boldsymbol{s},\boldsymbol{\theta})$ preserves the norm, the state $|\phi'\rangle$ is an $\varepsilon/2$-approximation of

$$|\phi_{\mathfrak{b}'}\rangle = C''^{-1}\sum_{\substack{\boldsymbol{x}\in\mathbb{Z}^d\setminus\{0\}\\ \|\boldsymbol{B}_{\mathfrak{b}'}\boldsymbol{x}\|\le R}}\rho_\sigma(\boldsymbol{B}_{\mathfrak{b}'}\boldsymbol{x})|\boldsymbol{B}_{\mathfrak{b}'}\cdot x\rangle.$$

Now, since the Enc is computed with precision 2^{-p} (we denote the approximated version Enc') we have that

$$\left\|\mathrm{Enc}'|\phi'\rangle - F_{R,\sigma}(\mathfrak{b}')\right\| \leq \left\|\mathrm{Enc}' - \mathrm{Enc}\right\| + \||\phi'\rangle - |\phi_{\mathfrak{b}'}\rangle\| \leq 2^{-p} + \varepsilon/2.$$

F Complexity of Algorithm 4.1 step by step

Recall that we take our error parameter to be $\varepsilon = 2^{-\Theta(d)} \cdot |\Delta_K|^{\Theta(1)}$. This implies that we take $\sigma = (2^d \cdot |\Delta_K|^{1/d})^{O(1)}$, $R = (2^d \cdot |\Delta_K|^{1/d})^{O(1)}$, $t = (2^{-d} \cdot |\Delta_K|^{-1/d})^{\Omega(1)}$, $p = O\left(d + \ln(|\Delta_K|)\right)$, $q = O\left(d + \ln\left(|\Delta_K|^{1/d}\right)\right)$. Now, we fix $\tau = 2^{-\Theta(d)} \cdot |\Delta_K|^{\Theta(1)}$ to be the error parameter of Theorem 2.2, this give that

$$\begin{aligned} Q =& O\left((d+s)^{1+o(1)} \left(\ln(|\Delta_K|) + \ln\ln(N_S) + (d+s)\ln(\mathrm{Lip}(G_{R,\sigma}))\right) + \ln(\tau)\right) \\ =& O\left((d+s)^{2+o(1)} \left(\ln\left(|\Delta_K|^{1/d}\right) + d\right)\right) \end{aligned}$$

We now bound the complexity of each steps of Algorithm 4.1 for the parameter values we computed. For readability we will omit the O notation. We have that

$$m = Q \cdot (d_{\mathbb{R}} + d_{\mathbb{C}} + s) = (d+s)^{3+o(1)} \left(\ln\left(|\Delta_K|^{1/d}\right) + d\right),$$

and

$$\begin{aligned} \ln(B_{K,p,m,Q}) =& d\left(\ln(m) + d + Q + p\right) + \ln(|\Delta_K|) \\ =& d(d+s)^{2+o(1)} \left(\ln\left(|\Delta_K|^{1/d}\right) + d\right). \end{aligned}$$

Step 3. Memory:

$$\begin{aligned} & \left(d^{\omega+1} \cdot m \cdot \ln(B_{K,p,m,Q})\right)^{1+o(1)} \\ =& d^{\omega+2+o(1)} \cdot (d+s)^{5+o(1)} \cdot \left(\ln\left(|\Delta_K|^{1/d}\right) + d\right)^{2+o(1)}. \end{aligned}$$

Gate count:

$$\begin{aligned} & \left(d^{\omega+1} \cdot m^2 \cdot \ln(B_{K,p,m,Q})\right)^{1+o(1)} \\ =& d^{\omega+2+o(1)} \cdot (d+s)^{8+o(1)} \cdot \left(\ln\left(|\Delta_K|^{1/d}\right) + d\right)^{3+o(1)}. \end{aligned}$$

Step 4. Memory

$$\mathrm{LLLMem}\left(d, (d+s)^{5+o(1)} \cdot \left(\ln\left(|\Delta_K|^{1/d}\right) + d\right)^{2+o(1)}\right).$$

Gate count

$$\mathrm{LLLGates}\left(d, (d+s)^{5+o(1)} \cdot \left(\ln\left(|\Delta_K|^{1/d}\right) + d\right)^{2+o(1)}\right).$$

Step 5. Memory and gate count:

$$d^2 \cdot (d+s)^{5+o(1)} \cdot (\ln\left(|\Delta_K|^{1/d}\right) + d)^{2+o(1)}$$

Step 6 Memory and gate count:

$$d^3 \cdot (d + \ln(|\Delta_K|)))^{1+o(1)}$$

Step 7. Memory:

$$\tilde{O}\left(d^2(d+\ln(|\Delta_K|))^{5/2}\right)$$

Gate Count:

$$d(d+\ln(|\Delta_K|))$$

G Postprocessing: from an approximate basis of the log-S-units to the compact representation of S-units

The aim of this section is to quickly explain how to efficiently compute (a compact representation of) the S-unit α from an approximation of $(\mathrm{LogEx}(\alpha), (v_{\mathfrak{p}}(\alpha))_{\mathfrak{p}\in S})$. In other words, because, generally, $\alpha \in K$ is too large to be represented by its coefficients (its ordinary representation), it is rather written symbolically as $(\gamma_1, \ldots, \gamma_k)$, with which is meant that

$$\alpha = \prod_{i=1}^{k} \gamma_i^{2^i},$$

and where each of the γ_i are reasonably small. Such a representation is called the 'compact representation'.

Another aim of this section is to estimate how precise this approximation of $(\mathrm{LogEx}(\alpha), (v_{\mathfrak{p}}(\alpha))_{\mathfrak{p}\in S})$ (which is in this paper quantified by the error parameter τ) is required to be in order to make this computation of a compact representation of α possible.

This section uses similar techniques as in [BS25v2, Section 8.1], where the computation of compact representation of S-units is discussed. Our approach seems to differ in the fact that we keep track of the error required, and that we use the Buchmann-Pohst-Kessler algorithm in a way that keeps the *normalized* logarithmic embedding small, which is essential for our approach to be efficient.

Throughout this section, we assume that S is a finite set of prime ideals generating Cl_K and write $s = |S|$. First, we state a simplified version of a result from Buchmann and Kessler [BK96], about the stability of the LLL algorithm.

Lemma G.1. *Consequence of [BK96, Th. 4.1, Cor. 4.2] Let $n \geq 1$, and $\boldsymbol{A} \in \mathbb{R}^{(n+1)\times n}$ generating a rank n lattice $\mathcal{L}(\boldsymbol{A})$ in $\mathbb{R}^{n+1}$. Let $q \in \mathbb{Z}_{>0}$ satisfy*

$$2^q > \frac{(n+3)\cdot 2^{(n-3)/2}\cdot (n^{3/2}(n+1)^{1/2}/2+n)^n \|\boldsymbol{A}\|^n}{\det(\mathcal{L}(\boldsymbol{A}))\cdot\lambda_1(\mathcal{L}(\boldsymbol{A}))} = \frac{e^{O(n\ln n)}\|\boldsymbol{A}\|^n}{\det(\mathcal{L}(\boldsymbol{A}))\cdot\lambda_1(\mathcal{L}(\boldsymbol{A}))}.$$

Let $\hat{\boldsymbol{A}} = \lfloor 2^q \boldsymbol{A} \rceil \in \mathbb{Z}^{(n+1)\times n}$. *Let* $\hat{\boldsymbol{B}} = \hat{\boldsymbol{A}} \cdot \boldsymbol{U}$ *be the output of the Buchmann-Pohst-Kessler algorithm [BK96, §4] on* $\hat{\boldsymbol{A}}$, *where* $\boldsymbol{U} = [\boldsymbol{u}_1, \ldots, \boldsymbol{u}_n] \in GL_n(\mathbb{Z})$. *Let* $\boldsymbol{B} = [\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n] = \boldsymbol{A} \cdot \boldsymbol{U}$ *with* $\boldsymbol{b}_i \in \mathbb{R}^{n+1}$.

Then, for all $j \in \{1, \ldots, n\}$, *it holds that*

$$\begin{aligned}\|\boldsymbol{u}_j\| &\leq 2^{(n-1)/2+q+1} \cdot \lambda_j(\mathcal{L}(\boldsymbol{A})),\\ \|\boldsymbol{b}_j\| &\leq (n+2) \cdot 2^{(n-1)/2} \cdot \lambda_j(\mathcal{L}(\boldsymbol{A})).\end{aligned}$$

Proof. This is [BK96, Theorem 4.1], with instantiations $n_2 = n+1$ and $k = r = n$, $\mu = \lambda_1(\mathcal{L}(\boldsymbol{A}))$, $\alpha = \sqrt{n}\|\boldsymbol{A}\| \geq \max_i \|\boldsymbol{A}_i\|$, λ as in [BK96, Proposition 3.2] and using $\sqrt{n(n+1)} \leq n+1$.

Lemma G.2. *Let* $\tilde{\boldsymbol{B}}$ *be an approximation of a basis* $\boldsymbol{B}$ *of the log-S-unit lattice* Λ_S, *with error* $\tau < ((d+s)^{O(d+s)} \cdot |\Delta_K|)^{-(d+s)}$ *(with* $s = |S|$*); which satisfies* $\|\tilde{\boldsymbol{B}}\| \leq |\Delta_K|^4 \cdot (d+s)^{O(d+s)}$.

Then there exists an efficient algorithm that computes an $\left((d+s)^{d+s} \cdot |\Delta_K|\right)^{-\Omega(d+s)}$*-close approximation* $\tilde{\boldsymbol{O}}$ *of a basis* $\boldsymbol{O}$ *of the log-S-unit lattice for which each row* $\boldsymbol{o}$ *in* $\boldsymbol{O}$ *satisfies*

$$\boldsymbol{o} = [\mathrm{Log}(\gamma), (v_{\mathfrak{p}}(\gamma))_{\mathfrak{p}\in S}]$$

for some $\gamma \in K$ *with* $\|\mathrm{Log}(\gamma/N(\gamma)^{1/d})\| = O(1)$ *and* $\|(v_{\mathfrak{p}}(\gamma))_{\mathfrak{p}\in S}\| \leq 2^{(d+s)\ln(d+s)} \cdot |\Delta_K|$

Proof. Let $f_1 : \mathbb{R}^{d_{\mathbb{R}}+d_{\mathbb{C}}} \to \mathbb{R}^{d_{\mathbb{R}}+d_{\mathbb{C}}+1}$ be the transformation

$$(x_{\sigma_i})_{1\leq i\leq d_{\mathbb{C}}+d_{\mathbb{R}}} \mapsto [(x_{\sigma_i} - \bar{x})_{1\leq i\leq d_{\mathbb{C}}+d_{\mathbb{R}}}, \bar{x}],$$

where $\bar{x} = \frac{1}{d}\sum_i n_{\sigma_i} x_{\sigma_i}$ with $n_{\sigma_i} = 1$ if σ_i is real and 2 otherwise. Let $\boldsymbol{C}$ (resp $\tilde{\boldsymbol{C}}$) be the matrix obtained by applying f_1 to the $d_{\mathbb{R}} + d_{\mathbb{C}}$ first coordinates of $\boldsymbol{B}$ (resp $\tilde{\boldsymbol{B}}$). It can be shown[5] that applying f_1 changes the volume of a lattice by a factor $1 + \frac{1}{2d_{\mathbb{C}}+d_{\mathbb{R}}}$. It holds that $\left\|\boldsymbol{C} - \tilde{\boldsymbol{C}}\right\| \leq 2\tau$.

Let $C > 1$ be a constant to be fixed later, and $f_2 : \mathbb{R}^{d_{\mathbb{R}}+d_{\mathbb{C}}+1} \to \mathbb{R}^{d_{\mathbb{R}}+d_{\mathbb{C}}+1}$ be the transformation scaling the first $d_{\mathbb{R}} + d_{\mathbb{C}}$ coordinates by C and the last one by $C^{-(d_{\mathbb{R}}+d_{\mathbb{C}})}$. Note that $\det(f_2) = 1$. Let $\boldsymbol{D}$ (resp $\tilde{\boldsymbol{D}}$) be the matrix obtained by applying f_2 to the $d_{\mathbb{R}} + d_{\mathbb{C}} + 1$ first coordinates of $\boldsymbol{C}$ (resp $\tilde{\boldsymbol{D}}$). It holds that $\left\|\boldsymbol{D} - \tilde{\boldsymbol{D}}\right\| \leq 2 \cdot \tau \cdot C$.

[5] Write $n = d_{\mathbb{R}}+d_{\mathbb{C}}$. By the fact that elementary transformations (subtracting from one coordinate the value of the other) does not change the determinant, we can see that one could instead study the transformation $(x_{\sigma_i})_{1\leq i\leq d_{\mathbb{C}}+d_{\mathbb{R}}} \mapsto [(x_{\sigma_i})_{1\leq i\leq d_{\mathbb{C}}+d_{\mathbb{R}}}, \bar{x}]$, which can be described by a $(n+1)\times n$ matrix T that consists of an $n \times n$ identity matrix on the top and a bottom row consisting of all $1/d$ or $2/d$ depending on whether $1 \leq i \leq d_{\mathbb{R}}$ or $d_{\mathbb{R}} + 1 \leq i \leq d_{\mathbb{R}} + d_{\mathbb{C}}$ (where $d = 2d_{\mathbb{C}} + d_{\mathbb{R}}$). We write this bottom row by the vector $\boldsymbol{v}$. We have $T^\top T = I_n + \boldsymbol{v}\boldsymbol{v}^\top$ where $\boldsymbol{v}\boldsymbol{v}^\top$ is the 'outer product' yielding an $n \times n$ matrix. By the Weinstein-Aronszajn identity, we obtain $\det(T^\top T) = \det(I_n + \boldsymbol{v}\boldsymbol{v}^\top) = 1 + \boldsymbol{v}^\top\boldsymbol{v} = 1 + \frac{1}{d}$.

We now apply Lemma G.1 on the approximated matrix $\tilde{\boldsymbol{D}}$. We have $n = 2d_{\mathbb{R}} + d_{\mathbb{C}} + s$, and that $\boldsymbol{D}$ generates the lattice $\Lambda'_S = f_2(f_1(\Lambda_S)) \subset \mathbb{R}^{m+1}$. We take the minimal $q \in \mathbb{Z}_{>0}$ that satisfies the requirements of Lemma G.1.

We show at the end of this proof that the precision $\tau = ((d+s)^{O(d+s)} \cdot |\Delta_K|)^{-(d+s)}$ is sufficiently small in order to be able to compute $\hat{\boldsymbol{D}}$ (as in Lemma G.1). Then the Buchmann-Kessler-Pohst algorithm from Lemma G.1 yields a $\boldsymbol{U}$ such that the vectors of $\boldsymbol{E} = [\boldsymbol{e}_1, \dots, \boldsymbol{e}_n] = \boldsymbol{D}\boldsymbol{U}$ satisfy, for $i \in \{1, \dots, n\}$,

$$\|\boldsymbol{e}_j\| \le (n+2) \cdot 2^{(n-1)/2} \lambda_j(\mathcal{L}(\boldsymbol{D})).$$

and

$$\|\boldsymbol{u}_j\| \le 2^{(n-1)/2+q+1} \cdot \lambda_j(\mathcal{L}(\boldsymbol{D})).$$

With any vector $\boldsymbol{e}_j$ of $\boldsymbol{E}$ is associated an S-unit $\gamma_j \in K$, and it is clear that the first $d_{\mathbb{R}} + d_{\mathbb{C}}$ coefficients of $\boldsymbol{e}_j$ are equal to $C \cdot \operatorname{Log}(\gamma_j / N(\gamma_j)^{1/d})$. By taking $C \ge 1000\sqrt{d}\ln(d)^3$, we can, by a very similar reasoning as in Lemma J.3, it holds that $\|\boldsymbol{e}_j\| \ge \lambda_1(\mathcal{L}(\boldsymbol{D})) \ge C \cdot \lambda_1(\Lambda_S) \ge 1$, and hence, by Minkowski's second theorem, writing $n = d + s$ (with $s = |S|$),

$$\begin{aligned}
C \cdot \|\operatorname{Log}(\gamma_j / N(\gamma_j)^{1/d})\| \le \|\boldsymbol{e}_j\| &\le (n+2) \cdot 2^{(n-1)/2} \lambda_j(\mathcal{L}(\boldsymbol{D})) \\
&\le (n+2) \cdot 2^{(n-1)/2} \cdot n^{n/2} \cdot \det(\mathcal{L}(\boldsymbol{D})) \\
&\le 2(n+2) \cdot 2^{(n-1)/2} \cdot n^{n/2} \cdot \det(\Lambda_S) \\
&\le 2(n+2) \cdot 2^{(n-1)/2} \cdot n^{n/2} \cdot |\Delta_K|^4 \qquad (13) \\
&\le 2^{O(n \ln n)} \cdot |\Delta_K|^4. \qquad (14)
\end{aligned}$$

where the last inequality follows from Lemma J.2. Hence, by choosing $C = 2(n+2) \cdot 2^{(n-1)/2} \cdot n^{n/2} \cdot |\Delta_K|^4 = \exp(O(n \ln n)) \cdot |\Delta_K|^4$ this yields the claim of the $O(1)$ upper bound on $\|\operatorname{Log}(\gamma / N(\gamma)^{1/d})\|$ in the statement of this lemma.

For the bound on $\|v_{\mathfrak{p}}(\gamma))_{\mathfrak{p} \in S}\|$, a similar argument holds: $\|v_{\mathfrak{p}}(\gamma_j))_{\mathfrak{p} \in S}\| \le \|\boldsymbol{e}_j\|$, and hence the bound immediately follows from Equation (14).

