

SoK: On the Security of Cryptographic Problems from Linear Algebra

Carl Bootland¹, Wouter Castryck¹, Alan Szepieniec², Frederik Vercauteren¹

carl.bootland@esat.kuleuven.be, wouter.castryck@esat.kuleuven.be,
alan@nervos.org, frederik.vercauteren@esat.kuleuven.be

¹ imec-COSIC, Department of Electrical Engineering, KU Leuven

² Nervos Foundation, Panama City

Abstract. There are two main aims to this paper. Firstly, we survey the relevant existing attack strategies known to apply to the most commonly used lattice-based cryptographic problems as well as to a number of their variants. In particular, we consider attacks against problems in the style of LWE, SIS and NTRU defined over rings of the form $\mathbb{Z}[X]/(f(X), g(X))$, where classically $g(X) = q$ is an integer modulus. We also include attacks on variants which use only large integer arithmetic, corresponding to the degree one case $g(X) = X - c$. Secondly, for each of these approaches we investigate whether they can be generalised to the case of a polynomial modulus $g(X)$ having degree larger than one, thus addressing the security of the generalised cryptographic problems from linear algebra introduced by Bootland et al. We find that some attacks readily generalise to a wide range of parameters while others require very specific conditions to be met in order to work.

Keywords: lattice-based cryptography, noisy linear algebra, learning with errors, short integer solutions, NTRU

1 Introduction

The area of lattice-based cryptography has rapidly grown to be one of the leading candidates offering post-quantum security for a wide variety of cryptographic primitives. In this work we will consider the three most widely used cryptographic problems within lattice-based cryptography and their generalisations; namely, the LWE, SIS and NTRU problems. Concretely, we will explain how the most relevant attack strategies can be applied to attempt to solve these hard problems using a classical computer. We do not consider any possible quantum attacks against our problems besides generic speed-ups to classical attacks due to Grover’s algorithm. As we will see, many of the same ideas are used in attacking each of these different problems and indeed a number of approaches involve reducing one of the three problems to one of the others.

The first of our three problems to appear was the short integer solutions (SIS) problem of Ajtai [3]. Informally, the problem is, given a set of vectors, to find a linear dependence between them in which the scalars used are all

small. Originally, the vectors were uniformly random elements of \mathbb{Z}_q^n for suitable integers q and n and the scalars being integers. This was first generalised to the ring version [60, 61] in which the vectors have only one component that is from some polynomial quotient ring $R_q = \mathbb{Z}_q[X]/(f(X))$ for a suitable monic polynomial $f \in \mathbb{Z}[X]$ of degree n and with the scalars from the ‘parent’ ring $R = \mathbb{Z}[X]/(f(X))$, and later to the module version [53] in which the vectors can have more than one components but still have entries from the ring R_q ; scalars remain elements of R .

The second problem is called the learning with errors (LWE) problem and was introduced by Regev in [65]. The problem can be described as that of solving a system of noisy linear equations, that is the constant terms of a system of linear equations have been modified by adding some small error terms and the task is to solve the system and hence also determine these errors. Again, originally the system of noisy linear equations was defined over the ring \mathbb{Z}_q but this was later modified and generalised to rings of the form R_q as defined above [71, 25] with the new problems eventually being known as the polynomial (module) learning with errors problem. A more mathematically involved modification to the problem was proposed by Lyubashevsky, Peikert and Regev in [57] where they argue this gives the ‘right’ definition of the modified problem using the ring of integers of a number field and its dual ideal; this problem is known as the ring learning with errors problem and also has a generalisation to modules [53].

Lastly, the third problem we discuss is the NTRU problem. Here, one must write a given ring element as a quotient of two small elements in a finite ring. Unlike the previous two problems, this problem was already initially instantiated using a ring of the form R_q as described above.

Recently, the NTRU problem was generalised in a new direction by Aggarwal et al. in an early version of [1] and the same generalisation was quickly applied to the (polynomial) learning with errors problem in the final version of their paper [2]. The connection between these new problems and the NTRU and LWE problems was made explicit in [24] but can be most simply explained by replacing the integer modulus q used to define the ring R_q by the linear modulus $X - 2$ which leads to a completely different structure, namely the new ring R_{X-2} which in this case is equivalent to large integer arithmetic in \mathbb{Z}_M for some large Mersenne number M .

In [24] the authors further propose to generalise all three problems to use the ring $R_g = \mathbb{Z}[X]/(f(X), g(X))$ for a general polynomial g on the assumption that this ring is finite and elements enjoy having a ‘nice’ canonical representative. We will present the exact formulation of the problems we are considering in Section 2.4.

As well as presenting attacks against the problems with integer moduli we also describe the newer attacks against instances with linear moduli. To the best of our knowledge this is the first time both cases have appeared together which allows us to highlight their similarities and differences. Furthermore, we assess how to generalise these attacks to deal with moduli of degree larger than one, an area which has not been studied in any great detail before.

We find that some attacks such as the straightforward lattice attack against the SIS problem can be applied more or less unchanged to the more general case. Other attacks such as the Blum–Kalai–Wasserman algorithm [21] and the dual attack on LWE [62] can be employed on a wide range of parameters though some restrictions are still required. Finally, attacks such as the subfield attack on the NTRU problem [5, 29] and the Arora-Ge attack [13] on LWE, which are only applicable in rather special circumstances, can be modified to apply to more general instances. However, they remain confined to working only in a restricted set of circumstances.

Outline

In Section 2 we introduce some notation and the definition of the problems considered in this work, with examples, this summarises [24, Section 2 and 3] to which we direct the reader if necessary. Furthermore, we give a number of preliminary results which help to illustrate how the generalised problems behave like the standard problems in some ways but not in others. In Section 3 we look at attacks against the SIS family of problems, followed by attacks on the NTRU family of problems in Section 4, while in Section 5 we describe attacks which work against the LWE family of problems. In each case, we examine how the attack can be generalised to work for a larger range of parameters within the framework of the problems given in [24].

2 The problems and basic results

2.1 Notation

As previously mentioned, we will be working with a general quotient polynomial ring, which we call the *parent ring*, $R := \mathbb{Z}[X]/(f(X))$, for which f is monic of degree n , as well as a quotient of this ring $R_g = \mathbb{Z}[X]/(f(X), g(X))$ for some polynomial g coprime to f . We will refer to g as the *ciphertext modulus* even though we do not actually consider any concrete encryption schemes in this work. Further, in this paper, as in [24], we will only consider the ring R_g for g such that $R_g = \mathbb{Z}[X]/(a, r(X))$ for an integer a and monic polynomial $r(X)$. We note that if one chooses f and g randomly and they satisfy our conditions then it is likely that r will be a linear polynomial and hence R_g is just the ring of integers modulo a , of course, one will not choose f and g at random but it is still true that r will have degree smaller than n unless g is an integer in which case $r = f$. An explanation of how to determine if g satisfies this constraint and how to compute such an a and $r(X)$ can be found in [24]. We merely mention here that this condition is not so restrictive and covers all currently known choices for g in the literature. The notation a and r will be used consistently throughout this paper.

We will also denote by $\text{Rep}(R_g)$ the following set of representatives for R_g in $\mathbb{Z}[X]$:

$$\text{Rep}(R_g) := \left\{ \alpha_0 + \alpha_1 X + \cdots + \alpha_{\deg(r)-1} X^{\deg(r)-1} \mid \alpha_i \in \{0, \dots, a-1\} \right\}.$$

We have the natural bijection $\text{rep}_g: R_g \rightarrow \text{Rep}(R_g)$ given by sending an element $\mathbf{x} \in R_g$ to the unique element $\text{rep}_g(\mathbf{x}) \in \text{Rep}(R_g)$ such that $\mathbf{x} = \text{rep}_g(\mathbf{x}) + (f(X), g(X))$. We abuse notation slightly by allowing arguments from R , so for $\mathbf{x} \in R$ we define $\text{rep}_g(\mathbf{x}) = \text{rep}_g(\mathbf{x} \bmod gR)$. Further, we allow arguments to be vectors or matrices with entries in either R_g or R by applying the map in a coordinate-wise manner

We also define the map $\iota: R \rightarrow \mathbb{Z}^n$ which takes an element, whose coset representative is of lowest degree, $c_0 + c_1X + \dots + c_{n-1}X^{n-1} + (f(X))$, to the (row) vector of its coefficients, $(c_0, c_1, \dots, c_{n-1})$. We naturally extend this to $\iota: R^m \rightarrow \mathbb{Z}^{mn}$, for a natural number m , by concatenating the vectors given by applying ι component-wise, as well as to matrices $\iota: R^{\ell \times m} \rightarrow \mathbb{Z}^{\ell mn}$ by concatenating the vectors given by applying ι to each row vector in R^m .

By an abuse of notation we define a ‘norm’ on the ring R by $\|\cdot\|: R \rightarrow \mathbb{R}_{\geq 0}$. This ‘norm’ will in general not satisfy $\|u \cdot v\| \leq \|u\| \cdot \|v\|$ so will not be a ring norm. However, we will require that the product of elements of ‘small’ norm will also be somewhat ‘small’. It is intuitive to think of the norm as being derived from one on \mathbb{R}^n by noting that $R \cong \mathbb{Z}^n$ as abelian groups (for example ι is one such isomorphism) and embedding \mathbb{Z}^n into \mathbb{R}^n . Throughout, we will use the isomorphism ι between R and \mathbb{Z}^n along with the standard embedding of \mathbb{Z}^n into \mathbb{R}^n , however any embedding into \mathbb{R}^n can be used instead depending on the ‘norm’ used. In particular, if R is a ring of integers of a number field then one can use the canonical embedding (see Section 2.3 below) to define ι and the corresponding norm instead.

Elements of free modules are taken to be row vectors, for example R^ℓ is the free module of ℓ -tuples having entries in R which we write as a row vector. We extend $\|\cdot\|$ to vectors and matrices by taking the maximum of the norm applied to all the entries; other choices are possible.

2.2 Distributions of small elements

As well as a notion of smallness we will also need to use distributions of small elements over R when defining the problems. One can do this by considering an elliptical Gaussian distribution on \mathbb{R}^n and pulling it back via the inverse of the map ι . This will however give an element of $R \otimes_{\mathbb{Q}} \mathbb{R}$ so we typically discretise the distribution, either before or after pulling it back.

Formally, the normalised Gaussian function with parameter α , $\varphi_\alpha: \mathbb{R}^n \rightarrow \mathbb{R}^+$ is defined as $\varphi_\alpha(\mathbf{x}) = \exp(-\pi \|\mathbf{x}\|^2 / \alpha^2) / \alpha^n$. One may replace $\|\mathbf{x}\|^2$ by $\mathbf{x}D\mathbf{x}^T$ for a diagonal matrix D to consider an elliptical Gaussian. We denote the n -dimensional (spherical) discrete Gaussian distribution with width parameter α as $\mathcal{D}_{\mathbb{Z}^n, \alpha}$ and define it to take the value $\mathbf{x} \in \mathbb{Z}^n$ with probability $\varphi_\alpha(\mathbf{x}) / \varphi_\alpha(\mathbb{Z}^n)$ where $\varphi_\alpha(\mathbb{Z}^n) = \sum_{\mathbf{x} \in \mathbb{Z}^n} \varphi_\alpha(\mathbf{x})$.

2.3 The canonical embedding and canonical norm

When the polynomial f defining the parent ring R is irreducible then the quotient field of R is a number field, namely it is of the form $K = \mathbb{Q}[X]/(f(X))$. Such a field can be mapped into the field \mathbb{C} in a one-to-one manner and in fact there are n such mappings, each defined by sending X to one of the complex roots of f , and these are all distinct. Let us denote these field embeddings by σ_i for $i = 1, \dots, n$, ordered so that the first s_1 are defined by any real roots of f and the latter $2s_2$ come in pairs defined by a complex root of f and its conjugate. The canonical embedding is defined to be the map $\boldsymbol{\sigma} = (\sigma_1, \dots, \sigma_n): K \rightarrow \mathbb{R}^{s_1} \times \mathbb{C}^{2s_2}$. We can define a corresponding norm with respect to the canonical embedding called the canonical norm as

$$\|x\|^{\text{can}} = \sqrt{\sum_{i=1}^n |\sigma_i(x)|^2}.$$

2.4 The problems and main examples

Here we introduce the three general problems as given in [24] and then explain which parameter choices give the main examples of the specific problems considered previously. The first problem is that based on the short integer solution problem.

Problem (Ideal short integer solution problem (Ideal-SIS $_{f,g,m,\ell,\rho}$)). For integers $\ell > m > 0$ and a positive real ρ , sample ℓ elements from R_g^m uniformly at random and denote them by $\mathbf{a}_1, \dots, \mathbf{a}_\ell$. The ideal short integer solution problem, Ideal-SIS $_{f,g,m,\ell,\rho}$, is to find a non-zero vector $\mathbf{z} = (z_1, \dots, z_\ell) \in R^\ell$ such that $\|\mathbf{z}\| \leq \rho$ and $\sum_{i=1}^{\ell} \mathbf{a}_i \cdot z_i = \mathbf{0}$.

Writing $A \in R_g^{m \times \ell}$ for the matrix having the vectors $\mathbf{a}_1^T, \dots, \mathbf{a}_\ell^T$ as its columns, we can rewrite the final condition as $A\mathbf{z}^T = \mathbf{0}^T$. One can consider an inhomogeneous version of this problem in which we replace this condition by $A\mathbf{z}^T = \mathbf{t}^T$ for some given target vector $\mathbf{t} \in R_g^m$. Secondly, for the learning with errors type problems there are two distinct variants, a search and a decision version.

Problem (Ideal learning with errors search problem (Ideal-LWE $_{f,g,m,k,\ell,\chi}$)). Let χ be a distribution of small elements over R and let k, ℓ and m be positive integers. Sample a uniformly random secret matrix $\mathbf{s} \in R_g^{m \times k}$. The ideal learning with errors search problem, Ideal-LWE $_{f,g,m,k,\ell,\chi}$, is to find \mathbf{s} given the tuple of matrices $(\mathbf{a}, \mathbf{b}) \in R_g^{\ell \times m} \times R_g^{\ell \times k}$ where $\mathbf{a} \in R_g^{\ell \times m}$ is sampled uniformly at random and $\mathbf{b} = \mathbf{a}\mathbf{s} + \mathbf{e} \in R_g^{\ell \times k}$ with \mathbf{e} sampled from $\chi^{\ell \times k}$.

Problem (Ideal learning with errors decision problem (Ideal-DLWE $_{f,g,m,k,\ell,\chi}$)). Let χ, k, ℓ, m and \mathbf{s} be as in the previous problem. Given $(\mathbf{a}, \mathbf{b}) \in R_g^{\ell \times m} \times R_g^{\ell \times k}$ where \mathbf{a} is uniformly random and \mathbf{b} is either uniformly random or of the form $\mathbf{b} = \mathbf{a}\mathbf{s} + \mathbf{e}$ for some \mathbf{e} sampled from $\chi^{\ell \times k}$, the Ideal-DLWE $_{f,g,m,k,\ell,\chi}$ problem is to determine which is the case.

While we have described the Ideal-LWE problem using a fixed ℓ it is common to allow the attacker to choose the value of ℓ so long as it remains polynomially bounded in n . In this case one usually considers the distribution $\mathcal{A}_{\mathbf{s},\chi}$ which returns a single sample, i.e. $\ell = 1$, for a fixed secret \mathbf{s} and allow the attacker access to an oracle which returns in constant time elements from $\mathcal{A}_{\mathbf{s},\chi}$. Further, a short or sparse secret variant of the problem is sometimes considered in which the secret \mathbf{s} has entries sampled from another distribution of small elements such as those polynomials having at most a fixed number of non-zero coefficients.

Lastly, we have the generalisation of the NTRU problem.

Problem (Ideal NTRU problem (Ideal-NTRU $_{f,g,m,\chi,\rho}$)). *Let χ be a distribution of small elements over R , ρ be a positive real which bounds χ and m be a positive integer. Sample two matrices $\mathbf{u}, \mathbf{v} \leftarrow \chi^{m \times m}$ such that \mathbf{u} is invertible when considered modulo g . Compute, as an element of $R_g^{m \times m}$ the quotient $\mathbf{h} = \mathbf{v}\mathbf{u}^{-1}$. The Ideal NTRU problem Ideal-NTRU $_{f,g,m,\chi,\rho}$ is, given \mathbf{h} and ρ , to find two matrices $\mathbf{u}', \mathbf{v}' \in R^{m \times m}$ with \mathbf{u}' invertible modulo g , $\mathbf{h} = \mathbf{v}'\mathbf{u}'^{-1} \bmod g$, $\|\mathbf{u}'\| < \rho$ and $\|\mathbf{v}'\| < \rho$.*

One may wish to sample the entries of \mathbf{u} and \mathbf{v} from slightly different distributions, for example to ensure that \mathbf{u} is invertible with a reasonable probability, however for simplicity we will not distinguish the two distributions.

Main Examples If we fix g to be an integer q then Ideal-SIS $_{X,q,m,\ell,\rho}$ is the original SIS problem [3] (any monic linear f can be used to remove the polynomial structure), Ideal-SIS $_{f,q,1,\ell,\rho}$ is the Ring-SIS problem [60, 61] and Ideal-SIS $_{f,q,m,\ell,\rho}$ is Module-SIS which bridges the two [53]. Furthermore, Ideal-LWE $_{X,q,m,1,\ell,\chi}$ is the original LWE problem when χ is a (discrete) Gaussian distribution [65], Ideal-LWE $_{f,q,1,1,\ell,\chi}$ is the polynomial LWE problem [71]³, and its extension to modules is Ideal-LWE $_{f,q,m,1,\ell,\chi}$ [25]. Finally, Ideal-NTRU $_{f,q,1,\chi,\rho}$ gives the typical NTRU problems depending on the choice of f and χ [46, 31, 19].

On the other hand, examples of g not being a constant polynomial are Ideal-NTRU $_{X^{n-1},X-2,1,\chi,\rho}$, with χ returning binary polynomials of Hamming weight h , which gives the Mersenne low Hamming ratio search problem, MLHR $_{n,h}$, and short secret Ideal-LWE $_{X^{n-1},X-2,1,1,\chi}$ with the same χ gives the Mersenne low Hamming combination search problem, MLHC $_{n,h}$ [1]. Also, for an integer q and χ a discrete Gaussian distribution, Ideal-LWE $_{X^{n+1},X-q,1,1,\ell,\chi}$, is the integer ring learning with errors problem of Gu [41]. Lastly, Hamburg [43] considered a module version of the decision problem, namely the Ideal-DLWE $_{X^n-X^{n/2}-1,X-q,m,1,\ell,\chi}$ problem for certain choices of n and q and with χ returning ternary polynomials.

2.5 Recognizing small elements in R_g

In this section we focus on the case that small elements in R have small coefficients; however the same discussion is relevant when ι is some other embedding though it may play out differently in that case.

³ Stehlé et al. originally called their problem Ideal-LWE however this is not the same as our problem which is much more general.

When the ciphertext modulus is an integer it is easy to recognize small elements in R_g as we can simply reduce every coefficient into the range $(-g/2, g/2]$ and smallness (which will be relative to g) is readily apparent. Alternatively, when $g = X - b$ then $R_g \cong \mathbb{Z}/f(b)\mathbb{Z}$ so we can take an element of R_g , lift it modulo $f(b)$ and then expand it using a (balanced) b -ary expansion before looking at the coefficients of this expansion to determine whether the element is a reduction of a small element in R or not.

For a ciphertext modulus of degree two or larger things get more complicated in general. One obvious approach which works for any f and g is to consider the lattice

$$\{\mathbf{x} \in \mathbb{Z}^n \mid \mathbf{x} = \iota(\mathbf{y}) \text{ for some } \mathbf{y} \in R \text{ such that } \mathbf{y} \equiv \mathbf{0} \pmod{gR}\}$$

and note that (any lift of) a small element of R_g is close to this lattice. Given an element $\mathbf{z} \in R_g$, one can attempt to solve the closest vector problem in this lattice with target vector any lift of \mathbf{z} using, for example, the embedding technique. In doing so, one recovers an element $\mathbf{w} \in R$ such that $\mathbf{w} \equiv \mathbf{z} \pmod{gR}$ and which has small coefficients if and only if \mathbf{z} is the reduction of a small element in R , namely of \mathbf{w} itself.

In some cases, this will be overkill though. For example, if f and g can be written as polynomials in some power of X , say X^p with $f(X) = F(X^p)$ and $g(X) = G(X^p)$, then we can split the problem into p cases since for a small element $\mathbf{w} \in R$ each coefficient of $\text{rep}_g(\mathbf{w})$ depends only on n/p coefficients of \mathbf{w} and each such coefficient only affects one coefficient of $\text{rep}_g(\mathbf{w})$. In this case one reduces the problem to determining smallness in the ring $\mathbb{Z}[Y]/(F(Y), G(Y))$. For more general ι than the coefficient embedding, one may also have a similar decomposition and in this case the splitting needs to respect the notion of smallness as well.

In summary, some choices of f and g have efficient methods to determine whether an element of R_g is the reduction of a small element in R while for others this is not true. It is unclear whether instances of R_g not having such an efficient test for smallness can be effectively used in cryptographic applications.

2.6 What happens when a is not prime?

In many cases, such as when g itself is an integer modulus and we consider the Ideal-LWE problem, there are no fine-grained restraints put on the possible values for a which can be used securely. On the other hand, in the MLHR and MLHC problems [2], the authors restrict to the case when a is a prime. Also, the divisibility of a should be considered in the Ideal-NTRU setting from a practical point of view as, if it is highly composite with many small primes as factors, it will be difficult to find invertible matrices \mathbf{u} to use in the problem. All of this begs the question of what happens when a is not a prime.

The obvious fact in this case is that the ring R_g can be written as a product of smaller rings using the Chinese remainder theorem. If a is written as a product

of coprime integers (e.g. powers of distinct primes) as $a = q_1 \cdots q_t$ then we have

$$R_g \cong \frac{\mathbb{Z}_{q_1}[X]}{(r(X))} \times \cdots \times \frac{\mathbb{Z}_{q_t}[X]}{(r(X))},$$

together with the t natural homomorphisms $\theta_i: R_g \rightarrow \mathbb{Z}_{q_i}[X]/(r(X))$. This leaves the potential for the problem to be split into t smaller problems which can be solved independently and whose solutions can be combined to solve the original problem.

For lattice attacks, the reduction in the size of the integer modulus has little effect on the algorithms as the important parameter is the dimension of the lattice. However, for combinatorial attacks it is possible that reducing a can have a positive effect in reducing the complexity of the algorithms. We discuss this further when explaining these attacks.

The problem with the homomorphisms θ_i is that they do not reduce the dimension n of the ring R as a \mathbb{Z} -module. There are however circumstances where lowering the dimension is possible; to see this we again use the MLHR problem. Here, the main reason a is taken to be prime is actually not because a itself is prime but rather that such an a forces the dimension n to be prime. If n were composite, say $n_0 \mid n$, then $X^{n_0} - 1$ divides $X^n - 1$ and we have $(X^n - 1, X - 2) \subseteq (X^{n_0} - 1, X - 2)$ which is the same as the statement that $(2^{n_0} - 1) \mid (2^n - 1)$. In this way the problem can be considered in a smaller dimension which in the case of the MLHR problem and for suitable n_0 is likely to still be an instance of the same problem. The issue arises here because f is not irreducible rather than a not being prime.

2.7 What happens when f is not irreducible?

If there exists $f' \mid f$ then there is a map $R \rightarrow R' = \mathbb{Z}[X]/(f'(X))$ which maps the distribution χ to a distribution χ' and if χ' can effectively be seen as a distribution of small elements it may be possible to mount an attack on this smaller dimensional problem to recover partial information about the solution of the original problem. Actually, one need only have f' dividing f modulo a as exemplified by the ‘evaluation at one’ attack in [36]. The main restriction however is that the distribution of small elements is very likely to become indistinguishable from random in most cases hence f being reducible doesn’t immediately imply the existence of an attack.

2.8 A Generic Transformation to Normal Form for Ideal-LWE

In [12, Section 3.1], Applebaum et al. give a generic transformation from the LWE problem with modulus a prime power to one in which the secret vector is sampled from the error distribution at the cost of reducing the number of samples. In our setting we can apply the same technique.

Suppose we are given an instance $(\mathbf{a}, \mathbf{b}) \in R_g^{\ell \times m} \times R_g^{\ell \times k}$ for some $\ell > m$. Suppose further that there is a submatrix A of \mathbf{a} , consisting of m rows of \mathbf{a} , such

that A is invertible over R_g . Write B for the corresponding submatrix of \mathbf{b} , that is taking the same rows we did to give A . Write $(\bar{\mathbf{a}}, \bar{\mathbf{b}})$ for the remaining rows not in (A, B) . Define $\mathbf{a}' := -\bar{\mathbf{a}}A^{-1}$ and $\mathbf{b}' = \bar{\mathbf{b}} + \mathbf{a}'B$.

If (\mathbf{a}, \mathbf{b}) consists of samples from the Ideal-LWE distribution, then $(\mathbf{a}', \mathbf{b}') \in R_g^{(\ell-m) \times m} \times R_g^{(\ell-m) \times k}$ are samples from the Ideal-LWE distribution whose secret is sampled from the error distribution. First, to see that \mathbf{a}' is uniformly random we note that the map $R_g^m \rightarrow R_g^m, \alpha \mapsto -\alpha A^{-1}$ which we are applying to the uniformly random rows of $\bar{\mathbf{a}}$ (i.e. the first component of the remaining samples) is an isomorphism. To see that \mathbf{b}' is of the required form write $E = B - As$ for the error in the samples used to construct (A, B) , here \mathbf{s} is the secret associated with the original samples. Then we have that the secret for the samples $(\mathbf{a}', \mathbf{b}')$ is E as we have

$$\mathbf{b}' - \mathbf{a}'E = \bar{\mathbf{b}} + \mathbf{a}'B - \mathbf{a}'(B - As) = \bar{\mathbf{b}} - \bar{\mathbf{a}}\mathbf{s} = \bar{\mathbf{e}}.$$

We also see that the error in the remaining samples does not change. Alternatively, if \mathbf{b} were uniformly random then so too is $\bar{\mathbf{b}}$ and hence also \mathbf{b}' .

We therefore see that the transformation works only when such an invertible matrix A can be found and the probability of this depends on ℓ but also on the value of a . If a is divisible by many small primes then the proportion of invertible elements in R_g decreases and the less likely one will be able to invoke the transformation.

2.9 Modulus switching for the short secret Ideal-LWE problem

In the case of an integer ciphertext modulus, one can use the technique of modulus switching. This is a technique which allows one to transform elements in the ring R_{q_1} , with integer ciphertext modulus q_1 to elements in a new ring R_{q_2} , with q_2 another integer while preserving the relative size of the elements. One can apply this transformation to the learning with errors family of problems as was done in the BGV levelled fully homomorphic encryption scheme [25] to reduce noise growth.

When replacing the integer modulus with a general polynomial ciphertext modulus one can do the same. Suppose we have an element $\mathbf{y} \in R_{g_1}$ and we want to convert it to an element $\mathbf{y}' \in R_{g_2}$, we can do so using Algorithm 1.

One might wonder if the output depends on the choice of lift in line 2 however since $g_1(X)\beta(X)g_2(X) \equiv g_2(X) \pmod{f(X)}$ this is not the case.

In the short secret variant of the Ideal-LWE problem we can use modulus switching to transform an instance of the problem over R_{g_1} to one over R_{g_2} as follows. Suppose we are given samples $(\mathbf{a}, \mathbf{b}) \in R_{g_1}^{\ell \times m} \times R_{g_1}^{\ell \times k}$ where $\mathbf{b} \equiv \mathbf{a}\mathbf{s} + \mathbf{e} \pmod{g_1R}$ for some small secret $\mathbf{s} \in R^{m \times k}$ and error $\mathbf{e} \in R^{\ell \times k}$. We can apply Algorithm 1 component-wise to both \mathbf{a} and \mathbf{b} to give $\mathbf{a}' \in R_{g_2}^{\ell \times m}$ and $\mathbf{b}' \in R_{g_2}^{\ell \times k}$ respectively. Then we have $\mathbf{b}' \equiv \mathbf{a}'\mathbf{s} + \mathbf{e}' \pmod{g_2R}$ for some element $\mathbf{e}' \in R^{\ell \times k}$, the size of which depends on f, g_1, g_2, \mathbf{s} , in particular on $\|\mathbf{s}\|_2$, and the original size of \mathbf{e} . If we write \mathbf{e}' for the result of applying ModulusSwitch to

Algorithm 1: ModulusSwitch

Input : An element $\mathbf{y} \in R_{g_1}$ with $R = \mathbb{Z}[X]/(f(X))$ and $n = \deg f$
Output: An element $\mathbf{y}' \in R_{g_2}$
1 $1, \beta(X), \gamma(X) \leftarrow \text{XGCD}_{\mathbb{Q}[X]}(g_1(X), f(X));$
2 $y(X) \leftarrow \text{rep}_{g_1}(\mathbf{y});$
3 $p(X) \leftarrow y(X)\beta(X)g_2(X) \bmod f(X);$
4 Write $p(X) = p_{n-1}X^{n-1} + p_{n-2}X^{n-2} + \dots + p_1X + p_0;$
5 $y'(X) \leftarrow \lfloor p_{n-1} \rfloor X^{n-1} + \lfloor p_{n-2} \rfloor X^{n-2} + \dots + \lfloor p_1 \rfloor X + \lfloor p_0 \rfloor;$
6 $\mathbf{y}' \leftarrow y'(X) \bmod g_2(X);$

$\mathbf{e} \bmod g_1 R$ and lifting it to an element of $R^{\ell \times k}$ then we can write $\mathbf{e}^* = \mathbf{e}' + \mathbf{e}_s + \mathbf{e}_+$ where

$$\mathbf{e}_s[i, j] := \sum_{t=1}^m (\lfloor a_{i,t} \mathbf{s}_{t,j} \beta g_2 \bmod f \rfloor - \lfloor a_{i,t} \beta g_2 \bmod f \rfloor \mathbf{s}_{t,j})$$

$$\mathbf{e}_+[i, j] := \lfloor b_{i,j} \beta g_2 \bmod f \rfloor - \sum_{t=1}^m \lfloor a_{i,t} \mathbf{s}_{t,j} \beta g_2 \bmod f \rfloor - \lfloor \mathbf{e}_{i,j} \beta g_2 \bmod f \rfloor$$

and $a_{i,j}, b_{i,j}$ are lifts of $\mathbf{a}_{i,j}$ and $\mathbf{b}_{i,j}$ to R . We note that $\|\iota(\mathbf{e}_+)\|_\infty \leq \frac{m+2}{2}$ while the size of \mathbf{e}_s depends on the size of \mathbf{s} .

