An Improved BKW Algorithm for LWE
with Applications to Cryptography and Lattices

Paul Kirchner[1] and Pierre-Alain Fouque[2]

[1] École normale supérieure
[2] Université de Rennes 1 and Institut universitaire de France
{paul.kirchner,pierre-alain.fouque}@ens.fr

Abstract. In this paper, we study the Learning With Errors problem and its binary variant, where secrets and errors are binary or taken in a small interval. We introduce a new variant of the Blum, Kalai and Wasserman algorithm, relying on a quantization step that generalizes and fine-tunes modulus switching. In general this new technique yields a significant gain in the constant in front of the exponent in the overall complexity. We illustrate this by solving within half a day a LWE instance with dimension $n = 128$, modulus $q = n^2$, Gaussian noise $\alpha = 1/(\sqrt{n/\pi} \log^2 n)$ and binary secret, using 2^{28} samples, while the previous best result based on BKW claims a time complexity of 2^{74} with 2^{60} samples for the same parameters.

We then introduce variants of BDD, GapSVP and UniqueSVP, where the target point is required to lie in the fundamental parallelepiped, and show how the previous algorithm is able to solve these variants in subexponential time. Moreover, we also show how the previous algorithm can be used to solve the BinaryLWE problem with n samples in subexponential time $2^{(\ln 2/2+o(1))n/\log\log n}$. This analysis does not require any heuristic assumption, contrary to other algebraic approaches; instead, it uses a variant of an idea by Lyubashevsky to generate many samples from a small number of samples. This makes it possible to asymptotically and heuristically break the NTRU cryptosystem in subexponential time (without contradicting its security assumption). We are also able to solve subset sum problems in subexponential time for density $o(1)$, which is of independent interest: for such density, the previous best algorithm requires exponential time. As a direct application, we can solve in subexponential time the parameters of a cryptosystem based on this problem proposed at TCC 2010.

1 Introduction

The Learning With Errors (LWE) problem has been an important problem in cryptography since its introduction by Regev in [46]. Many cryptosystems have been proven secure assuming the hardness of this problem, including Fully Homomorphic Encryption schemes [21,14]. The decision version of the problem can be described as follows: given m samples of the form $(\mathbf{a}, b) \in (\mathbb{Z}_q)^n \times \mathbb{Z}_q$, where $\mathbf{a}$ are uniformy distributed in $(\mathbb{Z}_q)^n$, distinguish whether b is uniformly chosen in $\mathbb{Z}_q$ or is equal to $\langle \mathbf{a}, \mathbf{s} \rangle + e$ for a fixed secret $\mathbf{s} \in (\mathbb{Z}_q)^n$ and e a noise value in $\mathbb{Z}_q$ chosen according to some probability distribution. Typically, the noise is sampled from some distribution concentrated on small numbers, such as a discrete Gaussian distribution with standard deviation αq for $\alpha = o(1)$. In the search version of the problem, the goal is to recover $\mathbf{s}$ given the promise that the sample instances come from the latter distribution. Initially, Regev showed that if $\alpha q \geq 2\sqrt{n}$, solving LWE on average is at least as hard as approximating lattice problems in the worst case to within $\tilde{\mathcal{O}}(n/\alpha)$ factors with a quantum algorithm. Peikert shows a classical reduction when the modulus is large $q \geq 2^n$ in [44]. Finally, in [13], Brakerski *et al.* prove that solving LWE instances with polynomial-size modulus in polynomial time implies an efficient solution to GapSVP.

There are basically three approaches to solving LWE: the first relies on lattice reduction techniques such as the LLL [32] algorithm and further improvements [15] as exposed in [34,35]; the second uses combinatorial techniques [12,47]; and the third uses algebraic techniques [9]. According to Regev in [1], the best known algorithm to solve LWE is the algorithm by Blum, Kalai and Wasserman in [12], originally proposed to solve the Learning Parities with Noise (LPN) problem, which can be viewed as a special case of LWE where $q = 2$. The time and memory requirements of this algorithm are both exponential for LWE and subexponential for

LPN in $2^{\mathcal{O}(n/\log n)}$. During the first stage of the algorithm, the dimension of $\mathbf{a}$ is reduced, at the cost of a (controlled) decrease of the bias of b. During the second stage, the algorithm distinguishes between LWE and uniform by evaluating the bias.

Since the introduction of LWE, some variants of the problem have been proposed in order to build more efficient cryptosystems. Some of the most interesting variants are Ring-LWE by Lyubashevsky, Peikert and Regev in [38], which aims to reduce the space of the public key using cyclic samples; and the cryptosystem by Döttling and Müller-Quade [18], which uses short secret and error. In 2013, Micciancio and Peikert [40] as well as Brakerski *et al.* [13] proposed a binary version of the LWE problem and obtained a hardness result.

Related Work. Albrecht *et al.* have presented an analysis of the BKW algorithm as applied to LWE in [4,5]. It has been recently revisited by Duc *et al.*, who use a multi-dimensional FFT in the second stage of the algorithm [19]. However, the main bottleneck is the first BKW step and since the proposed algorithms do not improve this stage, the overall asymptotic complexity is unchanged.

In the case of the BinaryLWE variant, where the error and secret are binary (or sufficiently small), Micciancio and Peikert show that solving this problem using $m = n(1 + \Omega(1/\log(n)))$ samples is at least as hard as approximating lattice problems in the worst case in dimension $\Theta(n/\log(n))$ with approximation factor $\tilde{\mathcal{O}}(\sqrt{n}q)$. We show in section B that existing lattice reduction techniques require exponential time. Arora and Ge describe a $2^{\tilde{\mathcal{O}}(\alpha q)^2}$-time algorithm when $q > n$ to solve the LWE problem [9]. This leads to a subexponential time algorithm when the error magnitude αq is less than $\sqrt{n}$. The idea is to transform this system into a noise-free polynomial system and then use root finding algorithms for multivariate polynomials to solve it, using either relinearization in [9] or Gröbner basis in [3]. In this last work, Albrecht *et al.* present an algorithm whose time complexity is $2^{\frac{(\omega+o(1))n \log \log \log n}{8 \log \log n}}$ when the number of samples $m = (1 + o(1))n \log \log n$ is super-linear, where $\omega < 2.3728$ is the linear algebra constant, under some assumption on the regularity of the polynomial system of equations; and when $m = \mathcal{O}(n)$, the complexity becomes exponential.

Contribution. Our first contribution is to present in a unified framework the BKW algorithm and all its previous improvements in the binary case [33,28,11,25] and in the general case [5]. We introduce a new quantization step, which generalizes modulus switching [5]. This yields a significant decrease in the constant of the exponential of the complexity for LWE. Moreover our proof does not require Gaussian noise, and does not rely on unproven independence assumptions. Our algorithm is also able to tackle problems with larger noise.

We then introduce generalizations of the BDD, GapSVP and UniqueSVP problems, and prove a reduction from these variants to LWE. When particular parameters are set, these variants impose that the lattice point of interest (the point of the lattice that the problem essentially asks to locate: for instance, in the case of BDD, the point of the lattice closest to the target point) lie in the fundamental parallelepiped; or more generally, we ask that the coordinates of this point relative to the basis defined by the input matrix $\mathbf{A}$ has small infinity norm, bounded by some value B. For small B, our main algorithm yields a subexponential-time algorithm for these variants of BDD, GapSVP and UniqueSVP.

Through a reduction to our variant of BDD, we are then able to solve the subset-sum problem in subexponential time when the density is $o(1)$, and in time $2^{(\ln 2/2+o(1))n/\log \log n}$ if the density is $\mathcal{O}(1/\log n)$. This is of independent interest, as existing techniques for density $o(1)$, based on lattice reduction, require exponential time. As a consequence, the cryptosystems of Lyubashevsky, Palacio and Segev at TCC 2010 [37] can be solved in subexponential time.

As another application of our main algorithm, we show that BinaryLWE with reasonable noise can be solved in time $2^{(\ln 2/2+o(1))n/\log \log n}$ instead of $2^{\Omega(n)}$; and the same complexity holds for secret of size up to $2^{\log^{o(1)} n}$. As a consequence, we can heuristically recover the secret polynomials $\mathbf{f}, \mathbf{g}$ of the NTRU problem in subexponential time $2^{(\ln 2/2+o(1))n/\log \log n}$ (without contradicting its security assumption). The heuristic assumption comes from the fact that NTRU samples are not random, since they are rotations of each other: the heuristic assumption is that this does not significantly hinder BKW-type algorithms. Note that there is a large value hidden in the $o(1)$ term, so that our algorithm does not yield practical attacks for recommended NTRU parameters.

