On the quantum attacks against schemes relying on the hardness of finding a short generator of an ideal in $\mathbb{Q}(\zeta_{p^n})$ (extended abstract)

Jean-François Biasse and Fang Song

University of Waterloo
Institute for Quantum Computing
200 University Avenue West
Waterloo, Ontario, Canada N2L 3G1

Date: August 31st 2015, **Revised**: September 28th 2015

Full version in appendix

1 Introduction

A series of works describe cryptosystems relying on the hardness of finding a small generator of a principal ideal in the ring of integers of $K = \mathbb{Q}(\zeta_{2^n})$. In particular, this problem allows to describe fully homomorphic schemes, such as that of Smart and Vercauteren [13], or the multilinear maps of Garg, Gentry and Halevi [8]. Moreover, these schemes have been described as quantum safe in the absence of quantum attacks against them. This potential for quantum safety was the main appeal to scientists from the CESG for the development of SOLILOQUY, a cryptosystem relying on the hardness of finding a short generator of a principal ideal.

Since then, the CESG has interrupted the SOLILOQUY program because there were indications that it was not as quantum safe as they originally thought. Campbel, Groves and Shepherd [4] released an online draft explaining the design of SOLILOQUY and its apparent weaknesses. Most notably, they observed that finding a short generator of an ideal in the ring of integers of $\mathbb{Q}(\zeta_{2^n})$ polynomially reduced to finding an arbitrary generator (which corresponds to the Principal Ideal Problem - PIP). This fact was rigorously proved by Cramer, Ducas, Peikert and Regev [5] shortly thereafter.

The bottleneck of a key-recovery attack against schemes relying on the hardness of finding a short generator of a principal ideal is the resolution of the PIP. A classical subexponential algorithm was described by Biasse and Fieker for this task [2, 3]. Meanwhile, the draft of Campbel et al. [4] describes a quantum attack and conjectures that it runs in polynomial time. Since then, this conjecture has been retracted by Pinch [12] in a talk on SOLILOQUY on behalf of Campbel, Groves and Shepherd. The possibility of a quantum polynomial time attack has generated a lot of attention, and was cited in [5], as well as in various blogs and discussion forum.

Contribution In this paper, we show how to derive a quantum polynomial time attack from a recent result of Eisenträger, Hallgren, Kitaev and Song [6] and the reduction from short-PIP to PIP of Cramer, Ducas, Peikert and Regev [5]. We focus on the task of finding a generator of a principal ideal in the ring of integers of $\mathbb{Q}(\zeta_{p^n})$. Our contribution is two fold.

On the negative side, we analyze the quantum algorithm mentioned in the draft of Campbel et al. [4], and we highlight the main obstructions to a polynomial run time and we put this into perspective with respect to the current state of the art in quantum computing.

On the positive side, we rigorously prove that we can derive a quantum polynomial time algorithm for the search of the generator of a principal ideal from the recent work of Eisenträger et al. [6].

2 The quantum algorithm of Campbel et al. [4]

Campbel, Groves and Shepherd [4] attempted to describe a quantum polynomial time algorithm for solving the Principal Ideal Problem (PIP) in totally real cyclotomic fields by using essentially the same technical tools than those available to Hallgren in his 2005 paper [10] (which gives a polynomial time solution to the PIP in classes of fixed degree number fields). Combined with the Gentry-Szydlo (classical) attack [9], this solves the PIP in any cyclotomic field. They sketched an attack in [4, Sec. 5], but it was never analyzed.

We give a high level description of the quantum attack of [4] and show how it suffers from the same fundamental obstruction as Hallgren's 2005 algorithm for solving the Hidden Subgroup Problem [10] which is only known to run in polynomial time in fixed dimension. The quantum PIP algorithm of Campbel et al. follows the usual two-step strategy consisting of first reducing the PIP to the task of finding the periods of a function (which is similar to the Hidden Subgroup Problem, except that the function they use does not fall into the formal definition for the HSP). This means exhibiting a function $f : G \subseteq \mathbb{R}^m \to \{\text{Lattices over } \mathbb{R}^n\}$ for some subgroup G and $m, n \in \mathbb{Z}_{>0}$ such that $f(x) = f(y)$ if and only if $x = y \bmod \Lambda$ for a lattice $\Lambda \subseteq \mathbb{R}^m$ whose knowledge answers the original problem (the PIP in this case). Then the second step consists of finding the periods of f.

Proposition 1 (exponential run time). *Assuming we use the same analysis as in [10, Sec 3.2] the run time of the overall algorithm is at least* 2^r *where* $r \geq \deg(K)/2$.

3 An algorithm for the PIP in totally real fields

It does not seem that the HSP algorithm proposed by Hallgren [10] (and used in the draft of Campbel et al. [4]) allows us to solve the Hidden Subgroup Problem in $\mathbb{R}^m$ in polynomial time in m. However, recent work from Eisenträger, Hallgren, Kitaev and Song [6] developed a new framework for HSP in $\mathbb{R}^m$, which admits an efficient quantum algorithm even for large values of m. They illustrated this by computing the unit group of a number field of arbitrary degree in polynomial time. The algorithm described in [6] returns generators of a secret subgroup H of $\mathbb{R}^m$ (where m depends on the degree n of the field) hidden in the periods of a function $f : \mathbb{R}^m \to \{\text{quantum states}\}$. Eisenträger et al. showed in [6, Th. 6.1] how to recover generators of a secret subgroup $H \subseteq \mathbb{R}^m$ in polynomial time in m if there is a function f satisfying the following properties:

1. f is periodic on H, that is $f(x+u) = f(x)\ \forall x \in \mathbb{R}^m,\ u \in H$,
2. f is Lipschitz for some constant $a : \forall\ x, y \in \mathbb{R}^m,\ \| |f(x)\rangle - |f(y)\rangle \| \leq a \cdot d_{\mathbb{R}^m}(x, y)$,
3. There are $r, \varepsilon > 0$ such that $\forall\ x, y \in \mathbb{R}^m$, if $d_{\mathbb{R}^m/H}(x, y) \geq r$, then $|\langle f(x)|f(y)\rangle| \leq \varepsilon$,

where $d_{\mathbb{R}^m}(x, y) = \|x - y\|$ and $d_{\mathbb{R}^m/H}(x, y) = \inf_{u \in H} \|x - y - u\|$ for the Euclidean norm $\|x\|$.

Given an input principal ideal $\mathfrak{a}$ of a totally real field K, we show how to construct a function $f_{\mathfrak{a}} : \mathbb{R}^m \to \{\text{quantum states}\}$ which hides a lattices $\Lambda_{\mathfrak{a}} \subseteq \mathbb{R}^m$ whose knowledge reveals a generator of $\mathfrak{a}$ and which satisfies Properties (1), (2) and (3). The main observation allowing

us to reuse the function f (defined in [6]) hiding the units of K is that if $g \in \mathbb{R}^m$ corresponds to a generator (unknown) of $\mathfrak{a}$, then $f_{\mathfrak{a}} : \mathbb{R}^m \times \mathbb{Z}$ defined by $f_{\mathfrak{a}}(x, i) = f(x - ig)$ hides a lattice corresponding to the i-th powers of generators of $\mathfrak{a}$ and can be extended to a function on $\mathbb{R}^{m+1}$ enjoying Properties (1), (2), and (3).

Proposition 2. *There is a function $f_{\mathfrak{a}}$ defined on $\mathbb{R}^m$, where m is polynomial in $n := \deg(K)$, that hides the lattice $\Lambda_{\mathfrak{a}}$ and satisfies conditions (1), (2) and (3) for the parameters $\overline{a}, \overline{r}, \overline{\varepsilon}$ defined by*

$$\begin{aligned}
\overline{a}^2 &= 6(r_1 + r_2 - 1)\log^2 |\mathcal{N}(\mathfrak{a})| \left(\frac{\sqrt{\pi n}s}{4\nu} + 2\right)^2 + l\left(\frac{\pi}{2\nu\lambda}(1+\nu)\right)^2 \\
\overline{r}^2 &= \left(\log\left(1 + (s\sqrt{n})^{n-1} 2\nu\sqrt{n}\right)\right)^2 + l(2\nu\lambda)^2 \\
\overline{\varepsilon} &= 3/4
\end{aligned}$$

4 Computing a short generator of a principal ideal in $\mathbb{Q}(\zeta_{p^n})$

We show how to combine the algorithm for the PIP described in the previous section with known techniques, in particular the recent reduction short-PIP to PIP proved by Cramer, Ducas, Peikert and Regev [5], to perform a key recovery attack. The general idea of first solving the PIP and then using a reduction from short-PIP to PIP probably goes back to the time when cryptosystems relying on the short-PIP were defined. However, in the absence of algorithms for efficiently solving these problems, there had not been any public description of it until recently. To the best of our knowledge, the first time such an approach was publicly suggested was by Bernstein [1]. The attack of Campbel et al. [4] also relies on the same idea. The outline of the algorithm for solving the short-PIP we present here, which is based on the same general strategy, is the following:

1. Compute the ideal $\mathfrak{b}$ of $K^+ = \mathbb{Q}(\zeta_{p^n} + \zeta_{p^n}^{-1})$ (totally real field) generated by $\mathcal{N}_{K/K^+}(\mathfrak{a})$.
2. Find a generator g of $\mathfrak{b}$.
3. Solve the norm equation $\mathcal{N}_{K/K^+}(x) = g$ with the generalization of the Howgrave-Graham-Szydlo algorithm [11] of Garg, Gentry and Halevi [8, 7.3].
4. Find a short generator α of $\mathfrak{a}$ from x with the techniques described by Cramer et al. [5].
5. Return either α or $\bar{\alpha}$ (depending on which one generates $\mathfrak{a}$).

Proposition 3. *There is an efficient quantum algorithm that recovers the short generator of an input ideal $\mathfrak{a}$ in a cyclotomic field of the form $\mathbb{Q}(\zeta_{p^n})$.*

5 Conclusion and significance

We provided the first polynomial time algorithm to compute the generator of a principal ideal in a totally real number field of arbitrary degree. We showed that it derives from the results of [6] in a rather straightforward way, and despite the fact that it only applies to totally real fields, it is a very significant result for post-quantum cryptography. Indeed. together with the reduction from the short-PIP to the PIP, originally observed by Campbel, Groves and Shepherd [4] and later proved by Cramer, Ducas, Peikert and Regev [5], it is enough to attack cryptosystems based on the hardness of finding a short generator of a principal ideal in a cyclotomic field of prime power conductor in quantum polynomial time. These include the multilinear maps of Garg, Gentry and Halevi [8] and the fully homomorphic encryption scheme of Smart and Vercauteren [13].

