On the hardness of the NTRU problem

Alice Pellet-Mary* and Damien Stehlé**

Abstract. The 25 year-old NTRU problem is an important computational assumption in public-key cryptography. However, from a reduction perspective, its relative hardness compared to other problems on Euclidean lattices is not well-understood. Its decision version reduces to the search Ring-LWE problem, but this only provides a hardness upper bound.

We provide two answers to the long-standing open problem of providing reduction-based evidence of the hardness of the NTRU problem. First, we reduce the worst-case approximate Shortest Vector Problem over ideal lattices to an average-case search variant of the NTRU problem. Second, we reduce another average-case search variant of the NTRU problem to the decision NTRU problem.

1 Introduction

In the NTRU encryption scheme [HPS98], the public key is an element h of a polynomial ring R_q that can be chosen as $\mathbb{Z}_q[x]/\Phi$ for some degree d monic irreducible polynomial Φ and some integer $q \geqslant 2$. This public key h is far from uniform in R_q, as it can be written as $h = f/g \bmod q$ where the secret key polynomials $f, g \in R = \mathbb{Z}[x]/\Phi$ have coefficients with small magnitudes compared to $\sqrt{q}$. In most concrete instantiations, such as the original scheme and the NTRU and NTRU Prime Round-3 candidates to the NIST post-quantum cryptography standardization project [CDH+20, BBC+20], the coefficients of f and g even belong to $\{-1, 0, 1\}$ and q grows as a small degree polynomial in d. As a result, the set of such h's is very sparse in R_q. The tasks of distinguishing h from uniform and recovering a sufficiently short pair (f, g) from h are respectively known as the decision and search variants of the NTRU problem.

The search NTRU problem can be solved with lattice reduction algorithms (such as [Sch87]), but to succeed for the most usual setting of $q \leqslant \mathrm{poly}(d)$, they require a computational effort growing as $\exp(O(d))$. In [KF15], Kirchner and Fouque described a heuristic algorithm with slightly subexponential cost $\exp(O(d/\log\log d))$ for the usual setting of $q \leqslant \mathrm{poly}(d)$ and $\|f\|_\infty, \|g\|_\infty \leqslant$

* CNRS, Univ. Bordeaux, and Inria, alice.pellet-mary@math.u-bordeaux.fr
** ENS de Lyon and Institut Universitaire de France, damien.stehle@ens-lyon.fr
© IACR 2021. This article is the full version of the article published by Springer-Verlag.

$O(1)$. If the magnitude bound grows as $\Omega(\sqrt{d})$, then the cost of this algorithm is $\exp(O(d))$. In the completely different regime of very large q (but with $\|f\|$ and $\|g\|$ growing at a much smaller pace), recent works [ABD16, CJL16, KF17] have shown that the NTRU problem is significantly easier than previously thought. For example, one can recover appropriately distributed f, g with $\|f\|, \|g\| \leqslant \text{poly}(d)$ from h in quantum polynomial time when $q \geqslant \exp(\widetilde{\Omega}(\sqrt{d}))$. Prior to those works, it was believed that $q \geqslant \exp(\widetilde{\Omega}(d))$ was necessary for polynomial cost. This range of modulus q is far from the one used for the NTRU encryption scheme. However, NTRU instances with a large modulus q can occur in more advanced cryptographic constructions such as [LTV12] and [GGH13].

On the lower-bound front, it was shown in [SS11] for Φ a power-of-2 cyclotomic and extended in [WW18] to all cyclotomics that if f, g are Gaussian over R (restricted to elements that are invertible modulo q) with standard deviation that is a little larger than $\sqrt{q}$, then the distribution of $h = f/g \bmod q$ is within $2^{-\Omega(d)}$ statistical distance from the uniform distribution over invertible elements of R_q. This variant of decision NTRU is therefore vacuously hard. This parameter regime is relevant to the NTRU signature algorithm [HHP+03, SS13]. It also allows to obtain an NTRU-like public-key encryption scheme, but less efficient than with smaller secret key polynomials f, g.

Despite 25 years of study, little is known about the relationships between the NTRU problem variants and between them and other well-studied problems over Euclidean lattices. To our knowledge, the only exceptions are the direct reduction from decision NTRU to search NTRU and a reduction from decision NTRU to the search version of the Ring-LWE problem [SSTX09, LPR10], sketched in [Pei16, Se. 4.4.4]. Note that this only provides an *upper bound* to the hardness of the NTRU problem. Given this state of affairs, Peikert asked the following question in [Pei16, Se. 7.1]:

Is there a worst-case hardness reduction, or a search-to-decision reduction, for an NTRU-like problem?

Contributions. We provide positive answers to both components of the above question.

First, we give a reduction from the approximate Shortest Vector Problem restricted to ideal lattices (ideal-SVP) to a worst-case variant of the search NTRU problem. Combining the latter with the recent worst-case to average-case reduction for ideal-SVP from [dBDPW20] leads to a reduction from worst-case ideal-SVP to an average-case version of the search NTRU problem. The instance distribution is inherited from the distribution over ideal lattices considered in [dBDPW20]. We also show that this distribution over NTRU instances h can be efficiently sampled from, together with a corresponding trapdoor (f, g), if one has access to a quantum computer or if the modulus q is sufficiently large: this property allows to sample an NTRU encryption public key along with a corresponding secret key.

Second, we exhibit a reduction from another (average-case) variant of the search NTRU problem (see below) to the decision NTRU problem. The reduc-

tion works for a wide set of distributions for the search NTRU instances, and the decision NTRU instance distribution is directly derived from the considered search NTRU distribution. A sufficient condition on the search NTRU distribution is that it produces with overwhelming probability an h with trapdoor (f, g) such that f and g have balanced coefficients (in canonical embedding) and f or g is coprime to q. This covers in particular the standard ternary distribution for f and g (i.e., $f, g \leftarrow \mathcal{U}(\{-1, 0, 1\}^d)$) provided we reject (f, g) when they are not balanced enough or not coprime to q (heuristically, this should happen with probability $\leqslant 1/2$). On the other hand, the choice of the decision NTRU distribution is much less flexible: even if we start with a ternary distribution for the search NTRU instances, it is very unlikely that the decision NTRU distribution we obtain is ternary. Similarly to the first reduction, we show that if the samples h from the search NTRU distribution can be efficiently sampled along with a corresponding trapdoor (f, g), then so can the samples from the resulting decision NTRU instance.

Technical overview. For the sake of simplicity, in the forthcoming discussion, we restrict ourselves to power-of-2 cyclotomic defining polynomials, i.e., $\Phi = x^d + 1$ for d a power of 2. In this case, the ring $R = \mathbb{Z}[x]/(x^d + 1)$ matches the ring of integers of the degree-d cyclotomic number field. Moreover, the coefficient embedding (which is the one usually considered in the NTRU literature) and the canonical embedding (used in this article) define the same geometry, up to scaling and rotation. (In the core of the paper, the results are presented for arbitrary number fields.)

To state the above contributions formally, we consider several variants of the NTRU problem. We say that $h \in R_q = \mathbb{Z}_q[x]/(x^d + 1)$ is an NTRU instance with gap γ if there exists $(f, g) \in R^2 \backslash \{(0, 0)\}$ such that $g \cdot h = f \bmod q$ and $\|f\|, \|g\| \leqslant \sqrt{q}/\gamma$. Note that writing $g \cdot h = f \bmod q$ rather than the more standard $h = f/g \bmod q$ allows one to consider g's that are not invertible modulo q and suffices for cryptographic applications. The norm $\|f\|$ is the Euclidean norm of the vector made of the coefficients of f, and the comparison to $\sqrt{q}$ is justified by the fact that for a uniformly chosen h, one expects the smallest such pair (f, g) to have Euclidean norm around $\sqrt{q}$, up to a small polynomial in d (in the core of the paper, we consider the Euclidean norm induced by the canonical embedding, which leads to a slightly different definition, differing by another $\sqrt{d}$ factor). In the literature, the bound on $\|f\|, \|g\|$ is often absolute rather than relative to $\sqrt{q}$: our definition variant stresses the distance to the uniform h regime. For a distribution $\mathcal{D}$ over NTRU instances with gap γ, the decision problem $(\mathcal{D}, \gamma, q)$-dNTRU consists in distinguishing between $\mathcal{D}$ and the uniform distribution over R_q. On the search NTRU side, the situation is more complex. We consider two variants of search NTRU, both of which with a worst-case and an average-case version. For $\gamma \geqslant \gamma'$, the worst-case vector NTRU problem $\text{wcNTRU}_{\text{vec}}$ consists, given as input an NTRU instance h with gap γ, in recovering $(f, g) \neq (0, 0)$ such that $g \cdot h = f \bmod q$ and $\|f\|, \|g\| \leqslant \sqrt{q}/\gamma'$. Note that if $h \in R_q$ has a trapdoor (f, g), then $(t \cdot f, t \cdot g)$ is another NTRU trapdoor of a possibly larger Euclidean norm, for any non-zero $t \in R$. The

$\text{wcNTRU}_{\text{vec}}$ definition allows solutions whose norms are within an approximation factor γ/γ' from the norms of the promise. Even though there may be plenty of solutions of the form $(t \cdot f, t \cdot g)$ for $t \in R$, the pair ratio $h_{\mathbb{R}} = (tf)/(tg) = f/g$ over $K := \mathbb{Q}[x]/(x^d + 1)$ is an invariant. This motivates the definition of the worst-case module NTRU problem $\text{wcNTRU}_{\text{mod}}$, which consists in recovering $h_{\mathbb{R}}$ from h. This is equivalent to recovering the rank-1 submodule $(f, g)^T \cdot K \cap M_h$ of the rank-2 R-module $M_h = \{(f', g')^T \in R^2 : g' \cdot h = f' \bmod q\}$, hence justifying the name. The average-case counterparts to $\text{wcNTRU}_{\text{vec}}$ and $\text{wcNTRU}_{\text{mod}}$ are defined analogously.

We now sketch the reduction from ideal-SVP to $\text{wcNTRU}_{\text{vec}}$. Let us consider the worst-case variants, and the restriction of ideal-SVP to principal ideals with a known generator: we are given as input a generator z of a principal ideal $I = \langle z \rangle$ of R, and want to use a $\text{wcNTRU}_{\text{vec}}$ oracle to find a short non-zero vector in I. Any element $g \in I$ is of the form $g = z \cdot r$ for some $r \in R$. Consider a short non-zero $g \in I$. Multiplying it by q/z, we obtain that $g \cdot (q/z) = 0 \bmod q$. This already looks like an NTRU equation with a candidate q/z for h. But note that q/z is in $K = \mathbb{Q}[x]/(x^d + 1)$ and has no a priori reason to belong to $R = \mathbb{Z}[x]/(x^d + 1)$, whereas the element h of an NTRU instance must belong to R. To handle this difficulty, we can round q/z to R (coefficient-wise). This leads to $g \cdot \lfloor q/z \rceil = -g \cdot \{q/z\} \bmod q$, where both g and $f := -g \cdot \{q/z\}$ are small elements of R. We obtain the existence of a small pair $(f, g) \in R^2 \backslash \{(0, 0)\}$ such that $g \cdot \lfloor q/z \rceil = f \bmod q$. We can then provide the element $h := \lfloor q/z \rceil$ to the $\text{wcNTRU}_{\text{vec}}$ oracle. The latter returns a pair $(f', g') \in R^2 \backslash \{(0, 0)\}$ such that $g' \cdot \lfloor q/z \rceil = f' \bmod q$, and it can be proved that for any such sufficiently short pair, we have that g' is a short non-zero element of I. To handle possibly non-principal ideals (and also principal ideals with unknown generator), we rely on the 2-element representation of ideals.

If we forget polynomial factors and rely on a $\text{wcNTRU}_{\text{vec}}$ oracle with parameters q, γ and γ', the above allows to find γ_{svp} approximations to a shortest non-zero vector of an arbitrary ideal of volume $\leqslant N$ for $N^{1/d} \approx \sqrt{q}/\gamma$ and $\gamma_{\text{svp}} \approx \gamma/\gamma'$. Note that the reduction is worst-case to worst-case and handles bounded-volume ideals. To handle both limitations, we rely on the recent worst-case to average-case reduction for ideal-SVP from de Boer *et al* [dBDPW20]. By using the reduction with ideals from the average-case distribution from [dBDPW20], we obtain a reduction from worst-case ideal-SVP to average-case NTRU_{vec}. Further, the ideals from the average-case distribution from [dBDPW20] have volumes bounded as $\exp(O(d^2))$. This leads to q of the order of $\exp(O(d))$, which is significantly larger than in many applications. We refine the analysis of [dBDPW20] to show that by allowing the worst-case to average-case ideal-SVP reduction to run in time higher than polynomial in d, the average-case ideals from [dBDPW20] can be chosen with smaller volumes. The resulting NTRU modulus q is still slightly larger than polynomial, but it can be chosen as small as $d^{\omega(1)}$ if one considers sub-exponential time reductions.

We now provide an overview of our second main result, which is a reduction from average-case NTRU_{mod} to dNTRU. This one is applicable for q larger

than some moderate poly(d). At the core of the reduction is an NTRU rerandomization process. Assume we are given some $h \in R_q$ for which there exists a short pair $(f, g) \neq (0, 0)$ with $g \cdot h = f \bmod q$. Now, for any $x_1, x_2 \in R$, we have $g \cdot (x_1 h + x_2) = x_1 f + x_2 g \bmod q$, which may be rewritten as $g \cdot h' = f' \bmod q$ with $h' = x_1 h + x_2$ and $f' = x_1 f + x_2 g$. Further, if x_1 and x_2 are short, then so is f'. This hence gives a way to produce arbitrarily many NTRU samples with a common denominator g, from a single one. Our aim is to query the dNTRU oracle on many such samples, and gather relevant information to solve $\mathrm{NTRU}_{\mathrm{mod}}$. Concretely, we define the dNTRU distribution and show how to tweak the rerandomization process to be able to use the Oracle Hidden Center Problem (OHCP) framework from [PRS17]. At a high level, in the OHCP framework, one is given access to a decision oracle whose acceptance probabilities on a family of distributions $(\mathcal{D}_z)_{z \in \mathbb{C}}$ is a function of the distance $|z - c|$ to a hidden center $c \in \mathbb{C}$. Under some conditions on the oracle behaviour, there exists an efficient algorithm that recovers an arbitrarily accurate approximation $\widetilde{c}$ to c, by querying the OHCP oracle on samples from $\mathcal{D}_z$ for well-chosen values of z. Prior to this work, the OHCP framework has been used to provide a reduction from ideal-SVP to the decision version of Ring-LWE [PRS17], and a search to decision reduction for Ring-LWE [RSW18].

Let us now look more closely at the rerandomization of f. It was shown in [LSS14] that by sampling x_1 and x_2 from spherical Gaussians over R with standard deviation sufficiently above $\max(\|f\|, \|g\|)$, the distribution of $x_1 f + x_2 g$ is Gaussian over the ideal $\langle f \rangle + \langle g \rangle$ with a covariance matrix that is a function of f and g. This spherical Gaussian rerandomization defines our dNTRU distribution. We extend the proof of [LSS14] to show that if instead we sample x_1 and x_2 from correlated non-spherical Gaussians over R, then the distribution of $x_1 f + x_2 g$ is Gaussian over $\langle f \rangle + \langle g \rangle$ with a covariance matrix that can be made to depend solely on $|f(\zeta) - z \cdot g(\zeta)|$ for ζ an arbitrary complex root of $\Phi = x^d + 1$, and $z \in \mathbb{C}$ arbitrary. The center of the OHCP instance is $c = f(\zeta)/g(\zeta) = h_{\mathbb{R}}(\zeta)$ (recall that $h_{\mathbb{R}} = f/g$ belongs to $K = \mathbb{Q}[x]/(x^d + 1)$). Using the dNTRU oracle within the OHCP framework hence allows us to recover an approximation to $h_{\mathbb{R}}(\zeta)$. In the applications from [PRS17, RSW18] of the OHCP framework, one recovers a vector $\mathbf{c}$ of OHCP centers from an approximation $\widetilde{\mathbf{c}}$ by observing that $\mathbf{c}$ belongs to a lattice: the exact center $\mathbf{c}$ can hence be obtained by simply rounding a sufficiently precise approximation $\widetilde{\mathbf{c}}$. In our case, we cannot proceed similarly, as $h_{\mathbb{R}}$ has rational coordinates. We instead show that the LLL algorithm [LLL82] can be used in a manner similar to [KLL84] to recover $h_{\mathbb{R}} = f/g \in K$ from a sufficiently precise approximation to $h_{\mathbb{R}}(\zeta)$, given an a priori upper bound to $\max(\|f\|, \|g\|)$.

Discussion. The two reductions put forward in this work provide some evidence towards supporting the conjectured hardness of the search vectorial NTRU problem and the decision NTRU problem. They may give the impression that the hardness of the NTRU problems lies somewhere between the hardness of the ideal-SVP and that of Ring-LWE. This is however neglecting the fact that there are several NTRU problem variants, and it is unclear whether they are compu-

tationally equivalent. In particular, the reductions are incompatible, in that the first one reduces to NTRU_{vec} and the second one from NTRU_{mod}. NTRU_{mod} reduces to NTRU_{vec}, but it is a reduction from NTRU_{vec} to NTRU_{mod} that we would need to obtain a chain of reductions from ideal-SVP to Ring-LWE via the computationally equivalent NTRU problems. Note that if we assume that ideal-SVP is easy, then these problems are computationally equivalent (see Subsection 3.4), but the reduction from ideal-SVP to NTRU_{vec} becomes vacuous. In fact, it seems that NTRU_{vec} and NTRU_{mod} could even be of different natures: when attempting to solve NTRU_{vec} using an NTRU_{mod} oracle, it is unclear how to make the approximation factor γ/γ' appear, as NTRU_{mod} is only parametrized by the promise gap γ. Better understanding the differences between the NTRU variants seems important to better capture the NTRU hardness. In this direction, note that the known attacks specific to NTRU [ABD16, CJL16, KF17] are mostly relevant for NTRU_{mod}: they can also be used to solve NTRU_{vec}, but the quality of the solution obtained for NTRU_{vec} is the same as the one we would obtain by running the attack to solve NTRU_{mod}, and then running an ideal-SVP solver on the dense rank-1 sub-module to obtain a somehow short vector.

Despite the apparent uncomposability of our two reductions, it would be interesting to have NTRU instance distributions that are compatible with both of them. The second reduction is very permissive with respect to the NTRU_{mod} instance distribution, but the latter still has to satisfy some properties (see Definition 5.1). In particular, the canonical embedding of f and g should be bounded from below and above, and the ideal $\langle f\rangle + \langle g\rangle$ should be coprime with $\langle q\rangle$. We note that in the reduction from ideal-SVP to $\text{wcNTRU}_{\text{vec}}$, the element g is an element of the ideal-SVP instance ideal, which could be chosen Gaussian. Using standard properties of lattice Gaussians, it is not unlikely that one can prove the desired property on its canonical embedding. There seems to be less flexibility in the choice of $f = -g \cdot \{q/z\}$. However, one could replace the deterministic rounding by a Gaussian rounding, to then use a similar approach as the one for g. Concerning the co-primality with $\langle g\rangle$, one could hope to use an inclusion-exclusion argument for Gaussian sums like the one in [SS11].

Concerning the hardness of the NTRU problems relatively to ideal-SVP and Ring-LWE, note that the state of the art suggests that ideal-SVP might be strictly easier than Ring-LWE, as ideal-SVP is known to reduce to Ring-LWE [SSTX09, LPR10, PRS17] but no reduction from Ring-LWE to ideal-SVP is known. In fact, Ring-LWE seems less related to ideal-SVP than to finding two short linearly independent vectors in rank-2 modules over R (SIVP): for an appropriate parametrisation, Ring-LWE reduces to the latter problem [LS15, Se. 5] and, although for some other parametrisation, the latter problem reduces to Ring-LWE (by combining [LS15, Se. 4] and [AD17]). From a lattice perspective, NTRU is a generalization of the unique Shortest Vector Problem to rank-2 modules. At this stage, it is unclear whether its complexity matches the one of ideal-SVP (i.e., SVP for rank-1 modules) or the one of SIVP restricted to rank-2 modules. It could also be strictly in between.

2 Preliminaries

The notations log and ln respectively denote the logarithms in bases 2 and e. For n an integer, we let $[n]$ denote the set $\{1, 2, \ldots, n\}$. Vectors and matrices are denoted with bold lower-case and upper-case letters, respectively. The statistical distance between two distributions D_1 and D_2 with compatible countable supports is defined as $\mathrm{dist}(D_1, D_2) = \frac{1}{2}\sum_x |D_1(x) - D_2(x)|$. We write $D_1 \approx_\varepsilon D_2$ if $\mathrm{dist}(D_1, D_2) \leqslant \varepsilon$ for some $\varepsilon > 0$. If X is a finite set, then we let $U(X)$ denote the uniform distribution over X. If $\mathbf{b}_1, \ldots, \mathbf{b}_n \in \mathbb{R}^m$ are linearly independent vectors, then the notation $(\widetilde{\mathbf{b}}_1, \ldots, \widetilde{\mathbf{b}}_n)$ refers to their Gram-Schmidt orthogonalization. The notation $|||\cdot|||$ refers to the matrix norm induced by the Euclidean norm. Finally, we define $\widetilde{O}(d^t)$ as $O(d^t \operatorname{poly}(\log d))$ for any $t \geqslant 0$ including $t = 0$.

2.1 Euclidean lattices

A lattice $L \subset \mathbb{R}^m$ is a set of the form $L = \mathbf{B} \cdot \mathbb{Z}^{m \times n}$ for some full column-rank matrix $\mathbf{B} \in \mathbb{R}^{m \times n}$ (for some $m \geqslant n \geqslant 1$). The columns of $\mathbf{B}$ are said to form a basis of L. For $i \in [n]$, the ith lattice minimum is defined as $\lambda_i(L) = \min(r : \dim L \cap \mathcal{B}(r) \geqslant i)$, where $\mathcal{B}(r)$ denotes the closed ball of $\mathbb{R}^m$ of radius r. The determinant $\det(L)$ is defined as $\sqrt{\det(\mathbf{B}^T\mathbf{B})}$, which is independent of the particular choice of basis $\mathbf{B}$ of L. Minkowski's (second) theorem states that $\prod_{i\in[n]} \lambda_i(L) \leqslant \sqrt{n}^n \cdot \det(L)$.

In this article, we will be interested in the ideal Hermite Shortest vector problem. We first recall below the definition of the Hermite Shortest Vector Problem (HSVP) for arbitrary lattices, and we will instantiate it for ideal lattices in Section 2.4.

Definition 2.1 (γ-HSVP)**.** *Let $\gamma \geqslant 1$. Given as input a lattice $L \subset \mathbb{Q}^n$ (represented by an arbitrary $\mathbb{Z}$-basis), the* γ-HSVP *problem asks to find a vector $\mathbf{w} \in L\backslash\{\mathbf{0}\}$ such that $\|\mathbf{w}\| \leqslant \gamma \cdot \sqrt{n} \cdot \det(L)^{1/n}$.*

By Minkowski's theorem, this problem is well-defined for any $\gamma \geqslant 1$.

2.2 Discrete Gaussian distributions

Let $\mathbf{S} \in \mathrm{GL}_n(\mathbb{R})$ be an invertible matrix. The Gaussian density function with parameter $\mathbf{S}$ is defined over $\mathbb{R}^n$ by

$$\rho_{\mathbf{S}}(\mathbf{x}) = \mathrm{e}^{-\pi\|\mathbf{S}^{-1}\mathbf{x}\|^2}.$$

When the matrix $\mathbf{S}$ is diagonal with diagonal coefficients all equal to some $\sigma > 0$, we also use the notation $\rho_\sigma = \rho_{\mathbf{S}}$. Let $L \subset \mathbb{R}^n$ be a full rank lattice, and $\mathbf{c} \in \mathbb{R}^n$. The discrete Gaussian distribution $D_{L,\mathbf{S},\mathbf{c}}$ over L with center $\mathbf{c}$ and parameter $\mathbf{S}$ is the distribution for which the probability of any $\mathbf{v} \in L$ is $\rho_{\mathbf{S}}(\mathbf{v}-\mathbf{c})/\rho_{\mathbf{S}}(L-\mathbf{c})$, where $\rho_{\mathbf{S}}(T) = \sum_{t\in T} \rho_{\mathbf{S}}(t)$ for any countable $T \subset \mathbb{R}^n$. Again, we will use the notation $D_{L,\sigma,\mathbf{c}}$ when $\mathbf{S} = \mathrm{diag}(\sigma)$ for some $\sigma > 0$. When $\mathbf{c} = \mathbf{0}$, we omit the subscript $\mathbf{c}$.

If $L \subset \mathbb{R}^n$ is a lattice, its smoothing parameter $\eta_\varepsilon(L)$ is defined as the smallest $\sigma > 0$ such that $\rho_{1/\sigma}(L^\star \backslash \{\mathbf{0}\}) \leqslant \varepsilon$, where $L^\star = \{\mathbf{c} \in \mathrm{span}(L) : \forall \mathbf{b} \in L : \langle \mathbf{b}^\star, \mathbf{b} \rangle \in \mathbb{Z}\}$ is the dual of L. For any n-dimensional lattice L and $\varepsilon > 0$, we have the following upper bound on the smoothing parameter (see [MR07, Le. 3.3]):

$$\eta_\varepsilon(L) \leqslant \sqrt{\frac{\ln(2n(1+1/\varepsilon))}{\pi}} \cdot \lambda_n(L). \tag{2.1}$$

The following Lemma (adapted from [GPV08, Th. 4.1]) shows that one can efficiently sample (bounded) elements from a distribution that is statistically close to a discrete Gaussian distribution. A proof can be found in Appendix A.2.

Lemma 2.2. *There exists a* ppt *algorithm that takes as input a basis* $\mathbf{B} = (\mathbf{b}_1, \ldots, \mathbf{b}_n)$ *of an* n*-dimensional lattice* L*, a parameter* $\sigma \geqslant \sqrt{n} \cdot \max_i \|\mathbf{b}_i\|$ *and a center* $\mathbf{c} \in \mathrm{Span}(L)$ *and outputs a sample from a distribution* $\widetilde{D}_{\mathbf{B},\sigma,\mathbf{c}}$ *such that*

- $D_{L,\sigma,\mathbf{c}} \approx_{2^{-\Omega(n)}} \widetilde{D}_{\mathbf{B},\sigma,\mathbf{c}}$*;*
- *for all* $\mathbf{v} \leftarrow \widetilde{D}_{B,\sigma,\mathbf{c}}$*, it holds that* $\|\mathbf{v} - \mathbf{c}\| \leqslant \sqrt{n} \cdot \sigma$ *and* $\mathbf{v} \neq \mathbf{0}$*.*

The following lemma bounds the statistical distance between two discrete Gaussian distributions over the same lattice L, depending on the distance between their centers and their parameter matrices. Similar results were already present in previous works, such as in [Reg09, Claim 2.2] for 1-dimensional continuous Gaussian distributions, and in the proof of [dBDPW20, Th. 4.4] for the case of ideal lattices with specific parameters and centers. Since the following precise statement seems new, we provide a proof in Appendix A.3 for the sake of completeness.

Lemma 2.3. *Let* $L \subset \mathbb{R}^n$ *be a full rank lattice,* $\mathbf{S}_1, \mathbf{S}_2 \in \mathrm{GL}_n(R)$ *be two invertible matrices and* $\mathbf{c}_1, \mathbf{c}_2 \in \mathbb{R}^n$ *be two vectors. If* $\eta_{1/2}(\mathbf{S}_1^{-1}L), \eta_{1/2}(\mathbf{S}_2^{-1}L) \leqslant 1/2$*, then it holds that*

$$\mathrm{dist}\left(D_{L,\mathbf{S}_1,\mathbf{c}_1}, D_{L,\mathbf{S}_2,\mathbf{c}_2}\right) \leqslant 4\sqrt{n} \cdot \left(\sqrt{\left\|\mathbf{S}_2^{-1}\mathbf{S}_1 - \mathbf{I}_n\right\|} + \sqrt{\|\mathbf{S}_2^{-1}(\mathbf{c}_1 - \mathbf{c}_2)\|}\right).$$

The next lemma states that a lattice Gaussian distribution with sufficiently large standard deviation is almost uniform when reduced modulo a sublattice.

Lemma 2.4 ([GPV08, Cor. 2.8]). *Let* $L_1 \subseteq L_2$ *be two lattices of rank* n*. If* $1 \geqslant \eta_\varepsilon(L_1)$ *for some* $\varepsilon < 1/2$*, then* $(D_{L_2,1} \bmod L_1) \approx_{2\varepsilon} U(L_2 \bmod L_1)$*.*

2.3 Number fields

Let K be a number field of degree $d \geqslant 2$ and $K_\mathbb{R} = K \otimes_\mathbb{Q} \mathbb{R}$. We let R denote its ring of integers. We identify any element of K with its canonical embedding vector $\sigma : x \mapsto (\sigma_1(x), \ldots, \sigma_d(x))^T \in \mathbb{C}^d$. This leads to an identification of $K_\mathbb{R}$ with $\{\mathbf{y} \in \mathbb{C}^d : \forall i \in [r_\mathbb{R}], y_i \in \mathbb{R} \text{ and } \forall i \in [r_\mathbb{C}], \overline{y_{r_\mathbb{R}+2(i+1)}} = y_{r_\mathbb{R}+2i+1}\}$,

where $r_{\mathbb{R}}$ and $r_{\mathbb{C}}$ respectively denote the number of real and complex embeddings. Via this identification, the set $K_{\mathbb{R}}$ is a real vector subspace of dimension d embedded in $\mathbb{C}^d$. In the following, for any element $x \in R, K$ or $K_{\mathbb{R}}$, we will let $\|x\|$ (resp. $\|x\|_\infty$) denote the Hermitian norm (resp. infinity norm) of the vector $\sigma(x) \in \mathbb{C}^d$. The set $\sigma(R)$ is a lattice, and the absolute field discriminant Δ_K is defined as $\Delta_K = |\det(\sigma(R))^2|$.[1] We have $\Delta_K \geqslant (\pi/4)^d \cdot (d^d/d!)^2$, which implies that we have $d = O(\log \Delta_K)$, for Δ_K growing to infinity.

The (absolute value of the) algebraic norm of $x \in K_{\mathbb{R}}$ is defined as $\mathcal{N}(x) = \prod_i |\sigma_i(x)|$. Any non-zero element $r \in R$ has algebraic norm $\geqslant 1$, which implies in particular that $\|r\|_\infty \geqslant 1$.