As a concrete example, for $f = X^n + 1$, $g_1 = X^{n_1} - q_1$ or $g_1 = q_1$ for some integer q_1 and similarly $g_2 = X^{n_2} - q_2$ or $g_2 = q_2$ for some integer q_2 (any combination is possible), we found experimentally that if σ_1^2 is the variance of the original error distribution, the variance of the new error distribution is $\sigma_2^2 \approx \sigma_1^2 (q_2/q_1)^2 + 0.085 \|\mathbf{s}\|_2^2$. This implies we can only apply modulus switching if the secret has very small 2-norm such as having a sparse binary vector of coefficients. In particular, this is typically not true when applying the generic transformation from the previous section as the new secret is sampled from the error distribution which usually cannot be too small due to Arora-Ge style attacks (see Section 5.7). In general, the relationship between all the parameters which determine how large \mathbf{e}^* is appears difficult to write down.

In particular, this gives us some confidence in the hardness of the Ideal-LWE problem with general polynomial modulus g . Suppose we could solve this problem for a given modulus g_2 , this would give us an algorithm to solve the short secret problem for a suitable *integer* modulus g_1 and narrow enough error distribution. Namely, we could use modulus switching to transform such samples modulo g_1 to samples modulo g_2 at the cost of increasing the size of the errors. After this, one could apply the attack which works modulo g_2 , allowing one to find the solution to the problem modulo g_1 . Of course, if the error distribution is too narrow, we already have attacks on the integer modulus Ideal-LWE problem such as the Arora-Ge attack and in that case the reduction may be meaningless.

3 Attacks on the SIS family of problems

3.1 Simple lattice attack

One can view the original SIS problem, $\text{Ideal-SIS}_{X,q,m,\ell,\rho}$, as an approximate shortest vector problem on the ℓ -dimensional q -ary lattice

$$\Lambda_q^\perp(A) := \{ \mathbf{z} \in \mathbb{Z}^\ell \mid A\mathbf{z}^T \equiv \mathbf{0}^T \pmod{q} \}.$$

where $A \in \mathbb{Z}_q^{m \times \ell}$ is the uniformly random matrix defining the problem. We note that with high probability the matrix A is full rank and further that the lattice $\Lambda_q^\perp(A)$ always has rank m and with high probability has a basis of the form

$$\begin{pmatrix} I_{\ell-m} & C \\ 0 & qI_m \end{pmatrix},$$

for some $(\ell - m) \times m$ integer matrix C , and thus volume q^m . Since we are looking for a non-zero vector of length at most ρ the problem is equivalent to the Hermite shortest vector problem HSVP_γ with approximation factor $\gamma = \rho q^{-m/\ell}$.

If we perform lattice reduction on the lattice $\Lambda_q^\perp(A)$ in an attempt to find a short enough non-zero vector then we need to use an algorithm that can achieve a root Hermite factor of at most $\delta_0 = \rho^{1/\ell} q^{-m/\ell^2}$.

Typically, the dimension ℓ of the lattice will be large so that running such a lattice reduction algorithm will be very costly. One can try to get around this by removing columns from the matrix A which lowers the dimension of the lattice however this may reduce the number of possible solutions to zero.

When considering the ring and module variant $\text{Ideal-SIS}_{f,q,m,\ell,\rho}$ of the problem the approach remains the same after rewriting the product $A\mathbf{z}^T$ as the product between a matrix $\tilde{A} \in \mathbb{Z}_q^{mn \times \ell n}$, depending on A and f , and $\iota(\mathbf{z}) \in \mathbb{Z}^{\ell n}$. Now one can construct the lattice $\Lambda_q^\perp(\tilde{A})$ and again apply a lattice reduction algorithm to it. More detail on exactly how to construct \tilde{A} is given below when we generalise to using a polynomial g in place of q .

Generalisation

For the more general problem $\text{Ideal-SIS}_{f,g,m,\ell,\rho}$, the SIS lattice becomes

$$\Lambda(\mathbf{a}_1, \dots, \mathbf{a}_\ell) = \left\{ (\iota(\mathbf{z}_1), \dots, \iota(\mathbf{z}_\ell)) \in \mathbb{Z}^{\ell n} \mid \mathbf{z}_i \in R, \sum_{i=1}^{\ell} \mathbf{a}_i \mathbf{z}_i \equiv \mathbf{0} \pmod{gR} \right\}.$$

To compute a basis of this lattice we can use standard methods for computing the left kernel of the $(mn + \ell \deg r) \times \ell \deg r$ matrix

$$\begin{pmatrix} M_{\mathbf{a}_{1,1}} & M_{\mathbf{a}_{2,1}} & \cdots & M_{\mathbf{a}_{\ell,1}} \\ M_{\mathbf{a}_{1,2}} & M_{\mathbf{a}_{2,2}} & \cdots & M_{\mathbf{a}_{\ell,2}} \\ \vdots & \vdots & \ddots & \vdots \\ M_{\mathbf{a}_{1,m}} & M_{\mathbf{a}_{2,m}} & \cdots & M_{\mathbf{a}_{\ell,m}} \\ aI_{\deg r} & & & \\ & aI_{\deg r} & & \\ & & \ddots & \\ & & & aI_{\deg r} \end{pmatrix},$$

where $M_{\mathbf{a}_{i,j}} \in \mathbb{Z}^{n \times \deg r}$ is the matrix of multiplication by $\mathbf{a}_{i,j}$ (the j th component of \mathbf{a}_i) taking an element from R , represented by its coefficient (row) vector, to an element of $\text{Rep}(R_g)$ represented by its coefficient vector up to multiples of a . Finding a basis for this kernel can be done by, for example, using the LLL algorithm. It is then a matter of running a strong enough lattice reduction algorithm on $\Lambda(\mathbf{a}_1, \dots, \mathbf{a}_\ell)$ in order to find a small enough non-zero vector in this lattice.

When g is a polynomial with small coefficients we may have many vectors in this lattice which are small but give rise to the trivial solution modulo gR . However, these are valid solutions to the Ideal-SIS problem and in this case the problem becomes trivial if the norm of the vector of coefficients of g is smaller than ρ .

If we allow only solutions which are non-zero modulo gR , as will be needed in Section 5.5 when solving Ideal-LWE via Ideal-SIS, then the existence of these trivial short vectors does not prevent us from finding small non-trivial vectors. We found that running the BKZ algorithm still enables one to recover non-trivial vectors of length roughly $\delta_0^{\ell n} \text{Vol}(\Lambda(\mathbf{a}_1, \dots, \mathbf{a}_\ell))^{1/\ell n}$ in the lattice and hence solve Ideal-SIS for values of ρ larger than this.

Requirements. *There are essentially no requirements for performing this type of attack besides the existence of a solution to the problem and the use of a strong enough lattice reduction algorithm. In particular, if the lattice reduction algorithm achieves a root-Hermite factor δ_0 , then the attack is likely to succeed if $\delta_0^{\ell n} \text{Vol}(\Lambda(\mathbf{a}_1, \dots, \mathbf{a}_\ell))^{1/\ell n} < \rho$.*

3.2 A meet-in-the-middle attack

The most naïve attack one can consider on the Ideal-SIS $_{X,q,m,\ell,\rho}$ problem is to perform a brute force attack by enumerating over all possible non-zero vectors $\mathbf{z} \in \mathbb{Z}^\ell$ such that $\|\mathbf{z}\| \leq \rho$ and testing if $A\mathbf{z}^T \equiv \mathbf{0}^T \pmod{q}$. Clearly, the running time of such an approach is exponential in ℓ .

One can improve the running time at the expense of using a larger amount of memory by using a meet-in-the-middle approach. Namely, as \mathbf{a}_i^T is the i th

column of the matrix A , we can rewrite the congruence as

$$\sum_{i=1}^k \mathbf{a}_i^T \cdot z_i \equiv - \sum_{i=k+1}^{\ell} \mathbf{a}_i^T \cdot z_i \pmod{q},$$

for some $1 \leq k \leq \ell$; typically $k = \ell/2$. After computing and storing the result of the left-hand side of the congruence for all possible choices of z_1, \dots, z_k that could lead to a solution, one can then enumerate over the possible remaining choices for z_{k+1}, \dots, z_{ℓ} , compute the right-hand side of the congruence, and search for a collision with the stored values. If a collision is found for which the corresponding \mathbf{z} satisfies the norm bound then one has found a solution. It is straightforward to generalise this to the inhomogeneous version of the problem.

The straightforward implementation of the meet-in-the-middle attack described above gives a time-memory trade-off which follows the curve $TS = \tilde{O}(\#\{\mathbf{z} \in \mathbb{Z}^{\ell} \mid \|\mathbf{z}\| \leq \rho\})$, where T is the time and S is the space used. Schroepel and Shamir [68] improve on this basic approach in the context of the (modular) subset-sum/knapsack problem ($m = 1$ and \mathbf{z} a binary vector), significantly reducing the memory requirements so that the left-hand side becomes TS^2 instead of TS . A simpler description of the Schroepel–Shamir algorithm was given by Howgrave-Graham and Joux [48]. The idea is that one does not need to compute the two lists in full to find a collision but instead one can compute them on the fly using priority queues. This approach uses four lists rather than two.

Generalisation

It is self-evident that this attack can be generalised to work against the Ideal-SIS $_{f,g,m,\ell,\rho}$ due to elements in R_g having a canonical representative, hence allowing us to efficiently find collisions.

3.3 Combinatorial attacks

Another strategy is a divide and conquer approach in which one solves smaller problems and combines the solutions to give a solution to the original problem. The smaller problems will, in general, be variants of the inhomogeneous problem, but the approach to solving them is much the same as with the homogeneous case. As explained in [17], these smaller problems will have a smaller solution space and a higher density, that is a higher expected number of solutions. Formally, the density of the inhomogeneous SIS problem is defined as

$$\delta = \frac{\#\{\mathbf{z} \in \mathbb{Z}^{\ell} \mid \|\mathbf{z}\| \leq \rho\}}{\#\mathbb{Z}_q^m} \approx \left(\frac{2\pi e}{\ell}\right)^{\ell/2} \frac{\rho^{\ell}}{\sqrt{\ell\pi}q^m},$$

where the approximation is for large ℓ using Stirling’s formula. For combinatorial attacks, it is more natural to consider the problem in which, rather than requiring

$\|\mathbf{z}\| \leq \rho$, we instead require the coordinates z_i to lie in some subset $\mathcal{Z} \subset \mathbb{Z}$ which we denote by (I)SIS $_{m,\ell,q,\mathcal{Z}}$. In this case, the density of the problem is defined to be $(\#\mathcal{Z})^\ell q^{-m}$. Typically, $\mathcal{Z} = \{0, 1\}$ is considered for combinatorial attacks, though this is not strictly necessary.

If the density of the problem is much less than 1 then the problem is said to have low density while if it is much larger than 1 it is a high density problem. If the density is close to one the problem is said to have “density 1”. Differing attacks apply to problems with different densities.

The attack of Camion, Patarin and Wagner

When the problem has very high density, one can use the attack first described by Camion and Patarin [26] for the subset-sum problem and generalised by Wagner [72]. Here, we present the analysis given in [62] to solve the SIS $_{m,\ell,q,\mathcal{Z}}$ problem. First, one splits the ℓ coordinates of \mathbf{z} into 2^k groups of roughly equal size. For each group, compute the list of all possible values $A\mathbf{z}^T \bmod q$ for \mathbf{z} having entries from \mathcal{Z} in the coordinates in the group and zero otherwise. Each list contains roughly $d := (\#\mathcal{Z})^{\ell/2^k}$ elements in \mathbb{Z}_q^m . Next, one combines the lists in pairs by finding all pairs, \mathbf{x} in the first list and \mathbf{y} in the second list, for which the first $\log_q d$ coordinates of $\mathbf{x} + \mathbf{y}$ are zero modulo q . The expected length of this new list is again approximately d . Now, one will have 2^{k-1} lists containing vectors that are zero in the first $\log_q d$ coordinates. Repeat the previous process on each consecutive set of $\log_q d$ coordinates until one has a single list of size roughly d and whose elements are zero in the first $k \log_q d$ coordinates. Search this final list for a solution in which all coordinates are zero. The parameter k is chosen such that $m \approx (k + 1) \log_q d$, which is equivalent to

$$\frac{2^k}{k + 1} \approx \frac{\ell \log(\#\mathcal{Z})}{m \log q}.$$

We note that, as opposed to lattice reduction techniques, in this case having a larger ℓ is beneficial as a larger k can be chosen which means the lists are shorter and the attack is more efficient.

We remark that Minder and Sinclair give some refinements on the above attack which speed it up slightly [63].

The attack of Howgrave-Graham and Joux and the improvement of Becker, Coron and Joux

In the case of a relatively sparse solution \mathbf{z} (typically with $\mathcal{Z} = \{0, 1\}$) one can attempt to split the solution by the weight (number of non-zero coordinates) of \mathbf{z} . In the context of the subset-sum problem, Howgrave-Graham and Joux [48] proposed this method of splitting up \mathbf{z} with the idea to reduce the problem to two smaller problems, solving each recursively, and combining the solutions to give a solution to the original problem.

Suppose one is looking for a solution \mathbf{z} of known weight ω to the $\text{SIS}_{m,\ell,q,\mathcal{Z}}$ problem defined by A . Concretely, the idea is to choose a subgroup $H \leq \mathbb{Z}_q^m$ and a random $\mathbf{r} \in \mathbb{Z}_q^m/H$ and split the problem into the two problems $A\mathbf{z}_1^T \equiv \mathbf{r}^T \pmod{H}$ and $A\mathbf{z}_2^T \equiv -\mathbf{r}^T \pmod{H}$ where \mathbf{z}_1 and \mathbf{z}_2 have weight $\omega/2$ (and entries in \mathcal{Z}). The hope is that there is a pair of solutions such that $\mathbf{z}_1 + \mathbf{z}_2$ is a solution to the original problem. The subgroup H is chosen to trade-off the probability that the random choice of \mathbf{r} leads to a valid splitting of \mathbf{z} . In the setting considered by Howgrave-Graham and Joux the subgroup H is equivalent to reducing modulo some modulus, that is H is isomorphic to $\mathbb{Z}_{q'}^m$ for some $q' \mid q$ but in our case there is no guarantee a q' of a suitable size exists. In this case one can instead consider H which correspond to the congruence being satisfied in a certain set of coordinates only, that is H is isomorphic to $\mathbb{Z}_q^{m'}$ for some $m' < m$.

As mentioned, one problem with this approach is one should know the weight of a solution; if this is not the case one can guess the weight is in some range $[\omega - 2\epsilon, \omega + 2\epsilon]$ and then the solutions of the smaller problems should have weight in $[\omega/2 - \epsilon, \omega/2 + \epsilon]$.

The original attack of Howgrave-Graham and Joux was improved upon by Becker, Coron and Joux [18] by allowing a larger coefficient set for the smaller problems allowing for some cancellation to occur in the sum $\mathbf{z}_1 + \mathbf{z}_2$; thus the smaller problems now have a larger density and better parameter choices can be used.

Reducing to LWE: the attack of Bai et al.

As well as analysing the combinatorial attacks above with an eye towards the inhomogeneous SIS problem, Bai et al. [17] also introduced another combinatorial attack on this problem by reducing it to the Ideal-LWE $_{X,q,\ell-m,1,m,\chi}$ problem in which the secret is short and where χ is some unknown distribution over the set \mathcal{Z} . The idea is to write the matrix A defining the problem in Hermite normal form. Assuming A has rank m so that there exists an invertible $m \times m$ submatrix of A (which we may assume by reordering the columns consists of the first m columns), there exists a matrix U such that $UA = (I_m \ A')$. Then, in the inhomogeneous case, $A\mathbf{z}^T \equiv \mathbf{t}^T \pmod{q}$ if and only if $U\mathbf{t}^T \equiv A'\mathbf{z}_2^T + \mathbf{z}_1^T \pmod{q}$ where $\mathbf{z} = (\mathbf{z}_1 \ \mathbf{z}_2)$.

Since there are only m such LWE samples, many of the attacks discussed in Section 5 below cannot be applied to this case. Instead, the authors propose to apply the same combinatorial algorithms described above to the inhomogeneous SIS problem defined by A' but adapted to look for approximate collisions due to the presence of the additional \mathbf{z}_1^T term.

In more detail, for the Camion, Patarin and Wagner approach, one computes the initial lists as before but when combining two lists one only requires the $\log_q d$ coordinates under consideration to be in \mathcal{Z} and hence only approximately zero modulo q . Suppose one is considering \mathbf{x} from one list and \mathbf{y} from the other with $\mathbf{x}^T \equiv A'\mathbf{z}_1^T \pmod{q}$ and $\mathbf{y}^T \equiv A'\mathbf{z}_2^T \pmod{q}$ where the first $j \log_q d$ coordinates of both \mathbf{x} and \mathbf{y} lie in \mathcal{Z} for some $j \geq 0$. One wants to check whether the coordinates

from $j \log_q d + 1$ to $(j + 1) \log_q d$ of $\mathbf{x} + \mathbf{y}$ lie in \mathcal{Z} ; however a problem occurs in that in the preceding coordinates one is no longer summing zeros but small elements which may grow to no longer be in \mathcal{Z} . To get around this, Bai et al. propose that one is only allowed to add $\mathbf{x} + \mathbf{y}$ to the new list if the non-zero elements in the first $j \log_q d$ coordinates of \mathbf{x} and \mathbf{y} occur in differing coordinates so that no growth occurs.

When adapting the Howgrave-Graham and Joux approach of splitting the solution by weight, Bai et al. split both \mathbf{z}_1 and \mathbf{z}_2 by weight and when combining solutions the non-zero coordinates in both parts should not overlap. As before, this can be extended to allow for a slightly larger allowed set of solutions for the subproblems to increase the density and allow some cancellation to occur when combining solutions but now separately in both \mathbf{z}_1 and \mathbf{z}_2 .

Generalisation

Bai et al. [17] offer a high-level framework encompassing the attacks described in this subsection. The approach is to consider what they call the (G, m, \mathcal{B}) -ISIS problem, where G is an Abelian group, m is a natural number and $\mathcal{B} \subset \mathbb{Z}$ is a small subset of the integers containing zero. The problem is defined by a pair $(\mathbf{A}, \mathbf{s}) \in G^m \times G$ and one must find an $\mathbf{x} \in \mathcal{B}^m$ such that $\mathbf{A}\mathbf{x} = \mathbf{s}$. In the words of the authors, all these combinatorial algorithms are obtained by combining two basic operations (possibly recursively):

1. Compute lists of small solutions to some constrained problem obtained by “splitting” the solution space (i.e., having a smaller set of possible \mathbf{x}) in a quotient group G/H . Splitting the solution space lowers the density (expected number of solutions), but working in the quotient group G/H compensates by raising the density again.
2. Merge two lists of solutions to give a new list of solutions in a larger quotient group G/H' .

Our Ideal-SIS problem almost fits within the framework of Bai et al., the obvious approach is to write it as the $(R_g^m, \ell, \{\mathbf{z} \in \mathbf{R} \mid \|\mathbf{z}\| \leq \rho\})$ -ISIS problem (with $\mathbf{s} = \mathbf{0}$), but our \mathcal{B} is not a subset of the integers. When g is an integer this problem can be rectified by forgetting the ring structure and considering it as the $(\mathbb{Z}_g^{mn}, ln, \mathcal{B}')$ -ISIS problem for some suitable $\mathcal{B}' \subset \mathbb{Z}$ depending on the bound on the infinity norm of a valid solution. The case of polynomial g may look trickier but actually there isn't an issue here when one notes that the approach of these algorithms is to first solve the problem in quotient groups G/H . We can simply choose $G = \mathbb{Z}_a^{mn}$ and as the final quotient group use $H = H_0^m$ where $H_0 \subseteq \mathbb{Z}_a^n$ is the group generated by $\{\iota(gX^i) : 0 \leq i < n\}$ when taken modulo a . Although such subgroups H were not explicitly used in [17], this choice of H is the natural generalisation of the subgroup $H = p\mathbb{Z}_q^n$ for $p \mid q$ for integers p and q .

With this style of attack it is potentially advantageous that a has many divisors as this provides many options for the choice of subgroup H . However it is more practical to consider subgroups which also decrease the dimension of the lattice associated with the quotient G/H .

For problems which have a more elaborate ring structure than \mathbb{Z} , Bai et al. propose to use the “symmetries” of the ring to speed up the attacks. These symmetries only appear in very special rings and those suggested do not apply when g is a non-constant polynomial as they are not fixed by the symmetry.

Requirements. *Such combinatorial attacks rely on the existence of suitable subgroups H of R_g^m . When m is large such subgroups always exist however for small m they may not, for example the MHLIC problem has $R_g^m = \mathbb{Z}_M$ for a Mersenne prime M which has no non-trivial subgroups. Furthermore, such an attack is aided when the solution space can be nicely partitioned using disjoint linear subspaces.*

4 Attacks on the NTRU family of problems

4.1 The standard NTRU lattice attack

Right from its very inception, attacks utilizing lattice reduction were considered against the NTRU problem. In the first draft of the NTRU scheme, circulated at the CRYPTO '96 rump session, a simple lattice attack was already briefly mentioned [45]; however, the lattice attack was analysed in detail by Coppersmith and Shamir [31]. We remark that these attacks are against a slightly different problem than the one we defined due to [45] including an extra factor p for an integer p coprime to q .

To begin, one notes that in the Ideal-NTRU $_{f,q,1,\chi,\rho}$ problem we are searching for a $\mathbf{u}' \in R = \mathbb{Z}[X]/(f(X))$ such that the scalar-vector product $\mathbf{u}'(1, \mathbf{h}) \bmod q$ consists of two elements with small coefficients. By making the relationship between the coefficients explicit we can formulate this problem in terms of a lattice problem.

Let us define h_i to be the coset representative of $\mathbf{h}X^i$ of degree at most $n - 1$, where as usual $n = \deg f$, and write $h_i = \sum_{j=0}^{n-1} h_{i,j}X^j$. Similarly, write $\mathbf{u}' = \sum_{j=0}^{n-1} u'_jX^j$ then one requires

$$(u'_0 \ u'_1 \ \cdots \ u'_{n-1}) \begin{pmatrix} 1 & 0 & \cdots & 0 & | & h_{0,0} & h_{0,1} & \cdots & h_{0,n-1} \\ 0 & 1 & \cdots & 0 & | & h_{1,0} & h_{1,1} & \cdots & h_{1,n-1} \\ \vdots & \vdots & \ddots & \vdots & | & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 & | & h_{n-1,0} & h_{n-1,1} & \cdots & h_{n-1,n-1} \end{pmatrix},$$

when reduced modulo q into the symmetric interval about zero, to have small components and thus be a short vector. Equivalently, we must find a short vector

in the lattice generated by the rows of the matrix

$$\begin{pmatrix} 1 & 0 & \cdots & 0 & h_{0,0} & h_{0,1} & \cdots & h_{0,n-1} \\ 0 & 1 & \cdots & 0 & h_{1,0} & h_{1,1} & \cdots & h_{1,n-1} \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 & h_{n-1,0} & h_{n-1,1} & \cdots & h_{n-1,n-1} \\ 0 & 0 & \cdots & 0 & q & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 & 0 & q & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 & 0 & 0 & \cdots & q \end{pmatrix}.$$

We call this lattice the standard NTRU lattice and denote it by $A_{\text{NTRU}}(\mathbf{h})$.

Clearly, with the obvious notation for the coefficients of \mathbf{u} and \mathbf{v} , the lattice contains the vector

$$(u_0 \ u_1 \ \cdots \ u_{n-1} \ v_0 \ v_1 \ \cdots \ v_{n-1})$$

which is short.

It is potentially profitable to multiply the first n columns of this matrix by a real scalar λ to balance the size of the coefficients of $\lambda\mathbf{u}$ with those of \mathbf{v} . In addition, if the coefficients of \mathbf{u} are distributed with a non-zero mean $\mu_{\mathbf{u}}$, such as in the binary coefficient case, and similarly for $\mu_{\mathbf{v}}$, then one can instead consider the closest vector problem in this lattice with target vector $(\mu_{\mathbf{u}} \ \cdots \ \mu_{\mathbf{u}} \ \mu_{\mathbf{v}} \ \cdots \ \mu_{\mathbf{v}})$.

When considering the shortest vector problem, the lattice has volume $(\lambda q)^n$ and the Gaussian heuristic states we expect the shortest non-zero vector to have length roughly at most $\sqrt{\lambda q n / \pi e}$. If $\sqrt{\lambda^2 \|\mathbf{u}\|^2 + \|\mathbf{v}\|^2}$ is sufficiently smaller than this value then performing lattice reduction on this lattice with a strong enough algorithm can recover a suitable \mathbf{u}' .

As an example, the parameters $f(X) = (X^{509} - 1)/(X - 1)$ and $q = 2048$ are specified in the specification document of the NTRUEncrypt submission to NIST's post-quantum cryptographic standardization process as suitable parameters for a category 1 public-key encryption scheme. Further, \mathbf{u} is a uniformly random ternary polynomial and \mathbf{v} is a ternary polynomial having $q/16 - 1$ coefficients equal to 1 and the same number equal to -1 . Thus $\|\mathbf{v}\|^2 = 254$ and for simplicity let us assume $\|\mathbf{u}\|^2 = 339$, then we take $\lambda = \sqrt{254/339}$. If the standard NTRU lattice were to behave like a random $2n$ -dimensional lattice with volume $(\lambda q)^n$ then we would expect a shortest non-zero vector of length roughly 324.7. On the other hand, we actually know that the lattice contains a vector of length roughly $\sqrt{2 \cdot 254} \approx 22.5$ which is much smaller than predicted by the Gaussian heuristic.

While the lattice contains a very short lattice vector, it turns out that in practice, for large enough parameters, lattice reduction algorithms which are strong enough to recover such a short vector are still prohibitively expensive in practice so such attacks cannot be applied directly to cryptosystems utilising the NTRU problem when instantiated properly.

Generalisation

Suppose we have an element $\mathbf{h} = \mathbf{v}\mathbf{u}^{-1} \in R_g^{m \times m}$ sampled from the Ideal-NTRU distribution. We can apply the standard lattice attack on this problem by considering the $2mn$ -dimensional lattice

$$\Lambda(\mathbf{h}) = \{(\iota(\mathbf{x}), \iota(\mathbf{y})) \in \mathbb{Z}^{2mn} \mid \mathbf{x}, \mathbf{y} \in R^m \text{ and } \mathbf{h}\mathbf{x}^T \equiv \mathbf{y}^T \pmod{gR^m}\},$$

which clearly contains the vectors $(\iota(\mathbf{u}_i), \iota(\mathbf{v}_i))$ where \mathbf{u}_i^T is the i th column of the secret matrix \mathbf{u} and similarly for the \mathbf{v}_i^T . This lattice therefore contains at least m linearly independent short vectors.

We can easily construct a spanning set for $\Lambda(\mathbf{h})$ by letting \mathbf{x} run over a \mathbb{Z} -basis of R^m (e.g. a copy of the power-basis of R in each of the m components) and taking $\mathbf{y}^T = \mathbf{h}\mathbf{x}^T$. Further, we must add the vectors $(\mathbf{0}, g\mathbf{z})$ as \mathbf{z} runs over a basis for R^m . The lattice $\Lambda(\mathbf{h})$ has volume $|R_g^m| = a^{m \deg r}$ with high probability.⁴ Thus, the Gaussian heuristic implies that the expected length of the shortest vector in $\Lambda(\mathbf{h})$ is about $\sqrt{mn/\pi e a^{\deg(r)/2n}}$ while if the error distribution samples elements with independent coefficients from a centred distribution having standard deviation σ then we expect to find at least m linearly independent vectors having norm roughly $\sqrt{2mn}\sigma$. If we are required to find \mathbf{u}' and \mathbf{v}' with norms less than ρ as in the definition of the Ideal-NTRU problem, then we only need to find lattice vectors of length approximately $\sqrt{2m}\rho$ and we see that if $\rho \ll \sqrt{n/2\pi e a^{\deg(r)/2n}}$ then we must find unusually short vectors in the lattice. By assumption, these vectors exist and if there aren't shorter non-zero vectors in the lattice it is simply the task of running a good enough lattice reduction algorithm in order to find them. In particular, we will need m such lattice vectors for which the first half of the coordinates correspond to vectors in R^m which are linearly independent modulo gR^m so that we can recover an invertible \mathbf{u}' .

As with the lattice attack against Ideal-SIS, our lattice contains trivial vectors when considered modulo gR and these trivial vectors can be very short as in the MLHR problem for which $g = X - 2$ itself gives rise to vectors of norm $\sqrt{5}$. Even if there are short trivial vectors we can still try to recover short non-trivial vectors from the reduced basis for the lattice which could allow one to obtain a small enough solution. However, experimentally we found the size of any solution we could recover using strong lattice reduction was too large when gR contains polynomials with very short coefficient vectors such as the case of MLHR.