The end output of the algorithm is $\tilde{\boldsymbol{O}} := \tilde{\boldsymbol{B}} \cdot \boldsymbol{U}$ (with exact analogue $\boldsymbol{O} := \boldsymbol{B} \cdot \boldsymbol{U}$). Hence, $\|\tilde{\boldsymbol{O}} - \boldsymbol{O}\| \le \|\boldsymbol{U}\| \cdot \|\tilde{\boldsymbol{B}} - \boldsymbol{B}\|$. We have, using that $\lambda_j(\mathcal{L}(\boldsymbol{D})) \le n^n \det(\mathcal{L}(\boldsymbol{D}))$, and using Lemma G.1,

$$\begin{aligned}
\|\boldsymbol{U}\| \le n \max_j \|\boldsymbol{u}_j\| \le n \cdot 2^{(n-1)/2+q+1} \cdot n^n \det(\mathcal{L}(\boldsymbol{D})) &\le \frac{2^{O(n \ln n)} \|\boldsymbol{D}\|^n \cdot \det(\mathcal{L}(\boldsymbol{D}))}{\det(\mathcal{L}(\boldsymbol{D})) \cdot \lambda_1(\mathcal{L}(\boldsymbol{D}))} \\
\le 2^{O(n \ln n)} \|\boldsymbol{D}\|^n \le 2^{O(n \ln n)} \cdot C^n \cdot \|\boldsymbol{B}\|^n &\le 2^{O(n^2 \ln n)} \cdot |\Delta_K|^n. \qquad (15)
\end{aligned}$$

Hence, (writing $n = d + s$) in order to have a $(n^n \cdot |\Delta_K|)^{-\Omega(n)}$-close approximation of $\boldsymbol{O}$, we must require $\tau < (n^{O(n)} \cdot |\Delta_K|)^{-n}$. Note that this τ is also sufficiently small in order to compute $\hat{\boldsymbol{D}}$. This finishes the proof.

Lemma G.3. *Let $\tilde{\boldsymbol{v}} = ((x_\sigma)_\sigma, (n_{\mathfrak{p}})_{\mathfrak{p}})$ be an approximation of the vector $\boldsymbol{v} = (\operatorname{LogEx}(\alpha), (v_{\mathfrak{p}}(\alpha))_{\mathfrak{p} \in S})$ with $\|\operatorname{Log}(\alpha / \ln(N(\alpha)^{1/d}))\| \le C = O(1)$, and $\|\boldsymbol{v}\| \le 2^{(d+s)\ln(d+s)} \cdot |\Delta_K|$, such that $\|\tilde{\boldsymbol{v}} - \boldsymbol{v}\|_\infty \le \varepsilon/2$ with $\varepsilon < 2^{-d} \cdot ((d+s)\ln(d+s) +$*

$\ln(\Delta_K))^{-\Omega(s)} < 1$. *Then there exists an algorithm running in time* $\mathrm{poly}(C, \ln|\Delta_K|, d, s)$ *that computes on input* $\tilde{\boldsymbol{v}} = ((x_\sigma)_\sigma, (n_\mathfrak{p})_\mathfrak{p})$, *elements* $\beta_1, \ldots, \beta_k \in K$ *such that*

$$\alpha = \prod_{i=1}^{k} \beta_i^{2^i}.$$

Proof. Since the error $\|\tilde{\boldsymbol{v}} - \boldsymbol{v}\|_\infty$ is bounded above by a half, we know exactly the values of $v_\mathfrak{p}(\alpha)$, since they are integer. Hence we deduce

$$(\alpha) = \prod_{\mathfrak{p} \in S} \mathfrak{p}^{v_\mathfrak{p}(\alpha)} =: \mathfrak{a}.$$

This ideal is enormous and cannot be computed directly. Instead we use Algorithm D.1, and the results Lemmas D.1 and D.2 to compute, for any $\mathfrak{p} \in S$, polynomially-sized elements $\gamma_{1,\mathfrak{p}}, \ldots \gamma_{k,\mathfrak{p}} \in K$ with $k = O((d+s)\ln(d+s) + \ln(|\Delta_K|)$ and an integral ideal $\mathfrak{b}_\mathfrak{p}$ for which $\ln N(\mathfrak{b}_\mathfrak{p})$ is polynomial in $\ln|\Delta_K|$, such that

$$\mathfrak{p}^{v_\mathfrak{p}(\alpha)} = \prod_{i=1}^{k} (\gamma_{i,\mathfrak{p}})^{2^i} \mathfrak{b}_\mathfrak{p}$$

for which

$$\delta_{\text{ideal}}\left(\frac{\mathfrak{b}_\mathfrak{p}}{N(\mathfrak{b}_\mathfrak{p})^{1/d}}, \frac{\mathfrak{p}^{v_\mathfrak{p}(\alpha)}}{N(\mathfrak{p}^{v_\mathfrak{p}(\alpha)})^{1/d}}\right) < \varepsilon/(2 \cdot s).$$

This takes time $\mathrm{poly}(C, \ln C', \ln|\Delta_K|, \ln s)$. We compute a compact representation $(\gamma_1, \ldots, \gamma_k, \mathfrak{b})$ (with $\mathfrak{b}$ integral) of $\mathfrak{a}$ by multiplying component-wise the compact representations of the $\mathfrak{p}^{v_\mathfrak{p}(\alpha)}$, still in polynomial time, with

$$\delta_{\text{ideal}}(\mathfrak{b}/N(\mathfrak{b})^{1/d}, \mathfrak{a}/N(\mathfrak{a})^{1/d}) < \varepsilon/2$$

By Lemma 4.3, it holds that $\mathcal{N}(\mathfrak{b}) = ((d+s)\ln(d+s) + \ln(C') + \ln(1/\varepsilon) + \ln(\Delta_K))^{O(d \cdot s)}$.

Since $\mathfrak{a} = (\alpha)$ is principal, $\mathfrak{b}$ is, too. We have that there exists a u with $\|\operatorname{Log}(u)\| < \varepsilon/2$ such that

$$u \cdot (\alpha)/N(\alpha)^{1/d} = \mathfrak{b}/N(\mathfrak{b})^{1/d},$$

and hence there must be an element $\beta = u \cdot N(\mathfrak{b})^{1/d}/N(\mathfrak{a})^{1/d} \cdot \alpha$ such that $(\beta) = \mathfrak{b}$. In other words,

$$\begin{aligned}
\|\alpha N(\mathfrak{b})^{1/d}/N(\mathfrak{a})^{1/d} - \beta\| &= \|u-1\|\|\alpha N(\mathfrak{b})^{1/d}/N(\mathfrak{a})^{1/d}\| \\
&\leq \varepsilon \cdot ((d+s)\ln(d+s) + \ln(1/\varepsilon) + \ln(\Delta_K))^{O(s)} \\
&\leq \varepsilon \cdot \ln(1/\varepsilon)^{O(s)} \cdot ((d+s)\ln(d+s) + \ln(\Delta_K))^{O(s)} \\
&\leq \sqrt{\varepsilon} \cdot s^{O(s)} \cdot ((d+s)\ln(d+s) + \ln(\Delta_K))^{O(s)} \\
&\leq \sqrt{\varepsilon} \cdot ((d+s)\ln(d+s) + \ln(\Delta_K))^{O(s)},
\end{aligned}$$

where we used that $\|\operatorname{LogEx}(\alpha/N(\alpha)^{1/d})\| \leq C = O(1)$ and that for all $x \geq 1$, $\ln(x)^s/\sqrt{x} \leq s^{O(s)}$, which holds by classical real analysis.

So, $t = \alpha N(\mathfrak{b})^{1/d}/N(\mathfrak{a})^{1/d}$ is a BDD-instance for $\beta \in \mathfrak{b}$ with error $\sqrt{\varepsilon} \cdot ((d+s)\ln(d+s) + \ln(\Delta_K))^{O(s)}$, and the size of t is $\operatorname{poly}(C, \ln|\Delta_K|, s)$.

By LLL-reducing the integral $\mathfrak{b}$, we obtain Gram-Schmidt vectors of a basis of $\mathfrak{b}$ that are in norm lower bounded by 2^{-d} (since such Gram-Schmidt vectors satisfy $\|\boldsymbol{b}_{i+1}^*\| \geq 3/4 \cdot \|\boldsymbol{b}_i^*\|$ for all i [NV09, Chapter 2], and $\|\boldsymbol{b}_1^*\| = \|\boldsymbol{b}_1\| \geq 1$ by integrality of $\mathfrak{b}$). Using the Babai round-off algorithm [Bab85] for solving this BDD instance, we can retrieve β, whenever $\sqrt{\varepsilon}\cdot((d+s)\ln(d+s)+\ln(\Delta_K))^{O(s)} < 2^{-d}$ (which is true by assumption). This yields the result.

Proposition G.1. *For the algorithm of [dBDF20] described in Section 2.7, it is sufficient to take $\tau = |\Delta_K|^{-(d+s)}(d+s)^{O(-(d+s)^2)}$ with $s := |S|$ in order for the output approximated basis allowing for computing a compact representation of all S-units.*

Proof. By Lemmas G.2 and G.3 (and the fact that $\left((d+s)^{d+s} \cdot |\Delta_K|\right)^{-\Omega(d+s)} < 2^{-d} \cdot ((d+s)\ln(d+s) + \ln(\Delta_K))^{-\Omega(s)}$) it is sufficient to show that $\|\tilde{\boldsymbol{B}}\| \leq |\Delta_K|^4 \cdot (d+s)^{O(d+s)}$. But this follows readily from [dBDF20, Corollary 6] (which effectively says $\|\tilde{\boldsymbol{B}}\| \leq 2^{3m}/\lambda_1(\Lambda_S^*)$, where m is the rank of $\boldsymbol{B}$) and the bound for $\lambda_1(\Lambda_S^*)$ in Appendix J.1.

H A result on almost-Lipschitz periodic functions

In this section, we prove the following result.

Theorem H.1. *Let m be an integer, $\alpha \in (0, 1/4), \varepsilon \in (0,1)$, $A, \nu > 0$, $\Lambda \subset \mathbb{R}^m$ a full rank lattice and $\mathcal{H}$ a Hilbert space. Let $f : \mathbb{R}^m \to \mathcal{H}$ that is (A, α)-almost Lipschitz and (ν, ε)-separative. Then there exists a function $g : \mathbb{R}^m \to \mathcal{H}$ that is $O(m \cdot A)$ Lipschitz, $(\nu, \varepsilon + 8\alpha)$-separative such that*

$$\max_{\mathbf{x} \in \mathbb{R}^m} \|g(\mathbf{x}) - f(\mathbf{x})\| \leq 4 \cdot \alpha$$

Proof. The majority of the proof is done in Lemma H.2. We only need to prove the separativity. Let $\mathbf{x}, \mathbf{y} \in \mathbb{R}^m$ such that $\|\mathbf{x} - \mathbf{y}\| \geq \nu$, then we have

$$|\langle g(\mathbf{x})|g(\mathbf{y})\rangle| \leq |\langle f(\mathbf{x})|g(\mathbf{y})\rangle| + 4\alpha \leq |\langle f(\mathbf{x})|f(\mathbf{y})\rangle| + 8\alpha \leq \varepsilon + 4\alpha.$$

□

The method to prove the result is to define a mollified version of the almost-Lipschitz function, to show that this mollified version is close to the original function and still have the relevant properties.

Definition H.1. *For $x \in \mathbb{R}$, we define*

$$\delta(x) = \begin{cases} \eta^{-2} \cdot (x + \eta) & \text{if } x \in [-\eta, 0], \\ \eta^{-2} \cdot (\eta - x) & \text{if } x \in [0, \eta], \\ 0 & \text{otherwise} \end{cases}$$

This function has the shape of a isosceles triangle starting from $-\eta$ and ending at η, having height $1/\eta$ (at zero). It is symmetric around zero, positive, is supported on $[-\eta, \eta]$, integrates to 1 and has maximum absolute slope η^{-2}.

For $\mathbf{x} = (x_1, \ldots, x_m) \in \mathbb{R}^m$, we set $\delta(\mathbf{x}) = \prod_{i=1}^m \delta(x_i)$.

Lemma H.1. *The function $\delta : \mathbb{R}^n \to \mathbb{R}$ is symmetric around zero, positive, is supported on $[-\eta, \eta]^m$, integrates to 1 and has maximum absolute slope $n\eta^{-(m+1)}$ and satisfies*

$$\int_{\mathbf{a}\in\mathbb{R}^m} |\delta(\mathbf{x}-\mathbf{a}) - \delta(\mathbf{y}-\mathbf{a})| d\mathbf{a} \leq 2\sqrt{m} \cdot \eta^{-1} \cdot \|\mathbf{x}-\mathbf{y}\|_2 \tag{16}$$

for any $\mathbf{x}, \mathbf{y} \in \mathbb{R}^m$.

Proof. The fact that δ is symmetric around zero (i.e., $\delta(-\mathbf{x}) = \delta(\mathbf{x})$), positive, and is supported on $[-\eta, \eta]^n$ follows from the definition of δ on $\mathbb{R}$ in Definition H.1. The function δ integrates to 1 since it can be integrated componentwise. The statement about the maximum absolute slope follows from an application of the product rule $|\frac{\partial}{\partial x_i}\delta(\mathbf{x})| \leq \eta^{-2} \prod_{j\neq i} \delta(x_j)$. Hence

$$\|\nabla\delta\| \leq \|\nabla\delta\|_1 \leq \eta^{-2} \cdot \sum_{i=1}^m \prod_{j\neq i} \delta(x_j) \leq m \cdot \eta^{-(m+1)}.$$

We finish with the proof of the statement in Equation (16), where we assume $\mathbf{y} = 0$ without loss of generality and use the fact that δ is symmetric. Note that from the trick $rs - r'\sigma' = (r-r')\sigma - r'(\sigma'-\sigma)$ with $r = \delta(a_1 - x_1)$ and $r' = \delta(a_1)$ follows that

$$\begin{aligned}\Big|\prod_{i=1}^m \delta(a_i - x_i) - \prod_{i=1}^m \delta(a_i)\Big| \leq |\delta(a_1 - x_1) - \delta(a_1)| \prod_{i=2}^m \delta(a_i - x_i) \\ + \delta(a_1)\Big|\prod_{i=2}^m \delta(a_i - x_i) - \prod_{i=2}^m \delta(a_i)\Big|\end{aligned}$$

Therefore, since $\int_{a_2,\ldots,a_m} \prod_{i=2}^m \delta(a_i - x_i) da_2 \ldots da_m = 1$ and $\int_{a_1} \delta(a_1) da_1 = 1$, we obtain

$$\int_{\mathbf{a}\in\mathbb{R}^m} |\delta(\mathbf{a}-\mathbf{x}) - \delta(\mathbf{a})| d\mathbf{a} = \int_{a_1} \ldots \int_{a_m} \Big|\prod_{i=1}^m \delta(a_i - x_i) - \prod_{i=1}^m \delta(a_i)\Big| da_1 \ldots da_m$$

$$\leq \int_{a_1} |\delta(a_1 - x_1) - \delta(a_1)| da_1 + \int_{a_2} \ldots \int_{a_m} \Big|\prod_{i=2}^m \delta(a_i - x_i) - \prod_{i=2}^m \delta(a_i)\Big| da_2 \ldots da_m$$

$$\leq \sum_{i=1}^m \int_{a_i} |\delta(a_i - x_i) - \delta(a_i)| da_i \leq \sum_{i=1}^m 2\eta^{-1}|x_i| = 2\eta^{-1}\|\mathbf{x}\|_1 \leq 2\sqrt{m} \cdot \eta^{-1} \cdot \|\mathbf{x}\|_2.$$

The second inequality follows from induction, and the third by inspection: assuming without loss of generality that $x_i > 0$, the function $|\delta(a_i + x_i) - \delta(a_i)|$ looks like a isosceles trapezoid (with a 'puncture' in the middle), see Figure 1. Hence its surface area is at most $2\eta \cdot \eta^{-2} x_i = 2\eta^{-1} x_i$.

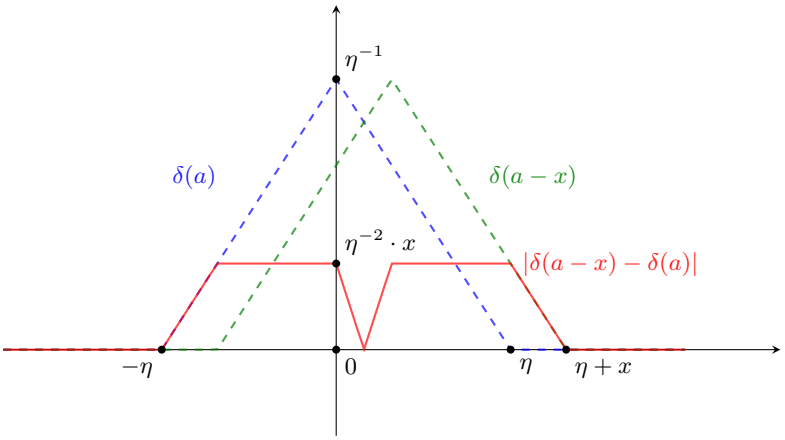

Fig. 1. The surface of $|\delta(a-x)-\delta(a)|$ can be upper bounded by the surface of the isosceles trapezoid with base length $2\eta+x$ and top length $2\eta-x$ and height $\eta^{-2}\cdot x$. This surface equals $2\eta\cdot x$.