In this work, we assume that we know a monic polynomial $\Phi \in \mathbb{Z}[X]$ defining K and a $\mathbb{Z}$-basis $(r_1, \ldots, r_d)$ of R, where the r_i's are represented by polynomials modulo Φ (of degree $< d$) with rational coefficients. Let $D_\Phi > 0$ be the smallest integer such that $D_\Phi \cdot r_i$ has integral coefficients for all i (i.e., D_Φ is the common denominator to all the r_i polynomials), then the bit-size of D_Φ is polynomial in d and $\|\Phi\|$, where $\|\Phi\|$ is the Euclidean norm of the vector of coefficients of Φ (see for instance [Sut16, Se. 12.4]).

We will assume that this basis has been LLL-reduced [LLL82]. We define $\delta_K = \max_i \|r_i\|_\infty$. Since $\|r\|_\infty \geqslant 1$ for all $r \in R \backslash \{0\}$, we know that $\delta_K \geqslant 1$. Using Minkowski's second theorem and the LLL-reducedness of $(r_1, \ldots, r_d)$, we have that $\delta_K \leqslant \Delta_K^{O(1)}$. In the case of cyclotomic number fields, taking the power basis gives $\delta_K = 1$. For an element $x = \sum_i x_i r_i \in K_{\mathbb{R}}$, define $\lfloor x \rceil = \sum_i \lfloor x_i \rceil r_i$. We will also use the notation $\{x\} = x - \lfloor x \rceil$. It holds that $\|\{x\}\|_\infty \leqslant d/2 \cdot \delta_K$, and hence that $\|\{x\}\| \leqslant d^{3/2} \cdot \delta_K$.

For a rational $x = x_1/x_2$ with $x_1, x_2 \in \mathbb{Z}$ and $\gcd(x_1, x_2) = 1$, we define $\mathrm{size}(x) = 1 + \log|x_1| + \log|x_2|$. For an element $x = \sum_i x_i r_i \in K$, we define $\mathrm{size}(x) = \sum_i \mathrm{size}(x_i)$. The following lemma shows that if we have a sufficiently precise approximation to an embedding of $x \in K$, then one can recover x exactly. This seems folklore, but as we were unable to find a proof, we provide one in Appendix A. The result and the proof strategy are mentioned in [Coh00, Se. 6.2.4] in the context of quadratic fields and in Roblot's PhD thesis [Rob97] (just after Lemma 2.14). But both references are very brief on the topic. We note that a detailed study was done on a $\mathfrak{p}$-adic counterpart in [Bel04a].

Lemma 2.5. *Let $k \leqslant d$ arbitrary. There exists an algorithm that, given $\widetilde{y}$ such that $|\widetilde{y} - \sigma_k(x)| \leqslant 2^{-p}$ for some $x \in K$ and some $p \geqslant \mathrm{poly}(d, \log \delta_K, \log \|\Phi\|, \mathrm{size}(x))$, recovers x as a rational linear combination of the basis $(r_1, \ldots, r_d)$ of R in* ppt *with respect to p.*

2.4 Ideals and Modules

Ideals. An integral ideal I is a subset of R that is stable by addition and by multiplication with any element of R. A fractional ideal is a subset of K of the

[1] Note that in order to avoid having absolute values everywhere in the rest of the article, we define Δ_K as the absolute value of the discriminant of K.

form $x \cdot I$ for some $x \in K$ and some integral ideal $I \subseteq R$. We write $\langle z \rangle$ the principal (fractional) ideal generated by $z \in K$. Using the canonical embedding, any non-zero fractional ideal of K is identified to a d-dimensional lattice, called ideal lattice. The algebraic norm of an integral ideal $I \subseteq R$ is defined by $\mathcal{N}(I) = |R/I|$. We extend the notation to a fractional ideal xI with $x \in K$ and I an integral ideal, by setting $\mathcal{N}(xI) = \mathcal{N}(x) \cdot \mathcal{N}(I)$. For a non-zero fractional ideal $I = I_1/2$ with $I_1, I_2 \subseteq R$ and $\gcd(I_1, I_2) = R$, we define the quantity $\mathrm{size}(I) := \log(\mathcal{N}(I_1)) + \log(\mathcal{N}(I_2))$.

Two-element representation of an ideal. Any fractional ideal I can be generated by only two elements, i.e., there exist $x, y \in K$ such that $I = \langle x \rangle + \langle y \rangle$ (see, e.g., [Coh95, Prop. 4.7.7]). In fact, for any $x \in I \backslash \{0\}$, there exists $y \in I$ such that $I = \langle x \rangle + \langle y \rangle$. The lemma below states that computing such a y, given as input (I, x), can be done in probabilistic polynomial time.

Lemma 2.6 (Adapted from [Bel04b, Alg. 6.15] and [FS10, Th. 3]). *There exists a probabilistic algorithm taking a fractional ideal $I \subset K$ and a non-zero $x \in I$ as inputs, computing $y \in I$ such that $I = \langle x \rangle + \langle y \rangle$, and whose run-time is polynomial in* $\mathrm{size}(x)$, $\mathrm{size}(I)$ *and* $\log(\Delta_K)$.

Proof. Wlog, we can restrict the study to non-zero integral ideals. The algorithm is the same as the one given in [FS10, Fig. 1], except that in Step 1, the element x_1 is chosen to be x, rather than the first vector of a reduced basis. The correctness proof is unchanged. The upper bounds on the bit-sizes of the elements appearing during the algorithm execution do change, but one can check that all these bit-sizes stay polynomial in $\mathrm{size}(x)$, as well as the other quantities related to I and K that were already present in [FS10] (which are all polynomial in $\mathrm{size}(I)$ and $\log \Delta_K$). So overall, the run-time remains polynomial in $\mathrm{size}(x)$, $\mathrm{size}(I)$ and $\log \Delta_K$. □

Algorithmic problems over ideal lattices. The ideal-HSVP (or id-HSVP for short) problem is the HSVP problem restricted to lattices that are (fractional) ideal lattices. Using the fact that for an ideal lattice $I \subset K$ we have $\det(I) = \sqrt{|\Delta_K|} \cdot \mathcal{N}(I)$, the problem admits the following equivalent formulation.

Definition 2.7 (γ-id-HSVP). *Let $\gamma \geqslant 1$. Given as input a non-zero fractional ideal $I \subset K$ (represented by an arbitrary $\mathbb{Z}$-basis), the γ-id-*HSVP *problem asks to find an element $\mathbf{w} \in I \backslash \{\mathbf{0}\}$ such that $\|\mathbf{w}\| \leqslant \gamma \cdot \sqrt{d} \cdot \Delta_K^{1/(2d)} \cdot \mathcal{N}(I)^{1/d}$.*

Observe that γ-id-HSVP is equivalent to γ'-SVP in ideal lattices, up to polynomial losses $\leqslant \sqrt{d} \cdot \Delta_K^{1/(2d)}$ in the approximation factors γ and γ', thanks to the inequalities

$$\mathcal{N}(I)^{1/d} \leqslant \lambda_1(I) \leqslant \sqrt{d} \cdot \Delta_K^{1/(2d)} \cdot \mathcal{N}(I)^{1/d},$$

which hold for any non-zero fractional ideal I. The approximation factor loss is polynomial when $\Delta_K^{1/(2d)} \leqslant \mathrm{poly}(d)$.

If $\gamma = \exp(\widetilde{O}(d^{\alpha}))$ for $\alpha \in [0,1]$, then Id-HSVP can be solved using lattice reduction algorithms [Sch87], in time $\exp(\widetilde{O}(d^{1-\alpha}))$. In [CDW21], Cramer, Ducas and Wesolowski obtained a heuristic quantum polynomial-time algorithm for $\gamma = \exp(\widetilde{O}(d^{1/2}))$ for cyclotomic fields. In [PHS19], Pellet-Mary, Hanrot and Stehlé gave a quantum heuristic algorithm for $\gamma = \exp(\widetilde{O}((\log \Delta_K)^{\alpha+1})/d)$ running in time $\exp(\widetilde{O}((\log \Delta_K)^{1-2\alpha}))$ for any field K, where $\alpha \in [0, 1/2]$ is arbitrary. They also propose a classical variant of their algorithm, achieving the same approximation factor γ in time $\exp(\widetilde{O}((\log \Delta_K)^{\max(2/3,1-2\alpha)}))$ for any field K; and in time $\exp(\widetilde{O}(d^{\max(1/2,1-2\alpha)}))$ for cyclotomic fields. Both the classical and the quantum algorithms require an advice depending only on the field K, whose bit-length is bounded as $\exp(\widetilde{O}((\log \Delta_K)^{1-2\alpha}))$.

Smoothing ideals. The following lemma from [PRS17] provides a sufficient condition for a diagonal matrix $\mathbf{S}$ to be above the smoothing parameter of an ideal lattice.

Lemma 2.8 ([PRS17, Le. 6.9]). *Let $I \subset K$ be a fractional ideal and $\mathbf{S} \in \mathbb{R}^{d\times d}$ be a diagonal matrix with positive diagonal coefficients. Assume that*

$$c := (\prod_i S_{ii})^{1/d} \cdot (\mathcal{N}(I)\Delta_K)^{-1/d} \geqslant 1,$$

then $1 \geqslant \eta_{\varepsilon}(\mathbf{S}^{-1}I)$, where $\varepsilon = \exp(-c^2 d)$.

Modules. For $\ell \geqslant k \geqslant 1$, a rank-$k$ module $M \subset K_{\mathbb{R}}^{\ell}$ is a set of the form $M = \mathbf{b}_1 I_1 + \ldots + \mathbf{b}_k I_k$ for some non-zero ideals $(I_i)_i$ and some $K_{\mathbb{R}}$-linearly independent vectors $(\mathbf{b}_i)_i$ (i.e., if $\sum_i y_i \mathbf{b}_i = \mathbf{0}$, then all y_i's must be 0). The tuple $((I_i, \mathbf{b}_i))_i$ is called a pseudo-basis of M. If M admits a pseudo-basis for which all the I_i's are equal to R, then M is called free. We define $\det(M)$ as the determinant of M when identified with a kd-dimensional lattice via the canonical embedding σ. For any pseudo-basis $((I_i, \mathbf{b}_i))_i$ of M, we have

$$\det(M)^2 = \Delta_K^k \cdot \mathcal{N}\Big(\det_{K_{\mathbb{R}}}(\overline{\mathbf{B}}^T \mathbf{B}) \prod_i I_i^2\Big), \tag{2.2}$$

where $\det_{K_{\mathbb{R}}}$ is the determinant of a square matrix over $K_{\mathbb{R}}$.

2.5 Oracle Hidden Center Problem

In the search to decision reduction from Section 5, we will make use of the OHCP technique from [PRS17]. The proof of Proposition 2.10 is provided in Appendix A.5.

Definition 2.9 (Oracle Hidden Center Problem [PRS17, Def. 4.3]). *Let $\varepsilon, \delta \in (0,1)$ and $\beta \geqslant 1$. An OHCP instance consists in a scale parameter $D > 0$ and a randomized oracle $\mathcal{O} : \mathbb{R}^k \times \mathbb{R}_{\geqslant 0} \to \{0,1\}$ such that, for all $\mathbf{z} \in \mathbb{R}^k$ with $\|\mathbf{z} - \mathbf{z}^*\| \leqslant \beta D$ and $t \in \mathbb{R}_{\geqslant 0}$, it holds that $\Pr(\mathcal{O}(\mathbf{z}, t) = 1) = p(t + \log \|\mathbf{z} - \mathbf{z}^*\|)$,*

where $\mathbf{z}^* \in \mathbb{R}^k$ *is an unknown center satisfying* $\delta D \leqslant \|\mathbf{z}^*\| \leqslant D$ *and* $p(\cdot)$ *is an unknown function. The goal of the OHCP is to recover* $\tilde{\mathbf{z}} \in \mathbb{R}^k$ *such that* $\|\tilde{\mathbf{z}} - \mathbf{z}^*\| \leqslant \varepsilon D$.

Proposition 2.10 (Adapted from [PRS17, Prop. 4.4]). *There exists an algorithm that takes as input a parameter* $\kappa \geqslant 20\log(k+1)$*, the scaling parameter* D *and the oracle* $\mathcal{O}$ *of a* $(\exp(-\kappa), \exp(-\kappa), 1+1/\kappa)$*-OHCP instance in dimension* k*, and solves it with probability* $\geqslant 1-\exp(-\kappa)$*, in time* $\mathrm{poly}(\kappa, k)$*, provided the oracle* $\mathcal{O}$ *satisfies the extra following conditions. For some* $p_\infty \in [0,1]$ *and* $t^* \geqslant 0$ *we have*

1. $p(s^*) - p_\infty \geqslant 1/\kappa$*;*
2. $|p(t) - p_\infty| \leqslant 2\exp(-t/\kappa)$ *for any* $t \geqslant 0$*;*
3. *for any* $t_1, t_2 \geqslant 0$*, it holds that* $|p(t_1) - p(t_2)| \leqslant \kappa\sqrt{|t_1 - t_2|}$*;*

where $p(t) = \Pr(\mathcal{O}(\mathbf{0}, t) = 1)$.

3 Different variants of the NTRU problem

In this section, we define the three variants of the NTRU problem that we will consider in this work.

3.1 NTRU instances

We first define NTRU instances, which will be the inputs to the NTRU problem variants. We also consider the less standard case of tuple NTRU instances, which has also been considered in cryptographic constructions (see, e.g., the variant of the candidate cryptographic multilinear map from [GGH13] proposed in [LSS14, Se. 6]). All definitions of this section readily extend to the tuple setting, in a manner that is consistent with the second part of Definition 3.1.

Definition 3.1 ((γ, q)-NTRU instance). *Let* $q \geqslant 2$ *an integer and* $\gamma > 0$ *a real number. A* (γ, q)-NTRU *instance is an element* $h \in R_q$ *such that there exists* $(f, g) \in R^2 \backslash \{(0,0)\}$ *with* $g \cdot h = f \bmod q$ *and* $\|f\|, \|g\| \leqslant \sqrt{q}/\gamma$*. The pair* (f, g) *is called a trapdoor of the* NTRU *instance* h.

For $t \geqslant 1$ *and* γ *and* q *as above, a* (γ, q, t)*-tuple-*NTRU *instance is a tuple* $(h_i)_{i \leqslant t} \in R_q$ *such that there exists* $((f_i)_{i \leqslant t}, g) \in R^{t+1} \backslash \{(0, \ldots, 0)\}$ *with* $g \cdot h_i = f_i \bmod q$ *and* $\max_i \|f_i\|, \|g\| \leqslant \sqrt{q}/\gamma$.

For a uniform h in R_q, we will see below that the expected norm of a smallest trapdoor (f, g) is of the order of $\sqrt{q}$ (up to factors depending on the field). Hence, the quantity γ of an NTRU instance measures the gap between the size of a short trapdoor of h and the size of a smallest trapdoor of h we would have expected if h was uniform modulo q. Note also that any (γ, q)-NTRU instance is also a (γ', q)-NTRU instance for any $\gamma' \leqslant \gamma$ (the quantity γ is only a lower bound on the promised gap).

We now consider distributions over NTRU instances. To be useful for constructing cryptosystems, these distributions must be efficiently samplable and we also need to be able to sample, together with the NTRU instance h, a trapdoor (f, g) for h. This motivates the following definition.

Definition 3.2 ($(\mathcal{D}, \gamma, q)$-NTRU setup). *Let $q \geqslant 2$, $\gamma > 0$ and $\mathcal{D}$ a distribution over (γ, q)-NTRU instances. A $(\mathcal{D}, \gamma, q)$-NTRU setup is a* ppt *algorithm (with respect to $\log q$ and $\log \Delta_K$) sampling triples $(h, f, g) \in R_q \times R^2$ such that*

- *the marginal distribution of h is $\mathcal{D}$,*
- *$(f, g) \neq (0, 0)$ and $\|f\|, \|g\| \leqslant \sqrt{q}/\gamma$,*
- *$g \cdot h = f \bmod q$.*

It was shown in [SS11] that for power-of-2 cyclotomic fields, there exists a $(\mathcal{D}, \gamma, q)$-NTRU setup with $\mathcal{D} \approx_{2^{-\Omega(d)}} U(R_q^\times)$ for any prime $q \geqslant 5$ and some $\gamma = 1/\operatorname{poly}(d)$. This was extended to any cyclotomic field in [WW18]. In such cases, the decision NTRU problem introduced below is information-theoretically hard, if we replace $U(R_q)$ by $U(R_q^\times)$. In this work, we rather focus on the case of $\gamma \geqslant 1$.

3.2 Decision NTRU problem

We can now define the decision variant of the NTRU problem.

Definition 3.3 ($(\mathcal{D}, \gamma, q)$-dNTRU). *Let $q \geqslant 2$, $\gamma \geqslant 1$ and $\mathcal{D}$ a distribution over (γ, q)-NTRU instances. The $(\mathcal{D}, \gamma, q)$ decisional* NTRU *problem ($(\mathcal{D}, \gamma, q)$-*dNTRU *for short) asks to distinguish between samples from $\mathcal{D}$ and from $U(R_q)$. The advantage of an algorithm $\mathcal{A}$ against the $(\mathcal{D}, \gamma, q)$-*dNTRU *problem is defined as*

$$\mathrm{Adv}(\mathcal{A}) := \Big| \Pr_{h \leftarrow \mathcal{D}} \big(\mathcal{A}(h) = 1\big) - \Pr_{u \leftarrow U(R_q)} \big(\mathcal{A}(u) = 1\big) \Big|,$$

where the probabilities are also over the internal randomness of $\mathcal{A}$.

A reduction from dNTRU to sRLWE is sketched in [Pei16, Se. 4.4.4].

3.3 Search NTRU problems

We consider two different search variants for the NTRU problem. The first one consists in finding a trapdoor (f, g) for an NTRU instance h such that $\|f\|$ and $\|g\|$ are as small as possible, whereas the second variant only asks to recover any multiple (xf, xg) (with $x \in K$) of a small trapdoor (f, g). We explain below why both variants may be of interest. Further, for both variants, the definition comes with worst-case and average-case flavours.

Definition 3.4 ($(\mathcal{D}, \gamma, \gamma', q)$-NTRU$_{\mathrm{vec}}$ and (γ, γ', q)-wcNTRU$_{\mathrm{vec}}$). *Let $q \geqslant 2$, $\gamma \geqslant \gamma' > 0$ and $\mathcal{D}$ a distribution over (γ, q)-NTRU instances. The $(\mathcal{D}, \gamma, \gamma', q)$ average-case search* NTRU *vector problem ($(\mathcal{D}, \gamma, \gamma', q)$-NTRU$_{\mathrm{vec}}$ for short) asks, given as input some h sampled from $\mathcal{D}$, to compute a pair $(f, g) \in R^2 \backslash \{(0, 0)\}$*

such that $g \cdot h = f \bmod q$ *and* $\|f\|, \|g\| \leqslant \sqrt{q}/\gamma'$*. The advantage of an algorithm* $\mathcal{A}$ *against the* $(\mathcal{D}, \gamma, \gamma', q)$-$\mathrm{NTRU}_{\mathrm{vec}}$ *problem is defined as*

$$\mathrm{Adv}(\mathcal{A}) = \Pr_{h \leftarrow \mathcal{D}}\left(\mathcal{A}(h) = (f,g) \text{ with } \left|\begin{array}{l} g \cdot h = f \bmod q \\ (f,g) \neq (0,0) \\ \|f\|, \|g\| \leqslant \sqrt{q}/\gamma' \end{array}\right.\right),$$

where the probability is also over the internal randomness of $\mathcal{A}$*.*

The (γ, γ', q) *worst-case search* NTRU *vector problem* ((γ, γ', q)-$\mathrm{wcNTRU}_{\mathrm{vec}}$ *for short) asks, given as input a* (γ, q)-NTRU *instance* h*, to compute a pair* $(f,g) \in R^2 \backslash \{(0,0)\}$ *such that* $g \cdot h = f \bmod q$ *and* $\|f\|, \|g\| \leqslant \sqrt{q}/\gamma'$*.*

Before describing the second search variant of the NTRU problem, we prove the following lemma, which states that all short trapdoors (f,g) of an NTRU instance h are K-multiples of one another.

Lemma 3.5. *Let* $q \geqslant 2$*,* $\gamma > \sqrt{2}$ *and* h *be a* (γ, q)-NTRU *instance. Then, for all trapdoors* $(f,g), (f',g') \in R^2 \backslash \{(0,0)\}$ *with* $\|f\|, \|g\|, \|f'\|, \|g'\| \leqslant \sqrt{q}/\gamma$ *and* $g \cdot h = f \bmod q$*,* $g' \cdot h = f' \bmod q$*, it holds that* $(f,g) = x \cdot (f',g')$ *for some* $x \in K$*.*

Equivalently, there exists a unique field element $h_K \in K$ *such that, for all trapdoors* $(f,g) \in R^2 \backslash \{(0,0)\}$ *with* $\|f\|, \|g\| \leqslant \sqrt{q}/\gamma$ *and* $g \cdot h = f \bmod q$*, it holds that* $f/g = h_K$ *(where the division is performed in* K *and not modulo* q*).*

Proof. Let (f,g) and (f',g') be as in the lemma statement. Then

$$g' \cdot f = g' \cdot (g \cdot h) = g \cdot (g' \cdot h) = g \cdot f' \bmod q.$$

This implies that $g'f - gf' \in qR$. Moreover, we know that $\|g'f - gf'\| \leqslant \|g'\| \cdot \|f\| + \|g\| \cdot \|f'\| \leqslant 2q/\gamma^2 < q$ by assumption on γ. Since any non-zero element of R has euclidean norm at least 1, we conclude that all non-zero elements of qR have norm at least q, and so $g'f - gf' = 0$ in K as desired. The equivalent formulation follows immediately by taking $h_K = f/g$ for any short trapdoor (f,g). Note that g must be invertible in K because otherwise $g = 0$, which implies that $f \in qR$ and so f cannot satisfy $\|f\| \leqslant \sqrt{q}/\gamma$. □

We now describe our second search variant of the NTRU problem. Since we have seen in Lemma 3.5 that recovering a K-multiple of a short trapdoor is equivalent to recovering the (unique) element h_K, we will use this second approach in the description of the problem.

Definition 3.6 (**$(\mathcal{D}, \gamma, q)$-$\mathrm{NTRU}_{\mathrm{mod}}$ and (γ, q)-$\mathrm{wcNTRU}_{\mathrm{mod}}$**)**.** *Let* $q \geqslant 2$*,* $\gamma > \sqrt{2}$ *and* $\mathcal{D}$ *a distribution over* (γ, q)-NTRU *instances. The* $(\mathcal{D}, \gamma, q)$ *search* NTRU *module problem* ($(\mathcal{D}, \gamma, q)$-$\mathrm{NTRU}_{\mathrm{mod}}$ *for short) asks, given as input an* NTRU *instance* h *sampled from* $\mathcal{D}$*, to recover the unique field element* $h_K \in K$ *associated to* h *(as defined in Lemma 3.5). The advantage of an algorithm* $\mathcal{A}$ *against the* $(\mathcal{D}, \gamma, q)$-$\mathrm{NTRU}_{\mathrm{mod}}$ *problem is defined as*

$$\mathrm{Adv}(\mathcal{A}) = \Pr_{h \leftarrow \mathcal{D}}\Big(\mathcal{A}(h) = h_K\Big),$$

where the probability is also over the internal randomness of $\mathcal{A}$.

The (γ, q) worst-case search NTRU *module problem ((γ, q)-*wcNTRU$_{\text{mod}}$ *for short) asks, given as input a (γ, q)-*NTRU *instance h, to recover the unique field element $h_K \in K$ associated to h.*

We note that NTRU$_{\text{mod}}$ is definitionally convenient in that the quantity h_K that we are looking for is unique. In NTRU$_{\text{vec}}$, on the contrary, the short trapdoor (f, g) that we are looking for is far from being unique: it can always be multiplied by small elements of R to obtain other trapdoors.

Given a (γ, q)-NTRU instance h, one can construct the following free rank-2 module M_h:

$$M_h := \begin{pmatrix} 1 & 0 \\ h & q \end{pmatrix} \cdot R^2 = \left\{ (g, f)^T \in R^2 \,|\, g \cdot h = f \bmod q \right\}.$$

This module is called the NTRU-module associated to h. As a lattice, it has determinant $\det M_h = \Delta_K \cdot q^d$ and dimension $2d$. If it were a generic lattice with such determinant and dimension, we would heuristically expect that its minimum is $\Theta(\sqrt{d} \cdot \Delta_K^{1/(2d)} \cdot \sqrt{q})$. However, since h is a (γ, q)-NTRU instance with $\gamma > \sqrt{2}$, we know that there exists an unexpectedly short vector $(g, f)^T$ in the module M_h. This short vector is not unique, any small multiple $(rg, rf)^T$ with $r \in R$ small is also a short vector of M_h. However, Lemma 3.5 implies that the module spanned by all these short vectors has rank 1 and is unique. Moreover, since this module contains unexpectedly short vectors, it will have an unexpectedly small volume. Summing up, the rank-2 module M_h has multiple unexpectedly short vectors and a unique unexpectedly short rank-1 sub-module. NTRU$_{\text{vec}}$ asks to find any of the unexpectedly short non-zero vectors of M_h, whereas NTRU$_{\text{mod}}$ asks to recover the unique short rank-1 sub-module (hence the names "NTRU vector" and "NTRU module").

3.4 Elementary relations between the different NTRU problems

NTRU$_{\text{mod}}$ and NTRU$_{\text{vec}}$ respectively reduce to their worst-case counterparts. The proof of the following lemma is similarly direct.

Lemma 3.7. *Let $q \geqslant 2$, $\gamma \geqslant \gamma' > \sqrt{2}$. Then there exists a* ppt *reduction from (γ, q)-*wcNTRU$_{\text{mod}}$ *to (γ, γ', q)-*wcNTRU$_{\text{vec}}$*. In the average-case setup, the reduction preserves the distribution of instances.*

If one assumes that ideal-HSVP is easy, then the latter admits a converse result. The proof of the following lemma is available in Appendix B.1.

Lemma 3.8. *Let $q \geqslant 2$, $\gamma \geqslant \gamma' > \sqrt{2}$ and $\varepsilon > 0$. Then there exists a* ppt *reduction from $(\gamma, \gamma_{\text{vec}}, q)$-*wcNTRU$_{\text{vec}}$ *to (γ, q)-*wcNTRU$_{\text{mod}}$ *and γ_{hsvp}-id-*HSVP*, where*

$$\gamma_{\text{vec}} = \frac{1}{(1+\varepsilon)\sqrt{2}\Delta_K^{1/(2d)}} \cdot \frac{\gamma}{\gamma_{\text{hsvp}}}.$$

In the average-case setup, the NTRU$_{\text{mod}}$ *and* NTRU$_{\text{vec}}$ *instance distributions are identical.*

To reduce dNTRU to $\mathrm{NTRU}_{\mathrm{mod}}$, it suffices to show that for a uniform h, we do not expect an unexpectedly short non-zero vector (or short rank-1 submodule) in M_h. We defer the proof of the following lemma to Appendix B.2.

Lemma 3.9. *Let $q \geqslant 2$ be a prime that does not divide Δ_K, $\gamma > 16 \cdot \sqrt{d} \cdot \Delta_K^{1/(2d)}$ and $\mathcal{D}$ a distribution over (γ, q)-NTRU instances. Then there exists a* ppt *reduction from $(\mathcal{D}, \gamma, q)$-dNTRU to $(\mathcal{D}, \gamma, q)$-$\mathrm{NTRU}_{\mathrm{mod}}$. Further, the reduction makes a single call to the $\mathrm{NTRU}_{\mathrm{mod}}$ oracle, and if the advantage of the $\mathrm{NTRU}_{\mathrm{mod}}$ solver is ε, then the advantage of the resulting* dNTRU *solver is $\geqslant \varepsilon - 2^{-d}$.*

The objective of the next two sections is to (partly) complete the picture by giving two more sophisticated reductions: a reduction from id-HSVP to $\mathrm{NTRU}_{\mathrm{vec}}$ and a reduction from $\mathrm{NTRU}_{\mathrm{mod}}$ to dNTRU.

4 Reduction from ideal-HSVP to $\mathrm{NTRU}_{\mathrm{vec}}$

This section is devoted to reducing worst-case id-HSVP to average-case $\mathrm{NTRU}_{\mathrm{vec}}$. For this purpose, we first exhibit a Karp reduction from worst-case id-HSVP to $\mathrm{wcNTRU}_{\mathrm{vec}}$. This reduction is then enhanced by using the worst-case to average-case reduction for id-HSVP from [dBDPW20], resulting in a reduction from worst-case id-HSVP to average-case $\mathrm{NTRU}_{\mathrm{vec}}$, where the $\mathrm{NTRU}_{\mathrm{vec}}$ average-case distribution is defined as the distribution obtained by applying the worst-case to worst-case reduction to the distribution on ideals from [dBDPW20]. In the process, we improve the reduction of [dBDPW20] to better suit our needs. We extend it to regimes in which it is not polynomial-time anymore, but allows to reach smaller values for the NTRU modulus q, and we show that it allows to sample from the average-case id-HSVP distribution along with a short non-zero element of the ideal (provided q is sufficiently large, or we have access to a quantum computer). The latter is important to allow to sample from the average-case distribution over NTRU instances, along with a trapdoor.

4.1 Transforming an ideal lattice into an NTRU module

In this section, we show how to efficiently 'embed' an ideal lattice into an NTRU module such that any sufficiently short vector of the NTRU module provides a short vector of the embedded ideal lattice. We first give an efficient reduction from ideal-HSVP to worst-case vectorial NTRU.

Theorem 4.1. *Let $q \geqslant 2$ and $\gamma \geqslant \gamma' > 0$ with $\gamma \cdot \gamma' \cdot \sqrt{d} > 1$. Let $\gamma_{\mathrm{hsvp}} = 4d\delta_K \cdot \gamma/\gamma'$. There is a* ppt *(Karp) reduction from γ_{hsvp}-id-HSVP to (γ, γ', q)-$\mathrm{wcNTRU}_{\mathrm{vec}}$ for ideals $I \subseteq R$ satisfying $\mathcal{N}(I) \in [N/2^d, N]$, with*

$$N = \left\lfloor \left(\frac{\sqrt{q}}{\gamma \cdot d^{1.5} \cdot \delta_K \cdot \Delta_K^{\frac{1}{2d}}} \right)^d \right\rfloor.$$

Note that the reduction is restricted to integral ideals of bounded norms. The lower bound is not restrictive: given a non-zero integral ideal I such that $\mathcal{N}(I) \leqslant N$, we can scale it to the non-zero integral ideal $I' = \lfloor (N/\mathcal{N}(I))^{1/d} \rfloor \cdot I$, which satisfies $\mathcal{N}(I') \in [N/2^d, N]$ and for which a γ_{hsvp}-id-HSVP solution directly leads to a γ_{hsvp}-id-HSVP solution for I. Concerning the upper bound restriction, the id-HSVP worst-case to average-case reduction from [dBDPW20] (as refined in Subsection 4.2) shows that we can wlog focus on integral ideals I of norms $N \approx 2^{d^{1+\alpha}}$ for some $\alpha \in (0, 1]$. This impacts the choice of the NTRU modulus q.