References

1. D. Bernstein. A subfield-logarithm attack against ideal lattices. `http://blog.cr.yp.to/20140213-ideal.html`. Feb. 2014.
2. J.-F. Biasse. Subexponential time relations in large degree number fields. Submitted to *Advances in Mathematics of Communications*.
3. J.-F. Biasse and C. Fieker. Subexponential class group and unit group computation in large degree number fields. *LMS Journal of Computation and Mathematics*, 17:385–403, 1 2014.
4. P. Campbel, M. Groves, and D. Shepherd. SOLILOQUY, a cautionary tale. `http://docbox.etsi.org/Workshop/2014/201410_CRYPTO/S07_Systems_and_Attacks/S07_Groves_Annex.pdf`, 2014.
5. R. Cramer, L. Ducas, C. Peikert, and O. Regev. Recovering short generators of principal ideals in cyclotomic rings. *IACR Cryptology ePrint Archive*, 2015:313, 2015.
6. K. Eisenträger, S. Halgren, A. Kitaev, and F. Song. A quantum algorithm for computing the unit group of an arbitrary degree number field. In *Proceedings of the 46th Annual ACM Symposium on Theory of Computing*, STOC '14, pages 293–302, New York, NY, USA, 2014. ACM.
7. C. Fieker. Algorithmic Number Theory. Lecture notes available at http://www.mathematik.uni-kl.de/agag/mitglieder/professoren/prof-dr-claus-fieker, 2014.
8. S. Garg, C. Gentry, and S. Halevi. Candidate multilinear maps from ideal lattices. In T. Johansson and P. Nguyen, editors, *Advances in Cryptology - EUROCRYPT 2013, 32nd Annual International Conference on the Theory and Applications of Cryptographic Techniques, Athens, Greece, May 26-30, 2013. Proceedings*, pages 1–17, 2013.
9. C. Gentry and M. Szydlo. Cryptanalysis of the Revised NTRU Signature Scheme. In Lars Knudsen, editor, *Advances in Cryptology EUROCRYPT 2002*, volume 2332 of *Lecture Notes in Computer Science*, pages 299–320. Springer Berlin Heidelberg, 2002.
10. S. Hallgren. Fast quantum algorithms for computing the unit group and class group of a number field. In H. Gabow and R. Fagin, editors, *Proceedings of the 37th Annual ACM Symposium on Theory of Computing, Baltimore, MD, USA, May 22-24, 2005*, pages 468–474. ACM, 2005.
11. N. Howgrave-Graham and M. Szydlo. A method to solve cyclotomic norm equations $f * \bar{f}$. In D. Buell, editor, *Algorithmic Number Theory*, volume 3076 of *Lecture Notes in Computer Science*, pages 272–279. Springer Berlin Heidelberg, 2004.
12. R. Pinch. SOLILOQUY, a cautionary tale. talk at the ICERM workshop on the mathematics of lattices and cybersecurity, 2015.
13. N. Smart and F. Vercauteren. Fully homomorphic encryption with relatively small key and ciphertext sizes. In P. Nguyen and D. Pointcheval, editors, *Public Key Cryptography - PKC 2010*, volume 6056 of *Lecture Notes in Computer Science*, pages 420–443. Springer Berlin Heidelberg, 2010.

On the quantum attacks against schemes relying on the hardness of finding a short generator of an ideal in $\mathbb{Q}(\zeta_{p^n})$

Jean-François Biasse and Fang Song

University of Waterloo
Institute for Quantum Computing
200 University Avenue West
Waterloo, Ontario, Canada N2L 3G1

Abstract. Some recent cryptosystems, including the multilinear maps of Garg, Gentry and Halevi [8] and the fully homomorphic encryption scheme of Smart and Vercauteren [17], are based on the hardness of finding a short generator of an principal ideal (short-PIP) in a number field (typically in cyclotomic fields). However, the assumption that short-PIP is hard has been challenged recently by Campbel et al. [4]. They proposed an approach for solving short-PIP that proceeds in two steps: first they sketched a quantum algorithm for finding *an* arbitrary generator (not necessarily short) of the input principal ideal. Then they suggested that it is feasible to compute a *short* generator efficiently from the generator in Step 1. Campbel et al. [4] conjectured that this attack could run in polynomial time, which drew a lot attention. Since then, the conjectured run-time for Step 1 has been retracted [15] while Cramer et al. [5] validated Step 2 of the approach by giving a detailed analysis. Whether the first step could be salvaged remains an open question.
In this paper we investigate the first step of [4] formally. We first observe that their quantum algorithm for finding a generator essentially falls into a framework of quantum algorithms for the hidden subgroup problem described by Hallgren [11]. Hence, it suffers from similar limits, and we can show that, according to the same line of analysis of Hallgren, the algorithm has running time exponential in the degree of the number field. It has been an open question whether one can improve the analysis of Hallgren [11]. Therefore it indicates that it is at least difficult to prove that the quantum algorithm of Campbel et al. [4] is efficient.
On the positive side, we show that if we adapt one component of the algorithm of Campbel et al. and combine it with techniques in a recent work by Eisenträger et al. [6], then we can essentially use the quantum algorithm for computing the unit group described in [6] to compute the a generator of a principal ideal, thus efficiently solving the problem of Step 1.

Keywords: Lattice-based cryptography, quantum attack, number theory

1 Introduction

A series of works describe cryptosystems relying on the hardness of finding a small generator of a principal ideal in the ring of integers of $\mathbb{Q}(\zeta_{2^n})$. In particular, this problem allows to describe fully homomorphic schemes, such as that of Smart and Vercauteren [17], or the multilinear maps of Garg, Gentry and Halevi [8]. Moreover, these schemes have been described as quantum safe in the absence of quantum attacks against them. This potential for quantum safety was the main appeal to scientists from the CESG for the development of SOLILOQUY, a cryptosystem relying on the hardness of finding a short generator of a principal ideal.

Since then, the CESG has interrupted the SOLILOQUY program because there were indications that it was not as quantum safe as they originally thought. Campbel, Groves and Shepherd [4] released an online draft explaining the design of SOLILOQUY and its apparent weaknesses. Most notably, they observed that finding a short generator of an ideal in the ring of integers of $\mathbb{Q}(\zeta_{2^n})$ polynomially reduced to finding an arbitrary generator (which corresponds

to the Principal Ideal Problem). This fact was rigorously proved by Cramer, Ducas, Peikert and Regev [5] shortly thereafter.

The bottleneck of a key-recovery attack against schemes relying on the hardness of finding a short generator of a principal ideal is the resolution of the PIP. A classical subexponential algorithm was described by Biasse and Fieker for this task [2,3]. Meanwhile, the draft of Campbel et al. [4] describes a quantum attack and conjectures that it runs in polynomial time. Since then, this conjecture has been retracted by Pinch [15] in a talk on SOLILOQUY on behalf of Campbel, Groves and Shepherd. The possibility of a quantum polynomial time attack has generated a lot of attention, and was cited in [5], as well as in various blogs and discussion forum.

Contribution In this paper, we show how to derive a quantum polynomial time attack from a recent result of Eisenträger, Hallgren, Kitaev and Song [6] and the reduction from short-PIP to PIP of Cramer, Ducas, Peikert and Regev [5]. We focus on the task of finding a generator of a principal ideal in the ring of integers of $\mathbb{Q}(\zeta_{p^n})$. Our contribution is two fold.

On the negative side, we analyze the quantum algorithm mentioned in the draft of Campbel et al. [4], and we highlight the main obstructions to a polynomial run time and we put this into perspective with respect to the current state of the art in quantum computing.

On the positive side, we rigorously prove that we can derive a quantum polynomial time algorithm for the search of the generator of a principal ideal from the recent work of Eisenträger et al. [6].

2 An (over) simplified presentation of quantum computing

In this section, we try to convey the aspects of quantum computing that are relevant to the quantum algorithm described in [4] as well as to other quantum cryptanalysis algorithms without getting too technical. This is achieved at the price of some simplifications. First of all, quantum computations occur on quantum states, which are vectors of the form

$$|x\rangle = \alpha_0|0\rangle + \alpha_2|1\rangle + \cdots + \alpha_{2^k-1}|2^k-1\rangle,$$

where values involved in this definitions are

- Complex numbers α_i such that $\sum_i |\alpha_i|^2 = 1$.
- Vectors $|i\rangle$ of $(\mathbb{C}^2)^{\otimes k}$ where $|i\rangle$ is the i-th element of an orthonormal basis.

The notation $|x\rangle|y\rangle$ denotes the tensor product of $|x\rangle$ and $|y\rangle$. A quantum algorithm can be viewed as a unitary matrix U in $\mathbb{C}^{2^k\times 2^k}$ acting on a state via $|x\rangle \mapsto U|x\rangle$ (matrix-vector multiplication). A quantum state only gives away information once it is *measured* (according to the chosen basis). This process returns the answer i with probability $|\alpha_i|^2$ and leaves the system in the state $|i\rangle$. Therefore, whatever happens to the original state (usually a trivial one) has to lead to a state whose measurement yields the result of the algorithm with good probability (typically a constant probability). More generally, when a state has the form $|\psi\rangle = \sum_i \phi_i \otimes |\gamma_i\rangle$ where the ϕ_i are orthogonal vectors of $(\mathbb{C}^2)^{\otimes k_1}$ such that $\sum_i \langle\phi_i, \phi_i\rangle = 1$ and the $|\gamma_i\rangle$ are an orthonormal basis of $(\mathbb{C}^2)^{\otimes k_2}$, then measuring the second register yields the answer γ_i with probability $\langle\phi_i, \phi_i\rangle$ and leaves the system in the state $\phi_i \otimes |\gamma_i\rangle$.

3 Shor's factoring algorithm

Post-quantum cryptography really became a concern when Shor proposed a quantum algorithm to factor RSA integers [16]. Moreover (as we see in the next section), this algorithm extends to the discrete logarithm problem in any group. The problem of factoring an RSA number reduces to an instance of the so-called *Hidden Subgroup Problem* (HSP).

Definition 1 (Hidden Subgroup Problem over $\mathbb{Z}$). *Given $f : \mathbb{Z} \to X$ for a finite set X such that there exists a subgroup $H \leq \mathbb{Z}$ with*

$$f(x+g) = f(x)\ \forall x \in \mathbb{Z} \text{ if and only if } \ g \in H,$$

the Hidden Subgroup Problem is the task of finding H given oracle access to f. This means finding r such that $H = r\mathbb{Z}$.

We want to factor an RSA number $N = pq$. Let a coprime with N (if $a \mid N$, the factorization problem is solved) and

$$\begin{aligned} \mathbb{Z} &\xrightarrow{\ f\ } \mathbb{Z}/N\mathbb{Z} \\ x &\longrightarrow a^x \bmod N \end{aligned}$$

A solution to the HSP with f yields r the order of a mod N and if a is a square we get

$$(a^{r/2} - 1)(a^{r/2} + 1) = 0 \bmod N.$$

This gives us a divisor of N with probability $1/4$.