2 Preliminaries

We identify any element of $\mathbb{Z}/q\mathbb{Z}$ to the smallest of its equivalence class, the positive one in case of tie. Any vector $\mathbf{x} \in \left(\mathbb{Z}/q\mathbb{Z}\right)^n$ has an Euclidean norm $||\mathbf{x}|| = \sqrt{\sum_{i=0}^{n-1} x_i^2}$ and $||\mathbf{x}||_\infty = \max_i |x_i|$. A matrix $\mathbf{B}$ can be Gram-Schmidt orthogonalized in $\widetilde{\mathbf{B}}$, and its norm $||\mathbf{B}||$ is the maximum of the norm of its columns. We denote by $(\mathbf{x}|\mathbf{y})$ the vector obtained as the concatenation of vectors $\mathbf{x}, \mathbf{y}$. Let $\mathbf{I}$ be the identity matrix and we denote by ln the neperian logarithm and log the binary logarithm. A lattice is the set of all integer linear combinations $\Lambda(\mathbf{b}_1, \ldots, \mathbf{b}_n) = \sum_i \mathbf{b_i} \cdot x_i$ (where $x_i \in \mathbb{Z}$) of a set of linearly independent vectors $\mathbf{b}_1, \ldots, \mathbf{b}_n$ called the basis of the lattice. If $\mathbf{B} = [\mathbf{b}_1, \ldots, \mathbf{b}_n]$ is the matrix basis, lattice vectors can be written as $\mathbf{Bx}$ for $\mathbf{x} \in \mathbb{Z}^n$. Its dual Λ^* is the set of $\mathbf{x} \in \mathbb{R}^n$ such that $\langle \mathbf{x}, \Lambda \rangle \subset \mathbb{Z}^n$. We have $\Lambda^{**} = \Lambda$. We borrow Bleichenbacher's definition of bias [42].

Definition 1. *The bias of a probability distribution ϕ over $\mathbb{Z}/q\mathbb{Z}$ is*

$$\mathbb{E}_{x \sim \phi}[\exp(2i\pi x/q)].$$

This definition extends the usual definition of the bias of a coin in $\mathbb{Z}/2\mathbb{Z}$: it preserves the fact that any distribution with bias b can be distinguished from uniform with constant probability using $\Omega(1/b^2)$ samples, as a consequence of Hoeffding's inequality; moreover the bias of the sum of two independent variable is still the product of their biases. We also have the following simple lemma:

Lemma 1. *The bias of the Gaussian distribution of mean 0 and standard deviation $q\alpha$ is $\exp(-2\pi^2\alpha^2)$.*

Proof. The bias is the value of the Fourier transform at $-1/q$. $\qquad\square$

We introduce a non standard definition for the LWE problem. However as a consequence of Lemma 1, this new definition naturally extends the usual Gaussian case (as well as its standard extensions such as the bounded noise variant [13, Definition 2.14]), and it will prove easier to work with. The reader can consider the distorsion parameter $\epsilon = 0$ as it is the case in other papers and a gaussian of standard deviation αq.

Definition 2. *Let $n \geq 0$ and $q \geq 2$ be integers. Given parameters α and ϵ, the LWE distribution is, for $\mathbf{s} \in (\mathbb{Z}/q\mathbb{Z})^n$, a distribution on pairs $(\mathbf{a}, b) \in (\mathbb{Z}/q\mathbb{Z})^n \times (\mathbb{R}/q\mathbb{Z})$ such that $\mathbf{a}$ is sampled uniformly, and for all $\mathbf{a}$,*
$$|\mathbb{E}[\exp(2i\pi(\langle \boldsymbol{a}, \boldsymbol{s} \rangle - b)/q)|\mathbf{a}]\exp(\alpha'^2) - 1| \leq \epsilon$$
for some universal $\alpha' \leq \alpha$.

For convenience, we define $\beta = \sqrt{n/2}/\alpha$. In the remainder, α is called the noise parameter[3], and ϵ the distortion parameter. Also, we say that a LWE distribution has a noise distribution ϕ if b is distributed as $\langle \mathbf{a}, \mathbf{s} \rangle + \phi$.

Definition 3. *The Decision-LWE problem is to distinguish a LWE distribution from the uniform distribution over $(\mathbf{a}, b)$. The Search-LWE problem is, given samples from a LWE distribution, to find $\mathbf{s}$.*

Definition 4. *The real λ_i is the radius of the smallest ball, centered in $\mathbf{0}$, such that it contains i vectors of the lattice Λ which are linearly independent.*

We define $\rho_s(\mathbf{x}) = \exp(-\pi||\mathbf{x}||^2/s^2)$ and $\rho_s(S) = \sum_{\mathbf{x} \in S} \rho_s(\mathbf{x})$ (and similarly for other functions). The discrete Gaussian distribution $D_{E,s}$ over a set E and of parameter s is such that the probability of $D_{E,s}(\mathbf{x})$ of drawing $\mathbf{x} \in E$ is equal to $\rho_s(\mathbf{x})/\rho_s(E)$. To simplify notation, we will denote by D_E the distribution $D_{E,1}$.

Definition 5. *The smoothing parameter η_ϵ of the lattice Λ is the smallest s such that $\rho_{1/s}(\Lambda^*) = 1 + \epsilon$.*

Now, we will generalize the BDD, UniqueSVP and GapSVP problems by using another parameter B that bounds the target lattice vector. For $B = 2^n$, we recover the usual definitions if the input matrix is reduced.

---

[3] Remark that it differs by a constant factor from other authors' definition of α.

Definition 6. *The* $\mathsf{BDD}_{B,\beta}^{||\cdot||_\infty}$ *(resp.* $\mathsf{BDD}_{B,\beta}^{||\cdot||}$) *problem is, given a basis* $\mathbf{A}$ *of the lattice* Λ, *and a point* $\mathbf{x}$ *such that* $||\mathbf{As} - \mathbf{x}|| \le \lambda_1/\beta < \lambda_1/2$ *and* $||\mathbf{s}||_\infty \le B$ *(resp.* $||\mathbf{s}|| \le B$), *to find* $\mathbf{s}$.

Definition 7. *The* $\mathsf{UniqueSVP}_{B,\beta}^{||\cdot||_\infty}$ *(resp.* $\mathsf{UniqueSVP}_{B,\beta}^{||\cdot||}$) *problem is, given a basis* $\mathbf{A}$ *of the lattice* Λ, *such that* $\lambda_2/\lambda_1 \ge \beta$ *and there exists* $\mathbf{s}$ *such that* $||\mathbf{As}|| = \lambda_1$ *with* $||\mathbf{s}||_\infty \le B$ *(resp.* $||\mathbf{s}|| \le B$), *to find* $\mathbf{s}$.

Definition 8. *The* $\mathsf{GapSVP}_{B,\beta}^{||\cdot||_\infty}$ *(resp.* $\mathsf{GapSVP}_{B,\beta}^{||\cdot||}$) *problem is, given a basis* $\mathbf{A}$ *of the lattice* Λ *to distinguish between* $\lambda_1(\Lambda) \ge \beta$ *and if there exists* $\mathbf{s} \ne \mathbf{0}$ *such that* $||\mathbf{s}||_\infty \le B$ *(resp.* $||\mathbf{s}|| \le B$) *and* $||\mathbf{As}|| \le 1$.

Definition 9. *Given two probability distributions* P *and* Q *on a finite set* S, *the Kullback-Leibler (or* KL) *divergence between* P *and* Q *is*

$$D_{\mathsf{KL}}(P||Q) = \sum_{x \in S} \ln\left(\frac{P(x)}{Q(x)}\right) P(x) \quad with \quad \ln(x/0) = +\infty \quad if \quad x > 0.$$

The following two lemmata are proven in [45] :

Lemma 2. *Let* P *and* Q *be two distributions over* S, *such that for all* x, $|P(x) - Q(x)| \le \delta(x)P(x)$ *with* $\delta(x) \le 1/4$. *Then :*

$$D_{\mathsf{KL}}(P||Q) \le 2 \sum_{x \in S} \delta(x)^2 P(x).$$

Lemma 3. *Let* A *be an algorithm which takes as input* m *samples of* S *and outputs a bit. Let* x *(resp.* y) *be the probability that it returns* 1 *when the input is sampled from* P *(resp.* Q). *Then :*

$$|x - y| \le \sqrt{m D_{\mathsf{KL}}(P||Q)/2}.$$

Finally, we say that an algorithm has a negligible probability of failure if its probability of failure is $2^{-\Omega(n)}$. [4]

2.1 Secret-Error Switching

At a small cost in samples, it is possible to reduce any LWE distribution with noise distribution ϕ to an instance where the secret follows the rounded distribution, defined as $\lfloor\phi\rceil$ [7,13].