Let us now focus on the problem parameters. If we put aside factors that depend only on the number field, we can set $N^{1/d} \approx \sqrt{q}/\gamma$, and we then obtain that $\gamma_{\mathrm{hsvp}} \approx \gamma/\gamma'$. This means that the approximation factor (which is γ/γ' in the NTRU case) stays roughly the same, and that the root determinant of the NTRU module is γ times larger than the one of the ideal lattice.

Algorithm 4.1 Transforming an ideal lattice into an NTRU instance

Input: A $\mathbb{Z}$-basis of a non-zero ideal $I \subseteq R$ and a modulus q.
Output: An NTRU instance h.
1: Compute $z \in K$ such that $I = R \cap \langle z \rangle$ (see Lemma 4.2).
2: Let $h = \lfloor q/z \rceil \bmod q \in R_q$.
3: **return** h

The transformation that embeds an ideal lattice into an NTRU module is described in Algorithm 4.1. In Lemma 4.3, we show some properties of Algorithm 4.1, which will be used to prove Theorem 4.1.

Lemma 4.2. *There exists a* ppt *algorithm (in* $\mathrm{size}(I)$ *and* $\log \Delta_K$*) which, given a non-zero integral ideal* I *as input, computes* $z \in K$ *such that* $I = R \cap \langle z \rangle$.

Proof. If $I = 0$, then the algorithm returns $z = 0$. If $I = R$, it returns $z = 1$. We now assume that I is neither 0 nor R. Since $I \subseteq R$, it holds that $1 \in I^{-1}$. Let $y \in I^{-1}$ be the output of the algorithm of Lemma 2.6, given $(I^{-1}, 1)$ as input: we have $I^{-1} = \langle 1 \rangle + \langle y \rangle$. Note that $I \neq R$ implies that $y \neq 0$. We then define $z = 1/y$, which fulfills our needs as $J_1 \cap J_2 = (J_1^{-1} + J_2^{-1})^{-1}$ for any non-zero fractional ideals J_1 and J_2. □

When using Lemma 4.2 in Algorithm 4.1, the element z is necessarily non-zero, as I is non-zero. The analysis of Algorithm 4.1 follows the intuition provided by the case of principal ideals (with a known generator) described in the introduction.

Lemma 4.3. *Let* $q \geqslant 2$ *and* $I \subseteq R$ *a non-zero integral ideal. On input* (I, q)*, Algorithm 4.1 outputs* $h \in R_q$ *such that*

- *there exists a pair* $(f, g) \in R^2 \backslash \{(0, 0)\}$ *with* $g \cdot h = f \bmod q$ *and* $\|f\|, \|g\| \leqslant d^{1.5} \cdot \delta_K \cdot \Delta_K^{1/(2d)} \cdot \mathcal{N}(I)^{1/d}$;

- *for any pair* $(f', g') \in R^2 \backslash \{(0,0)\}$ *with* $g' \cdot h = f' \bmod q$ *and* $\|f'\|_\infty, \|g'\|_\infty < q/(d \cdot \delta_K \cdot \Delta_K^{1/(2d)} \cdot \mathcal{N}(I)^{1/d})$*, we have* $g' \in I \backslash \{0\}$.

Moreover, Algorithm 4.1 runs in time polynomial in $\mathrm{size}(I)$, $\log q$ *and* $\log \Delta_K$.

Proof. The run-time of the algorithm follows from Lemma 4.2. For the proofs of the two main statements, we consider $g \in I \backslash \{0\}$ with minimal infinity norm. By Minkowski's bound, we have that $\|g\|_\infty \leqslant \Delta_K^{1/(2d)} \cdot \mathcal{N}(I)^{1/d}$.

We now prove the existence of f such that (f, g) is a short trapdoor for h. By multiplying g with h, we obtain

$$g \cdot h = g \cdot \lfloor q/z \rceil = g \cdot q/z + f,$$

with $f := -g \cdot \{q/z\}$. Since $g \in I$ and $z^{-1} \in I^{-1}$ (because $I \subseteq \langle z \rangle$), we have that $g \cdot q/z \in qR$. This implies that $f \in R$ and $gh = f \bmod q$, as desired. Let us now compute an upper bound on the norm of f (we already know that $\|g\| \leqslant \sqrt{d} \cdot \Delta_K^{1/(2d)} \cdot \mathcal{N}(I)^{1/d}$). We know from the preliminaries that $\|\{q/z\}\|_\infty \leqslant d/2 \cdot \delta_K$, from which we obtain:

$$\|f\| \leqslant \|g\| \cdot (d \cdot \delta_K) \leqslant d^{3/2} \cdot \delta_K \cdot \Delta_K^{\frac{1}{2d}} \cdot \mathcal{N}(I)^{\frac{1}{d}}.$$

Let us now prove the second property of the lemma. Let $(g', f') \in R^2 \backslash \{(0,0)\}$ be such that $g' \cdot h = f' \bmod q$ and

$$\|f'\|_\infty, \|g'\|_\infty < \frac{q}{d \cdot \delta_K \cdot \Delta_K^{\frac{1}{2d}} \cdot \mathcal{N}(I)^{\frac{1}{d}}}.$$

We first show that $g' \neq 0$. Assume by contradiction that $g' = 0$. Then $f' = 0 \bmod q$, i.e., $f' \in qR$. But any non-zero element of qR has infinity norm $\geqslant q$ (using the fact that any non-zero element of R has infinity norm $\geqslant 1$). Since we know that $\|f'\|_\infty < q$, we conclude that $f' = 0$, which contradicts the assumption that $(f', g') \neq (0,0)$.

We now show that $g' \in I$. Recall that z is such that $I = R \cap \langle z \rangle$. Since we already know that $g' \in R$, it suffices to prove that $g' \in \langle z \rangle$, i.e., that $g'/z \in R$. By definition of h, we have:

$$g' \cdot q/z = g' \cdot h + g' \cdot \{q/z\} = f' + g' \cdot \{q/z\} + q \cdot r,$$

for some $r \in R$. Multiplying this equation by g/q (recall that g is a shortest non-zero vector of I for the infinity norm), we obtain

$$g' \cdot g/z = (f' + g' \cdot \{q/z\}) \cdot g/q + g \cdot r.$$

We have seen that $g/z \in R$, so that both terms $g' \cdot g/z$ and $g \cdot r$ are in R. We hence have that the term $(f' + g' \cdot \{q/z\}) \cdot g/q$ must also belong to R. Further, we know that

$$\begin{aligned} \|(f' + g' \cdot \{q/z\}) \cdot g/q\|_\infty &\leqslant (\|f'\|_\infty + \|g'\|_\infty \cdot \|\{q/z\}\|_\infty) \cdot \|g\|_\infty/q \\ &\leqslant \max(\|f'\|_\infty, \|g'\|_\infty) \cdot (1 + d/2 \cdot \delta_K) \cdot \Delta_K^{\frac{1}{2d}} \cdot \mathcal{N}(I)^{\frac{1}{d}}/q. \end{aligned}$$

By assumption, the above is < 1. Since no non-zero element of R has infinity norm < 1, we conclude that $f' + g' \cdot \{q/z\} = 0$. This implies that $g' \cdot q/z = q \cdot r$. Dividing this equality by q, we obtain that $g'/z \in R$, as desired. □

We are now ready to prove Theorem 4.1.

Proof (Theorem 4.1). The reduction consists in calling Algorithm 4.1 on I and q to obtain some $h \in R_q$, then calling the wcNTRU$_{\text{vec}}$ oracle on h and returning the oracle output.

Let $I \subseteq R$ be a γ_{hsvp}-id-HSVP instance satisfying $\mathcal{N}(I) \in [N/2^d, N]$, with N as in the theorem statement. The first statement of Lemma 4.3 ensures that the element h computed by the reduction is a valid (γ, γ', q)-wcNTRU$_{\text{vec}}$ instance. The wcNTRU$_{\text{vec}}$ oracle hence outputs a pair $(f', g') \in R^2 \backslash \{(0,0)\}$ such that $g' \cdot h = f' \bmod q$ and $\|f'\|, \|g'\| \leqslant \sqrt{q}/\gamma'$. By the parameter conditions, the assumption of the second statement of Lemma 4.3 holds. We hence have that $g' \in I \backslash \{0\}$. Further, by definition of N, the lower bound on $\mathcal{N}(I)$ and definition of γ_{hsvp}, we have

$$\|g'\| \leqslant \frac{\sqrt{q}}{\gamma'} \leqslant \frac{2^{1/d} \cdot N^{\frac{1}{d}} \cdot \gamma \cdot d^{1.5} \cdot \delta_K \cdot \Delta_K^{\frac{1}{2d}}}{\gamma'} \leqslant \gamma_{\text{hsvp}} \cdot \sqrt{d} \cdot \Delta_K^{\frac{1}{2d}} \cdot \mathcal{N}(I)^{\frac{1}{d}}.$$

Note that we used the inequality $\lfloor x \rceil \geqslant x/2$, which holds for any $x \geqslant 1$. □

4.2 From worst-case id-HSVP to average-case id-HSVP

In [dBDPW20], the authors gave a worst-case to average-case reduction for id-HSVP, for a certain average-case distribution of ideals. We adapt [dBDPW20, Th. 4.5] to Theorem 4.4 below, so that it better fits with our application. We explain in Appendix C.1 how to adapt the proof.

Theorem 4.4 (Adapted from [dBDPW20, Th. 4.5], ERH). *Let K a number field of degree d and $N \geqslant (12d^{1.5} \log d \cdot \delta_K \cdot \Delta_K^{1/(2d)})^d$ an integer. Let $\gamma > 0$. There exist $\gamma' = \gamma \cdot O(d^{1.5} \Delta_K^{1/d})$, a distribution $\mathcal{D}_N^{\text{id-HSVP}}$ over non-zero integral ideals of K of norm $\leqslant N$ and a reduction:*

- *from worst-case γ'-id-*HSVP *for all fractional ideals of K,*
- *to average-case γ-id-*HSVP *for integral ideals distributed from $\mathcal{D}_N^{\text{id-HSVP}}$.*

*The reduction decreases the success probability by at most $2^{-\Omega(d)}$, makes a single call to the average-case γ-id-*HSVP *oracle, and runs in time $T_\beta^{\text{id-HSVP}} + \text{poly}(\log N, \text{size}(I), \log \Delta_K)$ where*

- *I is the input (worst-case) ideal;*
- *$T_\beta^{\text{id-HSVP}}$ is the time needed to solve id-HSVP with approximation factor $2^{d/\beta}$ and*

$$\beta = \left\lceil \frac{d}{\log\left(N^{1/d}/(6d^{1.5} \log d \cdot \delta_K \cdot \Delta_K^{1/(2d)})\right)} \right\rceil.$$

Moreover, there exist $N_0 = \mathrm{poly}(\Delta_K^{1/d}, \delta_K, d)^d$ *and a* ppt *algorithm* $\mathcal{A}$ *(with respect to* $\log N$ *and* $\log \Delta_K$*) such that, for all* $N \geqslant N_0$*, algorithm* $\mathcal{A}$ *samples pairs* (J, w) *such that:*

- *the ideal* J *is a non-zero integral ideal of norm* $\leqslant N$*;*
- *the distribution* $\widetilde{\mathcal{D}}_N^{\text{id-HSVP}}$ *of* J *satisfies* $\widetilde{\mathcal{D}}_N^{\text{id-HSVP}} \approx_{2^{-\Omega(d)}} \mathcal{D}_N^{\text{id-HSVP}}$*;*
- *the element* $w \in J \backslash \{0\}$ *satisfies* $\|w\| \leqslant \mathrm{poly}(d, \delta_K, \Delta_K^{1/d}, 2^{\sqrt{\log \Delta_K + d \log d}}) \cdot \mathcal{N}(J)^{1/d}$*.*

If we have access to a factoring oracle or if $N \geqslant N_0' = N_0 \cdot 2^{O(d\sqrt{\log \Delta_K + d \log d})}$*, then we can reduce the size of* w *down to* $\|w\| \leqslant \mathrm{poly}(d, \delta_K, \Delta_K^{1/d}) \cdot \mathcal{N}(J)^{1/d}$*.*

Note that even though the reduction relies on a worst-case id-HSVP solver, the latter is with an approximation factor $2^{d/\beta}$ which is typically much larger than γ'. This implies that $T_\beta^{\text{id-HSVP}}$ is expected to be much smaller than the time needed to solve γ'-id-HSVP. Assume that $\Delta_K^{1/(2d)}$ and δ_K are both $\mathrm{poly}(d)$ and that we use the lattice reduction algorithm from [Sch87] with block size β to solve $2^{d/\beta}$-id-HSVP. It runs in time $T_\beta^{\text{id-HSVP}} = 2^{O(\beta)}$ (up to a $\mathrm{poly}(\log N, \log \Delta_K)$ factor). Then, it can be seen that the reduction is polynomial-time when $N = 2^{\Omega(d^2)}$; it becomes more expensive when N is below this bound; and it ends up being $2^{O(d)}$ when $N \approx \mathrm{poly}(d)^d$. The run-time of the reduction can be improved using id-HSVP algorithms such as those mentioned in Subsection 2.3. In all cases, we note that one can sample ideals J from $\mathcal{D}_N^{\text{id-HSVP}}$, together with a short vector of J in quantum polynomial time even for small N, and in classical polynomial time for larger N's (of the order of $2^{O(d^{1.5}\sqrt{\log d})}$ if $\Delta_K^{1/(2d)}$ and δ_K are both $\mathrm{poly}(d)$).

All the ingredients for the proof of Theorem 4.4 are present in [dBDPW20], however the latter only considered the case of $N \geqslant (2^d \cdot 6d^{1.5} \log d \cdot \Delta_K^{1/(2d)} \cdot \delta_K)^d$, since this is the range of parameters for which the reduction runs in polynomial time. The generalization to smaller N and larger run-time is relatively immediate and is provided in Appendix C.1. A further difference with [dBDPW20] is that the distribution $\mathcal{D}_N^{\text{id-HSVP}}$ in [dBDPW20] is over the inverses of integral ideals (see [dBDPW20, Le. 4.1]) whereas here it is more convenient to have a distribution over integral ideals. Finally, we also explain in Appendix C.1 how to sample ideals from $\mathcal{D}_N^{\text{id-HSVP}}$ with a somehow short vector.

4.3 An average-case distribution of NTRU instances

In this subsection, we define a distribution $\mathcal{D}_{q,\gamma}^{\text{NTRU}}$ over (γ, q)-NTRU instances. This distribution is defined as the one being produced by Algorithm 4.2. In fact, Algorithm 4.2 actually provides a $(\widetilde{\gamma}, q)$-NTRU setup for some $\widetilde{\gamma} \geqslant \gamma$, i.e., the instance h can be sampled along with a trapdoor (f, g) that may be a little larger than a shortest one.

Lemma 4.5. *There exist* $\Gamma = \mathrm{poly}(d, \delta_K, \Delta_K^{1/d})$ *and* $\Gamma' = \Gamma \cdot 2^{O(\sqrt{\log \Delta_K + d \log d})}$ *such that if* $\sqrt{q}/\gamma \geqslant \Gamma$ *(resp.* $\sqrt{q}/\gamma \geqslant \Gamma'$*), then Algorithm 4.2 runs in quantum (resp. classical) polynomial time (with respect to* $\log q$ *and* $\log \Delta_K$*).*

Algorithm 4.2 Sampling h from $\mathcal{D}^{\mathrm{NTRU}}_{q,\gamma}$ together with a trapdoor

Input: An integer $q \geqslant 2$ and a real $\gamma \geqslant 1$
Output: A triple $(h, f, g) \in R_q \times R^2$.

1: Let $N = \left\lfloor \left(\frac{\sqrt{q}}{\gamma \cdot d^{1.5} \cdot \delta_K \cdot \Delta_K^{1/(2d)}} \right)^d \right\rfloor$.
2: Sample I from $\widetilde{\mathcal{D}}^{\text{id-HSVP}}_N$ with $v \in I \backslash \{0\}$ such that $\|v\| \leqslant \mathrm{poly}(d, \delta_K, \Delta_K^{1/d}) \cdot \mathcal{N}(I)^{1/d}$ (see Theorem 4.4).
3: Let $I' = \lfloor (N/\mathcal{N}(I))^{1/d} \rfloor \cdot I$ and $v' = \lfloor (N/\mathcal{N}(I))^{1/d} \rfloor \cdot v$.
4: Run Algorithm 4.1 on I'; let $h \in R_q$ be the output and z as in Algorithm 4.1.
5: Compute $g = v'$ and $f = -g \cdot \{q/z\}$.
6: **return** (h, f, g).

Proof. Let $\Gamma = 2d^{1.5} \cdot \delta_K \cdot \Delta_K^{1/(2d)} \cdot N_0^{1/d}$ (resp. $\Gamma' = 2d^{1.5} \cdot \delta_K \cdot \Delta_K^{1/(2d)} \cdot (N_0')^{1/d}$), where N_0 (resp. N_0') is as in the second part of Theorem 4.4. Note that we have $\Gamma = \mathrm{poly}(d, \delta_K, \Delta_K^{1/d})$ (resp. $\Gamma' = \Gamma \cdot 2^{O(\sqrt{\log \Delta_K + d \log d})}$) as desired. Moreover, by definition of N and using the fact that $\sqrt{q}/\gamma \geqslant \Gamma$ (resp. $\sqrt{q}/\gamma \geqslant \Gamma'$), we have $N \geqslant N_0$ (resp. $N \geqslant N_0'$). Hence, by Theorem 4.4, one can sample (I, v) in Step 2 in quantum (resp. classical) time $\mathrm{poly}(\log N, \log \Delta_K) = \mathrm{poly}(\log \Delta_K, \log q)$.

By Theorem 4.4, we also know that the ideal I is non-zero and satisfies $\mathcal{N}(I) \leqslant N$, hence $\lfloor (N/\mathcal{N}(I))^{1/d} \rfloor \neq 0$. Therefore, the ideal I' computed at Step 3 is also non-zero, and v' is a non-zero element of I'. Thanks to Lemma 4.3, we know that Algorithm 4.1 can be run on I' in time $\mathrm{poly}(\mathrm{size}(I'), \log q, \log \Delta_K)$. Since I' is integral and $\mathcal{N}(I') \leqslant N \leqslant q^d$, we conclude that $\mathrm{size}(I') \leqslant \mathrm{poly}(\log q, \log \Delta_K)$. Finally, computing f using the formula $-g \cdot \{q/z\}$ can also be done in time $\mathrm{poly}(\log q, \log \Delta_K)$, since the rounding operation in R is efficient. □

Now that it is established that Algorithm 4.2 terminates, we can formally define $\mathcal{D}^{\mathrm{NTRU}}_{\gamma,q}$ as the distribution produced by the algorithm.

Definition 4.6 (Distribution $\mathcal{D}^{\mathrm{NTRU}}_{q,\gamma}$). *Let q, γ as in Algorithm 4.2. The distribution $\mathcal{D}^{\mathrm{NTRU}}_{\gamma,q}$ over R_q is defined as the distribution of the element h produced by Algorithm 4.2 on input (q, γ).*

Lemma 4.7. *The support of the distribution $\mathcal{D}^{\mathrm{NTRU}}_{q,\gamma}$ is contained in the set of (γ, q)-NTRU instances.*

Proof. Let h be computed by Algorithm 4.2 on input (q, γ). By the first property of Lemma 4.3, there exists a trapdoor $(f^\star, g^\star) \neq (0, 0)$ for h, with $\|f^\star\|, \|g^\star\| \leqslant d^{1.5} \cdot \delta_K \cdot \Delta_K^{1/(2d)} \cdot \mathcal{N}(I')^{1/d}$. We have $\mathcal{N}(I') = \lfloor (N/\mathcal{N}(I))^{1/d} \rfloor^d \cdot \mathcal{N}(I) \leqslant N$. Using the definition of N, we conclude that $\|f^\star\|, \|g^\star\| \leqslant \sqrt{q}/\gamma$. □

Algorithm 4.2 gives a way to sample from $\mathcal{D}^{\mathrm{NTRU}}_{q,\gamma}$ together with a trapdoor.

Lemma 4.8. *Let q, γ as in Algorithm 4.2 and Γ (resp. Γ') as in Lemma 4.5. If $\sqrt{q}/\gamma \geqslant \Gamma$ (resp. $\sqrt{q}/\gamma \geqslant \Gamma'$), then there exist $\widetilde{\gamma} = \gamma / \mathrm{poly}(d, \delta_K, \Delta_K^{1/d})$ such that Algorithm 4.2 is a $(\mathcal{D}^{\mathrm{NTRU}}_{q,\gamma}, \widetilde{\gamma}, q)$-NTRU quantum (resp. classical) setup.*

Proof. We have already seen in Lemma 4.5 that Algorithm 4.2 is quantum (resp. classical) ppt. We have seen in Lemma 4.7 that $\mathcal{D}$ is a distribution over (γ, q)-NTRU instances. It is hence a distribution over $(\widetilde{\gamma}, q)$-NTRU instances, as $\widetilde{\gamma} \leqslant \gamma$. We now show that the sampled pair $(f, g) \neq (0, 0)$ satisfies $g \cdot h = f \bmod q$ and $\|f\|, \|g\| \leqslant \sqrt{q}/\gamma \cdot \mathrm{poly}(d, \delta_K, \Delta_K^{1/d})$.

We have already seen that $g = v'$ is non-zero. Moreover, by definitions of $f = -g \cdot \{q/z\}$ and $h = \lfloor q/z \rceil$, is holds that $f = g \cdot h \bmod q$ (see the proof of Lemma 4.3). Further, we have (successively using Theorem 4.4, the definition of I' and the definition of N):

$$\begin{aligned}\|g\| = \|v'\| &\leqslant \mathrm{poly}(d, \delta_K, \Delta_K^{1/d}) \cdot \mathcal{N}(I')^{1/d} \leqslant \mathrm{poly}(d, \delta_K, \Delta_K^{1/d}) \cdot N^{1/d} \\ &\leqslant \mathrm{poly}(d, \delta_K, \Delta_K^{1/d}) \cdot \frac{\sqrt{q}}{\gamma}.\end{aligned}$$

Moreover, by definition of f, we know that $\|f\| \leqslant \|g\| \cdot (d \cdot \delta_K)$. Hence, there exists some $\widetilde{\gamma} = \gamma/\mathrm{poly}(d, \delta_K, \Delta_K^{1/d})$ such that $\|f\|, \|g\| \leqslant \sqrt{q}/\widetilde{\gamma}$, as desired. □

4.4 From average-case id-HSVP to average-case NTRU

By combining the results from Subsections 4.1 and 4.3, we obtain that, for well-chosen parameters, average-case id-HSVP for distribution $\mathcal{D}_N^{\text{id-HSVP}}$ reduces to average-case $\mathrm{NTRU}_{\mathrm{vec}}$ for distribution $\mathcal{D}_{q,\gamma}^{\mathrm{NTRU}}$. The proof of Theorem 4.9 is available in Appendix C.2. This theorem can in turn be combined with Theorem 4.4 to obtain a reduction from worst-case id-HSVP to average-case $\mathrm{NTRU}_{\mathrm{vec}}$.

Theorem 4.9. *Let $q \geqslant 2$, $\gamma \geqslant 1$ and $\gamma' > 0$ such that $\gamma \cdot \gamma' \cdot \sqrt{d} > 1$ and $\sqrt{q}/\gamma \geqslant 13 \cdot d^3 \log d \cdot \delta_K^2 \cdot \Delta_K^{1/d}$. Define:*

$$N = \left\lfloor \left(\frac{\sqrt{q}}{\gamma \cdot d^{1.5} \cdot \delta_K \cdot \Delta_K^{1/(2d)}} \right)^d \right\rfloor \quad \text{and} \quad \gamma_{\mathrm{hsvp}} = \frac{\gamma}{\gamma'} \cdot 4d\delta_K.$$

There is a ppt *reduction (with respect to $\log \Delta_K$ and $\log q$) from average-case γ_{hsvp}-id-HSVP for ideals sampled from $\widetilde{\mathcal{D}}_N^{\text{id-HSVP}}$ to $(\mathcal{D}_{q,\gamma}^{\mathrm{NTRU}}, \gamma, \gamma', q)$-$\mathrm{NTRU}_{\mathrm{vec}}$. The reduction makes a single call to the $\mathrm{NTRU}_{\mathrm{vec}}$ oracle and preserves the success probability.*

5 A search to decision reduction for NTRU

In this section, we provide a reduction from average-case search-$\mathrm{NTRU}_{\mathrm{mod}}$ with distribution $\mathcal{D}^{\mathrm{s}}$ to average-case dec-NTRU with distribution $\mathcal{D}^{\mathrm{d}}$. The distribution $\mathcal{D}^{\mathrm{s}}$ can be chosen from a large class of distributions (it only has to be bounded and to have an invertible denominator, as per Definition 5.1 below) and the distribution $\mathcal{D}^{\mathrm{d}}$ is a function of $\mathcal{D}^{\mathrm{s}}$. Moreover, we show that if the distribution $\mathcal{D}^{\mathrm{s}}$ enjoys an NTRU setup, then so does $\mathcal{D}^{\mathrm{d}}$.

5.1 Choice of the distributions

We start by describing a property of distributions that we will need for our search to decision reduction. We also describe the distribution $\mathcal{D}^{\mathrm{d}}$ as a function of $\mathcal{D}^{\mathrm{s}}$, and explain how one can sample h with a trapdoor (f, g) from $\mathcal{D}^{\mathrm{d}}$, provided there is an efficient algorithm doing it for $\mathcal{D}^{\mathrm{s}}$.

Definition 5.1 (Well-behaved elements and distributions). *Let $q \geqslant 2$ be an integer and $B > 1$ be a real number. An element $h \in R_q$ is said to be B-well-behaved if there exists $f, g \in R$ such that $gh = f \bmod q$; $\langle f \rangle + \langle g \rangle + \langle q \rangle = R$; and for all $1 \leqslant i \leqslant d$ we have $1/B \leqslant |\sigma_i(f)|, |\sigma_i(g)| \leqslant B$.*

A distribution $\mathcal{D}$ over R_q is said to be (B, ε)-well-behaved for some $\varepsilon \geqslant 0$ if the probability that $h \leftarrow \mathcal{D}$ is B-well-behaved is $\geqslant 1 - \varepsilon$.

Observe that any $(B, 0)$-well-behaved distribution over R_q is a distribution over (γ, q)-NTRU instances, where $\gamma = \sqrt{q}/(B\sqrt{d})$. Observe also that the condition $\langle f \rangle + \langle g \rangle + \langle q \rangle = R$ is equivalent to asking that g is invertible modulo q. Indeed, since $gh = f \bmod q$, then any prime factor dividing both $\langle g \rangle$ and $\langle q \rangle$ would also be a prime factor of $\langle f \rangle$, contradicting the coprimality condition. Let us now define a randomized mapping ϕ_B over R_q.

Definition 5.2 (Function ϕ_B). *Let $q \geqslant 2$ and $B > 1$. We define the randomized mapping ϕ_B over R_q as follows*

$$\begin{aligned} \phi_B : \quad & R_q \to R_q \\ & h \mapsto xh + y \bmod q \quad \text{where } x, y \leftarrow D_{R, 2Bd\delta_K}. \end{aligned}$$

We extend ϕ_B to distributions over R_q: for a distribution $\mathcal{D}$, we let $\phi_B(\mathcal{D})$ be the distribution over R_q obtained by sampling $h \leftarrow \mathcal{D}$ and then outputting $\phi_B(h)$.

Finally, we show that if $\mathcal{D}$ enjoys an NTRU setup, then so does $\phi_B(\mathcal{D})$.

Lemma 5.3. *Let $B \geqslant 1$, $q \geqslant 2$, $\gamma > 0$ and $\mathcal{D}$ a distribution over (γ, q)-NTRU instances. If there exists a $(\mathcal{D}, \gamma, q)$-NTRU setup, then there exists a $(\mathcal{D}', \gamma', q)$-NTRU setup where $\mathcal{D}'$ is a distribution over R_q such that $\mathcal{D}' \approx_{2^{-\Omega(d)}} \phi_B(\mathcal{D})$ and $\gamma' = \gamma/(4Bd^{1.5}\delta_K)$.*

Proof. Let $\mathcal{A}$ be a ppt algorithm (with respect to $\log q$ and $\log \Delta_K$) sampling triples $(h, f, g) \in R_q \times R^2$ such that the marginal distribution of h is $\mathcal{D}$, $(f, g) \neq (0, 0)$, $\|f\|, \|g\| \leqslant \sqrt{q}/\gamma$ and $g \cdot h = f \bmod q$.

We consider the following algorithm $\mathcal{B}$:

- run $\mathcal{A}$; let (h, f, g) be the output;
- use the algorithm from Lemma 2.2 with parameters $\sigma = 2Bd\delta_K$ and $\mathbf{c} = \mathbf{0}$ to sample x and y (using the basis $(r_1, \ldots, r_d)$ of R);
- return $(h', f', g') = (xh + y, xf + yg, g)$.

Note that $\mathcal{B}$ is ppt and that (f', g') is non-zero and satisfies $g' \cdot h' = f' \bmod q$. By Lemma 2.2, we also have

$$\|f'\| \leqslant 2Bd^{1.5}\delta_K \cdot (\|f\| + \|g\|) \leqslant 4Bd^{1.5}\delta_K \cdot \frac{\sqrt{q}}{\gamma}.$$

Finally, as the residual distribution of h is $\mathcal{D}$, Lemma 2.2 also implies that the residual distribution of h' is within statistical distance $2^{-\Omega(d)}$ from $\phi_B(\mathcal{D})$. □

We can now state the main result of this section: a reduction from NTRU_{mod} to dNTRU, for well-chosen distributions. This theorem follows from Lemmas 2.5, 5.6 and 5.7, which are stated and proved in the following subsections. The proof of Theorem 5.4 is provided in Appendix D.2.