The first step of this method relies on the fact that if f is efficiently computable classically, one can create an efficient quantum algorithm to evaluate f in superposition. This yields a circuit for

$$\frac{1}{\sqrt{M}} \sum_{x \in \mathbb{Z}_M} |0\rangle|x\rangle \xrightarrow{f} \frac{1}{\sqrt{M}} \sum_{x \in \mathbb{Z}_M} |f(x)\rangle|x\rangle.$$

The other main ingredient we need to use in Shor's algorithm is the so-called Quantum Fourier Transform (QFT) over $\mathbb{Z}_M$ (for a large enough M). Let $\omega_M = e^{2\pi i/M}$, the QFT is the quantum algorithm realizing

$$\mathrm{QFT}_M : |x\rangle \longmapsto \frac{1}{\sqrt{M}} \sum_{y \in \mathbb{Z}_M} \omega_M^{x \cdot y} |y\rangle.$$

If we apply the QFT to the second register of the previous state, we obtain

$$\begin{aligned} \frac{1}{\sqrt{M}} \sum_{x \in \mathbb{Z}_M} |f(x)\rangle|x\rangle \xrightarrow{\mathrm{QFT}_N} & \frac{1}{\sqrt{M}} \sum_{x \in \mathbb{Z}_M} |f(x)\rangle \left(\frac{1}{\sqrt{M}} \sum_{y \in \mathbb{Z}_M} \omega_M^{x \cdot y} |y\rangle \right) \\ &= \frac{1}{M} \sum_{y \in \mathbb{Z}_M} \left(\sum_{x \in \mathbb{Z}_M} \omega_M^{x \cdot y} |f(x)\rangle \right) |y\rangle \\ &:= \frac{1}{M} \sum_{y \in \mathbb{Z}_M} \phi_y \otimes |y\rangle. \end{aligned}$$

We can easily verify that the ϕ_y are orthogonal vectors satisfying $\sum_y \langle \phi_y | \phi_y \rangle = 1$. We perform a measurement on the second register which yields the value y with probability $\frac{1}{M^2}\langle \phi_y, \phi_y \rangle \approx \frac{1}{M}\sum_{k \leq M/r} \left(\omega_M^{y \cdot r}\right)^k$.

$$\begin{aligned}
\Pr\left[\text{measure } y\right] &= \frac{1}{M^2}\left(\sum_{x_1 \in \mathbb{Z}_M} \langle f(x_1) | \omega_M^{-x_1 \cdot y}\right)\left(\sum_{x_2 \in \mathbb{Z}_M} \omega_M^{x_2 \cdot y} | f(x_2)\rangle\right) \\
&= \frac{1}{M^2}\sum_{x_1, x_2 \in \mathbb{Z}_M} \omega_M^{y(x_2 - x_1)} \langle f(x_1) | f(x_2) \rangle \\
&= \frac{1}{M^2}\sum_{x_1, x_2 \in \mathbb{Z}_M, f(x_1) = f(x_2)} \omega_M^{y(x_2 - x_1)} \\
&\approx \frac{1}{M}\sum_{k \leq M/r} \left(\omega_M^{y \cdot r}\right)^k
\end{aligned}$$

Then if y/M is close to an element of the form l/r, then the above probability will high (a constant) if it is not, it will, the probability of measuring y will be low. From the good rational approximation y/M of an element of the form l/r, one can recover the period r and thus solve the problem. This is not the only variant of Shor's algorithm for factoring algorithms. Often times, a partial measurement is performed on the $f(x)$ register before applying the Quantum Fourier Transform. This renormalizes nicely the resulting state, thus facilitating the analysis. We presented this way because we wanted to use an approach similar to that of the work of Campbel et al. [4] to emphasize the similarities between [4] and the original algorithm due to Shor.

4 The Hidden Subgroup Problem in higher dimension

The hidden subgroup problem has a straightforward generalization in higher dimension. Many problems in algebraic number theory can be reduced to an instance of the HSP.

Definition 2 (Hidden Subgroup Problem over $\mathbb{Z}^n$). *Given $f : \mathbb{Z}^n \to X$ for a set X such that there exists a subgroup $H \leq \mathbb{Z}^n$ with*

$$f(x + g) = f(x) \text{ if and only if } g \in H,$$

the Hidden Subgroup Problem is the task of finding H given oracle access to f.

The discrete logarithm problem is the search for $h \in \mathbb{Z}$ such that $b = a^h$ where a, b are given elements of a group $\mathcal{G}$. This can be reduced to an instance of the Hidden Subgroup Problem in $\mathbb{Z}^2$. We define the function

$$\begin{aligned}
\mathbb{Z} \times \mathbb{Z} &\xrightarrow{f} \quad \mathcal{G} \\
(x, y) &\longrightarrow a^x b^{-y}
\end{aligned}$$

The periods of this function are the subgroup $G = (1, h)\mathbb{Z}^2$, and finding the subgroup G hidden by f solves our problem. The analysis we carried on to solve the HSP in $\mathbb{Z}$ generalizes in higher

dimension by using the tensor product of the QFT

$$\mathrm{QFT}_M^{\otimes k} : |\mathbf{x}\rangle \longmapsto \frac{1}{\sqrt{M^k}} \sum_{\mathbf{y}\in\mathbb{Z}_M^k} \omega_M^{\mathbf{x}\cdot\mathbf{y}} |\mathbf{y}\rangle,$$

where $\mathbf{x}, \mathbf{y} \in \mathbb{Z}_M^k$, and $|\mathbf{x}\rangle$ is an encoding of the vector $\mathbf{x}$. Note that here again, M has to be chosen large enough with respect to the typical values we are calculating. As for factoring, applying the QFT yields a state of the form $\frac{1}{M^k}\sum_{\mathbf{y}\in\mathbb{Z}_M^k} \phi_{\mathbf{y}} \otimes |\mathbf{y}\rangle$ and we measure the vector $\mathbf{y} \in \mathbb{Z}_M^k$ with probability

$$\frac{1}{M^{2k}} \sum_{\mathbf{x}_1,\mathbf{x}_2\in\mathbb{Z}_M^k, f(x_1)=f(x_2)} \omega_M^{\mathbf{y}\cdot(\mathbf{x}_2-\mathbf{x}_1)} = \frac{1}{M^k} \sum_{\mathbf{u}\in\mathcal{L}\cap\mathbb{Z}_M^k} \omega_M^{\mathbf{y}\cdot\mathbf{u}}$$

where $\mathcal{L} \subseteq \mathbb{Z}^k$ is the hidden subgroup (a lattice) we are looking for. This sum is larger when $\mathbf{y}\cdot\mathbf{x}$ is an integer, that is when $\mathbf{y}/M \in \mathcal{L}^\perp$. It can be shown that when $\mathbf{y}/M$ is close enough to a point in the dual of $\mathcal{L}$, then it has a high probability of being sampled. This generalizes the factoring algorithm presented in the previous section which relies on the sampling of elements in the dual of the lattice $\mathcal{L} = r\mathbb{Z}$. After finding a good approximation of the dual lattice $\mathcal{L}^\perp$, we use classical linear algebra methods to compute $\mathcal{L}$.

To solve other number theoretic problems, we need to work with approximations of real numbers. This occurs for example in Hallgren's method [12] to solve the Pell equation in quantum polynomial time. The discretization method used by Hallgren was generalized by Hales [10] to derive a solution to the Hidden Subgroup Problem over (approximations of) the reals. To compute the ideal class group, the unit group and to solve instances of the Principal Ideal Problem in number fields of higher degree, the usual approach is to first reduce the problem to the task of finding the periods of a functions f defined over $\mathbb{R}^k$, and then find these periods with an algorithm for solving the HSP. For example, Hallgren [11] described a unit group algorithm in a field K consisting of finding the periods of the function

$$\begin{array}{ccc} \mathbb{R}^k & \xrightarrow{\ f\ } & \mathcal{I}\times\mathbb{R}^k \\ x & \longrightarrow & \left(\frac{1}{\mu}\mathcal{O}, x - \mathrm{Log}(\mu)\right) \end{array} \quad \text{where } \mu\in\mathcal{O} \text{ minimizes } \left|\mathrm{Log}(\mu) - x\right|.$$

Here $\mathrm{Log}(\mu) = (\log|\sigma_1(\mu)|, \cdots, \log|\sigma_r(\mu)|)$ is the vector of the logarithms of the Archimedean embeddings of μ. Since this function relies on the search for a minimum in $\mathcal{O}$, its evaluation costs exponential time in the degree, thus restricting its use for classes of number field with fixed degree. In the same paper, Hallgren [11] described quantum polynomial time algorithms for the unit group, the class group and the Principal Ideal Problem in classes of fixed degree number fields.

A necessary condition to ensure that these problems can be solved in polynomial time is that they reduce to the search for the periods of a function that is efficiently computable. The evaluation of the function described above is not polynomial in the degree of the extension, which is one reason why the overall algorithm does not run in polynomial time in k. The other obstruction lies within the resolution of the subsequent instance of the HSP. Indeed, the method used in [11] to solve the Hidden Subgroup Problem in $\mathbb{R}^k$ does not seem to run in polynomial time with respect to k. It relies on the creation and the measurement of the state

$$|\psi\rangle = \frac{1}{\sqrt{|\mathcal{L}_q|}}\frac{1}{\sqrt{M}} \sum_{\mathbf{x}\in\mathbb{Z}_M^k} \sum_{\mathbf{u}\in\mathcal{L}_q} \omega_M^{\mathbf{x}\cdot\lceil N\mathbf{u}\rfloor} |\mathbf{x}\rangle, \text{ where } \mathcal{L}_q = \mathcal{L}\cap[0,q]^k.$$

Hallgren showed that the probability of measuring $\mathbf{x}$ such that $\mathbf{x}/q$ was $1/q$-close to $\mathcal{L}^\perp$ was at least $\frac{1}{8n^k}$ where $n = \log(\mathrm{disc}(\mathcal{O})$ (a term corresponding to the zero-filling was omitted). In classes of fixed degree (i.e. when k is fixed), this gives a polynomial time algorithm to solve the HSP. However, deciding if this method could be adapted to have a polynomial complexity in k has been an open problem for over 10 years.

5 The quantum algorithm of Campbel et al. [4]

Campbel, Groves and Shepherd [4] attempted to describe a quantum polynomial time algorithm for solving the Principal Ideal Problem (PIP) in totally real cyclotomic fields by using essentially the same technical tools than those available to Hallgren in his 2005 paper [11] (which gives a polynomial time solution to the PIP in classes of fixed degree number fields). Combined with the Gentry-Szydlo (classical) attack [9], this solves the PIP in any cyclotomic field. They sketched an attack in [4, Sec. 5], but it was never analyzed.

In this section, we give a high level description of the quantum attack of [4] and show how it suffers from the same fundamental obstruction as Hallgren's 2005 algorithm for solving the Hidden Subgroup Problem [11] which is only known to run in polynomial time in fixed dimension. The quantum PIP algorithm of Campbel et al. follows the usual two-step strategy consisting of first reducing the PIP to the task of finding the periods of a function (which is similar to the Hidden Subgroup Problem, except that the function they use does not fall into the formal definition for the HSP). This means exhibiting a function $f : G \subseteq \mathbb{R}^m \to \{\text{Lattices over } \mathbb{R}^n\}$ for some subgroup G and $m, n \in \mathbb{Z}_{>0}$ such that $f(x) = f(y)$ if and only if $x = y \bmod \Lambda$ for a lattice $\Lambda \subseteq \mathbb{R}^m$ whose knowledge answers the original problem (the PIP in this case). Then the second step consists of finding the periods of f.