To summarise, the following are necessary conditions to be able to mount a successful attack; however, they may not be sufficient on their own:

Requirements. *For this attack to work on the Ideal-NTRU_{f,g,m,\chi,\rho} problem we require:*

$$\begin{aligned} \sqrt{2\pi e} \cdot \sigma &< a^{\deg(r)/2n} \\ \sqrt{n} \cdot \sigma &< \rho \\ \sqrt{2mn} \cdot \sigma &< \min\{\|\mathbf{x}\| \mid \mathbf{x} \in gR \setminus \{0\}\} \end{aligned}$$

⁴ If the GCD of all the entries of \mathbf{v} (lifted to R) together with g is not 1 then the volume is larger.

where σ is the standard deviation of the distribution χ . The three conditions come respectively from the expected length of any $(\iota(\mathbf{u}_i), \iota(\mathbf{v}_i))$ being smaller than the Gaussian heuristic, than the bound ρ required for a solution to exist and finally that it is shorter than any non-zero spurious vector.

Remark 1. Even if g is an integer, in some cases, such as when $f = X^n - 1$, the first m rows of the reduced basis for the lattice will not be linearly independent as elements of R^m since multiplication by X gives another short vector in the lattice. This means they will not give the invertible matrix required and so one must look further than just the first m rows of the reduced basis to find suitable linearly independent vectors. While this does add some more complexity to the attack it is not the bottleneck so we do not consider this issue further here.

4.2 Zero-forcing attacks

It was noted by May [58] that the standard lattice attack does not take into consideration that it is typical for the coefficient vectors of \mathbf{u} and \mathbf{v} not only to be short but actually be rather sparse; that is to say many of their coefficients are zero. To aid lattice reduction, he suggested to multiply certain columns in the standard lattice by a large scaling factor in order to reduce the space of short lattice vectors and all but necessitate that the vectors found will be zero in these columns. Initially, in the context of having $f(X) = X^n - 1$, the first c columns corresponding to coefficients of \mathbf{v} were suggested to be chosen, akin to the assumption that there exists a so-called zero-run of c zero coefficients in \mathbf{v} since multiplying by X^i (a so-called rotation) cyclically shifts the coefficients while leaving their value unchanged.

It was quickly noted by a number of people that one need not choose consecutive coefficients but any set of c columns works. In fact, May and Silverman argue that choosing columns uniformly at random is the best recourse for an attacker using this approach [59].

While the above method strongly encouraged lattice reduction algorithms to produce vectors with zeros in certain coordinates, it does not actually reduce the dimension of the lattice being reduced, only the dimension of the space of small solutions. Silverman [69] demonstrates a much more efficient manner of achieving this property that does reduce the dimension of the lattice to be reduced, which he calls a zero-forced lattice.

The approach is a straightforward application of simple linear algebra. One starts with the n linear equations in $2n$ unknowns

$$v_j \equiv \sum_{i=0}^{n-1} u_i h_{i,j} \pmod{q} \quad \text{for } j \in \{0, 1, \dots, n-1\}$$

and chooses subsets $\mathcal{I}, \mathcal{J} \subseteq \{0, 1, \dots, n-1\}$ which are the indices of \mathbf{u} and \mathbf{v} , respectively, which are being forced to be zero. Naturally, we assume $|\mathcal{I}| + |\mathcal{J}| \leq$

n . Setting $u_i = 0$ for $i \in \mathcal{I}$ and $v_j = 0$ for $j \in \mathcal{J}$ gives us $|\mathcal{J}|$ linear equations

$$\sum_{\substack{i=0 \\ i \notin \mathcal{I}}}^{n-1} u_i h_{i,j} \equiv 0 \pmod{q} \quad \text{for } j \in \mathcal{J}$$

in $n - |\mathcal{I}|$ unknowns. Suppose we can rewrite this system of linear equations in a way to express the variables u_i for $i \in \mathcal{I}' \subseteq \{0, 1, \dots, n-1\} \setminus \mathcal{I}$ in terms of the remaining variables, where $|\mathcal{I}'| = |\mathcal{J}|$:

$$u_\iota \equiv \sum_{\substack{i=0 \\ i \notin \mathcal{I} \cup \mathcal{I}'}}^{n-1} u_i \beta_{i,\iota} \pmod{q} \quad \text{for } \iota \in \mathcal{I}'.$$

Substituting these expressions back into the remaining equations we started with gives a reduced system of $n - |\mathcal{J}|$ equations

$$v_j \equiv \sum_{\substack{i=0 \\ i \notin \mathcal{I} \cup \mathcal{I}'}}^{n-1} u_i \alpha_{i,j} \pmod{q} \quad \text{for } j \in \{0, 1, \dots, n-1\} \setminus \mathcal{J}$$

in $2(n - |\mathcal{J}|) - |\mathcal{I}|$ unknowns, for some constants $\alpha_{i,j}$. This system will have a small solution if the initial choices of \mathcal{I} and \mathcal{J} were good ones. The lattice that is now of interest is generated by the rows of the matrix

$$\begin{pmatrix} \lambda I_{n-|\mathcal{I}|-|\mathcal{J}|} & A \\ 0 & qI_{n-|\mathcal{J}|} \end{pmatrix},$$

where $A = (\alpha_{i,j})$ where i runs through $\{0, 1, \dots, n-1\} \setminus (\mathcal{I} \cup \mathcal{I}')$ and j runs through $\{0, 1, \dots, n-1\} \setminus \mathcal{J}$. Running a strong enough lattice reduction algorithm on this lattice will reveal any pairs $(\mathbf{u}', \mathbf{v}')$ which conform to the choice of \mathcal{I} and \mathcal{J} .

It should be noted that for any vector found by reducing the zero-forced lattice above there is no absolute guarantee that the u_i are small for $i \in \mathcal{I}'$. One can consider so-called non-lossy zero-forced lattices which ensure these other coefficients are small [67]. The non-lossy zero-forced lattice is spanned by the rows of the block matrix

$$\begin{pmatrix} \lambda I_{n-|\mathcal{I}|-|\mathcal{J}|} & A & \lambda B \\ 0 & qI_{n-|\mathcal{J}|} & 0 \\ 0 & 0 & \lambda qI_{|\mathcal{J}|} \end{pmatrix},$$

where $B = (\beta_{i,\iota})$ for $i \in \{0, 1, \dots, n-1\} \setminus (\mathcal{I} \cup \mathcal{I}')$ and $\iota \in \mathcal{I}'$. However, Rosenberg notes that in most of the cases he considered the attack is more efficient if one does not include these extra columns.

Again, this type of attack works best when the defining polynomial modulus f is very sparse as then rotations of (\mathbf{u}, \mathbf{v}) are still sparse so the probability of choosing good sets \mathcal{I} and \mathcal{J} is significantly improved.

The attack of Beunardeau et al.

Soon after the initial appearance of [1] in which the authors claimed adapting known lattice attacks against NTRU would be ineffective against the MLHR problem, Beunardeau, Connolly, Géraud and Naccache [20] presented a combinatorial attack, using lattice reduction as a subroutine, which they claimed could be used to successfully solve the MLHR problem with parameters which were claimed to give a security of roughly 120 bits. While not mentioned anywhere, one should see this attack as the natural generalisation of the zero-forcing attack on the NTRU problem given above.

Recall that the $\text{MLHR}_{n,h}$ problem can be written in our language as the $\text{Ideal-NTRU}_{X^{n-1}, X^{-2}, 1, \chi, \rho}$ problem in which χ is the uniform distribution over polynomials with binary coefficients having exactly h coefficients equal to one. In this case, the ring R_g is equal to \mathbb{Z}_M for a Mersenne number $M = 2^n - 1$ hence the problem involves only large integer arithmetic. We will follow the original description of the attack given in [20].

The main starting point to the attack is to note that, as elements of \mathbb{Z}_M , \mathbf{v} and \mathbf{u} have Hamming weight h as binary strings and hence have sparse binary expansions. This allows one to guess that certain bits in \mathbf{v} or \mathbf{u} are zero. For this, Beunardeau et al. partition the set $\{0, 1, \dots, n-1\}$ into interval-like parts in two ways: one for \mathbf{v} and one for \mathbf{u} . Such an interval-like partition \mathcal{P} is given by indices $0 \leq p_1, \dots, p_k < n$ and the parts are of the form $\{p_i, \dots, p_{i+1} - 1\}$ for $1 \leq i < k$ and $\{p_k, \dots, n-1\}$ if $p_1 = 0$ and $\{p_k, \dots, n-1, 0, \dots, p_1 - 1\}$ otherwise.

Each part of an interval-like partition will be called a block and blocks will be classified as either type 0 or type 1 as follows. A type 0 block is one in which we guess that the binary expansion of the integer in question (either \mathbf{v} or \mathbf{u}) has all bits equal to 0; a type 1 block makes no assumption on the bits. Furthermore, only balanced partitions are considered which means that the total length of the type 0 blocks differs by at most one from the total length of the type 1 blocks in the partition. Blocks of type 0 will also be called zero blocks and those of type 1 called non-zero blocks.

In essence then, when choosing a pair of partitions for \mathbf{v} and \mathbf{u} we are guessing approximately half of the bit positions in each of \mathbf{v} and \mathbf{u} are zero. Now given that exactly h of the n bits are one in the case of interest, for a given partition there is a probability of approximately 2^{-h} that a given integer modulo $M = 2^n - 1$ having Hamming weight h conforms to the guess. Thus the probability over all possible pairs \mathbf{v} and \mathbf{u} that one chooses a correct pair of partitions is approximately 2^{-2h} .

Suppose the partitions we have chosen for \mathbf{u} and \mathbf{v} have non-zero blocks starting at bits $s_1, \dots, s_{k/2}$ and $t_1, \dots, t_{k/2}$ respectively. Further, denote by u_i and v_i the lengths of the non-zero blocks and set $w := \max_i(u_i, v_i)$. For a parameter

K , define the lattice $\Lambda_K(\mathbf{h})$ as the span of the rows of the matrix

$$\begin{pmatrix} 2^{w-u_1} & 0 & \dots & 0 & | & 0 & 0 & \dots & 0 & | & KH2^{s_1} \\ 0 & 2^{w-u_2} & \dots & 0 & | & 0 & 0 & \dots & 0 & | & KH2^{s_2} \\ \vdots & \vdots & \ddots & \vdots & | & \vdots & \vdots & \ddots & \vdots & | & \vdots \\ 0 & 0 & \dots & 2^{w-u_{k/2}} & | & 0 & 0 & \dots & 0 & | & KH2^{s_{k/2}} \\ \hline 0 & 0 & \dots & 0 & | & 2^{w-v_1} & 0 & \dots & 0 & | & -K2^{t_1} \\ 0 & 0 & \dots & 0 & | & 0 & 2^{w-v_2} & \dots & 0 & | & -K2^{t_2} \\ \vdots & \vdots & \ddots & \vdots & | & \vdots & \vdots & \ddots & \vdots & | & \vdots \\ 0 & 0 & \dots & 0 & | & 0 & 0 & \dots & 2^{w-v_{k/2}} & | & -K2^{t_{k/2}} \\ \hline 0 & 0 & \dots & 0 & | & 0 & 0 & \dots & 0 & | & KM \end{pmatrix}.$$

We can write

$$\mathbf{u} = \sum_{i=1}^{k/2} x_i 2^{s_i} \bmod M, \quad \mathbf{v} = \sum_{i=1}^{k/2} y_i 2^{t_i} \bmod M$$

for non-negative integers x_i, y_i which for a correctly chosen pair of partitions can be taken to be small. Choosing an appropriately large K further means that $(x_1, \dots, x_{k/2}, y_1, \dots, y_{k/2}, 0)$ is a short vector in $\Lambda_K(\mathbf{h})$; however, there may be shorter vectors in the lattice. In particular, there are vectors of length $\sqrt{2^\ell + 1}$ for any $\ell = s_{i+1} - s_i, \ell = t_{i+1} - t_i, \ell = n + s_1 - s_{k/2}$ or $\ell = n + t_1 - t_{k/2}$ so that for larger values of k there will be too many other shorter vectors in the lattice to stand a chance of finding one of interest.

The attack consists of sampling pairs of balanced interval-like partitions uniformly at random, constructing the lattice $\Lambda_K(\mathbf{h})$ corresponding to this pair of partitions and then running the LLL algorithm in the hope of recovering the vector corresponding to a suitable solution \mathbf{v}', \mathbf{u}' with both having binary expansions of Hamming weight h and with $\mathbf{h} = \mathbf{v}'\mathbf{u}'^{-1} \bmod M$. We note that as $f = X^n - 1$, rotations of \mathbf{u} and \mathbf{v} are also valid solutions which aids the attack.

This attack was analysed by de Boer, Ducas, Jeffery and de Wolf in [22]; they argue that under standard lattice heuristics, for each possible pair of partitions the probability of solving the MLHR $_{n,h}$ problem using this approach is $(\frac{1}{2} + c(\frac{k}{h} + o(1)))^{2h}$ for a small constant c which they estimate to be 1/140. They therefore suggest one should start by considering partitions with a small number of blocks k , taking advantage of the smaller dimension for the constructed lattice and slightly larger success probability and gradually increase k until one finds a suitable solution.

We remark that this analysis is done with respect to all possible choices of \mathbf{u} and \mathbf{v} . It is an open question as to whether there exist choices of \mathbf{u} and \mathbf{v} for which this attack does not succeed with constant probability in time $2^{(2+\delta)h+o(1)}$.

Generalisation

Switching from an integer to a polynomial ciphertext modulus presents a slight problem for the zero-forcing attack as now the coordinates of the lattice vectors

are not independent with respect to reduction modulo the ciphertext modulus, a fact which becomes important when choosing which coefficients to set to zero. As seen with the MLHR problem where $f = X^n - 1$ and the ciphertext modulus is $X - 2$, or more generally when g is monic and linear, one should instead partition the n coordinates into blocks which should be treated in a similar way to individual coordinates.

More generally, when there is coordinate dependency due to g being a non-constant polynomial, finding a basis for the sublattice of the standard Ideal-NTRU lattice which corresponds to setting a certain set of blocks to be zero is unfeasible. Instead, one can consider the superlattice

$$\Lambda'(\mathbf{h}) = \{(\iota(\mathbf{x}), \iota(\mathbf{y}), \mathbf{z}) \in \mathbb{Z}^{m(2n+\deg r)} \mid \mathbf{x}, \mathbf{y} \in R^m, \mathbf{z} \equiv \iota(\text{rep}_g(\mathbf{x}\mathbf{h}^T - \mathbf{y})) \pmod{a}\}$$

which contains $\Lambda(\mathbf{h})$ as the sublattice for which $\mathbf{z} = \mathbf{0}$. Here we have abused notation slightly as ι takes elements of R to elements of \mathbb{Z}^n , however elements of $\iota(\text{Rep}(R_g))$ only have non-zero coordinates in the first $\deg r$ positions so we can drop the remaining zero coordinates.

Using the trick of multiplying the final $m \deg r$ coordinates by a large scalar before applying lattice reduction means we will find many vectors for which these coordinates are zero, just as May did. In particular, we will want to perform lattice reduction on some sublattice of

$$\Lambda'_K(\mathbf{h}) = \{(\mathbf{x}, \mathbf{y}, \mathbf{z}) \in \mathbb{Z}^{m(2n+\deg r)} \mid (\mathbf{x}, \mathbf{y}, K^{-1}\mathbf{z}) \in \Lambda'(\mathbf{h})\},$$

for a large integer K , corresponding to setting some of the coordinates to zero.

Due to the freedom now present in introducing \mathbf{z} it is straightforward to compute a generating set of vectors for the sublattice of $\Lambda'_K(\mathbf{h})$ for which a given set among the first $2mn$ coordinates are all zero. The most important point here is to look at the form taken by the vectors of such a sublattice which are zero modulo the ciphertext modulus g as these are not actually solutions.

Returning to the case of the MLHR problem and the attack of Beunardeau et al. we saw that there were vectors in the standard lattice attack whose norm was $\sqrt{5}$. However, they always contained consecutive non-zero coordinates up to cycles since they correspond to $X^\ell(X - 2)$ for some ℓ . In the zero-forcing attack, let us assume for simplicity that the coordinate corresponding to the constant coefficient (of some copy of R) has not been picked to be set to zero. We know that $X^\ell - 2^\ell \equiv 0 \pmod{gR}$ so we have vectors of norm $\sqrt{1 + 2^{\ell+1}}$ for $1 \leq \ell < n$ whenever the ℓ th coordinate of the same copy of R can also be non-zero.

We therefore see that in general the length of these trivial vectors depends on the gap between consecutive coordinates which have not been set to zero. In this sense, it makes more sense to consider which coordinates we allow to be non-zero rather than the description in terms of blocks that was given in [20]. The connection between the two views is that the coordinates we allow to be non-zero are the coordinates which begin a non-zero block.

Explicitly, suppose we chose the sublattice so that the non-zero coordinates for the i th copy of R^m corresponding to \mathbf{x} are at columns $\alpha_{i,j} \in \{0, \dots, n - 1\}$ and are at columns $\beta_{i,j}$ for the i th copy of R^m corresponding to \mathbf{y} , where we

When $f = X^n + f_0$ for some small integer f_0 and g has small degree and small coefficients and the above paragraph does not apply, one should choose sets of $\deg g$ consecutive coordinates which are allowed to be non-zero. To illustrate this, we give the following example.

Example 1. Suppose we have $f = X^n - 1$ and $g = X^2 - X - 2$ for some odd n (so that there is no evaluation at -1 attack). It is easy to check that $X^p \equiv \frac{1}{3}(2^{j+1} + (-1)^j)X^{p-j} + \frac{2}{3}(2^j - (-1)^j)X^{p-j-1} \pmod{gR}$ for any $0 \leq j < p$. Taking our sublattice of $\Lambda'_K(\mathbf{h})$ to have pairs of consecutive non-zero coordinates means we will always be able to find a vector in this sublattice corresponding to a desired solution. Furthermore, there are trivial vectors in the sublattice of norm $\frac{1}{3}\sqrt{2^{2j+3} - 4(-2)^j + 14}$ where j is the difference (modulo n coming from the structure of R) between two such pairs of non-zero coordinates (counting from the first coordinate of each pair, say). Thus, so long as we spread out our guesses for these pairs of consecutive non-zero coordinates enough to ensure that any trivial vectors in the sublattice are not too short then we have a reasonable probability of solving the Ideal-NTRU problem by running the LLL algorithm on that sublattice (over a random distribution of \mathbf{u} and \mathbf{v}).

On the other hand, when g is not of this form due to having at least one coefficient that is not small then we can apply the standard Ideal-NTRU lattice attack of Section 4.1.

We further remark that in the case of the MLHR problem, de Boer et al. modify the lattice to balance the expected sizes of the first $k_{\mathbf{x}} + k_{\mathbf{y}}$ entries of the lattice vectors by scaling the i th coordinate by certain powers of two. Further, to simplify matters they chose exactly k non-zero coordinates in every copy of R so that $k_{\mathbf{x}} = k_{\mathbf{y}} = km$.

Without scaling, our lattice now has volume $(aK)^{rm}$ and as we take K to be very large the lattice is clearly in the approximation regime so we expect that running the LLL algorithm will output particularly short vectors. In practice, when attacking the MLHR search problem we found that even with a suitable guess for the non-zero coordinates, LLL is not guaranteed to find a valid solution since other shorter vectors may exist in the lattice which do not correspond to admissible solutions due to having negative entries (although they will most often do so).

Requirements. *The main requirement for this type of attack is that the small elements we are interested in have sparse coefficient vectors, meaning that many of the coefficients are zero. Further, the applicability and usefulness of the attack are very dependent on the shape of f and g . Again, if these polynomials are very sparse and have very small coefficients the attack works well while the more non-zero coefficients that f and g have and the larger they are, the harder the attack is to perform.*

4.3 Meet-in-the-middle attack

Following a description given by Odlyzko; Howgrave-Graham, Silverman and Whyte [49] describe and analyse a meet-in-the-middle attack on the NTRU

problem. One assumes that \mathbf{u} has a fixed number d of coefficients which are one and the remaining coefficients are all zero. The main idea is to split \mathbf{u} in two, $\mathbf{u} = \mathbf{u}_1 + \mathbf{u}_2$, such that both \mathbf{u}_1 and \mathbf{u}_2 have $d/2$ non-zero coefficients. One notes that $\mathbf{h}\mathbf{u}_1 + \mathbf{h}\mathbf{u}_2 \equiv \mathbf{v} \pmod q$ so that the coefficients of $\mathbf{h}\mathbf{u}_1$ and $-\mathbf{h}\mathbf{u}_2$ differ by either zero or one modulo q when \mathbf{v} has binary coefficients.

When $f = X^n - 1$, it is easy to show that some rotation of \mathbf{u} (i.e. some $\mathbf{u}X^j$) has exactly $d/2$ non-zero coefficients among its first $n/2$, thus we can restrict the possible \mathbf{u}_1 to only have non-zero coefficients among its first $n/2$ coefficients and to \mathbf{u}_2 only among its last $n/2$ coefficients. One must then compute all possible $\mathbf{h}\mathbf{u}_1$ and sort them into a suitably chosen set of buckets. Secondly, one begins computing $-\mathbf{h}\mathbf{u}_2$ for each \mathbf{u}_2 in turn, in each case searching in the associated buckets for a \mathbf{u}_1 such that $\mathbf{h}(\mathbf{u}_1 + \mathbf{u}_2)$ has binary coefficients. When such a pair $(\mathbf{u}_1, \mathbf{u}_2)$ is found one stops otherwise one continues to the next \mathbf{u}_2 .

In the basic approach, they suggest choosing an integer k such that 2^k is larger than $\binom{n/2}{d/2}$ and labelling each bucket by a binary string of length k . If \mathbf{b}_i is the most-significant bit of the i th coefficient of $\mathbf{h}\mathbf{u}_1$ then one places \mathbf{u}_1 in the bucket labelled $(\mathbf{b}_0\mathbf{b}_1 \cdots \mathbf{b}_{k-1})_2$. One computes the buckets in which to check for a collision in the same way, though now also any bucket with a label which arises by adding one to each element of any subset of the first k coefficients of $-\mathbf{h}\mathbf{u}_2$ is also checked.

The authors of [49] analyse the time and memory required for this as well as further improvements to this design.

The meet-in-the-middle attack of de Boer et al.

The same idea can be applied to the $\text{MLHR}_{n,h}$ problem. Indeed, de Boer et al. [22] not only analysed the attack of Beunardeau et al. but also such a meet-in-the-middle attack. If we define $|C|_{\text{Ham}}$ to be the Hamming weight of C as a bit string for any $C \in \{0, 1, \dots, M-1\}$ where again $M = 2^n - 1$, then the aim of the attack is to find a \mathbf{u} such that $|\mathbf{u}|_{\text{Ham}} = h$ and $|\mathbf{h}\mathbf{u} \pmod M|_{\text{Ham}} = h$. To do this one defines two sets, depending on a parameter $\alpha \in [0, 1]$,

$$S_1^{(\alpha)} := \left\{ s \in \{0, 1, \dots, M-1\} \mid 2^{\lceil(1-\alpha)n\rceil} \mid s \text{ and } |s|_{\text{Ham}} = \lfloor \alpha h \rfloor \right\}$$

$$S_2^{(\alpha)} := \left\{ s \in \{0, 1, \dots, 2^{\lceil(1-\alpha)n\rceil} - 1\} \mid |s|_{\text{Ham}} = \lceil (1-\alpha)h \rceil \right\}.$$

While there is no guarantee that \mathbf{u} can be written as $\mathbf{u}_1 + \mathbf{u}_2$ for some $\mathbf{u}_1 \in S_1^{(\alpha)}$ and $\mathbf{u}_2 \in S_2^{(\alpha)}$, due to the form of M , it is true that $2^k\mathbf{u}$ can be written in this form for at least one $0 \leq k < n$ and both $2^k\mathbf{u}$ and $2^k\mathbf{v} = \mathbf{h} \cdot 2^k\mathbf{u} \pmod M$ still have Hamming weight h .

The attack begins by enumerating the pairs $(\mathbf{u}_1, \mathbf{h}\mathbf{u}_1 \pmod M)$ for all $\mathbf{u}_1 \in S_1^{(\alpha)}$ and then for each $\mathbf{u}_2 \in S_2^{(\alpha)}$ computing $-\mathbf{h}\mathbf{u}_2 \pmod M$ and looking for an approximate collision with some $\mathbf{h}\mathbf{u}_1$ such that the Hamming distance between $\mathbf{h}\mathbf{u}_1 \pmod M$ and $-\mathbf{h}\mathbf{u}_2 \pmod M$ is not much bigger than $2h$.

In order to efficiently check for an approximate collision de Boer et al. employ a locality-sensing hash function. In particular, for any subset $\mathcal{B} \subseteq \{0, 1, \dots, n\}$

one defines the hash function $\mathcal{H}_{\mathcal{B}}: \mathbb{Z}/\mathbb{Z}M \rightarrow \mathbb{F}_2^{|\mathcal{B}|}$ which when applied to an n -bit integer $(b_{n-1}b_{n-2}\cdots b_0)_2$ returns $(b_{i_1}, \dots, b_{i_B})$ where $\mathcal{B} = \{i_1, \dots, i_B\}$ and the i_j are ordered in some fixed order, say ascending order. For a suitable size B of \mathcal{B} one has that if two n -bit integers have Hamming distance not much larger than $2h$ then they are likely to agree on $\mathcal{H}_{\mathcal{B}}$. By storing the hash value $\mathcal{H}_{\mathcal{B}}(Hs_1 \bmod M)$ instead of simply $Hs_1 \bmod M$ and sorting via this value one can quickly find all such collisions with $\mathcal{H}_{\mathcal{B}}(-\mathbf{h}\mathbf{u}_2 \bmod M)$ and then test whether each \mathbf{u}_1 coming from such a collision has the property that $|\mathbf{u}_1 + \mathbf{u}_2|_{\text{Ham}} = h$ and $|\mathbf{h}(\mathbf{u}_1 + \mathbf{u}_2) \bmod M|_{\text{Ham}} = h$. If so then $\mathbf{u} = \mathbf{u}_1 + \mathbf{u}_2$ gives a solution to the MLHR $_{n,h}$ problem.

On analysing this approach and under some simple heuristics the authors gave the following lemma.

Lemma 1 (Lemma 3 from [22]). *When $\alpha = 1/2$ and $B = \lceil \log_2 \binom{n/2}{h/2} \rceil$, the time complexity of the meet-in-the-middle algorithm is $\tilde{O}\left(\sqrt{\binom{n}{h}}\right)$.*

By using a quantum algorithm they also show that choosing $\alpha = 1/3$ and $B = \lceil \log_2 \binom{n/3}{h/3} \rceil$ gives a running time of $\tilde{O}\left(\sqrt[3]{\binom{n}{h}}\right)$ with the same memory requirement with most of that memory required to be quantumly accessible.

Generalisation

The generalisation of this type of attack is pretty straightforward. Given a quotient of small elements $\mathbf{h} = \mathbf{v}\mathbf{u}^{-1}$ in $R_g^{m \times m}$, the basic idea is to split the small element \mathbf{u} into two parts as $\mathbf{u} = \mathbf{u}_1 + \mathbf{u}_2$ where $\mathbf{u}_1 \in U_1$ and $\mathbf{u}_2 \in U_2$ for two sets U_1 and U_2 . Then one can compute and store all possible values $\mathbf{h}\mathbf{u}_1$ and attempt to find a \mathbf{u}_2 such that $-\mathbf{h}\mathbf{u}_2$ approximately collides with one of the stored values. An approximate collision occurs between $\mathbf{h}\mathbf{u}_1$ and $-\mathbf{h}\mathbf{u}_2$ if their difference consists of small elements; namely that $\mathbf{h}(\mathbf{u}_1 + \mathbf{u}_2)$ is small. The hope is that $\mathbf{u}_1 + \mathbf{u}_2$ and $\mathbf{h}(\mathbf{u}_1 + \mathbf{u}_2)$ satisfy the requirements to be a solution to the problem.

One wants to define the sets U_1 and U_2 in a way which minimises their size while still allowing for at least one solution \mathbf{u}' to be written as a sum of one element of each set. We have seen the two cases where $f = X^n - 1$ and g is either constant or linear and small elements of R are taken to have many zero coefficients and the remaining coefficients are one. In both cases determining an approximate collision is easy, though to speed up the process of finding one, the list of stored values is processed so that entries are placed in a number of different buckets.

When moving to the general setting a few issues can arise. Perhaps the most important of which is whether there is an efficient test for determining approximate collisions as this seems to rely on an efficient method for testing smallness. Secondly, for a more general polynomial f one may not be able to rely on a symmetry argument to reduce the size of U_1 and U_2 since multiplying by X need not preserve smallness. On a similar note, the error distribution used will play a

role in how to optimally define U_1 and U_2 . Alternatively, when $m > 1$ splitting \mathbf{u} component wise with respect to m gives an easy choice for U_1 and U_2 .

Requirements. *To be able to apply this attack one must be able to efficiently determine whether two elements of R_g differ by a small element: an approximation collision. When $m = 1$ the attack works best when the distribution of small elements χ is sparse and remains fixed under multiplication by X .*

4.4 A hybrid attack

In [47], Howgrave-Graham gives a hybrid attack on the NTRU problem, Ideal-NTRU $_{f,q,1,\chi,\rho}$, which combines lattice reduction and a meet-in-the-middle approach. This attack assumes that f is the original NTRU polynomial $f = X^n - 1$ and that \mathbf{u} and \mathbf{v} have binary coefficients.

The starting point for the attack is the observation that while running a moderately strong lattice reduction algorithm (say the BKZ algorithm with a relatively small block size) does not recover a solution, it does produce a reduced basis whose first few, say k , Gram-Schmidt vectors have length q and whose final few, say k' , Gram-Schmidt vectors have length 1.