Theorem 1. *Given an oracle that solves* LWE *with* m *samples in time* t *with the secret coming from the rounded error distribution, it is possible to solve* LWE *with* $m + \mathcal{O}(n \log \log q)$ *samples with the same error distribution (and any distribution on the secret) in time* $t + \mathcal{O}(mn^2 + (n \log \log q)^3)$, *with negligible probability of failure.*

Furthermore, if q *is prime, we lose* $n + k$ *samples with probability of failure bounded by* q^{-1-k}.

Proof. First, select an invertible matrix $\mathbf{A}$ from the vectorial part of $\mathcal{O}(n \log \log q)$ samples in time $\mathcal{O}((n \log \log q)^3)$ [13, Claim 2.13].

Let $\boldsymbol{b}$ be the corresponding rounded noisy dot products. Let $\mathbf{s}$ be the LWE secret and $\mathbf{e}$ such that $\mathbf{As} + \mathbf{e} = \mathbf{b}$. Then the subsequent m samples are transformed in the following way. For each new sample $(\mathbf{a}', b')$ with $b' = \langle \mathbf{a}', \mathbf{s} \rangle + e'$, we give the sample $(-{}^t\mathbf{A}^{-1}\mathbf{a}', b' - \langle {}^t\mathbf{A}^{-1}\mathbf{a}', \mathbf{b} \rangle)$ to our LWE oracle.

Clearly, the vectorial part of the new samples remains uniform and since

$$b' - \langle {}^t\mathbf{A}^{-1}\mathbf{a}', \mathbf{b} \rangle = \langle -{}^t\mathbf{A}^{-1}\mathbf{a}', \mathbf{b} - \mathbf{As} \rangle + b' - \langle \mathbf{a}', \mathbf{s} \rangle = \langle -{}^t\mathbf{A}^{-1}\mathbf{a}', \mathbf{e} \rangle + e'$$

the new errors follow the same distribution as the original, and the new secret is $\mathbf{e}$. Hence the oracle outputs $\mathbf{e}$ in time t, and we can recover $\mathbf{s}$ as $\mathbf{s} = \mathbf{A}^{-1}(\mathbf{b} - \mathbf{e})$.

If q is prime, the probability that the $n + k$ first samples are in some hyperplane is bounded by $q^{n-1}q^{-n-k} = q^{-1-k}$. □

---

[4] Some authors use another definition.

2.2   Low dimension algorithms

Our main algorithm will return samples from a LWE distribution, while the bias decreases. We describe two fast algorithms when the dimension is small enough.

Theorem 2. *If $n = 0$ and $m = k/b^2$, with b smaller than the real part of the bias, the Decision-LWE problem can be solved with advantage $1 - 2^{-\Omega(k)}$ in time $\mathcal{O}(m)$.*

Proof. The algorithm DISTINGUISH computes $x = \frac{1}{m} \sum_{i=0}^{m-1} \cos(2i\pi b_i/q)$ and returns the boolean $x \geq b/2$. If we have a uniform distribution then the average of x is 0, else it is larger than $b/2$. The Hoeffding inequality shows that the probability of $|x - \mathbb{E}[x]| \geq b/2$ is $2^{-k/8}$, which gives the result. $\qquad\square$

Algorithm 1 FindSecret

> **function** FINDSECRET($\mathcal{L}$)
>   **for all** $(\mathbf{a}, b) \in \mathcal{L}$ **do**
> $f[\mathbf{a}] \leftarrow f[\mathbf{a}] + \exp(2i\pi b/q)$
>   **end for**
> $t \leftarrow$ FASTFOURIERTRANSFORM(f)
>   **return** $\arg\max_{\mathbf{s} \in (\mathbb{Z}/q\mathbb{Z})^n} \Re(t[\mathbf{s}])$
> **end function**

Lemma 4. *For all $\mathbf{s} \neq \mathbf{0}$, if $\mathbf{a}$ is sampled uniformly, $\mathbb{E}[\exp(2i\pi\langle\mathbf{a}, \mathbf{s}\rangle/q)] = 0$.*

Proof. Multiplication by s_0 in $\mathbb{Z}_q$ is $\gcd(s_0, q)$-to-one because it is a group morphism, therefore $a_0 s_0$ is uniform over $\gcd(s_0, q)\mathbb{Z}_q$. Thus, by using $k = \gcd(q, s_0, \ldots, s_{n-1}) < q$, $\langle\mathbf{a}, \mathbf{s}\rangle$ is distributed uniformly over $k\mathbb{Z}_q$ so

$$\mathbb{E}[\exp(2i\pi\langle\mathbf{a}, \mathbf{s}\rangle/q)] = \frac{q}{k} \sum_{j=0}^{q/k-1} \exp(2i\pi jk/q) = 0. \qquad\square$$

Theorem 3. *The algorithm* FINDSECRET, *when given $m > (8n\log q + k)/b^2$ samples from a LWE problem with bias whose real part is superior to b returns the correct secret in time $\mathcal{O}(m + n\log^2(q)q^n)$ except with probability $2^{-\Omega(k)}$.*

Proof. The fast Fourier transform needs $\mathcal{O}(nq^n)$ operations on numbers of bit size $\mathcal{O}(\log(q))$. The Hoeffding inequality shows that the difference between $t[\mathbf{s}']$ and $\mathbb{E}[\exp(2i\pi(b - \langle\mathbf{a}, \mathbf{s}'\rangle)/q)]$ is at most $b/2$ except with probability at most $2\exp(-mb^2/2)$. It holds for all $\mathbf{s}'$ except with probability at most $2q^n\exp(-mb^2/2) = 2^{-\Omega(k)}$ using the union bound. Then $t[\mathbf{s}] \geq b - b/2 = b/2$ and for all $\mathbf{s}' \neq \mathbf{s}$, $t[\mathbf{s}'] < b/2$ so the algorithm returns $\mathbf{s}$. $\qquad\square$

3   Main algorithm

In this section, we present our main algorithm, prove its asymptotical complexity, and present practical results in dimension $n = 128$.

3.1 Rationale

A natural idea in order to distinguish between an instance of LWE (or LPN) and a uniform distribution is to select some k samples that add up to zero, yielding a new sample of the form $(\mathbf{0}, e)$. It is then enough to distinguish between e and a uniform variable. However, if δ is the bias of the error in the original samples, the new error e has bias δ^k, hence roughly δ^{-2k} samples are necessary to distinguish it from uniform. Thus it is crucial that k be as small a possible.

The idea of the algorithm by Blum, Kalai and Wasserman BKW is to perform "blockwise" Gaussian elimination. The n coordinates are divided into k blocks of length $b = n/k$. Then, samples that are equal on the first b coordinates are substracted together to produce new samples that are zero on the first block. This process is iterated over each consecutive block. Eventually samples of the form $(\mathbf{0}, e)$ are obtained.

Each of these samples ultimately results from the addition of 2^k starting samples, so k should be at most $\mathcal{O}(\log(n))$ for the algorithm to make sense. On the other hand $\Omega(q^b)$ data are clearly required at each step in order to generate enough collisions on b consecutive coordinates of a block. This naturally results in a complexity roughly $2^{(1+o(1))n/\log(n)}$ in the original algorithm for LPN. This algorithm was later adapted to LWE in [4], and then improved in [5].

The idea of the latter improvement is to use so-called "lazy modulus switching". Instead of finding two vectors that are equal on a given block in order to generate a new vector that is zero on the block, one uses vectors that are merely close to each other. This may be seen as performing addition modulo p instead of q for some $p < q$, by rounding every value $x \in \mathbb{Z}_q$ to the value nearest xp/q in $\mathbb{Z}_p$. Thus at each step of the algorithm, instead of generating vectors that are zero on each block, small vectors are produced. This introduces a new "rounding" error term, but essentially reduces the complexity from roughly q^b to p^b. Balancing the new error term with this decrease in complexity results in a significant improvement.