Theorem 5.4. *Let $q \geqslant 2$, $B \in (1, q]$, $\varepsilon \geqslant 0$ and $\mathcal{D}^{\mathrm{s}}$ be a (B, ε)-well-behaved distribution over R_q. Assume that $\log q, \log \Delta_K, \log \|\Phi\| \leqslant 2^{o(d)}$ (recall that Φ is a defining polynomial of K). Define $\gamma' := \frac{\sqrt{q}}{4B^2d^2\delta_K}$ and assume that $\gamma' \geqslant 1$. Let $\mathcal{A}$ be an algorithm solving $(\phi_B(\mathcal{D}^{\mathrm{s}}), \gamma', q)$-dNTRU with advantage $\mathrm{Adv}(\mathcal{A}) \geqslant 2^{-o(d)}$. Then, there exists an algorithm $\mathcal{B}$ solving $(\mathcal{D}^{\mathrm{s}}, \gamma, q)$-$\text{NTRU}_{\text{mod}}$ with $\gamma = \sqrt{q}/(B\sqrt{d})$ and advantage $\mathrm{Adv}(\mathcal{B}) \geqslant (\mathrm{Adv}(\mathcal{A}) - 2\varepsilon)/4$. Algorithm $\mathcal{B}$ is ppt (with respect to $\log q$, $\log \Delta_K$, $\log \|\Phi\|$ and $\mathrm{Adv}(\mathcal{A})^{-1}$) and makes (possibly that many) oracle queries to $\mathcal{A}$.*

Observe that up to polynomial factors depending on the number field K, we have $\gamma \approx \sqrt{q}/B$ and $\gamma' \approx \sqrt{q}/B^2$. Said differently, the Euclidean norm of the short trapdoor is squared when we go from $\mathcal{D}^{\mathrm{s}}$ (which has short trapdoors of size roughly B) to $\phi_B(\mathcal{D}^{\mathrm{s}})$ (which has short trapdoors of size roughly B^2). Hence, one should consider $B \leqslant q^{1/4}$ for the dNTRU instances to have short trapdoors of norm $\geqslant \sqrt{q}$.

5.2 Creating new NTRU instances

In this section, we give a lemma which will allow us to rerandomize an NTRU instance h so that the distribution of the new NTRU instance depends on $c_1\sigma_1(f) + c_2\sigma_1(g)$ for some parameters c_1 and c_2 that we can customize. This lemma will be used to prove Lemma 5.7, in the next subsection.

Lemma 5.5. *Let $(f, g) \in R^2 \backslash \{(0, 0)\}$ and $I = \langle f \rangle + \langle g \rangle$. Let $c_1, c_2 \in \sigma_1(K_{\mathbb{R}})$ (which is either $\mathbb{R}$ or $\mathbb{C}$), $s_0 > 0$ and $s \geqslant \sqrt{d}\delta_K \cdot (\|f\| + \|g\|)$.*

Given $t \in \sigma_1(K_{\mathbb{R}})$, we define $\psi(t) \in K_{\mathbb{R}}$ as $(t, 0, \ldots, 0)^T \in K_{\mathbb{R}}$ if σ_1 is a real embedding and as $(t/\sqrt{2}, \overline{t}/\sqrt{2}, 0, \ldots, 0)^T \in K_{\mathbb{R}}$ if σ_1 is a complex embedding with $\sigma_2 = \overline{\sigma_1}$.[2]

Let $\mathcal{D}$ be the output distribution of the following algorithm:

- *sample $c_0 \leftarrow D_{\sigma_1(K_{\mathbb{R}}), s_0, 0}$;*
- *sample $x \leftarrow D_{R, s, \psi(c_0 \cdot c_1)}$ and $y \leftarrow D_{R, s, \psi(c_0 \cdot c_2)}$;*
- *return $x \cdot f + y \cdot g \in I$.*

[2] The scaling by a factor $1/\sqrt{2}$ in the complex case ensures that the norm of $\psi(t)$ is still equal to $|t|$, which allows simpler expressions.

Then it holds that $\mathcal{D} \approx_{2^{-\Omega(d)}} D_{I,\mathbf{S},0}$*, where* $\mathbf{S}$ *is a diagonal matrix with*

$$S_{11} = \sqrt{s_0^2 \cdot |c_1\sigma_1(f) + c_2\sigma_1(g)|^2 + s^2 \cdot (|\sigma_1(f)|^2 + |\sigma_1(g)|^2)}$$
$$S_{22} = \begin{cases} S_{11} & \text{if } \sigma_1 \text{ is a complex embedding} \\ s \cdot \sqrt{|\sigma_2(f)|^2 + |\sigma_2(g)|^2} & \text{if } \sigma_1 \text{ is a real embedding} \end{cases}$$
$$S_{ii} = s \cdot \sqrt{|\sigma_i(f)|^2 + |\sigma_i(g)|^2} \qquad \text{for } \ i \geqslant 3.$$

The above can be obtained by combining the convolution result of [Pei10, Th. 3.1] and the discrete Gaussian leftover hash lemma from [LSS14, Th. 5.1]. Unfortunately, the statements of [Pei10, Th. 3.1] and [LSS14, Th. 5.1] do not exactly match what we need (in particular, non-zero centers are not considered in [LSS14, Th. 5.1] and the convolution result of [Pei10, Th. 3.1] does not consider c_0 being sampled from a smaller space and extended with zeros). In Appendix D.1, we prove some slight variants of these results, in order to prove Lemma 5.5.

Observe that by taking $s = 2Bd\delta_K$ and $c_1 = c_2 = 0$, then the distribution of $x \cdot f + y \cdot g$ is exactly the distribution of the numerator of $\phi_B(h)$, over the randomness of ϕ_B (i.e., when h, f and g are fixed). Note that for Lemma 5.5 to be applicable, we need $s = 2Bd\delta_K \geqslant \sqrt{d}\delta_K \cdot (\|f\| + \|g\|)$, which holds true if $\|f\|_\infty, \|g\|_\infty \leqslant B$. This is the source of the 'standard deviation squaring' in Theorem 5.4. Finally, note that by using the lemma multiple times with the same h, we obtain tuple NTRU instances (as defined in Definition 3.1), implying that the dNTRU and NTRU_{vec} problem variants reduce to their tuple counterparts (under proper parametrization).

5.3 Using the OHCP framework

We now prove two lemmas for the core of the proof of Theorem 5.4. Lemma 5.6 essentially states that when sampling h from $\mathcal{D}^s$, then one should get a "good" h with non-negligible probability. Lemma 5.7 then shows that when h is "good", it is possible to recover a very accurate approximation of $\sigma_1(h_K)$ using the dNTRU oracle. Combining these two lemmas with Lemma 2.5 (which states that one can recover an element $x \in K$ exactly from a sufficiently good approximation of $\sigma_1(x)$) then yields Theorem 5.4 (whose proof is provided in Appendix D.2).

Lemma 5.6. *Let* $q \geqslant 2$*,* $B \in (1, q]$*,* $\varepsilon \geqslant 0$ *and* $\mathcal{D}^s$ *be a* (B, ε)*-well-behaved distribution over* R_q*. Let* $\mathcal{A}$ *be an algorithm solving* $(\phi_B(\mathcal{D}^s), \gamma, q)$-dNTRU *for some* $\gamma \geqslant 1$*. Then, there exists a set* $H \subset R_q$ *such that every* h *in* H *is* B*-well-behaved;* $\Pr_{h \leftarrow \mathcal{D}^s}(h \in H) \geqslant \text{Adv}(\mathcal{A})/2 - \varepsilon$*; and for all* $h \in H$

$$\left| \Pr\left(\mathcal{A}(\phi_B(h)) = 1\right) - \Pr\left(\mathcal{A}(u) = 1\right) \right| \geqslant \text{Adv}(\mathcal{A})/2,$$

where the probabilities are taken over the internal randomness of $\mathcal{A}$*, the randomness of* ϕ_B *and the random choice of* $u \leftarrow U(R_q)$ *(but not over the choice of* h*).*

Proof. There exists $H_0 \subset R_q$ of weight $\geqslant \mathrm{Adv}(\mathcal{A})/2$ under $\mathcal{D}^{\mathrm{s}}$ such that for all $h \in H_0$, the advantage of $\mathcal{A}$ on $\phi_B(h)$ is at least $\mathrm{Adv}(\mathcal{A})/2$. We define H as the subset of the h's in H_0 that are B-well-behaved. The result follows from the definition of (B, ε)-well-behavedness and the union bound. □

Lemma 5.7. *Let $q \geqslant 2$, $B \in (1, q]$, $\varepsilon \geqslant 0$ and $\mathcal{D}^{\mathrm{s}}$ be a (B, ε)-well-behaved distribution over R_q. Let $\mathcal{D}^{\mathrm{d}} = \phi_B(\mathcal{D}^{\mathrm{s}})$. Let $\mathcal{A}$ and H as in Lemma 5.6. Assume that $\mathrm{Adv}(\mathcal{A})^{-1}, \log q, \log \Delta_K \leqslant 2^{o(d)}$. Then, there exists a probabilistic algorithm $\mathcal{B}$ that, given an integer $\ell \leqslant 2^{o(d)}$ and any $h \in H$, recovers $\sigma_1(h_K)$ with ℓ bits of absolute precision[3] with probability $\geqslant 1 - 2^{-\Omega(d)}$ (where h_K is defined as in Lemma 3.5). Moreover, algorithm $\mathcal{B}$ runs in time polynomial in $\ell, \mathrm{Adv}(\mathcal{A})^{-1}, \log q$ and $\log \Delta_K$ and makes (possibly that many) oracle queries to $\mathcal{A}$.*

Proof. In order to prove the lemma, we will express our problem as an instance of the Oracle Hidden Center Problem (see Definition 2.9) and then use Proposition 2.10 to conclude.

Let $h \in H$ be fixed once and for all, and given to $\mathcal{B}$. Let us also fix some (unknown) $(f, g) \in R^2$ such that $g \cdot h = f \bmod q$; g is invertible modulo q; and $|\sigma_i(f)|, |\sigma_i(g)| \in [1/B, B]$ for all embeddings σ_i (we know that such f and g exist since h is B-well-behaved by definition of H). We write $I = \langle f \rangle + \langle g \rangle$, which is also fixed once and for all (and is coprime to $\langle q \rangle$).

Let $k = 1$ if σ_1 is a real embedding and $k = 2$ if σ_1 is a complex embedding. In the following, we will identify $\mathbb{R}^k$ with $\sigma_1(K_{\mathbb{R}})$. Note that in both cases, the Euclidean norm of a vector in $\mathbb{R}^k$ corresponds to the absolute value of the element seen in $\mathbb{R}$ or $\mathbb{C}$.

In order to fit the OHCP framework, we need to describe a randomized oracle $\mathcal{O}$ that takes as input a pair $(z, t) \in \mathbb{R}^k \times \mathbb{R}^{\geqslant 0}$ and outputs 0 or 1 such that $\Pr_{\mathcal{O}}(\mathcal{O}(z, t) = 1) = P(t + \ln|z - \sigma_1(h_K)|)$, for some (unknown) function P (that may depend on h). In other words, we want that the acceptance probability of the oracle $\mathcal{O}$ only depends on $t + \ln|z - \sigma_1(h_K)|$ (when t and z vary).

We start by considering an oracle $\mathcal{O}^{\mathrm{ideal}}$ that we do not know how to implement efficiently, but which is more convenient for the analysis. We will later replace it by an oracle $\mathcal{O}^{\mathrm{approx}}$ that can be implemented efficiently and whose behavior is very close to the one of $\mathcal{O}^{\mathrm{ideal}}$. Oracle $\mathcal{O}^{\mathrm{ideal}}$ is as follows. On input $(z, t) \in \mathbb{R}^k \times \mathbb{R}^{\geqslant 0}$, it first samples $f' \leftarrow D_{I,\mathbf{S}}$, where $\mathbf{S}$ is a diagonal matrix with

$$S_{11} = \sqrt{\exp(t-d)^2|\sigma_1(f) - z\sigma_1(g)|^2 + 4B^2d^2\delta_K^2(|\sigma_1(f)|^2 + |\sigma_1(g)|^2)}$$

$$S_{22} = \begin{cases} S_{11} & \text{if } \sigma_1 \text{ is a complex embedding} \\ 2Bd\delta_K\sqrt{(|\sigma_2(f)|^2 + |\sigma_2(g)|^2)} & \text{if } \sigma_1 \text{ is a real embedding} \end{cases}$$

$$S_{ii} = 2Bd\delta_K\sqrt{(|\sigma_i(f)|^2 + |\sigma_i(g)|^2)} \qquad \text{if } i \geqslant 3.$$

The astute reader will observe that sampling such an f' may be difficult: this is why we will later introduce $\mathcal{O}^{\mathrm{approx}}$. Oracle $\mathcal{O}^{\mathrm{ideal}}$ then defines $h' = f'/g \bmod q$ (recall that g is invertible modulo q) and returns $\mathcal{A}(h')$.

[3] The term "absolute precision" refers here to $|\tilde{x} - x| \leqslant 2^{-\ell}$, as opposed to the "relative precision" which would be $\frac{|\tilde{x}-x|}{|x|} \leqslant 2^{-\ell}$.

Note that z and t only appear in S_{11} (and $S_{22} = S_{11}$ if σ_1 is a complex embedding). Since $|\sigma_1(f) - z\sigma_1(g)|/|\sigma_1(g)| = |\sigma_1(h_K) - z|$, we obtain that the success probability of the oracle depends only on $t + \ln|z - \sigma_1(h_K)|$ when t and z vary, as required (recall that h, f and g are fixed once and for all).

In Claim 5.8 below, we show that the oracle $\mathcal{O}^{\text{ideal}}$ satisfies all the desired conditions to be an OHCP oracle and the conditions of Proposition 2.10. This will imply that one can efficiently recover an approximation of $\sigma_1(h_K)$ by using the oracle $\mathcal{O}^{\text{ideal}}$ as a black box.

Claim 5.8. There exist a parameter $\kappa_0 = \text{poly}(\text{Adv}(\mathcal{A})^{-1}, \log q, \log \Delta_K)$ and an algorithm $\mathcal{B}'$ that takes as input any parameter $\kappa \geqslant \kappa_0$ and outputs $\widetilde{\sigma_1(h_K)} \in \sigma_1(K_{\mathbb{R}})$ such that $|\widetilde{\sigma_1(h_K)} - \sigma_1(h_K)| \leqslant B^2 \cdot \exp(-\kappa)$ with probability $\geqslant 1 - \exp(-\kappa)$. Algorithm $\mathcal{B}'$ runs in time $\text{poly}(\kappa)$ and makes (possibly that many) oracle queries to the OHCP oracle $\mathcal{O}^{\text{ideal}}$ described above.

The difficulty with algorithm $\mathcal{B}'$ from Claim 5.8 is that it makes oracle calls to $\mathcal{O}^{\text{ideal}}$, which we do not know how to run in polynomial time given only access to h and $\mathcal{A}$ (in order to run $\mathcal{O}^{\text{ideal}}$ efficiently, we would probably need to know f and g). To handle this difficulty, we describe another oracle $\mathcal{O}^{\text{approx}}$, whose behavior is very close to the one of $\mathcal{O}^{\text{ideal}}$, but which can be run efficiently.

On input $(z, t) \in \mathbb{R}^k \times \mathbb{R}^{\geqslant 0}$, the randomized oracle $\mathcal{O}^{\text{approx}}$ proceeds as follows. It first samples c_0 in $\mathbb{R}^k$ from the continuous Gaussian distribution $D_{\mathbb{R}^k, \exp(t-d), \mathbf{0}}$; it then defines $\mathbf{c}_1 = \psi(c_0) \in K_{\mathbb{R}}$ and $\mathbf{c}_2 = \psi(-c_0 \cdot z) \in K_{\mathbb{R}}$ (where ψ is as defined in Lemma 5.5); the oracle then samples $x \leftarrow \tilde{D}_{R, 2Bd \cdot \delta_K, \mathbf{c}_1}$ and $y \leftarrow \tilde{D}_{R, 2Bd \cdot \delta_K, \mathbf{c}_2}$ (see Lemma 2.2); finally, the oracle runs $\mathcal{A}$ on input $\hat{h} = x \cdot h + y \bmod q$, and outputs $\mathcal{A}(\hat{h})$.

Oracle $\mathcal{O}^{\text{approx}}$ can indeed be run in polynomial time from h. Let us now write $\hat{f} = x \cdot f + y \cdot g$, so that $\hat{h} = \hat{f}/g \bmod q$. Observe that $\Pr(\mathcal{O}^{\text{approx}}(z, t) = 1) = \Pr(\mathcal{A}(\hat{h}) = 1)$, and $\Pr(\mathcal{O}^{\text{ideal}}(z, t) = 1) = \Pr(\mathcal{A}(h') = 1)$, where $\hat{h}$ and h' are two random variables. So $|\Pr(\mathcal{O}^{\text{approx}}(z, t) = 1) - \Pr(\mathcal{O}^{\text{ideal}}(z, t) = 1)| \leqslant \text{dist}(\hat{h}, h')$. Since g is fixed, we have $\text{dist}(\hat{h}, h') = \text{dist}(\hat{f}, f')$, and we obtain that

$$|\Pr(\mathcal{O}^{\text{approx}}(z, t) = 1) - \Pr(\mathcal{O}^{\text{ideal}}(z, t) = 1)| \leqslant \text{dist}(\hat{f}, f') \leqslant 2^{-\Omega(d)}.$$

The last inequality comes from Lemma 5.5 and Lemma 2.2.

To conclude, algorithm $\mathcal{B}$ is obtained by taking algorithm $\mathcal{B}'$ of Claim 5.8, but replacing its oracle calls to $\mathcal{O}^{\text{ideal}}$ by oracle calls to $\mathcal{O}^{\text{approx}}$, and taking $\kappa = \max(\kappa_0, d, \ell + 2\ln(B))$. By assumption on $\log q, \text{Adv}(\mathcal{A}), \ell$ and $\log \Delta_K$, we know that $\kappa \leqslant 2^{o(d)}$ (recall that $B \leqslant q$), so that algorithm $\mathcal{B}$ makes at most $2^{o(d)}$ oracle calls to $\mathcal{O}^{\text{approx}}$. Hence, we obtain that

$$|\Pr(\mathcal{B} \text{ succeeds}) - \Pr(\mathcal{B}' \text{ succeeds})| \leqslant 2^{o(d)} \cdot 2^{-\Omega(d)} = 2^{-\Omega(d)}.$$

This completes the proof of Lemma 5.7. □

Proof (Claim 5.8). First, we need to check that the oracle $\mathcal{O}^{\text{ideal}}$ is a valid OHCP oracle. Let us write $z^* = \sigma_1(h_K)$. Since $\sigma_1(h_K) = \sigma_1(f)/\sigma_1(g)$, we know by

choice of f and g that $\|z^*\| \in [1/B^2, B^2]$. Hence, the oracle $\mathcal{O}^{\text{ideal}}$ and scale parameter $D = B^2$ form a valid instance of the $(\varepsilon, \delta, \beta)$-OHCP problem (cf Definition 2.9), for any $\varepsilon \in (0,1)$, any $\delta \in (0, 1/B^4]$ and any $\beta \geqslant 1$.

We will show below that for all $\kappa \geqslant \kappa_0$ with

$$\kappa_0 := \max\left(4\,\mathrm{Adv}(\mathcal{A})^{-1},\, 8d(1+\ln(q\Delta_K^{1/d})),\, 4\ln(B)\right),$$

the OHCP oracle satisfies the conditions of Proposition 2.10, with

$$p_\infty = \Pr_{u \leftarrow U(R_q)}(\mathcal{A}(u) = 1) \quad \text{and} \quad s^* = 0.$$

More formally, letting $p(t)$ denote $\Pr(\mathcal{O}^{\text{ideal}}(\mathbf{0}, t) = 1)$ as in Proposition 2.10, we prove that

1. $p(s^*) - p_\infty \geqslant 1/\kappa$;
2. $|p(t) - p_\infty| \leqslant 2\exp(-t/\kappa)$ for any $t \geqslant 0$;
3. for any $t_1, t_2 \geqslant 0$, it holds that $|p(t_1) - p(t_2)| \leqslant \kappa\sqrt{|t_1 - t_2|}$.

Using Proposition 2.10, we the conclude that there exists an algorithm $\mathcal{B}'$ solving the $(\exp(-\kappa), \exp(-\kappa), 1 + 1/\kappa)$-OHCP problem in time $\mathrm{poly}(\kappa)$ by making oracle calls to $\mathcal{O}^{\text{ideal}}$. Thanks to the condition $\kappa \geqslant 4\ln(B)$, it holds that $\exp(-\kappa) \leqslant 1/B^4$ is a valid choice of δ. Moreover, using the fact that $B \leqslant q$, we see that $\kappa_0 = \mathrm{poly}(\mathrm{Adv}(\mathcal{A})^{-1}, \log q, \log \Delta_K)$, which proves Claim 5.8. We now proceed to prove the three properties above.

Property 1. We want to show that $p(s^*)$ is very close to $\Pr(\mathcal{A}(\phi_B(h)) = 1)$, which will allow us to conclude with Lemma 5.6. Observe that by definition of the OHCP oracle $\mathcal{O}^{\text{ideal}}$, we know that $p(s^*) = \Pr(\mathcal{A}(h') = 1)$, where $h' = f'/g \bmod q$. So in order to bound the difference between $\Pr\left(\mathcal{A}(\phi_B(h) = 1\right)$ and $p(s^*)$, it suffices to bound the statistical distance between the two random variables $\phi_B(h)$ and h', which is equivalent to bounding $\mathrm{dist}(g \cdot \phi_B(h), f')$ (i.e., it suffices to consider the numerator since the denominator is g in both cases).

Using Lemma 5.5 with $c_1 = c_2 = 0$ and $s = 2Bd\delta_K$, we know that the distribution of $g \cdot \phi_B(h)$ is within $2^{-\Omega(d)}$ statistical distance from $D_{I,\mathbf{S}_2,\mathbf{0}}$, where $\mathbf{S}_2$ is a diagonal matrix with i-th diagonal entry equal to $2Bd\delta_K \cdot \sqrt{|\sigma_i(f)|^2 + |\sigma_i(g)|^2}$. Moreover, by definition of $\mathcal{O}^{\text{ideal}}$, the distribution of f' is $D_{I,\mathbf{S}_1,\mathbf{0}}$, where $\mathbf{S}_1$ is identical to $\mathbf{S}_2$, except for first diagonal coefficient (or first two diagonal coefficients if σ_1 is complex), which is equal to

$$\sqrt{(2Bd\delta_K)^2(|\sigma_1(f)|^2 + |\sigma_1(g)|^2) + \exp(-2d) \cdot |\sigma_1(f)|^2}.$$

We now apply Lemma 2.3 to show that these two Gaussian distributions are statistically close. We first check that $\eta_{1/2}(\mathbf{S}_i^{-1} I) \leqslant 1/2$, for $i \in \{1, 2\}$. We know from Equation (2.1) that

$$\begin{aligned}\eta_{1/2}(\mathbf{S}_i^{-1} I) &\leqslant \sqrt{\frac{\ln(2d(1+2))}{\pi}} \cdot \lambda_d(\mathbf{S}_i^{-1} I) \\ &\leqslant \sqrt{d} \cdot \lambda_d(\mathbf{S}_i^{-1} I)\end{aligned}$$

Recall that $I = \langle f \rangle + \langle g \rangle$, so that $f \in I$. Hence, we know that the $\mathbf{S}_i^{-1} \cdot f \cdot r_j$'s are linearly independent vectors of $\mathbf{S}_i^{-1} \cdot I$ (recall that the r_j's form a basis of R). For every j, it holds that $\|\mathbf{S}_i^{-1} \cdot f \cdot r_j\| \leqslant \delta_K \cdot \|\mathbf{S}_i^{-1} \cdot f\| \leqslant \delta_K \cdot \sqrt{d}/(2Bd\delta_K)$ (since every diagonal coefficient of $\mathbf{S}_i$ is no smaller than the corresponding coefficient of f multiplied by $2Bd\delta_K$). Hence, we conclude that $\lambda_d(\mathbf{S}_i^{-1} I) \leqslant 1/(2\sqrt{d})$ and that $\eta_{1/2}(\mathbf{S}_i^{-1} I) \leqslant 1/2$, as desired. We can apply Lemma 2.3 and we obtain that

$$\operatorname{dist}(D_{I,\mathbf{S}_1,\mathbf{0}}, D_{I,\mathbf{S}_2,\mathbf{0}}) \leqslant 4\sqrt{d} \cdot \sqrt{\left\|\left\|\mathbf{S}_2^{-1}\mathbf{S}_1 - \mathbf{I}_d\right\|\right\|}.$$

The matrix $\mathbf{S}_2^{-1}\mathbf{S}_1 - \mathbf{I}_d$ is zero, except for the top-left coefficient (or for the first two top-left coefficients if σ_1 is a complex embedding), which is equal to $\sqrt{1+\eta} - 1$ where $\eta = \exp(-2d) \cdot |\sigma_1(f)|^2/((2Bd\delta_K)^2 \cdot (|\sigma_1(f)|^2 + |\sigma_1(g)|^2))$. Since $\eta \leqslant \exp(-2d)$, we conclude that $|\sqrt{1+\eta} - 1| \leqslant \exp(-2d)$, and so $\left\|\left\|\mathbf{S}_2^{-1}\mathbf{S}_1 - \mathbf{I}_d\right\|\right\| \leqslant \exp(-2d)$ (or $\leqslant 2\exp(-2d)$ in case we had two non-zero coefficients). We finally obtain that $D_{I,\mathbf{S}_1,\mathbf{0}} \approx_{2^{-\Omega(d)}} D_{I,\mathbf{S}_2,\mathbf{0}}$, which in turn shows that

$$|p(s^*) - \Pr\left(\mathcal{A}(\phi_B(h) = 1\right)| \leqslant 2^{-\Omega(d)}.$$

Finally, since $h \in H$, we know from Lemma 5.6 that $|\Pr(\mathcal{A}(\phi_B(h) = 1) - p_\infty| \geqslant \operatorname{Adv}(\mathcal{A})/2$. Wlog, we can assume that $\Pr(\mathcal{A}(\phi_B(h) = 1) - p_\infty \geqslant 0$ (otherwise we can simply consider $\mathcal{A}' = 1 - \mathcal{A}$), from which we obtain that

$$p(s^*) - p_\infty \geqslant \operatorname{Adv}(\mathcal{A})/2 - 2^{-\Omega(d)} \geqslant \operatorname{Adv}(\mathcal{A})/4,$$

where the last inequality holds asymptotically when d tends to infinity, since we assumed that $1/\operatorname{Adv}(\mathcal{A}) \leqslant 2^{o(d)}$. By choice of κ, this implies that $p(s^*) - p_\infty \geqslant 1/\kappa$.

Property 2. To prove this second property, we want to show that when t is sufficiently large, then the distribution of $f' \bmod q$ (where f' is implicitly computed by the oracle $\mathcal{O}^{\text{ideal}}$ as defined above) is statistically close to uniform in $R \bmod qR$. Recall that the support of f' is I, which may be a strict subset of R. However, we know that $I = \langle f \rangle + \langle g \rangle$ is coprime to $\langle q \rangle$. So if $\widetilde{f} \in I$ is uniform in $I/(qI)$, then $\widetilde{f} + qR$ is a uniform class of $R/(qR)$. Hence, it suffices to show that f' is statistically close to uniform in $I/(qI)$.

Recall that f' is sampled from the distribution $D_{I,\mathbf{S}}$, where $\mathbf{S}$ is a diagonal matrix with positive diagonal coefficients, with $S_{11} \geqslant \exp(t-d) \cdot |\sigma_1(f)|$ (we consider $z = 0$ here) and $S_{ii} \geqslant |\sigma_i(f)|$ for $i \geqslant 2$. Taking the product, we conclude that $\prod_i S_{ii} \geqslant \exp(t-d) \cdot \mathcal{N}(f)$. Let us call c the quantity $c = (\exp(t-d)\mathcal{N}(f)/(\mathcal{N}(qI) \cdot \Delta_K))^{1/d}$. Using Lemma 2.8, we know that when t is sufficiently large so that $c \geqslant 1$, then it holds that $1 \geqslant \eta_\varepsilon(\mathbf{S}^{-1} \cdot (qI))$ for $\varepsilon = \exp(-c^2 d)$. Moreover, applying Lemma 2.4 to $L_1 = \mathbf{S}^{-1} \cdot (qI)$ and $L_2 = \mathbf{S}^{-1} \cdot I$, we see that

$$\operatorname{dist}\left(D_{\mathbf{S}^{-1} \cdot I,1} \bmod \mathbf{S}^{-1} \cdot (qI), U(\mathbf{S}^{-1} \cdot I \bmod \mathbf{S}^{-1} \cdot (qI))\right) \leqslant 2\exp(-c^2 d).$$

Multiplying the outputs of these two distributions by $\mathbf{S}$, we finally obtain

$$\operatorname{dist}\left(D_{I,\mathbf{S}} \bmod qI, U(I \bmod qI)\right) \leqslant 2\exp(-c^2 d).$$

Using the fact that $c^2 \geqslant c$ (as $c \geqslant 1$), that $\exp(x) \geqslant x$ for all $x \in \mathbb{R}$, and that $\mathcal{N}(I) \leqslant \mathcal{N}(f)$, we obtain the upper bound

$$\begin{aligned} 2\exp(-c^2 d) \leqslant 2\exp(-cd) &\leqslant 2\exp\big(-e^{(t-d-\ln(q^d\Delta_K))/d} \cdot d\big) \\ &\leqslant 2\exp\big(-(t - d(1+\ln(q\Delta_K^{1/d})))\big). \end{aligned}$$

If $t \geqslant 2d(1+\ln(q\Delta_K^{1/d}))$, then $(t-d(1+\ln(q\Delta_K^{1/d})) \geqslant t/2$ and $c \geqslant 1$, which implies that

$$|p(t) - p_\infty| \leqslant 2\exp(-t/2) \leqslant 2\exp(-t/\kappa).$$

For smaller t, note that $t \leqslant \kappa/2$. In this case, the upper bound $2\exp(-t/\kappa)$ is at least 1, and so the property is also satisfied.

Property 3. Let us fix some $t_1 \geqslant t_2 \geqslant 0$. We want to show that $|p(t_1) - p(t_2)| \leqslant \kappa \cdot \sqrt{|t_1 - t_2|}$. Observe first that since p takes values in $[0,1]$ and $\kappa \geqslant 1$, then the condition is always satisfied when $|t_1 - t_2| \geqslant 1$. We will hence assume wlog that $0 \leqslant t_1 - t_2 \leqslant 1$.