Defining the slightly modified basis matrix for the standard NTRU lattice as

$$B := \begin{pmatrix} qI_n & 0 \\ H & I_n \end{pmatrix},$$

where $H = (h_{i,j})$ with the $h_{i,j}$ as defined in Section 4.1, then this means there are matrices $P \in \text{GL}(2n, \mathbb{Z})$ and Q orthogonal such that the partially reduced basis matrix is PB and $T := PBQ$ is lower-triangular. More illustratively, this final product can be written as

$$\begin{pmatrix} I_k & 0 & 0 \\ 0 & P' & 0 \\ 0 & 0 & I_{k'} \end{pmatrix} \begin{pmatrix} qI_k & 0 & 0 \\ * & B' & 0 \\ * & * & I_{k'} \end{pmatrix} \begin{pmatrix} I_k & 0 & 0 \\ 0 & Q' & 0 \\ 0 & 0 & I_{k'} \end{pmatrix} = \begin{pmatrix} qI_k & 0 & 0 \\ * & T' & 0 \\ * & * & I_{k'} \end{pmatrix},$$

with $P'B'Q' = T'$. Since Q is orthogonal, the lattice spanned by the rows of T contains the short vector $(v_0 \ v_1 \ \cdots \ v_{n-1} \ u_0 \ u_1 \ \cdots \ u_{n-1})Q$ as well as the $n - 1$ other similar vectors coming from the cyclic symmetry of R . Furthermore, due to the structure of Q , these short vectors are binary in their first k and last k' coordinates.

Suppose we are looking for the lattice vector \mathbf{v} in the row span of T and let $\bar{\mathbf{v}}$ be the vector whose first $2n - r'$ entries are zero and final r' binary entries match those of \mathbf{v} . Howgrave-Graham showed that, if $\mathbf{v} - \bar{\mathbf{v}}$ is small enough then applying Babai's nearest plane algorithm to the target vector $\bar{\mathbf{v}}T - \bar{\mathbf{v}}$ and the lattice generated by the rows of T , then one can recover \mathbf{v} . Thus a valid strategy is to enumerate over possible $\bar{\mathbf{v}}$ for a suitable r' .

The more efficient attack proposed is to apply a meet-in-the-middle attack of the same type described in Section 4.3 on the vector $\bar{\mathbf{v}}$. With a modified approach requiring less memory, this hybrid attack is claimed to be the most practical attack on the NTRU problem for these parameters.

4.5 A folding attack

When the defining polynomial f is of the form $X^n - 1$ and n is composite, say $d \mid n$ and $1 \leq d < n$, then $X^d - 1$ divides f and hence there is a natural ring homomorphism

$$\pi: R = \frac{\mathbb{Z}[X]}{(X^n - 1)} \rightarrow \frac{\mathbb{Z}[X]}{(X^d - 1)}$$

given by simply reducing modulo $X^d - 1$.

Gentry [39] showed that this ring homomorphism can be used to transform the $2n$ -dimensional standard NTRU lattice to a $2d$ -dimensional folded lattice which contains small vectors corresponding to the rotations of the folded secret elements $\pi(\mathbf{u})$ and $\pi(\mathbf{v})$.

There are two points to this, firstly that we can construct this smaller dimensional lattice from the public value \mathbf{h} and secondly that the shortest vectors in this lattice do correspond to a rotation of $\pi(\mathbf{u})$ and $\pi(\mathbf{v})$.

The first point is straightforward; since π is a ring homomorphism which fixes q we have that $\pi(\mathbf{h})\pi(\mathbf{u}) = \pi(\mathbf{h}\mathbf{u}) \equiv \pi(\mathbf{v}) \pmod{q}$. Thus we can construct the folded lattice in the same way as the standard lattice by replacing \mathbf{h} with $\pi(\mathbf{h})$ and n by d .

For the second point, we must look at how small the coefficient vectors of $\pi(\mathbf{u})$ and $\pi(\mathbf{v})$ are. Since each coefficient of the folded element is a sum of n/d of the original coefficients, they will remain small when d is not too small.

If one can mount a successful attack on this smaller dimensional problem one can recover the folded secret elements $\pi(\mathbf{u})$ and $\pi(\mathbf{v})$. Gentry gives a method to recover the full secret (\mathbf{u}, \mathbf{v}) from this partial information which reduces the dimension $2n$ of the standard lattice attack to a dimension of roughly $2(n - d)$. This reduction comes from the fact that knowledge of $\pi(\mathbf{u})$ gives us d linear relations between the coefficients of \mathbf{u} and similarly for the coefficients of \mathbf{v} . We refer to the paper [39] for the explicit case when $n = 2d$.

Gentry remarks that this attack only really requires that, modulo q , the polynomial f has a factor f_1 such that the projection $\mathbb{Z}_q[X]/(f(X)) \rightarrow \mathbb{Z}_q[X]/(f_1(X))$ does not distort the notion of smallness too much. Namely, the coefficients of f_1 must be very small and, ideally, only the low degree monomials (except the leading term) should have non-zero coefficients. This condition on f_1 seems highly sporadic in general.

Generalisation

Due to π being a ring homomorphism such an attack can be directly applied to the more general Ideal-NTRU $_{f,q,m,\chi,\rho}$ problem component-wise. For a more general ciphertext modulus g , we would need an appropriate factor of f when considered modulo g and again $\pi(\chi)$ must remain a distribution of small elements.

4.6 A subfield attack

In the case that the defining polynomial f is irreducible then the ring R can be seen as an order in the number field $K = \mathbb{Q}(X)/(f(X))$. If this number field has subfields then one can consider maps whose codomain is contained in such a subfield, L say. Such maps include the relative norm and trace maps, $N_{K/L}$ and $\text{Tr}_{K/L}$.

Since the norm map is multiplicative and the trace map is additive it is natural to consider the norm map first as the public value \mathbf{h} is a quotient of small elements. Importantly though, this quotient is taken modulo q so one must consider how the norm map interacts with reduction modulo q . As the relative norm map is defined as the product of the field embeddings, σ_i , that fix L and hence q we have for any lift \tilde{h} of \mathbf{h} to $R \subseteq K$ that for any $k \in R$

$$\begin{aligned} N_{K/L}(\tilde{h} + qk) &= \prod_i \sigma_i(\tilde{h} + qk) = \prod_i (\sigma_i(\tilde{h}) + q\sigma_i(k)) \\ &\equiv \prod_i \sigma_i(\tilde{h}) = N_{K/L}(\tilde{h}) \pmod{q\mathcal{O}_L}, \end{aligned}$$

where \mathcal{O}_L is the ring of integers of L .⁵ This is true since considering the left hand side of the equivalence as a polynomial in q , all coefficients are symmetric with respect to the field embeddings and hence lie in \mathcal{O}_L . For much the same reasoning, we also have $\text{Tr}_{K/L}(\tilde{h} + qk) \equiv \text{Tr}_{K/L}(\tilde{h}) \pmod{q\mathcal{O}_L}$. Hence, there is a well-defined notion of taking the norm or trace of an element of R_q which we denote in the same manner.

Using the relative norm map Now it is clear that $N_{K/L}(\mathbf{u})N_{K/L}(\mathbf{h}) = N_{K/L}(\mathbf{v})$, we are in much the same situation as with the folding attack, discussed above, only with $N_{K/L}$ instead of π . Firstly, one needs to see how large the elements $N_{K/L}(\mathbf{u})$ and $N_{K/L}(\mathbf{v})$ are and secondly, if one can recover these elements, how can one recover the original \mathbf{u} and \mathbf{v} ?

This sort of approach was first considered in [40, Section 6 and 7] where it is attributed to Gentry, Szydlo, Jonsson, Nguyen and Stern. The setting is slightly different in the fact that they work with the defining polynomial $f(X) = X^n - 1$ for a prime n (so folding is not possible) which is not irreducible however it is closely related to the case when the defining polynomial is the irreducible n th cyclotomic polynomial $\Phi_n(X)$ and the subfield is the maximal real subfield. The Gentry-Szydlo algorithm described in Section 7 of their paper can be seen as a method for computing \mathbf{v} from \mathbf{h} and the relative norm of \mathbf{v} with respect to the maximal real subfield.

More generally, Albrecht, Bai and Ducas [5] were the first to consider the case of arbitrary subfields of cyclotomic number fields. They use the following heuristic on the growth of the canonical norm $\|\cdot\|^{\text{can}}$ and the operator norm $\|\cdot\|_{\text{op}}$, defined as $\|y\|_{\text{op}} := \sup_{x \in K^\times} \|xy\|^{\text{can}} / \|x\|^{\text{can}}$.

⁵ The ring of integers of L , \mathcal{O}_L , is the set of elements in L which are the root of a monic polynomial with integer coefficients.

Heuristic 1. Let $[K : \mathbb{Q}] = n$ and $[K : L] = \ell$ and suppose that \mathbf{u} and \mathbf{v} are sampled from a reasonable isotropic distribution of variance ς^2 . Then, for any $c > 0$, there exists a constant C such that

$$\begin{aligned} \|\mathbf{N}_{K/L}(\mathbf{v})\|^{\text{can}} &\leq (\varsigma n^C)^\ell, & \|\mathbf{N}_{K/L}(\mathbf{u})\|^{\text{can}} &\leq (\varsigma n^C)^\ell, \\ |\mathbf{N}_{K/L}(\mathbf{u})|_{\text{op}} &\leq (\varsigma n^C)^\ell, & |\mathbf{N}_{K/L}(\mathbf{u})^{-1}|_{\text{op}} &\leq (n^C/\varsigma)^\ell \end{aligned}$$

except with probability $O(n^{-c})$.

This heuristic tells us that for suitable subfields, so that the numerator and the denominator are small compared to q , one may be able to recover $\mathbf{N}_{K/L}(\mathbf{u})$ and $\mathbf{N}_{K/L}(\mathbf{v})$ by using a strong enough lattice reduction algorithm on the standard NTRU lattice $\Lambda_{\text{NTRU}}(\mathbf{N}_{K/L}(\mathbf{h}))$, as long as the associated lattice vector remains an unusually small vector. In particular, Albrecht et al. give the following theorem.

Theorem 1 (Theorem 2 from [5]). Let $\hat{\mathbf{u}}$ and $\hat{\mathbf{v}}$ be elements of \mathcal{O}_L such that the principal ideals they generate are coprime and that $\hat{\mathbf{u}}\hat{\mathbf{h}} \equiv \hat{\mathbf{v}} \pmod{q\mathcal{O}_L}$ for some $\hat{\mathbf{h}} \in \mathcal{O}_L$. By an abuse of notation write $(\hat{\mathbf{x}}, \hat{\mathbf{y}})$ for the vector which concatenates the coefficients of $\hat{\mathbf{x}}$ and $\hat{\mathbf{y}}$. If $(\hat{\mathbf{x}}, \hat{\mathbf{y}}) \in \Lambda_{\text{NTRU}}(\hat{\mathbf{h}})$ has length satisfying

$$\|(\hat{\mathbf{x}}, \hat{\mathbf{y}})\| \leq \frac{q}{\|(\hat{\mathbf{u}}, \hat{\mathbf{v}})\|}$$

then $\hat{\mathbf{x}} = w\hat{\mathbf{u}}$ and $\hat{\mathbf{y}} = w\hat{\mathbf{v}}$ for some $w \in \mathcal{O}_L$.

Finally, the authors give a simple method to lift a short lattice vector $(\hat{\mathbf{x}}, \hat{\mathbf{y}}) \in \Lambda_{\text{NTRU}}(\mathbf{N}_{K/L}(\mathbf{h}))$ as in the above theorem to a pair $(\mathbf{u}', \mathbf{v}') \in \Lambda_{\text{NTRU}}(\mathbf{h})$ by setting $\mathbf{u}' = \hat{\mathbf{x}}$ and $\mathbf{v}' = \mathbf{h}\hat{\mathbf{x}}^{-1}$. While this will not return the shortest vector in $\Lambda_{\text{NTRU}}(\mathbf{h})$ it may be small enough to be a solution to the NTRU problem $\text{Ideal-NTRU}_{f,q,1,\chi,\rho}$ when q is exponentially large. Explicit details of this can be found in [5].

Using the relative trace map Another approach to a subfield attack is to use the relative trace map as was done by Cheon, Jeong and Lee [29] in work that was done concurrently to [5]. Unlike with the relative norm map, the relative trace map is not multiplicative. Instead, one has when $[K : L] = \ell$ that

$$\begin{aligned} \text{Tr}_{K/L}(\mathbf{h}) &= \sum_{i=1}^{\ell} \sigma_i(\mathbf{h}) \equiv \sum_{i=1}^{\ell} \frac{\sigma_i(\mathbf{v})}{\sigma_i(\mathbf{u})} = \frac{\sum_{i=1}^{\ell} \sigma_i(\mathbf{v}) \prod_{j \neq i} \sigma_j(\mathbf{u})}{\prod_{i=1}^{\ell} \sigma_i(\mathbf{u})} \\ &\equiv \frac{\text{Tr}_{K/L}(\sigma_1(\mathbf{v}) \prod_{i=2}^{\ell} \sigma_i(\mathbf{u}))}{\mathbf{N}_{K/L}(\mathbf{u})} \pmod{q\mathcal{O}_L}. \end{aligned}$$

From this, one can notice that when \mathbf{u} and \mathbf{v} have roughly the same size, applying the norm map gives a better bound on the numerator of the result. However,

when \mathbf{v} is significantly larger than \mathbf{u} the trace map is better. Since one can compute \mathbf{h}^{-1} this also holds when \mathbf{u} is significantly larger than \mathbf{v} .

Cheon et al. focus on the power-of-two cyclotomic case and consider \mathbf{u} and \mathbf{v} to have Euclidean norms bounded by reals D and N , respectively. Let us write ψ_D and χ_N for distributions which satisfy these bounds and $\text{NTRU}_{f,q,\chi_N,\psi_D,\rho}$ for the NTRU problem (with $m = 1$) which uses these two distributions for \mathbf{v} and \mathbf{u} , respectively. Then they prove the following reduction can be achieved by using the trace map.

Theorem 2 (Theorem 1 from [29]). *Let q be a positive integer and D and N be positive real numbers and let n be a power of two. Set $\rho := \min\{q/2D\sqrt{n}, q/2N\sqrt{n}\}$. Then for $m > 1$ with $m \mid n$, there is a reduction from $\text{NTRU}_{X^{n+1},q,\chi_N,\psi_D,\rho}$ to $\text{NTRU}_{X^{m+1},q,\chi_{N'},\psi_{D'},\rho'}$, where*

$$\begin{aligned} \rho' &= \min\{q/2D'\sqrt{n}, q/2N'\sqrt{n}, q/2n^{3/2}N^2 \|\mathbf{v}^{-1}\|\}, \\ D' &= D^m \sqrt{(n/\sqrt{m})^{\log_2 m} / \sqrt{m}} \text{ and} \\ N' &= ND^{m-1} \sqrt{(n/\sqrt{m})^{\log_2 m} / \sqrt{m}}. \end{aligned}$$

The proof uses the same method as [5] to lift a solution from a subfield to the full field. Further, much the same requirements on q are required for this result to be used in a practical attack; a very large q . Again, for what this means precisely, see [29].

Other work Finally, we comment that Kirchner and Fouque [52] revisited the subfield attack and proposed a variant of these subfield attacks which performs better in practice.

Generalisation

Let us assume that f is irreducible and hence $K = \mathbb{Q}[X]/(f(X))$ is a number field containing R as a subring. If this is not the case we may be able to first apply the ideas from Section 4.5 to reduce to this case. Further, we let L be a subfield of K and denote by $N_{K/L}$ and $\text{Tr}_{K/L}$ the relative norm and trace maps from K to L , respectively. For simplicity, we assume $m = 1$ however we later show how this generalises for larger m .

We first need to determine whether the multiplicative property of the norm map respects reduction modulo g when g is a polynomial. To do this we note that $\mathbf{h}\mathbf{u} \equiv \mathbf{v} \pmod{gR}$ is equivalent to the existence of $\mathbf{k} \in R$ such that $\mathbf{h}\mathbf{u} = \mathbf{v} + \mathbf{k}g$ in R . Suppose $[K : L] = \ell$ and $\sigma_1, \sigma_2, \dots, \sigma_\ell$ are the ℓ distinct field embeddings $K \hookrightarrow \mathbb{C}$ fixing L . For the approach to work we require a modulus $g' \in \mathcal{O}_L \subset L$

such that $N_{K/L}(\mathbf{h})N_{K/L}(\mathbf{u}) \equiv N_{K/L}(\mathbf{v}) \pmod{g'\mathcal{O}_L}$. Now we compute

$$\begin{aligned} N_{K/L}(\tilde{\mathbf{h}})N_{K/L}(\mathbf{u}) &= N_{K/L}(\tilde{\mathbf{h}}\mathbf{u}) = N_{K/L}(\mathbf{v} + \mathbf{k}g) \\ &= \prod_{i=1}^{\ell} \sigma_i(\mathbf{v} + \mathbf{k}g) = \prod_{i=1}^{\ell} (\sigma_i(\mathbf{v}) + \sigma_i(\mathbf{k})\sigma_i(g)) \\ &= N_{K/L}(\mathbf{v}) + \sum_{j=1}^{\ell} \sigma_j(\mathbf{k})\sigma_j(g) \prod_{\substack{i=1 \\ i \neq j}}^{\ell} \sigma_i(\mathbf{v}) + \cdots \end{aligned}$$

and since \mathbf{k} and \mathbf{v} are unknown we realistically need g' to divide $\sigma_i(g)$ for every i which seems to force $g \in L$, so that g is fixed by each σ_i , and then we take $g' = g$. The result would also hold if there was a way to choose $\tilde{\mathbf{h}}$ such that $\mathbf{k} = 0$ however we have no way to know how to choose such $\tilde{\mathbf{h}}$ without knowledge of \mathbf{u} and \mathbf{v} already. For arbitrary g not in L then, the required condition will not hold in general and the attack appears to be foiled in this case.

When using the trace map instead of the norm map a similar obstruction occurs. In the simplest case we take $\ell = 2$ and let σ_1 be the identity map embedding K into \mathbb{C} . Now, writing our relation instead as $\mathbf{v}\mathbf{u}^{-1} = \tilde{\mathbf{h}} + \mathbf{k}g$ in K we have

$$\mathrm{Tr}_{K/L}(\mathbf{v}\mathbf{u}^{-1}) = \frac{\mathbf{v}}{\mathbf{u}} + \sigma_2\left(\frac{\mathbf{v}}{\mathbf{u}}\right) = \frac{\mathbf{v}\sigma_2(\mathbf{u}) + \mathbf{u}\sigma_2(\mathbf{v})}{\mathbf{u}\sigma_2(\mathbf{u})} = \frac{\mathrm{Tr}_{K/L}(\mathbf{v}\sigma_2(\mathbf{u}))}{N_{K/L}(\mathbf{u})}$$

so that if σ_2 and both the trace and norm maps sufficiently maintain smallness, the trace of a quotient of small elements is also a quotient of small elements. This time we require a modulus $\hat{g} \in L$ such that $\mathrm{Tr}_{K/L}(\tilde{\mathbf{h}}) \equiv \mathrm{Tr}_{K/L}(\mathbf{v}\mathbf{u}^{-1}) \pmod{\hat{g}\mathcal{O}_L}$. We again compute

$$\begin{aligned} \mathrm{Tr}_{K/L}(\mathbf{v}\mathbf{u}^{-1}) &= \mathrm{Tr}_{K/L}(\tilde{\mathbf{h}} + \mathbf{k}g) = \mathrm{Tr}_{K/L}(\tilde{\mathbf{h}}) + \mathrm{Tr}(\mathbf{k}g) \\ &= \mathrm{Tr}_{K/L}(\tilde{\mathbf{h}}) + \mathbf{k}g + \sigma_2(\mathbf{k})\sigma_2(g) \end{aligned}$$

and for the same reasoning, the condition $g \in L$ is sufficient to allow $\hat{g} = g$ and appears to be necessary for the congruence to hold for any arbitrary lift $\tilde{\mathbf{h}}$.

For larger values of ℓ , the formula for the trace of a quotient becomes

$$\mathrm{Tr}_{K/L}(\mathbf{v}\mathbf{u}^{-1}) = \frac{\mathrm{Tr}_{K/L}\left(\mathbf{v} \prod_{i=2}^{\ell} \sigma_i(\mathbf{u})\right)}{N_{K/L}(\mathbf{u})},$$

which again for appropriate choices may be seen as a quotient of small elements and the attack can proceed as before.

In both approaches then, the subfields of K we can use are those between K and $\mathbb{Q}(g(\theta))$ for θ a root of f . If g is linear then there are no such (non-trivial) subfields. An example for which g is a non-constant polynomial and this attack can be applied is when f is a power-of-two cyclotomic polynomial, and g can be written as a polynomial in some power of X , X^p say, where p is a power of

two that is larger than one. Concretely, for example, the choice $f = X^{1024} + 1$, $g = X^8 - X^4 - q$ for very large q would allow the choice $L = \mathbb{Q}(\zeta_{512})$ of subfield of $K = \mathbb{Q}(\zeta_{2048})$.

When considering the problem for larger values of m we note that the entries of the matrix \mathbf{h} are of the form $\mathbf{h}_{i,j} = h_{i,j}(\mathbf{v}_i, \mathbf{u}) / \det(\mathbf{u})$ where \mathbf{v}_i is the i th row of \mathbf{v} and $h_{i,j}$ is a sum of $m!$ distinct monomials of degree m in the entries of \mathbf{v}_i and \mathbf{u} . This can be seen by using Cramer's rule. One can attempt to recover the numerators and common denominator assuming they are small enough via the above methods, since the determinant is exponential in m we can only realistically expect this to work for small m . Indeed, in the next section we briefly look at using the determinant map as a multiplicative homomorphism, this allows one to recover $\det(\mathbf{u})$ with high probability, when applicable.

Remark 2. We point out that, as described in [5, Section 3.3], one can naïvely lift a solution $\mathbf{h}' \equiv \mathbf{v}'\mathbf{u}'^{-1} \pmod{g}$ in the subfield L to one in K for $\mathbf{h} \equiv \mathbf{t}\mathbf{s}^{-1}$ by setting $\mathbf{s} = \pi(\mathbf{u}')$ and $\mathbf{t} = \pi(\mathbf{u}'(\mathbf{h}\pi(\mathbf{h}')^{-1}))$ where π is the natural inclusion map $L \rightarrow K$.

Requirements. *In conclusion, for this attack we need to be able to consider the problem in a number field K with a subfield L for which $g \in L$. Further, the infinity norm of g should be exponential in the degree of K .*

Determinant attack

In the specific case where the module structure introduces square matrices over the ciphertext space R_g , the determinant map $\det : R_g^{m \times m} \rightarrow R_g$ provides a similar norming down function to the trace and norm. In this case, there is no restriction on g as we always have

$$\mathbf{v} \equiv \mathbf{h}\mathbf{u} \pmod{gR^{m \times m}} \Rightarrow \det(\mathbf{v}) \equiv \det(\mathbf{h})\det(\mathbf{u}) \pmod{gR}.$$

However, with the determinant map the size of elements blows up exponentially in m , and so the attack will only be applicable for very small m or an exponentially large infinity norm $|g|_\infty$. Further, there is no simple way to find a solution to the original problem from only $\det(\mathbf{u})$ and $\det(\mathbf{v})$ meaning this attack can only be applied to try to solve a decisional version of the Ideal-NTRU problem.

5 Attacks on the LWE family of problems

5.1 A brute force attack on the LWE secret

The most naïve approach to solving the search LWE problem $\text{Ideal-LWE}_{X,q,m,1,\ell,\chi}$ is to try to guess the secret \mathbf{s} . If we guess that the secret is \mathbf{s}' we can compute the value $\mathbf{c} = \mathbf{b} - \mathbf{a}\mathbf{s}'$. If $\mathbf{s}' = \mathbf{s}$ then the components of \mathbf{c} will be samples from the LWE error distribution χ , otherwise they will be essentially uniformly random values modulo q . With enough samples one can determine which is the case with

any desired probability. By searching over all possible \mathbf{s}' until we find one where the components of \mathbf{c} are suitably distributed we can solve the LWE problem. This approach is a brute force approach.

One can perform a more intelligent brute force attack by first using m samples to convert the problem to normal form as described in Section 2.8. Now, one has a much smaller search space for the possible secret.

Albrecht, Player and Scott [7] give the following result on the time and memory complexity of this exhaustive attack. Here t is chosen so that the error is bounded by $t\alpha q$ with overwhelming probability and $\mathcal{D}_{\mathbb{Z},\alpha}$ is the discretised Gaussian with parameter α .

Theorem 3 (Theorem 5.1 from [7]). *The time complexity of solving the Ideal-LWE $_{X,q,m,1,\ell,\mathcal{D}_{\mathbb{Z},\alpha}}$ with success probability ϵ using an exhaustive search is $\ell \cdot (2t\alpha q + 1)^m \cdot 2m$ and the memory complexity is m when $\ell \geq m + m'$, $t = \omega(\sqrt{\log m})$ and*

$$m' = \frac{\log(1 - \epsilon) - m \log(2t\alpha q + 1)}{\log(2t\alpha)}.$$

It is clear that exactly the same approach can be carried out for the general Ideal-LWE $_{f,g,m,k,\ell,\chi}$ problem as long as the distribution χ can be efficiently distinguished from uniform when reduced modulo g .

5.2 A meet-in-the-middle attack on sparse secret LWE

Adapting the idea of Odlyzko's meet-in-the-middle attack on the NTRU problem described in Section 4.3, Cheon, Hhan, Hong and Son [28] give a meet-in-the-middle attack on the LWE problem when the secret is both sparse and has ternary entries, that is from $\{-1, 0, 1\}$. We note that Bai and Galbraith [16] mentioned the existence of such a meet-in-the-middle attack on LWE but did not give details however they did state that the attack requires $\tilde{O}(3^{n/2})$ space and time.

The basic idea is to assume that the secret vector \mathbf{s} has Hamming weight at most h and to split it in some way as $\mathbf{s} = \mathbf{s}_1 + \mathbf{s}_2$ where both parts have Hamming weight at most $h/2$. One then has the approximate equality, when considered modulo q ,

$$\mathbf{as}_1 \approx \mathbf{b} - \mathbf{as}_2$$

since $\mathbf{b} \equiv \mathbf{as} + \mathbf{e} \pmod{q}$. Further, suppose that the entries of \mathbf{e} are all bounded by B . Naïvely, one can first list all possible values of \mathbf{as}_1 as \mathbf{s}_1 varies through all possible ternary vectors of Hamming weight at most $h/2$ and then compute $\mathbf{b} - \mathbf{as}_2$ for each possible \mathbf{s}_2 until one finds an approximate collision on the list; that is the coordinates differ by at most B .

Rather than this naïve approach, Cheon et al. apply the same techniques described in Section 4.3 to sort the list into 2^ℓ buckets based on the most significant bit of each entry. Now, instead of checking all items on the list in the second step, one just needs to check a limited number of buckets; the only adaptation

needed is that one must potentially check many more buckets as one must accommodate coordinates differing by up to B rather than simply by 1. Further, they suggest splitting up the secret in a potentially unbalanced way; namely, into \mathbf{s}_1 of Hamming weight at most h_1 and \mathbf{s}_2 of Hamming weight at most h_2 with $h = h_1 + h_2$.

As written above, the attack is against the search version of the LWE problem, however Cheon et al. consider it as an attack on the decision problem by deciding the samples are uniformly random if no secret \mathbf{s}_2 can be found which creates an approximate collision with any of the \mathbf{as}_1 .

Generalisation

One can readily adapt this approach to work more generally. One assumes that the secret is distributed according to a distribution of small elements (not necessarily the same as the errors); if not one first applies the transformation to normal form given in Section 2.8. Again, the attack then follows the same approach as the meet-in-the-middle attack on Ideal-NTRU but instead we are looking for an approximate collision between $\mathbf{b} - \mathbf{as}_2$ and \mathbf{as}_1 such that $\mathbf{s}_1 + \mathbf{s}_2 = \mathbf{s}$. Again, we will assume \mathbf{s}_1 and \mathbf{s}_2 lie in two sets S_1 and S_2 and we try to find the smallest possible choices for S_1 and S_2 which allow such a splitting of an arbitrary secret.

Typically, this is done by only allowing certain module coordinates to be non-zero in the case where the module dimension m is much larger than one, or else allowing only certain coefficients of elements from R to be non-zero.

One small difference arises when considering the Ideal-LWE problem instead of the Ideal-NTRU problem. Firstly, one need not compute \mathbf{as}_1 using the full element \mathbf{a} . Instead one can choose a smaller ℓ' with $1 \leq \ell' \leq \ell$ such that after dropping the last $\ell - \ell'$ rows of \mathbf{a} and \mathbf{b} the secret is still (almost) unique with high probability. If one does find multiple possible secrets one can use the remaining samples to check whether each one is valid or a spurious solution.

As before, the main obstruction which could stop this attack from working is if there is no efficient way to test for an approximate collision which would be the case if there is no efficient test of smallness in R_g .

In general, assuming that one can efficiently test for approximate collisions, if S is the set of all possible secrets (with high probability) then the meet-in-the-middle attack takes $\tilde{O}(\sqrt{|S|})$ time and memory in the classical setting and $\tilde{O}(\sqrt[3]{|S|})$ time and memory in the quantum setting due to Grover's algorithm.