However it may be observed that this rounding error is much more costly for the first few blocks than the last ones. Indeed samples produced after, say, one iteration step are bound to be added together 2^{a-1} times to yield the final samples, resulting in a corresponding blowup of the rounding error. By contrast, later terms will undergo less additions. Thus it makes sense to allow for progressively coarser approximations (i.e. decreasing the modulus) at each step. On the other hand, to maintain comparable data requirements to find collisions on each block, the decrease in modulus is compensated by progressively longer blocks.

What we propose here is a more general view of the BKW algorithm that allows for this improvement, while giving a clear view of the different complexity costs incurred by various choice of parameters. Balancing these terms is the key to finding an optimal complexity. We forego the "modulus switching" point of view entirely, while retaining its core ideas. The resulting algorithm generalizes several variants of BKW, and will be later applied in a variety of settings.

Also, each time we combine two samples, we never use again these two samples so that the combined samples are independent. Previous works used repeatedly one of the two samples, so that independency can only be attained by repeating the entire algorithm for each sample needed by the distinguisher, as was done in [12].

3.2 Quantization

The goal of quantization is to associate to each point of $\mathbb{R}^k$ a center from a *small* set, such that the expectancy of the distance between a point and its center is small. We will then be able to produce small vectors by substracting vectors associated to the same center.

Modulus switching amounts to a simple quantizer which rounds every coordinate to the nearest multiple of some constant. Our proven algorithm uses a similar quantizer, except the constant depends on the index of the coordinate.

It is possible to decrease the average distance from a point to its center by a constant factor for large moduli [24], but doing so would complicate our proof without improving the leading term of the complexity. When the modulus is small, it might be worthwhile to use error-correcting codes as in [25].

3.3 Main Algorithm

Let us denote by $\mathcal{L}_0$ the set of starting samples, and $\mathcal{L}_i$ the sample list after i reduction steps. The numbers $d_0 = 0 \leq d_1 \leq \cdots \leq d_k = n$ partition the n coordinates of sample vectors into k buckets. Let $\mathbf{D} = (D_0, \ldots, D_{k-1})$ be the vector of quantization coefficients associated to each bucket.

Algorithm 2 Main resolution

1: **function** REDUCE($\mathcal{L}_{in}, D_i, d_i, d_{i+1}$)
2:     $\mathcal{L}_{out} \leftarrow \varnothing$
3:     $t[] \leftarrow \varnothing$
4:     **for all** $(\mathbf{a}, b) \in \mathcal{L}_{in}$ **do**
5:         $\mathbf{r} = \lfloor \frac{(\mathbf{a}_{d_i}, \ldots, \mathbf{a}_{d_{i+1}-1})}{D} \rceil$
6:         **if** $t[\mathbf{r}] = \varnothing$ **then**
7:             $t[\mathbf{r}] \leftarrow (\mathbf{a}, b)$
8:         **else**
9:             $\mathcal{L}_{out} \leftarrow \mathcal{L}_{out} :: \{(\mathbf{a}, b) - t[r]\}$
10:           $t[\mathbf{r}] \leftarrow \varnothing$
11:         **end if**
12:     **end for**
13:     **return** $\mathcal{L}_{out}$
14: **end function**
15: **function** SOLVE($\mathcal{L}_0, \mathbf{D}, (d_i)$)
16:     **for** $0 \leq i < k$ **do**
17:         $\mathcal{L}_{i+1} \leftarrow$ REDUCE($\mathcal{L}_i, D_i, d_i, d_{i+1}$)
18:     **end for**
19:     **return** DISTINGUISH($\{b | (\mathbf{a}, b) \in \mathcal{L}_k\}$)
20: **end function**

In order to allow for a uniform presentation of the BKW algorithm, applicable to different settings, we do not assume a specific distribution on the secret. Instead, we assume there exists some *known* $\mathbf{B} = (B_0, \ldots, B_{n-1})$ such that $\sum_i (s_i/B_i)^2 \leq n$. Note that this is in particular true if $|s_i| \leq B_i$. We shall see how to adapt this to the standard Gaussian case later on. Without loss of generality, $\mathbf{B}$ is non increasing.

There are a phases in our reduction : in the i-th phase, the coordinates from d_i to d_{i+1} are reduced. We define $m = |\mathcal{L}_0|$.

Lemma 5. SOLVE *terminates in time* $\mathcal{O}(mn \log q)$.

Proof. The REDUCE algorithm clearly runs in time $\mathcal{O}(|\mathcal{L}|n \log q)$. Moreover, $|\mathcal{L}_{i+1}| \leq |\mathcal{L}_i|/2$ so that the total running time of SOLVE is $\mathcal{O}(n \log q \sum_{i=0}^{k} m/2^i) = \mathcal{O}(mn \log q)$. $\square$

Lemma 6. *Write $\mathcal{L}'_i$ for the samples of $\mathcal{L}_i$ where the first d_i coordinates of each sample vector have been truncated. Assume $|s_j|D_i < 0.23q$ for all $d_i \leq j < d_{i+1}$. If $\mathcal{L}'_i$ is sampled according to the LWE distribution of secret $\mathbf{s}$ and noise parameters α and $\epsilon \leq 1$, then $\mathcal{L}'_{i+1}$ is sampled according to the LWE distribution of the truncated secret with parameters:*

$$\alpha'^2 = 2\alpha^2 + 4\pi^2 \sum_{j=d_i}^{d_{i+1}-1} (s_j D_i/q)^2 \quad \text{and} \quad \epsilon' = 3\epsilon.$$

On the other hand, if $D_i = 1$, then $\alpha'^2 = 2\alpha^2$.

Proof. The independence of the outputted samples and the uniformity of their vectorial part are clear. Let $(\mathbf{a}, b)$ be a sample obtained by substracting two samples from $\mathcal{L}_i$. For $\mathbf{a}'$ the vectorial part of a sample, define

7

$\epsilon(\mathbf{a}')$ such that $\mathbb{E}[\exp(2i\pi(\langle \mathbf{a}', \mathbf{s}\rangle - b')/q)|\mathbf{a}'] = (1 + \epsilon(\mathbf{a}'))\exp(-\alpha^2)$. By definition of LWE, $|\epsilon(\mathbf{a}')| \leq \epsilon$, and by independence:

$$\mathbb{E}[\exp(2i\pi(\langle \mathbf{a}, \mathbf{s}\rangle - b)/q)|\mathbf{a}] = \exp(-2\alpha^2)\mathbb{E}_{\mathbf{a}'-\mathbf{a}''=\mathbf{a}}[(1 + \epsilon(\mathbf{a}'))(1 + \epsilon(\mathbf{a}''))],$$

with $|\mathbb{E}_{\mathbf{a}'-\mathbf{a}''=\mathbf{a}}[(1 + \epsilon(\mathbf{a}'))(1 + \epsilon(\mathbf{a}''))] - 1| \leq 3\epsilon$.

Thus we computed the noise corresponding to adding two samples of $\mathcal{L}_i$. To get the noise for a sample from $\mathcal{L}_{i+1}$, it remains to truncate coordinates from d_i to d_{i+1}. A straightforward induction on the coordinates shows that this noise is :

$$\exp(-2\alpha^2)\mathbb{E}_{\mathbf{a}'-\mathbf{a}''=\mathbf{a}}[(1 + \epsilon(\mathbf{a}'))(1 + \epsilon(\mathbf{a}''))] \prod_{j=d_i}^{d_{i+1}-1} \mathbb{E}[\exp(2i\pi\boldsymbol{a}_j\boldsymbol{s}_j/q)].$$

Indeed, if we denote by $\mathbf{a}^{(j)}$ the vector $\mathbf{a}$ where the first j coordinates are truncated and α_j the noise parameter of $\mathbf{a}^{(j)}$, we have:

$$|\mathbb{E}[\exp(2i\pi(\langle \mathbf{a}^{(j+1)}, \mathbf{s}^{(j+1)}\rangle - b)/q)|\mathbf{a}^{(j+1)}] - \exp(-\alpha_n^2)\mathbb{E}[\exp(2i\pi\boldsymbol{a}_j\boldsymbol{s}_j/q)]|$$
$$= |\mathbb{E}[\exp(-2i\pi\boldsymbol{a}_j\boldsymbol{s}_j/q)(\exp(2i\pi(\langle \mathbf{a}^{(j)}, \mathbf{s}^{(j)}\rangle - b)/q) - \exp(-\alpha_j^2))]|$$
$$\leq \epsilon'\exp(-\alpha_j^2)\mathbb{E}[\exp(2i\pi\boldsymbol{a}_j\boldsymbol{s}_j/q)].$$