Reduction to the search for the periods of a function Let $\alpha' \in K$ be a generator (not necessarily small) of the input fractional ideal $\mathfrak{a}$ of K and let $u_1, \cdots, u_r$ be a system of fundamental units of the ring of integers $\mathcal{R}$ of K. Then every generator of the principal ideal $\mathfrak{a}$ is of the form $\alpha' \cdot u_1^{x_1} \cdots u_r^{x_r}$. This means that $-k\,\mathrm{Log}(\alpha') + \sum_i x_i \,\mathrm{Log}(u_i) = \mathrm{Log}(\beta)$ for some $\beta \in K$ satisfying $\beta \cdot \mathcal{O} = \mathfrak{a}^{-k}$, which is equivalent to $\beta \cdot \mathcal{O}\mathfrak{a}^k = \mathcal{O}$ (where $\mathrm{Log}(x) := (\log|\sigma_1(x)|, \cdots, \log|\sigma_r(x)|)$). Let $v = (v_1, \cdots, v_r) \in \mathbb{R}^r$, we define $e^v := (e^{v_1}, \cdots, e^{v_r})$, and if K is a totally real field (the case considered in [4]), then for $v = \mathrm{Log}(x)$ wth $x \in K$, we have $e^v = (|\sigma_1(x)|, \cdots, |\sigma_r(x)|)$. For $k \in \mathbb{Z}$ and $v \in \mathbb{R}^r$ (not necessarily corresponding to the valuations of an element in K), let us denote by $e^v \cdot \mathfrak{a}^k$ the lattice generated by the elements of the form $e^v \cdot a$ for $a \in \mathfrak{a}$ (where elements of $\mathfrak{a}$ are represented by their vector of real embedding, and multiplication is component-wise). In the special case where $v = \mathrm{Log}(x)$ for $x \in K$, we have

$$e^v \cdot \mathfrak{a}^k = \pm x\mathcal{O} \cdot \mathfrak{a}^k = x\mathcal{O} \cdot \mathfrak{a}^k.$$

An element of the form $(-k, v)$ satisfies $e^v \cdot \mathfrak{a}^k = \mathcal{O}$ if and only if $v = k\,\mathrm{Log}(\alpha') + \sum_i x_i \,\mathrm{Log}(u_i)$ for some $x_i \in \mathbb{Z}$. This means that (k, v) is in the hidden subgroup of $\mathbb{Z} \times \mathbb{R}^r$ defined by

$$\Lambda_{\alpha'} := \mathbb{Z}(-1, \mathrm{Log}(\alpha')) + \mathbb{Z}(0, \mathrm{Log}(u_1)) + \cdots + \mathbb{Z}(0, \mathrm{Log}(u_r)).$$

As each element (k, v) of $\Lambda_{\alpha'}$ satisfies $\sum_i v_i = -k \log(\mathcal{N}(\mathfrak{a}))$, the search of the corresponding hidden subgroup can be restricted to the control space

$$G = \left\{ (k, v) \in \mathbb{Z} \times \mathbb{R}^r \text{ such that } \sum_i v_i = -k \log(\mathcal{N}(\mathfrak{a})) \right\}.$$

The function $F : G \to \{\text{lattices over } \mathbb{R}^n\}$ defined by $F(k, v) := e^v \mathfrak{a}^k$ can be then composed by a quantum encoding to uniquely identify the lattice $e^v \mathfrak{a}^k$. This encoding of lattices is called the "quantum fingerprint" and it gives the map

$$f : (k, v) \in G \xrightarrow{F} F(k, v) \xrightarrow{\text{fingerprint}} |\psi_{k,v}\rangle.$$

Campbel et al. conjectured that the quantum encodings of almost identical lattices have inner product close to 1 while the quantum encodings of essentially different lattices have inner product close to 0. Although this property was not proved, it seems likely to hold true. The function f "hides" $\Lambda_{\alpha'}$ in the sense that

$$f(k_1, v_1) = f(k_2, v_2) \iff u := (k_1, v_1) - (k_2, v_2) \in \Lambda_{\alpha'}.$$

Identifying $\Lambda_{\alpha'}$ from the periods of this map is an analogue of the so-called Hidden Subgroup Problem (HSP). This reduction between the search for a generator of an ideal and the computation of the periods of a function is different from what was done by Hallgren in [11], and it is a suitable one even in the case of large degree number fields. Indeed, unlike Hallgren in [11], Campbel et al. use a function F whose evaluation has polynomial complexity in k.

Computing the periods of f However, the method proposed by Campbel, Groves and Shepherd for computing the periods of f does not seem to overcome the obstruction faced by the HSP resolution method of [11]. They propose to to find the periods of f in the same way as the HSP is solved in Hallgren's 2005 paper [11], thus encountering the same limitations.

1. Discretize and bound G, and then create the state $|\psi\rangle := \frac{1}{\sqrt{M}} \sum_{(k,v)\in G'} |\psi_{k,v}\rangle|(k, v)\rangle$.
2. Apply the Quantum Fourier Transform over G to $|\psi\rangle$.
3. Measure (k, v) and check if we obtain a good approximation of an element in $\Lambda_{\alpha'}^{\perp}$.
4. Repeat Step 2 and 3 until a basis of good approximations of $\Lambda_{\alpha'}^{\perp}$ is found.
5. Find an approximation of a basis of $\Lambda_{\alpha'}$ from $\Lambda_{\alpha'}^{\perp}$ with classical methods.

In Step 1, M is the normalization factor depending on the radius and the precision of the bounded discretized version G' of G.

Proposition 1 (Sampling probability). *Using the same techniques as in [11, Sec 3.2], the probability of drawing a rational approximation that is* $1/q$*-close to a vector in* $\Lambda_{\alpha'}^{\perp}$ *for* $q \geq (r + 1)^2\lambda$ *where* λ *is a bound on the size of the vectors in a reduced basis of* $\Lambda_{\alpha'}$ *is at least*

$$P \geq \frac{1}{8\,(\log |\Delta| t)^{r+1}},$$

where $t \geq 8(r + 1)$.

The above statement gives a lower bound on the probability of drawing points that are approximations of elements in $\Lambda_{\alpha'}^{\perp}$. This in turns give an upper bound on the run time to obtain enough approximations of lattice points before being able to find a basis of $\Lambda_{\alpha'}$. Still assuming that the same techniques are used, we can also derive an upper bound on the probability of sampling an approximation of a dual lattice points, which in turns gives a lower bound on the run time of the algorithm.

Proposition 2 (exponential run time). *Still assuming the techniques of [11, Sec 3.2] with the parameters described in the previous proposition, the run time of the overall algorithm is at least* 2^r.

Proof (of proposition 1). To bound and discretize G, we need three parameters that were not explicitly given in [4]. The grid has precision $1/N$ for some $N > 0$, and we choose to restrict the QFT to $G \cap [0, q]^{r+1}$ for a large enough integer q. We also enlarge the grid by a factor t that will be used to analyze the complexity (this is the so-called zero-filling technique). As before, we denote the dimension by $k = r + 1$ and the normalization factor $M = qtN$. We can identified the discretized and bounded G' with $\mathbb{Z}_M^k$. Then the algorithm is the same as for factoring,

$$\frac{1}{\sqrt{M^k}} \sum_{\mathbf{x} \in \mathbb{Z}_M^k} |0\rangle|x\rangle \xrightarrow{F} \frac{1}{\sqrt{M^k}} \sum_{\mathbf{x} \in \mathbb{Z}_M} |\psi_{\mathbf{x}}\rangle|\mathbf{x}\rangle \xrightarrow{\mathrm{QFT}_M^{\otimes k}} \frac{1}{M^k} \sum_{\mathbf{y} \in \mathbb{Z}_M^k} \phi_{\mathbf{y}} \otimes |\mathbf{y}\rangle.$$

We measure $\mathbf{y}$ and hope that it is close enough to a vector in $\Lambda_{\alpha'}$. To analyze this technique, we use the same approach as Hallgren's 2005 paper [11]. As for Shor's factoring algorithm, the probability of drawing $\mathbf{y} \in G'$ (regardless of its properties) is

$$\frac{1}{M^{2k}} \langle \phi_{\mathbf{y}}, \phi_{\mathbf{y}} \rangle = \frac{1}{M^{2k}} \sum_{\mathbf{x}_1, \mathbf{x}_2 \in \mathbb{Z}_M^k} \omega_M^{\mathbf{y} \cdot (\mathbf{x}_2 - \mathbf{x}_1)} \langle \psi_{\mathbf{x}_1} | \psi_{\mathbf{x}_2} \rangle$$

Unlike in the exact case where $\langle \psi_{\mathbf{x}_1} | \psi_{\mathbf{x}_2} \rangle$ is either 1 when $\mathbf{x}_2 - \mathbf{x}_1 \in \mathcal{L}$ and 0 otherwise (here $\mathcal{L} = \Lambda_{\alpha'}$), we are dealing with approximations. We assume that the fingerprint behaves as conjectured in [4, Sec. 3.6]. We formalize this by $\langle \psi_{\mathbf{x}_1} | \psi_{\mathbf{x}_2} \rangle = 1$ if $\mathbf{x}_2 - \mathbf{x}_1$ is ε-close to $\mathcal{L}$ for some $\varepsilon < 1/N$ and $\langle \psi_{\mathbf{x}_1} | \psi_{\mathbf{x}_2} \rangle = 0$ otherwise. For each lattice vector $\mathbf{u} \in \mathcal{L}$, we have $\langle \psi_{\mathbf{x}_1} | \psi_{\mathbf{x}_2} \rangle = 1$ for all the $\mathbf{x}_1, \mathbf{x}_2$ such that $\mathbf{x}_2 - \mathbf{x}_1$ is in a ball of radius ε centered around $\mathbf{u}$. So the probability of measuring $\mathbf{y}$ is

$$\frac{1}{M^{2k}} \sum_{\mathbf{x}_1, \mathbf{x}_2 \in \mathbb{Z}_M^k} \omega_M^{\mathbf{y} \cdot (\mathbf{x}_2 - \mathbf{x}_1)} \langle \psi_{\mathbf{x}_1} | \psi_{\mathbf{x}_2} \rangle = \frac{1}{M^{2k}} \sum_{\mathbf{x}_1 \in \mathbb{Z}_M^k} \sum_{\substack{\mathbf{x}_2 \in \mathbb{Z}_M^k \\ \mathbf{x}_1 - \mathbf{x}_2 \in \mathcal{L} + (0,\varepsilon)^k}} \omega_M^{\mathbf{y} \cdot (\mathbf{x}_2 - \mathbf{x}_1)}$$

To bound this probability from below, we show that the phases corresponding to an element $\mathbf{y}$ close to a dual lattice vector are small. Each term $\mathbf{x}_2 - \mathbf{x}_1$ is of the form $Nv + \varepsilon_v$ where $v \in \mathcal{L}$ and $|\varepsilon_v| < 1$. The $\mathbf{y}$ that we hope to measure are of the form $\lfloor tqw \rceil$ for $w \in \mathcal{L}^\perp$. Moreover, to make sure that the phase terms remain bounded, we restrict ourselves to vectors with entries satisfying $|y_i| \leq \frac{qNt}{\log |\Delta|}$. This means that we are measuring approximations of $w \in \mathcal{L}^\perp$ with $|w_i| \leq \frac{qNt}{\log |\Delta|} + 1$, and that N has to be chosen large enough so that we measure a significant portion of $\mathcal{L}^\perp$. So $\mathbf{y} = qtw + \delta_w$ for $\|\delta_w\| < 1/2$ and

$$\begin{aligned} \mathbf{y} \cdot (\mathbf{x}_2 - \mathbf{x}_1) &= (qtw + \delta_w) \cdot (Nv + \varepsilon_v) \\ &= qNt(w \cdot v) + qt(w \cdot \varepsilon_v) + \delta_w \cdot (Nv + \varepsilon_v). \end{aligned}$$