Requirements. *This approach requires an efficient method of determining approximate collisions. Further, when $m = 1$ the attack works best when R has symmetries which allow the sets S_1 and S_2 to be chosen to be smaller than without such symmetries.*

5.3 Reducing LWE to BDD: The primal attack

One can view the Ideal-LWE $_{X,q,m,1,\ell,\chi}$ problem as the bounded distance decoding (BDD $_\gamma$) problem on ℓ -dimensional integer q -ary lattices of the form⁶

$$\Lambda_q(\mathbf{a}) := \{ \mathbf{z} \in \mathbb{Z}^\ell \mid \mathbf{z}^T \equiv \mathbf{a}\mathbf{s}^T \pmod{q} \text{ for some } \mathbf{s} \in \mathbb{Z}_q^m \}$$

with target vector (any lift of) \mathbf{b} . Here, the approximation factor γ defining the problem depends on the choice of χ . This assumes that one can find n linearly independent rows of \mathbf{a} so that one can recover \mathbf{s} from $\mathbf{a}\mathbf{s}$, if not one can only recover partial information about \mathbf{s} without more samples.

The primal attack consists of solving this bounded distance decoding problem using lattice reduction as was first suggested by Lindner and Peikert [54]. Since lattice reduction techniques strongly depend on the dimension of the lattice, which here is the number of LWE samples, it is not wise to use too many samples when constructing the so called primal lattice $\Lambda_q(\mathbf{a})$. Further, there are a number of different approaches to solving the BDD $_\gamma$ problem, perhaps the most straightforward of which is to use Babai's nearest plane algorithm [14] which takes as input a basis matrix B for a lattice and a target vector \mathbf{t} and outputs a vector \mathbf{e} such that $\mathbf{t} - \mathbf{e}$ lies in the lattice $A(B)$.

The idea of Babai's nearest plane algorithm is to recursively compute the closest vector to the target vector in the sublattice spanned by the last i basis vectors. This process can be performed in polynomial time as follows. Let $\mathbf{b}_1^*, \dots, \mathbf{b}_d^*$ be the Gram-Schmidt vectors in order of increasing length. Then setting $\mathbf{t}_d := \mathbf{t}$ one computes for i from d to 1 the vectors

$$\mathbf{t}_{i-1} := \mathbf{t}_i - \left\lfloor \frac{\langle \mathbf{t}_i, \mathbf{b}_i^* \rangle}{\langle \mathbf{b}_i^*, \mathbf{b}_i^* \rangle} \right\rfloor \mathbf{b}_i$$

and returns \mathbf{t}_0 . Denoting the fundamental parallelepiped of the lattice spanned by the Gram-Schmidt vectors by $\mathcal{P}(B^*)$, Babai gave the following result.

Lemma 2 ([15]). *Let B be a basis matrix for a lattice Λ and B^* be the corresponding Gram-Schmidt matrix. For a target vector \mathbf{t} in the span of Λ , Babai's nearest plane algorithm returns the unique vector $\mathbf{e} \in \mathcal{P}(B^*)$ such that $\mathbf{t} - \mathbf{e} \in \Lambda$.*

Clearly then, using Babai's nearest plane algorithm requires a well-reduced basis of the lattice as input if it is to be used to solve the bounded distance decoding problem. Hence, one applies a strong lattice reduction algorithm to the basis before applying this so-called decoding step. Many methods for solving the BDD problem follow this approach of first reducing the lattice and then applying some kind of decoding step, however not all do.

Lindner and Peikert [54] use a simple extension of Babai's nearest plane algorithm tailored to the known Gaussian distribution of the error typically used in the LWE problem. This approach introduces a quality/time trade-off

⁶ We note that the LWE and SIS lattices are dual to each other up to a scaling factor q , $\Lambda_q(A) = q\Lambda_q^\perp(A^T)^*$.

in decoding allowing a faster but weaker lattice reduction step at the cost of increasing the time for decoding, this however can give a lower overall running time for the attack.

The main drawback of Babai's nearest plane algorithm is that for a typical reduced basis the first few Gram-Schmidt vectors are much shorter than average and the final few much longer, thus the parallelepiped $\mathcal{P}(B^*)$ is very 'long and skinny' so the algorithm is unlikely to recover the Gaussian error in the LWE samples. To overcome this Lindner and Peikert introduce a second recursion layer which recurses over some $d_i \geq 1$ distinct planes on the i th outer recursion with the effect of making the parallelepiped wider in the direction of \mathbf{b}_i^* by a factor of d_i . One should then choose the d_i which maximises $\min_i(d_i \|\mathbf{b}_i^*\|)$ so as to capture the most probability mass of the error distribution. One can see this as trying the d_i closest integers to $\langle \mathbf{t}_i, \mathbf{b}_i^* \rangle / \langle \mathbf{b}_i^*, \mathbf{b}_i^* \rangle$ in Babai's nearest plane algorithm rather than simply only the closest. One then has the following lemma.

Lemma 3 (Lemma 4 from [54]). *For $\mathbf{t} \in \text{Span}(B)$, the modified nearest plane algorithm returns the set of all $\mathbf{v} \in \Lambda(B)$ such that $\mathbf{t} \in \mathbf{v} + \mathcal{P}(DB^*)$ where D is the diagonal matrix with diagonal d_i . The running time is essentially $\prod_i d_i$ times that of Babai's nearest plane algorithm.*

On the assumption that the discrete Gaussian error $\mathcal{D}_{\mathbb{Z}, \alpha}$ used has large enough width α , the success probability of the modified nearest plane algorithm is very close to $\prod_i \text{erf}(d_i \|\mathbf{b}_i^*\| \sqrt{\pi}/(2\alpha q))$.

Returning to the lattice reduction phase, Lindner and Peikert employ BKZ reduction which achieves a given root Hermite factor δ_β depending on the block size β . They give the optimal dimension ℓ , corresponding to the number of LWE samples used, as $\ell = \lfloor \sqrt{m \log q / \log \delta_\beta} \rfloor$.

Improvements to this approach have been suggested by Liu and Nguyen [55] using (pruned) enumeration, and further by Aono et al. [11]. However, a generalised framework was proposed by Herold et al. [44] which encompassed all such enumeration techniques for decoding and showed that asymptotically they achieve the same running time.

Alternatively, one can use the embedding technique of Kannan [50] to reduce the BDD_γ problem to an instance of the unique shortest vector problem, $\text{uSVP}_{\gamma'}$, as was done by Lyubashevsky and Micciancio [56]. Here, one embeds the target vector \mathbf{t} together with the original lattice $\Lambda(B)$ in a higher dimensional lattice with basis matrix

$$\begin{pmatrix} B & \mathbf{0} \\ \mathbf{t} & t \end{pmatrix}$$

for some embedding factor t .

This approach to solving the BDD problem in the context of the LWE problem was considered by Albrecht et al. in [6]. They give the following lemma on the gap between the shortest non-zero vector and the second lattice minimum.

Lemma 4 (Lemma 2 from [6]). *Let $\mathbf{a} \in \mathbb{Z}_q^{\ell \times m}$, $\alpha > 0$ and let $c > 1$. Further, let \mathbf{e} be drawn from the discrete Gaussian $\mathcal{D}_{\mathbb{Z}^\ell, \alpha q}$. Under the assumption that the*

shortest vector of $\Lambda_q(\mathbf{a})$ is at least as large as predicted by the Gaussian heuristic and the assumption that the columns of \mathbf{a} are linearly independent over \mathbb{Z}_q , one can create an embedded lattice with λ_2/λ_1 -gap greater than

$$\frac{\min \left\{ q, \frac{q^{1-m/\ell} \Gamma(1+\ell/2)^{1/\ell}}{\sqrt{\pi}} \right\}}{\frac{c\alpha q \sqrt{\ell}}{\sqrt{\pi}}} \approx \frac{\min \left\{ q, q^{1-m/\ell} \sqrt{\frac{\ell}{2\pi e}} \right\}}{\frac{c\alpha q \sqrt{\ell}}{\sqrt{\pi}}}$$

with probability greater than $1 - (c \cdot \exp((1 - c^2)/2))^\ell$.

When applying a lattice reduction algorithm achieving root Hermite factor δ_0 , the experimental results of Albrecht et al. match those of Gama and Nyugen [38] whereby the vector $\pm(\mathbf{e}, t)$ lies in the reduced basis with some fixed probability whenever the gap satisfies

$$\frac{\lambda_2}{\lambda_1} \geq \tau_t \delta_0^\ell$$

for some real constant $0 < \tau_t \leq 1$ depending on the desired probability level. Experimentally, they found that $\tau_{\|\mathbf{e}\|} \geq 0.4$ is needed for a success probability of 0.1 with the parameters of Regev [65], depending on the algorithm used. Albrecht et al. also determine that a value of $\ell = \left\lfloor \sqrt{m \log q / \log \delta_\beta} \right\rfloor$ is optimal in this case too.

Determining the optimal choice for t does not appear to be a simple task. One choice proposed by Lyubashevsky and Micciancio [56] is $t = \text{Dist}(A(B), \mathbf{t})$ so that the new lattice contains a vector of length $\sqrt{2}t$. Although this value is not known exactly, it can be approximated. According to Albrecht et al., for smaller values of t it is difficult to determine the gap λ_2/λ_1 . The choice $t = 1$ has been found to be more efficient in practice giving a value of $\tau_1 \approx 0.3$ [6].

One concludes with the following result.

Lemma 5 (Lemma 5.18 from [7]). *Any lattice reduction algorithm achieving log root Hermite factor*

$$\log \delta_0 = \frac{\log^2(c\tau\alpha\sqrt{2e})}{4m \log q}$$

can be used to solve the Ideal-LWE $_{X,q,m,1,\ell,\mathcal{D}_{\mathbb{Z},\alpha}}$ problem via reduction to uSVP with success probability greater than $\epsilon_\tau \cdot (1 - (c \cdot \exp((1 - c^2)/2))^\ell)$ for some $c > 1$ and some fixed $\tau \leq 1$ and $0 < \epsilon_\tau < 1$ as a function of τ .

Generalisation

We can straightforwardly generalise the LWE lattice to

$$\Lambda(\mathbf{a}) := \{\iota(\mathbf{r}) \in \mathbb{Z}^{k\ell n} \mid \mathbf{r} \in R^{\ell \times k}, \mathbf{r} \equiv \mathbf{a}\mathbf{t} \pmod{gR} \text{ for some } \mathbf{t} \in R_g^{m \times k}\}$$

and note that the target vector $\iota(\text{rep}_g(\mathbf{b}))$ is somewhat close to a point in this lattice. Indeed if \mathbf{e} is not too large we again have an instance of the bounded

distance decoding problem. With high probability $\Lambda(\mathbf{a})$ is full rank and has volume $a^{\deg(r)k(\ell-m)}$.

We can perform lattice reduction on $\Lambda(\mathbf{a})$ to find a reduced basis of the lattice and with this basis we can perform a *decoding step* in an attempt to find a lattice vector close to the target vector as before.

The constructed lattice will have k orthogonal components corresponding to the k independent columns of the secret matrix \mathbf{s} so we can independently perform lattice reduction on each orthogonal component. This is equivalent to the fact that we can view $\mathbf{b} = \mathbf{a}\mathbf{s} + \mathbf{e}$ as k distinct instances of Ideal-LWE (with $k = 1$) which have the same public matrix \mathbf{a} . Thus without loss of generality for security, we can consider the case $k = 1$ only so that \mathbf{s} and \mathbf{e} are vectors having entries in R_g and R respectively.

To be able to recover the secret by solving the bounded distance decoding instance we require that the norm of the error vector is not much larger than the length of the shortest vector in $\Lambda(\mathbf{a})$. This is true in the standard settings of LWE and polynomial LWE but we can see immediately that if g is a polynomial with small coefficients, or indeed any of the polynomials in gR has only small coefficients, then the shortest vector in $\Lambda(\mathbf{a})$ may be much smaller than the norm of the error vector, essentially meaning that in practice such a lattice attack is futile. This is not immediately evident though since lattice basis reduction outputs a whole basis for the lattice. Thus, in theory, it might be that one can still recover some secret information by solving the BDD instance. However, this has not been possible in practice and remains merely speculative.

Requirements. *For the primal attack to work on the Ideal-LWE $_{f,g,m,1,\ell,\chi}$ problem, and hence also on the Ideal-LWE $_{f,g,m,k,\ell,\chi}$ problem, when using the embedding technique with embedding constant 1 we require:*

$$\begin{aligned} \sqrt{\ell n \sigma^2 + 1} &< \sqrt{(\ell n + 1)/2\pi e} \cdot a^{\deg(r)(\ell-m)/(\ell n + 1)} \\ \sqrt{\ell n \sigma^2 + 1} &< \min\{\|\mathbf{x}\| \mid \mathbf{x} \in gR \setminus \{0\}\} \end{aligned}$$

where σ is the standard deviation of the distribution χ . The conditions come from the fact that we need the vector we want to find to have norm smaller than the Gaussian heuristic as well as that no other shorter non-zero vectors exist in the lattice.

5.4 Reducing short secret LWE to inhomogeneous SIS

When considering the short secret variant of the learning with errors problem one can enhance the primal attack as was done by Bai and Galbraith [16] where they focussed on the case when the secret has components in $\{0, 1\}$ or $\{-1, 0, 1\}$. The idea is to reduce the problem to the inhomogeneous SIS problem as we now explain.

Suppose one is given an Ideal-LWE $_{X,q,m,1,\ell,\chi}$ instance $(\mathbf{a}, \mathbf{b}) \in \mathbb{Z}_q^{\ell \times m} \times \mathbb{Z}_q^{\ell \times 1}$ in which the secret $\mathbf{s} \in \mathbb{Z}^{m \times 1}$ is short. Define $\ell' := \ell + m$, $\mathbf{a}' := (\mathbf{a} \ I_\ell)$ and $\mathbf{z} := (\mathbf{s}^T \ \mathbf{e}^T)$ where $\mathbf{e} \equiv \mathbf{b} - \mathbf{a}\mathbf{s} \pmod{q}$ is the short error vector. Then $\mathbf{a}'\mathbf{z}^T \equiv \mathbf{b} \pmod{q}$

is an instance of the inhomogeneous Ideal-SIS $_{X,q,\ell,\ell',\rho}$ problem with target vector \mathbf{b} and ρ an upper bound on $\|\mathbf{z}\|$ (with high probability).

The standard approach to solving the inhomogeneous version of the SIS problem is to find one solution \mathbf{w} without consideration of its size, for example in this case $\mathbf{w} = (\mathbf{0} \ \mathbf{b}^T)$, and then attempt to solve the BDD $_{\rho}$ problem in the associated SIS lattice $\Lambda_q^{\perp}(\mathbf{a}')$ with target vector \mathbf{w} . On finding a vector $\mathbf{v} \approx \mathbf{w}$ with $\mathbf{a}'\mathbf{v}^T \equiv \mathbf{0}^T \pmod{q}$, we note that $\mathbf{z} = \mathbf{w} - \mathbf{v}$ is short and satisfies $\mathbf{a}'\mathbf{z}^T \equiv \mathbf{b} \pmod{q}$ as required. One hopes that the first m coordinates of $-\mathbf{v}$ are indeed the LWE secret \mathbf{s} .

To solve the bounded distance decoding problem one can use any of the approaches given in the preceding section discussing the primal attack. Bai and Galbraith use the embedding technique with embedding constant $t = 1$.

If one has $\|\mathbf{s}\| \ll \|\mathbf{e}\|$, Bai and Galbraith suggest multiplying the first m columns of $\Lambda_q^{\perp}(\mathbf{a}')$ by a scalar μ to balance the vector $\mathbf{w} - \mathbf{v}$ before solving the closest vector problem (the target vector \mathbf{w} does not need to change as it is zero in these coordinates). A further trick suggested when the secret has binary coefficients is to change the target vector to $(-\frac{1}{2}\mu\mathbf{1}_m \ \mathbf{b}^T)$ where $\mathbf{1}_m$ is the vector of all 1s of length m . In this manner the difference $\mathbf{w} - \mathbf{v} = (\pm\frac{1}{2}\mu \cdots \pm\frac{1}{2}\mu \ e_1 \cdots e_m)$ is more balanced.

Generalisation

This reduction works more generally and allows one to reduce short secret Ideal-LWE $_{f,g,m,k,\ell,\chi}$ to k instances of the inhomogeneous Ideal-SIS $_{f,g,\ell,m+\ell,\rho}$ problem which all have the same matrix $A = (\mathbf{a} \ I_{\ell})$ defining the problem but with differing target vector \mathbf{t} , namely the rows of \mathbf{b}^T , and solution vector \mathbf{z} the corresponding row of $(\mathbf{s}^T \ \mathbf{e}^T)$; hence ρ should be taken as an upper bound on the size of such vectors. One important distinction must be made however in that while the inhomogeneous Ideal-SIS problem allows any solution which satisfies the bound ρ , we are looking for a particular solution in order to solve the short secret Ideal-LWE problem which may be an issue for certain choices of g .

For example, if using the embedding technique as mentioned above, then one must consider the lattice

$$\Lambda(\mathbf{a}, \mathbf{b}) = \{(\iota(\mathbf{x}), \iota(\mathbf{y}), z) \in \mathbb{Z}^d \mid \mathbf{x} \in R^m, \ \mathbf{y} \in R^{\ell}, \ \mathbf{a}\mathbf{x}^T + \mathbf{y}^T \equiv z\mathbf{b}^T \pmod{gR}\},$$

where $d := (\ell + m)n + 1$ and attempt to find the short lattice vector $(\iota(\mathbf{s}), \iota(\mathbf{e}), 1)$. We can construct this lattice by computing a spanning set $\{\mathbf{v}_i\}_{i=0}^{mn}$ of the solution space to $\mathbf{a}\mathbf{x} + \mathbf{y} - z\mathbf{b} = \mathbf{0}$ over R . Namely, we set $\mathbf{v}_0 = (\mathbf{0}, \iota(\text{rep}_g(\mathbf{b})), 1)$ and the other vectors $\mathbf{v}_1, \dots, \mathbf{v}_{mn}$ as $(\iota(\mathbf{p}_{i,j}), \iota(-\mathbf{a}\mathbf{p}_{i,j}^T), 0)$ for $i = 1, \dots, m$ and $j = 1, \dots, n$, where $\mathbf{p}_{i,j}$ is the power-basis for the i -th copy of R in R^m . Then the lattice is spanned by the rows of the block-matrix

$$\left(\begin{array}{cccc} & \mathbf{v}_0 & & \\ & \vdots & & \\ & \mathbf{v}_{mn} & & \\ \mathbf{G} & & 0 & \\ & \mathbf{G} & & 0 \\ & & \ddots & \vdots \\ & & & \mathbf{G} \\ & & & 0 \end{array} \right) \in \mathbb{Z}^{(d+mn) \times d}, \quad \mathbf{G} := \begin{pmatrix} \iota(g) \\ \iota(Xg) \\ \vdots \\ \iota(X^{n-1}g) \end{pmatrix} \in \mathbb{Z}^{n \times n}. \quad (1)$$

It is clear that the vector $(\iota(\mathbf{s}), \iota(\mathbf{e}), 1)$ lies in this lattice and that it is rather short. However, the important fact for this approach to work isn't that this vector is short but again that it is shorter than all the other non-zero vectors in the lattice $\Lambda(\mathbf{a}, \mathbf{b})$, which is not necessarily the case. This issue is something that also other methods for solving the inhomogeneous Ideal-SIS problem must contend with in this setting.

The main example of this is with the MLHC problem in which the lattice (1) contains many vectors of Euclidean length $\sqrt{5}$; this will be much smaller than the length of an element sampled from the error distribution used in practice. A similar phenomenon is observed when considering the LPN problem (LWE with modulus 2); lattice reduction fails because there are many vectors of norm 2. Other approaches are required to attack these sorts of problems.

Again, one may wish to try to balance the entries of the solution vector if the secret and error distributions are different in the original Ideal-LWE instance.

5.5 Reducing LWE to SIS: The dual attack

The idea of the dual attack is to reduce the problem of solving the decision LWE problem to solving the SIS problem as was suggested by Micciancio and Regev [62]. The approach is to find a short vector in the (scaled) lattice dual to the primal lattice $\Lambda_q(\mathbf{a})$ which we noted is the SIS lattice $\Lambda_q^\perp(\mathbf{a}^T)$.

Suppose that one has found a short vector $\mathbf{v} \in \Lambda_q^\perp(\mathbf{a}^T)$ so that $\mathbf{v}\mathbf{a} \equiv \mathbf{0} \pmod{q}$, the attack proceeds by taking the inner product of \mathbf{v} with the vector \mathbf{b}^T . If we are in the case that the samples came from the LWE distribution $\mathcal{A}_{\mathbf{s}, \chi}$ then

$$\langle \mathbf{v}, \mathbf{b}^T \rangle \equiv \mathbf{v}\mathbf{a}\mathbf{s} + \langle \mathbf{v}, \mathbf{e}^T \rangle \equiv \langle \mathbf{v}, \mathbf{e}^T \rangle \pmod{q}$$

which is the inner product between two short vectors and so is also relatively small. If however, the samples were uniformly random and assuming the greatest common divisor of the coordinates of \mathbf{v} together with q is one, the inner product will also be uniformly random modulo q . Distinguishing between these two distributions therefore allows one to attack the decision LWE problem.

In [62], the authors suggest that when χ is a discrete Gaussian with parameter α , $\mathcal{D}_{\mathbb{Z}, \alpha}$, then if one can only find a vector \mathbf{v} with $\|\mathbf{v}\| \geq 1.5\sqrt{2\pi}/\alpha$ then the two distributions are within negligible statistical distance of one another. On the other hand and in the same setting, Lindner and Peikert [54]

state that when $\|\mathbf{v}\|$ is not much larger than $1/\alpha$ then the advantage in distinguishing is very close to $\exp(-\pi(\|\mathbf{v}\|\alpha)^2)$. Hence, to use this approach to solve Ideal-DLWE $_{X,q,m,1,\ell,\mathcal{D}_{z,\alpha}}$ with advantage ϵ one must solve the Ideal-SIS $_{X,q,m,\ell,\rho}$ problem for $\rho = \frac{1}{\alpha}\sqrt{\ln(1/\epsilon)/\pi}$.

If one uses lattice reduction to solve the SIS instance then typically one finds a number of short vectors rather than just one. Naturally, having many relatively short vectors \mathbf{v}_i such that $\mathbf{v}_i\mathbf{a} \equiv \mathbf{0} \pmod q$ increases the chance of being able to distinguish between the uniform distribution and the LWE distribution by using the Chernoff bound [30].

It has been suggested by Albrecht [4] that when using lattice reduction it may be possible to amortize its cost to find multiple short vectors in the dual lattice by first performing one strong lattice reduction step, noting the shortest vector and then re-randomising the reduced basis by a sparse unimodular matrix and running a cheaper reduction algorithm to recover another short vector, repeating this process a number of times gives multiple short vectors which experimentally enable a better advantage in distinguishing.

Generalisation

Suppose we are given the pair $(\mathbf{a}, \mathbf{b}) \in R_g^{\ell \times m} \times R_g^{\ell \times k}$ then we again aim to find a *short* vector $\mathbf{v} \in R^\ell$ such that $\mathbf{v}\mathbf{a} \equiv \mathbf{0} \pmod{gR}$. Then we have that if (\mathbf{a}, \mathbf{b}) is from the Ideal-LWE distribution then $\mathbf{v}\mathbf{b} \equiv \mathbf{v}\mathbf{a}\mathbf{s} + \mathbf{v}\mathbf{e} \equiv \mathbf{v}\mathbf{e} \pmod{gR}$ which has coefficients that are the inner products of two short vectors. Again, we can assume $k = 1$ so we do. If \mathbf{v} is short enough this should be distinguishable from $\mathbf{v}\mathbf{b}$ with \mathbf{b} a random matrix. This can be rephrased as finding a short vector \mathbf{v} in the scaled dual lattice of \mathbf{a} :

$$A^\perp(\mathbf{a}) = \{\iota(\mathbf{x}) \in \mathbb{Z}^{\ell n} \mid \mathbf{x} \in R^\ell, \mathbf{x}\mathbf{a} \equiv \mathbf{0} \pmod{gR}\}.$$

To find a basis for this lattice, see Section 3.1 where we consider the general lattice attack on the Ideal-SIS problem. This time, with high probability, the lattice has full rank ℓn and volume $a^{m \deg r}$. We remark that for there to be non-trivial vectors in this lattice we require $\ell > m$ which we already assume to be the case in the definition of the problem.

As with the primal attack, if the secret \mathbf{s} is known to be sampled from a distribution of small elements then this information can be used to enhance the dual attack [16]. We can construct the lattice

$$A'_\lambda(\mathbf{a}) = \{(\lambda\iota(\mathbf{x}), \iota(\mathbf{y})) \in \mathbb{R}^{\ell n} \times \mathbb{Z}^{mn} \mid \mathbf{x} \in R^\ell, \mathbf{y} \in R^m, \mathbf{x}\mathbf{a} \equiv \mathbf{y} \pmod{gR}\}$$

for a real scalar λ and find a short non-zero vector $(\lambda\iota(\mathbf{v}), \iota(\mathbf{w}))$ in this lattice. If (\mathbf{a}, \mathbf{b}) is taken from the Ideal-LWE distribution then we have $\mathbf{v}\mathbf{b} \equiv \mathbf{v}\mathbf{a}\mathbf{s} + \mathbf{v}\mathbf{e} \equiv \lambda\mathbf{w}\mathbf{s} + \mathbf{v}\mathbf{e} \pmod{gR}$. One chooses λ to balance the size of $\lambda\mathbf{s}$ and \mathbf{e} so that we have the sum of two products of small elements of roughly the same size. Alternatively, if \mathbf{b} is uniformly random then $\mathbf{v}\mathbf{b}$ is uniformly random over $\mathbf{v}R_g^\ell$.

Just as in the primal attack, if gR contains short vectors then the lattice $A^\perp(\mathbf{a})$ (or $A'_\lambda(\mathbf{a})$) will contain short trivial vectors. However, unlike in the primal

attack this does not immediately cause the attack to fail as we are not interested in finding a specific vector, only a short enough non-trivial vector (that is non-zero when considered as an element of R_g^ℓ). Since these non-trivial vectors must exist, it is just a matter of running a strong enough lattice reduction algorithm to find them. We found that running BKZ can find non-trivial vectors of length roughly $\delta_0^{\ell n} a^{m \deg(r)/\ell n}$, where δ_0 depends on the block size used as given in [27].

In the general case, we observed that the coefficients of \mathbf{v} roughly follow a (discretised) Gaussian distribution centred about zero so its norm is approximately Chi distributed. If we denote the standard deviation of the distribution of the coefficients of the vectors found by solving the Ideal-SIS problem by σ_{SIS} then, together with the approximation for the length of the short vector we can find, we have

$$\sqrt{\ell n} \sigma_{\text{SIS}} \approx \sqrt{2} \frac{\Gamma((\ell n + 1)/2)}{\Gamma(\ell n/2)} \sigma_{\text{SIS}} \approx \delta_0^{\ell n} a^{m \deg(r)/\ell n}.$$

Suppose (\mathbf{a}, \mathbf{b}) is from the Ideal-LWE distribution so that $\mathbf{v}\mathbf{b} \equiv \mathbf{v}\mathbf{e} \pmod{gR}$, we first consider $\mathbf{v}\mathbf{e}$ in R as $\sum_{i=1}^{\ell} \mathbf{v}_i \mathbf{e}_i$ where $\mathbf{v}_i, \mathbf{e}_i \in R$ with \mathbf{e}_i sampled from the error distribution χ . Suppose for simplicity that the coefficients of elements sampled from χ have variance σ_χ^2 and further define the constants $f_{\alpha, \beta, \gamma}$, for $0 \leq \alpha, \beta, \gamma < n$ by

$$X^\alpha \cdot X^\beta \equiv \sum_{\gamma=0}^{n-1} f_{\alpha, \beta, \gamma} X^\gamma \pmod{f},$$

then

$$\mathbf{v}\mathbf{e} = \sum_{\gamma=0}^{n-1} \left(\sum_{\alpha=0}^{n-1} \sum_{\beta=0}^{n-1} \left(\sum_{i=1}^{\ell} v_{i, \alpha} e_{i, \beta} \right) f_{\alpha, \beta, \gamma} \right) X^\gamma$$

where $v_{i, \alpha}$ are the coefficients of \mathbf{v}_i and $e_{i, \beta}$ are the coefficients of \mathbf{e}_i . For f of cryptographic interest, by which we mean having small coefficients so that the $f_{\alpha, \beta, \gamma}$ are small, we find the coefficient of X^γ of $\mathbf{v}\mathbf{e}$ computed over R is approximately normally distributed with zero mean and variance $\sigma_\gamma^2 := \ell \sigma_{\text{SIS}}^2 \sigma_\chi^2 \sum_{\alpha, \beta=0}^{n-1} |f_{\alpha, \beta, \gamma}|$. The analysis assumes that the standard deviations are large enough that discrete Gaussians behave like their continuous counterparts.

However, we can only compute $\mathbf{v}\mathbf{b}$ modulo gR . Since R_g has $a^{\deg r}$ distinct elements, if $\prod_{\gamma=0}^{n-1} \sigma_\gamma$ is much larger than this value we do not expect to be able to distinguish using $\mathbf{v}\mathbf{b} \pmod{gR}$. In the case $n = 1$ of LWE this reduces to the same remark of Lindner and Peikert [54] that $\|\mathbf{v}\|$ should not be much larger than g/σ_χ . In their case they can conclude by considering the statistical distance that the advantage of distinguishing is very close to $\exp(-2\pi^2 \sigma_\chi^2 / g^2)$. In our more general case we can do the same although we were unable to find a formula for the advantage in the general case due to its intricate dependence on the shape of the ciphertext modulus g .