It remains to compute $\mathbb{E}[\exp(2i\pi\boldsymbol{a}_j\boldsymbol{s}_j/q)]$ for $d_i \leq j < d_{i+1}$. Let $D = D_i$. The probability mass function of $\boldsymbol{a}_j$ is even, so $\mathbb{E}[\exp(2i\pi\boldsymbol{a}_j\boldsymbol{s}_j/q)]$ is real. Furthermore, since $|\boldsymbol{a}_j| \leq D$,

$$\mathbb{E}[\exp(2i\pi\boldsymbol{a}_j\boldsymbol{s}_j/q)] \geq \cos(2\pi\boldsymbol{s}_j D/q).$$

Simple function analysis shows that $\ln(\cos(2\pi x)) \geq -4\pi^2 x^2$ for $|x| \leq 0.23$, and since $|\boldsymbol{s}_j|D < 0.23q$, we get :

$$\mathbb{E}[\exp(2i\pi\boldsymbol{a}_j\boldsymbol{s}_j/q)] \geq \exp(-4\pi^2\boldsymbol{s}_j^2 D^2/q^2).$$

On the other hand, if $D_i = 1$ then $\boldsymbol{a}_j = 0$ and $\mathbb{E}[\exp(2i\pi\boldsymbol{a}_j\boldsymbol{s}_j/q)] = 1$. $\qquad\square$

Finding optimal parameters for BKW amounts to balancing various costs: the baseline number of samples required so that the final list $\mathcal{L}_k$ is non-empty, and the additional factor due to the need to distinguish the final error bias. This final bias itself comes both from the blowup of the original error bias by the BKW additions, and the "rounding errors" due to quantization. Balancing these costs essentially means solving a system.

For this purpose, it is convenient to set the overall target complexity as $2^{n(x+o(1))}$ for some x to be determined. The following auxiliary lemma essentially gives optimal values for the parameters of SOLVE assuming a suitable value of x. The actual value of x will be decided later on.

Lemma 7. *Pick some value x (dependent on LWE parameters). Choose:*

$$k \leq \left\lfloor \log\left(\frac{nx}{6\alpha^2}\right)\right\rfloor \qquad\qquad m = n2^k2^{nx}$$

$$D_i \leq \frac{q\sqrt{x/6}}{\pi B_{d_i}2^{(a-i+1)/2}} \qquad\qquad d_{i+1} = \min\left(d_i + \left\lfloor \frac{nx}{\log(1+q/D_i)}\right\rfloor, n\right).$$

Assume $d_k = n$ and $\epsilon \leq 1/(\beta^2 x)^{\log 3}$, and for all i and $d_i \leq j < d_{i+1}$, $|s_j|D_i < 0.23q$. SOLVE runs in time $\mathcal{O}(mn)$ with negligible failure probability.

Proof. Remark that for all i,

$$|\mathcal{L}_{i+1}| \geq (|\mathcal{L}_i| - (1+q/D_i)^{d_{i+1}-d_i})/2 \geq (|\mathcal{L}_i| - 2^{nx})/2.$$

Using induction, we then have $|\mathcal{L}_i| \geq (|\mathcal{L}_0| + 2^{nx})/2^i - 2^{nx}$ so that $|\mathcal{L}_k| \geq n2^{nx}$.

By induction and using the previous lemma, the input of DISTINGUISH is sampled from a LWE distribution with noise parameter:

$$\alpha'^2 = 2^k \alpha^2 + 4\pi^2 \sum_{i=0}^{k-1} 2^{k-i-1} \sum_{j=d_i}^{d_{i+1}-1} (s_j D_i/q)^2.$$

By choice of k the first term is smaller than $nx/6$. As for the second term, since B is non increasing and by choice of D_i, it is smaller than:

$$4\pi^2 \sum_{i=0}^{k-1} 2^{k-i-1} \frac{x/6}{\pi^2 2^{k-i+1}} \sum_{j=d_i}^{d_{i+1}-1} \left(\frac{s_j}{B_j}\right)^2 \leq (x/6) \sum_{j=0}^{n-1} \left(\frac{s_j}{B_j}\right)^2 \leq nx/6.$$

Thus the real part of the bias is superior to $\exp(-nx/3)(1 - 3^a\epsilon) \geq 2^{-nx/2}$, and hence by Theorem 2.2, DISTINGUISH fails with negligible probability. □

Theorem 4. *Assume that for all i, $|s_i| \leq B$, $B \geq 2$, $\max(\beta, \log(q)) = 2^{o(n/\log n)}$, $\beta = \omega(1)$, and $\epsilon \leq 1/\beta^4$. Then* SOLVE *takes time $2^{(n/2+o(n))/\ln(1+\log \beta/\log B))}$.*

Proof. We apply Lemma 7, choosing

$$k = \lfloor \log(\beta^2/(12\ln(1+\log\beta)))\rfloor = (2 - o(1))\log\beta \in \omega(1)$$

and we set $D_i = q/(Bk2^{(k-i)/2})$. It now remains to show that this choice of parameters satisfies the conditions of the lemma.

First, observe that $BD_i/q \leq 1/k = o(1)$ so the condition $|s_j|D_i < 0.23q$ is fulfilled. Then, $d_k \geq n$, which amounts to:

$$\sum_{i=0}^{k-1} \frac{x}{(k-i)/2 + \log \mathcal{O}(kB)} \geq 2x\ln(1 + k/2/\log\mathcal{O}(kB)) \geq 1 + k/n = 1 + o(1)$$

If we have $\log k = \omega(\log\log B)$ (so in particular $k = \omega(\log B)$), we get $\ln(1 + k/2/\log\mathcal{O}(kB)) = (1 + o(1))\ln(k) = (1 + o(1))\ln(1 + \log\beta/\log B)$.

Else, $\log k = \mathcal{O}(\log\log B) = o(\log B)$ (since necessarily $B = \omega(1)$ in this case), so we get $\ln(1 + k/2/\log\mathcal{O}(kB)) = (1 + o(1))\ln(1 + \log\beta/\log B)$.

Thus our choice of x fits both cases and we have $1/x \leq 2\ln(1 + \log\beta)$. Second, we have $1/k = o(\sqrt{x})$ so D_i, ϵ and k are also sufficiently small and the lemma applies. Finally, note that the algorithm has complexity $2^{\Omega(n/\log n)}$, so a factor $n2^k \log(q)$ is negligible. □

This theorem can be improved when the use of the given parameters yields $D < 1$, since $D = 1$ already gives a lossless quantization.

Theorem 5. *Assume that for all i, $|s_i| \leq B = n^{b+o(1)}$. Let $\beta = n^c$ and $q = n^d$ with $d \geq b$ and $c + b \geq d$. Assume $\epsilon \leq 1/\beta^4$. Then* SOLVE *takes time $2^{n/(2(c-d+b)/d+2\ln(d/b)-o(1))}$.*

Proof. Once again we aim to apply Lemma 7, and choose k as above:

$$k = \log(\beta^2/(12\ln(1+\log\beta))) = (2c - o(1))\log n$$

If $i < \lceil 2(c-d+b)\log n\rceil$, we take $D_i = 1$, else we choose $q/D_i = \Theta(B2^{(a-i)/2})$. Satisfying $d_a \geq n-1$ amounts to:

$$2x(c-d+b)\log n/\log q + \sum_{i=\lceil 2(c-d+b)\log n\rceil}^{a-1} \frac{x}{(a-i)/2 + \log\mathcal{O}(B)}$$
$$\geq 2x(c-d+b)/d + 2x\ln((a - 2(c-d+b)\log n + 2\log B)/2/\log\mathcal{O}(B))$$
$$\geq 1 + a/n = 1 + o(1)$$

So that we can choose $1/x = 2(c-d+b)/d + 2\ln(d/b) - o(1)$. □

Corollary 1. *Given a* LWE *problem with* $q = n^d$, *Gaussian errors with* $\beta = n^c$, $c > 1/2$ *and* $\epsilon \leq n^{-4c}$, *we can find a solution in* $2^{n/(1/d + 2\ln(d/(1/2 + d - c))) - o(1))}$ *time.*

Proof. Apply Theorem 1 : with probability $2/3$, the secret is now bounded by $B = \mathcal{O}(q\sqrt{n}/\beta\sqrt{\log n})$. The previous theorem gives the complexity of an algorithm discovering the secret, using $b = 1/2 - c + d$, and which works with probability $2/3 - 2^{-\Omega(n)}$. Repeating n times with different samples, the correct secret will be outputted at least $n/2 + 1$ times, except with negligible probability. By returning the most frequent secret, the probability of failure is negligible. □

In particular, if $c \leq d$, it is possible to quantumly approximate lattice problems within factor $\mathcal{O}(n^{c+1/2})$ [46]. Setting $c = d$, the complexity is $2^{n/(1/c + 2\ln(2c) - o(1))}$, so that the constant slowly converges to 0 when c goes to infinity.