We know from the definition of $\mathcal{O}^{\text{ideal}}$ that $|p(t_1) - p(t_2)| \leqslant \operatorname{dist}(D_{I,\mathbf{S}_1}, D_{I,\mathbf{S}_2})$, where $\mathbf{S}_1$ and $\mathbf{S}_2$ are diagonal and equal, except for their for top-left coefficient (or two top-left coefficients if σ_1 is a complex embedding):

$$(S_1)_{11} = \sqrt{c + (\exp(t_1 - d)|\sigma_1(f)|)^2} \quad \text{and} \quad (S_2)_{11} = \sqrt{c + (\exp(t_2 - d)|\sigma_1(f)|)^2},$$

for some $c \geqslant 0$. As when proving Property 1, one can check that $\eta_{1/2}(\mathbf{S}_1^{-1} I)$, $\eta_{1/2}(\mathbf{S}_2^{-1} I) \leqslant 1/2$. Therefore, we can apply Lemma 2.3 to obtain that

$$\operatorname{dist}\left(D_{I,\mathbf{S}_1}, D_{I,\mathbf{S}_2}\right) \leqslant 4\sqrt{d} \cdot \sqrt{\left\|\left\|\mathbf{S}_2^{-1}\mathbf{S}_1 - \mathbf{I}_d\right\|\right\|}.$$

Once again, the matrix $\mathbf{S}_2^{-1}\mathbf{S}_1 - \mathbf{I}_d$ is zero, except for its top-left coefficient (or two top-left coefficients) which is equal to

$$\sqrt{\frac{c + (\exp(t_1 - d)|\sigma_1(f)|)^2}{c + (\exp(t_2 - d)|\sigma_1(f)|)^2}} - 1 \leqslant \sqrt{\frac{(\exp(t_1 - d)|\sigma_1(f)|)^2}{(\exp(t_2 - d)|\sigma_1(f)|)^2}} - 1 = \exp(t_1 - t_2) - 1.$$

The first inequality comes from the fact that $t_1 \geqslant t_2$ (and c and $(\exp(t_2 - d)|\sigma_1(f)|)^2$ are non-negative). Finally, since $0 \leqslant t_1 - t_2 \leqslant 1$, we conclude that $\exp(t_1 - t_2) - 1 \leqslant 2|t_1 - t_2|$. This in turns implies that $|p(t_1) - p(t_2)| \leqslant 8\sqrt{d}\sqrt{|t_1 - t_2|} \leqslant \kappa\sqrt{|t_1 - t_2|}$, as desired. □

Acknowledgments. The authors thank Koen de Boer, Léo Ducas, Joël Felderhoff, Guillaume Hanrot, Miruna Rosca and Adeline Roux-Langlois for insightful discussions. The first author was supported in part by CyberSecurity Research Flanders with reference number VR20- 192203 and by the Research Council KU Leuven grant C14/18/067 on Cryptanalysis of Post-quantum Cryptography. The second author was supported in part by European Union Horizon 2020 Research and Innovation Program Grant 780701 and BPI-France in the context of the national project RISQ (P141580).

References

ABD16. M. R. Albrecht, S. Bai, and L. Ducas. A subfield lattice attack on overstretched NTRU assumptions - cryptanalysis of some FHE and graded encoding schemes. In *CRYPTO*, 2016.

AD17. M. R. Albrecht and A. Deo. Large Modulus Ring-LWE $\geqslant$ Module-LWE. In *ASIACRYPT*, 2017.

AGHS13. S. Agrawal, C. Gentry, S. Halevi, and A. Sahai. Discrete gaussian leftover hash lemma over infinite domains. In *ASIACRYPT*, 2013.

BBC+20. D. J. Bernstein, Brumley B. B., M.-S. Chen, C. Chuengsatiansup, T. Lange, A. Marotzke, B.-Y. Peng, N. Tuveri, C. van Vredendaal, and B.-Y. Yang. NTRU Prime round-3 candidate to the NIST post-quantum cryptography standardisation project, 2020. Available at https://ntruprime.cr.yp.to/.

Bel04a. K. Belabas. A relative van Hoeij algorithm over number fields. *J Symb Comput*, 37(5), 2004.

Bel04b. K. Belabas. Topics in computational algebraic number theory. *J. théorie des nombres de Bordeaux*, 16, 2004.

BS96. E. Bach and J. O. Shallit. *Algorithmic Number Theory: Efficient Algorithms*. MIT Press, 1996.

CDH+20. C. Chen, O. Danba, J. Hoffstein, A. Hülsing, J. Rijneveld, T. Saito, J. M. Schank, P. Schwabe, W. Whyte, K. Xagawa, T. Yamakawa, and Z. Zhang. NTRU round-3 candidate to the NIST post-quantum cryptography standardisation project, 2020. Available at https://ntru.org/.

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

CJL16. J. H. Cheon, J. Jeong, and C. Lee. An algorithm for NTRU problems and cryptanalysis of the GGH multilinear map without an encoding of zero. *LMS J Comput Math*, 19(A), 2016.

CL15. J. H. Cheon and C. Lee. Approximate algorithms on lattices with small determinant. *IACR ePrint 2015/461*, 2015.

Coh95. H. Cohen. *A Course in Computational Algebraic Number Theory*. Springer, 1995.

Coh00. H. Cohen. *Advanced topics in computational number theory*. Springer, 2000.

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.

FS10. C. Fieker and D. Stehlé. Short bases of lattices over number fields. In *ANTS*, 2010.

GGH13. S. Garg, C. Gentry, and S. Halevi. Candidate multilinear maps from ideal lattices. In *EUROCRYPT*, 2013.

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

HHP+03. J. Hoffstein, N. Howgrave-Graham, J. Pipher, J. H. Silverman, and W. Whyte. NTRUSIGN: digital signatures using the NTRU lattice. In *CT-RSA*, 2003.

HPS98. J. Hoffstein, J. Pipher, and J. H. Silverman. NTRU: a ring based public key cryptosystem. In *ANTS*, 1998.

KF15. P. Kirchner and P.-A. Fouque. An improved BKW algorithm for LWE with applications to cryptography and lattices. In *CRYPTO*, 2015.

KF17. P. Kirchner and P.-A. Fouque. Revisiting lattice attacks on overstretched NTRU parameters. In *EUROCRYPT*, 2017.
Kle00. P. N. Klein. Finding the closest lattice vector when it's unusually close. In *SODA*, 2000.
KLL84. R. Kannan, A. K. Lenstra, and L. Lovász. Polynomial factorization and nonrandomness of bits of algebraic and some transcendental numbers. In *STOC*, 1984.
LLL82. A. K. Lenstra, H. W. Lenstra, Jr., and L. Lovász. Factoring polynomials with rational coefficients. *Math Ann*, 1982.
LPR10. V. Lyubashevsky, C. Peikert, and O. Regev. On ideal lattices and learning with errors over rings. In *EUROCRYPT*, 2010.
LPSW19. C. Lee, A. Pellet-Mary, D. Stehlé, and A. Wallet. An LLL algorithm for module lattices. In *ASIACRYPT*, 2019.
LS15. A. Langlois and D. Stehlé. Worst-case to average-case reductions for module lattices. *Des Codes Cryptography*, 2015.
LSS14. A. Langlois, D. Stehlé, and R. Steinfeld. GGHLite: More efficient multilinear maps from ideal lattices. In *EUROCRYPT*, 2014.
LTV12. A. López-Alt, E. Tromer, and V. Vaikuntanathan. On-the-fly multiparty computation on the cloud via multikey fully homomorphic encryption. In *STOC*, 2012.
MG02. D. Micciancio and S. Goldwasser. *Complexity of lattice problems: a cryptographic perspective*. Kluwer Academic Press, 2002.
MR07. D. Micciancio and O. Regev. Worst-case to average-case reductions based on gaussian measures. *SIAM J. Comput.*, 37(1), 2007.
Pei10. C. Peikert. An efficient and parallel gaussian sampler for lattices. In *CRYPTO*, 2010.
Pei16. C. Peikert. A decade of lattice cryptography. *Found. Trends Theor. Comput. Sci.*, 10(4), 2016.
PHS19. A. Pellet-Mary, G. Hanrot, and D. Stehlé. Approx-SVP in ideal lattices with pre-processing. In *EUROCRYPT*, 2019.
PRS17. C. Peikert, O. Regev, and N. Stephens-Davidowitz. Pseudorandomness of ring-LWE for any ring and modulus. In *STOC*, 2017.
Reg09. O. Regev. On lattices, learning with errors, random linear codes, and cryptography. *J ACM*, 2009.
Rob97. F.-X. Roblot. *Algorithmes de factorisation dans les extensions relatives et applications de la conjecture de Stark à la construction des corps de classes de rayon*. PhD thesis, Université Bordeaux 1, 1997. Available at http://math.univ-lyon1.fr/~roblot/resources/these.pdf.
RSW18. M. Rosca, D. Stehlé, and A. Wallet. On the ring-LWE and polynomial-LWE problems. In *EUROCRYPT*, 2018.
Sch87. C.-P. Schnorr. A hierarchy of polynomial lattice basis reduction algorithms. *Theoretical Computer Science*, 53, 1987.
SMSV14. Saruchi, I. Morel, D. Stehlé, and G. Villard. LLL reducing with the most significant bits. In *ISSAC*, 2014.
SS11. D. Stehlé and R. Steinfeld. Making NTRU as secure as worst-case problems over ideal lattices. In *EUROCRYPT*, 2011.
SS13. D. Stehlé and R. Steinfeld. Making NTRUEncrypt and NTRUSign as secure as standard worst-case problems over ideal lattices. *IACR ePrint 2013/004*, 2013.
SSTX09. D. Stehlé, R. Steinfeld, K. Tanaka, and K. Xagawa. Efficient public key encryption based on ideal lattices. In *ASIACRYPT*, 2009.

Sut16. A. Sutherland. Lecture notes of Number Theory I, taught at MIT. Available at https://math.mit.edu/classes/18.785/2016fa/LectureNotes12.pdf, 2016.

WW18. Y. Wang and M. Wang. Provably secure NTRUEncrypt over any cyclotomic field. In *SAC*, 2018.

A Additional preliminaries and missing proofs from Section 2

This section contains some missing proofs from Section 2, together with some additional preliminaries that is used only for the appendices.

A.1 Additional preliminaries for the appendices

Properties of Gaussian distributions. When the standard deviation of the discrete Gaussian distribution is larger than the smoothing parameter, then it somehow behaves like a continuous discrete Gaussian distribution. This is formalized for different properties in the lemmas below.

Lemma A.1 (Proof of [MR07, Le. 4.4]). *Let $L \subset \mathbb{R}^n$ be a full rank lattice, $\mathbf{c} \in \mathbb{R}^n$ and $\sigma \geqslant \eta_\varepsilon(L)$ for some $\varepsilon > 0$. Then it holds that*

$$\rho_\sigma(L + \mathbf{c}) \in [1-\varepsilon, 1+\varepsilon] \cdot \frac{\sigma^n}{\det(L)}.$$

Lemma A.2 ([MR07, Le. 4.4]). *Let $L \subset \mathbb{R}^n$ be a full rank lattice, $\mathbf{c} \in \mathbb{R}^n$ and $\sigma \geqslant \eta_\varepsilon(L)$ for some $\varepsilon > 0$. Then it holds that*

$$\Pr_{\mathbf{v} \leftarrow D_{L,\sigma,\mathbf{c}}} \left(\|\mathbf{v} - \mathbf{c}\| > \sqrt{n} \cdot \sigma \right) \leqslant \frac{1+\varepsilon}{1-\varepsilon} \cdot 2^{-n}.$$

It is possible to efficiently sample elements from a distribution that is statistically close to a discrete Gaussian distribution over lattices. This can be done using an algorithm proposed by Klein [Kle00] and analyzed by Gentry, Peikert and Vaikuntanathan [GPV08], provided that the standard deviation is sufficiently large.

Lemma A.3 (Adapted from [GPV08, Th. 4.1]). *There exists a* ppt *algorithm that takes as input a basis $\mathbf{B} = (\mathbf{b}_1, \ldots, \mathbf{b}_n)$ of an n-dimensional lattice L, a real number $s > 0$ with $s \geqslant \sqrt{n} \cdot \max_i \|\widetilde{\mathbf{b}_i}\|$ and a center $\mathbf{c} \in \mathrm{Span}(L)$ and outputs a sample from a distribution $\widetilde{D}_{\mathbf{B},S,\mathbf{c}}$ such that $D_{L,S,\mathbf{c}} \approx_{2^{-n}} \widetilde{D}_{\mathbf{B},S,\mathbf{c}}$.*

Integer lattices of small volume. The following result, proved by Cheon and Lee in [CL15], states that for an integral lattice L of rank n with sufficiently small volume, then one can find, in polynomial time, significantly smaller non-zero vectors of L than one would get by using the LLL algorithm.

Lemma A.4 (Adapted from [CL15, Th. 2]). *Let $L \subseteq \mathbb{Z}^n$ be a full rank integral lattice. Then, there is a* ppt *algorithm computing $\mathbf{v} \in L \backslash \{\mathbf{0}\}$ such that*

$$\|\mathbf{v}\| \leqslant 2^{\sqrt{\log \det(L)}} \cdot \det(L)^{1/n}.$$

Proof. The algorithm takes as input a basis $\mathbf{B}$ of L and starts by computing its Hermite Normal Form $\mathbf{B}'$. It then selects the first m vectors of $\mathbf{B}'$ (with $m \leqslant n$ to be defined later), runs LLL on them and outputs the first vector $\mathbf{v} \neq \mathbf{0}$ of the LLL output basis. We have:

$$\|\mathbf{v}\| \leqslant 2^{\frac{m-1}{4}} \cdot |\det \mathbf{B}'|^{1/m} \leqslant 2^{\frac{m-1}{4}} \cdot \det(L)^{1/m}.$$

If $2\sqrt{\log \det(L)} \leqslant n$, then we set $m = \lceil 2\sqrt{\log \det(L)} \rceil$. Else we set $m = n$. In the first case, the above inequality leads to $\|\mathbf{v}\| \leqslant 2^{\sqrt{\log \det(L)}}$. In the second case, we have

$$\|\mathbf{v}\| \leqslant 2^{\frac{n-1}{4}} \cdot \det(L)^{1/n} \leqslant 2^{\sqrt{\log \det(L)}} \cdot \det(L)^{1/n}.$$

This completes the proof. □

Complex conjugation in a number field. For $x \in K_{\mathbb{R}}$, we define $\overline{x} \in K_{\mathbb{R}}$ as the element obtained by componentwise complex conjugation of the canonical embedding vector of x. We extend this notation to vectors and matrices over $K_{\mathbb{R}}$. We define $\overline{K}$ and $\overline{R}$ as the subsets of $K_{\mathbb{R}}$ obtained by applying complex conjugation to elements of K and R, respectively. We define $K_{\mathbb{R}}^+ = K_{\mathbb{R}}^\times \cap (\mathbb{R}_{>0})^d$ and, for $x \in K_{\mathbb{R}}^+$, we define $\sqrt{x} = \sigma^{-1}((\sqrt{\sigma_i(x)})_i)$.

Ideals versus rank-1 modules. We now observe that γ-HSVP for rank-1 modules is computationally equivalent to γ-id-HSVP. By mapping an ideal I to the rank-1 module $I \cdot (1, 0, \ldots, 0)^T$, we obtain a reduction from γ-id-HSVP to γ-HSVP for rank-1 modules. The following lemma handles the converse.

Lemma A.5. *Let $\gamma \geqslant 1$, $\ell \geqslant 1$ and $\varepsilon > 0$. Then there exists a reduction from γ-HSVP for rank-1 modules $M \subset K^\ell$ to $(1+\varepsilon)\gamma$-id-HSVP.*

Proof. The proof relies on the QR-factorization over $K_{\mathbb{R}}$. For any $\mathbf{B} \in K_{\mathbb{R}}^{\ell \times k}$ with $K_{\mathbb{R}}$-linearly independent columns, there exist $\mathbf{Q} \in K_{\mathbb{R}}^{\ell \times k}$ and $\mathbf{R} \in K_{\mathbb{R}}^{k \times k}$ such that $\mathbf{B} = \mathbf{QR}$, $\overline{\mathbf{Q}}^T\mathbf{Q} = \mathbf{I}_k$ and $\mathbf{R}$ is upper triangular with all diagonal coefficients in $K_{\mathbb{R}}^+$. This decomposition is unique and called the QR-factorization of $\mathbf{B}$. As implied by [LPSW19, Se. 2.3], if $\mathbf{B} \in K^{\ell \times k}$ has $K_{\mathbb{R}}$-linearly independent columns, then the coefficients of its Q-factor and R-factor may be computed to arbitrary precision in polynomial time.

Let $M \subset K^\ell$ be a rank-1 module represented by a pseudo-basis $I \cdot \mathbf{b}$ for some non-zero vector $\mathbf{b} \in K^\ell$ and non-zero fractional ideal I. The algorithm first computes the QR-factorization $\mathbf{b} = \mathbf{Q}r$, with $\mathbf{Q} \in K_{\mathbb{R}}^{\ell \times 1}$ and $r \in K_{\mathbb{R}}^+$. We have $\sigma_i(r) = \sqrt{\sum_{j \leqslant \ell} |\sigma_i(b_j)|^2}$, where $\mathbf{b} = (b_1, \ldots, b_\ell)^T$. The algorithm then calls the oracle on the rank-1 module $M' = \overline{\mathbf{Q}}^T M = r \cdot I \subset K_{\mathbb{R}}$. As left-multiplication by $\mathbf{Q}$ and $\overline{\mathbf{Q}}^T$ preserves the Euclidean norm, solving γ-id-HSVP for ideal M' suffices. The oracle finds $x \in M' \backslash \{\mathbf{0}\}$ such that $\|x\| \leqslant \gamma \cdot \sqrt{d} \cdot \det(M')^{1/d}$. The algorithm then outputs $\mathbf{Q}x \in M \backslash \{\mathbf{0}\}$, which satisfies:

$$\|\mathbf{Q}x\| = \|x\| \leqslant \gamma \cdot \sqrt{d} \cdot \det(M')^{1/d} = \gamma \cdot \sqrt{d} \cdot \det(M)^{1/d}.$$

In the above, we assumed exact computations over complex numbers and a γ-id-HSVP for ideals over $K_{\mathbb{R}}$. These precision issues can be handled using LLL-preprocessing and rounding techniques such as described in [SMSV14]. This allows to find $\mathbf{s} \in M \backslash \{\mathbf{0}\}$ such that $\|\mathbf{s}\| \leqslant (1+\varepsilon)\gamma \cdot \sqrt{d} \cdot \det(M)^{1/d}$, with a time overhead that is polynomial in $\log 1/\varepsilon$. □

A.2 Proof of Lemma 2.2

This lemma is obtained by combining Lemmas A.1, A.2 and A.3. The algorithm consists in running Klein's sampler, and restarting if the output vector $\mathbf{v}$ is at distance $\geqslant \sqrt{n} \cdot \sigma$ from $\mathbf{c}$ or if $\mathbf{v} = \mathbf{0}$. As the minimum over all bases of $\max_i \|\mathbf{b}_i\|$ is bounded from below by $\lambda_n(L)$, it holds that $\sigma \geqslant \sqrt{n}\lambda_n(L) \geqslant \eta_\varepsilon(L)$ for $\varepsilon = 2^{-\Omega(n)}$ (by (2.1)). The probability that $\|\mathbf{v} - \mathbf{c}\| \geqslant \sqrt{n}\sigma$ is hence bounded as $2^{-\Omega(n)}$, by Lemma A.2. By the triangular inequality applied to the statistical distance and Lemma A.1, we have:

$$\Pr(\mathbf{v} = \mathbf{0}) \leqslant \frac{1}{\rho_\sigma(L)} + 2^{-\Omega(n)} \leqslant 2\frac{\det(L)}{\sigma^n} + 2^{-\Omega(n)}.$$

Since $\det(L) \leqslant \lambda_n(L)^n \leqslant (\sigma/\sqrt{n})^n$, we conclude that $\Pr(\mathbf{v} = \mathbf{0}) \leqslant 2^{-\Omega(n)}$. Combining both results, we see that the probability to restart the algorithm is bounded from above by $2^{-\Omega(n)}$, implying that the algorithm runs in polynomial time. The claim follow from Lemma A.3. □

A.3 Proof of Lemma 2.3

As in [dBDPW20], it is convenient to use the Kullback-Leibler divergence. The Kullback-Leibler divergence of two random variables X, Y defined over the same set S and such that $\Pr(Y = s) > 0$ for all $s \in S$ is defined as

$$\mathrm{d}_{\mathrm{KL}}(X\|Y) = \sum_{s \in S} \Pr(X = s) \cdot \ln\left(\frac{\Pr(X = s)}{\Pr(Y = s)}\right).$$

The Kullback-Leibler divergence can be used to bound the statistical distance between X and Y, using Pinsker's inequality:

$$\mathrm{dist}(X, Y) \leqslant \sqrt{\frac{1}{2} \cdot \mathrm{d}_{\mathrm{KL}}(X\|Y)}.$$

In the following, we will compute an upper bound on $\mathrm{d}_{\mathrm{KL}}(D_{L,\mathbf{S}_1,\mathbf{c}_1}\|D_{L,\mathbf{S}_2,\mathbf{c}_2})$ (observe that both distributions are defined over the same set $S := L$ and any element of S has a non-zero weight for both).

In the first part of this proof, we assume that $\rho_{\mathbf{S}_2}(L - \mathbf{c}_2) \leqslant \rho_{\mathbf{S}_1}(L - \mathbf{c}_1)$. We will then make sure at the end of the proof that the lemma statement also holds when $\rho_{\mathbf{S}_2}(L - \mathbf{c}_2) \leqslant \rho_{\mathbf{S}_1}(L - \mathbf{c}_1)$ (observe that the upper bound given in the lemma is not symmetric in $\mathbf{S}_1$ and $\mathbf{S}_2$).

By definition of the Kullback-Leibler divergence, we have

$$\begin{aligned}\mathrm{d_{KL}}(D_{L,\mathbf{S}_1,\mathbf{c}_1}\|D_{L,\mathbf{S}_2,\mathbf{c}_2}) &= \mathbb{E}_{\mathbf{v}}\left(\ln\frac{\rho_{\mathbf{S}_1}(\mathbf{v}-\mathbf{c}_1)\cdot\rho_{\mathbf{S}_2}(L-\mathbf{c}_2)}{\rho_{\mathbf{S}_2}(\mathbf{v}-\mathbf{c}_2)\cdot\rho_{\mathbf{S}_1}(L-\mathbf{c}_1)}\right)\\ &= \ln\left(\frac{\rho_{\mathbf{S}_2}(L-\mathbf{c}_2)}{\rho_{\mathbf{S}_1}(L-\mathbf{c}_1)}\right)\\ &\quad+\pi\cdot\mathbb{E}_{\mathbf{v}}(\|\mathbf{S}_2^{-1}(\mathbf{v}-\mathbf{c}_2)\|^2-\|\mathbf{S}_1^{-1}(\mathbf{v}-\mathbf{c}_1)\|^2),\end{aligned}$$

where the expectancies are for $\mathbf{v}\leftarrow D_{L,\mathbf{S}_1,\mathbf{c}_1}$. As we assumed that $\rho_{\mathbf{S}_2}(L-\mathbf{c}_2)\leqslant\rho_{\mathbf{S}_1}(L-\mathbf{c}_1)$, the first term of the sum above is $\leqslant 0$. Let us now consider the second term. We start by fixing some $\mathbf{v}\in L$. In order to simplify the equation below, we will write $\widetilde{\mathbf{v}}=\mathbf{S}_1^{-1}(\mathbf{v}-\mathbf{c}_1)$ and $\widetilde{\mathbf{c}}=\mathbf{S}_2^{-1}(\mathbf{c}_1-\mathbf{c}_2)$. With these notations, we have $\mathbf{S}_2^{-1}(\mathbf{v}-\mathbf{c}_2)=\mathbf{S}_2^{-1}\mathbf{S}_1\widetilde{\mathbf{v}}+\widetilde{\mathbf{c}}$. Then, we have

$$\begin{aligned}\|\mathbf{S}_2^{-1}(\mathbf{v}-\mathbf{c}_2)\|^2-\|\mathbf{S}_1^{-1}(\mathbf{v}-\mathbf{c}_1)\|^2 &= \|\mathbf{S}_2^{-1}\mathbf{S}_1\widetilde{\mathbf{v}}+\widetilde{\mathbf{c}}\|^2-\|\widetilde{\mathbf{v}}\|^2\\ &\leqslant\Big(\|\mathbf{S}_2^{-1}\mathbf{S}_1\widetilde{\mathbf{v}}+\widetilde{\mathbf{c}}\|+\|\widetilde{\mathbf{v}}\|\Big)\cdot\Big|\|\mathbf{S}_2^{-1}S_1\widetilde{\mathbf{v}}+\widetilde{\mathbf{c}}\|-\|\widetilde{\mathbf{v}}\|\Big|\\ &\leqslant\Big((\|\mathbf{S}_2^{-1}\mathbf{S}_1\|+1)\cdot\|\widetilde{\mathbf{v}}\|+\|\widetilde{\mathbf{c}}\|\Big)\cdot\|\mathbf{S}_2^{-1}\mathbf{S}_1\widetilde{\mathbf{v}}+\widetilde{\mathbf{c}}-\widetilde{\mathbf{v}}\|\\ &\leqslant\Big((\|\mathbf{S}_2^{-1}\mathbf{S}_1\|+1)\cdot\|\widetilde{\mathbf{v}}\|+\|\widetilde{\mathbf{c}}\|\Big)\cdot\Big(\|\mathbf{S}_2^{-1}\mathbf{S}_1-\mathbf{I}_n\|\cdot\|\widetilde{\mathbf{v}}\|+\|\widetilde{\mathbf{c}}\|\Big)\\ &\leqslant\|\widetilde{\mathbf{v}}\|^2\cdot(\|\mathbf{S}_2^{-1}\mathbf{S}_1\|+1)\cdot\|\mathbf{S}_2^{-1}\mathbf{S}_1-\mathbf{I}_n\|\\ &\quad+2\cdot\|\widetilde{\mathbf{v}}\|\cdot(\|\mathbf{S}_2^{-1}\mathbf{S}_1\|+1)\cdot\|\widetilde{\mathbf{c}}\|+\|\widetilde{\mathbf{c}}\|^2.\end{aligned}$$

Observe that in the last bound above, the only quantity depending on $\mathbf{v}$ is $\widetilde{\mathbf{v}}$. Still for $\mathbf{v}\leftarrow D_{L,\mathbf{S}_1,\mathbf{c}_1}$, we obtain that that

$$\begin{aligned}&\mathbb{E}_{\mathbf{v}}(\|\mathbf{S}_2^{-1}(\mathbf{v}-\mathbf{c}_2)\|^2-\|\mathbf{S}_1^{-1}(\mathbf{v}-\mathbf{c}_1)\|^2)\\ &\qquad\leqslant\mathbb{E}_{\mathbf{v}}(\|\widetilde{\mathbf{v}}^2\|)\cdot(\|\mathbf{S}_2^{-1}\mathbf{S}_1\|+1)\cdot\|\mathbf{S}_2^{-1}\mathbf{S}_1-I_n\|\\ &\qquad\quad+2\cdot\mathbb{E}_{\mathbf{v}}(\|\widetilde{\mathbf{v}}\|)\cdot(\|\mathbf{S}_2^{-1}\mathbf{S}_1\|+1)\cdot\|\widetilde{\mathbf{c}}\|+\|\widetilde{\mathbf{c}}\|^2.\end{aligned}$$

By rewriting $\mathbf{v}=\mathbf{S}_1\cdot\mathbf{w}$, we see that

$$\mathbb{E}_{\mathbf{v}\leftarrow D_{L,\mathbf{S}_1,\mathbf{c}_1}}(\|\mathbf{S}_1^{-1}(\mathbf{v}-\mathbf{c}_1)\|^2)=\mathbb{E}_{\mathbf{w}\leftarrow D_{\mathbf{S}_1^{-1}L,1,\mathbf{S}_1^{-1}\mathbf{c}_1}}(\|\mathbf{w}-\mathbf{S}_1^{-1}\mathbf{c}_1\|^2).$$

Using [MR07, Lemma 4.3] instantiated with $\varepsilon=1/2$ (and using the assumption that $1\geqslant 2\eta_{1/2}(\mathbf{S}_1^{-1}L)$), we have the upper bound $\mathbb{E}_{\mathbf{w}\leftarrow D_{\mathbf{S}_1^{-1}L,1,\mathbf{S}_1^{-1}\mathbf{c}_1}}(\|\mathbf{w}-\mathbf{S}_1^{-1}\mathbf{c}_1\|^2)\leqslant 2n$. Hence, we obtain that $\mathbb{E}_{\mathbf{v}}(\|\widetilde{\mathbf{v}}\|^2)\leqslant 2n$ and $\mathbb{E}_{\mathbf{v}}(\|\widetilde{\mathbf{v}}\|)\leqslant\sqrt{2n}$ (by Jensen's inequality). Combining everything, we obtain that

$$\begin{aligned}&\mathbb{E}_{\mathbf{v}}(\|\mathbf{S}_2^{-1}(\mathbf{v}-\mathbf{c}_2)\|^2-\|\mathbf{S}_1^{-1}(\mathbf{v}-\mathbf{c}_1)\|^2)\\ &\quad\leqslant 2n\cdot(\|\mathbf{S}_2^{-1}\mathbf{S}_1\|+1)\cdot\|\mathbf{S}_2^{-1}\mathbf{S}_1-\mathbf{I}_n\|+2\sqrt{2n}\cdot(\|\mathbf{S}_2^{-1}\mathbf{S}_1\|+1)\cdot\|\widetilde{\mathbf{c}}\|+\|\widetilde{\mathbf{c}}\|^2.\end{aligned}$$

In order to simplify this expression, let us observe that whenever we have $\|\mathbf{S}_2^{-1}\mathbf{S}_1-\mathbf{I}_n\|\geqslant 1/16$ or $\|\widetilde{\mathbf{c}}\|\geqslant 1/16$, then the upper bound in the statement of

the lemma is $\geqslant 1$, and so the lemma is vacuously true. Hence, we can assume that these two quantities are $\leqslant 1/16$. This implies in particular that $\left\|\mathbf{S}_2^{-1}\mathbf{S}_1\right\| \leqslant 17/16$ and $\|\widetilde{\mathbf{c}}\|^2 \leqslant \|\widetilde{\mathbf{c}}\|/16$. Doing so, we finally obtain

$$\mathrm{d_{KL}}(D_{L,\mathbf{S}_1,\mathbf{c}_1} \| D_{L,\mathbf{S}_2,\mathbf{c}_2}) \leqslant 19n \cdot (\left\|\mathbf{S}_2^{-1}\mathbf{S}_1 - \mathbf{I}_n\right\| + \|\mathbf{S}_2^{-1}(\mathbf{c}_1 - \mathbf{c}_2)\|).$$