Example 2. As a simple yet important example, consider the case when $f = X^n \pm 1$; then

$$|f_{\alpha,\beta,\gamma}| = \begin{cases} 1 & \text{if } \alpha + \beta \equiv \gamma \pmod{n} \\ 0 & \text{otherwise} \end{cases},$$

and hence each coefficient of \mathbf{ve} is approximately normal with variance $\ell n \sigma_{\text{SIS}}^2 \sigma_{\chi}^2$. In the case that g is a positive integer, the advantage of distinguishing can be considered coefficient wise so that it is close to $\exp(-2\pi^2 \ell n \sigma_{\text{SIS}}^2 \sigma_{\chi}^2 / g^2)$ per coefficient when $\sqrt{2\pi} \ell n \sigma_{\text{SIS}}^2 \sigma_{\chi}^2$ is not much larger than g . With the integer version of Ring-LWE [41] where $g = X - q$ we again have a similar result for q not much smaller than $\sqrt{2\pi} \|\mathbf{v}\| \sigma_{\chi}$, one must essentially distinguish the uniform distribution on \mathbb{Z}_q^n from the spherical discrete Gaussian distribution of variance roughly $\ell n \|\mathbf{v}\|^2 \sigma_{\chi}^2$ and the advantage will be essentially the same.

Of course, having multiple short non-trivial vectors can increase the advantage of distinguishing.

Requirements. For the dual attack on the Ideal-LWE $_{f,g,m,k,\ell,\chi}$ problem using a lattice reduction algorithm achieving root Hermite factor δ_0 one must have

$$\delta_0^{\ell n^2} \sigma_{\chi}^n n^{-n/2} \prod_{\gamma=0}^{n-1} \sqrt{\sum_{\alpha,\beta=0}^{n-1} |f_{\alpha,\beta,\gamma}|} \gg a^{(1-m/\ell) \deg r}.$$

This condition is very dependent on the shape of both f and g , the attack is most feasible when f is sparse and/or has small coefficients and at least one of the coefficients of g are large.

5.6 The Blum-Kalai-Wasserman algorithm

In [21] Blum, Kalai and Wasserman introduced an algorithm which solves the learning parity with noise problem using a slightly subexponential time and number of samples. This algorithm came to be known as the BKW algorithm. Regev [66], noted that one can adapt the BKW algorithm to work against the LWE problem but it requires $2^{O(m)}$ time and samples. We remark that in the case of LWE one can use the sample amplification technique of Herold et al. [44] to increase the number of samples one has available if required.

In its simplest form, the approach is somewhat similar to the dual attack in that one finds a short vector in the dual lattice of A though this time this vector will be a ternary vector of length $\sqrt{2^t}$ for some chosen t , having entries in $\{-1, 0, 1\}$, and so we will require many more samples; one can compare this with the attack of Camion, Patarin and Wagner in Section 3.3. As with the dual attack, this will attempt to solve the decisional variant of the LWE problem.

Instead of using lattice reduction to find such a vector in the dual lattice, the BKW algorithm splits the dimension m of the first component of the LWE samples into t blocks of length roughly m/t and proceeds iteratively block by block computing, at each stage $i = 1, \dots, t$, samples whose first components are

zero on the first i blocks and which are a signed sum of 2^i original samples. One thus ends up with samples of the form $(\mathbf{0}, \mathbf{b}')$ where \mathbf{b}' is equal to either the signed sum of 2^i samples from the LWE error distribution χ or is uniformly random.

During each step, one simply considers samples produced in the previous step, say that have first component having zeros in the first i blocks, and looks for two samples which agree on the $(i + 1)$ th block.⁷ Upon finding such a pair, subtracting one sample from the other component-wise produces a sample of the required form. If no such collision is found, more samples are requested and processed until such a collision is found. Thus, the approach can be seen as constructing $t + 1$ sorted lists \mathcal{L}_i of samples, indexed from 0 to t , all of which are initially empty and for each sample we proceed through each list modifying the sample until we can insert it into a list for which no sample in list \mathcal{L}_i matches the (modified) sample on block i . The modifications come when a match is found in a previous list \mathcal{L}_j and then one subtracts the sample found in that list from the modified sample so that the newly modified sample has first component which is zero on the j th block. All samples which have not been inserted into a list by \mathcal{L}_{t-1} are inserted into \mathcal{L}_t and when there are enough samples in \mathcal{L}_t one stops. This can be seen as analogous to performing Gaussian elimination on the rows of the matrix \mathbf{a} but instead of considering each column independently we consider blocks of (roughly) m/t columns.

It can immediately be seen that such an approach requires storing t lists each containing approximately $q^{m/t}$ entries and while this can be reduced somewhat it is ultimately the dominant factor in determining the running time of this style of attack. Albrecht et al. [10] analysed the complexity of this approach for attacking the decision LWE problem. Here we give the slightly modified conclusion presented in [64] which assumes the need for more samples than the optimistic number used in [10].

Theorem 4 (Theorem 6 from [64]). *Let $0 < u \leq m$ and set $t = \lceil \frac{m}{u} \rceil$. The expected cost of using the BKW algorithm to solve the Ideal-DLWE $_{X,q,m,1,\ell,\mathcal{D}_{z,\alpha}}$ problem with success probability ϵ is*

$$\frac{q^u - 1}{2} \left(\frac{t(t-1)}{2}(m+1) - \frac{ut(t-1)}{4} \right) - \frac{u(q^u - 1)}{12} \left((t-1)^3 + \frac{3}{2}(t-1)^2 + \frac{1}{2}(t-1) \right)$$

additions/subtractions in \mathbb{Z}_q to produce the lists and

$$\epsilon \exp(2^t \pi \alpha^2) \frac{t(m+2)}{2}$$

⁷ Since changing the sign of both components of a sample gives another valid sample with the same magnitude of noise we should also look for blocks which sum to zero rather than whose difference is zero, this will be implicitly assumed throughout this whole section.

additions/subtractions in \mathbb{Z}_q to produce the new samples, all of which uses

$$t \left\lceil \frac{q^u}{2} \right\rceil + \epsilon \exp(2^t \pi \alpha^2)$$

original samples. The memory requirement is

$$\frac{q^u}{2} t \left(m + 1 - u \frac{t-1}{2} \right)$$

elements of \mathbb{Z}_q .

In order to be able to distinguish the two distributions using the same ideas in the dual attack one should choose t such that $\sqrt{2^t} \alpha q \leq q$ and hence $t \leq -2 \log \alpha$. This gives the following corollary.

Corollary 1. *The expected cost of applying the BKW algorithm to attack the Ideal-DLWE $_{X,q,m,1,\ell,\mathcal{D}_{z,\alpha}}$ problem is $O(q^{n/(-2 \log \alpha)} (-2 \log \alpha)^2 n)$ operations in \mathbb{Z}_q using $\ell \geq -2 \log \alpha \lceil q^{m/(-2 \log \alpha)} / 2 \rceil + \text{poly}(m)$ samples.*

Attacking the search variant

More generally, one can consider using the BKW algorithm to attack the search version of the problem as explained in [10]. This approach can be split into three different stages. The first stage is the iterative Gaussian elimination stage explained above, though one stops before the last iteration. The second stage is to perform a hypothesis test on a candidate for a part of the secret vector \mathbf{s} in order to recover a component of it. Finally, using this partial information a back substitution stage is performed so that one can proceed to solving a smaller instance of the problem.

Splitting up the secret vector \mathbf{s} into blocks in the same way as when splitting the first component of a sample, the hypothesis testing stage will attempt to recover in reverse order the blocks of \mathbf{s} . The back substitution phase takes this knowledge, say one concluded that blockwise the secret has final blocks s'_{i+1}, \dots, s'_t and write $\mathbf{s}'_i{}^T = (0, 0, \dots, 0, s'_{i+1}, \dots, s'_t)$, and computes for each sample (\mathbf{a}, \mathbf{b}) in list \mathcal{L}_{i-1} the new sample $(\mathbf{a}', \mathbf{b}') = (\mathbf{a}^{(i)}, \mathbf{b} - \mathbf{a}\mathbf{s}'_i \bmod q)$ where $\mathbf{a}^{(i)}$ is the vector consisting of the first i blocks of \mathbf{a} . If s'_i is a correct guess then $\mathbf{s} - \mathbf{s}'_i$ is non-zero only on the first i blocks. This means that $(\mathbf{a}', \mathbf{b}')$ is a lower dimensional sample with secret $\mathbf{s}^{(i)}$ and error a sum of 2^i outputs from the original error distribution; further, \mathbf{a}' has the property that it is zero on all but its last block. Hence the problem has been reduced to a smaller dimensional one of the same form and one can proceed to the hypothesis testing stage using these new samples.

In [10], the authors note that for the hypothesis testing stage one can simply use an exhaustive search over the part of the secret being tested since even with this approach the running time is dominated by the first stage. They score each guess using the log-likelihood ratio and take the guess with the highest score.

Alternatively, any other method which can tolerate the enlarged errors can be used in this step.

Duc, Tramèr and Vaudenay [35] replaced the log-likelihood ratio approach in the hypothesis testing stage by one using a multidimensional discrete Fourier transform and made further optimizations. They give the following analysis of the BKW algorithm when used against the search LWE problem.

Theorem 5 (Theorem 17 from [35]). *Let u and t be positive integers such that $ut = m$ and denote by C the small constant in the complexity of the fast Fourier transform computation. Further, let $0 < \epsilon < 1$ be a targeted success rate and define $\epsilon' := (1 - \epsilon)/t$. For $0 \leq j \leq t - 1$, set*

$$\ell_{j,\epsilon} := 8u \log(q/\epsilon) (1 - \pi\alpha^2)^{-2^{t-j}}.$$

The time complexity to solve the Ideal-LWE $_{X,q,m,1,\ell,\mathcal{D}_{\mathbb{Z},\alpha}}$ problem with probability at least ϵ is $c_1 + c_2 + c_3 + c_4$ where

$$c_1 = \frac{q^u - 1}{2} \left(\frac{(t-1)(t-2)}{2} (m+1) - \frac{u}{6} (t(t-1)(t-2)) \right)$$

is the number of additions in \mathbb{Z}_q to produce the lists,

$$c_2 = \sum_{j=0}^{t-1} \ell_{j,\epsilon} \frac{t-1-j}{2} (n+2)$$

is the number of additions in \mathbb{Z}_q to produce the samples required to recover all blocks of \mathbf{s} with probability ϵ ,

$$c_3 = 2 \left(\sum_{j=0}^{t-1} \ell_{j,\epsilon} \right) + Cmq^u \log q$$

is the number of operations in \mathbb{C} to prepare and compute the discrete Fourier transforms, and

$$c_4 = (t-1)(t-2)u \frac{q^u - 1}{2}$$

is the number of operations in \mathbb{Z}_q for back substitution. The number of samples required is

$$(t-1) \frac{q^u - 1}{2} + \ell_{0,\epsilon}.$$

Finally, the memory complexity in number of elements from \mathbb{Z}_q and \mathbb{C} is respectively

$$\frac{q^u - 1}{2} (t-1) \left(m+1 - u \frac{t-2}{2} \right) + \ell_{0,\epsilon} \quad \text{and} \quad q^u.$$

Further improvements The first stage was later modified by Guo, Johansson and Stankovski [42] in order to more efficiently find collisions. Instead of looking for exact collisions between two samples on a given block of the first component they relax this by using a q -ary linear code of length the given block size and consider there to be a collision when the two blocks map to the same codeword. This gives rise to some additional error which is the inner product of the actual difference on the two blocks and the corresponding secret block; hence, if the secret is not initially short one should first apply the normal form transformation given in Section 2.8. Again, one iterates the procedure over the different blocks and since additional errors arising in the first blocks increase exponentially in the later iterations one should choose different codes with decreasing rates as one runs over the different blocks. This approach is called coded-BKW. Since coded-BKW improves on the first stage of the attack which is the bottleneck of the algorithm, this approach outperforms the previous works; the exact analysis is rather technical and relies on a number of algorithm specific parameters so we do not state it here.

A similar proposal to coded-BKW was given in concurrent work by Kirchner and Fouque [51]. Instead of using coding theory to improve the first stage they generalise the approach of Albrecht et al. [8] called lazy modulus switching which was used to attack the LWE problem with a binary secret. There, a collision is taken to occur between two samples whose first components after modulus switching are equal on a given block, though one does not actually perform modulus switching until it is needed, hence the term lazy; this can be seen as requiring a collision in only the most significant bits of coefficients in the block. This latter point of view is adopted by Kirchner and Fouque who allow the number of significant bits required for a collision to decrease as one iterates over the blocks, at the same time allowing the remaining blocks to consist of a larger number of coefficients to keep the overall size of the lists roughly constant. They give the following analysis of this approach under a bound on the secret.

Theorem 6 (Theorem 4 from [51]). *Assume that the secret \mathbf{s} is such that $|s_i| \leq S$ for all $i = 1, \dots, m$ with $S \geq 2$ and define $\beta = \sqrt{m}/2/\alpha$. Assume further that $\max(\beta, \log q) = 2^{o(m/\log n)}$ and $\beta = \omega(1)$. Then one can solve the Ideal-DLWE $_{X,q,m,1,\ell,\mathcal{D}_{z,\alpha}}$ in time $2^{(m/2+o(m))/\ln(1+\log \beta/\log S)}$ for large enough ℓ .*

Finally, in the case of polynomial LWE, Stange [70] offers some methods to speed up the hypothesis testing phase and removing the need for a back-substitution phase. Since it is the iterative phase which is the bottleneck of this approach these ideas do not significantly improve the running time of the attack however they are still interesting. We discuss these ideas in Appendix A.

Generalisation

Since this attack only uses the module structure it can be applied in much the same way whenever m is reasonably large. In the other case, for example where $m = 1$ so we have polynomial LWE, we could convert the problem to an LWE

instance by considering R_g as a free \mathbb{Z}_a -module of dimension $\deg r$ however this loses sight of some of the additional ring structure that we would like to take advantage of. We will see in Appendix A how one can use this structure although this will not significantly speed up this approach.

Requirements. *One can readily apply the BKW algorithm to instances of the Ideal-LWE $_{f,g,m,k,\ell,\chi}$ problem when the module dimension m is relatively large and ℓ is very large. If m is small one may be able to use the fact that R_g is a free \mathbb{Z}_a -module of dimension $\deg r$ with which to define a block structure. If ℓ is not large enough one may be able to apply sample amplification techniques to increase it.*

5.7 The Arora-Ge attack

When the error distribution χ used in the LWE distribution $\mathcal{A}_{\mathbf{s},\chi}$ is very narrow, Arora and Ge [13] noticed that one can attack the problem by defining a system of non-linear equations in the entries of the secret vector \mathbf{s} . With enough samples one can linearise the system, and solve for the secret.

One first chooses an integer d such that the (discretised) error is bounded by d with very high probability when sampled from χ . Define the polynomial

$$P(\eta) := \eta \prod_{i=1}^d (\eta - i)(\eta + i);$$

then with very high probability we have $P(e) = 0$ when $e \leftarrow \chi$. Let $\mathbf{x} = (x_1, \dots, x_m)$ be m variables. Then for each sample $(\mathbf{a}_i, \mathbf{b}_i) \leftarrow \mathcal{A}_{\mathbf{s},\chi}$ we have a multivariate polynomial

$$p_i(\mathbf{x}) = P(\mathbf{b}_i - \langle \mathbf{a}_i, \mathbf{x} \rangle)$$

for which $p_i(\mathbf{s}) \equiv 0 \pmod q$. Define the variables $y_{\mathbf{v}}$, indexed by a vector $\mathbf{v} \in \mathbb{N}_0^m$ with $0 \leq \sum_{i=1}^m v_i \leq 2d + 1$, where $y_{\mathbf{v}} = \prod_{i=1}^m x_i^{v_i}$. In total there are $\binom{m+2d+1}{m}$ such variables so with enough samples it is likely that, when linearised, the equations $p_i(\mathbf{y}) \equiv 0 \pmod q$ are overdetermined and one can attempt to solve the linear system of equations. If all the error terms are indeed bounded by d then one can recover the secret as a solution.

As we require many samples to set up the linear system we must worry about the possibility that one of the error terms is larger than d causing the attack to fail. If one increases d even slightly then one needs significantly more samples and we have the same problem that it is now much more likely that one of the samples contains an error term larger than the new value for d . In practice then, the attack works well only when we can take d to be very small. More generally, Albrecht, Cid, Faugère and Perret give the following result after refining the analysis of this approach.

Theorem 7 (Theorem 5 from [9]). *Let $\mathcal{D}_{\mathbb{Z},\alpha}$ be the discrete Gaussian with parameter α and define $D := 8(\alpha q)^2 \log m + 1$. Denote by ω the linear algebra constant. If $D \in o(m)$ then the Arora-Ge algorithm solves the Ideal-LWE $_{X,q,m,1,\ell,\mathcal{D}_{\mathbb{Z},\alpha}}$*

problem in time complexity

$$O\left(2^{\omega D \log(m/D)} \alpha q^2 \log q\right)$$

and memory complexity

$$O\left(2^{2D \log(m/D)} \alpha q^2 \log q\right).$$

If $D \in o((\alpha q)^2 \log n)$ then the Arora-Ge algorithm solves the Ideal-LWE $_{X,q,m,1,\ell,\mathcal{D}_z,\alpha}$ problem in time complexity

$$O\left(2^{\omega m \log(D/m)} \alpha q^2 \log q\right)$$

and memory complexity

$$O\left(2^{2m \log(D/m)} \alpha q^2 \log q\right).$$

When performing the linearisation approach given by Arora and Ge one must have access to as many samples as required and if this is not the case then this approach fails. However, other approaches to solving a system of multivariate equations exist. Both Ding [34, 33] and Albrecht et al. [9] have proposed to use Gröbner bases to find a solution to the system of multivariate equations. Under the assumption that the polynomials p_i form a semi-regular system, they give the following result showing that using Gröbner bases gives an exponential speed-up over linearisation. However, it does not lead to a subexponential attack on the LWE problem.

Theorem 8 (Special case of Theorem 6 from [9]). *Let ω be the linear algebra constant and set $\alpha = \sqrt{m}/q$. There is an algorithm solving the Ideal-LWE $_{X,q,m,1,\ell,\mathcal{D}_z,\alpha}$ problem, where $\ell = \exp(\pi m/4)$, in time complexity $O\left(2^{m(2.35\omega+1.13)}\right)$ and memory complexity $O\left(2^{5.85m}\right)$ with success probability at least $1 - 2/\pi\sqrt{m}$.*

Generalisation

The attack also works against polynomial LWE, that is Ideal-LWE $_{f,q,1,1,\ell,\mathcal{X}}$, simply by considering a polynomial LWE sample as n LWE samples; however, this is best suited for error distributions that are defined coefficient-wise. For example, if the error distribution produces error polynomials with coefficient vectors of at most a fixed Hamming weight h , one can proceed as in the original attack against the learning parity with structured noise problem and consider any suitable polynomial $p(x)$, defined with respect to a full polynomial LWE sample, of degree $h + 1$ using the idea that among any $h + 1$ coefficients of the error there is at least one that is zero so their product is guaranteed to be zero. Since any set of $h + 1$ coefficients gives a distinct polynomial it makes sense to use all of them so in this case instead of defining p as a single polynomial we define it as $\binom{n}{h+1}$ polynomials each of degree $h + 1$ and depending on a single

polynomial LWE sample. We follow this latter approach of having p be a set of polynomials defined by a sample from $R_g^n \times R_g$.

Porting this attack to the generic ciphertext modulus setting presents a complication. In the simplest case, the polynomial $p(x)$ evaluates to zero only in the components of samples of χ . However, when the ciphertext modulus g is not just a constant, the small coefficients of samples from χ become intertwined when reduced modulo g . One may think of trying to reverse the operation of reduction modulo g when this is easy to do such as the case of a linear g (this amounts to expanding with respect to some integer base) however this is not a polynomial operation so is not compatible with this attack.

Instead, we must consider the coefficients of the distribution χ when reduced into $\text{Rep}(R_g)$, these coefficients will be polynomials in the original error coefficients. In the integer modulus case the polynomial p had degree $2D + 1$ because we assumed that with overwhelming probability the error coefficient can be one of only $2D + 1$ values. In the general case that g is not a constant polynomial we must typically use a larger degree for p due to the increased range of possible values a given coefficient in $\text{Rep}(R_g)$ can take. This can dramatically increase the degree of the polynomials making up p ; how much depends on how the coefficients of the original error e in R are mixed when reduced modulo g into $\text{Rep}(R_g)$.

For example, if g is linear then elements in $\text{Rep}(R_g)$ are constants so all coefficients are mixed; in this case p will consist of a single polynomial whose degree is now $(2D + 1)^n$ in the simple bounded error case. As another example we can consider $f = X^n + 1$ and $g = X^{n/2} - b$ for some b so that $a = b^2 + 1$ and $r = g$. In this case, when mapping an error from R to $\text{Rep}(R_g)$, each coefficient of the output depends only on two of the input coefficients so that we need to use a p consisting of $n/2$ polynomials of degree $(2D + 1)^2$ when the error coefficients are bounded by D .

This growth in the maximal degree of the polynomials defining p renders this attack all but useless when either D is large or the coefficients of an error term mix too much in $\text{Rep}(R_g)$. In the case of the MLHC problem, the error polynomials have a fixed Hamming weight, say h , and $g = X - 2$, this leads to the polynomial p having degree at least $\binom{n}{h}$, again making this approach intractable.

Requirements. *The Arora-Ge attack requires that the ciphertext modulus is an integer or more generally that very little mixing of error coefficients occurs when going from R to $\text{Rep}(R_g)$. Further, the errors must be taken from a very narrow distribution.*

5.8 Evaluation attacks

One can consider the folding attack of Gentry on NTRU described in Section 4.5 as the first evaluation attack. Here we consider how such an attack can be mounted on the polynomial LWE problem instead. In [36], Eisenträger,

Hallgren and Lauter gave a simple attack on the Ideal-LWE $_{f,q,1,1,\ell,\chi}$ problem when the defining polynomial f has a root at 1 when taken modulo a prime modulus q : $f(1) \equiv 0 \pmod{q}$. Due to this property, the evaluation at one map $\mathbb{Z}_q[X]/(f(X)) \rightarrow \mathbb{Z}_q$ given by $\mathbf{a}(X) + (q, f(X)) \mapsto \mathbf{a}(1) + q\mathbb{Z}$ is well defined. The attack consists of applying this map component wise to the samples (\mathbf{a}, \mathbf{b}) and noting that the evaluation at one map is a ring homomorphism so that if the samples are sampled from the polynomial LWE distribution then $\mathbf{b}(1) = \mathbf{a}(1)\mathbf{s}(1) + \mathbf{e}(1) \pmod{q}$ and we have one dimensional LWE samples with secret $\mathbf{s}(1)$. Hence, one can test each possible value of $\mathbf{s}(1)$, computing $\mathbf{b}(1) - \mathbf{a}(1)\mathbf{s}(1) \pmod{q}$ and consider this new distribution. In the case of an incorrect guess for $\mathbf{s}(1)$ or the case of uniformly random \mathbf{b} , since q is prime, the distribution is uniformly random modulo q . On a correct guess for $\mathbf{s}(1)$ however, the distribution will be non-uniform so that for large enough q it is distinguishable and hence we can determine that the original samples were not uniformly distributed. In this way, one can solve the Ideal-DLWE $_{f,q,1,1,\ell,\chi}$ problem in time $\tilde{O}(q)$.

The authors further suggest a slight generalisation of the attack for which f has a root ξ modulo q of small order in \mathbb{Z}_q^\times . The attack now uses the evaluation at ξ map instead of the evaluation at 1 map. Further, $e_i(\xi)$ may no longer be small but due to the small order of ξ can still be distinguished from uniform with non-negligible advantage for suitably large q .

The above evaluation attack was analysed by Elias et al. in [37] where they give the following proposition.

Proposition 1 (Proposition 1 and 2 from [37]). *Let q be a prime and $f(X) \in \mathbb{Z}[X]$ of degree n be such that there exists $\xi \in \mathbb{Z}$ with $f(\xi) \equiv 0 \pmod{q}$ and ξ having order t in the multiplicative group \mathbb{Z}_q^\times . Also let χ_σ be the spherical Gaussian distribution on $\mathbb{Z}[X]/(f(X))$, with respect to the power basis, with standard deviation σ but that has been truncated at width 2σ . Assume one of the three following cases:*

1. $(4\sigma n/t)^t < q$ and set $p = (4\sigma n/t)^t/2$;
2. $\xi = \pm 1$ and $8\sigma\sqrt{n} < q$ and set $p = 1/2$;
3. ξ has small multiplicative order $t \geq 3$ modulo q and

$$8\sigma \frac{\sqrt{n}}{\sqrt{t}} \frac{\sqrt{\beta^{2t} - 1}}{\sqrt{\beta^2 - 1}} < q,$$

and set $p = 1/2$.

Then one can solve Ideal-DLWE $_{f,q,1,1,\ell,\chi_\sigma}$ with probability at least $1 - p^\ell$. The running time of this attack is $\tilde{O}((\ell + n)q)$ in case 1 and $\tilde{O}(\ell q)$ otherwise.

Generalisation

The attack can be generalised to the generic ideal setting upon considering that evaluation-at- z is equivalent to reduction modulo $X - z$. The resulting congruence

is well defined as long as $(X - z, f(X), g(X)) \neq (1)$ as $\mathbb{Z}[X]$ -ideals. In fact, one can consider more generally reduction modulo a polynomial $h(X)$ as long as $(f(X), g(X), h(X)) \neq (1)$ as ideals. Reducing samples modulo $h(X)$ into the ring $\mathbb{Z}[X]/(f(X), g(X), h(X))$ gives samples in a smaller ring which may be easier to solve than in R_g if the error distribution is mapped to one which is still distinguishable from uniform. For larger values of m it is straightforward to consider applying the same techniques coordinate-wise.

While this attack is potentially very powerful, it is straightforward to choose parameters where there are no suitable choices for h . We note that the MLHC problem is naturally immune to this attack, since neither evaluation at 2, nor evaluation at 1, results in any non-trivial information.

Requirements. *For this attack to work we require the existence of a polynomial $h(X) \in \mathbb{Z}[X]$ for which the $\mathbb{Z}[X]$ -ideal $(f(X), g(X), h(X))$ is not trivial and where the reduction modulo $(g(X), h(X))$ of the error distribution is still distinguishable from the uniform distribution.*

5.9 Zero-forcing: The attack of Coron and Gini

Inspired by the zero-forcing attack of Beunardeau et al. on the MLHR $_{n,h}$ problem [20], Coron and Gini [32] give a variant of the attack against the Mersenne low Hamming combination assumption, i.e. Ideal-DLWE $_{X^n-1, X-2, 1, 1, 2, \chi}$ with χ the uniform distribution on binary polynomials in R having h non-zero coefficients and the secret also sampled from χ . Further, this can easily be modified to attack the MLHC $_{n,h}$, Ideal-LWE $_{X^n-1, X-2, 1, 1, 1, \chi}$, problem which success 2^{-2h} over all possible choices of \mathbf{s} and \mathbf{e} .

First, assume that one is given $(\mathbf{a}_1, \mathbf{b}_1)$ and $(\mathbf{a}_2, \mathbf{b}_2)$ where $\mathbf{b}_i \equiv \mathbf{a}_i \mathbf{s} + \mathbf{e}_i \pmod{M}$ for $i = 1, 2$, where again $M = 2^n - 1$, and $\mathbf{s}, \mathbf{e}_1, \mathbf{e}_2$ are n bit integers with binary expansions having Hamming weight h . One again chooses a balanced interval-like partition for each of \mathbf{s}, \mathbf{e}_1 and \mathbf{e}_2 this time consisting of k, ℓ and j blocks respectively. In their approach, blocks are not classified as zero or non-zero and can be thought of as implicitly consisting of a non-zero block together with its following zero block. Let the partition of \mathbf{s} have blocks starting at indices p_1, \dots, p_k , and similarly q_1, \dots, q_ℓ for \mathbf{e}_1 and r_1, \dots, r_j for \mathbf{e}_2 .

For a constant β , one constructs the lattice spanned by the rows of the matrix

$$\begin{pmatrix} \beta|0\ 0 \cdots 0|0 \cdots 0 & \mathbf{b}_1 2^{-q_1} & |0 \cdots 0 & \mathbf{b}_2 2^{-r_1} \\ 0|1\ 0 \cdots 0|0 \cdots 0 & -\mathbf{a}_1 2^{p_k - q_1} & |0 \cdots 0 & -\mathbf{a}_2 2^{p_k - r_1} \\ 0|0\ 1 \cdots 0|0 \cdots 0 & -\mathbf{a}_1 2^{p_{k-1} - q_1} & |0 \cdots 0 & -\mathbf{a}_2 2^{p_{k-1} - r_1} \\ \vdots & \vdots & \vdots & \vdots \\ 0|0\ 0 \cdots 1|0 \cdots 0 & -\mathbf{a}_1 2^{p_1 - q_1} & |0 \cdots 0 & -\mathbf{a}_2 2^{p_1 - r_1} \\ 0|0\ 0 \cdots 0|1 \cdots 0 & -2^{q_\ell - q_1} & |0 \cdots 0 & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 0|0\ 0 \cdots 0|0 \cdots 1 & -2^{q_2 - q_1} & |0 \cdots 0 & 0 \\ 0|0\ 0 \cdots 0|0 \cdots 0 & M & |0 \cdots 0 & 0 \\ 0|0\ 0 \cdots 0|0 \cdots 0 & 0 & |1 \cdots 0 & -2^{r_j - r_1} \\ \vdots & \vdots & \vdots & \vdots \\ 0|0\ 0 \cdots 0|0 \cdots 0 & 0 & |0 \cdots 1 & -2^{r_2 - r_1} \\ 0|0\ 0 \cdots 0|0 \cdots 0 & 0 & |0 \cdots 0 & M \end{pmatrix}$$

and notes that, on writing

$$\mathbf{s} = \sum_{i=1}^k x_i 2^{p_i} \bmod M, \quad \mathbf{e}_1 = \sum_{i=1}^{\ell} y_i 2^{q_i} \bmod M, \quad \text{and} \quad \mathbf{e}_2 = \sum_{i=1}^j z_i 2^{r_i} \bmod M,$$

the vector $(\beta x_1 \cdots x_k y_1 \cdots y_\ell z_1 \cdots z_j)$ lies in the lattice and for a well chosen set of partitions it is a short vector.