A simple BKW using the bias would have a complexity of $2^{d/cn + o(n)}$, the analysis of [5] or [4] only conjectures $2^{dn/(c-1/2) + o(n)}$ for $c > 1/2$. In [5], the authors incorrectly claim a complexity of $2^{cn + o(n)}$ when $c = d$, because the blowup in the error is not explicitely computed [5].

Though the upper bound on ϵ is small, provably solving the learning with rounding problem where $b = \frac{q}{p}\lfloor \frac{p}{q}\langle \boldsymbol{a}, \boldsymbol{s}\rangle \rceil$ for some p seems out of reach[6].

Finally, if we want to solve the LWE problem for different secrets but with the same vectorial part of the samples, it is possible to be much faster if we work with a bigger final bias, since the REDUCE part needs to be called only once.

3.4 Some Practical Improvements

In this section we propose a few heuristic improvements for our main algorithm, which speed it up in practice, although they do not change the factor in the exponent of the overall complexity. In our main algorithm, after a few iterations of REDUCE, each sample is a sum of a few original samples, and these sums are disjoint. It follows that samples are independent. This may no longer be true below, and sample independence is lost, hence the heuristic aspect. However this has negligible impact in practice.

First, in the non-binary case, when quantizing a sample to associate it to a center, we can freely quantize its opposite as well (i.e. quantize the sample made up of opposite values on each coordinate) as in [4].

Second, at each REDUCE step, instead of substracting any two samples that are quantized to the same center, we could choose samples whose difference is as small as possible, among the list of samples that are quantized to the same center. The simplest way to achieve this is to generate the list of all differences, and pick the smallest elements (using the L2 norm). We can thus form a new sample list, and we are free to make it as long as the original. Thus we get smaller samples overall with no additional data requirement, at the cost of losing sample independence.[7]

Analyzing the gain obtained using this tactic is somewhat tricky, but quite important for practical optimization. One approach is to model reduced sample coordinates as independent and following the same Gaussian distribution. When adding vectors, the coordinates of the sum then also follows a Gaussian distribution. The (squared) norm of a vector with Gaussian coordinates follows the χ^2 distribution. Its cumulative distribution for a k-dimensional vector is the regularized gamma function, which amounts to $1 - \exp(-x/2)\sum_{i=0}^{k/2}(x/2)^i/i!$ for even k (assuming standard deviation 1 for sample coordinates).

Now suppose we want to keep a fixed proportion of all possible sums. Then using the previous formula we can compute the radius R such that the expected number of sample sums falling within the ball of radius R is equal to the desired proportion. Thus using this Gaussian distribution model, we are able to predict how much our selection technique is able to decrease the norm of the samples.

---

[5] They claim it is possible to have $\approx \log n$ reduction steps while the optimal number is $\approx 2c\log(n)$ so that is loose for $c > 1/2$ and wrong for $c < 1/2$.

[6] [19] claims to do so, but it actually assumes the indepedency of the errors from the vectorial part of the samples.

[7] A similar approach was taken in [5] where the L1 norm was used, and where each new sample is reduced with the shortest element ever quantized on the same center.

Practical experiments show that for a reasonable choice of parameters (namely keeping a proportion 1/1000 of samples for instance), the standard deviation of sample norms is about twice as high as predicted by the Gaussian model for the first few iterations of REDUCE, then falls down to around 15% larger. This is due to a variety of factors; it is clear that for the first few iterations, sample coordinates do not quite follow a Gaussian distribution. Another notable observation is that newly reduced coordinates are typically much larger than others.

While the previous Gaussian model is not completely accurate, the ability to predict the expected norms for the optimized algorithm is quite useful to optimize parameters. In fact we could recursively compute using dynamic programming what minimal norm is attainable for a given dimension n and iteration count a within some fixed data and time complexities. In the end we will gain a constant factor in the final bias.

In the binary case, we can proceed along a similar line by assuming that coordinates are independent and follow a Bernoulli distribution.

Regarding secret recovery, in practice, it is worthwhile to compute the Fourier transform on a high dimension, such that its cost approximately matches that of SOLVE. On the other hand, for a low enough dimension, computing the experimental bias for each possible high probability secret may be faster.

Another significant improvement in practice can be to apply a linear quantization step just before secret recovery. The quantization steps we have considered are linear, in the sense that centers are a lattice. If $\mathbf{A}$ is a basis of this lattice, in the end we are replacing a sample $(\mathbf{a}, b)$ by $(\mathbf{Ax}, b)$. We get $\langle \mathbf{x}, \mathbf{A}^t \mathbf{s} \rangle + \mathbf{e} = \langle \mathbf{Ax}, \mathbf{s} \rangle + \mathbf{e} = b - \langle (\mathbf{a} - \mathbf{Ax}), \mathbf{s} \rangle$. Thus the dimension of the Fourier transform is decreased as remarked by [25], at the cost of a lower bias. Besides, we no longer recover $\mathbf{s}$ but $\mathbf{y} = \mathbf{A}^t \mathbf{s}$. Of course we are free to change $\mathbf{A}$ and quantize anew to recover more information on $\mathbf{s}$. In some cases, if the secret is small, it may be worth simply looking for the small solutions of $\mathbf{A}^t \mathbf{x} = \mathbf{y}$ (which may in general cost an exponential time) and test them against available samples.

In general, the fact that the secret is small can help its recovery through a maximum likelihood test [43]. If the secret has been quantized as above however, $\mathbf{A}$ will need to be chosen such that $\mathbf{As}$ is small (by being sparse or having small entries). The probability to get a given secret can then be evaluated by a Monte-Carlo approach with reasonable accuracy within negligible time compared to the Fourier transform.

If the secret has a coordinate with non-zero mean, it should be translated.

Finally, in the binary case, it can be worthwhile to combine both of the previous algorithms: after having reduced vectors, we can assume some coordinates of the secret are zero, and quantize the remainder. Depending on the number of available samples, a large number of secret-error switches may be possible. Under the assumption that the success probability is independent for each try, this could be another way to essentially proceed as if we had a larger amount of data than is actually available. We could thus hope to operate on a very limited amount of data (possibly linear in n) for a constant success probability.

The optimizations above are not believed to change the asymptotic complexity of the algorithm, but have a significant impact in practice.

3.5 Experimentation

We have implemented our algorithm, in order to test its efficiency in practice, as well as that of the practical improvements in subsection 3.4. We have chosen dimension $n = 128$, modulus $q = n^2$, binary secret, and Gaussian errors with noise parameter $\alpha = 1/(\sqrt{n/\pi} \log^2 n)$. The previous best result for these parameters, using a BKW algorithm with lazy modulus switching, claims a time complexity of 2^{74} with 2^{60} samples [5].

Using our improved algorithm, we were able to recover the secret using $m = 2^{28}$ samples within 13 hours on a single PC equipped with a 16-core Intel Xeon. The computation time proved to be devoted mostly to the computation of $9 \cdot 10^{13}$ norms, computed in fixed point over 16 bits in SIMD. The implementation used a naive quantizer.

In section B, we compare the different techniques to solve the LWE problem when the number of samples is large or small. We were able to solve the same problem using BKZ with block size 40 followed by an enumeration in two minutes.