Lemma H.2. *Let $\Lambda\subseteq\mathbb{R}^m$ be a lattice, let $\mathcal{H}$ be a Hilbert space and let $\mathcal{S}\subseteq\mathcal{H}$ be the unit vectors in that Hilbert space. Let $a\in\mathbb{R}_{>0}$ and $\alpha\in[0,\frac{1}{4}]$, and let $f:\mathbb{R}^m\to\mathcal{S}$ be a Λ-periodic function that satisfies $\|f(\mathbf{x})-f(\mathbf{y})\|\leq a\|\mathbf{x}-\mathbf{y}\|+\alpha$ for all $\mathbf{x},\mathbf{y}\in\mathbb{R}^m$; that is, f is (a,α)-almost Lipschitz.*

Then there exists a Λ-periodic function $g:\mathbb{R}^m\to\mathcal{S}$ such that,

$$\max_{\mathbf{x}\in\mathbb{R}^m}\|g(\mathbf{x})-f(\mathbf{x})\|\leq 4\cdot\alpha$$

and

$$\|g(\mathbf{x})-g(\mathbf{y})\|\leq 24\cdot m\cdot a\|\mathbf{x}-\mathbf{y}\| \text{ for all } \mathbf{x},\mathbf{y}\in\mathbb{R}^m,$$

that is, g is Λ-periodic, 4α-close to f, and $(24\cdot m\cdot a)$-Lipschitz.

Proof. **Definition of g**
We first put

$$g_0(\mathbf{x}):=(f\star\delta)(\mathbf{x}):=\int_{\mathbf{a}\in\mathbb{R}^m}f(\mathbf{a})\delta(\mathbf{x}-\mathbf{a})d\mathbf{a} \tag{17}$$

with $\delta:\mathbb{R}^m\to\mathbb{R}$ as in Definition H.1, with $\eta=\alpha/(\sqrt{m}\cdot a)$. Since $\mathcal{H}$ is a complete space, and $g_0(\mathbf{x})=\int_{\mathbf{a}\in\mathbb{R}^m}f(\mathbf{a})\delta(\mathbf{x}-\mathbf{a})d\mathbf{a}$ can be seen as a limit (for a fixed $\mathbf{x}\in\mathbb{R}^m$), we have that $g_0(\mathbf{x})$ is a well-defined value in $\mathcal{H}$ for every $\mathbf{x}$. In order to have $g(\mathbf{x})\in\mathcal{S}$ (i.e., the elements in $\mathcal{H}$ with norm 1), we put

$$g(\mathbf{x})=\frac{g_0(\mathbf{x})}{\|g_0(\mathbf{x})\|} \tag{18}$$

g is Λ-periodic.

As $g(\cdot) = \frac{g_0(\cdot)}{\|g_0(\cdot)\|}$, it is sufficient to show that g_0 as in Equation (17) is Λ-periodic. For all $\mathbf{x} \in \mathbb{R}^m$ and $\ell \in \Lambda$, we have $g_0(\mathbf{x}+\ell) = (f \star \delta)(\mathbf{x}+\ell) = \int_{\mathbf{a}} f(\mathbf{x}+\ell-\mathbf{a})\delta(\mathbf{a})d\mathbf{a} = \int_{\mathbf{a}} f(\mathbf{x}-\mathbf{a})\delta(\mathbf{a})d\mathbf{a} = (f \star \delta)(\mathbf{x}) = g_0(\mathbf{x})$.

g is close to f.

Write $f(\mathbf{x}) = \int_{\mathbf{a}\in\mathbb{R}^m} f(\mathbf{x})\delta(\mathbf{a})d\mathbf{a}$ (use that δ integrates to 1). Then, using the definition of g_0 (Equation (17)), we obtain

$$\begin{aligned}\|f(\mathbf{x}) - g_0(\mathbf{x})\| &= \int_{\mathbf{a}\in\mathbb{R}^m} \|f(\mathbf{x}) - f(\mathbf{x}-\mathbf{a})\| \cdot \delta(\mathbf{a})d\mathbf{a} \leq \int_{\mathbf{a}\in\mathbb{R}^m} (a\|\mathbf{a}\| + \alpha)\delta(\mathbf{a})d\mathbf{a} \\ &= \alpha + a\int_{\mathbf{a}\in\mathbb{R}^m} \|\mathbf{a}\|\delta(\mathbf{a})d\mathbf{a} \\ &\leq \alpha + a \max_{\mathbf{a}\in[-\eta,\eta]^m} \|\mathbf{a}\| \cdot \int_{\mathbf{a}\in\mathbb{R}^m} \delta(\mathbf{a})d\mathbf{a} \leq \alpha + a\sqrt{m}\cdot\eta \leq 2\alpha.\end{aligned} \tag{19}$$

where we use that $\delta(\mathbf{a})$ only has support on $[-\eta,\eta]^m$, and where we use the instantiation $\eta = \alpha/(\sqrt{m}a)$. Since this inequality holds for all $\mathbf{x} \in \mathbb{R}^m$, we obtain $\max_{\mathbf{x}\in\mathbb{R}^m} \|f(\mathbf{x}) - g_0(\mathbf{x})\| \leq 2\alpha$.

By the 'reverse triangle inequality', we have $|\|g_0(\mathbf{x})\| - 1| \leq 2\alpha$ for every $\mathbf{x} \in \mathbb{R}^m$. Writing $a = g_0(\mathbf{x})$ and $b = g(\mathbf{x}) = a/\|a\|$, we have $|\|a\| - 1| \leq 2\alpha$ and $\|b\| = 1$. We can therefore deduce

$$\|g_0(\mathbf{x}) - g(\mathbf{x})\| = \|a-b\| = \frac{1}{\|a\|}\Big\|\|a\|\cdot a - a\Big\| = \frac{1}{\|a\|}\Big\|(\|a\|-1)\cdot a\Big\| = |\|a\|-1| \leq 2\alpha \tag{20}$$

Combining Equation (19) and Equation (20) we thus obtain

$$\max_{\mathbf{x}\in\mathbb{R}^m} \|g(\mathbf{x}) - f(\mathbf{x})\| \leq 4\cdot\alpha.$$

g is Lipschitz.

We first focus on the Lipschitz constant of the function $g_0(\cdot)$. At the end of this proof we will show that the Lipschitz constant of $g(\cdot) = \frac{g_0(\cdot)}{\|g_0(\cdot)\|}$ is then obtained by multiplying the Lipschitz constant of g by $\frac{2}{1-2\alpha} \leq 4$ (by the assumption $\alpha \in [0, \frac{1}{4}]$).

We distinguish two cases, namely $\|\mathbf{x}-\mathbf{y}\| \geq \alpha/a$ and $\|\mathbf{x}-\mathbf{y}\| < \alpha/a$. For $\|\mathbf{x}-\mathbf{y}\| \geq \alpha/a$, we have $\|f(\mathbf{x}) - f(\mathbf{y})\| \leq a\|\mathbf{x}-\mathbf{y}\| + \alpha \leq 2a\|\mathbf{x}-\mathbf{y}\|$. So

$$g_0(\mathbf{x}) - g_0(\mathbf{y}) = (f\star\delta)(\mathbf{x}) - (f\star\delta)(\mathbf{y}) = \int_{\mathbf{t}\in\mathbb{R}^n} (f(\mathbf{x}-\mathbf{t}) - f(\mathbf{y}-\mathbf{t}))\delta(\mathbf{t})d\mathbf{t}. \tag{21}$$

Therefore, by the triangle inequality, the positivity of δ, the fact that the distance between $\mathbf{x}-\mathbf{t}$ and $\mathbf{y}-\mathbf{t}$ is the same as the distance between $\mathbf{x}$ and $\mathbf{y}$, and the

fact that δ integrates to 1,

$$\begin{aligned}\|g_0(\mathbf{x}) - g_0(\mathbf{y})\| &\leq \int_{\mathbf{t}\in\mathbb{R}^m} \|f(\mathbf{x}-\mathbf{t}) - f(\mathbf{y}-\mathbf{t})\|\delta(\mathbf{t})d\mathbf{t} \\ &\leq 2a \int_{\mathbf{t}\in\mathbb{R}^m} \|\mathbf{x}-\mathbf{y}\|\delta(\mathbf{t})d\mathbf{t} \leq 2\cdot a\|\mathbf{x}-\mathbf{y}\|.\end{aligned}$$

So it remains to show Lipschitzianity for $\mathbf{x},\mathbf{y}\in\mathbb{R}^m$ that are closer to each other than α/a. For this, we assume $\|\mathbf{x}-\mathbf{y}\| \leq \alpha/a$, which implies $\|f(\mathbf{x})-f(\mathbf{y})\| \leq 2\alpha$. Put $\mathbf{a} = (\mathbf{x}+\mathbf{y})/2$ for the average of these points. Write $f_0(\cdot) = f(\cdot) - f(\mathbf{a})$. Then

$$\|f_0(\mathbf{t})\| = \|f(\mathbf{t}) - f(\mathbf{a})\| \leq a\|\mathbf{t}-\mathbf{a}\| + \alpha \leq 3\alpha$$

for all $\mathbf{t}\in\mathbb{R}^m$ satisfying $\|\mathbf{t}-\mathbf{a}\| \leq 2\alpha/a$. Notice that points $\mathbf{t}$ satisfying $\|\mathbf{t}-\mathbf{x}\| \leq \alpha/a$ or $\|\mathbf{t}-\mathbf{y}\| \leq \alpha/a$ satisfy $\|\mathbf{t}-\mathbf{a}\| \leq 2\alpha/a$. We have

$$\begin{aligned}g_0(\mathbf{x}) - g_0(\mathbf{y}) &= (f\star\delta)(\mathbf{x}) - (f\star\delta)(\mathbf{y}) = (f_0\star\delta)(\mathbf{x}) - (f_0\star\delta)(\mathbf{y}) \\ &= \int_{\mathbf{t}\in\mathbb{R}^m} f_0(\mathbf{t})[\delta(\mathbf{x}-\mathbf{t}) - \delta(\mathbf{y}-\mathbf{t})]d\mathbf{t}. \qquad (22)\end{aligned}$$

Now choose $\eta = \frac{\alpha}{\sqrt{m}a}$, such that $\delta(\cdot)$ is supported on the α/a-ball around 0. Hence, the integrand of Equation (22) is nonzero only if $\|\mathbf{t}-\mathbf{a}\| \leq 2\alpha/a$, hence only if $\|f_0(\mathbf{t})\| \leq 3\alpha$. Therefore, Equation (22) is bounded by

$$\begin{aligned}3\alpha\cdot\int_{\mathbf{t}\in[-\eta,\eta]^m} |\delta(\mathbf{x}-\mathbf{t}) - \delta(\mathbf{y}-\mathbf{t})|d\mathbf{t} &\leq 3\alpha\cdot 2\sqrt{m}\cdot\eta^{-1}\cdot\|\mathbf{x}-\mathbf{y}\| \\ &= 6\cdot m\cdot a\cdot\|\mathbf{x}-\mathbf{y}\|,\end{aligned}$$

where we use the property in Equation (16) of Lemma H.1.

Note that, by definition, $g(\mathbf{x}) = \frac{g_0(\mathbf{x})}{\|g_0(\mathbf{x})\|}$. Hence, writing $a = g(\mathbf{x}), a' = g(\mathbf{y}), b = g_0(\mathbf{x}), b' = g_0(\mathbf{y})$, we have $a = b/\|b\|$, $a' = b'/\|b'\|$ and $\|b'\|, \|b\| \in (1-2\alpha, 1+2\alpha)$ (by the fact that g_0 is 2α-close to f). Hence, we obtain

$$\|g(\mathbf{x}) - g(\mathbf{y})\| = \|a-a'\| = \|b\|^{-1}\left\|\frac{\|b\|}{\|b'\|}b' - b\right\|.$$

$$= \|b\|^{-1}\left(\|b'-b\| + \left|\frac{\|b\|}{\|b'\|} - 1\right|\|b'\|\right) \leq \frac{2}{1-2\alpha}\cdot\|b'-b\| = \frac{2}{1-2\alpha}\cdot\|g_0(\mathbf{x})-g_0(\mathbf{y})\|.$$

Hence, the Lipschitz constant of g is bounded by $\frac{12\cdot m\cdot a}{1-2\alpha} \leq 24\cdot m\cdot a$ (by the assumption that $\alpha\in[0,\frac{1}{4}]$.

I Arithmetic over algebraic objects

In this section we show how we represent algebraic elements and how we perform classical number theoretic algorithms. In the quantum setting, we do not have specific quantum versions of these algorithm, but rather simulate the classical algorithms by means of Theorem 2.1.

Arithmetic Real numbers are represented in fixed point precision 2^{-p}, for a value of p determined in Section 4. We have that

- The sum of two integers $x, y \leq B$ can be computed with $O(\ln(B))$ quantum gates and memory.
- The multiplication of two integers x, y can be computed with $O(\ln(B)^{1+o(1)})$ quantum gates and memory using FFT.
- The inversion of an integer $x \leq B$ can be computed with $O((\ln(B)+p)^{1+o(1)})$ quantum gates and memory.
- Operations over real numbers are done by multiplying the real numbers by 2^p (yielding an integer by fixed precision 2^{-p}) and performing operations on the resulting integers. So, for real numbers the complexities for addition, multiplication and inversion is the same as above, with $\ln(B)$ being replaced by $\ln(B) + p$.

I.1 Representation of algebraic objects

Representation of the field We assume that the field K is given by a defining polynomial P_K of degree d such that $K = \mathbb{Q}[X]/P_K$ with[6] $\text{size}(P_K) = \widetilde{O}(\ln^2(|\Delta_K|))$. We assume that a $\mathbb{Z}$-basis of elements $(\omega_1, \ldots, \omega_d)$ for the ring of integers $\mathcal{O}_K$ is given, with the additional property that $\boldsymbol{B}_{\mathcal{O}_K} = [\Phi(\omega_1), \ldots, \Phi(\omega_d)]$ which is LLL-reduced. Here, every ω_i is given as a rational polynomial P_{ω_i} of degree $\leq d-1$ (seen as element in $K = \mathbb{Q}[X]/P_K$). We define $|\Delta_K| = \max_i \|\Phi(\omega_i)\|_\infty$ and $s_K = \max \text{size}(P_{\omega_i})$ where $\text{size}(P)$ is the bit-size of the rational polynomial P. Since $\boldsymbol{B}_{\mathcal{O}_K}$ is LLL-reduced, we have $\ln(|\Delta_K|) = O(d + \ln\left(|\Delta_K|^{1/d}\right))$. Let $\boldsymbol{P}_{\mathcal{O}_K} \in \mathbb{Q}^{d\times d}$ the column matrix of the P_{ω_i} over the basis $1, X, \ldots, X^{d-1}$, we assume that the matrix $\boldsymbol{P}^{-1}_{\mathcal{O}_K} \in \mathbb{Q}^{d\times d}$ is pre-computed and given. This $\boldsymbol{P}_{\mathcal{O}_K}$ and $\boldsymbol{P}^{-1}_{\mathcal{O}_K}$ then allows to go back and forth efficiently between the polynomial representation of an element $\alpha \in K$ and its representation with respect to the basis $(\omega_1, \ldots, \omega_d)$.

Representation of elements of K Every $x \in \mathcal{O}_K$ is represented as a tuple $((x_i)_{i=1,\ldots,d}, P_x) \in \mathbb{Z}^d \times \mathbb{Q}[X]$, with $x = \sum_i x_i\omega_i$ and P_x is the representation of x as an element of $\mathbb{Q}[X]/P_K$. By LLL-reduction of $\boldsymbol{B}_{\mathcal{O}_K}$ we have that $\max_i |x_i| \leq 2^d\|\Phi(x)\|$. We have that $P_x = \sum_i x_i P_{\omega_i}$, the bit-size of P_x is then bounded by $d \cdot (\ln(\max(x_i)) + s_K)$. Finally, the bit-size of $x \in \mathcal{O}_K$ is bounded by

$$\text{size}(x) \leq d \cdot \max_i \text{size}(x_i) + \text{size}(P_x) = O\left(d \cdot (d + s_K + \ln(\|x\|))\right)$$

For any $x \in K$, there exists $N \in \mathbb{Z}_{>0}$ such that $N \cdot x \in \mathcal{O}_K$. Such an element is then represented by the triple $(N, ((y_i)_{i=1,\ldots,d}, P_y))$ with $y = N \cdot x \in \mathcal{O}_K$. Such a representation of of $x \in K$ then satisfies

$$\text{size}(x) = \text{size}(y/N) \leq \ln(N) + \text{size}(y)$$

[6] Such a polynomial always exists and can be found efficiently, given a good basis of the ring of integers, see [BPW25, §A.4]

Representation of ideals of K There are two ways to represent integral ideals. The first one is a $\mathbb{Z}$-basis in Hermite normal form (HNF) over $\boldsymbol{B}_{\mathcal{O}_K}$. When an ideal $\mathfrak{a}$ is represented by a matrix $\boldsymbol{H}_\mathfrak{a}$, we have that $\mathcal{N}(\mathfrak{a}) = \det(\boldsymbol{H}_\mathfrak{a})$, and the size of every element in $\boldsymbol{H}_\mathfrak{a}$ is bounded by $\mathcal{N}(\mathfrak{a})$, we then have

$$\text{size}(\boldsymbol{H}_\mathfrak{a}) \leq d^2 \cdot \ln(\mathcal{N}(\mathfrak{a}))$$

The other way to represent the ideal $\mathfrak{a}$ is to write it in two-element representation: $\mathfrak{a} = (x_\mathfrak{a}) + (y_\mathfrak{a})$. In this case, the size of the two-element representation is just the sum of the sizes of $x_\mathfrak{a}$ and $y_\mathfrak{a}$.
Every fractional ideal can be written $I = \mathfrak{a}/N$, with $N \in \mathbb{Z}_{>0}$ and $\mathfrak{a}$ an integral ideal. We represent the ideal I by the tuple $(N, \mathfrak{a})$ for which then holds

$$\text{size}(I) = \text{size}(\mathfrak{a}/N) = \ln(N) + \text{size}(\mathfrak{a})$$

I.2 Algorithms on K

We will now provide the time and memory complexity of the operations on the algebraic objects discussed. We write $\omega \in (2, 3]$ for the polynomial exponent of the complexity of matrix multiplication over $\mathbb{Z}$. Addition of two integers of size B can be performed in time $O(B)$, multiplication can be performed in time $O(B^{1+o(1)})$ using the Schönhage-Strassen algorithm.
The addition of two elements $x, y \in \mathcal{O}_K$ is done coordinate-wise within time $\tilde{O}(d \cdot \text{size}(x, y))$. Multiplication of two elements is done by subsequently applying fast-Fourier multiplication on the respective polynomial parts of their representation, yielding a polynomial representation of the product, and apply $\boldsymbol{P}_{\mathcal{O}_K}^{-1}$ to obtain the $\mathcal{O}_K$-basis representation as well. This can be done within $\tilde{O}(d^2 \cdot \text{size}(x, y)^{1+o(1)})$ operations.