Just as in the attack of Beunardeau et al. the approach is to sample random balanced interval-like partitions with $k = \ell = j$, construct the associated lattice and run LLL reduction on it in the hope of recovering \mathbf{s} , \mathbf{e}_1 and \mathbf{e}_2 (or some rotation). The probability of success is now given by $(\frac{2}{3})^{3h} \approx 2^{-1.75h}$ over all possible choices of \mathbf{s} , \mathbf{e}_1 and \mathbf{e}_2 . Now, if \mathbf{b}_1 and \mathbf{b}_2 are actually random integers modulo M then the success probability of finding a solution consisting of three n -bit integers having Hamming weight h is negligible so one can distinguish between the two cases with non-negligible advantage in time $O(2^{1.75h})$.

If one removes the final j columns of the lattice and sets $\mathbf{e}_1 = \mathbf{e}$ and $\mathbf{a}_1 = \mathbf{a}$ then one recovers the attack on the MLHC $_{n,h}$ problem mentioned above which runs in time $O(2^{2h})$.

Generalisation

The idea here is the same as the zero-forcing attacks on Ideal-NTRU only applied to the primal-LWE lattice instead of the NTRU lattice. This time, we will first have to slightly modify the LWE lattice. We assume that k , the number of independent secrets, is one as before and that the secret \mathbf{s} is sampled from the error distribution (we drop this condition later but it helps to ease the exposition) and that the error distribution produces elements of R which have many zero coefficients. Define $d = (m + \ell)n + \ell \deg r + 1$ and the modified lattice

$$A'(\mathbf{a}, \mathbf{b}) = \left\{ (\iota(\mathbf{x}), \iota(\mathbf{y}), \mathbf{z}, w) \in \mathbb{Z}^d \mid \begin{array}{l} \mathbf{x} \in R^m, \mathbf{y} \in R^\ell, \\ \mathbf{z} \equiv \iota(\text{rep}_g(w\mathbf{b} + \mathbf{a}\mathbf{x}^T + \mathbf{y}^T)) \bmod a \end{array} \right\},$$

where again we have abused notation and dropped the zero coefficients appearing at the end of \mathbf{z} .

As before, we are only interested in vectors for which \mathbf{z} is zero so we will scale these coordinates by a large constant K . We may also scale the final coordinate by some small non-zero scalar W to balance the coordinates of the small vector we are trying to recover. For this purpose we define the lattice

$$\Lambda'_{K,W}(\mathbf{a}, \mathbf{b}) = \{(\mathbf{x}, \mathbf{y}, \mathbf{z}, w) \in \mathbb{Z}^d \mid (\mathbf{x}, \mathbf{y}, K^{-1}\mathbf{z}, W^{-1}w) \in \Lambda'(\mathbf{a}, \mathbf{b})\}.$$

It is clear that if $\mathbf{b} = \mathbf{a}\mathbf{s} + \mathbf{e}$ then $(\iota(\mathbf{s}), \iota(\mathbf{e}), \mathbf{0}, -W)$ is a short element in this lattice. Further, we can easily compute a basis for the lattice by setting the first basis vector as $(\mathbf{0}, \mathbf{0}, K\iota(\text{rep}_g(\mathbf{b})), W)$, the next mn vectors running over the power-basis for $\mathbf{x} \in R^m$ and fixing $\mathbf{y} = \mathbf{0}$ and $w = 0$, the next ℓn vectors running over the power-basis for $\mathbf{y} \in R^\ell$ while \mathbf{x} and w are fixed to zero and finally $\ell \deg r$ vectors which perform reduction modulo Ka in the third component, while all other entries are set to zero.

Just as before, the important point now is that it is easy to find a basis for the lattice corresponding to setting certain coefficients of some element of R to be zero. The approach of the zero-forcing attack is to guess positions that can be set to zero to significantly reduce the dimension of this lattice, then reducing this sublattice using LLL and assuming that the guess was correct we may be able to find the corresponding secret and errors. For more details on how exactly this is done see the zero-forcing attack on the Ideal-NTRU problem as the process is almost identical.

Remark 3. Finally, we remark that the attack is still applicable if the secret is not sampled from the error distribution and one cannot convert the problem to such an instance. To proceed, one simply drops the columns corresponding to \mathbf{x} in $\Lambda'(\mathbf{a}, \mathbf{b})$ and allow \mathbf{z} to be any element satisfying $\mathbf{z} \equiv \iota(\text{rep}_g(w\mathbf{b} + \mathbf{a}\mathbf{x}^T + \mathbf{y}^T))$ for some $\mathbf{x} \in R^m$.

Requirements. *The requirements for this attack are much the same the zero-forcing attack on Ideal-NTRU, only this time the errors also need to be sparse; something that is typically not the case for standard LWE type problems. Further, f and g should have sparse coefficient vectors with small coefficients.*

6 Conclusion

In this paper we have detailed the most relevant attacks on standard variants of the NTRU, LWE and SIS problems as well as those on the newer MLHR and MLHC problems and considered if and how they can be applied to the more general Ideal-NTRU, Ideal-LWE and Ideal-SIS problems which use a general polynomial ciphertext modulus.

We have seen that attacks such as the Arora-Ge attack and the zero-forcing attacks require very specific parameter choices in order to be generalised beyond their original intended use while other attacks such as lattice attacks on

the Ideal-SIS problem can be applied for any choice of ciphertext modulus. In between these two extremes, we have attacks such as subfield attacks on the Ideal-NTRU problem and the primal attack on Ideal-LWE which still require the ciphertext modulus to conform to a rather restrictive set of requirements as well as the dual attack on Ideal-LWE and combinatorial attacks on Ideal-SIS which have more minimal requirements on the ciphertext modulus.

Whilst we were able to give somewhat high-level conditions which need to be satisfied for the attacks we consider to be applicable, determining a set of concrete parameter choices for which a problem achieves a certain level of security is currently not possible outside of the standard problems. It remains important future work to obtain a deeper understanding of the applicability and running times of the generalised attacks presented in this work. In doing so it may then be possible to provide concrete estimates for the security of a given instance of either the Ideal-NTRU, the Ideal-LWE, or the Ideal-SIS problems.

Acknowledgements

The material in this work first appeared in various parts of the first author’s PhD thesis [23] and has been reworked and reordered here. We would like to thank all the jury members for their feedback and in particular the external jury members Damien Stehlé and Joppe Bos.

This work was supported in part by the Research Council KU Leuven grants C14/18/067 and STG/17/019, and by CyberSecurity Research Flanders with reference number VR20192203. Carl Bootland was funded by an FWO fellowship. Alan Szepieniec was supported by Nervos.

References

- [1] Divesh Aggarwal et al. *A New Public-Key Cryptosystem via Mersenne Numbers*. Cryptology ePrint Archive, Report 2017/481. <https://eprint.iacr.org/2017/481>. 2017.
- [2] Divesh Aggarwal et al. “A New Public-Key Cryptosystem via Mersenne Numbers”. In: *Advances in Cryptology – CRYPTO 2018, Part III*. Ed. by Hovav Shacham and Alexandra Boldyreva. Vol. 10993. Lecture Notes in Computer Science. Springer, Heidelberg, Aug. 2018, pp. 459–482. DOI: 10.1007/978-3-319-96878-0_16.
- [3] Miklós Ajtai. “Generating Hard Instances of Lattice Problems (Extended Abstract)”. In: *28th Annual ACM Symposium on Theory of Computing*. ACM Press, May 1996, pp. 99–108. DOI: 10.1145/237814.237838.
- [4] Martin R. Albrecht. “On Dual Lattice Attacks Against Small-Secret LWE and Parameter Choices in HELib and SEAL”. In: *Advances in Cryptology – EUROCRYPT 2017, Part II*. Ed. by Jean-Sébastien Coron and Jesper Buus Nielsen. Vol. 10211. Lecture Notes in Computer Science. Springer, Heidelberg, Apr. 2017, pp. 103–129. DOI: 10.1007/978-3-319-56614-6_4.

- [5] Martin R. Albrecht, Shi Bai, and Léo Ducas. “A Subfield Lattice Attack on Overstretched NTRU Assumptions - Cryptanalysis of Some FHE and Graded Encoding Schemes”. In: *Advances in Cryptology – CRYPTO 2016, Part I*. Ed. by Matthew Robshaw and Jonathan Katz. Vol. 9814. Lecture Notes in Computer Science. Springer, Heidelberg, Aug. 2016, pp. 153–178. DOI: 10.1007/978-3-662-53018-4_6.
- [6] Martin R. Albrecht, Robert Fitzpatrick, and Florian Göpfert. “On the Efficacy of Solving LWE by Reduction to Unique-SVP”. In: *ICISC 13: 16th International Conference on Information Security and Cryptology*. Ed. by Hyang-Sook Lee and Dong-Guk Han. Vol. 8565. Lecture Notes in Computer Science. Springer, Heidelberg, Nov. 2014, pp. 293–310. DOI: 10.1007/978-3-319-12160-4_18.
- [7] Martin R. Albrecht, Rachel Player, and Sam Scott. “On the concrete hardness of Learning with Errors”. In: *Journal of Mathematical Cryptology* 9.3 (2015). An updated version of this paper can be found at <https://eprint.iacr.org/2015/046.pdf>, pp. 169–203. DOI: 10.1515/jmc-2015-0016.
- [8] Martin R. Albrecht et al. “Lazy Modulus Switching for the BKW Algorithm on LWE”. In: *PKC 2014: 17th International Conference on Theory and Practice of Public Key Cryptography*. Ed. by Hugo Krawczyk. Vol. 8383. Lecture Notes in Computer Science. Springer, Heidelberg, Mar. 2014, pp. 429–445. DOI: 10.1007/978-3-642-54631-0_25.
- [9] Martin R. Albrecht et al. “Algebraic Algorithms for LWE Problems”. In: *ACM Commun. Comput. Algebra* 49.2 (2015), p. 62. DOI: 10.1145/2815111.2815158.
- [10] Martin R. Albrecht et al. “On the complexity of the BKW algorithm on LWE”. In: *Des. Codes Cryptogr.* 74 (Feb. 2015), pp. 325–354. DOI: 10.1007/s10623-013-9864-x.
- [11] Yoshinori Aono, Le Trieu Phong, and Lihua Wang. *Hardness Estimation of LWE via Band Pruning*. Cryptology ePrint Archive, Report 2015/1026. <https://eprint.iacr.org/2015/1026>. 2015.
- [12] Benny Applebaum et al. “Fast Cryptographic Primitives and Circular-Secure Encryption Based on Hard Learning Problems”. In: *Advances in Cryptology – CRYPTO 2009*. Ed. by Shai Halevi. Vol. 5677. Lecture Notes in Computer Science. Springer, Heidelberg, Aug. 2009, pp. 595–618. DOI: 10.1007/978-3-642-03356-8_35.
- [13] Sanjeev Arora and Rong Ge. “New Algorithms for Learning in Presence of Errors”. In: *ICALP 2011: 38th International Colloquium on Automata, Languages and Programming, Part I*. Ed. by Luca Aceto, Monika Henzinger, and Jiri Sgall. Vol. 6755. Lecture Notes in Computer Science. Springer, Heidelberg, July 2011, pp. 403–415. DOI: 10.1007/978-3-642-22006-7_34.
- [14] László Babai. “On Lovász’ lattice reduction and the nearest lattice point problem (shortened version)”. In: *STACS 1985*. Ed. by Kurt Mehlhorn.

- Vol. 182. Lecture Notes in Computer Science. Springer, Heidelberg, 1985, pp. 13–20. DOI: 10.1007/BFb0023990.
- [15] László Babai. “On Lovász’ lattice reduction and the nearest lattice point problem”. In: *Combinatorica* 6.1 (1986), pp. 1–13. DOI: 10.1007/BF02579403.
 - [16] Shi Bai and Steven D. Galbraith. “Lattice Decoding Attacks on Binary LWE”. In: *ACISP 14: 19th Australasian Conference on Information Security and Privacy*. Ed. by Willy Susilo and Yi Mu. Vol. 8544. Lecture Notes in Computer Science. Springer, Heidelberg, July 2014, pp. 322–337. DOI: 10.1007/978-3-319-08344-5_21.
 - [17] Shi Bai et al. “Improved Combinatorial Algorithms for the Inhomogeneous Short Integer Solution Problem”. In: *Journal of Cryptology* 32.1 (Jan. 2019), pp. 35–83. DOI: 10.1007/s00145-018-9304-1.
 - [18] Anja Becker, Jean-Sébastien Coron, and Antoine Joux. “Improved Generic Algorithms for Hard Knapsacks”. In: *Advances in Cryptology – EUROCRYPT 2011*. Ed. by Kenneth G. Paterson. Vol. 6632. Lecture Notes in Computer Science. Springer, Heidelberg, May 2011, pp. 364–385. DOI: 10.1007/978-3-642-20465-4_21.
 - [19] Daniel J. Bernstein et al. “NTRU Prime: Reducing Attack Surface at Low Cost”. In: *SAC 2017: 24th Annual International Workshop on Selected Areas in Cryptography*. Ed. by Carlisle Adams and Jan Camenisch. Vol. 10719. Lecture Notes in Computer Science. Springer, Heidelberg, Aug. 2017, pp. 235–260. DOI: 10.1007/978-3-319-72565-9_12.
 - [20] Marc Beunardeau et al. “On the Hardness of the Mersenne Low Hamming Ratio Assumption”. In: *Progress in Cryptology - LATINCRYPT 2017: 5th International Conference on Cryptology and Information Security in Latin America*. Ed. by Tanja Lange and Orr Dunkelman. Vol. 11368. Lecture Notes in Computer Science. Springer, Heidelberg, Sept. 2017, pp. 166–174. DOI: 10.1007/978-3-030-25283-0_9.
 - [21] Avrim Blum, Adam Kalai, and Hal Wasserman. “Noise-Tolerant Learning, the Parity Problem, and the Statistical Query Model”. In: *J. ACM* 50.4 (2003), pp. 506–519. DOI: 10.1145/792538.792543.
 - [22] Koen de Boer et al. “Attacks on the AJPS Mersenne-Based Cryptosystem”. In: *Post-Quantum Cryptography - 9th International Conference, PQCrypto 2018*. Ed. by Tanja Lange and Rainer Steinwandt. Springer, Heidelberg, 2018, pp. 101–120. DOI: 10.1007/978-3-319-79063-3_5.
 - [23] Carl Bootland. “Efficiency and security aspects of lattice-based cryptography”. <https://www.esat.kuleuven.be/cosic/publications/thesis-399.pdf>. PhD thesis. KU Leuven, 2021.
 - [24] Carl Bootland et al. “A framework for cryptographic problems from linear algebra”. In: *Journal of Mathematical Cryptology* 14.1 (2020), pp. 202–217. DOI: 10.1515/jmc-2019-0032.
 - [25] Zvika Brakerski, Craig Gentry, and Vinod Vaikuntanathan. “(Leveled) fully homomorphic encryption without bootstrapping”. In: *ITCS 2012: 3rd Innovations in Theoretical Computer Science*. Ed. by Shafi Goldwasser.

- Association for Computing Machinery, Jan. 2012, pp. 309–325. DOI: 10.1145/2090236.2090262.
- [26] Paul Camion and Jacques Patarin. “The Knapsack Hash Function proposed at Crypto’89 can be broken”. In: *Advances in Cryptology – EUROCRYPT’91*. Ed. by Donald W. Davies. Vol. 547. Lecture Notes in Computer Science. Springer, Heidelberg, Apr. 1991, pp. 39–53. DOI: 10.1007/3-540-46416-6_3.
- [27] Yuanmi Chen. “Lattice reduction and concrete security of fully homomorphic encryption”. PhD thesis. Paris Diderot University (Paris 7), 2013.
- [28] J. H. Cheon et al. “A Hybrid of Dual and Meet-in-the-Middle Attack on Sparse and Ternary Secret LWE”. In: *IEEE Access* 7 (2019), pp. 89497–89506. DOI: 10.1109/ACCESS.2019.2925425.
- [29] Jung Hee Cheon, Jinhyuck Jeong, and Changmin Lee. “An algorithm for NTRU problems and cryptanalysis of the GGH multilinear map without a low-level encoding of zero”. In: *LMS Journal of Computation and Mathematics* 19.A (2016), pp. 255–266. DOI: 10.1112/S1461157016000371.
- [30] Herman Chernoff. “A Measure of Asymptotic Efficiency for Tests of a Hypothesis Based on the sum of Observations”. In: *Ann. Math. Statist.* 23.4 (1952), pp. 493–507.
- [31] Don Coppersmith and Adi Shamir. “Lattice Attacks on NTRU”. In: *Advances in Cryptology – EUROCRYPT’97*. Ed. by Walter Fumy. Vol. 1233. Lecture Notes in Computer Science. Springer, Heidelberg, May 1997, pp. 52–61. DOI: 10.1007/3-540-69053-0_5.
- [32] Jean-Sébastien Coron and Agnese Gini. “Improved cryptanalysis of the AJPS Mersenne based cryptosystem”. In: *Journal of Mathematical Cryptology* 14.1 (2020), pp. 218–223. DOI: 10.1515/jmc-2019-0027.
- [33] Jintai Ding. *Fast Algorithm to solve a family of SIS problem with ℓ_∞ norm*. Cryptology ePrint Archive, Report 2010/581. <https://eprint.iacr.org/2010/581>. 2010.
- [34] Jintai Ding. *Solving LWE problem with bounded errors in polynomial time*. Cryptology ePrint Archive, Report 2010/558. <https://eprint.iacr.org/2010/558>. 2010.
- [35] Alexandre Duc, Florian Tramèr, and Serge Vaudenay. “Better Algorithms for LWE and LWR”. In: *Advances in Cryptology – EUROCRYPT 2015, Part I*. Ed. by Elisabeth Oswald and Marc Fischlin. Vol. 9056. Lecture Notes in Computer Science. Springer, Heidelberg, Apr. 2015, pp. 173–202. DOI: 10.1007/978-3-662-46800-5_8.
- [36] Kirsten Eisenträger, Sean Hallgren, and Kristin E. Lauter. “Weak Instances of PLWE”. In: *SAC 2014: 21st Annual International Workshop on Selected Areas in Cryptography*. Ed. by Antoine Joux and Amr M. Youssef. Vol. 8781. Lecture Notes in Computer Science. Springer, Heidelberg, Aug. 2014, pp. 183–194. DOI: 10.1007/978-3-319-13051-4_11.
- [37] Yara Elias et al. “Provably Weak Instances of Ring-LWE”. In: *Advances in Cryptology – CRYPTO 2015, Part I*. Ed. by Rosario Gennaro and Matthew

- J. B. Robshaw. Vol. 9215. Lecture Notes in Computer Science. Springer, Heidelberg, Aug. 2015, pp. 63–92. DOI: 10.1007/978-3-662-47989-6_4.
- [38] Nicolas Gama and Phong Q. Nguyen. “Predicting Lattice Reduction”. In: *Advances in Cryptology – EUROCRYPT 2008*. Ed. by Nigel P. Smart. Vol. 4965. Lecture Notes in Computer Science. Springer, Heidelberg, Apr. 2008, pp. 31–51. DOI: 10.1007/978-3-540-78967-3_3.
- [39] Craig Gentry. “Key Recovery and Message Attacks on NTRU-Composite”. In: *Advances in Cryptology – EUROCRYPT 2001*. Ed. by Birgit Pfitzmann. Vol. 2045. Lecture Notes in Computer Science. Springer, Heidelberg, May 2001, pp. 182–194. DOI: 10.1007/3-540-44987-6_12.
- [40] Craig Gentry and Michael Szydlo. “Cryptanalysis of the Revised NTRU Signature Scheme”. In: *Advances in Cryptology – EUROCRYPT 2002*. Ed. by Lars R. Knudsen. Vol. 2332. Lecture Notes in Computer Science. Springer, Heidelberg, Apr. 2002, pp. 299–320. DOI: 10.1007/3-540-46035-7_20.
- [41] Chunsheng Gu. “Integer Version of Ring-LWE and Its Applications”. In: *Security and Privacy in Social Networks and Big Data*. Ed. by Weizhi Meng and Steven Furnell. Springer, Heidelberg, 2019, pp. 110–122. DOI: 10.1007/978-981-15-0758-8_9.
- [42] Qian Guo, Thomas Johansson, and Paul Stankovski. “Coded-BKW: Solving LWE Using Lattice Codes”. In: *Advances in Cryptology – CRYPTO 2015, Part I*. Ed. by Rosario Gennaro and Matthew J. B. Robshaw. Vol. 9215. Lecture Notes in Computer Science. Springer, Heidelberg, Aug. 2015, pp. 23–42. DOI: 10.1007/978-3-662-47989-6_2.
- [43] Mike Hamburg. *Three Bears*. Tech. rep. available at <https://csrc.nist.gov/projects/post-quantum-cryptography/round-2-submissions>. National Institute of Standards and Technology, 2019.
- [44] Gottfried Herold, Elena Kirshanova, and Alexander May. “On the asymptotic complexity of solving LWE”. In: *Des. Codes Cryptogr.* 86 (Jan. 2017), pp. 55–83. DOI: 10.1007/s10623-016-0326-0.
- [45] Jeffrey Hoffstein, Jill Pipher, and Joseph H. Silverman. *NTRU: A new high-speed public key cryptosystem*. Circulated at CRYPTO ‘96 rump session, <https://ntru.org/f/hps96.pdf>. 1996.
- [46] Jeffrey Hoffstein, Jill Pipher, and Joseph H. Silverman. “NTRU: A ring-based public key cryptosystem”. In: *Algorithmic Number Theory*. Ed. by Joe P. Buhler. Vol. 1423. Lecture Notes in Computer Science. Springer, Heidelberg, June 1998, pp. 267–288. DOI: 10.1007/BFb0054868.
- [47] Nick Howgrave-Graham. “A Hybrid Lattice-Reduction and Meet-in-the-Middle Attack Against NTRU”. In: *Advances in Cryptology – CRYPTO 2007*. Ed. by Alfred Menezes. Vol. 4622. Lecture Notes in Computer Science. Springer, Heidelberg, Aug. 2007, pp. 150–169. DOI: 10.1007/978-3-540-74143-5_9.
- [48] Nick Howgrave-Graham and Antoine Joux. “New Generic Algorithms for Hard Knapsacks”. In: *Advances in Cryptology – EUROCRYPT 2010*. Ed. by Henri Gilbert. Vol. 6110. Lecture Notes in Computer Science. Springer,

- Heidelberg, May 2010, pp. 235–256. DOI: 10.1007/978-3-642-13190-5_12.
- [49] Nick Howgrave-Graham, Joseph H. Silverman, and William Whyte. *A Meet-In-The-Middle Attack on an NTRU Private Key*. Tech. rep. 004. <https://ntru.org/f/tr/tr004v2.pdf>. NTRU Cryptosystems, June 2003.
- [50] Ravi Kannan. “Minkowski’s Convex Body Theorem and Integer Programming”. In: *Math. Oper. Res.* 12.3 (1987), pp. 415–440. DOI: 10.1287/moor.12.3.415.
- [51] Paul Kirchner and Pierre-Alain Fouque. “An Improved BKW Algorithm for LWE with Applications to Cryptography and Lattices”. In: *Advances in Cryptology – CRYPTO 2015, Part I*. Ed. by Rosario Gennaro and Matthew J. B. Robshaw. Vol. 9215. Lecture Notes in Computer Science. Springer, Heidelberg, Aug. 2015, pp. 43–62. DOI: 10.1007/978-3-662-47989-6_3.
- [52] Paul Kirchner and Pierre-Alain Fouque. “Revisiting Lattice Attacks on Overstretched NTRU Parameters”. In: *Advances in Cryptology – EURO-CRYPT 2017, Part I*. Ed. by Jean-Sébastien Coron and Jesper Buus Nielsen. Vol. 10210. Lecture Notes in Computer Science. Springer, Heidelberg, Apr. 2017, pp. 3–26. DOI: 10.1007/978-3-319-56620-7_1.
- [53] Adeline Langlois and Damien Stehlé. “Worst-case to average-case reductions for module lattices”. In: *Des. Codes Cryptogr.* 75 (June 2015), pp. 565–599. DOI: 10.1007/s10623-014-9938-4.
- [54] Richard Lindner and Chris Peikert. “Better Key Sizes (and Attacks) for LWE-Based Encryption”. In: *Topics in Cryptology – CT-RSA 2011*. Ed. by Aggelos Kiayias. Vol. 6558. Lecture Notes in Computer Science. Springer, Heidelberg, Feb. 2011, pp. 319–339. DOI: 10.1007/978-3-642-19074-2_21.
- [55] Mingjie Liu and Phong Q. Nguyen. “Solving BDD by Enumeration: An Update”. In: *Topics in Cryptology – CT-RSA 2013*. Ed. by Ed Dawson. Vol. 7779. Lecture Notes in Computer Science. Springer, Heidelberg, Feb. 2013, pp. 293–309. DOI: 10.1007/978-3-642-36095-4_19.
- [56] Vadim Lyubashevsky and Daniele Micciancio. “On Bounded Distance Decoding, Unique Shortest Vectors, and the Minimum Distance Problem”. In: *Advances in Cryptology – CRYPTO 2009*. Ed. by Shai Halevi. Vol. 5677. Lecture Notes in Computer Science. Springer, Heidelberg, Aug. 2009, pp. 577–594. DOI: 10.1007/978-3-642-03356-8_34.
- [57] Vadim Lyubashevsky, Chris Peikert, and Oded Regev. “On Ideal Lattices and Learning with Errors over Rings”. In: *Advances in Cryptology – EUROCRYPT 2010*. Ed. by Henri Gilbert. Vol. 6110. Lecture Notes in Computer Science. Springer, Heidelberg, May 2010, pp. 1–23. DOI: 10.1007/978-3-642-13190-5_1.
- [58] Alexander May. *Auf Polynomgleichungen basierende Public-Key-Kryptosysteme*. <http://publikationen.ub.uni-frankfurt.de/opus4/frontdoor/index/index/year/2006/docId/2439>. 1999.

- [59] Alexander May and Joseph H. Silverman. “Dimension Reduction Methods for Convolution Modular Lattices”. In: *Cryptography and Lattices*. Ed. by Joseph H. Silverman. Springer, Heidelberg, 2001, pp. 110–125. DOI: 10.1007/3-540-44670-2_10.
- [60] Daniele Micciancio. “Generalized Compact Knapsacks, Cyclic Lattices, and Efficient One-Way Functions from Worst-Case Complexity Assumptions”. In: *43rd Annual Symposium on Foundations of Computer Science*. IEEE Computer Society Press, Nov. 2002, pp. 356–365. DOI: 10.1109/SFCS.2002.1181960.
- [61] Daniele Micciancio. “Generalized Compact Knapsacks, Cyclic Lattices, and Efficient One-Way Functions”. In: *Comput. Complex.* 16 (Dec. 2007), pp. 365–411. DOI: 10.1007/s0037-007-0234-9.
- [62] Daniele Micciancio and Oded Regev. “Lattice-based Cryptography”. In: *Post-Quantum Cryptography*. Ed. by Daniel J. Bernstein, Johannes Buchmann, and Erik Dahmen. Springer, Heidelberg, 2009, pp. 147–191. DOI: 10.1007/978-3-540-88702-7_5.
- [63] Lorenz Minder and Alistair Sinclair. “The Extended k-tree Algorithm”. In: *Journal of Cryptology* 25.2 (Apr. 2012), pp. 349–382. DOI: 10.1007/s00145-011-9097-y.
- [64] Rachel Player. “Parameter selection in lattice-based cryptography”. PhD thesis. Royal Holloway, University of London, 2018.
- [65] Oded Regev. “On lattices, learning with errors, random linear codes, and cryptography”. In: *37th Annual ACM Symposium on Theory of Computing*. Ed. by Harold N. Gabow and Ronald Fagin. ACM Press, May 2005, pp. 84–93. DOI: 10.1145/1060590.1060603.
- [66] Oded Regev. “On Lattices, Learning with Errors, Random Linear Codes, and Cryptography”. In: *J. ACM* 56.6 (2009). Preliminary version in STOC 2005 [65]. DOI: 10.1145/1568318.1568324.
- [67] Daniel Rosenberg. *NTRUEncrypt and Lattice Attacks*. http://kiosk.nada.kth.se/utbildning/grukth/exjobb/rapportlistor/2004/rapporter04/rosenberg_daniel_04163.pdf. 2004.
- [68] Richard Schroepel and Adi Shamir. “A $T = O(2^{n/2})$, $S = O(2^{n/4})$ Algorithm for Certain NP-Complete Problems”. In: *SIAM J. Comput.* 10.3 (1981), pp. 456–464. DOI: 10.1137/0210033.
- [69] Joseph H. Silverman. *Dimension-Reduced Lattices, Zero-Forced Lattices, and the NTRU Public Key Cryptosystem*. Tech. rep. 013. <https://ntru.org/f/tr/tr013v1.pdf>. NTRU Cryptosystems, Mar. 1999.
- [70] Katherine E. Stange. *Algebraic aspects of solving Ring-LWE, including ring-based improvements in the Blum-Kalai-Wasserman algorithm*. Cryptology ePrint Archive, Report 2019/183. <https://eprint.iacr.org/2019/183>. 2019.
- [71] Damien Stehlé et al. “Efficient Public Key Encryption Based on Ideal Lattices”. In: *Advances in Cryptology – ASIACRYPT 2009*. Ed. by Mitsuru Matsui. Vol. 5912. Lecture Notes in Computer Science. Springer, Heidelberg, Dec. 2009, pp. 617–635. DOI: 10.1007/978-3-642-10366-7_36.