The first term of the sum vanishes from the phase because it equals zero modulo qtN. Indeed $v \cdot w \in \mathbb{Z}$. The second term satisfies

$$\left| \frac{qt(w \cdot \varepsilon_v)}{qtN} \right| \leq \frac{k \max_i |w_i|}{N} \leq \frac{k}{\log |\Delta|} \approx \frac{1}{\log(n)}.$$

Finally, the third term of the phase satisfies

$$\left|\frac{\delta_w \cdot (Nv + \varepsilon_v)}{qtN}\right| \leq \frac{|\delta_w \cdot v|}{qt} + \frac{|\delta_w \cdot \varepsilon_v|}{qtN} \leq \frac{k \max |v_i|}{qt} \leq \frac{1}{8}$$

if we choose $t \geq 8k$. So for large enough n, we have $\left|\frac{\mathbf{y}\cdot(\mathbf{x}_2-\mathbf{x}_1)}{qtN}\right| < \frac{1}{6}$, and the probability $P_{\mathbf{z}}$ of measuring $\mathbf{z}$ satisfies

$$\begin{aligned} P_{\mathbf{z}} &= \frac{1}{M^{2k}} \sum_{\mathbf{x}_1 \in \mathbb{Z}_M^k} \sum_{\substack{\mathbf{x}_2 \in \mathbb{Z}_M^k \\ \mathbf{x}_1 - \mathbf{x}_2 \in \mathcal{L} + (0,\varepsilon)^k}} \omega_M^{\mathbf{y}\cdot(\mathbf{x}_2-\mathbf{x}_1)} = \frac{1}{M^k} \sum_{\mathbf{u} \in \mathcal{L} \cap [0,q]^k} \omega_M^{\mathbf{y}\cdot[N\mathbf{u}]} \\ &\geq \frac{1}{2M^k} \sum_{\mathbf{u} \in \mathcal{L} \cap [0,q]^k} \left(e^{2i\pi/3} + e^{-2i\pi/3}\right) = \frac{|\mathcal{L} \cap [0,q]^k|}{2M^k}. \end{aligned}$$

The above probability holds for all $\mathbf{z} \in \mathcal{L}^\perp$ with entries bounded by $N/\log|\Delta|$. As in [11], we need to relate the number of points in $\mathcal{L}_q = \mathcal{L} \cap [0,q]^k$ to the number of points of $\mathcal{L}^\perp_{N/\log|\Delta|} = \mathcal{L}^\perp \cap \left[0, \frac{N}{\log|\Delta|}\right]^k$. Let λ be a bound on the length of the vectors in a reduced basis of $\mathcal{L}$, by [14, Prop. 8.7] we have $|\mathcal{L}_q| \geq \frac{q^k}{2\det(\mathcal{L}}$ if $q \geq k^2\lambda$ and $|\mathcal{L}^\perp_{N/\log|\Delta|}| \geq \frac{(N/\log|\Delta|)^k}{2\det(\mathcal{L}^\perp)}$ if $N \geq \log|\Delta|^k k^2 \lambda$. Therefore

$$|\mathcal{L}_q||\mathcal{L}^\perp_{N/\log|\Delta|}| \geq \frac{q^k (N/\log|\Delta|)^k}{4\det(\mathcal{L})\det(\mathcal{L}^\perp)} = \frac{q^k (N/\log|\Delta|)^k}{4},$$

and the probability of drawing $\mathbf{z}$ such that $qt\mathbf{z}$ is $1/q$-close to $w \in \mathcal{L}^\perp_{N/\log|\Delta|}$ satisfies

$$P_{\mathbf{z}} \geq \frac{|\mathcal{L}_q|}{2M^k} \geq \frac{1}{8(\log|\Delta|t)^k} \frac{1}{\left|\mathcal{L}^\perp_{N/\log|\Delta|}\right|}.$$

As pointed out in [11], such $\mathbf{z}$ are the points of our grid such that $\frac{\mathbf{y}}{qt}$ is $1/q$-close to a $w \in \mathcal{L}^\perp_{N/\log|\Delta|}$. As there are $\left|\mathcal{L}^\perp_{N/\log|\Delta|}\right|$ vectors $\mathbf{y}$ associated to such a w, the probability of measuring one is at least $\frac{1}{8(\log|\Delta|t)^k}$.

Proof (of Proposition 2). With the same choice of parameters as in the proof of the previous proposition, the probability of drawing $\mathbf{z}$ satisfies

$$P_{\mathbf{z}} = \frac{1}{M^k} \sum_{\mathbf{u} \in \mathcal{L} \cap [0,q]^k} \omega_M^{\mathbf{y}\cdot[N\mathbf{u}]} \leq \frac{|\mathcal{L} \cap [0,q]^k|}{M^k} \approx \frac{q^k}{M^k \det(\mathcal{L})}.$$

There are $\left|\mathcal{L}^\perp_{N/\log|\Delta|}\right| \approx \frac{(N/\log|\Delta|)^k}{\det(\mathcal{L}^\perp}$ such points, which means that the probability of drawing a rational approximation that is $1/q$-close to a point in $\mathcal{L}^\perp_{N/\log|\Delta|}$ is no more than

$$P \leq \frac{(N/\log|\Delta|)^k}{\det(\mathcal{L}^\perp)} \frac{q^k}{M^k \det(\mathcal{L})} = \frac{1}{(\log|\Delta|t)^k} \leq \frac{1}{2^k}.$$

The total run time is at least as much as the time taken to draw a single approximation of a dual lattice point, which is at least $2^k = 2^{r+1}$.

Remark This means that according to the analysis of Hallgren's HSP algorithm for $\mathbb{R}^k$, then only upper bound on the run time that we can derive is exponential in the degree of the number field. Moreover, we can also show that the run time to derive a rational approximation that is $1/q$-close to a vector in $\Lambda_\alpha^\perp$ for $q \geq (r+1)^2\lambda$ is in fact at least exponential in the degree. This shows that according to the state of the art on the resolution of the HSP, the quantum algorithm of Campbel et al. [4] does not run in polynomial time. However, it does not formally prove that there is no other way to choose the parameters and analyze its behavior differently. Indeed, we followed the method of [11, Sec 3.2] that forces us to consider the probability of drawing elements ε-close to the dual lattice for a very small $\varepsilon = 1/q$. It is unclear to us how to analyze a variant with a relaxed condition on ε, but it is certain that a different analysis than Hallgren's would have to be used.

6 An algorithm for the PIP in totally real fields

As mentioned before, it does not seem that the HSP algorithm proposed by Hallgren [11] (and used in the draft of Campbel et al. [4]) allows us to solve the Hidden Subgroup Problem in $\mathbb{R}^m$ in polynomial time in m. However, recent work from Eisenträger, Hallgren, Kitaev and Song [6] developed a new framework for HSP in $\mathbb{R}^m$, which admits efficient an quantum algorithm even for large values of m. They illustrated this by computing the unit group of a number field of arbitrary degree in polynomial time. The algorithm described in [6] returns generators of a secret subgroup H of $\mathbb{R}^m$ (where m depends on the degree n of the field) hidden in the periods of a function $f : \mathbb{R}^m \to \{\text{quantum states}\}$. Eisenträger et al. showed in [6, Th. 6.1] how to recover generators of a secret subgroup $H \subseteq \mathbb{R}^m$ in polynomial time in m if there is a function f satisfying the following properties:

1. f is periodic on H, that is $f(x+u) = f(x)\ \forall x \in \mathbb{R}^m,\ u \in H$,
2. f is Lipschitz for some constant $a : \forall\ x, y \in \mathbb{R}^m,\ \||f(x)\rangle - |f(y)\rangle\| \leq a \cdot d_{\mathbb{R}^m}(x,y)$,
3. There are $r, \varepsilon > 0$ such that $\forall\ x, y \in \mathbb{R}^m$, if $d_{\mathbb{R}^m/H}(x,y) \geq r$, then $|\langle f(x)|f(y)\rangle| \leq \varepsilon$,

where $d_{\mathbb{R}^m}(x,y) = \|x-y\|$ and $d_{\mathbb{R}^m/H}(x,y) = \inf_{u\in H} \|x-y-u\|$ for the Euclidean norm $\|x\|$.

To construct such a function, it is possible to start from a function defined on a subgroup G of $\mathbb{R}^m$. As shown in [6, Sec. 6.1], if a function defined on $G \subseteq \mathbb{R}^m$ hides H and satisfies conditions (2) and (3) on all $x, y \in G$, it can be used to define a function on $\mathbb{R}^m$ hiding (the embedding of) H and satisfying (2) and (3). For simplicity, we use the following notation.

Definition 3 ((a,r,ε)-oracle). *. Let G be a subgroup of $\mathbb{R}^m$ and $f : G \to \{$quantum states$\}$. We say that f is a (a,r,ε)-oracle on G if*

- *f is Lipschitz for some constant $a : \forall\ x, y \in G,\ \||f(x)\rangle - |f(y)\rangle\| \leq a \cdot d_{\mathbb{R}^m}(x,y)$,*
- *There are $r, \varepsilon > 0$ such that $\forall\ x, y \in G$, if $d_{\mathbb{R}^m/H}(x,y) \geq r$, then $|\langle f(x)|f(y)\rangle| \leq \varepsilon$,*

Our goal is to find a (a,r,ε)-oracle on $\mathbb{R}^m$ that hides the right subgroup H of $\mathbb{R}^m$. Then it can be used with the HSP algorithm of [6] to find a generator of a principal ideal in a totally real field.

6.1 Computation of the unit group: Review of [6]

To compute the unit group, Eisenträger et al. used a function of the form $f(x) = |e^x \cdot \mathcal{O}\rangle$ where $e^x \cdot \mathcal{O}$ is the lattice generated by the elements of the form $e^x \cdot \omega_i$ for $\mathcal{O} = \sum_i \mathbb{Z}\omega_i$. Such a function

hides the unit group of the order $\mathcal{O}$ because $f(x+u) = f(x)$ if and only if $e^u \cdot \mathcal{O} = \mathcal{O}$ which means that e^u is a unit in $\mathcal{O}$. It is derived from a function $f_G : G \subseteq \mathbb{R}^m \to \{\text{Quantum States}\}$ where G is a hyperplane containing H. They show that if f_G is a (a, r, ε)-oracle on G that hides H. then it can be extended to $f : \mathbb{R}^m \to \{\text{Quantum States}\}$ satisfying (1), (2) and (3). The first step of the description of a function hiding the unit group is to find a classical function f_c on a certain hyperplane G, then we compose it with a quantum encoding f_q, and finally we extend $f_G : f_q \circ f_c$ to a function f on $\mathbb{R}^m$ that satisfies (1), (2) and (3).