Let $\mathfrak{a} = (x_\mathfrak{a}) + (y_\mathfrak{a})$ be an ideal given in two element representation and $\mathfrak{b}$ given by its HNF $[\boldsymbol{b_1}, \ldots, \boldsymbol{b_d}]$. The HNF of $\mathfrak{a} \cdot \mathfrak{b}$ is computed by computing $\boldsymbol{B'} = [x_\mathfrak{a} \cdot \boldsymbol{b_1}, \ldots, x_\mathfrak{a} \cdot \boldsymbol{b_d}, y_\mathfrak{a} \cdot \boldsymbol{b_1}, \ldots, y_\mathfrak{a} \cdot \boldsymbol{b_d}]$, computing the HNF $[\boldsymbol{H} \in \mathbb{Z}^{d \times d}, \boldsymbol{0}^{d \times d}]$ of $\boldsymbol{B'}$ and then computing the polynomial representation of the elements represented by the columns of $\boldsymbol{H}$ by applying $\boldsymbol{P}_{\mathcal{O}_K}^{-1}$. The first computation can be done within time $\tilde{O}(d^3 \cdot \text{size}(\mathfrak{b}, x_\mathfrak{a}, y_\mathfrak{a})]$ operations; the HNF is computed in time [SL96] $O((d^{\omega+1} \ln(\max(\|x_\mathfrak{a}\|, \|y_\mathfrak{a}\|) \cdot \mathcal{N}(\mathfrak{b})))^{1+o(1)})$, and the final matrix multiplication is done in time $O(d^\omega \cdot \ln(\mathcal{N}(\mathfrak{a} \cdot \mathfrak{b})))$. So, the cost of computing the HNF of the ideal $\mathfrak{a} \cdot \mathfrak{b}$ given the two-element representation of $\mathfrak{a}$ and the HNF of $\mathfrak{b}$ is bounded by

$$O\left((d^{\omega+1} \left(\text{size}(x_\mathfrak{a}) + \text{size}(y_\mathfrak{a}) + \ln(\mathcal{N}(\mathfrak{a}\mathfrak{b}))\right))^{1+o(1)}\right)$$

J Computations for Λ_S

In this section, we compute bounds on the lattice invariants of Λ_S. Recall that

$$\Lambda_S = \left\{ (\boldsymbol{x}, \boldsymbol{a}) \in (\text{Arg}_K \oplus \text{Log}\, K_\mathbb{R}^0) \times \mathbb{Z}^s, \quad \text{ExpEx}(\boldsymbol{x}) \cdot \prod_{i=1}^{s} (\mathfrak{p}_i / \mathcal{N}(\mathfrak{p}_i)^{1/d})^{a_i} = \mathcal{O}_K \right\}$$
$$\subset \mathbb{R}^{2(d_\mathbb{R} + d_\mathbb{C}) + s}.$$

We prove here the following proposition, some of those follows from computations in [BPW25, (eprint)]:

Lemma J.1. *It holds that*

- $\mathrm{Vol}(\Lambda_S) \leq |\Delta_K|^4$,
- $\lambda_1(\Lambda_S) \geq (\mathrm{poly}(d))^{-1}$,
- $\lambda_1(\Lambda_S^\star) \geq (2d+s)^{-(2d+s)/2+1} \cdot |\Delta_K|^{-4}$.

Lemma J.2. *If S is a set of prime ideals generating the class group, then it holds that* [7] $\mathrm{Vol}(\Lambda_S) = 2^{d_\mathbb{R}} \cdot (2\pi)^{d_\mathbb{C}} \cdot h_K \cdot \mathrm{Vol}(\mathrm{Log}(\mathcal{O}_K{}^\times)) = 2^{d_\mathbb{R}} \cdot (2\pi)^{d_\mathbb{C}} \cdot h_K \cdot R_K \cdot \sqrt{d_\mathbb{R} + d_\mathbb{C}}$, *where h_K is the class number of K, R_K is the regulator, $d_\mathbb{R}$ is the number of real embeddings and $d_\mathbb{C}$ is the number of complex pairs of embeddings. In particular, $\mathrm{Vol}(\Lambda_S)$ does not depend on the choice of S (as long as it generates the class group). Moreover, we have*

$$\ln(\mathrm{Vol}(\Lambda_S)) \leq 4 \ln |\Delta_K|.$$

Proof. The lemma follows from the fact that, for S generating the class group, there exists a surjective map $\mathrm{span}_\mathbb{R}(\Lambda_S)/\Lambda_S$ to Pic_K^0 in [dBDPW20], whose kernel is precisely the extended logarithm of the group $\{x \in K_\mathbb{R}^\times \mid |\sigma_i(x)| = 1 \text{ for all } i = 1, \ldots, d\}$, which has volume $2^{d_\mathbb{R}} \cdot (2\pi)^{d_\mathbb{C}}$, as it is isomorphic to $\{\pm 1\}^{d_\mathbb{R}} \times (S^1)^{d_\mathbb{C}}$ (where S^1 denotes the unit circle).

Hence, using a bound by Louboutin [Lou00] on the residue ρ_K of the Dedekind zeta function at $s = 1$, applying the class number formula [NS13, VII.§5, Cor 5.11], we obtain

$$\begin{aligned}
\mathrm{Vol}(\mathrm{Pic}_K^0) \leq h_K \cdot R_K \cdot \sqrt{d_\mathbb{R} + d_\mathbb{C}} &= \frac{\rho_K \cdot \sqrt{|\Delta_K|} \cdot |\mu_K| \cdot \sqrt{d_\mathbb{R} + d_\mathbb{C}}}{2^{d_\mathbb{R}} \cdot (2\pi)^{d_\mathbb{C}}} \\
&\leq \sqrt{|\Delta_K|} \cdot \rho_K \leq \sqrt{|\Delta_K|} \cdot \left(\frac{e \ln |\Delta_K|}{2(d-1)}\right)^{d-1} \\
&\leq |\Delta_K|.
\end{aligned}$$

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

The final bound on $\ln(\mathrm{Vol}(\Lambda_S))$ then follows from the fact that $\ln\left(2^{d_\mathbb{R}} \cdot (2\pi)^{d_\mathbb{C}}\right) \leq d \ln(2\pi)/2 \leq 3 \ln |\Delta_K|$, since $\ln |\Delta_K| \geq \ln(\pi/2)d$, by Minkowski's bound.

The first minimum of the log-S-unit lattice Λ_S can be lower bounded by applying Kessler's lower bound on the first minimum of the (ordinary) log-unit lattice ([Kes91]).

[7] We understand $\mathrm{Vol}(\Lambda_S)$ as the Volume of the disconnected quotient group.

Lemma J.3. *For any set of prime ideals S, it holds that $\lambda_1(\Lambda_S) \geq \frac{1}{1000\sqrt{d}\ln(d)^3}$.*

Proof. Let $\alpha \in \mathcal{O}_{K,S}^\times$ be such that $\mathrm{Log}_S(\alpha)$ reaches the first minimum of Λ_S. If $\alpha \notin \mathcal{O}_K^\times$ (and hence has a prime ideal divisor, say $\mathfrak{p}$), $\|\mathrm{Log}_S(\alpha)\| \geq |v_\mathfrak{p}(\alpha)| \geq 1 \geq \frac{1}{1000\sqrt{d}\ln(d)^3}$.

For $\alpha \in \mathcal{O}_K^\times$ we apply the lower bound of Kessler [Kes91] to obtain $\|\mathrm{Log}_S(\alpha)\| = \|\mathrm{Log}(\alpha)\| \geq \frac{1}{1000\sqrt{d}\ln(d)^3}$.

J.1 Bound on the first minimum of Λ_S^*

Let $m = d_\mathbb{R} + d_\mathbb{C} - 1$ be the rank of the unit group of $\mathcal{O}_K$. The upper bound on $\lambda_1(\Lambda_S^\star)$ is computed using transference lemmas and Minkowski's second theorem. By Minkowski's second theorem we have that,

$$\lambda_1(\Lambda_S) \cdot \ldots \cdot \lambda_{2m+s+1}(\Lambda_S) \leq (2m+s+1)^{(2m+s+1)/2} \cdot \mathrm{Vol}(\Lambda_S),$$

the lower bound on $\lambda_1(\Lambda_S)$ and the upper bound on $\mathrm{Vol}(\Lambda_S)$ give that (note that $2m+s+1 \leq 2d+s$):

$$\lambda_{2m+s+1}(\Lambda_S) \leq (2d+s)^{(2d+s)/2} \cdot |\Delta_K|^4,$$

and finally, the transference theorem [Ban93] gives us that there exists a constant $c_2 > 0$ such that

$$\lambda_1(\Lambda_S^\star) \geq (2d+s)^{-(2d+s)/2+1} \cdot |\Delta_K|^{-4}.$$

K The CHSP theorem for S-units

Theorem K.1 (From [Boe22, Theorem 3.3]). *Let $\delta, a, \lambda, \lambda^\star, D > 0$, $m, n \geq 1$. Let $\mathcal{H}$ a quantum Hilbert space of dimension 2^n. There exists*

$$Q = O\left(mk + \ln\left(\frac{a}{\lambda^\star \cdot \tau}\right)\right), \quad V = O\left(\frac{m\sqrt{n}}{\lambda^\star \cdot \tau}\right)$$

and a quantum procedure which, given oracle access to a function $\mathbf{f} : \mathbb{R}^m \to \mathcal{S} \subseteq \mathcal{H}$ satisfying

- *Λ-periodic for a full rank lattice $\Lambda \subset \mathbb{R}^m$ satisfying*
  - $\det(\Lambda) \leq D$,
  - $\lambda_1(\Lambda) \geq \lambda$.
  - $\lambda_1(\Lambda^\star) \geq \lambda^\star$
- *$\mathbf{f}$ is a-Lipschitz over $V\mathbb{D}$,*
- *$\mathbf{f}$ is (ν, ε)-separative for $\nu < \lambda/6$ and $\varepsilon \leq 1/4$,*

outputs with constant success probability an approximate basis $\tilde{\boldsymbol{B}} = \boldsymbol{B} + \boldsymbol{\Delta_B}$ *of the lattice* Λ *satisfying* $\|\boldsymbol{\Delta_B}\| \leq \tau$ *and* $\left\|\tilde{\boldsymbol{B}}\right\| \leq (d+s)^{O(d+s)} \cdot |\Delta_K|^4$. *This procedure makes* $k = O(m \ln(\sqrt{m} \cdot a \cdot D^{1/m}))$ *oracle calls to* $\boldsymbol{f}$, *and uses* $mQ+n$ *qubits,* $O\left(kmQ \cdot (\ln(kmQ))^2\right)$ *quantum gates and* $\mathrm{poly}(m, \ln(a/\lambda))$ *classical bit operations. All calls to* $\mathbf{f}$ *are made on the set* $V\mathbb{D}$ *where*

$$V\mathbb{D} = \frac{V}{2^Q} \cdot [\![-2^Q, 2^Q]\!]^m$$

The theorem above is a modified version of [Boe22, Theorem 3.3], where the bound on $\|\boldsymbol{B}\|$ follows from [dBDF20, Corollary 6] (which says $\|\tilde{\boldsymbol{B}}\| \leq 2^{3m}/\lambda_1(\Lambda_S^*)$, where m is the rank of $\boldsymbol{B}$) and the bound for $\lambda_1(\Lambda_S^*)$ in Appendix J.1. We emphasize the following facts:

- The query set over which $\mathbf{f}$ is called is $V\mathbb{D}$, it is finite.
- We give bounds for the invariants of the lattice instead of putting these invariants directly inside the algorithm parameter.

Note that the previously introduced quantum procedure solves the CHSP assuming that the function takes input in $\mathbb{R}^m$ for $m \in \mathbb{Z}_{>0}$, the function presented takes values in $\mathbb{R}^d \times \mathbb{Z}^s$, we therefore need to modify it. The way to do this is presented in [EHKS14b, Section 6.1]. The hypothesis in [EHKS14b, Appendix F]does not match the hypothesis of [EHKS14b, Theorem 6.1], so we re-state this result here. The result can be summarized by the following lemma.

Lemma K.1. *Let* $d, s \in \mathbb{Z}_{>0}$, *let* $\lambda \geq 1$ *and* $\mathcal{H}$ *a qubit space with* n *qubits. Let* $\Lambda \subset \mathbb{R}^d \oplus \mathbb{Z}^s$ *a lattice of* $\mathbb{R}^{d+s}$ *and* $f : \mathbb{R}^d \oplus \mathbb{Z}^s \to \mathcal{H}$ *a function satisfying*

- f *is* a*-Lipschitz,*
- f *is* (ν, ε)*-separative, for* $\nu < 1/4$.
- f *is* Λ*-periodic for some full rank lattice* $\Lambda \subset \mathbb{R}^d \oplus \mathbb{Z}^s \subset \mathbb{R}^{d+s}$.
- f *is efficiently computable as a quantum algorithm.*

Assume that $\lambda_1(\Lambda) \geq \lambda$. *There exists an efficiently computable function* $f' : \mathbb{R}^{d+s} \to \mathcal{H}'$ *which has the following properties.*

1. f' *is* a'*-Lipschitz on* $\mathbb{R}^{d+s}$ *for* $a' = a + O(\sqrt{s}/\nu)$.
2. f' *is* Λ'*-periodic , where* $\Lambda' = \{(\boldsymbol{r}, \lambda \boldsymbol{n}) \mid (\boldsymbol{r}, \boldsymbol{n}) \in \Lambda\}$, *where* $\boldsymbol{r} \in \mathbb{R}^d$ *and* $\boldsymbol{n} \in \mathbb{Z}^s$.
3. f' *is* (ν', ε)*-separative for* $\nu' = (1 + 3\lambda)\nu$.
4. *The codomain* $\mathcal{H}'$ *of* f' *is a qubit space with* $n + O(s \cdot \ln(1/\nu))$ *qubits.*
5. *A call to* f' *at the point* $(x_1, \ldots, x_d, r_1, \ldots, r_s)$ *makes exactly one call in superposition at* f *at the points*

$$\{(x_1, \ldots, x_d, \lfloor r_1/\lambda \rfloor + z_1, \ldots, \lfloor r_s/\lambda \rfloor + z_s), \quad (z_1, \ldots, z_s) \in \{-1, 0, 1\}^s\}$$

Proof. We provide this proof for completeness (as per request by a reviewer), but it is almost verbatim from [EHKS14b, Section 6.1, Section F].

We will first define the function f', which just uses the same straddle encoding (see Appendix M) as in the rest of the paper, with the addition that we keep track of the 'remainders of the straddles' as well (see [EHKS14b, Section 6.1, Section F]).

Using $t = \nu$ (an inverse integer) for the straddle parameter (see Definition M.2) we define f' by the following rule, where $\{\cdot\}$ denotes the fractional part of a real number.

$$(\boldsymbol{x}, \boldsymbol{r}) \longmapsto \sum_{\boldsymbol{z}\in\{0,1\}^s} c_{\boldsymbol{z}}(\boldsymbol{r}) \cdot f(\boldsymbol{x}, \lfloor \boldsymbol{r}/\lambda \rfloor - \boldsymbol{z}) \otimes \mathrm{Str}_\nu^{\otimes s}(\{\boldsymbol{r}/\lambda\} + \boldsymbol{z}),$$

where $c_{\boldsymbol{z}}(\boldsymbol{r})$ are the coefficients arising from the s-fold straddle encoding (Definition M.1),

$$(\mathrm{Str}_t)^{\otimes s}(\boldsymbol{r}) = \sum_{\boldsymbol{z}\in\{0,1\}^s} c_{\boldsymbol{z}}(\boldsymbol{r}) \cdot |\lfloor \boldsymbol{r}/t \rfloor + \boldsymbol{z}\rangle,$$

more explicitly, $c_{\boldsymbol{z}}(\boldsymbol{r}) = \prod_{i=1}^{s} \sin(\pi\{r_i/\nu\}/2)^{1-z_i} \cos(\pi\{r_i/\nu\}/2)^{z_i}$.

Note that by the efficiency of the straddle encoding together with the efficiency of f, indeed, f' is efficiently computable and satisfies property (5). Additionally, the codomain of f' requires $O(s \cdot \ln(1/\nu))$ extra qubits, which proves property (4) for f'.

We now proceed with proving properties (1) – (3).