- [72] David Wagner. “A Generalized Birthday Problem”. In: *Advances in Cryptology – CRYPTO 2002*. Ed. by Moti Yung. Vol. 2442. Lecture Notes in Computer Science. Springer, Heidelberg, Aug. 2002, pp. 288–303. DOI: 10.1007/3-540-45708-9_19.

A Ring-BKW

One can apply the BKW attack directly to any ring variant of the learning with errors problem, under the assumption that the error term has small coefficients, by considering one sample as n LWE samples. If one does this however, a ring sample which is zero on the first block only has one LWE sample with this property in general.

Instead of simply splitting the coefficients into blocks as in the original BKW attack, Stange, in an unpublished work [70], proposes to use the ring structure to do better. We remark that the exposition here mainly follows the first version of [70] which includes the use of the Chinese remainder theorem to reduce the size of the problem. This idea was removed in later versions of the paper since such a reduction can be achieved using the other main idea present in the first version which is to use the trace map.

Stange’s attack restricts to using the power-of-two cyclotomic polynomial in the polynomial LWE setting and further assumes that q is prime and⁸ $q \equiv 1 \pmod 4$ so that, on defining $\nu := \text{ord}_2(q-1) \geq 2$, $t := 2^{\nu-1}$ and $u := n/t$, we have

$$X^n + 1 \equiv f_1(X) \cdots f_t(X) \pmod q$$

with each f_i irreducible modulo q and having degree u . Hence, we have the isomorphisms

$$R_q := \frac{\mathbb{Z}_q[X]}{(X^n + 1)} \cong \frac{\mathbb{F}_q[X]}{(f_1(X))} \times \cdots \times \frac{\mathbb{F}_q[X]}{(f_t(X))} \cong \mathbb{F}_{q^u} \times \cdots \times \mathbb{F}_{q^u} = \mathbb{F}_{q^u}^t$$

using the Chinese remainder theorem. In fact, we have $f_i = X^u - c^{2i-1}$ where the c is any root of $X^t + 1$ in \mathbb{F}_q .

We further assume, for simplicity, that the error distribution can be written so that we can sample it coefficient wise from χ and allow the secret to be sampled from an arbitrary distribution.

This structure of R_q gives us two tools which are used to improve the BKW attack; firstly, we can apply a ring homomorphism from R_q to some smaller ring and secondly, one can use the subfield structure inside the fields \mathbb{F}_{q^u} or the number field $\mathbb{Q}[X]/(f(X))$. Both tools cannot simply be applied naïvely as we now see.

For a ring homomorphism $\rho: R_q \rightarrow R/\mathfrak{a} \cong \mathbb{F}_{q^u}$, for $\mathfrak{a} \mid qR$, the coefficient-wise error distribution is mapped to the coefficient wise distribution $\xi := \sum_{i=0}^{t-1} \rho(X^u)^i \chi_i$

⁸ If not using the CRT decomposition then q is assumed only to be an odd prime which is unramified in R .

where χ_i are independent and identically distributed according to χ . This is true since ρ fixes \mathbb{F}_q and hence χ . We therefore see that smallness is not preserved in general.

In the case of going from the field \mathbb{F}_{q^u} to a subfield \mathbb{F}_{q^d} or from $\mathbb{Q}[X]/(f(X))$ to a subfield we can use the field norm or trace maps. However, applying these maps also increases the size of the errors in much the same way as we saw in Section 4.6.

Once these two obstacles have been overcome the approach is as before. First choose a block size $\beta \mid u$ and perform the BKW algorithm on the coefficients of the CRT representations of the first component of the samples if using the CRT decomposition or simply on the first component of the samples if not. In doing so we can reduce the problem to solving the problem for instances in smaller subfields. After solving these instances we can rebuild the original secret. We now give the details.

CRT Reduction Let us write $\rho_i: R_q \rightarrow \mathbb{F}_q[X]$ for $i = 1, \dots, t$ defined simply by reduction with respect to $f_i(X)$ to a polynomial of degree at most u . We choose not to map into $\mathbb{F}_q[X]/(f_i(X))$ for clarity later on and instead define multiplicative binary operations $\odot_i: \mathbb{F}_q[X] \times \mathbb{F}_q[X] \rightarrow \mathbb{F}_q[X]$ given by $\mathbf{a} \odot_i \mathbf{b} := \rho_i(\mathbf{a}\mathbf{b})$. Define the $t \times t$ matrix P as

$$\begin{aligned} P &:= \begin{pmatrix} \rho_1(X^0) & \rho_2(X^0) & \cdots & \rho_t(X^0) \\ \rho_1(X^u) & \rho_2(X^u) & \cdots & \rho_t(X^u) \\ \vdots & \vdots & \ddots & \vdots \\ \rho_1(X^{n-u}) & \rho_2(X^{n-u}) & \cdots & \rho_t(X^{n-u}) \end{pmatrix} \\ &= \begin{pmatrix} 1 & 1 & \cdots & 1 \\ c & c^3 & \cdots & c^{2t-1} \\ \vdots & \vdots & \ddots & \vdots \\ c^{t-1} & c^{3(t-1)} & \cdots & c^{(2t-1)(t-1)} \end{pmatrix}. \end{aligned}$$

We note that the entries of P are elements of \mathbb{F}_q in our specific case so we consider P to be a matrix in $\mathbf{GL}(t, \mathbb{F}_q)$. If we write $\rho = (\rho_1, \dots, \rho_t)$ as the full CRT isomorphism then for an element $\mathbf{a} \in R_q$ with coefficients $a_j \in \mathbb{F}_q$ we have

$$\rho(\mathbf{a}) = \rho \left(\sum_{j=0}^{n-1} a_j X^j \right) = \sum_{j=0}^{n-1} a_j \rho(X^j) = \sum_{i=0}^{t-1} \left(\sum_{j=0}^{u-1} a_{iu+j} X^j \right) \rho(X^{iu}).$$

We thus define $\alpha_i := \sum_{j=0}^{u-1} a_{iu+j} X^j$, $\alpha = (\alpha_i)_{i=0}^{t-1}$ and note that $\rho(\mathbf{a}) = \alpha P$. Further, denoting $P^{-1} = (\mu_{i,j})_{i,j=1}^t$ we have $\mu_{i,j} = c^{-(2i-1)(j-1)/r}$ and $\alpha = \rho(\mathbf{a})P^{-1}$ which implies $\alpha_i = \sum_{j=1}^t \mu_{j,i+1} \rho_j(\mathbf{a})$. Assume now that $(\mathbf{a}, \mathbf{b} = \mathbf{a}\mathbf{s} + \mathbf{e})$ is a sample from the polynomial learning with errors problem. We thus see that if $\rho_j(\mathbf{a}) = 0$ for all j except t then we have $\rho_j(\mathbf{b}) = \rho_j(\mathbf{e})$ for all j except for t

where we have $\rho_r(\mathbf{b}) = \rho_t(\mathbf{a}) \odot_t \rho_t(\mathbf{s}) + \rho_t(\mathbf{e})$. Hence for each i we have

$$\underbrace{\sum_{j=1}^t \mu_{j,i} \rho_j(\mathbf{b})}_{\beta_i} = \underbrace{\mu_{t,i} \rho_t(\mathbf{a})}_{\alpha_i} \odot_t \rho_t(\mathbf{s}) + \underbrace{\sum_{j=1}^t \mu_{j,i} \rho_j(\mathbf{e})}_{\epsilon_i}$$

which we can consider as a polynomial LWE sample in the ring $\mathbb{Z}_q[X]/(f_t(X))$ with secret $\rho_t(\mathbf{s})$ and error $\epsilon_i = \sum_{j=0}^{u-1} e_{iu+j} X^j$ which has small coefficients assuming the original samples had small error coefficients e_j .

More generally, if for some j_0 we have $\rho_j(\mathbf{a}) = 0$ for $j = 1, \dots, j_0 - 1$ and $\rho_j(\mathbf{s})$ is known for $j = j_0 + 1, \dots, t$ then we can compute samples

$$(\tilde{\mathbf{a}}_i, \tilde{\mathbf{b}}_i) := \left(\mu_{j_0,i} \rho_{j_0}(\mathbf{a}), \sum_{j=1}^t \mu_{j,i} \rho_j(\mathbf{b}) - \sum_{j=j_0+1}^t \mu_{j,i} \rho_j(\mathbf{a}) \odot_j \rho_j(\mathbf{s}) \right) \quad (2)$$

which when considered modulo f_{j_0} are polynomial LWE samples with secret $\rho_{j_0}(\mathbf{s})$ and error distribution χ^u .

The trace map Let us first suppose that we have polynomial LWE samples (\mathbf{a}, \mathbf{b}) where the ring is also a field isomorphic to \mathbb{F}_{q^u} as is the case with the samples $(\tilde{\mathbf{a}}_i, \tilde{\mathbf{b}}_i)$ (over multiple instances of suitable (\mathbf{a}, \mathbf{b})). We consider the trace map $\text{Tr}: \mathbb{F}_{q^u} \rightarrow \mathbb{F}_{q^d}$ for some $d \mid u$ in this case. For convenience we denote by S_q the subfield \mathbb{F}_{q^d} in this case.

Alternatively, if we have samples in R_q where $R = \mathcal{O}_K$ for $K = \mathbb{Q}[X]/(f(X))$ we let $L \subseteq K$ be the $2d$ th-cyclotomic subfield and note that the trace map $\text{Tr}_{K/L}: K \rightarrow L$ respects reduction modulo q . That is for $\mathbf{a}, \mathbf{b} \in R$ we have $\text{Tr}_{K/L}(\mathbf{a} + q\mathbf{b}) = \text{Tr}_{K/L}(\mathbf{a}) + q\mathbf{b}'$ for some $\mathbf{b}' \in \mathcal{O}_L$. This time, denote by S_q the subset of R_q corresponding to $\mathcal{O}_L/q\mathcal{O}_L$, then we have a well defined trace map $\text{Tr}: R_q \rightarrow S_q$ (again see Section 4.6 for more detail). We also set $u = n$ in this case.

We specifically use the same notation Tr and S_q in both cases as the maps behave in the same manner so we can give a unified treatment. Whichever case we are in, we can apply the trace map to the samples to give

$$\underbrace{\text{Tr}(\mathbf{b})}_{\mathbf{b}'} = \text{Tr}(\mathbf{a}\mathbf{s} + \mathbf{e}) = \text{Tr}(\mathbf{a}\mathbf{s}) + \text{Tr}(\mathbf{e}) = \underbrace{\text{Tr}(\mathbf{a})}_{\mathbf{a}'} \underbrace{\frac{\text{Tr}(\mathbf{a}\mathbf{s})}{\text{Tr}(\mathbf{a})}}_{\mathbf{s}'} + \underbrace{\text{Tr}(\mathbf{e})}_{\mathbf{e}'}$$

We need \mathbf{s}' to depend only on \mathbf{s} and not on \mathbf{a} . To achieve this one notes that, for $\lambda \in S_q$, $\text{Tr}(\lambda\mathbf{a}) = \lambda\text{Tr}(\mathbf{a})$ so if every sample has \mathbf{a} in some fixed coset of S_q^\times this is true. In particular, suppose they all lie in the coset vS_q^\times and $\text{Tr}(v) \in S_q^\times$, then we have $\mathbf{s}' = \text{Tr}(v\mathbf{s})/\text{Tr}(v)$.

Looking at the new error $\mathbf{e}' = \text{Tr}\left(\sum_{i=0}^{u-1} e_i X^i\right) = \sum_{i=0}^{u-1} e_i \text{Tr}(X^i)$, one notes

$$\text{Tr}(X^i) = \begin{cases} 0 & \text{if } i \not\equiv 0 \pmod{u/d} \\ \frac{u}{d} X^i & \text{otherwise,} \end{cases} \quad (3)$$

irrespective of which case we are in and hence $\mathbf{e}' = \frac{u}{d} \sum_{i=0}^{d-1} e_{iu/d} X^{iu/d}$. Thus the (non-zero) coefficients of \mathbf{e}' are $\frac{u}{d}$ times a sample from the original coefficient error distribution. Thus, if one instead defines new samples $(\tilde{\mathbf{a}}, \tilde{\mathbf{b}}) := (\frac{d}{u} \text{Tr}(\mathbf{a}), \frac{d}{u} \text{Tr}(\mathbf{b}))$ we have polynomial LWE samples with secret \mathbf{s}' and errors with coefficients sampled from χ .

Solving this new instance of the polynomial LWE problem using some other method gives us \mathbf{s}' but we cannot recover \mathbf{s} immediately. For this one notes that $(\mathbf{a}, -\mathbf{b}X^{n-j})$ is a sample with secret $-\mathbf{s}X^{n-j}$ and more importantly the same error distribution for $j = 1, \dots, d-1$ after taking the trace, namely we will have $\frac{d}{u} \text{Tr} \left(-\sum_{i=0}^{u-1} e_i X^{n-j+i} \right) = \sum_{i=0}^{d-1} e_{iu/d+j} X^{iu/d}$.⁹ We can therefore apply the same approach to recover $\varsigma_j := \text{Tr}(-v\mathbf{s}X^{n-j})/\text{Tr}(v)$ for each $j = 0, \dots, u/d-1$. Using (3) we can see that

$$\frac{d}{u} \text{Tr}(v) \sum_{j=0}^{u/d-1} \varsigma_j X^j = \frac{d}{u} \sum_{j=0}^{u/d-1} \text{Tr}(-v\mathbf{s}X^{n-j}) X^j = v\mathbf{s}$$

and then multiplying by v^{-1} gives us the secret \mathbf{s} in the full field.

Putting it together We have seen that one can reduce the polynomial LWE problem with $f = X^n + 1$, n a power of two, and $q \equiv 1 \pmod{4}$ to one in a finite field using the Chinese remainder theorem, assuming that all but one of the components of the ‘ \mathbf{a} -part’ of the samples are zero. Further, if required we can reduce the problem in a finite field to one in a subfield if all the samples have their ‘ \mathbf{a} -part’ in the same coset of the multiplicative group of the subfield.

Alternatively, for the same f and now q any odd prime that is unramified we can use the cyclotomic subfield structure to reduce the problem to one using a smaller power-of-two cyclotomic field if we can recover enough samples for which the ‘ \mathbf{a} -part’ lies in a fixed coset of the ring of integers of this smaller cyclotomic field.

Such conditions can be satisfied by a suitable choice of block size β and ordered basis with respect to which BKW reduction is performed. In particular, to use the above mentioned subfields we must choose a block size $\beta \mid d$. Further, the basis should be ordered globally with respect to the CRT mapping, if used, and then we use the basis $1, X, X^2, \dots, X^{u-1}$ ordered according to the following rules:

- if one of X^i and X^j generates a strictly smaller subfield¹⁰ than the other, then it comes after the other;
- if X^i and X^j generate the same subfield then the ordering is the standard one.

⁹ Stange instead multiplies the \mathbf{b} by X^j rather than its inverse which leads to a distortion of the error when applying the trace map.

¹⁰ The subfield is modulo q in the case of using the CRT decomposition and of K otherwise.

If using the CRT decomposition, we will recover samples in the final BKW list whose first component's CRT decomposition is all zero except in the last coordinate which will lie in \mathbb{F}_{q^β} . Using such samples, we can recover the final CRT component of the secret (in \mathbb{F}_{q^u}). With this knowledge we can find the appropriate BKW list to repeat the process for the previous CRT components until we have recovered all components of the original secret.

Otherwise, we will recover samples in the final BKW list whose 'a-part' lies in S_q , while not guaranteed to be in S_q^\times , this is enough to be able to apply the trace map as in the previous section to be able to recover the original secret by solving n/d polynomial LWE problems in S_q . This method has the advantage that no back-substitution of the partial secret is required so all smaller dimensional problems can be solved in parallel, something which is not true when using the CRT decomposition.

We remark that the only growth in the size of the error coefficients occurs during the BKW reduction step and depends exponentially on the number of blocks n/β used. Further, it is possible to use any of the improvements to the basic BKW reduction algorithm together with this attack as well as using blocks of differing size though we do not expand on this here.

Generalisation

We attempt to generalise Stange's attack to Ideal-LWE, remaining in the case $m = 1$ for simplicity, as we now explain. Firstly, we consider in which cases one can successfully apply the CRT map. Secondly, we look at what to replace the trace map with when it does not exist due to the problem no longer being defined in a finite field or with respect to a power-of-two cyclotomic number field where the trace map is very well behaved. We will see that we have differing levels of success in both parts.

CRT reduction To be able to utilize the CRT decomposition we assume that we can write $f = \prod_{i=1}^t f_i + \kappa g$ for polynomials $f_i \in \mathbb{Z}[X]$ of the form $f_i = X^u - c_i$ for distinct $c_i \in \mathbb{Z}$ and a polynomial $\kappa \in \mathbb{Z}[X]$. Assume further that as ideals (f_i, g) can be written as (a_i, r_i) where $a_i \mid a$. We also require that the $t \times t$ Vandermonde matrix with (i, j) th entry c_j^{i-1} is invertible modulo a ; that is its determinant $\prod_{1 \leq i < j \leq t} (c_j - c_i)$ is coprime to a , hence so is each $c_j - c_i$. Further, define integers $\mu_{i,j}$ which are thought of as the entries of the inverse of the above Vandermonde matrix so that we have

$$\sum_{u=1}^t \mu_{u,i} c_u^{j-1} \equiv \delta_{i,j} \pmod{a},$$

where $\delta_{i,j}$ is the Kronecker delta function.

Then we define the maps $\rho_i: R_g \rightarrow \mathbb{Z}[X]/(f_i, g) \cong \mathbb{Z}_{a_i}[X]/(r_i(X))$ for $i = 1, \dots, t$ as reduction modulo f_i . We will actually need to use maps $\tilde{\rho}_i: R_g \rightarrow R_g$ which map \mathbf{a} to the representative of $\rho_i(\mathbf{a})$ in $\text{Rep}(\mathbb{Z}[X]/(f_i, g))$ and then reduce

this modulo g so that it is an element of R_g . Suppose $\mathbf{a} \in R_g$ is such that for some $1 < j_0 \leq t$ we have $\tilde{\rho}_i(\mathbf{a}) = 0$ for $1 \leq i < j_0$ and that we know the values of $\tilde{\rho}_i(\mathbf{s})$ for $j_0 < i \leq t$. For $\mathbf{b} \equiv \mathbf{a} + \mathbf{e} \pmod{gR}$, we then have $\tilde{\rho}_i(\mathbf{b}) = \tilde{\rho}_i(\mathbf{e})$ for $1 \leq i < j_0$ but we don't necessarily have $\tilde{\rho}_i(\mathbf{b}) - \tilde{\rho}_i(\tilde{\rho}_i(\mathbf{a})\tilde{\rho}_i(\mathbf{s})) = \tilde{\rho}_i(\mathbf{e})$ for $j_0 \leq i \leq t$ as this equation only holds modulo the ideal $(f_i, g) = (a_i, r_i)$. By considering degrees, we can instead write

$$\tilde{\rho}_i(\mathbf{b}) - \tilde{\rho}_i(\tilde{\rho}_i(\mathbf{a})\tilde{\rho}_i(\mathbf{s})) = \tilde{\rho}_i(\mathbf{e}) + a_i \Delta_i,$$

with Δ_i a polynomial of degree at most $\deg(r_j) - 1$ and with small coefficients. This is our first potential obstruction.

Next, we define

$$(\tilde{\mathbf{a}}_i, \tilde{\mathbf{b}}_i) := \left(\mu_{j_0, i} \tilde{\rho}_{j_0}(\mathbf{a}), \sum_{j=1}^t \mu_{j, i} \tilde{\rho}_j(\mathbf{b}) - \sum_{j=j_0+1}^t \mu_{j, i} \tilde{\rho}_j(\tilde{\rho}_j(\mathbf{a})\tilde{\rho}_j(\mathbf{s})) \right).$$

We then have $\tilde{\mathbf{b}}_i - \tilde{\mathbf{a}}_i \tilde{\rho}_{j_0}(\mathbf{s}) \equiv \sum_{j=1}^t \mu_{j, i} \tilde{\rho}_j(\mathbf{e}) + \sum_{j=j_0}^t \mu_{j, i} a_j \Delta_j \pmod{(f_{j_0}, g)}$. We further consider the term $\sum_{j=1}^t \mu_{j, i} \tilde{\rho}_j(\mathbf{e})$; to this end we define the polynomials $\epsilon_\ell := \sum_{i=0}^{k-1} e_{(\ell-1)k+i} X^i$ where the e_i are the coefficients of \mathbf{e} . It is clear that $\tilde{\rho}_j(\mathbf{e}) \equiv \sum_{\ell=1}^t \epsilon_\ell c_j^{\ell-1} \pmod{(f_j, g)}$ so we define κ_j such that we have

$$\tilde{\rho}_j(\mathbf{e}) = \sum_{\ell=1}^t \epsilon_\ell c_j^{\ell-1} + a_j \kappa_j$$

in R_g . Ideally, we want $\kappa_j = 0$ for all j but this is not true in general so we have a second obstruction. We therefore have

$$\sum_{j=1}^t \mu_{j, i} \tilde{\rho}_j(\mathbf{e}) = \sum_{\ell=1}^t \epsilon_\ell \sum_{j=1}^t \mu_{j, i} c_j^{\ell-1} + \sum_{j=1}^t \mu_{j, i} a_j \kappa_j = \epsilon_i + \sum_{j=1}^t \mu_{j, i} a_j \kappa_j$$

since by definition $\sum_{j=1}^t \mu_{j, i} c_j^{\ell-1} = \delta_{i, \ell}$. Thus we can conclude that

$$\tilde{\mathbf{b}}_i - \tilde{\mathbf{a}}_i \tilde{\rho}_{j_0}(\mathbf{s}) = \epsilon_i + \sum_{j=1}^t \mu_{j, i} a_j \kappa_j + \sum_{j=j_0}^t \mu_{j, i} a_j \Delta_j.$$

Considering the sample $(\tilde{\mathbf{a}}_i, \tilde{\mathbf{b}}_i)$ modulo (f_{j_0}, g) with secret $\rho_{j_0}(\mathbf{s})$ we see that the error is given by

$$\epsilon_i + \sum_{j=1}^t \mu_{j, i} a_j \kappa_j + \sum_{j=j_0+1}^t \mu_{j, i} a_j \Delta_j \pmod{(f_{j_0}, g)} \quad (4)$$

which in general is not small due to the last two terms.

We note two cases where we can still apply this attack. Firstly, for the case that g is an integer we have that for each i , $a_i = a = g$, and hence the only

term left in the error in (4) is ϵ_i which has small coefficients when lifted to $\mathbb{Z}[X]/(f_{j_0}(X))$. We remark that there is no reason to assume g is prime, though the Vandermonde matrix is less likely to be invertible if g has many small factors.

Secondly, we look at the case of $g = X - b$; in this case $a = \prod_i a_i$ and the fact that the Vandermonde matrix is assumed to be invertible implies that the $a_i = b^u - c_i$ are coprime. We have that $\epsilon_\ell \approx e_{\ell u-1} b^{u-1}$ and hence $\tilde{\rho}_j(\mathbf{e}) = \sum_{\ell=1}^t \epsilon_\ell c_j^{\ell-1} \approx \sum_{\ell=1}^t e_{\ell u-1} b^{u-1} c_j^{\ell-1} \approx e_{n-1} b^{u-1} c_j^t$ which may be relatively small compared to a_{j_0} when c_j^t is of the order of b or smaller. In other words, if this is the case then κ_j is small.

Further, we note that $\sum_{j=1}^t \mu_{j,i} a_j \equiv \sum_{j=1}^t \mu_{j,i} (c_{j_0} - c_j) \equiv \delta_{1,i} c_{j_0} - \delta_{2,i} \pmod{a_{j_0}}$. Unfortunately, in general the terms $\kappa_j + \Delta_j$ are not equal so this is not very useful however when $t = 2$ we see that the additional error term is equal to $(\kappa_{3-j_0} + \Delta_{3-j_0}) c_{j_0}$ if $i = 1$ and $-(\kappa_{3-j_0} + \Delta_{3-j_0})$ when $i = 2$; thus choosing the samples with $i = 2$ still gives small error terms assuming κ_j is small for all j .

Requirements. *In summary, we have two situations in which we can use the CRT decomposition. Firstly, when g is an integer and f can be written as $\prod_{i=1}^t (X^u - c_i)$ modulo g for integer c_i such that the $c_i - c_j$ are invertible modulo g for $j \neq i$. Secondly, when $g = X - b$ with $f \equiv (X^u - c_1)(X^u - c_2) \pmod{g}$ for integers c_1, c_2 such that $c_1 - c_2$ invertible in R_g and c_1^2 and c_2^2 are of the order of b in magnitude or smaller. Finally, in both cases we require a large number of samples to be able to run the BKW reduction.*

Generalised trace reduction For this approach we assume that we can write $f(X) = \tilde{f}(X^d)$ and $g(X) = \tilde{g}(X^d)$ for some $d > 1$. Let $\deg \tilde{f} = \tilde{n}$. We first define the map $\theta: \mathbb{Z}[X] \rightarrow \mathbb{Z}[X]$ by $\theta(\sum_i a_i X^i) = \sum_{i: d|i} a_i X^i$. Clearly the map is a homomorphism of abelian groups (with respect to addition). We define $C \subseteq \mathbb{Z}[X]$ to be the subring of elements for which θ is the identity map, i.e. C is the set of elements which can be written as an integer sum of powers of X^d . Then for $\mathbf{c} = \sum_j c_j X^{dj} \in C$ and $\mathbf{a} = \sum_i a_i X^i$ we have $\mathbf{a}\mathbf{c} = \sum_i \sum_j a_i c_j X^{i+dj}$ which implies

$$\theta(\mathbf{a}\mathbf{c}) = \sum_{i: d|i} \sum_j a_i c_j X^{i+dj} = \theta(\mathbf{a})\mathbf{c}.$$

Since $f, g \in C$, we see that θ respects reduction modulo f and g , namely that $\theta(\mathbf{a} + \mathbf{k}f + \mathbf{l}g) = \theta(\mathbf{a}) + \theta(\mathbf{k})f + \theta(\mathbf{l})g$. We can therefore define $\tilde{\theta}: R_g \rightarrow R_g$ in the same manner as θ . Define the set S_g to be the subring of R_g which is fixed by $\tilde{\theta}$.

Now, suppose we have samples $(\mathbf{a}_i, \mathbf{b} = \mathbf{a}_i \mathbf{s} + \mathbf{e}_i) \in R_g \times R_g$ such that each \mathbf{a}_i lies in the same coset of R_g^\times / S_g^\times , say $v S_g^\times$. We can relax this slightly to assuming that we can write $\mathbf{a}_i = \mathbf{c}_i v$ for some $\mathbf{c}_i \in S_g$ and $v \in R_g^\times$. Then, assuming further that $\tilde{\theta}(v) \in S_g^\times$ so that $\mathbf{c}_i = \tilde{\theta}(\mathbf{a}_i) / \tilde{\theta}(v)$, we have

$$\tilde{\theta}(\mathbf{b}_i) = \tilde{\theta}(\mathbf{c}_i v \mathbf{s} + \mathbf{e}_i) = \mathbf{c}_i \tilde{\theta}(v \mathbf{s}) + \tilde{\theta}(\mathbf{e}_i) = \tilde{\theta}(\mathbf{a}_i) \frac{\tilde{\theta}(v \mathbf{s})}{\tilde{\theta}(v)} + \tilde{\theta}(\mathbf{e}_i).$$

Clearly, if the coefficients of \mathbf{e}_i are small then so are those of $\tilde{\theta}(\mathbf{e}_i)$. Thus the samples $(\tilde{\theta}(\mathbf{a}_i), \tilde{\theta}(\mathbf{b}_i))$ are from a smaller instance (the dimension has been reduced by a factor d) of the same problem with the secret now being $\tilde{\theta}(vs)/\tilde{\theta}(v)$.

More generally, assuming that the constant term of f is invertible modulo a , one can consider the samples $(\tilde{\theta}(\mathbf{a}_i), \tilde{\theta}(\mathbf{b}_i X^{-j}))$ for $0 \leq j < d$ which for the same reasoning can be seen to have secret $\sigma_j := \tilde{\theta}(vsX^{-j})/\tilde{\theta}(v)$ and errors $\tilde{\theta}(\mathbf{e}_i X^{-j})$. Due to the shape of f and g we will have $\tilde{\theta}(\mathbf{e}_i X^{-j}) \equiv \sum_{t=0}^{\tilde{n}-1} e_{i,t+d+j} X^{td} \pmod{(f, g)}$ where $e_{i,t}$ are the coefficients of \mathbf{e}_i . We can solve these smaller instances of the problem to recover the σ_j using any other suitable method. Once one has recovered the σ_j it is easy to see that $\tilde{\theta}(v) \sum_{j=0}^{d-1} \sigma_j X^j = vs$ and hence dividing this by v gives the full secret.

What remains is thus to derive samples which have $\mathbf{a}_i \in vS_g$ for some invertible v , for which $\tilde{\theta}(v)$ is also invertible, from ordinary samples more efficiently than simply computing a coset representative $\mathbf{a}/\tilde{\theta}(\mathbf{a})$ and looking for multiple collisions. This is where the BKW algorithm is used. One can perform BKW reduction until we find samples whose first component \mathbf{a} is in S_g since this can be viewed as an additive group; this amounts to taking $v = 1$. More generally, one could consider \mathbf{a}/v rather than \mathbf{a} for any suitable v and run the same reduction in parallel for differing values of v if computing power and memory is cheap; this will find samples with first component in the vS_g .

Requirements. *To be able to apply the generalised trace reduction, both f and g must be integer sums of powers of X^d for some $d > 1$. Also the constant term of f should be invertible modulo a . As for all BKW attacks one must have access to a large number of samples or use some type of sample amplification technique that is compatible with the ring structure.*