Using Pinsker's inequality and the fact that $\sqrt{a+b} \leqslant \sqrt{a} + \sqrt{b}$ for all $a, b \geqslant 0$, we obtain that

$$\mathrm{dist}\left(D_{L,\mathbf{S}_1,\mathbf{c}_1}, D_{L,\mathbf{S}_2,\mathbf{c}_2}\right) \leqslant 3.1\sqrt{n} \cdot \left(\sqrt{\left\|\mathbf{S}_2^{-1}\mathbf{S}_1 - \mathbf{I}_n\right\|} + \sqrt{\|\mathbf{S}_2^{-1}(\mathbf{c}_1 - \mathbf{c}_2)\|}\right).$$

The above holds if $\rho_{\mathbf{S}_2}(L - \mathbf{c}_2) \leqslant \rho_{\mathbf{S}_1}(L - \mathbf{c}_1)$. Let us now consider the case where $\rho_{\mathbf{S}_2}(L - \mathbf{c}_2) \geqslant \rho_{\mathbf{S}_1}(L - \mathbf{c}_1)$. In this case, we have a symmetric upper bound

$$\mathrm{dist}\left(D_{L,\mathbf{S}_1,\mathbf{c}_1}, D_{L,\mathbf{S}_2,\mathbf{c}_2}\right) \leqslant 3.1\sqrt{n} \cdot \left(\sqrt{\left\|\mathbf{S}_1^{-1}\mathbf{S}_2 - \mathbf{I}_n\right\|} + \sqrt{\|\mathbf{S}_1^{-1}(\mathbf{c}_1 - \mathbf{c}_2)\|}\right).$$

Note that $\left\|\mathbf{S}_1^{-1}\mathbf{S}_2\right\| \leqslant \left\|\mathbf{S}_1^{-1}\mathbf{S}_2 - \mathbf{I}_n\right\| + 1 \leqslant \left\|\mathbf{S}_1^{-1}\mathbf{S}_2\right\| \cdot \left\|\mathbf{I}_n - \mathbf{S}_2^{-1}\mathbf{S}_1\right\| + 1$. Using the fact that the lemma statement is vacuously true when $\left\|\mathbf{S}_2^{-1}\mathbf{S}_1 - 1\right\| \geqslant 1/16$, we can assume that $\left\|\mathbf{S}_2^{-1}\mathbf{S}_1 - 1\right\| \leqslant 1/16$ and so $\left\|\mathbf{S}_1^{-1}\mathbf{S}_2\right\| \leqslant 17/16$. We hence obtain

$$\left\|\mathbf{S}_1^{-1}\mathbf{S}_2 - \mathbf{I}_n\right\| \leqslant \left\|\mathbf{S}_1^{-1}\mathbf{S}_2\right\| \cdot \left\|\mathbf{S}_2^{-1}\mathbf{S}_1 - \mathbf{I}_n\right\| \leqslant \frac{17}{16}\left\|\mathbf{S}_2^{-1}\mathbf{S}_1 - \mathbf{I}_n\right\|,$$
$$\|\mathbf{S}_1^{-1}(\mathbf{c}_1 - \mathbf{c}_2)\| \leqslant \left\|\mathbf{S}_1^{-1}\mathbf{S}_2\right\| \|\mathbf{S}_2^{-1}(\mathbf{c}_1 - \mathbf{c}_2)\| \leqslant \frac{17}{16}\|\mathbf{S}_2^{-1}(\mathbf{c}_1 - \mathbf{c}_2)\|.$$

We finally conclude that

$$\mathrm{dist}\left(D_{L,\mathbf{S}_1,\mathbf{c}_1}, D_{L,\mathbf{S}_2,\mathbf{c}_2}\right) \leqslant 4\sqrt{n} \cdot \left(\sqrt{\left\|\mathbf{S}_2^{-1}\mathbf{S}_1 - \mathbf{I}_n\right\|} + \sqrt{\|\mathbf{S}_2^{-1}(\mathbf{c}_1 - \mathbf{c}_2)\|}\right),$$

even when $\rho_{\mathbf{S}_2}(L - \mathbf{c}_2) \leqslant \rho_{\mathbf{S}_1}(L - \mathbf{c}_1)$. □

A.4 Proof of Lemma 2.5

We will adapt the approach of Kannan, Lenstra and Lovász [KLL84] for computing the minimal polynomial of an algebraic number, using lattice reduction. Define $\sigma = \sigma_k$. We assume that we are given as input some $\widetilde{y}$ (in $\mathbb{C}$, with a zero imaginary part if the embedding σ is real) such that $|\widetilde{y} - \sigma(x)| \leqslant 2^{-p}$ for some $x \in K$, where p will be determined later (in Lemma A.10 below). We first compute same-quality approximations to the $\sigma(r_i)$'s where $(r_1, \ldots, r_d)$ is the given basis of R:

$$\forall i \leqslant d : |\widetilde{\sigma(r_i)} - \sigma(r_i)| \leqslant 2^{-p}.$$

This can be performed in ppt (with respect to $\log \|\Phi\|$, $\log \Delta_K$ and p) using the defining polynomial Φ (recall that the r_i's are given as polynomials in $\mathbb{Q}[X]/\Phi(X)$, with degree $< d$).

We consider the following matrix in $\mathbb{C}^{(d+3)\times(d+1)}$, where $y = \sigma(x)$:

$$\mathbf{M} = \begin{bmatrix} 2^p \cdot \mathrm{Re}(y) & 2^p \cdot \mathrm{Re}(\sigma(r_1)) & 2^p \cdot \mathrm{Re}(\sigma(r_2)) & \cdots & 2^p \cdot \mathrm{Re}(\sigma(r_d)) \\ 2^p \cdot \mathrm{Im}(y) & 2^p \cdot \mathrm{Im}(\sigma(r_1)) & 2^p \cdot \mathrm{Im}(\sigma(r_2)) & \cdots & 2^p \cdot \mathrm{Im}(\sigma(r_d)) \\ 1 & 0 & 0 & \cdots & 0 \\ 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \end{bmatrix},$$

as well as its approximated version:

$$\widetilde{\mathbf{M}} = \begin{bmatrix} 2^p \cdot \mathrm{Re}(\widetilde{y}) & 2^p \cdot \mathrm{Re}(\widetilde{\sigma(r_1)}) & 2^p \cdot \mathrm{Re}(\widetilde{\sigma(b_2)}) & \cdots & 2^p \cdot \mathrm{Re}(\widetilde{\sigma(r_d)}) \\ 2^p \cdot \mathrm{Im}(\widetilde{y}) & 2^p \cdot \mathrm{Im}(\widetilde{\sigma(r_1)}) & 2^p \cdot \mathrm{Im}(\widetilde{\sigma(r_2)}) & \cdots & 2^p \cdot \mathrm{Im}(\widetilde{\sigma(r_d)}) \\ 1 & 0 & 0 & \cdots & 0 \\ 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \end{bmatrix}.$$

Matrix $\mathbf{M}$ is the one we would like to work with, but instead we use $\widetilde{\mathbf{M}}$, which can be represented with a finite number of bits. We will show that if the approximations are sufficiently accurate, matrix $\widetilde{\mathbf{M}}$ suffices to provide the result. The strategy to recover x is given in Algorithm A.1, for a sufficiently large value of p, to be determined later.

Algorithm A.1

Input: An integer p, a complex $\widetilde{y}$ such that $|\widetilde{y} - \sigma(x)| \leqslant 2^{-p}$ for some $x \in K$.
Output: Element x expressed as a $\mathbb{Q}$-combination of the r_i's.

1: Compute $\widetilde{\mathbf{M}}$.
2: Apply the LLL algorithm on (the columns of) $\widetilde{\mathbf{M}}$ and let $\widetilde{\mathbf{v}} = (\widetilde{e}_1, \widetilde{e}_2, Q_0, \ldots, Q_d)^T$ be the first output vector.
3: Return $x = \sum_{i=1}^{d} -\frac{Q_i}{Q_0} r_i$.

We first analyze a simpler case where $\mathbf{M}$ is used instead of $\widetilde{\mathbf{M}}$. Correctness is easier to prove, assuming that the LLL can handle real numbers. Later, we show that the rational matrix $\widetilde{\mathbf{M}}$ also leads to x.

The exact case. We want to apply the LLL algorithm to the $(d+1)$-dimensional lattice L generated by the columns of $\mathbf{M}$ and we hope that the short vector we get will allow us to find x. This holds if p is sufficiently large, as implied by the following lemmas.

Lemma A.6. *If* $\mathbf{v} = (0, 0, Q_0, \ldots, Q_d)^T$ *is a non-zero vector of* L*, then the* Q_i*'s are integers,* Q_0 *is non-zero and* $x = \sum_{1 \leqslant i \leqslant d} -(Q_i/Q_0)r_i$*.*

Lemma A.7. *If* $\mathbf{v} = (e_1, e_2, Q_0, \ldots, Q_d)^T$ *is a vector of* L *with* $(e_1, e_2) \neq (0, 0)$ *and if* p *satisfies*

$$2^p \geqslant \left(2^{7/2 \cdot (d+1)} \cdot \delta_K \cdot ((d+1) \cdot \|\Phi\|)^{3d+3} \cdot 2^{4\,\mathrm{size}(x)}\right)^d,$$

then $\|\mathbf{v}\| \geqslant 2^{(d+1)/2} \cdot \lambda_1(L)$*.*

These lemmas show that if we applied the LLL algorithm on basis $\mathbf{M}$ of L (assuming it were rational), we would get a vector $\mathbf{w}$ that would allow us to recover x. Indeed, LLL returns a non-zero vector $\mathbf{w}$ such that $\|\mathbf{w}\| < 2^{d/2} \cdot \lambda_1(L)$. By Lemma A.7, its first two entries must be zero, and by Lemma A.6, its coordinates with respect to the basis $\mathbf{M}$ of L reveal x. Below, we will show that LLL applied on the exactly known rational $\widetilde{\mathbf{M}}$ suffices to recover x.

Proof (Proof of Lemma A.6). If we let $(\mathbf{v}_i)_i$ denote the columns of $\mathbf{M}$. We have that $\mathbf{v} = Q_0\mathbf{v}_0 + \cdots + Q_d\mathbf{v}_d$. By looking at the first two rows, we see that $Q_0\sigma(x) + \sum_{1 \leqslant i \leqslant d} Q_i\sigma(r_i) = 0$. As σ is a morphism, we can rewrite the latter as $\sigma(Q_0 x + \sum_{1 \leqslant i \leqslant d} Q_i r_i) = 0$. Since σ is injective, we have $Q_0 x + \sum_{1 \leqslant i \leqslant d} Q_i r_i = 0$. As we assumed that the Q_i's are not all zero and as the r_i's form a $\mathbb{Q}$-basis of K, this implies that $Q_0 \neq 0$. The result follows. □

Proof (Proof of Lemma A.7). Let $\mathbf{v} = (e_1, e_2, Q_0, \ldots, Q_d)^T$ be a vector of L with $(e_1, e_2) \neq (0, 0)$ and the Q_i's in $\mathbb{Z}$. We want to prove that $\|\mathbf{v}\| \geqslant 2^{(d+1)/2} \cdot \lambda_1(L)$. Let D_x be the least common multiple of the denominators of the x_i's (writing $x = \sum_i x_i r_i$). Note that $\mathbf{v}' := (0, 0, -D_x, D_x \cdot x_1, \ldots, D_x \cdot x_d)^T$ belongs to L. Moreover, we know that $\max_i |x_i|, D_x \leqslant 2^{\mathrm{size}(x)}$, so we get

$$\lambda_1(L) \leqslant \|\mathbf{v}'\| \leqslant \sqrt{d+1} \cdot 2^{2\,\mathrm{size}(x)}.$$

To prove the lemma, it thus suffices to show that $\|\mathbf{v}\| \geqslant \sqrt{d+1} \cdot 2^{(d+1)/2} \cdot 2^{2\,\mathrm{size}(x)}$. Without loss of generality, we assume that $(\sum_{0 \leqslant i \leqslant d} Q_i^2)^{1/2}$ is smaller than this quantity.

Let $\zeta \in \mathbb{C}$ be the root of Φ such that for all $g \in K$ seen as a rational polynomial, we have $\sigma(g) = g(\zeta)$. We can then write $e_1 + Ie_2 = 2^p \cdot R(\zeta)$ for the rational polynomial R defined by

$$R = Q_0 x + \sum_{1 \leqslant i \leqslant d} Q_i r_i = \sum_{1 \leqslant i \leqslant d} (Q_0 x_i + Q_i) r_i.$$

We then use a result of [KLL84] to get a lower bound on $|R(\zeta)|$ and prove that $\|\mathbf{v}\|$ is large.

Lemma A.8 (Adapted from [KLL84, Prop. 1.6]). *Let* h *and* g *be non-zero polynomials of* $\mathbb{Z}[X]$ *of degrees* n *and* m*, respectively. Let* ζ *be a root of* h*. If* h *is irreducible and* $g(\zeta) \neq 0$ *then,*

$$|g(\zeta)| \geqslant \frac{1}{n \cdot \|h\|^m \cdot \|g\|^{n-1} \cdot \max(1, |\zeta|^{n-1})} \geqslant \frac{1}{n \cdot (1 + \|h\|)^{m+n-1} \cdot \|g\|^{n-1}}$$

Proof (Proof of Lemma A.8). The first inequality comes from the proof of [KLL84, Prop 1.6], with a small modification to handle the case where $|\zeta| > 1$ (just after [KLL84, Eq. (1.7)]). The second inequality follows from the Cauchy bound on roots of polynomials, which implies that $|\zeta| \leqslant 1 + \|h\|$. □

We now use this lemma with $h = \Phi$ and $g = D_x \cdot D_r \cdot R$, where D_r is the least common multiple of the denominators of the coefficients of the r_i's when viewed as rational polynomials. Note that $D_r \leqslant (2d\|\Phi\|)^{3d}$ as $R \subseteq \frac{1}{\Phi'(\zeta)}\mathbb{Z}[\zeta] \subset \frac{1}{\mathcal{N}(\Phi')}\mathbb{Z}[\zeta]$, where Φ' is the derivative of Φ. Both h and g have integer coefficients. We have $\|h\| = \|\Phi\|$ and, using the triangular inequality,

$$\begin{aligned}\|g\| &\leqslant D_x D_r \sum_{1\leqslant i\leqslant d} |Q_0 x_i + Q_i| \cdot \|r_i\| \\ &\leqslant 2D_x D_r \sqrt{d}\delta_K 2^{\operatorname{size}(x)} \sum_{1\leqslant i\leqslant d} \Big(\sum_{0\leqslant j\leqslant d} Q_j^2\Big)^{1/2} \\ &\leqslant 2(d+1)^2 2^{(d+1)/2}\delta_K (2d\|\Phi\|)^{3d} 2^{4\operatorname{size}(x)}.\end{aligned}$$

Using Lemma A.8 with these bounds on $\|h\|$ and $\|g\|$, we obtain

$$|g(\zeta)| \geqslant \frac{1}{d \cdot 2^{(d+1)(d-1)/2} \cdot K^{d-1}}$$

with $K = \delta_K(2(d+1)\|\Phi\|)^{3d+3}2^{4\operatorname{size}(x)}$. We then have

$$\|\mathbf{v}\| \geqslant\ 2^p \cdot \frac{|g(\zeta)|}{D_x D_r}\ \geqslant \frac{2^p}{d \cdot 2^{(d+1)(d-1)/2} \cdot K^{d-1} \cdot D_x \cdot D_r}.$$

It hence suffices to set p so that $2^p \geqslant (2^{(d+1)/2}K)^d$ to obtain that $\|\mathbf{v}\| \geqslant 2^{(d+1)/2} \cdot \lambda_1(L)$. The condition in the lemma statement guarantees it. □

The approximate case. We first prove that if we take a vector $\mathbf{v} = Q_0\mathbf{v}_0 + \cdots + Q_d\mathbf{v}_d$ in L (recall that the $\mathbf{v}_i$'s are the columns of $\mathbf{M}$) and its approximation $\widetilde{\mathbf{v}} = Q_0\widetilde{\mathbf{v}}_0 + \cdots + Q_d\widetilde{\mathbf{v}}_d$ in $\widetilde{L}$ (with the $\widetilde{\mathbf{v}}_i$'s denoting the columns of $\widetilde{\mathbf{M}}$), then the difference between their norms is small.

Lemma A.9. *Let* $\mathbf{v} = (e_1, e_2, Q_0, \ldots, Q_d)^T \in L$ *and* $\widetilde{\mathbf{v}} = (\widetilde{e_1}, \widetilde{e_2}, Q_0, \ldots, Q_d)^T \in \widetilde{L}$. *Then* $|\|\mathbf{v}\| - \|\widetilde{\mathbf{v}}\|| \leqslant \sum_{0\leqslant i\leqslant d} |Q_i|$.

Proof. By definition of $\widetilde{\mathbf{M}}$, we have

$$\begin{aligned}\delta &:= |(Q_0 y + \sum_{1\leqslant i\leqslant d} Q_i\sigma(r_i)) - (Q_0\widetilde{y} + \sum_{1\leqslant i\leqslant d} Q_i\widetilde{\sigma(r_i)})| \\ &\leqslant |Q_0| \cdot |y - \widetilde{y}| + \sum_{1\leqslant i\leqslant d} |Q_i| \cdot |\sigma(r_i) - \widetilde{\sigma(r_i)}| \\ &\leqslant (\sum_{0\leqslant i\leqslant d} |Q_i|) \cdot 2^{-p}.\end{aligned}$$

We then obtain that $|\|\mathbf{v}\| - \|\widetilde{\mathbf{v}}\|| \leqslant \|\mathbf{v} - \widetilde{\mathbf{v}}\| \leqslant 2^p \cdot \delta \leqslant \sum_{0\leqslant i\leqslant d} |Q_i|$. □

This lemma allows us to obtain a bound on $\lambda_1(\widetilde{L})$. Let $\mathbf{v}' = (0, 0, -D_x, D_x \cdot x_1, \ldots, D_x \cdot x_d)^T \in L$ as in the proof of Lemma A.7 (i.e., with D_x the least common multiple of the x_i's), and let $\widetilde{\mathbf{v}}' \in \widetilde{L}$ be its corresponding approximation. Then

$$\lambda_1(\widetilde{L}) \leqslant \|\widetilde{\mathbf{v}'}\| \leqslant \|\mathbf{v}'\| + D_x(1 + \sum_{1 \leqslant i \leqslant d} |x_i|) \leqslant 2(d+1)2^{2\,\mathrm{size}(x)}.$$

The latter results combined with Lemma A.7 allow us to ensure that all vectors that do not correspond to x are significantly longer than $\lambda_1(\widetilde{L})$.

Lemma A.10. *Assume that p satisfies:*

$$2^p \geqslant \left(2^{(7d+9)/2} \cdot \delta_K \cdot ((d+1) \cdot \|\Phi\|)^{3d+4} \cdot 2^{4\,\mathrm{size}(x)}\right)^d.$$

Let $\widetilde{\mathbf{v}} = \sum_{0 \leqslant i \leqslant d} Q_i \widetilde{\mathbf{v}}_i \in \widetilde{L}$. If the first two entries of the corresponding $\mathbf{v} = \sum_{0 \leqslant i \leqslant d} Q_i \mathbf{v}_i \in L$ are not both zero, then

$$\|\widetilde{\mathbf{v}}\| \geqslant 2^{(d+1)/2} \cdot \lambda_1(\widetilde{L}).$$

Else, we have $x = \sum_{1 \leqslant i \leqslant d} -(Q_i/Q_0) r_i$.

A direct consequence of this lemma is that when given $\widetilde{\mathbf{M}}$ as input, the LLL algorithm returns a vector $\widetilde{\mathbf{v}} = \sum_{0 \leqslant i \leqslant d} Q_i \cdot \widetilde{\mathbf{v}}_i \in \widetilde{L}$ such that the first two entries of $\mathbf{v} = \sum_{0 \leqslant i \leqslant d} Q_i \cdot \mathbf{v}_i \in L$ are zero. By Lemma A.6, the Q_i's reveal x.

Proof. As in the proof of Lemma A.7, we may assume that $(\sum_{0 \leqslant i \leqslant d} Q_i^2)^{1/2} < 2^{(d+3)/2} \cdot (d+1) 2^{2\,\mathrm{size}(x)}$. We know that in this case (see the end of the proof of Lemma A.7):

$$\|\mathbf{v}\| \geqslant \frac{2^p}{d \cdot 2^{(d+1)(d-1)/2} \cdot \widetilde{K}^{d-1} \cdot D_x \cdot D_r},$$

where $\widetilde{K} = \delta_K (2(d+1)\|\Phi\|)^{3d+4} 2^{4\,\mathrm{size}(x)}$ (the update from K to $\widetilde{K}$ comes from the upper bound on $(\sum_{0 \leqslant i \leqslant d} Q_i^2)^{1/2}$, which has been multiplied by $2\sqrt{d+1}$). Thanks to Lemma A.9, we obtain that

$$\begin{aligned}
\|\widetilde{\mathbf{v}}\| &\geqslant \frac{2^p}{d \cdot 2^{(d+1)(d-1)/2} \cdot \widetilde{K}^{d-1} \cdot D_x \cdot D_r} - \sum_{0 \leqslant i \leqslant d} |Q_i| \\
&\geqslant \frac{2^p}{d \cdot 2^{(d+1)(d-1)/2} \cdot \widetilde{K}^{d-1} \cdot D_x \cdot D_r} - 2^{(d+3)/2}(d+1)^2 2^{2\,\mathrm{size}(x)}.
\end{aligned}$$

The condition on p implies that $2^p \geqslant (2^{(d+1)/2}\widetilde{K})^d$, which ensures that $\|\widetilde{\mathbf{v}}\| \geqslant 2^{(d+1)/2} \cdot \lambda_1(\widetilde{L})$. □

The fact that Algorithm A.1 returns x follows from Lemma A.10. Its polynomial complexity follows from the facts that $\widetilde{\mathbf{M}}$ may be computed in polynomial time and that LLL runs in polynomial time. This completes the proof of Lemma 2.5. □

A.5 Proof of Proposition 2.10

We only explain how to adapt the proof of [PRS17, Prop. 4.4]. Observe that in [PRS17, Prop. 4.4], the third condition is rather

3'. for any $t_1, t_2 \geqslant 0$, it holds that $|p(t_1) - p(t_2)| \leqslant \kappa |t_1 - t_2|$.

However, we argue that Property 3' can be replaced by Property 3 without impacting the result and with limited impact on the proof. Indeed, in [PRS17], Property 3' is only used to ensure that, in the proof of Lemma 4.2 (on page 13), it holds that $|p(t + \chi\Delta) - p(t)| \leqslant 1/(100\lambda t_{\max})$ for some $\Delta \geqslant 1/\operatorname{poly}(\kappa, \lambda)$ and all $\chi \in [0, 1]$ (the exact definitions of λ and $t_{\max}$ are given in [PRS17] but are not necessary here; we only need to know that $t_{\max} = \operatorname{poly}(\kappa, \lambda)$). In order to prove this, the authors pick $\Delta := 1/(100\kappa\lambda t_{\max}) \geqslant 1/\operatorname{poly}(\kappa, \lambda)$ and then use Property 3' to show that

$$|p(t + \chi\Delta) - p(t)| \leqslant \kappa\Delta \leqslant 1/(100\lambda t_{\max}).$$

If we have Property 3 instead of Property 3', we obtain the same result by setting $\Delta := 1/(100\kappa\lambda t_{\max})^2$, which is also $\geqslant 1/\operatorname{poly}(\kappa, \lambda)$. The rest of the proof only uses the fact that $\Delta \geqslant 1/\operatorname{poly}(\kappa, \lambda)$ and not the exact value of Δ, so it is unchanged. □

B Missing proofs from Section 3

B.1 Reducing $\mathrm{NTRU}_{\mathrm{vec}}$ to $\mathrm{NTRU}_{\mathrm{mod}}$– Proof of Lemma 3.8

Let h be a (γ, q)-NTRU instance. Using the (γ, q)-$\mathrm{NTRU}_{\mathrm{mod}}$ oracle, the reduction first recovers $h_K \in K$ as defined in Lemma 3.5. Then it computes a lattice basis of the rank-1 submodule $(1, h_K)^T \cdot K \cap M_h$ of M_h. It contains $(g, f)^T$, where (f, g) is arbitrary satisfying $g \cdot h = f \bmod q$ and $\|f\|, \|g\| \leqslant \sqrt{q}/\gamma$. In particular, by inclusion into $(g, f)^T \cdot R$, the corresponding lattice has determinant at most $\sqrt{\Delta_K}(\sqrt{2q}/\gamma)^d$. By using the γ_{hsvp}-id-HSVP oracle via Lemma A.5, we obtain $(g', f')^T \in (1, h_K)^T \cdot K \cap M_h$ such that $\|f'\|, \|g'\| \leqslant (1 + \varepsilon)\gamma_{\mathrm{hsvp}}\Delta_K^{1/(2d)}(\sqrt{2q}/\gamma)$. □

B.2 Reducing dNTRU to $\mathrm{NTRU}_{\mathrm{mod}}$– Proof of Lemma 3.9

Let h be a $(\mathcal{D}, \gamma, q)$-dNTRU instance. The goal is to decide whether h has been sampled from $\mathcal{D}$ or $U(R_q)$. For this, the reduction $\mathcal{B}$:

1. calls the $(\mathcal{D}, \gamma, q)$-$\mathrm{NTRU}_{\mathrm{mod}}$ oracle $\mathcal{A}$; let h_K be its (possibly incorrect) output;
2. computes the intersection M' between $(1, h_K) \cdot K$ and M_h;
3. outputs 1 if and only if M' is a rank-1 module and $\det(M') \leqslant \sqrt{\Delta_K}(\sqrt{2q}/\gamma)^d$.

If h is sampled from $\mathcal{D}$, then $\mathcal{A}$ receives a correctly distributed $(\mathcal{D}, \gamma, q)$-$\mathrm{NTRU}_{\mathrm{mod}}$ instance and returns h_K with probability $\mathrm{Adv}(\mathcal{A})$. In that case, the module M' has rank 1 and contains a non-zero vector $(g, f)^T$ with $\|f\|, \|g\| \leqslant \sqrt{q}/\gamma$. In this situation, algorithm $\mathcal{B}$ outputs 1.

If h is sampled from $U(R_q)$, we claim that M_h does not contain any rank-1 module M' with $\det(M') \leqslant \sqrt{\Delta_K}(\sqrt{2q}/\gamma)^d$. If it was, then, by Minkowski's theorem, it would contain a vector $(g, f)^T$ such that $\|f\|, \|g\| \leqslant \Delta_K^{1/(2d)}\sqrt{2dq}/\gamma$. We prove that for a sufficiently large γ, this can happen only with exponentially small probability. Unless this unlikely event occurs, algorithm $\mathcal{B}$ outputs 0. Proving the lemma below will hence complete the proof of Lemma 3.9.

Lemma B.1. *Let $q \geqslant 2$ be a prime that does not divide Δ_K. Except with a probability $\leqslant 2^{-d}$ over the uniform choice of $h \in R_q$, we have $\lambda_1(M_h) > \sqrt{q}/8$.*

The proof follows the design of the standard proofs of the Minkowski-Hlawka theorem. More concretely, we adapt the proof of [RSW18, Le. 5.2] to our precise setup.

Proof. By choice of q, we can write $qR = \prod_{i \leqslant k} \mathfrak{p}_i$ for distinct prime ideals $\mathfrak{p}_i$. By the Chinese Remainder Theorem, we have $R_q \simeq \bigoplus_{i \leqslant k} \mathbb{F}_{q^{d_i}}$ with $q^{d_i} = \mathcal{N}(\mathfrak{p}_i)$. Fix some bound B (to be determined later) and define p_B as the probability that $\lambda_1(M_h) \leqslant B$ over $h \leftarrow U(R_q)$. By definition of M_h and the union bound, we have:

$$p_B \leqslant \sum_{\substack{(f,g) \in R^2 \setminus \{(0,0)\} \\ \|f\|, \|g\| \leqslant B}} \Pr_h[g \cdot h = f \bmod q].$$

This leads us to studying the number of solutions $h \in R_q$ to the equation $g \cdot h = f \bmod q$. Note that if there is an i such that $g = 0 \bmod \mathfrak{p}_i$ and $f \neq 0 \bmod \mathfrak{p}_i$, then there is no solution. Now, let S denote the set of i's such that $g = 0 \bmod \mathfrak{p}_i$ and assume that $f = 0 \bmod \mathfrak{p}_i$ for all $i \in S$. Let $d_S = \sum_{i \in S} d_i$. Then, by considering the Chinese Remainder Theorem decomposition, one sees that there are exactly $\prod_{i \in S} q^{d_i} = q^{d_S}$ solutions h to $g \cdot h = f \bmod q$. We hence obtain that $\Pr_h[g \cdot h = f \bmod q] = q^{d_S - d}$, and:

$$p_B \leqslant \sum_{S \subseteq [k]} \sum_{\substack{(f,g) \in R^2 \setminus \{(0,0)\} \\ \|f\|, \|g\| \leqslant B \\ \forall i \in S : f, g \in \mathfrak{p}_i}} q^{d_S - d}.$$

For any B and S, define $N(B, S)$ as the number of elements of $I_S := \prod_{i \in S} \mathfrak{p}_i$ of Euclidean norms $\leqslant B$. As there are $2^k \leqslant 2^d$ possible choices for S, we obtain that:

$$p_B \leqslant 2^d \cdot \max_{S \subseteq [k]} \frac{N(B, S)^2 - 1}{q^{d - d_S}}.$$

To bound $N(B, S)$, observe that the balls of radius $\lambda_1(I_S)/2$ and centers the elements of I_S cannot intersect. This implies that $N(B, S) \leqslant (1 + 2B/\lambda_1(I_S))^d$.

Further, as seen is Subsection 2.4, we have that $\lambda_1(I_S) \geqslant \mathcal{N}(I_S)^{1/d} = q^{d_S/d}$. Overall, we obtain that:

$$N(B,S) \leqslant \begin{cases} 1 & \text{if} \quad B < q^{d_S/d}, \\ 2^{2d} B^d q^{-d_S} & \text{else,} \end{cases}$$

where the first line comes from the definition of $N(B,S)$ and $\lambda_1(I_S)$. Plugging the latter in the last upper bound on p_B, we obtain:

$$p_B \leqslant 2^{5d} \cdot \max_{\substack{S \subseteq [k] \\ q^{d_S/d} \leqslant B}} \frac{B^{2d} q^{-2d_S}}{q^{d-d_S}} = 2^{5d} \cdot B^{2d} \cdot q^{-d}.$$

The result follows. □

C Missing proofs from Section 4

C.1 Reducing worst-case to average-case id-HSVP – Proof of Theorem 4.4

We start with a lemma which will be used in the proof of Theorem 4.4. It considers the task of sampling a prime ideal together with a short non-zero element of it. Note that it holds under the Extended Riemann Hypothesis (ERH).