Classical function The function F used by Campbel et al. [4] is very similar to the classical oracle f_c used in [6]. The latter is defined by

$$\begin{array}{ccc} G \subseteq \mathbb{R}^m & \xrightarrow{f_c} & \{\text{lattices in } \mathbb{R}^k\} \\ v & \longrightarrow & e^v \cdot \mathcal{O} \end{array}.$$

Here $G \subseteq \mathbb{R}^{r_1+r_2} \times (\mathbb{Z}/2\mathbb{Z})^{r_1} \times (\mathbb{R}/\mathbb{Z})^{r_2}$ is the hyperplane such that $\sum_{i \leq r_1+r_2} v_i = 0$. In particular, it contains the elements x of the number field K such that $\mathcal{N}(x) = \pm 1$ via the correspondence

$$x \leftrightarrow (\log|\sigma_1(x)|, \cdots, \log|\sigma_{r_1+r_2}(x)|, \text{sign}(\sigma_1(x)), \cdots, \text{sign}(\sigma_{r_1}(x)), \theta_1, \cdots, \theta_{r_2}),$$

where the θ_j are the phases of the complex embeddings $\sigma_j(x)$. Then for

$$v = (v_1, \cdots ., v_{r_1+r_2}, \delta_1, \cdots, \delta_{r_1}, \theta_1, \cdots, \theta_{r_2}),$$

we define the exponentiation

$$e^v = \left((-1)^{\delta_1} e^{v_1}, \cdots, (-1)^{\delta_{r_1}} e^{v_{r_1}}, e^{2i\pi\theta_1} e^{v_{r_1+1}}, \cdots, e^{2i\pi\theta_{r_2}} e^{v_{r_1+r_2}}\right) \in \mathbb{R}^{r_1} \times \mathbb{C}^{r_2}.$$

This can be naturally embedded into $\mathbb{R}^k$ for $k = r_1 + 2r_2$, and in the case of v corresponding to an $x \in K$, we have $e^v = x$. Multiplication in $\mathbb{R}^k$ being considered component-wise, we have $e^v \cdot \mathcal{O} = \mathcal{O}$ if and only if v corresponds to a unit of $\mathcal{O}$. This also implies that $f_c(v_1) = f_c(v_2)$ if and only if $v_1 - v_2 = u$ where e^u is a unit of $\mathcal{O}$.

The quantum encoding The properties that $f_G = f_q \circ f_c$ has to satisfy also depend on the quantum encoding that was chosen, which is one of the important contributions of Eisenträger et al. Let $g_s(\cdot)$ be the Gaussian function $g_s(x) := e^{\pi\|x\|^2/s^2}, x \in \mathbb{R}^k$. For any set $S \subset \mathbb{R}^k$, denote $g_s(S) := \sum_{x \in S} g_s(x)$. Given a a lattice L, the quantum encoding maps L to the lattice Gaussian state via

$$\begin{array}{ccc} \{\text{Lattices over } \mathbb{R}^k\} & \xrightarrow{f_q} & \mathcal{S} \text{ (unit vectors in a Hilbert space)} \\ L & \longrightarrow & |L\rangle := \gamma \sum_{v \in L} g_s(v) |\text{str}_{\nu,k}(v)\rangle \end{array},$$

where γ is a normalization factor. Here $|\text{str}_{\nu,k}(v)\rangle$ is the straddle encoding of a real-valued vector $v \in \mathbb{R}^k$, as defined in [6]. Intuitively, we discretize the space $\mathbb{R}^k$ by a grid $\nu\mathbb{Z}^k$, and we encode the information about v by a superposition over all grid nodes surrounding v. Specifically, for the one-dimensional case, the straddle encoding of a real number is

$$x \in \mathbb{R} \mapsto |\text{str}_\nu(x)\rangle := \cos(\frac{\pi}{2}t)|j\rangle + \sin(\frac{\pi}{2}t)|j+1\rangle,$$

where $j := \lfloor x/\nu \rfloor$ denotes the nearest grid point no bigger than x and $t := x/\nu - j$ denotes the (scaled) offset. Repeat this for each coordinate of $v = (v_1, \ldots, v_n)$ we get $|\mathrm{str}_{\nu,k}(v)\rangle := \bigotimes_{i=1}^{n} |\mathrm{str}_{\nu}(v_i)\rangle$. To analyze our function hiding generators of a principal ideal $\mathfrak{a}$, we rely on the properties of the quantum encoding of the function hiding the unit group of $\mathcal{O}$.

A simplified oracle To show that the HSP algorithm of [6] can be used to compute the unit group, we need to prove that f_G is a (a, r, ε)-oracle for some (a, r, ε) on G. Only a simplified version of this statement appears in [6] where the phases are discarded. A simplified oracle defined over $\mathbb{R}^{r_1+r_2}$ (instead of G) was described and rigorously analyzed. In this paper, we restrict our analysis to the results that can be derived from the following statement.

Proposition 3 (Th. 5.7 of [6]). *Let $f_{\mathbb{R},c}$ be the classical oracle defined by*

$$\begin{array}{ccc} \mathbb{R}^{r_1+r_2} & \xrightarrow{f_{\mathbb{R},c}} & \{\textit{lattices in } \mathbb{R}^k\} \\ v & \longrightarrow & e^v \cdot \mathcal{O} \end{array}.$$

Then $f_{\mathbb{R}} := f_q \circ f_{\mathbb{R},c}$ is an (a, r, ε)-oracle on $\mathbb{R}^{r_1+r_2}$ with

$$a = \frac{\sqrt{\pi n}s}{4\nu} + 1 \;\; \varepsilon = 3/4 \;, \;\; r = \log\left(1 + (s\sqrt{n})^{n-1} 2\nu\sqrt{n}\right).$$

This simplified function hides the free part of the unit group in a totally real field ($r_2 = 0$). It is claimed in [6] that the methods used to prove[6, Th. 5.7] can be generalized to prove that f_G is a (a, r, ε)-oracle for some (a, r, ε) on G that hides the unit group of an arbitrary field. Then by the methods of [4, Sec 6.1], it can be extended to a function $f : \mathbb{R}^m \to \{\text{Quantum States}\}$ that satisfies conditions (1), (2) and (3) for

$$\begin{aligned} \overline{a}^2 &= a^2 + l\left(\frac{\pi}{2\nu\lambda}(1+\nu)\right)^2 \\ \overline{r}^2 &= r^2 + l(2\nu\lambda)^2 \\ \overline{\varepsilon} &= \varepsilon \end{aligned}$$

6.2 Computing a generator of a principal ideal in a totally real field

In this section, we assume that we are given the $\mathbb{Z}$-basis of a principal ideal $\mathfrak{a}$ of an order $\mathcal{O}$ in a totally real field K. We show that there is a polynomial time algorithm to compute $(\log|g|_1, \cdots, \log|g|_n)$ where g is a generator of $\mathfrak{a}$, $n = \deg(K)$ and $|g|_i = |\sigma_i(g)|$ is the i-th Archimedean valuation of g. We reduce this problem to an instance of the HSP and we use the framework of Eisenträger et al. [6]. We start from the same classical oracle as the function F defined by Campbel et at. [4] which we compose with f_q and extend to $\mathbb{R}^n$ (in this case $m = n$). The main observation that allows us to reuse the analysis of the oracle f_c hiding the unit group in [6] is that $F(v, j) = f_c(v - jg)$ where e^g is an arbitrary generator of $\mathfrak{a}$. We denote The classical function we use is defined by

$$\begin{array}{ccc} G \subseteq \mathbb{R}^n \times \mathbb{Z} & \xrightarrow{f_c^{\mathfrak{a}}} & \{\text{lattices in } \mathbb{R}^n\} \\ (v, j) & \longrightarrow & e^v \cdot \mathcal{O} \cdot \mathfrak{a}^{-j} \end{array}.$$

for a certain hyperplane G. The function $f_q \circ f_c^{\mathfrak{a}}$ can be then extended from G to $\mathbb{R}^n$ while preserving the essential continuity properties that allow us to reuse the framework of Eisenträger

et al. for the resolution of the continuous HSP. The careful analysis of the properties of $f_q \circ f_c^{\mathfrak{a}}$, and that of its extension to $\mathbb{R}^n$ lead to Proposition 6 which shows that there is a polynomial time algorithm to find the generator of a principal ideal in a number field.

A function hiding generators of $\mathfrak{a}$ The goal of this paragraph is to prove the following proposition.

Proposition. There is a function $f_{\mathfrak{a}}$ defined on $\mathbb{R}^n$ that hides the lattice $\Lambda_{\mathfrak{a}}$ and satisfies conditions (1), (2) and (3) for the parameters $\overline{a}, \overline{r}, \overline{\varepsilon}$ defined by

$$\begin{aligned}
\overline{a}^2 &= a_1^2 + l\left(\frac{\pi}{2\nu\lambda}(1+\nu)\right)^2 \\
&= 6(r_1+r_2-1)\log^2|\mathcal{N}(\mathfrak{a})|\left(\frac{\sqrt{\pi n}s}{4\nu}+2\right)^2 + l\left(\frac{\pi}{2\nu\lambda}(1+\nu)\right)^2 \\
\overline{r}^2 &= \left(\log\left(1+(s\sqrt{n})^{n-1}2\nu\sqrt{n}\right)\right)^2 + l(2\nu\lambda)^2 \\
\overline{\varepsilon} &= 3/4
\end{aligned}$$

The unit group algorithm of [6] can be generalized to return a generator α' of an ideal $\mathfrak{a}$ in an order $\mathcal{O}$ of a number field K in the case where $\mathfrak{a}$ is known to be principal (as it is the case in cryptography). We focus on the cases where we can use Proposition 3, which restricts the scope of this analysis to totally real number fields. Let $G_{\mathfrak{a}} \subseteq \mathbb{R}^n \times \mathbb{Z}$ be the hyperplane such that $\sum_{i \leq r_1+r_2} v_i = j\log(|\mathcal{N}(\mathfrak{a})|)$ where j is the last coordinate (in $\mathbb{Z}$). This contains all the elements x of K such that $\mathcal{N}(x) = \pm\mathcal{N}(\mathfrak{a})$ via the same embedding $G \hookrightarrow \mathbb{R}^{r_1} \times \mathbb{C}^{r_2}$ as before (with $r_1 = n$ and $r_2 = 0$), and therefore it contains all the vectors of the form $(\log|g|_1, \cdots, \log|g|_n, j)$ where g is a generator of $\mathfrak{a}^j$. Now we define the modified classical oracle (which is essentially the same function as the function F used by Campbel et al. [4])

$$\begin{array}{ccc}
G_{\mathfrak{a}} \subseteq \mathbb{R}^n \times \mathbb{Z} & \xrightarrow{f_c^{\mathfrak{a}}} & \{\text{lattices in } \mathbb{R}^n\} \\
(v,j) & \longrightarrow & e^v \cdot \mathcal{O} \cdot \mathfrak{a}^{-j}
\end{array}.$$

where like in [6], the multiplication of two lattices is the lattice generated by all the products of lattice elements (multiplication still being considered component-wise). We see that $e^v \cdot \mathcal{O} \cdot \mathfrak{a}^{-j} = \mathcal{O}$ if and only if v corresponds to a generator of $\mathfrak{a}^j$, and that $f_c^{\mathfrak{a}}(v_1) = f_c^{\mathfrak{a}}(v_2)$ if and only if $v_1 - v_2 = g$ where e^g is a generator of $\mathfrak{a}^j$. This gives us a classical oracle hiding the lattice $\Lambda_{\mathfrak{a}} \subseteq G_{\mathfrak{a}}$ of elements (v,j) where e^v is a generator of $\mathfrak{a}^j$ from which we can easily derive a generator of $\mathfrak{a}$. But to apply the framework of Eisenträger et al., we need to analyze the continuity of $f_q \circ f_c^{\mathfrak{a}}$.