Table 1. Complexity of solving LWE with the Regev parameters [46], i.e. the error distribution is a continuous gaussian of standard deviation $\frac{q}{\sqrt{2\pi n}\log^2(n)}$ and with modulus $q \approx n^2$. The reasonable column corresponds to multiplying the predicted reduction factor at each step by 1.1, and assuming that the quantizers used reduce the variance by a factor of 1.3. The corresponding parameters of the algorithm are shown in column k (the number of reduction steps), $\log(m)$ (m is the list size), and $\log(N)$ (one vector is kept for the next iteration for each N vectors tested). The complexities are expressed as logarithm of the number of bit operations of each reduction step. Pessimistic uses a multiplier of 2 and a naive quantifier (factor 1). Optimistic uses a multiplier of 1 and an asymptotical quantifier (factor $2\pi e/12 \approx 1.42$). The asymptotical complexity is $2^{0.93n+o(n)}$ instead of $2^{n+o(n)}$.

n	q	k	$\log(m)$	$\log(N)$	Reasonable	Optimistic	Pessimistic	Previous [19]
64	4099	16	30	0	39.6	39.6	40.6	56.2
80	6421	17	38	0	47.9	46.0	48.0	66.9
96	9221	18	45	0	55.3	54.3	56.3	77.4
112	12547	18	54	0	64.6	60.6	65.6	89.6
128	16411	19	60	0	70.8	67.8	72.8	98.8
160	25601	20	75	0	86.2	82.2	88.2	119.7
224	50177	21	93	13	117.8	111.8	121.8	164.3
256	65537	22	106	15	133.0	125.0	137.0	182.7
384	147457	24	164	18	194.7	183.7	201.7	273.3
512	262147	25	219	25	257.2	242.2	266.2	361.6

Table 2. Complexity of solving LWE with the Lindner-Peikert parameters [34]. The error distribution is $D_{\mathbb{Z},s}$. The security of the second parameter set was said to be better than AES-128. We assume that we have access to as many samples as we want, while it is not the case in the proposed cryptosystem.

n	q	s	k	$\log(m)$	$\log(N)$	Reasonable	Optimistic	Pessimistic
192	4099	8.87	19	68	5	84.2	79.2	84.2
256	6421	8.35	20	82	8	101.7	95.7	103.7
320	9221	8.00	22	98	9	119.0	112.0	122.0

Table 3. Complexity of solving LWE with binary ($\{0,1\}$) secret with the Regev parameters [46].

n	q	k	$\log(m)$	$\log(N)$	Reasonable	Optimistic	Pessimistic	Previous [5]
128	16411	16	28	0	38.8	38.8	39.8	74.2
256	65537	19	52	0	64.0	62.0	67.0	132.5
512	262147	22	99	0	112.2	104.2	117.2	241.8

3.6 Extension to the norm L2

One can think that in Lemma 6, the condition $|s_j|D_i < 0.23q$ is artificial. In this section, we show how to remove it, so that the only condition on the secret is $||s|| \leq \sqrt{n}B$ which is always better. The basic idea is to use rejection sampling to transform the coordinates of the vectorial part of the samples into small discrete gaussians. The following randomized algorithm works only for some moduli, and is expected to take more time, though the asymptotical complexity is the same.

1: **function** ACCEPT($\boldsymbol{a}$,$\boldsymbol{u}$,σ,k)   ▷ For all i, $\boldsymbol{a}_i \in [0; k[$
2:     **return true** with probability $\exp(-\pi(\sum_i \min(\boldsymbol{u}_i^2, (\boldsymbol{u}_i + 1)^2) - (\boldsymbol{u_i} + \boldsymbol{a}_i/k)^2)/\sigma^2)$
3: **end function**
4: **function** REDUCEL2($\mathcal{L}_{in}$,D_i,d_i,d_{i+1},σ_i)
5:     $t[] \leftarrow \varnothing$
6:     **for all** $(\mathbf{a}, b) \in \mathcal{L}_{in}$ **do**
7:         $\mathbf{r} = \lfloor \frac{(\boldsymbol{a}_{d_i}, ..., \boldsymbol{a}_{d_{i+1}-1})}{D} \rceil$
8:         PUSH($t[\mathbf{r}], (\mathbf{a}, b)$)
9:     **end for**
10:    $\mathcal{L}_{out} \leftarrow \varnothing$
11:    **while** $|\{\boldsymbol{r} \in (\mathbb{Z}/(q/D_i)\mathbb{Z})^{d_{i+1}-d_i}; t[r] \neq \varnothing\}| \geq (q/D_i)^{d_{i+1}-d_i}/3$ **do**
12:        Sample $\boldsymbol{x}$ and $\boldsymbol{y}$ according to $D_{\mathbb{Z}^{d_{i+1}-d_i}, \sigma_i}$
13:        **repeat**
14:            Sample $\boldsymbol{u}$ and $\boldsymbol{v}$ uniformly in $(\mathbb{Z}/(q/D_i)\mathbb{Z})^{d_{i+1}-d_i}$
15:        **until** $t[\boldsymbol{u} + \boldsymbol{x}] \neq \varnothing$ and $t[\boldsymbol{v} + \boldsymbol{y}] \neq \varnothing$
16:        $(\boldsymbol{a_0}, b_0) \leftarrow$ POP($t[\boldsymbol{u} + \boldsymbol{x}]$)
17:        $(\boldsymbol{a_1}, b_1) \leftarrow$ POP($t[\boldsymbol{v} + \boldsymbol{y}]$)
18:        **if** ACCEPT($\boldsymbol{a_0}$ mod $q/D_i, \boldsymbol{u}, \sigma_i, q/D_i$) and ACCEPT($\boldsymbol{a_1}$ mod $q/D_i, \boldsymbol{v}, \sigma_i, q/D_i$) **then**
19:            $\mathcal{L}_{out} \leftarrow (\boldsymbol{a_0} - \boldsymbol{a_1}, b_0 - b_1) :: \mathcal{L}_{out}$
20:        **end if**
21:    **end while**
22:    **return** $\mathcal{L}_{out}$
23: **end function**

Lemma 8. *Assume that $D_i|q$, σ_i is larger than some constant and*

$$|\mathcal{L}_i| \geq 2n \max(n \log(q/D_i)(q/D_i)^{d_{i+1}-d_i}, \exp(5(d_{i+1} - d_i)/\sigma_i)).$$

Write $\mathcal{L}'_i$ for the samples of $\mathcal{L}_i$ where the first d_i coordinates of each sample vector have been truncated. If $\mathcal{L}'_i$ is sampled according to the LWE *distribution with secret* **s** *and noise parameters α and ϵ, then $\mathcal{L}'_{i+1}$ is sampled according to the* LWE *distribution of the truncated secret with parameters:*

$$\alpha'^2 = 2\alpha^2 + 2\pi \sum_{j=d_i}^{d_{i+1}-1} (s_j \sigma_i D_i/q)^2 \quad \text{and} \quad \epsilon' = 3\epsilon.$$

On the other hand, if $D_i = 1$, then $\alpha'^2 = 2\alpha^2$. Furthermore, REDUCEL2 *runs in time $\mathcal{O}(n \log q|\mathcal{L}_i|)$ and $|\mathcal{L}_{i+1}| \geq |\mathcal{L}_i| \exp(-5(d_{i+1} - d_i)/\sigma_i)/6$ except with probability $2^{-\Omega(n)}$.*

Proof. On lines 16 and 17, $\boldsymbol{a_0}$ mod q/D_i and $\boldsymbol{a_1}$ mod q/D_i are uniform and independent. On line 19, because of the rejection sampling theorem, we have $(q/D_i)\boldsymbol{u} + (\boldsymbol{a_0}$ mod $q/D_i)$ and $(q/D_i)\boldsymbol{v} + (\boldsymbol{a_1}$ mod $q/D_i)$ sampled according to $D_{\mathbb{Z}^{d_{i+1}-d_i}, \sigma_i q/D_i}$. Also, the unconditional acceptance probability is, for sufficiently large σ_i,

$$(\rho_{\sigma_i q/D_i}(\mathbb{Z})/(q/D_i(1 + \rho_{\sigma_i}(\mathbb{Z}))))^{2(d_{i+1}-d_i)} \geq$$
$$(\sigma_i q/D_i/(q/D_i(2 + \sigma_i)))^{2(d_{i+1}-d_i)} \geq$$
$$\exp(-5(d_{i+1} - d_i)/\sigma_i)$$

where the first inequality comes from Poisson summation on both ρ. Next, the bias added by truncating the samples is the bias of the scalar product of $(q/D_i)\boldsymbol{u} + (\boldsymbol{a_0} \bmod q/D_i) - (q/D_i)\boldsymbol{v} - (\boldsymbol{a_1} \bmod q/D_i)$ with the corresponding secret coordinates. Therefore, using a Poisson summation, we can prove that α' is correct.

The loop on line 15 is terminated with probability at least $2/3$ each time so that the Hoeffding inequality proves the complexity.