1. To prove Lipschitzianity, we compute the derivatives. If the function is continuous and differentiable almost everywhere, the Lipschitz constant is equal to the maximum Euclidean norm of the (full) derivative of f'.
   We start with showing that the function f' is continuous. The only parts where f' might be discontinuous is where $r_i = \lambda j$ for $j \in \mathbb{Z}$, for some $i \in \{1, \ldots, s\}$. We give an exemplaric proof that f' is continuous for the case $s = 1$. The case for general $s \in \mathbb{Z}_{>0}$ follows very similarly (but with more book keeping). For the case $s = 1$, the function reads

   $$f'(\boldsymbol{x}, r) = c_0(r) f(\boldsymbol{x}, \lfloor r/\lambda \rfloor) \otimes \mathrm{Str}_\nu(\{r/\lambda\}) + c_1(r) f(\boldsymbol{x}, \lfloor r/\lambda \rfloor - 1) \otimes \mathrm{Str}_\nu(\{r/\lambda\} + 1)$$

   Again, as an example, we show continuity at $r = 0$, since the case $r = \lambda j$ for some $j \in \mathbb{Z}$ follows similarly. We have $\lim_{r\downarrow 0} c_z(r) = z$ and $\lim_{r\uparrow 0} c_z(r) = 1 - z$ for $z \in \{0, 1\}$. Furthermore, $\lim_{r\downarrow 0} \mathrm{Str}_\nu(\{r/\lambda\} + z) = \mathrm{Str}_\nu(z)$ and $\lim_{r\uparrow 0} \mathrm{Str}_\nu(\{r/\lambda\} + z) = \mathrm{Str}_\nu(1 + z)$ (by continuity of Str_ν).
   Now we compute the right- and lefthand limit of $f'(\boldsymbol{x}, r)$ for $r \to 0$, where we omit the $\boldsymbol{x}$ from the computations.

   $$\lim_{r\downarrow 0} f'(r) = 0 \cdot f(0)\, \mathrm{Str}_\nu(0) + f(-1)\, \mathrm{Str}_\nu(1) = f(-1)\, \mathrm{Str}_\nu(1)$$

   $$\lim_{r\uparrow 0} f'(r) = 1 \cdot f(-1) \cdot \mathrm{Str}_\nu(1) + 0 \cdot f(-2) \cdot \mathrm{Str}_\nu(0) = f(-1)\, \mathrm{Str}_\nu(1)$$

   Hence, indeed, for the case $s = 1$, the function f' is continuous.

We proceed by computing the derivatives. With respect to $\boldsymbol{x}$, one can show that, using the Pythagorean theorem and the assumption that $\nu < 1/4$ (so that the straddle states $\mathrm{Str}_\nu^{\otimes s}(\{\boldsymbol{r}/\lambda\} + \boldsymbol{z})$ are pair-wise orthogonal),

$$\|\nabla_{\boldsymbol{x}} f'(\boldsymbol{x}, \boldsymbol{r})\|^2 = \left\| \sum_{\boldsymbol{z} \in \{0,1\}^s} c_{\boldsymbol{z}}(\boldsymbol{r}) \cdot \nabla_{\boldsymbol{x}} f(\boldsymbol{x}, \lfloor \boldsymbol{r}/\lambda \rfloor - \boldsymbol{z}) \otimes \mathrm{Str}_\nu^{\otimes s}(\{\boldsymbol{r}/\lambda\} + \boldsymbol{z}) \right\|^2$$

$$\leq \sum_{\boldsymbol{z} \in \{0,1\}^s} |c_{\boldsymbol{z}}(\boldsymbol{r})|^2 \cdot \|\nabla_{\boldsymbol{x}} f(\boldsymbol{x}, \lfloor \boldsymbol{r}/\lambda \rfloor - \boldsymbol{z})\|^2 \leq a^2,$$

where a is the Lipschitz constant of f. For the derivative with respect to $\boldsymbol{r}$, we compute the first partial derivative (as the others are very similar) of a single summand of $f'(\boldsymbol{x}, \boldsymbol{r})$. Almost everywhere, $f(\boldsymbol{x}, \lfloor \boldsymbol{r}/\lambda \rfloor + \boldsymbol{z})$ is constant for varying $\boldsymbol{r}$, hence we can leave it out of the computation, yielding, by the Pythagorean theorem (and the fact that the straddle states are orthogonal)

$$\|\frac{\partial}{\partial r_1} f'(\boldsymbol{x}, \boldsymbol{r})\|^2 = \sum_{\boldsymbol{z}} \left\| \frac{\partial}{\partial r_1} [c_{\boldsymbol{z}}(\boldsymbol{r}) \, \mathrm{Str}_\nu^{\oplus s}(\{\boldsymbol{r}/\lambda\} + \boldsymbol{z})] \right\|^2$$

$$\sum_{\hat{\boldsymbol{z}}} |c_{\hat{\boldsymbol{z}}}(\hat{\boldsymbol{r}})|^2 \sum_{z_1} \| \frac{\partial}{\partial r_1} [c_{z_1}(r_1) \, \mathrm{Str}_\nu(\{r_1/\lambda\} + z_1)] \|^2$$

Here, $\hat{\boldsymbol{z}}$ and $\hat{\boldsymbol{r}}$ are the vectors $\boldsymbol{z}$ and $\boldsymbol{r}$ respectively, where the first coordinate is omitted. Since the Lipschitz constant of the straddle encoding (with the extra factor λ) is at most $\pi/(2\nu\lambda)$, and since the derivative of $c_{z_1}(r_1) = \sin(\pi\{r_1/\nu\}/2)^{1-z_1} \cos(\pi\{r_1/\nu\}/2)^{z_1}$ equals $\pi/(2\nu) \cos(\pi\{r_1/\nu\}/2)$ if $z_1 = 0$ and $-\pi/(2\nu) \cdot \sin(\pi\{r_1/\nu\}/2)$ otherwise, we arrive at an upper bound of

$$2 \sum_{\hat{\boldsymbol{z}}} |c_{\hat{\boldsymbol{z}}}(\hat{\boldsymbol{r}})|^2 [(c_{z_1}(r_1) \cdot \pi/(2\nu\lambda) + \pi/(2\nu))^2]$$

$$\leq (\pi/(2\nu))^2 (1 + 1/\lambda)^2$$

Summing over all coordinates $i \in \{1, \ldots, s\}$, we arrive at a Lipschitz constant of at most $a + \sqrt{s} \cdot \frac{\pi(1+1/\lambda)}{2\nu}$. This proves the claim.

2. For Λ'-periodicity, suppose that $(\boldsymbol{u}, \lambda\boldsymbol{n}) \in \Lambda'$. Then, using that ν is an inverse integer,

$$f'(\boldsymbol{x}+\boldsymbol{u}, \boldsymbol{r}+\lambda\boldsymbol{n}) = \sum_{\boldsymbol{z} \in \{0,1\}^s} c_{\boldsymbol{z}}(\boldsymbol{r}) \cdot f(\boldsymbol{x}+\boldsymbol{u}, \lfloor \boldsymbol{r}/\lambda \rfloor + \boldsymbol{n} - \boldsymbol{z}) \otimes \mathrm{Str}_\nu^{\oplus s}(\{\boldsymbol{r}/\lambda\} + \boldsymbol{z})$$

which we can rewrite, by periodicity of f, into

$$= \sum_{\boldsymbol{z} \in \{0,1\}^s} c_{\boldsymbol{z}}(\boldsymbol{r}) \cdot f(\boldsymbol{x}, \lfloor \boldsymbol{r}/\lambda \rfloor - \boldsymbol{z}) \otimes \mathrm{Str}_\nu^{\oplus s}(\{\boldsymbol{r}/\lambda\} + \boldsymbol{z}) = f'(\boldsymbol{x}, \boldsymbol{r}).$$

3. For seperativity, recall Definition 2.2, assume that two points $(\boldsymbol{x}, \boldsymbol{r}), (\boldsymbol{x'}, \boldsymbol{r'})$ are at least distance $> (1+3\lambda)\nu$ from each other (with respect to the quotient metric of Λ'). Assume that the inner product $\langle f'(\boldsymbol{x}, \boldsymbol{r}))|f'(\boldsymbol{x'}, \boldsymbol{r'})\rangle$ is nonzero, i.e., there exist $\boldsymbol{z}, \boldsymbol{z'} \in \{0,1\}^s$ such that $\mathrm{Str}_\nu^{\oplus s}(\{\boldsymbol{r}/\lambda\}+\boldsymbol{z})$ and $\mathrm{Str}_\nu^{\oplus s}(\{\boldsymbol{r'}/\lambda\}+\boldsymbol{z'})$ have non-zero inner product. This means, by total 2ν-separativity of Str_ν (see Lemma M.1) that any such pair $(\boldsymbol{z}, \boldsymbol{z'})$ for which the regarding inner product is non-zero, satisfies

$$\|\{\boldsymbol{r'}/\lambda\} + \boldsymbol{z'} - \{\boldsymbol{r}/\lambda\} - \boldsymbol{z}\| \leq 2\nu. \tag{23}$$

Since $\nu < 1/4$, any such pair $(\boldsymbol{z}, \boldsymbol{z'})$ with non-zero corresponding inner product satisfies $\boldsymbol{z'} - \boldsymbol{z} = \boldsymbol{c}$, a constant vector. Hence, for each $\boldsymbol{z} \in \{0,1\}^s$ there is a unique $\boldsymbol{y_z} \in \{0,1\}^s$ such that $\mathrm{Str}_\nu^{\oplus s}(\{\boldsymbol{r}/\lambda\} + \boldsymbol{z})$ and $\mathrm{Str}_\nu^{\oplus s}(\{\boldsymbol{r'}/\lambda\} + \boldsymbol{y_z})$ have non-zero inner product.
Hence, $\langle f'(\boldsymbol{x}, \boldsymbol{r}))|f'(\boldsymbol{x'}, \boldsymbol{r'})\rangle$ then equals

$$\sum_{\boldsymbol{z}\in\{0,1\}^s} c_{\boldsymbol{z}}(\boldsymbol{r}) c_{\boldsymbol{y_z}}(\boldsymbol{r'}) \cdot \langle f(\boldsymbol{x}, \lfloor \boldsymbol{r}/\lambda \rfloor - \boldsymbol{z})|f(\boldsymbol{x'}, \lfloor \boldsymbol{r'}/\lambda \rfloor - \boldsymbol{y_z})\rangle \langle \mathrm{Str}_\nu^{\oplus s}\{\boldsymbol{r}/\lambda\} + \boldsymbol{z})|\, \mathrm{Str}_\nu^{\oplus s}\{\boldsymbol{r'}/\lambda\} + \boldsymbol{y_z})\rangle \tag{24}$$

If we assume that the absolute value of all $\langle f(\boldsymbol{x}, \lfloor \boldsymbol{r}/\lambda \rfloor - \boldsymbol{z})|f(\boldsymbol{x'}, \lfloor \boldsymbol{r'}/\lambda \rfloor - \boldsymbol{y_z})\rangle$ are bounded by ε, we can indeed conclude that

$$|\langle f'(\boldsymbol{x}, \boldsymbol{r}))|f'(\boldsymbol{x'}, \boldsymbol{r'})\rangle| \leq \varepsilon,$$

by the fact that the absolute inner products of the straddle encodings are at most 1 and by an application of the Cauchy-Schwarz inequality (since the square sum of the $c_{\boldsymbol{z}}(\boldsymbol{r})$ equals one).
So, to derive a contradiction, assume that for some pair $\boldsymbol{z}, \boldsymbol{y_z}$, we have $|\langle f(\boldsymbol{x}, \lfloor \boldsymbol{r}/\lambda \rfloor - \boldsymbol{z})|f(\boldsymbol{x'}, \lfloor \boldsymbol{r'}/\lambda \rfloor - \boldsymbol{y_z})\rangle| \geq \varepsilon$. Then, by separativity of f, we can directly deduce that $\|(\boldsymbol{x} - \boldsymbol{x'}, \lfloor \boldsymbol{r}/\lambda \rfloor - \lfloor \boldsymbol{r'}/\lambda \rfloor - \boldsymbol{z} + \boldsymbol{y_z})\|_\Lambda \leq \nu$ with respect to the quotient metric of Λ. I.e., there is a $(\boldsymbol{u}, \boldsymbol{v}) \in \Lambda$ such that

$$\|(\boldsymbol{x} - \boldsymbol{x'} - \boldsymbol{u}, \lfloor \boldsymbol{r}/\lambda \rfloor - \lfloor \boldsymbol{r'}/\lambda \rfloor - \boldsymbol{z} + \boldsymbol{y_z} - \boldsymbol{v})\| \leq \nu$$

Then, using both above expression and Equation (23),

$$\begin{aligned}
&\|(\boldsymbol{x} - \boldsymbol{x'}, \boldsymbol{r} - \boldsymbol{r'})\|_{\Lambda'} \leq \|(\boldsymbol{x} - \boldsymbol{x'} - \boldsymbol{u}, \boldsymbol{r} - \boldsymbol{r'} - \lambda\boldsymbol{v})\| \\
\leq &\underbrace{\|(\boldsymbol{x} - \boldsymbol{x'} - \boldsymbol{u}, \lambda(\lfloor \boldsymbol{r}/\lambda \rfloor - \lfloor \boldsymbol{r'}/\lambda \rfloor - \boldsymbol{z} + \boldsymbol{y_z} - \boldsymbol{v}))\|}_{\leq (1+\lambda)\nu} \\
&+ \underbrace{\|(0, \boldsymbol{r} - \boldsymbol{r'} - \lambda(\lfloor \boldsymbol{r}/\lambda \rfloor - \lfloor \boldsymbol{r'}/\lambda \rfloor - \boldsymbol{z} + \boldsymbol{y_z}))\|}_{= \|\lambda\{\boldsymbol{r}/\lambda\} + \lambda\boldsymbol{z} - \lambda\{\boldsymbol{r'}/\lambda\} - \lambda\boldsymbol{y_z})\|} \\
\leq &(1+\lambda)\nu + 2\lambda\nu = (1+3\lambda)\nu.
\end{aligned}$$

This is a contradiction with the initial assumption, hence f' is $((1+3\lambda)\nu, \varepsilon)$-separative.

Theorem 2.2. *Let $\mathcal{H}$ be a qubit space of dimension 2^n, $\alpha \in (0, 1/32)$ and $\tau \in (0,1)$ be error parameters, $A \geq 1$ and $\nu \leq \mathrm{poly}(d)^{-1}$ two real numbers. Let $\mathbf{f} : \mathbb{R}^{2(d_{\mathbb{R}}+d_{\mathbb{C}})-1+s} \to \mathcal{H}$ be a quantum function that is (A, α)-almost-Lipschitz, $(\nu, 1/4 - 8\alpha)$-separative and Λ_S-periodic.*

Then there exist a quantum procedure

- *making k oracle calls to $\mathbf{f}$ over the set $V\mathbb{D}_Q^{2(d_{\mathbb{R}}+d_{\mathbb{C}})-1+s}$,*
- *using $O((d+s)Q+n)$ qubits,*
- *using $O\left(kQ(d+s)\cdot(\ln(kQ(d+s)))^2\right)$ quantum gates,*
- *$\mathrm{poly}(s, \ln(a))$ classical bit operations,*

which outputs with probability $\geq 1/2 - 4k\alpha$ a matrix $\tilde{\boldsymbol{B}}$ for which holds that $\left\|\boldsymbol{B} - \tilde{\boldsymbol{B}}\right\| \leq \tau$, where $\boldsymbol{B}$ is a basis of Λ_S satisfying $\|\boldsymbol{B}\| \leq (d+s)^{O(d+s)} \cdot |\Delta_K|^4$. Here, $Q, k \in \mathbb{Z}_{>0}$, $V \in \mathbb{R}_{>0}$ satisfy

- $Q = O\big((d+s)^2(\ln(d+s)+\ln(A)) + (d+s)\ln|\Delta_K| + \ln(1/\tau)\big)$
- $V \geq 1$ *with* $\ln(V) = O\big((d+s)\ln(d+s) + \ln(n) + \ln(|\Delta_K|) + \ln(1/\tau)\big)$,
- $k = O\big((d+s)[\ln(d+s)+\ln(A)] + \ln(|\Delta_K|)\big)$,

Proof. This theorem follows from combining Theorem K.1 and Lemma K.1, instantiating the relevant parameters from Appendix J, with the exception of the almost Lipschitz-continuity. We mitigate this discrepancy by utilizing Theorem H.1, which states that a (A, α)-almost Lipschitz, separative and Λ_S-periodic function is 4α-close in the maximum norm to a (fully) Lipschitz, separative (with slightly worse parameters) and Λ_S-periodic function.

The loss in probability comes from the fact that this latter, fully Lipschitz function is approximated within error 4α, and that the oracle is queried k times.

L Quantum computing the discrete Gaussian state

L.1 A quantum version of the algorithm of Gentry, Peikert and Vaikuntanathan

Recall the definition of the Gaussian function from Section 2.4. For $\boldsymbol{c}, \boldsymbol{x} \in \mathbb{R}^m$, we denote $\rho_{\sigma,\boldsymbol{c}}(\boldsymbol{x}) = e^{-\pi\|\boldsymbol{x}-\boldsymbol{c}\|^2/\sigma^2}$. The dimension m of the Gaussian is often left implicit.