Proposition 4. *With the $f_c^{\mathfrak{a}}$ and f_q defined above, $s = 2^{2n}\sqrt{n|\Delta|}$ and $\nu = \frac{1}{4n(s\sqrt{n})^{2n}}$, $f_{G_{\mathfrak{a}}} := f_q \circ f_c^{\mathfrak{a}}$ is a (a,r,ε)-oracle on $G_{\mathfrak{a}}$ for*

$$a = \frac{\sqrt{\pi n}s}{4\nu} + 2\ ,\ \varepsilon = 3/4\ ,\ r = \log\left(1+(s\sqrt{n})^{n-1}2\nu\sqrt{n}\right).$$

Proof. Let us fix a generator g of $\mathfrak{a}$ and its corresponding $(v_g, 1) \in G_{\mathfrak{a}} \subseteq \mathbb{R}^n \times \mathbb{Z}$. The main observation leading to the result is that $f_c^{\mathfrak{a}}(v,j) = f_c(v - jv_g)$, and therefore $|f_{G_{\mathfrak{a}}}(v,j)\rangle = |f_G(v - jv_g)\rangle$. In the following, $m := n+1$.

a) Lipschitz condition: if $j_1 \neq j_2$, then $d_{\mathbb{R}^m}((v_1, j_1), (v_2, j_2)) \geq 1$ while at the same time $\| |f_{G_{\mathfrak{a}}}(v_1, j_1)\rangle - |f_{G_{\mathfrak{a}}}(v_2, j_2)\rangle \| \leq 2$. So in this case

$$\| |f_{G_{\mathfrak{a}}}(v_1, j_1)\rangle - |f_{G_{\mathfrak{a}}}(v_2, j_2)\rangle \| \leq 2 d_{\mathbb{R}^m}((v_1, j_1), (v_2, j_2)).$$

On the other hand, if $j_1 = j_2 = j$, then

$$\begin{aligned} d_{\mathbb{R}^m}((v_1, j_1), (v_2, j_2)) &= d_{\mathbb{R}^{m-1}}(v_1, v_2) \\ &= d_{\mathbb{R}^{m-1}}(v_1 - j v_g, v_2 - j v_g) \\ &= d_{\mathbb{R}^{m-1}}(v_1 - j_1 v_g, v_2 - j_1 v_g) \\ &\geq a \| |f_G(v_1 - j_1 v_g)\rangle - |f_G(v_2 - j_2 v_g)\rangle \| \\ &= a \| |f_{G_{\mathfrak{a}}}(v_1, j_1)\rangle - |f_{G_{\mathfrak{a}}}(v_2, j_2)\rangle \|, \end{aligned}$$

for $a = \frac{\sqrt{\pi n} s}{4\nu} + 1$. Therefore, we have the Lipschitz condition is always satisfied for $a = \frac{\sqrt{\pi n} s}{4\nu} + 2$.

b) The (r, ε) condition: we simply need to notice that $d_{\mathbb{R}^m/\Lambda_{\mathfrak{a}}}((v_1, j_1), (v_2, j_2)) \leq d_{\mathbb{R}^{m-1}/\mathcal{O}^*}(v_1 - j_1 v_g, v_2 - j_2 v_g)$. This is indeed the case because

$$\begin{aligned} d_{\mathbb{R}^m/\Lambda_{\mathfrak{a}}}((v_1, j_1), (v_2, j_2)) &= \inf_{\substack{u \in \mathcal{O}^* \\ j \in \mathbb{Z}}} \| (v_1, j_1) - (v_2, j_2) - (j v_g, j) - (u, 0) \| \\ &\leq \inf_{u \in \mathcal{O}^*} \| (v_1 - j_1 v_g, 0) - (v_2 - j_2 v_g, 0) - (u, 0) \| \text{ (by choosing } j = j_1 + j_2) \\ &= d_{\mathbb{R}^{m-1}/\mathcal{O}^*}(v_1 - i_1 v_g, v_2 - i_2 v_g). \end{aligned}$$

This means that for $r = \log\left(1 + (s\sqrt{n})^{n-1} 2\nu\sqrt{n}\right)$ and $\varepsilon = 3/4$, if $d_{\mathbb{R}^m/\Lambda_{\mathfrak{a}}}((v_1, j_1), (v_2, j_2)) \geq r$, then $d_{\mathbb{R}^{m-1}/\mathcal{O}^*}(v_1 - i_1 v_g, v_2 - i_2 v_g) \geq r$ as well, and then necessarily

$$\langle f_{G_{\mathfrak{a}}}(v_1, j_1) | f_{G_{\mathfrak{a}}}(v_2, j_2)\rangle = \langle f_G(v_1 - j_1 v_g) | f_G(v_2 - j_2 v_g)\rangle \leq \varepsilon.$$

Reduction to the case $G = \mathbb{R}^m$ We described a (a,r,ε)-oracle $f_{G_{\mathfrak{a}}}$ on a hyperplane $G_{\mathfrak{a}}$ of $\mathbb{R}^n \times \mathbb{Z}$ hiding the lattice $\Lambda_{\mathfrak{a}}$ for

$$a = \frac{\sqrt{\pi n} s}{4\nu} + 2 \;\; \varepsilon = 3/4 \;, \;\; r = \log\left(1 + (s\sqrt{n})^{n-1} 2\nu\sqrt{n}\right).$$

To apply [6, Th. 6.1], we need a function $f_{\mathfrak{a}}$ on $\mathbb{R}^m$ for some m that hides the lattice $\Lambda_{\mathfrak{a}}$ and which is an $(\overline{a}, \overline{r}, \overline{\varepsilon})$-oracle in $\mathbb{R}^m$ for some $\overline{a}$, $\overline{r}$, $\overline{\varepsilon}$, not necessarily equal to a, r, ε. A general guideline for performing such a task is given in [6, 6.1]. By following it, we find such a function $f_{\mathfrak{a}}$, and we can apply the quantum algorithm of [6] to derive $\Lambda_{\mathfrak{a}}$, thus obtaining a generator for $\mathfrak{a}$.

First of all, we can easily turn $f_{G_{\mathfrak{a}}}$ defined on the hyperplane $G_{\mathfrak{a}}$ into a function defined over $\mathbb{R}^{n-1} \times \times \mathbb{Z}$ with the intermediate operation

$$\begin{array}{ccc} \mathbb{R}^n \times \mathbb{Z} & \xrightarrow{\phi} & G_{\mathfrak{a}} \\ (v, j) & \longrightarrow & (v_1, \cdots, v_{r_1+r_2-1}, -\sum_i v_i + j \log|\mathcal{N}(\mathfrak{a})|, j) \end{array}.$$

Proposition 5. *If $f_{G_{\mathfrak{a}}}$ is an (a, r, ε)-oracle hiding $\Lambda_{\mathfrak{a}}$ on $G_{\mathfrak{a}}$, then $f_{G_1} := f_{G_{\mathfrak{a}}} \circ \phi$ is an (a_1, r, ε)-oracle hiding $\Lambda_{\mathfrak{a}}$ on $G_1 := \mathbb{R}^{n-1} \times \times \mathbb{Z}$ for*

$$a_1 = a\sqrt{6(r_1 + r_2 - 1)} \log|\mathcal{N}(\mathfrak{a})|.$$

Proof. The fact that the (r,ε) condition is preserved is obvious because we are dropping one coordinate. This means that if the distance in $G_1 = \mathbb{R}^{n-1} \times \mathbb{Z}$ (modulo $\Lambda_\mathfrak{a}$) is greater than r, then so is the distance in $G_\mathfrak{a}$ (modulo $\Lambda_\mathfrak{a}$), and therefore the inner product of the two states has to be less than ε. The Lipschitz condition comes from the fact that

$$\begin{aligned}
\| |f_{G_1}(x)\rangle - |f_{G_1}(y)\rangle \|^2 &= \| |f_{G_\mathfrak{a}}(\phi(x))\rangle - |f_{G_\mathfrak{a}}(\phi(y))\rangle \|^2 \\
&\leq a^2 d^2(\phi(x), \phi(y)) \\
&= a^2 \left(\sum_{k \leq r_1+r_2-1} v_k^2 + \left(j \log |\mathcal{N}(\mathfrak{a}) - \sum_k v_k \right)^2 + j^2 \right) \\
&\text{(where } (v,j) := x - y) \\
&= a^2 \left(\sum_{k \leq r_1+r_2-1} v_k^2 + j^2 \log^2 |\mathcal{N}(\mathfrak{a})| + \sum_{k \leq r_1+r_2-1} v_k^2 \right. \\
&\left. -2j \log |\mathcal{N}(\mathfrak{a})| \left(\sum_{k \leq r_1+r_2-1} v_k \right) + 2 \sum_{k \neq l \leq r_1+r_2-1} v_k v_l + j^2 \right) \\
&\leq 6a^2 (r_1 + r_2 - 1) \log^2 |\mathcal{N}(\mathfrak{a})| \left(\sum_{k \leq r_1+r_2-1} v_k^2 + j^2 \right) \\
&= \left(6a^2 (r_1 + r_2 - 1) \log^2 |\mathcal{N}(\mathfrak{a})| \right) d_{G_1}^2(x,y).
\end{aligned}$$

We have now a function on $\mathbb{R}^k \times \mathbb{Z}^l$ for $k = n-1$ and $l = 1$ that hides $\Lambda_\mathfrak{a}$ and that is an (a_1, r, ε)-oracle on $\mathbb{R}^k \times \mathbb{Z}^l$. Following the guidelines of [6, 6.1], we can turn it into an $(\overline{a}, \overline{r}, \overline{\varepsilon})$-oracle $f_\mathfrak{a}$ on $\mathbb{R}^{k+l}$ that hides $\Lambda_\mathfrak{a}$. To do so, we define

$$|f_\mathfrak{a}(\mathbf{x}, x_1, \cdots, x_l)\rangle := \sum_{z_1, \cdots, z_l \in \{0,1\}} \left(\bigotimes_{j=1}^{l} |\psi(x_j, z_j)\rangle \right) \otimes |f_{G_1}(\mathbf{x}, s(x_1, z_1), \cdots, s(x_l, z_l)\rangle,$$

where $s(x,z) = \lfloor x/\lambda \rfloor + z$, $|\psi(x,z)\rangle = \cos\left(\frac{\pi}{2}\right) \mathrm{str}_\nu(t)$ with $t = x/\lambda - s(x,z)$ for a lower bound λ on the shortest vector of $\Lambda_\mathfrak{a}$.