Lemma C.1 (Sampling prime ideals with a short vector, ERH). *There exists a classical algorithm that on input an integer $B \geqslant 1$ samples pairs $(v, \mathfrak{p})$ in time polynomial in $\log B$ and $\log \Delta_K$, such that*

1. *$\mathfrak{p}$ is a prime ideal of norm $\leqslant B$ and its distribution is within $2^{-\Omega(d)}$ statistical distance from the uniform distribution over prime ideals of norm $\leqslant B$;*
2. *$v \in \mathfrak{p} \backslash \{0\}$;*
3. *$\|v\| \leqslant 2^{\sqrt{\log B}} \cdot d \cdot \delta_K \cdot B^{1/d}$.*

Moreover, if we have access to an oracle factoring integers of bit-lengths $\leqslant \log(B) + \log \Delta_K + d \cdot \log(d^{1.5} \cdot \delta_K)$, then Property 3 can be improved to

3'. $\|v\| \leqslant d^{1.5} \cdot \delta_K \cdot \Delta_K^{1/d} \cdot B^{1/d}$.

Proof. We describe two different algorithms, one using a factoring oracle and the other one not.

Algorithm without factoring oracle. The algorithm proceeds as follows. It first samples a uniform prime ideal $\mathfrak{p}$ of norm $\leqslant B$ in time polynomial in $\log \Delta_K$ and $\log B$, using for instance [dBDPW20, Le. 2.2] (which holds under ERH). Note that [dBDPW20, Le. 2.2] requires that B should be larger than some quantity that is polynomial in $\log \Delta_K$: for B smaller than this bound, one can simply enumerate in polynomial time all prime ideals of norm $\leqslant B$ and then pick one uniformly at random in the list. Then the algorithm would use Lemma A.4

to find a somehow short vector in $\mathfrak{p}$. Note however that Lemma A.4 cannot be directly used, as $\mathfrak{p}$ may not be an integral lattice. In order to make it into an integral ideal, we consider a basis of $\mathfrak{p}$ over R. If $\mathbf{B}_\mathfrak{p}$ is a basis of $\mathfrak{p}$, then since $\mathfrak{p}$ is a sub-lattice of R, we know that $\mathbf{B}_\mathfrak{p} = \mathbf{B}_R \cdot \mathbf{S}$, for some matrix $\mathbf{S} \in \mathbb{Z}^{d\times d}$ and where $\mathbf{B}_R$ is the known basis of R (recall that each coefficient of $\mathbf{B}_R$ has absolute value at most δ_K). Let us consider the lattice $L = \mathcal{L}(S)$, which is integral. We know that $\det(L) = |\det(S)| = \mathcal{N}(\mathfrak{p}) \leqslant B$. Using Lemma A.4, we can find in polynomial time a vector $\mathbf{w} \in L\backslash\{\mathbf{0}\}$ such that

$$\|\mathbf{w}\| \leqslant 2^{\sqrt{\log B}} \cdot B^{1/d}.$$

Multiplying by $\mathbf{B}_R$, we obtain a vector $\mathbf{v} = \mathbf{B}_R \cdot \mathbf{w} \in \mathfrak{p}\backslash\{0\}$ such that

$$\|\mathbf{v}\| \leqslant \max_i \|r_i\| \cdot \|\mathbf{w}\|_1 \leqslant d \cdot \delta_K \cdot 2^{\sqrt{\log B}} \cdot B^{1/d}.$$

Algorithm with a factoring oracle. Contrary to the previous algorithm, which constructed the prime ideal $\mathfrak{p}$ first and then computed a short vector in $\mathfrak{p}$, this algorithm first samples a small $v \in R\backslash\{0\}$ and then constructs a prime ideal containing v. More formally, we will consider a first algorithm $\mathcal{A}$ that proceeds as follows

- sample $v \leftarrow \widetilde{D}_{\mathbf{B}_R,\sigma}$, where $\mathbf{B}_R$ is the known basis of R and $\sigma = B^{1/d} \cdot \Delta_K^{1/d} \cdot d \cdot \delta_K$;
- factor $\langle v\rangle = \prod_{i=1}^r \mathfrak{p}_i^{e_i} \cdot \prod_{j=1}^s \mathfrak{q}_i^{f_j}$, where the $\mathfrak{p}_i$'s are distinct prime ideals of norm $\leqslant B$, the $\mathfrak{q}_j$'s are distinct prime ideals of norm $> B$ and $e_i, f_j \geqslant 1$;
- pick $i_0 \in \{1,\dots,r\}$ uniformly at random (r is the number of distinct prime factors of $\langle v\rangle$ of norm $\leqslant B$); if $r = 0$, then $\mathcal{A}$ outputs $\perp$;
- output $(v, \mathfrak{p}_{i_0})$ with probability $\frac{r}{\log(B\cdot\Delta_K\cdot(d^{1.5}\cdot\delta_K)^d)} \cdot \frac{\mathcal{N}(\mathfrak{p}_{i_0})}{B}$, and otherwise output $\perp$.

Thanks to Lemma 2.2, if one can factor $\langle v\rangle$, then $\mathcal{A}$ runs in polynomial time. To factor $\langle v\rangle$, it suffices to factor $\mathcal{N}(v)$, which is an integer bounded from above by $\|v\|^d \leqslant (\sqrt{d}\cdot\sigma)^d \leqslant B\Delta_K\cdot(d^{1.5}\cdot\delta_K)^d$ (where we used Lemma 2.2 and the definition of σ). Observe also that since $\mathcal{N}(v) \leqslant B \cdot \Delta_K \cdot (d^{1.5}\cdot\delta_K)^d$, then v has at most $\log(B \cdot \Delta_K \cdot (d^{1.5}\cdot\delta_K)^d)$ prime factors, and so the quantity $\frac{r}{\log(B\cdot\Delta_K\cdot(d^{1.5}\cdot\delta_K)^d)} \cdot \frac{\mathcal{N}(\mathfrak{p}_{i_0})}{B}$ in the last step of the algorithm is indeed in $[0,1]$. Finally, observe that if $\mathcal{A}$ outputs $(v,\mathfrak{p})$, then $v \in \mathfrak{p}\backslash\{0\}$, and $\|v\| \leqslant B^{1/d} \cdot \Delta_K^{1/d} \cdot d^{1.5} \cdot \delta_K$ (using Lemma 2.2).

Let us now fix some prime ideal $\mathfrak{p}$ of norm $\leqslant B$ and compute the probability that $\mathcal{A}$ outputs $\mathfrak{p}$ (together with some vector v).

$$\Pr(\mathcal{A} \text{ outputs } \mathfrak{p}) = \sum_{v\in R} \widetilde{D}_{\mathbf{B}_R,\sigma}(v) \cdot \Pr(\mathcal{A} \text{ outputs } \mathfrak{p}|v).$$

Let us fix some $v \in R$. We know that if $v \notin \mathfrak{p}$, then $\Pr(\mathcal{A} \text{ outputs } \mathfrak{p}|v) = 0$. If $v \in \mathfrak{p}$, then $\mathfrak{p}|\langle v\rangle$, and so in the decomposition $\langle v\rangle = \prod_{i=1}^r \mathfrak{p}_i^{e_i} \cdot \prod_{j=1}^s \mathfrak{q}_i^{f_j}$ of the

algorithm, we can assume wlog that $\mathfrak{p}_1 = \mathfrak{p}$ (and in particular $r \geqslant 1$). Then we have

$$\begin{aligned}\Pr(\mathcal{A} \text{ outputs } \mathfrak{p}|v) &= \Pr(i_0 = 1) \cdot \frac{r}{\log(B \cdot \Delta_K \cdot (d^{1.5} \cdot \delta_K)^d)} \cdot \frac{\mathcal{N}(\mathfrak{p})}{B} \\ &= \frac{1}{\log(B \cdot \Delta_K \cdot (d^{1.5} \cdot \delta_K)^d)} \cdot \frac{\mathcal{N}(\mathfrak{p})}{B}.\end{aligned}$$

Observe that this probability does not depend on v (as long as $v \in \mathfrak{p}$). Combining everything, we obtain that

$$\begin{aligned}\Pr(\mathcal{A} \text{ outputs } \mathfrak{p}) &= \Pr_{v \leftarrow \tilde{D}_{\mathbf{B}_R, \sigma}}\Big(v \in \mathfrak{p}\Big) \cdot \frac{1}{\log(B \cdot \Delta_K \cdot (d^{1.5} \cdot \delta_K)^d)} \cdot \frac{\mathcal{N}(\mathfrak{p})}{B} \\ &= \frac{\delta_{\mathfrak{p}}}{\log(B \cdot \Delta_K \cdot (d^{1.5} \cdot \delta_K)^d) \cdot B},\end{aligned}$$

for some $\delta_{\mathfrak{p}} \in [1 - 2^{-\Omega(d)}, 1 + 2^{-\Omega(d)}]$. Here we used the fact that $\Pr(v \in \mathfrak{p}) = \mathcal{N}(\mathfrak{p})^{-1} \cdot (1 \pm 2^{-\Omega(d)})$, thanks to Lemmas 2.2 and A.1. Note that we can indeed use Lemma A.1, as $\sigma \geqslant \sqrt{d}\lambda_d(\mathfrak{p}) \geqslant \eta_{2^{-d}}(\mathfrak{p})$ (due to Eq. (2.1) and the fact that $\lambda_d(\mathfrak{p}) \leqslant \delta_K \lambda_1(\mathfrak{p})$). To conclude, algorithm $\mathcal{A}$ has the same probability to output any prime ideal $\mathfrak{p}$ of norm $\leqslant B$, up to a factor $1 \pm 2^{-\Omega(d)}$.

Let us now consider algorithm $\mathcal{B}$, which runs $\mathcal{A}$ until $\mathcal{A}$ outputs a pair $(v, \mathfrak{p})$ (rather than $\perp$), and then outputs this pair. We have seen that the pair $(v, \mathfrak{p})$ output by $\mathcal{B}$ satisfies Properties 1, 2 and 3' from the lemma statement. We also know that $\mathcal{A}$ runs in polynomial time (given access to a factoring oracle). Hence, it suffices to show that $\mathcal{B}$ calls $\mathcal{A}$ an expected number of times that is polynomial in $\log B$ and $\log \Delta_K$. To do so, we bound the probability that $\mathcal{A}$ outputs a pair $(v, \mathfrak{p}) \neq \perp$ from below.

$$\begin{aligned}\Pr(\mathcal{A} \text{ does not output } \perp) &= \sum_{\substack{\mathfrak{p} \text{ prime} \\ \mathcal{N}(\mathfrak{p}) \leqslant B}} \Pr(\mathcal{A} \text{ outputs } \mathfrak{p}) \\ &= \sum_{\substack{\mathfrak{p} \text{ prime} \\ \mathcal{N}(\mathfrak{p}) \leqslant B}} \frac{\delta_{\mathfrak{p}}}{\log(B \cdot \Delta_K \cdot (d^{1.5} \cdot \delta_K)^d) \cdot B} \\ &\geqslant \frac{(1 - 2^{-\Omega(d)}) \cdot |\{\text{prime } \mathfrak{p} \,|\, \mathcal{N}(\mathfrak{p}) \leqslant B\}|}{\log(B \cdot \Delta_K \cdot (d^{1.5} \cdot \delta_K)^d) \cdot B}.\end{aligned}$$

By [BS96, Th. 8.7.4] (under ERH), we have that $|\{\text{prime } \mathfrak{p} \,|\, \mathcal{N}(\mathfrak{p}) \leqslant B\}| \geqslant B/\operatorname{poly}(\log \Delta_K, \log B)$. Hence, we conclude that $\mathcal{A}$ outputs a good pair $(v, \mathfrak{p})$ with probability at least $1/\operatorname{poly}(\log \Delta_K, \log B)$, and so $\mathcal{B}$ runs in expected polynomial time (provided we have a factoring oracle). □

Let us now recall and prove Theorem 4.4.

Theorem 4.4 (Adapted from [dBDPW20, Th. 4.5], ERH) *Let K a number field of degree d and $N \geqslant (12d^{1.5} \log d \cdot \delta_K \cdot \Delta_K^{1/(2d)})^d$ an integer. Let $\gamma > 0$.*

There exist $\gamma' = \gamma \cdot O(d^{1.5}\Delta_K^{1/d})$*, a distribution* $\mathcal{D}_N^{\text{id-HSVP}}$ *over non-zero integral ideals of* K *of norm* $\leqslant N$ *and a reduction:*

- *from worst-case* γ'*-id-*HSVP *for all fractional ideals of* K*,*
- *to average-case* γ*-id-*HSVP *for integral ideals distributed from* $\mathcal{D}_N^{\text{id-HSVP}}$*.*

The reduction decreases the success probability by at most $2^{-\Omega(d)}$*, makes a single call to the average-case* γ*-id-*HSVP *oracle, and runs in time* $T_\beta^{\text{id-HSVP}} + \mathrm{poly}(\log N, \mathrm{size}(I), \log \Delta_K)$ *where*

- I *is the input (worst-case) ideal;*
- $T_\beta^{\text{id-HSVP}}$ *is the time needed to solve id-HSVP with approximation factor* $2^{d/\beta}$ *and*

$$\beta = \left\lceil \frac{d}{\log\left(N^{1/d}/(6d^{1.5}\log d \cdot \delta_K \cdot \Delta_K^{1/(2d)})\right)} \right\rceil.$$

Moreover, there exist $N_0 = \mathrm{poly}(\Delta_K^{1/d}, \delta_K, d)^d$ *and a* ppt *algorithm* $\mathcal{A}$ *(with respect to* $\log N$ *and* $\log \Delta_K$*) such that, for all* $N \geqslant N_0$*, algorithm* $\mathcal{A}$ *samples pairs* (J, w) *such that:*

- *the ideal* J *is a non-zero integral ideal of norm* $\leqslant N$*;*
- *the distribution* $\widetilde{\mathcal{D}}_N^{\text{id-HSVP}}$ *of* J *satisfies* $\widetilde{\mathcal{D}}_N^{\text{id-HSVP}} \approx_{2^{-\Omega(d)}} \mathcal{D}_N^{\text{id-HSVP}}$*;*
- *the element* $w \in J\backslash\{0\}$ *satisfies* $\|w\| \leqslant \mathrm{poly}(d, \delta_K, \Delta_K^{1/d}, 2^{\sqrt{\log \Delta_K + d\log d}}) \cdot \mathcal{N}(J)^{1/d}$*.*

If we have access to a factoring oracle or if $N \geqslant N_0' = 2^{O(d\sqrt{\log \Delta_K + d\log d})} \cdot N_0$*, then we can reduce the size of* w *down to* $\|w\| \leqslant \mathrm{poly}(d, \delta_K, \Delta_K^{1/d}) \cdot \mathcal{N}(J)^{1/d}$*.*

Proof. We first recall the high level idea of the worst-case to average-case reduction of [dBDPW20, Se. 4]. In order to re-randomize an ideal I, the reduction proceeds as follows. It multiplies I by a product of n randomly chosen prime ideals of algebraic norms $\leqslant B$ (where n and B have to be chosen carefully), and by an element $x \in K_\mathbb{R}$ very close to 1. Let us call $L = x \cdot I \cdot \prod_{i=1}^n \mathfrak{p}_i$ the obtained ideal. The reduction then computes an LLL-reduced basis of L, and samples $v \in L$ from a Gaussian distribution. It then outputs $J := v^{-1} \cdot L$ as the average-case ideal. The multiplication of I by the $x \cdot \prod_{i=1}^n \mathfrak{p}_i$ is used to make the ideal somehow uniform, and the multiplication by v^{-1} is used to extract a canonical representation of L, that does not depend on x nor on the product $\prod_{i=1}^n \mathfrak{p}_i$. The high level intuition behind the reduction is the following. The ideal L is very close to the ideal $I \cdot \prod_{i=1}^n \mathfrak{p}_i$ (since x is very close to 1), which is itself a (not too sparse) sublattice of I. Hence, any vector of L multiplied by x^{-1} is a vector of I, and the short vectors of L (after multiplication by x^{-1}) will still be somehow short in I (since multiplication by x and x^{-1} does not distort the space too much). Then, since v is sampled from a Gaussian distribution, it should be balanced with high probability, and so multiplication by v or v^{-1} should not change the geometry of the ideal too much. In particular, any short vector $w \in J$ can be transformed into a somehow short vector $x^{-1} \cdot v \cdot w$ of I.

Let us now consider the first part of the theorem, i.e., the reduction from worst-case to average-case ideal-HSVP. The statement above differs from what was proven in [dBDPW20, Se. 4] in three ways: the average-case distribution $\mathcal{D}_N^{\text{id-HSVP}}$ is over integral ideals instead of inverses of integral ideals; we have the ability to reduce the upper bound N on the norm of the ideals in exchange for a longer run-time; and the success probability is decreased by a factor $2^{-\Omega(d)}$ instead of $d^{-\omega(1)}$.

Let us start with a high level overview. We have seen that the average case ideal that is output by the reduction in [dBDPW20, Se. 4] is $J = v^{-1} \cdot L$ for some $v \in L$, which is indeed the inverse of an integral ideal. In order to output an integral ideal, we sample $v \in L^{-1}$ and then output $J = v \cdot L$. This does not change anything for the rest of the reduction, since the only important aspect here is that the multiplication by v or v^{-1} does not change the geometry of the corresponding lattices too much. Let us now consider the norm of J. This norm will be roughly equal to $\|v\|^d \cdot \mathcal{N}(L)$. If we have a basis of L^{-1} whose vectors have norms $\leqslant \gamma \cdot \mathcal{N}(L^{-1})^{1/d}$ (for some approximation factor $\gamma \geqslant 1$), then we can sample v as small as this quantity (up to polynomial factors in d), and so we obtain an ideal $v \cdot L$ of norm bounded by γ^d. The reduction of [dBDPW20] runs the LLL algorithm on L^{-1} in order to obtain $\gamma = 2^d$, which means that the output ideal has norm $2^{O(d^2)}$. By running a stronger reduction algorithm than LLL, one can decrease γ, which in turn will decrease the norm of the output ideal, but will increase the reduction run-time. Finally, the somehow large loss in the success probability in [dBDPW20] (i.e., $d^{-\omega(1)}$) comes from the fact that the authors tried to obtain a very tight bound for the new approximation factor γ' for cyclotomic fields. To do so, they consider a distortion x which is close to 1, but only with probability $d^{-\omega(1)}$. In our case, we consider a distribution of x close to 1 with probability $2^{-\Omega(d)}$, even if this leads to a slightly larger loss in the approximation factor γ'.

We now give more precise details on which parts of [dBDPW20, Se. 4] need to be updated to obtain the first part of our theorem. The distribution that we call $\mathcal{D}_N^{\text{id-HSVP}}$ in this article is the distribution that was called $\mathcal{D}_{\varsigma,M}^{\text{perfect}}$ in [dBDPW20] (and defined in [dBDPW20, Eq. (11)]). It was proven in [dBDPW20, Le. 4.1] that this distribution $\mathcal{D}_{\varsigma,M}^{\text{perfect}}$ is over fractional ideals $\mathfrak{b}$ such that $\mathfrak{b}^{-1}$ is an integral ideal and $\mathcal{N}(\mathfrak{b}^{-1}) \leqslant (\sqrt{d}\varsigma + M)^d$.

We first observe that one can easily obtain integral ideals instead of inverses of integral ideals. To do so, it is sufficient to change the second and third steps of the Extract procedure described in [dBDPW20, Algorithm 1]: instead of sampling v in L and outputting $v^{-1} \cdot L$, one can sample v in L^{-1} and output vL (here L is an ideal lattice). Since any $v \in L^{-1}$ is of the form $\langle v \rangle = \mathfrak{b} \cdot L^{-1}$ for some integral ideal $\mathfrak{b}$, we have that $v \cdot L = \mathfrak{b}$ is an integral ideal as desired. Moreover, sampling v in L^{-1} instead of L does not change the rest of the reduction (note that L is an ideal of norm 1, hence L^{-1} is also an ideal of norm 1). With this small modification, the proof of [dBDPW20, Le. 4.1] still works, and we can show that the new distribution $\mathcal{D}_{\varsigma,M}^{\text{perfect}}$ outputs integral ideals $\mathfrak{b}$ such that $\mathcal{N}(\mathfrak{b}) \leqslant (\sqrt{d}\varsigma + M)^d$.

Let us now see that we can lower the upper bound $N := (\sqrt{d}\varsigma + M)^d$ on the norms of the ideals, if we are willing to spend more time on the reduction. In the reduction proven in [dBDPW20, Th. 4.5], the parameters ς and M are chosen to be equal to $\varsigma = 2^{d+1} \cdot \sqrt{d} \cdot \Delta_K^{1/(2d)} \cdot \lambda_d(R)$ and $M = 2\sqrt{d}\varsigma$. This gives us the upper bound $N = (6 \cdot 2^d \cdot d^{1.5} \cdot \Delta_K^{1/(2d)} \cdot \delta_K)^d$ on the norms of the ideals produced by $\mathcal{D}_{\varsigma,M}^{\text{perfect}}$ (where we used the fact that $\lambda_d(R) \leqslant \sqrt{d}\delta_K$). This can be achieved with a polynomial time reduction.

If one wants to lower the bound $N = (\sqrt{d}\varsigma + M)^d$, one could keep $M = 2\sqrt{d}\varsigma$ and simply try to lower ς. The lower bound on ς provided in [dBDPW20, Th. 4.5] actually comes from [dBDPW20, Th. 4.4]. In this theorem, the lower bound on ς is used twice in the proof. First, in order to show that the distributions D_3 and D_4 introduced in the proof are statistically close, one needs to ensure that $\varsigma \geqslant \sqrt{d \log d} \cdot \lambda_d(R) \cdot \Delta_K^{1/(2d)}$. This lower bound is much smaller than the one we try to improve (it does not contain the 2^d term), and so we will keep it. The second place in the proof where the lower bound on ς is used is to show that the distributions D_6 and D_7 are statistically close. In this part of the proof, we need to ensure that it is possible to use Klein's algorithm to sample from a Gaussian distribution with parameter ς in an ideal lattice $(L')^{-1}$ of norm $\mathcal{N}((L')^{-1}) \leqslant 2$.[4] This will be doable only if $\varsigma \geqslant \sqrt{d} \cdot \max_i \|c_i\|$ where $(c_i)_i$ is a known basis of the ideal lattice $(L')^{-1}$. In order to obtain a somehow short basis $(c_i)_i$ of $(L')^{-1}$ in polynomial time, the authors of [dBDPW20] use the LLL algorithm in the lattice $(L')^{-1}$, resulting in the 2^d term appearing in the lower bound of ς. However, we observe here that one could spend more time in reducing the ideal lattice L', in order to obtain a shorter basis $(c_i)_i$, and then decrease the lower bound on ς.

Let us assume that instead of running the LLL algorithm in $(L')^{-1}$, one uses an ideal-HSVP solver in the ideal lattice $(L')^{-1}$, with approximation factor $2^{d/\beta}$ for some integer $\beta \geqslant 1$. This can be performed in time $T_\beta^{\text{id-HSVP}}$. Let $v \in (L')^{-1}$ be the obtained short vector. From v, one can compute a basis $(c_i)_i$ of $(L')^{-1}$ such that

$$\begin{aligned}\max_i \|c_i\| &\leqslant \delta_K \cdot \|v\| \\ &\leqslant 2^{d/\beta} \cdot \delta_K \sqrt{d} \cdot \Delta_K^{1/(2d)} \cdot \mathcal{N}(L')^{-1/d} \\ &\leqslant 2 \cdot 2^{d/\beta} \cdot \delta_K \sqrt{d} \cdot \Delta_K^{1/(2d)} \quad \text{(recall that } \mathcal{N}(L')^{-1} \leqslant 2\text{)}.\end{aligned}$$

Using this smaller basis, the lower bound on ς becomes

$$\varsigma \geqslant 2 \cdot 2^{d/\beta} \cdot \delta_K \cdot d \log d \cdot \Delta_K^{1/(2d)},$$

where the $\log(d)$ is here to ensure that the other lower bound on ς is also always satisfied.

[4] Compared to [dBDPW20], we are taking $L' = L \cdot \mathcal{N}(\prod_i \mathfrak{p}_i \cdot \mathfrak{a})^{-1/d}$, instead of multiplying ς by this same quantity; this does not change anything to the reasoning. Recall also that compared to [dBDPW20], we want to sample v in $(L')^{-1}$ instead of L' (in order to obtain an integral ideal).

Taking ς as small as possible, we finally obtain an upper bound

$$N = \left(6 \cdot 2^{d/\beta} \cdot d^{1.5} \log d \cdot \delta_K \cdot \Delta_K^{1/(2d)}\right)^d$$

on the norms of the ideals sampled from $\mathcal{D}_{\varsigma,M}^{\text{perfect}}$. Note however that the run time of the reduction is now $T_\beta^{\text{id-HSVP}} + \text{poly}(\log \varsigma, \text{size}(I), \log \Delta_K)$ (where I is the input ideal). If N is given as input as in the statement of Theorem 4.4, then one can choose

$$\beta = \left\lceil \frac{d}{\log\left(N^{1/d}/(6 \cdot d^{1.5} \log d \cdot \delta_K \cdot \Delta_K^{1/(2d)})\right)} \right\rceil.$$

Note that the lower bound on N in the theorem statement ensures that the denominator of the expression above is always $\geqslant 1$, so that $\beta \leqslant d$. This completes the proof of the first part of the theorem.

Finally, let us prove that the success probability can be decreased by a factor at most $2^{-\Omega(d)}$. In the proof of [dBDPW20, Theorem 5], all the probability losses are of the form $2^{-\Omega(d)}$, except at the very end, when one wants to upper bound $\|x\|_\infty \cdot \|x\|_\infty^{-1}$. In [dBDPW20, Algorithm 2], x is sampled such that $\text{Log}(x)$ follows a continuous Gaussian distribution of parameter $1/\log(d)^2$. In the proof of [dBDPW20, Theorem 5], the authors conclude that with probability at least $1 - d^{-\omega(1)}$, it holds that $\|\text{Log}(x)\|_\infty \leqslant 1$, and so they have a nice bound on $\|x\|_\infty \cdot \|x\|_\infty^{-1}$. In order to increase this probability, we can simply sample $\text{Log}(x)$ from a Gaussian distribution with parameter $3/d$, instead of $1/\log(d)$. Changing the value of this standard deviation has no impact on the reduction, except on the choices of B and n needed for the random walk to produce uniform ideals. One can see in [dBDPW20, Corollary 3.4] that taking the standard deviation to be $3/d$ is allowed, and does not change the value of B and n compared to a standard deviation $1/(\log d)^2$. Hence, we obtain the same bound on γ' as in [dBDPW20, Theorem 5], in the case of arbitrary number fields.[5] Note that the standard deviation $3/d$ is not allowed in [dBDPW20, Corollary 3.5], so we cannot obtain a better bound on γ' in the special case of prime power cyclotomic fields (as in [dBDPW20, Theorem 5]). This is not really important in our case, and we prefer to keep a unique bound for γ', which holds for all number fields.

Let us now prove the second part of the theorem, i.e., that one can sample from $\mathcal{D}_N^{\text{id-HSVP}}$ in polynomial time and produce a somehow short vector of the sampled ideal at the same time.

Let us start again with the high level idea. In order to sample an ideal from the average-case distribution, we will simply run the reduction algorithm, starting with the ideal $I = R$. In other words, we sample n random prime ideals $\mathfrak{p}_i$ of norms $\leqslant B$; we define $L = \prod_{i=1}^n \mathfrak{p}_i$; we sample v in L^{-1} and we output $J = v \cdot L$. In fact, we choose $n = 1$. It was shown in [dBDPW20] that as long as B is

[5] This value of γ' is slightly modified in our statement, since we do not consider exactly the same definition of ideal-HSVP as in [dBDPW20].

sufficiently large, this will achieve the same distribution as taking B smaller and n larger. So we will have $L = \mathfrak{p}$ a prime ideal of norm at most B. Using Lemma C.1, we will sample $\mathfrak{p}$ together with a short vector w'. Sampling $v \in \mathfrak{p}^{-1}$ will be easy because we know a short basis of $\mathfrak{p}^{-1}$, since $R \subset \mathfrak{p}^{-1}$. So we will be able to sample $v \in L^{-1}$ quite small. And we know that $w := v \cdot w'$ will be a small element of the output ideal $J = v \cdot L$.

More formally, we know from [dBDPW20, Th. 3.3] that there exists some $B_0 = \mathrm{poly}(\Delta_K, d^d)$ such that the random walk with parameters $n = 1$, $B = B_0$ and $s = 1/(\log(d))^2$ provides a distribution that is within statistical distance 2^{-d} from uniform in the Arakelov class group. Hence, if we run [dBDPW20, Algorithm 2] (with the modifications described above) with $n = 1$ and $B = B_0$, then the distribution of J will be the same as before. Let us run [dBDPW20, Algorithm 2] with $\mathfrak{a} = R$ and $n = 1$. The lattice L computed at Step 4 will be of the form $L = x \cdot \mathfrak{p}$, for some $\mathfrak{p}$ uniformly chosen among prime ideals of norms $\leqslant B$ and $x \in K_{\mathbb{R}}$ such that $\mathcal{N}(x) \in [1/2, 2]$ and $\|x\|, \|x^{-1}\| \leqslant 2$. Let us sample $\mathfrak{p}$ together with a short vector $w' \in \mathfrak{p}$ as in Lemma C.1. At Step 7 of [dBDPW20, Algorithm 2], we want to sample v in L^{-1}, from a distribution with standard deviation that is as small as possible. We know that $\|x^{-1}\| \leqslant 2$ and that $R \subset \mathfrak{p}^{-1}$. Hence, we can compute a basis of L^{-1} with vectors of norms at most $2d \cdot \delta_K$ (using for instance [MG02, Le. 7.2] to compute a short basis from an arbitrary basis and a short set of linearly independent vectors). This implies that we can sample v in L^{-1} from a Gaussian distribution with standard deviation parameter $\varsigma' = \mathcal{N}(\mathfrak{p})^{-1/d}\varsigma$ as small as $2d^{1.5} \cdot \delta_K$. Overall, this means that one can take ς as small as $2d^{1.5} \cdot \delta_K \cdot B_0^{1/d}$. Recall that the parameter M of [dBDPW20, Th. 2] is chosen to be equal to $2\sqrt{d}\varsigma$, and that we have seen that the norm of the output ideal is bounded from above by $(M + \sqrt{d}\varsigma)^d = (3\sqrt{d}\varsigma)^d$. Hence, we can sample from $\mathcal{D}_N^{\text{id-HSVP}}$ for N as small as $\mathrm{poly}(\Delta_K, d^d, \delta_K^d)$.