Definition L.1 (Periodized Discrete Gaussian). *Let $\boldsymbol{B} \in \mathbb{R}^{m\times n}$ be a (column-oriented) basis , and let $\boldsymbol{c} \in \mathbb{R}^m$ be a center, let $\sigma > 0$ be the Gaussian width and let $q \in \mathbb{Z}_{>0}$ be the periodization parameter. Then we define the periodized Gaussian $\xi_{\sigma,q}(\boldsymbol{B}, \boldsymbol{z})$ for $\boldsymbol{z} \in (\mathbb{Z}/2^Q\mathbb{Z})^n$ by the following rule.*

$$\xi_{\sigma,\boldsymbol{c},q}(\boldsymbol{B}, \boldsymbol{z}) = \left(\sum_{\boldsymbol{x}\in q\mathbb{Z}^n} \rho_{\sigma,\boldsymbol{c}}(\boldsymbol{B}(\boldsymbol{z}+\boldsymbol{x}))^2\right)^{1/2} \tag{25}$$

Lemma L.1. *Let $\boldsymbol{B} = (\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n) \in \mathbb{R}^{m \times n}$ be a (column-oriented) basis and let $\boldsymbol{B}^* = (\boldsymbol{b}_1^*, \ldots, \boldsymbol{b}_n^*)$ its (non-normalized) Gram-Schmidt orthogonalization, let $\sigma > 0$, let $\boldsymbol{c} \in \mathbb{R}^m$ and $q \in \mathbb{Z}_{>0}$.*

Put $\sigma' = \sigma / \|\boldsymbol{b}_n^*\|$ and $c' = \frac{\langle \boldsymbol{c}, \boldsymbol{b}_n^* \rangle}{\langle \boldsymbol{b}_n^*, \boldsymbol{b}_n^* \rangle}$. Then, for any $z \in \mathbb{Z}/q\mathbb{Z}$ and $\boldsymbol{z}_\circ \in (\mathbb{Z}/q\mathbb{Z})^{n-1}$, we have

$$\xi_{\sigma,\boldsymbol{c},q}(\boldsymbol{B}, \boldsymbol{z}) = \xi_{\sigma',c',q}(1, z) \cdot \xi_{\sigma,\boldsymbol{c}_z,q}(\boldsymbol{B}_\circ, \boldsymbol{z}_\circ), \tag{26}$$

where $\boldsymbol{c}_z = \pi_{n-1}(\boldsymbol{c} - z\boldsymbol{b}_n) \in \mathrm{span}(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_{n-1})$, and π_{n-1} is the orthogonal projection to the first $n-1$ basis vectors of $\boldsymbol{B}^*$, and $\boldsymbol{B}_\circ = (\boldsymbol{b}_1, \ldots, \boldsymbol{b}_{n-1})$.

Proof. We prove the identity Equation (26) by squaring both sides (which is possible by positivity). This yields

$$\xi_{\sigma',c',q}(1, z) \cdot \xi_{\sigma,\boldsymbol{c}_z,q}(\boldsymbol{B}_\circ, \boldsymbol{z}_\circ) = \sum_{x \in q\mathbb{Z}} \sum_{\boldsymbol{x}_\circ \in q\mathbb{Z}^{n-1}} \rho_{\sigma',c'}(z + x)^2 \rho_{\sigma,\boldsymbol{c}_z}(\boldsymbol{B}_\circ(\boldsymbol{z}_\circ + \boldsymbol{x}_\circ))^2 \tag{27}$$

$$= \sum_{\boldsymbol{x} \in q\mathbb{Z}^n} \rho_{\sigma,\boldsymbol{c}}(\boldsymbol{B}(\boldsymbol{z} + \boldsymbol{x}))^2 = \xi_{\sigma,\boldsymbol{c},q}(\boldsymbol{B}, \boldsymbol{z})^2, \tag{28}$$

where $\boldsymbol{z} = (\boldsymbol{z}_\circ, z) \in \mathbb{Z}^n$ and $\boldsymbol{x} = (\boldsymbol{x}_\circ, x) \in \mathbb{Z}^n$. The second equality is not trivial and needs its own computation. Writing $\boldsymbol{t} = (\boldsymbol{t}_1, \ldots, \boldsymbol{t}_n)$ and $t = \boldsymbol{t}_n$, and recalling that $\sigma' = \sigma / \|\boldsymbol{b}_n^*\|$, we have

$$\rho_{\sigma',c'}(t) \rho_{\sigma,\boldsymbol{c}_t}(\boldsymbol{B}_\circ \boldsymbol{t}_\circ) = \exp\left(-\pi \|\boldsymbol{b}_n^*\|^2 (t - c')^2 / \sigma^2 - \pi \| \sum_{i=1}^{n-1} \boldsymbol{t}_i \boldsymbol{b}_i - \boldsymbol{c}_t \|^2 / \sigma^2 \right) \tag{29}$$

Using the Pythagorean theorem in the shape $\|\boldsymbol{v}\|^2 = \|\pi_{n-1}(\boldsymbol{v})\|^2 + \frac{\langle \boldsymbol{v}, \boldsymbol{b}_n^* \rangle^2}{\langle \boldsymbol{b}_n^*, \boldsymbol{b}_n^* \rangle}$, and using that $\pi_{n-1}(\boldsymbol{b}_j) = \boldsymbol{b}_j$ and $\langle \boldsymbol{b}_n^*, \boldsymbol{b}_j \rangle = 0$ for $j < n$, we deduce

$$\begin{aligned}
\| \sum_{i=1}^{n} \boldsymbol{t}_i \boldsymbol{b}_i - \boldsymbol{c} \|^2 &= \| \sum_{i=1}^{n} \pi_{n-1}(\boldsymbol{t}_i \boldsymbol{b}_i - \boldsymbol{c}) \|^2 + \frac{1}{\langle \boldsymbol{b}_n^*, \boldsymbol{b}_n^* \rangle} \langle \sum_{i=1}^{n} \boldsymbol{t}_i \boldsymbol{b}_i - \boldsymbol{c}, \boldsymbol{b}_n^* \rangle^2 \\
&= \| \sum_{i=1}^{n-1} \boldsymbol{t}_i \boldsymbol{b}_i + \pi_{n-1}(\boldsymbol{t}_n \boldsymbol{b}_n - \boldsymbol{c}) \|^2 + \frac{(\boldsymbol{t}_n \langle \boldsymbol{b}_n, \boldsymbol{b}_n^* \rangle - \langle \boldsymbol{c}, \boldsymbol{b}_n^* \rangle)^2}{\langle \boldsymbol{b}_n^*, \boldsymbol{b}_n^* \rangle} \\
&= \| \sum_{i=1}^{n-1} \boldsymbol{t}_i \boldsymbol{b}_i - \boldsymbol{c}_t \|^2 + \|\boldsymbol{b}_n^*\|^2 (t - c')
\end{aligned}$$

Hence, substituting this into Equation (29), we obtain, with $\boldsymbol{t} = (\boldsymbol{t}_\circ, t)$,

$$\rho_{\sigma',c'}(t) \rho_{\sigma,\boldsymbol{c}_t}(\boldsymbol{B}_\circ \boldsymbol{t}_\circ) = \rho_{\sigma,\boldsymbol{c}}(\boldsymbol{B}\boldsymbol{t}).$$

Substituting $\boldsymbol{t} = \boldsymbol{z} + \boldsymbol{x}$ then yields the second inequality in Equation (28).

Corollary L.1. *Let $\boldsymbol{R} = [\boldsymbol{r}_1, \ldots, \boldsymbol{r}_n] \in \mathbb{Z}^{n\times n}$ be a upper triangular invertible matrix with positive diagonal, a center $\boldsymbol{c} \in \mathbb{Z}^n$, let $\sigma \in \mathbb{Q}_{>0}$ and $q \in \mathbb{Z}_{>0}$.*

Put $\sigma' = \sigma/\boldsymbol{r}_{n,n}$ and $c' = \boldsymbol{c}_n/\boldsymbol{r}_{n,n}$. Then, for any $z \in \mathbb{Z}/q\mathbb{Z}$ and $\boldsymbol{z}_\circ \in (\mathbb{Z}/q\mathbb{Z})^{n-1}$, we have

$$\xi_{\sigma,\boldsymbol{c},q}(\boldsymbol{R}, \boldsymbol{z}) = \xi_{\sigma',c',q}(1, z) \cdot \xi_{\sigma,\boldsymbol{c}_z,q}(\boldsymbol{R}_\circ, \boldsymbol{z}_\circ), \tag{30}$$

where $\boldsymbol{c}_z = \pi_{n-1}(\boldsymbol{c} - z\boldsymbol{b}_n)$ with π_{n-1} projection to the first $n-1$ coordinates, and $\boldsymbol{R}_\circ = (\boldsymbol{r}_1, \ldots, \boldsymbol{r}_{n-1})$.

Proof. This is Lemma L.1 restated for upper-triangular matrices. This uses the fact that for upper-triangular matrices we have that $\boldsymbol{b}_i^*/\|\boldsymbol{b}_i^*\|$ is the ith canonical vector. □

Lemma L.2 ([Boe22, Lemma A.26 and Proposition A.28]). *There exists a quantum $\mathrm{QGauss}_{\mathbb{Z}}^{(q,\varepsilon)}$ algorithm taking as input $|c', (\sigma')^2\rangle$ with $\sigma, c \in \mathbb{Q}$ and an empty register of size $\ln(q)$ outputting a state which is within ε trace-distance from the state*

$$|c', (\sigma')^2\rangle \cdot C_0^{-1} \sum_{z\in\mathbb{Z}/q\mathbb{Z}} \xi_{\sigma',c',q}(1, z)|z\rangle$$

where C_0 is the normalization factor. This quantum algorithm can be implemented using $O(\log_2(q) + \ln(1/\varepsilon))$ qubits and $\tilde{O}\left(\log_2(q) \cdot (\ln(1/\varepsilon))^{3/2}\right)$ quantum gates.

Lemma L.3. *For any $\varepsilon \in (0, 1/2)$, $\eta_\varepsilon(\mathbb{Z}) \le \sqrt{\ln(1/\varepsilon)}$.*

Proof. This is a direct application of [MR07, Lemma 3.3]. □

Lemma L.4. *Let $\boldsymbol{R} = [\boldsymbol{r}_1, \ldots, \boldsymbol{r}_n] \in \mathbb{Z}^{n\times n}$ be a upper triangular invertible matrix with positive diagonal, a center $\boldsymbol{c} \in \mathbb{Z}^n$, let $\sigma \in \mathbb{Q}_{>0}$ and $q \in \mathbb{Z}_{>0}$.*

Assume that $\sigma \ge \sqrt{2} \cdot \sqrt{\ln\left(\frac{4n^3}{\varepsilon^2}\right)} \cdot \max_j \boldsymbol{r}_{j,j}$. Then, Algorithm 6.1, after terminating, computes a state that is ε-close in the trace distance to

$$C^{-1} \sum_{\boldsymbol{z}\in(\mathbb{Z}/q\mathbb{Z})^n} \xi_{\sigma,\boldsymbol{c},q}(\boldsymbol{R}, \boldsymbol{z})|\boldsymbol{z}\rangle, \tag{31}$$

where $C \in \mathbb{R}_{>0}$ satisfies $C^2 = \sum_{\boldsymbol{z}\in\mathbb{Z}^n} \rho_{\sigma,\boldsymbol{c}}(\boldsymbol{R}\boldsymbol{z})^2$.

Proof. In line 3 of Algorithm 6.1, an approximation (signified by the tilde on $\tilde{\xi}$) of the periodized discrete Gaussian state over $\mathbb{Z}$ is computed, within trace distance $\varepsilon/(2n)$. Since the rest of the algorithm is a (trace preserving) quantum operation, we can deduce that at the expense of a final trace distance error $\varepsilon/(2n)$, we may assume that this particular state is exact, i.e., that line 3 computes (note the ξ without tilde)

$$|\boldsymbol{R}\rangle|\boldsymbol{c}\rangle|\sigma\rangle \left(|c', \sigma'\rangle \cdot C_0^{-1} \sum_{z\in\mathbb{Z}/q\mathbb{Z}} \xi_{\sigma',c',q}(1, z)|z\rangle \right) |\boldsymbol{0}\rangle.$$

Then, in line 6, another approximation of the periodized discrete Gaussian (but this time over $\boldsymbol{R}_\circ\mathbb{Z}^{n-1}$) is recursively computed, within trace distance $(n-1)\varepsilon/n$. Abstractly, this state is of the shape $|\phi\rangle = \sum_{z\in\mathbb{Z}/q\mathbb{Z}} a_z|z\rangle|\tilde{\psi}_z\rangle$ as an approximation of $|\tilde{\phi}\rangle = \sum_{z\in\mathbb{Z}/q\mathbb{Z}} a_z|z\rangle|\psi_z\rangle$, where $T(|\tilde{\psi}_z\rangle, |\psi_z\rangle) \leq (n-1)\varepsilon/n$ for all $z \in \mathbb{Z}/q\mathbb{Z}$, per assumption. Using the identity $T(|\phi\rangle, |\phi'\rangle)^2 = 1 - |\langle\phi|\phi'\rangle|^2$ [Wil17, Eq. (9.172),Eq. (9.85)] one then deduces that

$$\begin{aligned}1 - T(|\phi\rangle, |\tilde{\phi}\rangle)^2 &= \sum_{z\in\mathbb{Z}/q\mathbb{Z}} |a_z|^2 |\langle\psi_z|\tilde{\psi}_z\rangle|^2 = \sum_{z\in\mathbb{Z}/q\mathbb{Z}} |a_z|^2 (1 - T(|\psi_z\rangle, |\tilde{\psi}_z\rangle)^2) \\ &= 1 - \sum_{z\in\mathbb{Z}/q\mathbb{Z}} |a_z|^2 T(|\psi_z\rangle, |\tilde{\psi}_z\rangle)^2\end{aligned}$$

Hence, using $\sum_{z\in\mathbb{Z}/q\mathbb{Z}} |a_z|^2 = 1$, we obtain that $T(|\phi\rangle, |\tilde{\phi}\rangle) \leq \frac{(n-1)\varepsilon}{n}$.

Therefore, up to a trace-distance error $\frac{(n-1)\varepsilon}{n} + \frac{\varepsilon}{2n} = \frac{(2n-1)\varepsilon}{2n}$, we may assume that the final state of Algorithm 6.1 equals (note the lack of a tilde)

$$\begin{aligned}|\boldsymbol{R}\rangle|0\rangle|\boldsymbol{c}\rangle|\sigma\rangle|0\rangle C_0^{-1} \sum_{z\in\mathbb{Z}/q\mathbb{Z}} \xi_{\sigma',c',q}(1,z)|z\rangle C_z^{-1} \sum_{\boldsymbol{z}_\circ\in(\mathbb{Z}/q\mathbb{Z})^{n-1}} \xi_{\sigma,\boldsymbol{c}_z,q}(\boldsymbol{R}_\circ, \boldsymbol{z}_\circ)|\boldsymbol{z}_\circ\rangle \\ = |\boldsymbol{R}\rangle|0\rangle|\boldsymbol{c}\rangle|\sigma\rangle|0\rangle C_0^{-1} \sum_{\boldsymbol{z}\in(\mathbb{Z}/q\mathbb{Z})^n} C_{\boldsymbol{z}_n}^{-1} \cdot \xi_{\sigma,\boldsymbol{c},q}(\boldsymbol{R},\boldsymbol{z})|\boldsymbol{z}\rangle\end{aligned}$$

where the identity comes from Corollary L.1. To show that this is close to the desired state in Equation (31), we compute the trace distance. Writing

$$|\tilde{\phi}\rangle = C_0^{-1} \sum_{\boldsymbol{z}\in(\mathbb{Z}/q\mathbb{Z})^n} C_{\boldsymbol{z}_n}^{-1} \cdot \xi_{\sigma,\boldsymbol{c},q}(\boldsymbol{R},\boldsymbol{z})|\boldsymbol{z}\rangle \text{ and } |\phi\rangle = C^{-1} \sum_{\boldsymbol{z}\in(\mathbb{Z}/q\mathbb{Z})^n} \xi_{\sigma,\boldsymbol{c},q}(\boldsymbol{R},\boldsymbol{z})|\boldsymbol{z}\rangle,$$

we have, using $C^2 = \sum_{\boldsymbol{z}\in(\mathbb{Z}/q\mathbb{Z})^n} \xi_{\sigma,\boldsymbol{c},q}(\boldsymbol{R},\boldsymbol{z})^2$ and the identity $T(|\phi\rangle, |\phi'\rangle)^2 = 1 - |\langle\phi|\phi'\rangle|^2$,

$$\begin{aligned}\sqrt{1 - T(|\phi\rangle, |\tilde{\phi}\rangle)^2} &= \sum_{\boldsymbol{z}\in(\mathbb{Z}/q\mathbb{Z})^n} (C \cdot C_0 \cdot C_{\boldsymbol{z}_n})^{-1} \cdot \xi_{\sigma,\boldsymbol{c},q}(\boldsymbol{R},\boldsymbol{z})^2 &(32)\\ &= 1 - \sum_{\boldsymbol{z}\in(\mathbb{Z}/q\mathbb{Z})^n} (C^{-2} - (C \cdot C_0 \cdot C_{\boldsymbol{z}_n})^{-1}) \cdot \xi_{\sigma,\boldsymbol{c},q}(\boldsymbol{R},\boldsymbol{z})^2 &(33)\\ &\geq 1 - \max_{z\in\mathbb{Z}/q\mathbb{Z}} \left(1 - \frac{C}{C_0 \cdot C_z}\right) &(34)\end{aligned}$$

where the inequality is Hölder's inequality. So it remains to estimate the fraction $\frac{C}{C_0 \cdot C_z}$. By induction, one obtains that, for each $z \in \mathbb{Z}/q\mathbb{Z}$, there exists $c_j^{(z)} \in \mathbb{R}$ for $j \in \{1, \ldots, n\}$ such that

$$C_0^2 \cdot C_z^2 = \prod_{j=1}^{n} \left(\sum_{t\in\mathbb{Z}} \rho_{\frac{\sigma}{r_{j,j}}, c_j^{(z)}}(t)^2 \right)$$

Likewise, there exists $c'_j \in \mathbb{R}$ for $j \in \{1, \ldots, n\}$ such that

$$C^2 = \prod_{j=1}^{n} \left(\sum_{t \in \mathbb{Z}} \rho_{\frac{\sigma}{\boldsymbol{r}_{j,j}}, c'_j}(t)^2 \right).$$

By the assumption that $\sigma > \sqrt{2}\eta_{\varepsilon'}(\mathbb{Z}) \cdot \max_j \boldsymbol{r}_{j,j}$ with $\varepsilon' = \varepsilon^2/(4n^3)$, we can use smoothing arguments [MR07] similar as in the work of Gentry, Peikert and Vaikuntanathan [GPV08] to obtain

$$\begin{aligned}\sum_{t \in \mathbb{Z}} \rho_{\frac{\sigma}{\boldsymbol{r}_{j,j}}, c'_j}(t)^2 &= \sum_{t \in \mathbb{Z}} \rho_{\frac{\sigma}{\sqrt{2}\boldsymbol{r}_{j,j}}, c'_j}(t) \in [1-\varepsilon', 1+\varepsilon'] \sum_{t \in \mathbb{Z}} \rho_{\frac{\sigma}{\sqrt{2}\boldsymbol{r}_{j,j}}, c_j^{(z)}}(t) \\ &\in [1-\varepsilon', 1+\varepsilon'] \sum_{t \in \mathbb{Z}} \rho_{\frac{\sigma}{\boldsymbol{r}_{j,j}}, c_j^{(z)}}(t)^2\end{aligned}$$

Therefore, for all $z \in \mathbb{Z}/q\mathbb{Z}$, we have $C^2 \in [(1-\varepsilon')^n, (1+\varepsilon')^n] C_0^2 C_z^2$ leading to

$$\max_{z \in \mathbb{Z}/q\mathbb{Z}} \left(1 - \frac{C}{C_0 \cdot C_z} \right) \leq 1 - (1-\varepsilon')^{n/2}.$$

Plugging into Equation (34), we obtain

$$T(|\phi\rangle, |\tilde{\phi}\rangle)^2 \leq 1 - (1-\varepsilon')^n \leq n\varepsilon',$$

where we use that $\varepsilon' < 1/n$. Hence $T(|\phi\rangle, |\tilde{\phi}\rangle) \leq \sqrt{n\varepsilon'} \leq \varepsilon/(2n)$, by definition of ε'.