Proposition 6. *The function $f_\mathfrak{a}$ hides the lattice $\Lambda_\mathfrak{a}$ and satisfies conditions (1), (2) and (3) for the parameters $\overline{a}, \overline{r}, \overline{\varepsilon}$ defined by*

$$\begin{aligned}
\overline{a}^2 &= a_1^2 + l \left(\frac{\pi}{2\nu\lambda} (1+\nu) \right)^2 \\
&= 6(r_1 + r_2 - 1) \log^2 |\mathcal{N}(\mathfrak{a})| \left(\frac{\sqrt{\pi n} s}{4\nu} + 2 \right)^2 + l \left(\frac{\pi}{2\nu\lambda} (1+\nu) \right)^2 \\
\overline{r}^2 &= \left(\log \left(1 + (s\sqrt{n})^{n-1} 2\nu\sqrt{n} \right) \right)^2 + l(2\nu\lambda)^2 \\
\overline{\varepsilon} &= 3/4
\end{aligned}$$

Proof. See [6, 6.1].

7 Computing a short generator of a principal ideal in $\mathbb{Q}(\zeta_{p^n})$

Several cryptosystems including the multilinear maps of Garg et al. [8] and the Smart Vercauteren homomorphic encryption scheme [17] rely on the hardness of finding a short generator in an ideal $\mathfrak{a}$ of the ring of integers $\mathcal{O}_K$ of a field of the form $K = \mathbb{Q}(\zeta_{p^n})$ (in the particular case $p = 2$). In this section we show how to combine the algorithm for the PIP described in the previous section with known techniques, in particular the recent reduction short-PIP to PIP proved by Cramer, Ducas, Peikert and Regev [5], to perform a key recovery attack. The general idea of first solving the PIP and then using a reduction from short-PIP to PIP probably goes back to the time when cryptosystems relying on the short-PIP were defined. However, in the absence of algorithms for efficiently solving these problems, there had not been any public description of it until recently. To the best of our knowledge, the first time such an approach was publicly suggested was by Bernstein [1]. The attack of Campbel et al. [4] also relies on the same idea. The outline of the algorithm for solving the short-PIP we present here, which is based on the same general strategy, is the following:

1. Compute the ideal $\mathfrak{b}$ of $K^+ = \mathbb{Q}(\zeta_{p^n} + \zeta_{p^n}^{-1})$ generated by $\mathcal{N}_{K/K^+}(\mathfrak{a})$.
2. Find $(\log |g|_1, \cdots, \log |g|_n)$ where $n = \deg(K^+)$ and g is a generator of $\mathfrak{b}$.
3. Compute a compact representation $|g| = g_0 \cdot g_1^2 \cdots g_k^{2^k}$ of $|g|$.
4. Solve each norm equation $\mathcal{N}_{K/K^+}(\alpha_i) = g_i$ with the generalization of the Howgrave-Graham-Szydlo algorithm [13] of Garg, Gentry and Halevi [8, 7.3].
5. Find the vector $v = \sum_i x_i \operatorname{Log}(u_i) \in \operatorname{Log}(\mathcal{O}_K^*)$ that is the closest to $\sum_j 2^k \operatorname{Log}(\alpha_j)$ with the techniques described by Cramer et al. [5] where $u_i = (\zeta_{p^n}^i - 1)/(\zeta_{p^n} - 1)$.
6. Compute $\alpha = \left(\prod_j \alpha_j^{2^j}\right) \cdot (\prod_i u_i^{x_i})$ (modulo small primes, then reconstruct it by the CRT).
7. Return either α or $\bar{\alpha}$ (depending on which one generates $\mathfrak{a}$).

Step 1 brings the PIP in the totally real field $K^+ = \mathbb{Q}(\zeta_{p^n} + \zeta_{p^n}^{-1})$. This way, we can use the oracle to the function $f_{\mathfrak{b}}$ described in Section 6 together with the continuous Hidden Subgroup Problem of [6] to find $(\log |g|_1, \cdots, \log |g|_n)$ where $n = \deg(K^+)$ and g is a generator of $\mathfrak{b}$. From there, we can compute a compact representation of $|g|$, which is a generator of $\mathfrak{b}$ (the information on the sign of g cannot be retrieved). This is done by using the methods described in [7, Alg. 7.53] and [3, Sec. 5]. Given a constant $l > 0$ (which we can set to $l = 2$), $\mathfrak{b}$, and $\operatorname{Log}(g)$, we get polynomial sized (on the integral basis) elements g_i and a polynomial bound k such that

$$|g| = g_0 \cdot g_1^l \cdots g_k^{l^k}.$$

This compact representation allows us to solve the norm equation $\mathcal{N}_{K/K^+}(x) = |g|$ in polynomial time and to get the output x in exact compact representation. We proceed by simply solving each each norm equation $\mathcal{N}_{K/K^+}(\alpha_i) = g_i$ involving the polynomially sized g_i. This gives us a generator $\beta = \prod_i u_i^{x_i}$ of either $\mathfrak{a}$ or $\bar{\mathfrak{a}}$ in compact representation. The cyclotomic units $u_i = (\zeta_{p^n}^i - 1)/(\zeta_{p^n} - 1)$ are conjectured to generate $\mathcal{O}_K^*$, therefore, all the other generators are of the form $\beta \cdot (\prod_i u_i^{x_i})$. Finding the x_i leading to the shortest generator boils down to an instance of the Bounded Distance Decoding problem in $\operatorname{Log}(\mathcal{O}_K^*)$. It was observed by Campbel et al. [4] and later proved by Cramer et al. [5] that Babai's round-off algorithm allows us to solve this problem in polynomial time. Then all we have to do is to compute $\alpha = \left(\prod_j \alpha_j^{2^j}\right) \cdot (\prod_i u_i^{x_i})$ modulo a collection of small primes and then to reconstruct it.

Proposition 7. *There is an efficient quantum algorithm that recovers the short generator of an input ideal* $\mathfrak{a}$ *in a cyclotomic field of the form* $\mathbb{Q}(\zeta_{p^n})$.

8 Conclusion and significance

We provided the first polynomial time algorithm to compute the generator of a principal ideal in a totally real number field of arbitrary degree. We showed that it derives from the results of [6] in a rather straightforward way, and despite the fact that it only applies to totally real fields, it is a very significant result for post-quantum cryptography. Indeed. together with the reduction from the short-PIP to the PIP, originally observed by Campbel, Groves and Shepherd [4] and later proved by Cramer, Ducas, Peikert and Regev [5], it is enough to attack cryptosystems based on the hardness of finding a short generator of a principal ideal in a cyclotomic field of prime power conductor in quantum polynomial time. These include the multilinear maps of Garg, Gentry and Halevi [8] and the fully homomorphic encryption scheme of Smart and Vercauteren [17].

Strictly speaking the algorithm we discussed in Section 6 does not solve the standard Principal Ideal Problem since the algorithm cannot decide if an input ideal (of a totally real field) is principal (rather it takes as promise that it is principal). Further generalizations of the methods of [6] will lead to the resolution of related problems in number theory in arbitrary fields including the PIP, the computation of the ideal class group, the computation of S-units, or the resolution of norm equations. For these problems however, more technical contributions on the metrical properties of lattices will be needed to apply the general HSP framework of [6]. An extended abstract giving directions on how to solve these problems in polynomial time will be published in the proceedings of the SODA 2016 conference. These methods are conditional on a generalization of the HSP framework of [6] which is still under development.

References

1. D. Bernstein. A subfield-logarithm attack against ideal lattices. `http://blog.cr.yp.to/20140213-ideal.html`. Feb. 2014.
2. J.-F. Biasse. Subexponential time relations in large degree number fields. Submitted to *Advances in Mathematics of Communications*.
3. J.-F. Biasse and C. Fieker. Subexponential class group and unit group computation in large degree number fields. *LMS Journal of Computation and Mathematics*, 17:385–403, 1 2014.
4. P. Campbel, M. Groves, and D. Shepherd. SOLILOQUY, a cautionary tale. `http://docbox.etsi.org/Workshop/2014/201410_CRYPTO/S07_Systems_and_Attacks/S07_Groves_Annex.pdf`, 2014.
5. R. Cramer, L. Ducas, C. Peikert, and O. Regev. Recovering short generators of principal ideals in cyclotomic rings. *IACR Cryptology ePrint Archive*, 2015:313, 2015.
6. K. Eisenträger, S. Halgren, A. Kitaev, and F. Song. A quantum algorithm for computing the unit group of an arbitrary degree number field. In *Proceedings of the 46th Annual ACM Symposium on Theory of Computing*, STOC '14, pages 293–302, New York, NY, USA, 2014. ACM.
7. C. Fieker. Algorithmic Number Theory. Lecture notes available at http://www.mathematik.uni-kl.de/agag/mitglieder/professoren/prof-dr-claus-fieker, 2014.
8. S. Garg, C. Gentry, and S. Halevi. Candidate multilinear maps from ideal lattices. In T. Johansson and P. Nguyen, editors, *Advances in Cryptology - EUROCRYPT 2013, 32nd Annual International Conference on the Theory and Applications of Cryptographic Techniques, Athens, Greece, May 26-30, 2013. Proceedings*, pages 1–17, 2013.
9. C. Gentry and M. Szydlo. Cryptanalysis of the Revised NTRU Signature Scheme. In Lars Knudsen, editor, *Advances in Cryptology EUROCRYPT 2002*, volume 2332 of *Lecture Notes in Computer Science*, pages 299–320. Springer Berlin Heidelberg, 2002.

10. L. Hales. *The Quantum Fourier Transform and extensions of the Abelian Hidden Subgroup Problem*. PhD thesis, University of California Berkeley, 2002.
11. S. Hallgren. Fast quantum algorithms for computing the unit group and class group of a number field. In H. Gabow and R. Fagin, editors, *Proceedings of the 37th Annual ACM Symposium on Theory of Computing, Baltimore, MD, USA, May 22-24, 2005*, pages 468–474. ACM, 2005.
12. S. Hallgren. Polynomial-time quantum algorithms for pell's equation and the principal ideal problem. *Journal of the ACM*, 54(1):4:1–4:19, March 2007.
13. N. Howgrave-Graham and M. Szydlo. A method to solve cyclotomic norm equations $f * \bar{f}$. In D. Buell, editor, *Algorithmic Number Theory*, volume 3076 of *Lecture Notes in Computer Science*, pages 272–279. Springer Berlin Heidelberg, 2004.
14. D. Micciancio and S. Goldwasser. *Complexity of Lattice Problems: a cryptographic perspective*, volume 671 of *The Kluwer International Series in Engineering and Computer Science*. Kluwer Academic Publishers, Boston, Massachusetts, March 2002.
15. R. Pinch. SOLILOQUY, a cautionary tale. talk at the ICERM workshop on the mathematics of lattices and cybersecurity, 2015.
16. P. Shor. Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. *SIAM Journal on Computing*, 26(5):1484–1509, 1997.
17. N. Smart and F. Vercauteren. Fully homomorphic encryption with relatively small key and ciphertext sizes. In P. Nguyen and D. Pointcheval, editors, *Public Key Cryptography - PKC 2010*, volume 6056 of *Lecture Notes in Computer Science*, pages 420–443. Springer Berlin Heidelberg, 2010.