Moreover, we know that the vector $w := v \cdot x \cdot w'$ is a vector of the output ideal $J = v \cdot x \cdot \mathfrak{p}$. By Lemma C.1, this vector has norm at most

$$\|w\| \leqslant 2\|v\| \cdot 2^{\sqrt{\log B}} \cdot d \cdot \delta_K \cdot B^{1/d},$$

if we use the sampling algorithm without factoring oracle, and

$$\|w\| \leqslant 2\|v\| \cdot d^{1.5} \cdot \delta_K \cdot \Delta_K^{1/d} \cdot B^{1/d},$$

if we use the sampling algorithm with factoring oracle. The element v is sampled from a Gaussian distribution with parameter $\varsigma' = \mathcal{N}(\mathfrak{p})^{-1/d} \cdot \varsigma$ and center $c' = \mathcal{N}(\mathfrak{p})^{-1/d} \cdot c \in K_{\mathbb{R}}$ with $|\sigma_i(c)| = M = 2\sqrt{d}\varsigma$ for all $i \leqslant d$. Hence, we know that with overwhelming probability, it holds that $\sigma_i(v) \in [\sqrt{d} \cdot \varsigma', 3\sqrt{d} \cdot \varsigma']$ for all embeddings σ_i. This implies in particular that

$$\|v\| \leqslant \sqrt{d} \cdot (3\sqrt{d} \cdot \varsigma') \leqslant 3\sqrt{d} \cdot |\mathcal{N}(v)|^{1/d}.$$

Using the fact that $\mathcal{N}(J) = \mathcal{N}(vL) \geqslant |\mathcal{N}(v)|/2$ (since $\mathcal{N}(L) \geqslant \mathcal{N}(x) \geqslant 1/2$), we conclude that $\|w\| \leqslant \mathrm{poly}(d, \delta_K, \Delta_K^{1/d}, 2^{\sqrt{\log \Delta_K + d \log d}}) \cdot \mathcal{N}(J)^{1/d}$, and that this

can be decreased to $\|w\| \leqslant \mathrm{poly}(d, \delta_K, \Delta_K^{1/d}) \cdot \mathcal{N}(J)^{1/d}$ if we have a factoring oracle.

It remains to show that one can also achieve $\|w\| \leqslant \mathrm{poly}(d, \delta_K, \Delta_K^{1/d}) \cdot \mathcal{N}(J)^{1/d}$ if N is sufficiently large. To do so, we will use the same algorithm as above, except that we will take $L_1 = x^{-1} \cdot \mathfrak{p}^{-1}$ instead of $L = x \cdot \mathfrak{p}$. The random walk theorem [dBDPW20, Th. 3.3] states that $L/\det(L)^{1/d}$ is a uniform ideal of norm 1, hence its inverse also is a uniform ideal of norm 1. Therefore, replacing L by its inverse $L_1 = L^{-1}$ leads to the same output distribution. Recall that we know a very short vector of L^{-1} but only a moderately short vector of L. This is why we were able to sample v in L^{-1} very short, but could only compute a moderately short element of the output ideal $J = v \cdot L$. Taking $L_1 = L^{-1}$ instead of L means that we know a very short vector of L_1, and so we will be able to produce a very short vector w of the output ideal $J = vL_1$. However, we only know a moderately short vector of L_1^{-1}, and so we will only be able to sample a moderately short v in L_1^{-1}.

More concretely, using Lemma C.1, we sample in classical polynomial time the ideal $\mathfrak{p}$ together with a somehow short vector $w' \in \mathfrak{p}$ such that $\|w'\| \leqslant 2^{\sqrt{\log B}} \cdot d \cdot \delta_K \cdot B^{1/d}$. We know that $x \cdot w'$ is a vector of L_1^{-1} of norm $\|xw'\| \leqslant 2\|w'\|$. From this, we can obtain a basis of L_1^{-1} with vectors of norms at most $2\sqrt{d} \cdot \delta_K \cdot \|w'\|$, and so we can sample in L_1^{-1} from a Gaussian distribution with parameter ς' as small as $2d \cdot \delta_K \cdot \|w'\|$. This means that we can take $\varsigma = \varsigma' \cdot \mathcal{N}(\mathfrak{p})^{-1/d}$ as small as $2d \cdot \delta_K \cdot \|w'\|$, and so N as small as $\mathrm{poly}(2^{\sqrt{\log \Delta_K + d \log d}}, \Delta_K^{1/d}, \delta_K, d)^d$. Let v be the vector that we sample in L_1^{-1}. We have already seen that thanks to the center c' and by choice of M, it holds that $\|v\| \leqslant 3\mathcal{N}(v)^{1/d}$. Using this and the fact that $w := vx^{-1} \in vL_1 = J$, we obtain a vector $w \in J$ of norm

$$\|w\| \leqslant 2\|v\| \leqslant 6\mathcal{N}(v)^{1/d} \leqslant 6 \cdot \mathcal{N}(L_1)^{-1/d} \cdot \mathcal{N}(J)^{1/d} \leqslant 12 \cdot B^{1/d} \cdot \mathcal{N}(J)^{1/d}.$$

This concludes the proof of the second part of the theorem. □

C.2 Reducing average-case id-HSVP to average-case NTRU – Proof of Theorem 4.9

We first observe that thanks to the lower bound on $\sqrt{q}/\gamma$, we have $N \geqslant (12d^{1.5} \cdot \log d \cdot \delta_K \cdot \Delta_K^{1/(2d)})^d$, and so the distribution $\widetilde{\mathcal{D}}_N^{\text{id-HSVP}}$ is well-defined.

On input an ideal I sampled from $\widetilde{\mathcal{D}}_N^{\text{id-HSVP}}$, the reduction proceeds as follows. It defines $I' = \lfloor (N/\mathcal{N}(I))^{1/d} \rfloor \cdot I$; it runs Algorithm 4.1 on (I, q) to obtain an NTRU instance $h \in R_q$; it calls the $(\mathcal{D}_{q,\gamma}^{\text{NTRU}}, \gamma, \gamma', q)$-$\text{NTRU}_{\text{vec}}$ algorithm $\mathcal{A}$ on h to obtain a pair (f, g); and, finally, it outputs the element $v := g/\lfloor (N/\mathcal{N}(I))^{1/d} \rfloor$. The reduction is indeed ppt (by Lemma 4.3) and makes a single call to the NTRU_{vec} oracle. We are left to compute a lower bound on its success probability.

As I is distributed according to $\widetilde{\mathcal{D}}_N^{\text{id-HSVP}}$ and the definition of N in the theorem is consistent with the one in Algorithm 4.2, the element h is exactly distributed from $\mathcal{D}_{q,\gamma}^{\text{NTRU}}$. Therefore, with probability $\mathrm{Adv}(\mathcal{A})$, the pair (f, g) output by $\mathcal{A}$ satisfies $(f, g) \neq (0, 0)$, $g \cdot h = f \bmod q$ and $\|f\|, \|g\| \leqslant \sqrt{q}/\gamma'$. Let us

show that for such a trapdoor, the reduction output an element v which is a solution to γ_{hsvp}-id-HSVP for ideal I, which will complete the proof.

We have

$$\|f\|_\infty, \|g\|_\infty \leqslant \frac{\sqrt{q}}{\gamma'} < \sqrt{d} \cdot \sqrt{q} \cdot \gamma \leqslant \frac{q}{d \cdot \delta_K \cdot \Delta_K^{1/(2d)} \cdot \mathcal{N}(I')^{1/d}},$$

where we used the inequalities $\gamma \cdot \gamma' \cdot \sqrt{d} > 1$ and $\mathcal{N}(I')^{1/d} \leqslant N^{1/d} \leqslant \sqrt{q}/(\gamma \cdot d^{1.5} \cdot \delta_K \cdot \Delta_K^{1/(2d)})$. We can hence use Lemma 4.3 to conclude that $g \in I' \backslash \{0\}$, which implies that $v \in I \backslash \{0\}$.

It finally remains to bound $\|g\|$. First, observe that thanks to our scaling, the ideal I' satisfies $\mathcal{N}(I')^{\frac{1}{d}} \geqslant N^{\frac{1}{d}}/2$. We also have $N^{1/d} \geqslant \sqrt{q}/(2\gamma \cdot d^{1.5} \cdot \delta_K \cdot \Delta_K^{1/(2d)})$. Combining everything, and using the definition of γ_{hsvp}, we see that

$$\|g\| \leqslant \frac{\sqrt{q}}{\gamma'} = \gamma_{\text{hsvp}} \frac{\sqrt{q}}{\gamma 4 d \delta_K} \leqslant \gamma_{\text{hsvp}} \sqrt{d} \Delta_K^{\frac{1}{2d}} \cdot \mathcal{N}(I')^{\frac{1}{d}}. \quad \square$$

D Missing proofs from Section 5

D.1 Proof of Lemma 5.5

Lemma 5.5. *Let $(f, g) \in R^2 \backslash \{(0,0)\}$ and $I = \langle f \rangle + \langle g \rangle$. Let $c_1, c_2 \in \sigma_1(K_\mathbb{R})$ (which is either $\mathbb{R}$ or $\mathbb{C}$), $s_0 > 0$ and $s \geqslant \sqrt{d}\delta_K \cdot (\|f\| + \|g\|)$.*

Given $t \in \sigma_1(K_\mathbb{R})$, we define $\psi(t) \in K_\mathbb{R}$ as $(t, 0, \ldots, 0)^T \in K_\mathbb{R}$ if σ_1 is a real embedding and as $(t/\sqrt{2}, \overline{t}/\sqrt{2}, 0, \ldots, 0)^T \in K_\mathbb{R}$ if σ_1 is a complex embedding with $\sigma_2 = \overline{\sigma_1}$.

Let $\mathcal{D}$ be the output distribution of the following algorithm:

- *sample $c_0 \leftarrow D_{\sigma_1(K_\mathbb{R}), s_0, 0}$;*
- *sample $x \leftarrow D_{R, s, \psi(c_0 \cdot c_1)}$ and $y \leftarrow D_{R, s, \psi(c_0 \cdot c_2)}$;*
- *return $x \cdot f + y \cdot g \in I$.*

Then it holds that $\mathcal{D} \approx_{2^{-\Omega(d)}} D_{I, \mathbf{S}, 0}$, where $\mathbf{S}$ is a diagonal matrix with

$$S_{11} = \sqrt{s_0^2 \cdot |c_1\sigma_1(f) + c_2\sigma_1(g)|^2 + s^2 \cdot (|\sigma_1(f)|^2 + |\sigma_1(g)|^2)}$$

$$S_{22} = \begin{cases} S_{11} & \text{if } \sigma_1 \text{ is a complex embedding} \\ s \cdot \sqrt{|\sigma_2(f)|^2 + |\sigma_2(g)|^2} & \text{if } \sigma_1 \text{ is a real embedding} \end{cases}$$

$$S_{ii} = s \cdot \sqrt{|\sigma_i(f)|^2 + |\sigma_i(g)|^2} \qquad \text{for } i \geqslant 3.$$

In order to prove this lemma, we will use the two lemmas below, which are very similar to [LSS14, Th. 5.1] and [Pei10, Th. 3.1], respectively.

Lemma D.1 (Adapted from [LSS14, Th. 5.1]). *Let $(f, g) \in R^2 \backslash \{(0,0)\}$ and $I = \langle f \rangle + \langle g \rangle$. Let $c_1, c_2 \in K_\mathbb{R}$ and $s \geqslant \sqrt{d}\delta_K \cdot (\|f\| + \|g\|)$. Then*

$$D_{R, s, c_1} \cdot f + D_{R, s, c_2} \cdot g \approx_{2^{-\Omega(d)}} D_{I, \mathbf{S}, c},$$

where $c = c_1 \cdot f + c_2 \cdot g \in K_\mathbb{R}$ and $\mathbf{S}$ is a diagonal matrix with $S_{i,i} = s \cdot \sqrt{|\sigma_i(f)|^2 + |\sigma_i(g)|^2}$.

This lemma is very similar to [LSS14, Th. 5.1], except that it is stated for arbitrary number fields, the ideal I needs not be principal and the elements x and y may be sampled with a non-zero center. In order to prove the lemma, we use the same approach as [LSS14, Th. 5.1].

Proof. Let us define $A = \{(x, y)^T \in R^2 \,|\, xf + yg = 0\}$, which is a rank-1 R-module contained in R^2. Via the canonical embedding, this can be viewed as a d-dimensional lattice living in $\mathbb{C}^{2d}$. The first step is to show that if $s \geqslant \max(\eta_{2^{-\Omega(d)}}(A), \eta_{2^{-\Omega(d)}}(R))$, then Lemma D.1 holds; and the second step will be to show that we indeed have $s \geqslant \max(\eta_{2^{-\Omega(d)}}(A), \eta_{2^{-\Omega(d)}}(R))$. For the first step, the authors of [LSS14] used [AGHS13, Le. 10], which we will also need to adapt to our setting.

Let us fix some $z \in I$. To prove the desired upper bound on the statistical distance, it suffices to show that:

$$\Pr_{\substack{x \leftarrow D_{R,s,c_1} \\ y \leftarrow D_{R,s,c_2}}} \Big(x \cdot f + y \cdot g = z \Big) \;\in\; D_{I,\mathbf{S},c}(z) \cdot \Big[1 - 2^{-\Omega(d)}, 1 + 2^{-\Omega(d)}\Big],$$

Let A_z denote the shifted lattice $A_z = \{(x, y)^T \in R^2 \,|\, xf + yg = z\}$. With this notation, we have

$$\begin{aligned}
\Pr_{\substack{x \leftarrow D_{R,s,c_1} \\ y \leftarrow D_{R,s,c_2}}} \Big(x \cdot f + y \cdot g = z \Big) &= \sum_{(x,y)^T \in A_z} D_{R,s,c_1}(x) \cdot D_{R,s,c_2}(y) \\
&= \sum_{(x,y)^T \in A_z} \frac{\exp\Big(- \pi \frac{\|x - c_1\|^2 + \|y - c_2\|^2}{s^2} \Big)}{\rho_s(R - c_1) \cdot \rho_s(R - c_2)} \\
&= \eta \cdot \Big(\frac{\det(R)}{s^d} \Big)^2 \cdot \sum_{(x,y)^T \in A_z} \exp\Big(- \pi \frac{\|(x, y)^T - (c_1, c_2)^T\|^2}{s^2} \Big),
\end{aligned}$$

for some $\eta \in [1 - 2^{-\Omega(d)}, 1 + 2^{-\Omega(d)}]$. For the last equality, we used Lemma A.1 and the fact that $s \geqslant \eta_{2^{-\Omega(d)}}(R)$.

Let us now define $T = \mathrm{Span}_{K_\mathbb{R}}(A)$, which is an $\mathbb{R}$-vector space of dimension d contained in $K_\mathbb{R}^2$. Note that $\mathrm{Span}_{K_\mathbb{R}}((\bar{f}, \bar{g})^T)$ is also a d-dimensional $\mathbb{R}$-vector space contained in $K_\mathbb{R}^2$ (because $K_\mathbb{R}$ is stable by complex conjugation, as opposed to K which may not be), and that it is orthogonal to T, where orthogonality is with respect to the Hermitian inner product over $K_\mathbb{R}^2$. Since $K_\mathbb{R}^2$ has dimension $2d$ over $\mathbb{R}$, we conclude that $K_\mathbb{R}^2 = T \oplus_\perp \mathrm{Span}_{K_\mathbb{R}}((\bar{f}, \bar{g})^T)$. In other words, given any vector $(u, v)^T \in K_\mathbb{R}^2$, there exists $\alpha \in K_\mathbb{R}$ and $(u', v')^T \in T$ such that $(u, v) = \alpha(\bar{f}, \bar{g}) + (u', v')$. Moreover, we know that $\alpha = (fu + gv)/(f\bar{f} + g\bar{g})$.

Let us fix some $(x, y)^T \in A_z$. We know that

$$\|(x, y)^T - (c_1, c_2)^T\|^2 = \left\| \frac{(fx + gy) - (fc_1 + gc_2)}{f\bar{f} + g\bar{g}} (\bar{f}, \bar{g})^T \right\|^2 + \|(x_0, y_0)^T\|^2,$$

for some $(x_0, y_0)^T \in T$. By definitions of $\mathbf{S}$ and c, and using the fact that $(x, y)^T \in A_z$, the first term is exactly $s^2 \cdot \|\mathbf{S}^{-1}(z - c)\|^2$. Observe also that when $(x, y)^T$

varies in A_z, then the second term varies in $(x_0, y_0)^T + A$. Hence, we obtain that

$$\sum_{(x,y)^T \in A_z} \exp\Big(-\pi \frac{\|(x,y)^T - (c_1,c_2)^T\|^2}{s^2}\Big)$$
$$= \exp(-\pi \cdot \|\mathbf{S}^{-1}(z-c)\|^2) \cdot \rho_s(A + (x_0,y_0)^T)$$
$$= \eta_1 \cdot \rho_{\mathbf{S}}(z-c) \cdot \frac{s^d}{\det(A)},$$

for some $\eta_1 \in [1 - 2^{-\Omega(d)}, 1 + 2^{-\Omega(d)}]$. For the last equality, we used Lemma A.1 and the fact that $s \geqslant \eta_{2^{-\Omega(d)}}(A)$.

Combining everything, we obtain that

$$\Pr_{\substack{x \leftarrow D_{R,s,c_1} \\ y \leftarrow D_{R,s,c_2}}} \Big(x \cdot f + y \cdot g = z\Big) = \eta \cdot \eta_1 \cdot \frac{(\det(R))^2}{s^d \cdot \det(A)} \cdot \rho_{\mathbf{S}}(z-c).$$

Since the term $(\det(R))^2/(s^d \cdot \det(A))$ is independent from z, we conclude that the probability that $xf + yg = z$ is within a factor $1 \pm 2^{-\Omega(d)}$ from $\rho_{\mathbf{S}}(z-c)/\rho_{\mathbf{S}}(I-c) = D_{I,\mathbf{S},c}(z)$, as desired.

It remains to show that the lower bound on s implies that $s \geqslant \max(\eta_{2^{-\Omega(d)}}(A), \eta_{2^{-\Omega(d)}}(R))$. We know thanks to Equation (2.1) that for some $\varepsilon = 2^{-\Omega(d)}$, it holds that $\eta_\varepsilon(A) \leqslant \sqrt{d}\lambda_d(A)$ and $\eta_\varepsilon(R) \leqslant \sqrt{d}\lambda_d(R)$. Regarding R, we know that $\lambda_d(R) \leqslant \delta_K \|f\|$ since $f \in R$, and so we indeed have $s \geqslant \eta_\varepsilon(R)$. Regarding A, we know that all the vectors $r_j \cdot (-g, f)^T$ are in A and are linearly independent (recall that the r_j's form a basis of R), hence we have that $\lambda_d(A) \leqslant \delta_K \cdot (\|f\| + \|g\|)$, and the lower bound on s implies that $s \geqslant \eta_\varepsilon(A)$ as well. □

Lemma D.2. *Let $I \subset K_{\mathbb{R}}$ be a lattice of dimension d and $\alpha \in \sigma_1(K_{\mathbb{R}})$. Let $s_0 > 0$ and $\mathbf{S} \in \mathrm{GL}_d(\mathbb{R})$ be a diagonal matrix with positive diagonal coefficients $(S_{jj})_j$ such that $S_{ii} = S_{jj}$ if $\sigma_i = \overline{\sigma_j}$ forms a pair of conjugate complex embeddings. Assume that $1 \geqslant \eta_\varepsilon(\mathbf{S}^{-1}I)$ for some $\varepsilon = 2^{-\Omega(d)}$. Let $\psi : \sigma_1(K_{\mathbb{R}}) \to K_{\mathbb{R}}$ be as defined in Lemma 5.5. Let $\mathcal{D}$ be the output distribution of the following algorithm:*

- *sample $c_0 \leftarrow D_{\sigma_1(K_{\mathbb{R}}),s_0,0}$;*
- *return $z \leftarrow D_{I,\mathbf{S},\psi(c_0 \cdot \alpha)}$.*

Then it holds that $\mathcal{D} \approx_{2^{-\Omega(d)}} D_{I,\mathbf{S}',0}$, where $\mathbf{S}'$ is a diagonal matrix with the following positive diagonal coefficients

$$S'_{11} = \sqrt{S_{11}^2 + s_0^2|\alpha|^2}$$
$$S'_{22} = S_{11} \quad \text{if } \sigma_1 \text{ is a complex embedding}$$
$$S'_{22} = S_{22} \quad \text{if } \sigma_1 \text{ is a real embedding}$$
$$S'_{ii} = S_{ii} \quad \text{for } \ i \geqslant 3.$$

This result is very close to an instantiation of [Pei10, Th. 3.1], except that we want to sample c_0 in a space of dimension smaller than d (equivalently, we could

sample c_0 in $K_{\mathbb{R}}$ with covariance matrix $s_0 \cdot \sqrt{\psi(\alpha) \cdot \psi(\overline{\alpha})^T}$ but this matrix is not invertible, and [Pei10, Th. 3.1] requires the covariance matrices to be invertible). We hence prove the result for this specific setting. The proof follows the same steps as the one of [Pei10, Th. 3.1].

Proof. For the proof, we assume that σ_1 is a complex embedding and that $\sigma_2 = \overline{\sigma_1}$. The case of a real embedding is very similar, except that we only have one coordinate to handle (and no scaling factor $1/\sqrt{2}$). Let us fix some $z \in I$. We have

$$\begin{aligned}\Pr(\mathcal{D} = z) &= \int_{c_0 \in \mathbb{C}} D_{I,\mathbf{S},\psi(c_0\cdot\alpha)}(z) \cdot D_{\mathbb{C},s_0,0}(c_0)\mathrm{d}c_0 \\ &= \frac{\eta \cdot \det(\mathbf{S}^{-1}I)}{s_0^2} \cdot \int_{c_0 \in \mathbb{C}} \exp\Big(-\pi\big(\|\mathbf{S}^{-1}(z - \psi(c_0 \cdot \alpha))\|^2 + |s_0^{-1}c_0|^2\big)\Big)\mathrm{d}c_0,\end{aligned}$$

for some $\eta \in [1-2^{-\Omega(d)}, 1+2^{-\Omega(d)}]$. For the second equality, we used Lemma A.1 and the fact that $\eta_\varepsilon(\mathbf{S}^{-1}I) \leqslant 1$ for some $\varepsilon = 2^{-\Omega(d)}$.

Let us focus on the term $\|\mathbf{S}^{-1}(z - \psi(c_0 \cdot \alpha))\|^2$. This is equal to

$$2S_{11}^{-2}|\sigma_1(z) - c_0\alpha/\sqrt{2}|^2 + \sum_{j \geqslant 3} S_{jj}^{-2}|\sigma_j(z)|^2,$$

where we used the fact that $S_{11} = S_{22}$ and that $\sigma_1(z) = \overline{\sigma_2(z)}$ since we are in the complex embedding case. By term rearrangement, we have:

$$\begin{aligned}&2S_{11}^{-2}|\sigma_1(z) - c_0\alpha/\sqrt{2}|^2 + |s_0^{-1}c_0|^2 \\ &\qquad = 2\big(\sqrt{S_{11}^2 + |s_0 \cdot \alpha|^2}\big)^{-1} \cdot |\sigma_1(z)|^2 + s_1^{-2} \cdot |c_0 - c_1|^2,\end{aligned}$$

for $s_1 = 1/\sqrt{s_0^{-2} + S_{11}^{-2}|\alpha|^2} > 0$ that depends only on α, s_0 and S_{11} and $c_1 = \sqrt{2}s_1^2 S_{11}^{-2}|\sigma_1(z)\alpha| \in \mathbb{C}$ that does not depend on c_0. Using the fact that $2|\sigma_1(z)|^2 = |\sigma_1(z)|^2 + |\sigma_2(z)|^2$ and that $S_{11} = S_{22}$, we finally see that

$$\|\mathbf{S}^{-1}(z - \psi(c_0 \cdot \alpha))\|^2 + |s_0^{-1}c_0|^2 = \|(\mathbf{S}')^{-1}z\|^2 + s_1^{-2} \cdot |c_0 - c_1|^2.$$

Going back, we obtain that

$$\begin{aligned}\mathcal{D}(z) &= \frac{\eta \cdot \det(\mathbf{S}^{-1}I)}{s_0^2} \cdot \exp\Big(-\pi\|(\mathbf{S}')^{-1}z\|^2\Big) \cdot \int_{c_0 \in \mathbb{C}} \exp\Big(-\pi s_1^{-2} \cdot |c_0 - c_1|^2\Big)\mathrm{d}c_0 \\ &= \frac{\eta \cdot \det(\mathbf{S}^{-1}I)}{s_0^2 \cdot s_1^2} \cdot \exp\Big(-\pi\|(\mathbf{S}')^{-1}z\|^2\Big),\end{aligned}$$

where we used the fact that c_1 does not depend on c_0 for the last equality. Since s_1 does not depend on z, we finally see that $\mathcal{D}(z)$ is within a factor $1 \pm 2^{-\Omega(d)}$ from $\rho_{\mathbf{S}'}(z)$ (up to a uniform normalization factor), which proves that $\mathcal{D} \approx_{2^{-\Omega(d)}} D_{I,\mathbf{S}',\mathbf{0}}$ as desired. □

Proof (Proof of Lemma 5.5). Let z be the random variable $xf + yg$ produced by the algorithm. Since $s \geqslant \sqrt{d}\delta_K \cdot (\|f\| + \|g\|)$, we can apply Lemma D.1 and we know that the conditional distribution of z once c_0 has been fixed is statistically close to $D_{I,\mathbf{S},c}$, where

$$c = f \cdot \psi(c_0 c_1) + g \cdot \psi(c_0 c_2) = \psi\Big(c_0 \cdot (c_1\sigma_1(f) + c_2\sigma_1(g))\Big),$$

and $\mathbf{S}$ is a diagonal matrix with positive diagonal coefficients equal to $S_{ii} = s\sqrt{|\sigma_i(f)|^2 + |\sigma_i(g)|^2}$.

We now want to apply Lemma D.2 with $\alpha = c_1\sigma_1(f) + c_2\sigma_1(g) \in \sigma_1(K_{\mathbb{R}})$. For this, we need to check that the assumption $1 \geqslant \eta_{2^{-\Omega(d)}}(\mathbf{S}^{-1}I)$ is satisfied. From Equation (2.1) we know that $\eta_{2^{-\Omega(d)}}(\mathbf{S}^{-1}I) \leqslant \sqrt{d} \cdot \lambda_d(\mathbf{S}^{-1}I)$. We also know that the $\mathbf{S}^{-1}r_j f$'s are linearly independent vectors of $\mathbf{S}^{-1}I$, which all have norms $\leqslant \delta_K \cdot \sqrt{d}/s$. Combining both, we obtain that $\eta_{2^{-\Omega(d)}}(\mathbf{S}^{-1}I) \leqslant \delta_K \cdot d/s$. Recall that by assumption, we have $s \geqslant \sqrt{d}\delta_K \cdot (\|f\| + \|g\|)$, hence it suffices to prove that $(\|f\| + \|g\|) \geqslant \sqrt{d}$. This is indeed the case, since $(f, g) \in R^2\backslash(0, 0)$ and any non-zero element $x \in R$ satisfies $\|x\| \geqslant \sqrt{d} \cdot \mathcal{N}(x)^{1/d} \geqslant \sqrt{d}$ (the first inequality follows from the inequality of arithmetic and geometric means applied to the $|\sigma_i(x)|^2$'s). We conclude that $1 \geqslant \eta_{2^{-\Omega(d)}}(\mathbf{S}^{-1}I)$ as required. Applying Lemma D.2 completes the proof. □

D.2 Proof of Theorem 5.4

Let $h \in R_q$ be B-well-behaved. We start by bounding $\mathrm{size}(h_K)$ from above. We know that $h_K = f/g \in K$ for some $\|f\|, \|g\| \leqslant B$ and $f, g \in R$. This implies that $h_K \cdot \mathcal{N}(g) \in R$, and since $\mathcal{N}(g) \in \mathbb{Z}$, we see that $\mathrm{size}(h_K) \leqslant d \log |\mathcal{N}(g)| + \mathrm{size}(h_K \cdot \mathcal{N}(g))$. Since $\|g\| \leqslant B$, we know that $|\mathcal{N}(g)| \leqslant B^d$. Further, as the basis $(r_1, \ldots, r_d)$ of R is LLL-reduced, for any $r \in R$, it holds that $\mathrm{size}(r) = \mathrm{poly}(d, \log \|r\|)$. Combining everything, we obtain that $\mathrm{size}(h_K) = \mathrm{poly}(d, \log B)$.

Using Lemma 2.5, we know that there exists some $\ell_0 = \mathrm{poly}(\log B, d, \log \delta_K, \log \|\Phi\|)$ such that given $\widetilde{\sigma_1(h_K)} \in \mathbb{C}$ with $|\widetilde{\sigma_1(h_K)} - \sigma_1(h_K)| \leqslant 2^{-\ell_0}$, one can recover h_K as a rational linear combination of the r_j's in time $\mathrm{poly}(\ell_0)$ (note that this choice of ℓ_0 works for any h that is B-well-behaved).

Let us now describe an algorithm for solving $(\mathcal{D}^s, \gamma, q)$-$\mathrm{NTRU}_{\mathrm{mod}}$. On input some $h \leftarrow \mathcal{D}^s$, the algorithm runs Algorithm $\mathcal{B}$ from Lemma 5.7 with $\ell = \ell_0$, in order to recover some $\widetilde{\sigma_1(h_K)}$. Then, the algorithm uses Lemma 2.5 on $\widetilde{\sigma_1(h_K)}$ to recover h_K exactly.

Combining Lemmas 5.7 and 2.5, one can see that this $\mathrm{NTRU}_{\mathrm{mod}}$ solver succeeds with probability $\geqslant 1 - 2^{-\Omega(d)}$ as long as $h \in H$ (where H is as defined in Lemma 5.6). We also know from Lemma 5.6 that $\Pr_{h \leftarrow \mathcal{D}^s}(h \in H) \geqslant \mathrm{Adv}(\mathcal{A})/2 - \varepsilon$. Hence, we conclude that the $\mathrm{NTRU}_{\mathrm{mod}}$ solver succeeds with probability $\geqslant (\mathrm{Adv}(\mathcal{A})/2 - \varepsilon)/2$. The run-time of the algorithm follows by observing that $B = \mathrm{poly}(q, d)$ since one can always reduce f and g modulo q. Finally, the values of γ and γ' come from the remark after Definition 5.1 and from Lemma 5.3. □