Therefore, the trace distance between the computed state in line 8 of Algorithm 6.1 and the desired state as in Equation (31) is at most $\frac{\varepsilon}{2n} + \frac{(n-1)\varepsilon}{n} + \frac{\varepsilon}{2n} = \varepsilon$. □

Lemma L.5. *Let $\boldsymbol{R} = (\boldsymbol{r}_1, \ldots, \boldsymbol{r}_n) \in \mathbb{Z}^{n \times n}$ be an upper triangular invertible matrix, let $\sigma \in \mathbb{Q}_{>0}$, let $\boldsymbol{c} \in \mathbb{Z}^m$ and $q \in \mathbb{Z}_{>0}$ a periodization parameter which is a power of 2. Let $\varepsilon \in (0, 2^{-n}) > 0$, Then on input $|\boldsymbol{R}, \sigma, \boldsymbol{c}\rangle$, Algorithm 6.1 uses*

$$\tilde{O}\left(n^2 \cdot \beta^{1+o(1)} + n \cdot \log_2(q) \cdot (\ln(1/\varepsilon))^{3/2} \right)$$

quantum gates and

$$O\left(n \cdot \beta^{1+o(1)} + \ln(1/\varepsilon) \right)$$

ancillary qubits, where

$$\beta = \ln(n \cdot q \cdot \|\boldsymbol{R}\|) + \max_i(\mathit{size}(\boldsymbol{c}_i)) + \mathit{size}(\sigma).$$

Proof. The algorithm only works on rationals, whose denominator is at most the denominator of σ times $\|\boldsymbol{R}\|$. The size of c' is then bounded by $\ln(\|\boldsymbol{R}\|) + \max_i \mathrm{size}(\boldsymbol{c}_i)$ (we consider the recursive calls). By the same argument, the size

of $(\sigma')^2$ is bounded by $\mathrm{size}(\sigma)+\ln(\|\boldsymbol{R}\|)$. This implies that the size of the (c', σ'^2) register is bounded by $\beta_1 = O(\ln(\|\boldsymbol{R}\|) + \max_i(\mathrm{size}(\boldsymbol{c}_i)) + \mathrm{size}(\sigma))$.

Over the course of the recursive calls, the quantum register associated to $\boldsymbol{c}$ is going to contain vectors of the form $\boldsymbol{c} + \boldsymbol{R} \cdot \boldsymbol{z}$, with $\boldsymbol{z} \in \mathbb{Z}^n$ and $\|\boldsymbol{z}\|_\infty \leq q$. Let $\beta_2 = \lceil \ln(n \cdot q \cdot \|\boldsymbol{R}\|) \rceil$, the norm of the vector contained in the register is then bounded by 2^{β_2}, the size of the quantum register associated to $\boldsymbol{c}$ is taken to be $n \cdot \beta_2$.

In line 2, the size of the elements implies that the computation can be used using using $O(\beta_1^{1+o(1)})$ qubits and quantum gates.

In line 3, $\mathrm{QGauss}_{\mathbb{Z}}^{(q,\varepsilon/(2n))}$ is used, which cost $O(\log_2(q) + \ln(1/\varepsilon))$ quantum memory and $\tilde{O}\left(\log_2(q) \cdot (\ln(1/\varepsilon))^{3/2}\right)$ quantum gates (note that we removed the dependence in $O(\ln(n))$ since ε is assumed to be $< 2^{-n}$).

In line 4 an uncomputation of line 2 is done, hence takes the same number of qubits and quantum gates.

In line 5, a shift of the center $\boldsymbol{c}$ is computed. This operation is performed with n multiplications and n additions of number of size β_2, resulting in $O(n \cdot \beta_2^{1+o(1)})$ quantum gates and memory.

In line 6, the algorithm is called recursively, with lower dimension and a slightly lower error parameter.

In line 7 an uncomputation of line 5 is done. This takes the same number of qubits and quantum gates.

Let $G(n, \varepsilon)$ (resp $M(n, \varepsilon)$) be the number of quantum gates (resp the number of ancillary qubits) needed to perform Algorithm 6.1 in dimension n with error ε, we proved that

$$G(n, \varepsilon) = 2n\beta_2^{1+o(1)} + 2\beta_1^{1+o(1)} + \tilde{O}\left(\log_2(q) \cdot (\ln(1/\varepsilon))^{3/2}\right) + G(n, (n-1)\varepsilon/n)$$

and that the number of ancillary qubits is

$$M(n, \varepsilon) = \max\left(O(\beta_1^{1+o(1)}), O(n \cdot \beta_2^{1+o(1)}), O(\log_2(q) + \ln(1/\varepsilon)), M(n-1, (n-1) \cdot \varepsilon/n)\right)$$

By induction we have that

$$G(n, \varepsilon) = \tilde{O}\left(n^2 \cdot \beta^{1+o(1)} + n \cdot \log_2(q) \cdot (\ln(1/\varepsilon))^{3/2}\right)$$

and

$$M(n, \varepsilon) = O\left(n \cdot \beta^{1+o(1)} + \ln(1/\varepsilon)\right)$$

which concludes the proof. □

Lemma L.6. *Let $x \geq \sqrt{d/(2\pi)}$, and assume that $q \geq \left\|\boldsymbol{B}^{-1}\right\| \cdot \sigma \cdot x/\sqrt{2}$, then the following two states*

$$C^{-1} \sum_{\mathbf{z} \in (\mathbb{Z}/q\mathbb{Z})^n} \xi_{\sigma,q}(\boldsymbol{B}, \boldsymbol{z})|\boldsymbol{z}\rangle,$$

and

$$D^{-1}\sum_{\boldsymbol{z}\in[q]^n}\rho_\sigma(\boldsymbol{B}\boldsymbol{z})|\boldsymbol{z}\rangle$$

are $\beta_n(x)$-close in trace distance. Here $C, D \in \mathbb{R}_{>0}$ satisfy $C^2 = \sum_{\boldsymbol{z}\in\mathbb{Z}^n}\rho_\sigma(\boldsymbol{B}\boldsymbol{z})^2$ and $D^2 = \sum_{\boldsymbol{z}\in[q]_c^n}\rho_\sigma(\boldsymbol{B}\boldsymbol{z})^2$.

Proof. By inequalities regarding the trace distance and a simple calculation[8], we have that the trace distance satisfies $D(|\phi\rangle, |\psi\rangle) \le \||\phi\rangle - |\psi\rangle\|$. Hence, we will bound the square of the 2-norm of the difference of the two states of this lemma.

$$\left\| C^{-1}\sum_{\boldsymbol{z}\in[q]_c^n}\xi_{\sigma,q}(\boldsymbol{B},\boldsymbol{z})|\boldsymbol{z}\rangle - D^{-1}\sum_{\boldsymbol{z}\in[q]_c^n}\rho_\sigma(\boldsymbol{B}\boldsymbol{z})|\boldsymbol{z}\rangle\right\|^2 \tag{35}$$

$$= \sum_{\boldsymbol{z}\in[q]_c^n}|C^{-1}\xi_{\sigma,q}(\boldsymbol{B},\boldsymbol{z}) - D^{-1}\rho_\sigma(\boldsymbol{B}\boldsymbol{z})|^2 \tag{36}$$

$$\le \sum_{\boldsymbol{z}\in[q]_c^n}\left(C^{-2}\sum_{\boldsymbol{x}\in q\mathbb{Z}^n\setminus\{0\}}\rho_\sigma(\boldsymbol{B}(\boldsymbol{z}+\boldsymbol{x}))^2 + |(C^{-1}-D^{-1})\rho_\sigma(\boldsymbol{B}\boldsymbol{z})|^2\right). \tag{37}$$

where the last inequality is due to the reverse triangle inequality (for 2-norms), by seeing the function $\boldsymbol{x}\mapsto\rho_\sigma(\boldsymbol{B}(\boldsymbol{z}+\boldsymbol{x}))$ as a vector in $\mathbb{R}^{q\mathbb{Z}^n}$, as well as $\boldsymbol{x}\mapsto 1_{\boldsymbol{x}=0}\cdot\rho_\sigma(\boldsymbol{B}\boldsymbol{z})$.

Equation (37) can be further simplified

$$\begin{aligned}&C^{-2}\sum_{\boldsymbol{z}\in\mathbb{Z}^n\setminus[q]_c^n}\rho_\sigma(\boldsymbol{B}\boldsymbol{z})^2 + |C^{-1}-D^{-1}|^2\sum_{\boldsymbol{z}\in[q]_c^n}\rho_\sigma(\boldsymbol{B}\boldsymbol{z})^2\\ =&C^{-2}(C^2-D^2) + |D/C-1|^2 = |1-(D/C)^2| + |1-(D/C)|^2\end{aligned}$$

Writing $D^2/C^2 = 1-\eta/2$ for some $\eta\in[0,1]$, we obtain that above is bounded by $\eta + |1-\sqrt{1-\eta}|^2 \le \eta$. Hence we will estimate such an η.

We have, by Banaszczyk's bound (see Lemma A.2),

$$D^2/C^2 - 1 = \frac{D^2-C^2}{C^2} = C^{-2}\sum_{\boldsymbol{z}\in\mathbb{Z}^n\setminus[q]_c^n}\rho_{\frac{\sigma}{\sqrt{2}}}(\boldsymbol{B}\boldsymbol{z}) \le \beta_n(R\sqrt{2}/\sigma),$$

where $R = \min\{\|\boldsymbol{B}\boldsymbol{z}\| \mid \boldsymbol{z}\in\mathbb{Z}^n\setminus[q]^n\}$. We have that $R \ge q/\left\|\boldsymbol{B}^{-1}\right\|$, so by hypothesis of q, $R\sqrt{2}/\sigma \ge x$ and then $\beta_n(R\sqrt{2}/\sigma)\le\beta_n(x)$. □

[8] For pure states, we have $D(|\psi\rangle,|\phi\rangle) = \sqrt{1-|\langle\psi|\phi\rangle|^2} \le \||\psi\rangle - |\phi\rangle\|$ [Wil17, Eq. (9.172),Eq. (9.85)]. Writing $\langle\psi|\phi\rangle = a+bi$ with $a^2+b^2 = |a+bi|^2 \le 1$ (due to Hölders inequality), the last inequality follows from $1-|\langle\psi|\phi\rangle|^2 \le 1-a^2-b^2 \le 1-a^2 \le 2-2a = \langle\psi|\psi\rangle + \langle\phi|\phi\rangle - \langle\psi|\phi\rangle - \langle\phi|\psi\rangle = \||\psi\rangle - |\phi\rangle\|^2$.

Lemma L.7. *Let $\varepsilon \in (0, 2^{-n})$. Assume that q, R satisfies that $q > R \cdot \left\|\boldsymbol{B}^{-1}\right\|$, $\|\boldsymbol{B}\| \cdot q \geq \sigma \cdot \sqrt{\ln(2/\varepsilon)}$ and $R \geq \sigma \cdot \sqrt{\ln(2/\varepsilon)}$ then the following two states*

$$A^{-1} \sum_{\mathbf{x} \in [q]^n} \rho_\sigma(\boldsymbol{B}\boldsymbol{x})|\boldsymbol{z}\rangle$$

and

$$B^{-1} \sum_{\substack{\boldsymbol{x} \in \mathbb{Z}^n \\ \|\boldsymbol{B}\boldsymbol{x}\| \leq R}} \rho_\sigma(\boldsymbol{B}\boldsymbol{x})|\boldsymbol{x}\rangle \tag{38}$$

are ε-close in trace distance. Here $A, B \in \mathbb{R}_{>0}$ are chosen such that the two above state have norm 1.

Proof. Let $x = \sqrt{2\ln(2/\varepsilon)} \geq \sqrt{n}$. Note that we have $A = \sqrt{\rho_{\sigma/\sqrt{2}}(\boldsymbol{B} \cdot [q]^n)}$ and $B = \sqrt{\rho_{\sigma/\sqrt{2}}(L(\boldsymbol{B}) \cap B(0, R))}$. By assumption on q, we have that for any $\boldsymbol{x} \in \mathbb{Z}^n \backslash [q]^n$, $\|\boldsymbol{B} \cdot \boldsymbol{x}\| \geq q/\left\|B^{-1}\right\| > R$. In particular, $\{\boldsymbol{x} \in \mathbb{Z}^n, \ \ \|\boldsymbol{B}\boldsymbol{x}\| \leq R\} \subseteq [q]^n$ and $A \geq B$. Let us bound the norm of the Gaussian superposition over all $\boldsymbol{x} \in [q]^n$ satisfying that $\|\boldsymbol{B}\boldsymbol{x}\| > R$. We have that

$$\begin{aligned}
\left\| A^{-1} \sum_{\substack{\mathbf{x} \in [q]^n \\ \|\boldsymbol{B}\boldsymbol{x}\| > R}} \rho_\sigma(\boldsymbol{B}\boldsymbol{x})|\boldsymbol{x}\rangle \right\|^2 =& \frac{\rho_{\sigma/\sqrt{2}}(\mathcal{L}(\boldsymbol{B}) \setminus B(0, R))}{\rho_{\sigma/\sqrt{2}}(\boldsymbol{B} \cdot [q]^n)} \\
\leq& \frac{\rho_{\sigma/\sqrt{2}}(\mathcal{L}(\boldsymbol{B}) \setminus B(0, R))}{\rho_{\sigma/\sqrt{2}}(\mathcal{L}(\boldsymbol{B}) \cap B(0, R))} \\
\leq& \frac{\beta_n(x)}{1 - \beta_n(x)} \qquad \text{By Lemma A.2}
\end{aligned}$$

Now, we bound the norm of the remaining difference.

$$\begin{aligned}
\left\| \sum_{\substack{\mathbf{x} \in [q]^n \\ \|\boldsymbol{B}\boldsymbol{x}\| \leq R}} \rho_\sigma(\boldsymbol{B}\boldsymbol{x})(A^{-1} - B^{-1})|\boldsymbol{x}\rangle \right\|^2 =& (B^{-1} - A^{-1})^2 \cdot \rho_{\sigma/\sqrt{2}}\left(\boldsymbol{B} \cdot [q]^n \cap B(0, R)\right) \\
\leq& (B^{-1} - A^{-1})^2 \cdot A^2 = (\frac{A}{B} - 1)^2.
\end{aligned}$$

We have that

$$\frac{A}{B} = \sqrt{\frac{\rho_{\sigma/\sqrt{2}}(\boldsymbol{B} \cdot [q]^n)}{\rho_{\sigma/\sqrt{2}}(L(\boldsymbol{B}) \cap B(0, R))}} \leq \sqrt{\frac{\rho_{\sigma/\sqrt{2}}(\|\boldsymbol{B}\| \cdot q)}{\rho_{\sigma/\sqrt{2}}(L(\boldsymbol{B}) \cap B(0, R))}} \leq \sqrt{\frac{1 + \beta_d(x)}{1 - \beta_d(x)}},$$

and finally the claimed bound follows from the fact that $\sqrt{(1+z)/(1-z)} - 1 \leq 2z$ if $z \leq 1$. □

Lemma L.8. *Let $\varepsilon_0 > 0$ and $\boldsymbol{B}, \tilde{\boldsymbol{B}} \in \mathbb{R}^{n\times n}$ non-singular satisfying $\|\boldsymbol{B}\tilde{\boldsymbol{B}}^{-1} - \boldsymbol{I}\| < \varepsilon_0$ and $\eta_{1/2}(\mathcal{L}(\boldsymbol{B})), \eta_{1/2}(\mathcal{L}(\tilde{\boldsymbol{B}})) < \sigma/2$. Then*

$$C^{-1} \sum_{\boldsymbol{z}\in(\mathbb{Z}/q\mathbb{Z})^n} \xi_{\sigma,\boldsymbol{c},q}(\boldsymbol{B},\boldsymbol{z})|\boldsymbol{z}\rangle \text{ and } \tilde{C}^{-1} \sum_{\boldsymbol{z}\in(\mathbb{Z}/q\mathbb{Z})^n} \xi_{\sigma,\boldsymbol{c},q}(\tilde{\boldsymbol{B}},\boldsymbol{z})|\boldsymbol{z}\rangle$$

are $4\cdot\sqrt{n}\cdot\sqrt{\varepsilon_0}$-close in trace distance.

Proof. We use a result of Pellet–Mary and Stehlé [PMS21, Lemma 2.3], and the following computation. Writing $\boldsymbol{B}\tilde{\boldsymbol{B}}^{-1} = \boldsymbol{I} + \boldsymbol{E}$ with $\|\boldsymbol{E}\| \le \varepsilon_0$, we have that the square of the two-norm distance between the states equals

$$\begin{aligned}
&\sum_{\boldsymbol{z}\in(\mathbb{Z}/q\mathbb{Z})^n} |\xi_{\sigma,\boldsymbol{c},q}(\boldsymbol{B},\boldsymbol{z}) - \xi_{\sigma,\boldsymbol{c},q}(\tilde{\boldsymbol{B}},\boldsymbol{z})|^2 \\
&\qquad \le \sum_{\boldsymbol{z}\in\mathbb{Z}^n} |\rho_{\sigma,\boldsymbol{c}}(\boldsymbol{B}\boldsymbol{z}) - \rho_{\sigma,\boldsymbol{c}}(\tilde{\boldsymbol{B}}\boldsymbol{z})|^2 \\
&\le \left(\sum_{\boldsymbol{z}\in\mathbb{Z}^n} |\rho_{\sigma,\boldsymbol{c}}(\boldsymbol{B}\boldsymbol{z}) - \rho_{\sigma,\boldsymbol{c}}(\tilde{\boldsymbol{B}}\boldsymbol{z})|\right)^2 \\
&\qquad \le 16n\|\boldsymbol{B}\tilde{\boldsymbol{B}}^{-1} - \boldsymbol{I}\| \le 16n\varepsilon_0.
\end{aligned}$$

here we use [PMS21, Lemma 2.3] with $\boldsymbol{S}_2^{-1} = \sigma^{-1}\boldsymbol{B}$ and $\boldsymbol{S}_1^{-1} = \sigma^{-1}\tilde{\boldsymbol{B}}$. The assumption used by Pellet–Mary and Stehlé, that $\eta_{1/2}(\boldsymbol{S}_1^{-1}\mathbb{Z}^n) < 1/2$ and $\eta_{1/2}(\boldsymbol{S}_2^{-1}\mathbb{Z}^n) < 1/2$ is equivalent to the similar assumptions here (just a scaling by σ).

As earlier in this section, the trace distance of two states can be upper bounded by the 2-norm distance between these two states, which proves the claim. □

Lemma L.9. *Let $\boldsymbol{R}$ be a full rank matrix of dimension n and $\varepsilon > 0$, there exists an absolute polynomial P such that if $\sigma \ge P(n)\cdot\varepsilon^{2/n}\cdot\sqrt{\ln(\varepsilon)}\cdot\det(\boldsymbol{R})^{1/n}$ and $R \ge \sigma\cdot\sqrt{d}$, then the two states*

$$C^{-1} \sum_{\substack{\boldsymbol{z}\in\mathbb{Z}^n\setminus\{0\},\\ \|\boldsymbol{R}\boldsymbol{z}\|\le R}} \rho_\sigma(\boldsymbol{R}\cdot\boldsymbol{z})|\boldsymbol{z}\rangle \quad \text{and} \quad C'^{-1} \sum_{\substack{\boldsymbol{z}\in\mathbb{Z}^n,\\ \|\boldsymbol{R}\boldsymbol{z}\|\le R}} \rho_\sigma(\boldsymbol{R}\cdot\boldsymbol{z})|\boldsymbol{z}\rangle$$

(where C, C' are normalization factors) are within distance ε.

Theorem 6.1. *For any $\varepsilon \in (0, 2^{-n})$, and any non-singular upper triangular matrices $\boldsymbol{R}, \boldsymbol{R}' \in \mathbb{Z}^{n\times n}$ with positive diagonal satisfying $\left\|\boldsymbol{R}'\boldsymbol{R}^{-1} - \boldsymbol{I}\right\| \le \varepsilon^2/(16n) \le 1$, and $\sigma \ge \sqrt{2\cdot\ln(64n^3/\varepsilon^2)}\cdot\|\boldsymbol{R}\|$, then the output of Algorithm 6.1 on input $(\boldsymbol{R}', \sigma, R, q, \boldsymbol{c} = \boldsymbol{0})$ is ε-close to the state*

$$C'^{-1} \sum_{\substack{\boldsymbol{z}\in\mathbb{Z}^n,\\ \|\boldsymbol{R}\boldsymbol{z}\|\le R}} \rho_\sigma(\boldsymbol{R}\cdot\boldsymbol{z})|\boldsymbol{z}\rangle, \tag{7}$$

where $R = \sqrt{\ln(1/\varepsilon) \cdot n} \cdot \sigma$ *and* q *is the smallest power of two such that*

$$q \geq \sqrt{2n \cdot \ln(1/\varepsilon) \cdot \ln(64n^3/\varepsilon^2)} \cdot \|\boldsymbol{R}\| \cdot \mathrm{cond}(\boldsymbol{R}).$$

Moreover, Algorithm 6.1 uses $\tilde{O}\left(n^2 \cdot \beta^{1+o(1)} + n \cdot \log_2(q) \cdot (\ln(1/\varepsilon))^{3/2}\right)$ *quantum gates and* $O\left(n \cdot \beta^{1+o(1)} + \ln(1/\varepsilon)\right)$ *ancillary qubits, where* $\beta = \ln(n \cdot q \cdot \|\boldsymbol{R}\|) + \max_i(\mathit{size}(\boldsymbol{c}_i)) + \mathit{size}(\sigma)$.

Proof. Let $|\phi_0\rangle$ be the output state of Algorithm 6.1 on input $(\boldsymbol{R}', \sigma, R, q, \boldsymbol{0})$, let $|\phi_1\rangle$ be the state described in Eq. (31) with matrix $\boldsymbol{R}'$, let $|\phi_2\rangle$ be the same state but with matrix $\boldsymbol{R}$, let $|\phi_3\rangle$ be the state described in Eq. (38) and finally $|\phi_4\rangle$ the state of Eq. (39). Now note that our choices of σ, q and R satisfy:

- Lemma L.4 implies that $\||\phi_0\rangle - |\phi_1\rangle\| \leq \varepsilon/4$;
- $\||\phi_1\rangle - |\phi_2\rangle\| \leq \varepsilon/4$, by Lemma L.8. We use here the fact that $\eta_{1/2}(\mathcal{L}(\boldsymbol{R})) \leq \|\boldsymbol{R}\|$ and that $\|\boldsymbol{R}'\| \leq 2\|\boldsymbol{R}\|$;
- Lemma L.7 with error term is $\leq \varepsilon/4$, implies that $\||\phi_2\rangle - |\phi_3\rangle\| \leq \varepsilon/4$,

which, in combination with the complexity statements of Lemma L.5, concludes the proof. □

Corollary L.2. *Let* $\varepsilon, \sigma, q, R, \boldsymbol{R}, \boldsymbol{R}'$ *satisfying the hypotheses of Theorem 6.1. Furthermore, assume that* $\sigma \geq 4 \cdot \varepsilon^{-1/n} \cdot \det(\boldsymbol{R})^{1/n}$. *Then the output of Algorithm 6.1 on input* $(\boldsymbol{R}', \sigma, R, q, \boldsymbol{c} = \boldsymbol{0})$ *is* 2ε*-close to the state*

$$C^{-1} \sum_{\substack{\boldsymbol{z} \in \mathbb{Z}^n \setminus \{0\}, \\ \|\boldsymbol{R}\boldsymbol{z}\| \leq R}} \rho_\sigma(\boldsymbol{R} \cdot \boldsymbol{z}) |\boldsymbol{z}\rangle. \tag{39}$$

Proof. The squared trace distance between Eq. (39) and Eq. (7) (recall that $C'^2 = \rho_{\sigma/\sqrt{2}}(\mathcal{L}(\boldsymbol{R})|_R)$ and $C^2 = C'^2 - 1$) is equal to

$$\begin{aligned} &\sum_{\substack{\boldsymbol{z} \in \mathbb{Z}^n \setminus \{0\}, \\ \|\boldsymbol{R}\boldsymbol{z}\| \leq R}} \rho_{\sigma/\sqrt{2}}(\boldsymbol{R} \cdot \boldsymbol{z}) \cdot \left(\frac{1}{C^2} - \frac{1}{C'^2}\right) + \frac{1}{C'^2} \\ =&1 - \frac{C^2}{C'^2} + \frac{1}{C'^2} = 1 - \frac{C'^2 - 1}{C'^2} + \frac{1}{C'^2} = \frac{2}{C'^2}. \end{aligned}$$

Now, by the condition on R, it holds that $\rho_{\sigma/\sqrt{2}}(\mathcal{L}(\boldsymbol{R})|_R) \geq (1-\varepsilon) \cdot (\sigma/\sqrt{2})^n / \det(\boldsymbol{R})$. The condition on σ gives that this is $\geq 1/(2\varepsilon)$, which ends the proof. □

M About the straddle encoding

Definition M.1. *The straddle encoding of parameter* t *is defined as:*

$$\begin{aligned} \mathrm{Str}_t : \mathbb{R} &\longrightarrow \mathcal{S}(\mathbb{C}^{\mathbb{Z}}) \\ x &\longmapsto \cos\left(\frac{\pi}{2}\{x/t\}\right)|\lfloor x/t \rfloor\rangle + \sin\left(\frac{\pi}{2}\{x/t\}\right)|\lfloor x/t \rfloor + 1\rangle \end{aligned}$$

Where $\{x\} = x - \lfloor x \rfloor$. *Note that if we restrict* Str_t *to* $[a, b]$ *for* $a, b \in \mathbb{R}$, *then its codomain becomes* $\mathbb{C}^{[\![\lfloor a/t \rfloor, \lceil b/t \rceil]\!]}$

In order to encode complex arguments in $\mathbb{R}/\mathbb{Z}$, we use a modified version of the straddle encoding.

Definition M.2. *Let $t \in (0,1)$ be the inverse of an integer. The torus-straddle encoding of parameter t is defined as*

$$\begin{aligned}\mathrm{Str}'_t : \mathbb{R}/\mathbb{Z} &\longrightarrow \mathcal{S}(\mathbb{C}^{[\![0,1/t-1]\!]}) \\ x &\longmapsto \cos\Big(\frac{\pi}{2}\{x/t\}\Big)|\lfloor x/t\rfloor\rangle + \sin\Big(\frac{\pi}{2}\{x/t\}\Big)|\lfloor x/t\rfloor + 1 \bmod 1/t\rangle\end{aligned}$$

Lemma M.1 ([EHKS14b, Example 5.3]). *The functions Str_t and Str'_t are $\frac{\pi}{2t}$-Lipschitz, and $2t$-totally separative.*

Lemma M.2 ([EHKS14b, Lemma 5.2.b]). *Let $f_i : X_i \to \mathcal{H}_i$ a a_i-Lipschitz function for $i \in [\![m]\!]$ and let $f : \bigtimes_i X_i \to \bigotimes_i \mathcal{H}_i$ defined by $f((x_i)_i) = \otimes_i f_i(x_i)$. Then f is $\sqrt{\sum_i a_i^2}$-Lipschitz.*

Definition M.3. *Let $t \in (0,1)$ be the inverse of an integer. We define*

$$\begin{aligned}\mathrm{EncArg}_t : \quad \mathrm{Arg}_K &\longrightarrow (\mathbb{C}^2)^{\otimes d_{\mathbb{R}}} \otimes (\mathbb{C}^{[\![0,1/t-1]\!]})^{\otimes d_{\mathbb{C}}} \\ (s_i)_{i\in[\![d_{\mathbb{R}}]\!]} \times (\theta_i)_{i\in[\![d_{\mathbb{C}}]\!]} &\longmapsto \otimes_{i\in[\![d_{\mathbb{R}}]\!]}(|s_i\rangle) \otimes \otimes_{i\in[\![d_{\mathbb{C}}]\!]}(\mathrm{Str}'_t(\theta_i))\end{aligned}$$

where we set $|\pm 1\rangle$ as a basis for $\mathbb{C}^2$. Let $\mathcal{X}_R = \big\{x \in K_{\mathbb{R}}^{\times}|_R,\ |\mathcal{N}(x)| \geq 1\big\}$), we define

$$\begin{aligned}\mathrm{EncLog}_{R,t} : \quad \mathrm{Log}(\mathcal{X}_R) &\longrightarrow (\mathbb{C}^{[\![\lfloor -(d-1)\ln(R)/t\rfloor, \lceil \ln(R)/t\rceil]\!]})^{\otimes d_{\mathbb{R}}+d_{\mathbb{C}}} \\ (x_i)_{i\in[\![d_{\mathbb{R}}+d_{\mathbb{C}}]\!]} &\longmapsto \otimes_{i\in[\![d_{\mathbb{R}}+d_{\mathbb{C}}]\!]} \mathrm{Str}_t(x_i)\end{aligned}$$

Finally, we define

$$\begin{aligned}\mathrm{Enc}_{R,t} : \mathcal{X}_R &\longrightarrow \mathcal{H}_{R,t} \\ x &\longmapsto \mathrm{EncArg}(\arg(x)) \otimes \mathrm{EncLog}(\mathrm{Log}(x))\end{aligned},$$

where

$$\mathcal{H}_{R,t} = (\mathbb{C}^2)^{\otimes d_{\mathbb{R}}} \otimes (\mathbb{C}^{[\![0,1/t-1]\!]})^{\otimes d_{\mathbb{C}}} \otimes (\mathbb{C}^{[\![\lfloor -(d-1)\ln(R)/t\rfloor, \lceil \ln(R)/t\rceil]\!]})^{\otimes d_{\mathbb{R}}+d_{\mathbb{C}}}$$

We have that

$$\begin{aligned}\ln(\dim(\mathcal{H}_{R,t})) =& 2d_{\mathbb{R}} + d_{\mathbb{C}} \cdot \ln(1/t) + (d_{\mathbb{R}} + d_{\mathbb{C}})\ln(d\ln(R)/t) \\ =& O(d \cdot (\ln(d\ln(R)) + \ln(1/t))).\end{aligned} \tag{40}$$

Lemma M.3. *Let $t \in (0,1)$ be the inverse of an integer. The function $\mathrm{Enc}_{R,t}$ is $\sqrt{d} \cdot \frac{\pi}{2t}$-Lipschitz and $2\sqrt{d} \cdot t$-totally separative.*

Proof. This is a direct application of the definition of the distance function over $K_{\mathbb{R}}^{\times}$ and the separativity and Lipschitzianity of Str. □

M.1 Properties of the straddle encoding

In this section we give properties of the straddle encoding. We use the modified version given in the previous subsection.

Lemma M.4. *Let $x, y \in \mathbb{R}$, then*

$$\langle \mathrm{Str}_t(x) | \mathrm{Str}_t(y) \rangle \leq \max(0, 1 - \frac{\pi^2}{12 \cdot t^2} |x - y|^2).$$

Proof. We prove the result for $t = 1$, the general case follows. Note that for any $|x| \leq \pi/2$, it holds that $\cos(x) \leq 1 - x^2/3$. If $|x - y| \geq 2$, this is trivially true. Without loss of generality, we assume that $x \in [0, 1]$ and $y \in [x, 2]$. We distinguish between $y \leq 1$ and $y > 1$. If $y \leq 1$, it holds that

$$\begin{aligned}\langle \mathrm{Str}_1(x) | \mathrm{Str}_1(y) \rangle =& \cos\Big(\frac{\pi}{2}x\Big)\cos\Big(\frac{\pi}{2}y\Big) - \sin\Big(\frac{\pi}{2}x\Big)\sin\Big(\frac{\pi}{2}y\Big) \\ =& \cos\Big(\frac{\pi}{2}(x-y)\Big) \leq 1 - \frac{\pi^2}{12}|x-y|^2\end{aligned}$$

Now, assume that $0 \leq x \leq 1 \leq y \leq 2$, and write $x = 1 - a$, $y = 1 + b$. Note that $y - x = a + b$. It holds that

$$\begin{aligned}\langle \mathrm{Str}_1(x) | \mathrm{Str}_1(y) \rangle =& \sin\Big(\frac{\pi}{2}(1-a)\Big)\cos\Big(\frac{\pi}{2}b\Big) = \cos\Big(\frac{\pi}{2}a\Big)\cos\Big(\frac{\pi}{2}b\Big) \\ =& \frac{1}{2}\left(\cos\Big(\frac{\pi}{2}(a-b)\Big) + \cos\Big(\frac{\pi}{2}(a+b)\Big)\right) \\ \leq& 1 - \frac{\pi^2}{12} \cdot ((a+b)^2 + (a-b)^2) = 1 - \frac{\pi^2}{6}(a^2 + b^2) \\ \leq& 1 - \frac{\pi^2}{12}(a+b)^2.\end{aligned}$$

□

Corollary M.1. *Let $t \in (0, 1)$. Then Str_t is $(\nu', 1-\varepsilon')$-separative over $\mathbb{R}^d$ where $\nu' = 2\sqrt{d} \cdot t/(\sqrt{10} \cdot \pi)$ and $\varepsilon' = 1/30$.*

Proof. Let $\nu' = 2\sqrt{d} \cdot t/(\sqrt{10} \cdot \pi)$. Let $\boldsymbol{x}, \boldsymbol{y}$ such that $\|\boldsymbol{x} - \boldsymbol{y}\| \geq \nu'$, in particular, there exists a component which is greater than $\nu'/\sqrt{d}$, without loss of generality, assume that it is the first. Then it holds that

$$\left\langle \mathrm{Str}_t^{(d)}(\boldsymbol{x}) \middle| \mathrm{Str}_t^{(d)}(\boldsymbol{y}) \right\rangle \leq \langle \mathrm{Str}_t(x_1) | \mathrm{Str}_t(y_1) \rangle \leq 1 - \frac{\pi^2}{12 \cdot t^2} \cdot \frac{\nu'}{d},$$

Hence the result. □