On line 10, the Hoeffding inequality proves that $\min_{\boldsymbol{r}} t[\boldsymbol{r}] \geq |\mathcal{L}_i|/(2(q/D_i)^{d_{i+1}-d_i})$ holds except with probability $2^{-\Omega(n)}$. Under this assumption, the loop on lines 11-21 is executed at least $|\mathcal{L}_i|/6$ times. Then, the Hoeffding inequality combined with the bound on the unconditional acceptance probability proves the lower bound on $|\mathcal{L}_{i+1}|$. □

Theorem 6. *Assume that* $||s|| \leq \sqrt{n}B$, $B \geq 2$, $\max(\beta, \log(q)) = 2^{o(n/\log n)}$, $\beta = \omega(1)$, *and* $\epsilon \leq 1/\beta^4$. *Then, there exists an integer Q smaller than $B^n 2^{2n^2}$ such that if q is divisible by this integer, we can solve* Decision-LWE *time*

$$2^{(n/2+o(n))/\ln(1+\log\beta/\log B)}.$$

Proof. We use $\sigma_i = \sqrt{\log \beta}$, $m = 2n^2 6^k \exp(5n/\sigma_i)2^{nx}$,

$$k = \lfloor \log(\beta^2/(12\ln(1+\log\beta))) \rfloor = (2 - o(1))\log\beta \in \omega(1)$$

and we set $D_i = q/(Bk2^{(k-i)/2})$, $d_{i+1} = \min(d_i + \lfloor \frac{nx}{\log(q/D_i)} \rfloor, n)$ and $Q = \prod_i Bk2^{(k-i)/2}$. We can then see as in Theorem 4 that these choices lead to some $x = 1/(2 - o(1))/\ln(1 + \log\beta/\log B)$. Finally, note that the algorithm has complexity $2^{\Omega(n/\log\log\beta)}$, so a factor $2n^2 6^k \log(q) \exp(5n/\sqrt{\log(\beta)})$ is negligible. □

The condition on q can be removed using modulus switching [13], unless q is tiny.

Theorem 7. *If $B\beta \leq q$, then the previous theorem holds without the divisibility condition.*

Proof. Let $p \geq q$ be the smallest modulus such that the previous theorem applies and $\varsigma = \sqrt{n}p/q$. For each sample $(\boldsymbol{a}, b)$, sample $\boldsymbol{x}$ from $D_{\mathbb{Z}^n - \boldsymbol{a}/q, \varsigma}$ and use the sample $(p/q\boldsymbol{a} + p\boldsymbol{x}, p/qb)$ with the algorithm of the previous theorem. Clearly, the vectorial part has independent coordinates, and the probability that one coordinate is equal to y is proportional to $\rho_\varsigma(p/q\mathbb{Z} - y)$. Therefore, the Kullback-Leibler distance with the uniform distribution is $2^{-\Omega(n)}$. If the original error distribution has a noise parameter α, then the new error distribution has noise parameter α' with

$$\alpha'^2 \leq \alpha^2 + \sum_i \pi\varsigma^2 s_i^2/p^2 \leq \alpha^2 + \pi B^2 \varsigma^2/p^2 \leq n/2/\beta^2 + \pi n B^2/q^2$$

so that β is only reduced by a constant. □

4 Applications to Lattice Problems

We first show that $\mathsf{BDD}_{B,\beta}$ is easier than LWE for some large enough modulus and then that $\mathsf{UniqueSVP}_{B,\beta}$ and $\mathsf{GapSVP}_{B,\beta}$ are easier than $\mathsf{BDD}_{B,\beta}$.

4.1 Variant of Bounding Distance Decoding

The main result of this subsection is close to the classic reduction of [46]. However, our definition of LWE allows to simplify the proof, and gain a constant factor in the decoding radius. The use of the KL divergence instead of the statistical distance also allows to gain a constant factor, when we need an exponential number of samples, or when λ_n^* is really small.

The core of the reduction lies in Lemma 9, assuming access to a Gaussian sampling oracle. This hypothesis will be taken care of in Lemma 10.

Lemma 9. *Let* $\mathbf{A}$ *be a basis of the lattice* Λ *of full rank* n. *Assume we are given access to an oracle outputting a vector sampled under the law* $D_{\Lambda^*,\sigma}$ *and* $\sigma \geq q\eta_\epsilon(\Lambda^*)$, *and to an oracle solving the* LWE *problem in dimension* n, *modulus* $q \geq 2$, *noise parameter* α, *and distortion parameter* ξ *which fails with negligible probability and use* m *vectors if the secret* $\mathbf{s}$ *verifies* $|s_i| \leq B_i$.

Then, if we are given a point $\mathbf{x}$ *such that there exists* $\mathbf{s}$ *with* $\mathbf{v} = \mathbf{As} - \mathbf{x}$, $||\mathbf{v}|| \leq \sqrt{1/\pi}\alpha q/\sigma$, $|s_i| \leq B_i$ *and* $\rho_{\sigma/q}(\Lambda \setminus \{\mathbf{0}\} + \mathbf{v}) \leq \xi\exp(-\alpha^2)/2$, *we are able to find* $\mathbf{s}$ *in at most* mn *calls to the Gaussian sampling oracle,* n *calls to the* LWE *solving oracle, with a probability of failure* $n\sqrt{m}\epsilon + 2^{-\Omega(n)}$ *and complexity* $\mathcal{O}(mn^3 + n^c)$ *for some* c.

Proof. Let $\mathbf{y}$ be sampled according to $D_{\Lambda^*,\sigma}$, $\mathbf{v} = \mathbf{As} - \mathbf{x}$. Then

$$\langle \mathbf{y}, \mathbf{x} \rangle = \langle \mathbf{y}, \mathbf{As} \rangle + \langle \mathbf{y}, \mathbf{v} \rangle = \langle \mathbf{A}^t\mathbf{y}, \mathbf{s} \rangle - \langle \mathbf{y}, \mathbf{v} \rangle$$

and since $\mathbf{y} \in \Lambda^*$, $\mathbf{A}^t\mathbf{y} \in \mathbb{Z}^n$. We can thus provide the sample $(\mathbf{A}^t\mathbf{y}, \langle \mathbf{y}, \mathbf{x} \rangle)$ to the LWE solving oracle.

The probability of obtaining the sample $\mathbf{A}^t\mathbf{y} \in (\mathbb{Z}/q\mathbb{Z})^n$ is proportional to $\rho_\sigma(q\Lambda^* + \mathbf{y})$. Using the Poisson summation formula,

$$\rho_\sigma(q\Lambda^* + \mathbf{y}) = s^n \det(\Lambda/q) \sum_{\mathbf{z} \in \Lambda/q} \exp(2i\pi\langle \mathbf{z}, \mathbf{y} \rangle)\rho_{1/\sigma}(\mathbf{z}).$$

Since $\sigma \geq q\eta_\epsilon(\Lambda^*) = \eta_\epsilon(q\Lambda^*)$, we have

$$1 - \epsilon \leq \sum_{\mathbf{z} \in \Lambda/q} \cos(2\pi\langle \mathbf{z}, \mathbf{y} \rangle)\rho_{1/\sigma}(\mathbf{z}) \leq 1 + \epsilon.$$

Therefore, with Lemma 2, the KL-divergence between the distribution of $\mathbf{A}^t\mathbf{y} \bmod q$ and the uniform distribution is less than $2\epsilon^2$.

Also, with $\mathbf{c} \in \mathbf{y} + q\Lambda^*$,

$$\mathbb{E}[\exp(2i\pi(\langle \mathbf{A}^t\mathbf{y}, \mathbf{s} \rangle - \langle \mathbf{y}, \mathbf{x} \rangle)/q)|\mathbf{A}^t\mathbf{y} \bmod q] = \mathbb{E}_{D_{q\Lambda^*+\mathbf{c},\sigma}}[\exp(2i\pi\langle \mathbf{y}, \mathbf{v}/q \rangle)].$$

Let $f(\mathbf{y}) = \exp(2i\pi\langle \mathbf{y}, \mathbf{v}/q \rangle)\rho_\sigma(\mathbf{y})$ so that the bias is $f(q\Lambda^* + \mathbf{c})/\rho_\sigma(q\Lambda^* + \mathbf{c})$. Using the Poisson summation formula on both terms, this is equal to

$$\left(\sum_{\mathbf{y} \in \Lambda/q} \rho_{1/\sigma}(\mathbf{y} - \mathbf{v}/q) \exp(-2i\pi\langle \mathbf{y}, \mathbf{c} \rangle) \right) \Big/ \left(\sum_{\mathbf{y} \in \Lambda/q} \rho_{1/\sigma}(\mathbf{y}) \cos(2\pi\langle \mathbf{y}, \mathbf{c} \rangle) \right).$$

In this fraction, the numerator is at distance at most $\xi\exp(-\alpha^2)/(1 + \epsilon)$ to $\exp(-\pi\sigma^2/q^2||\mathbf{v}||^2) \geq \exp(-\alpha^2)$ and the denominator is in $[1 - \epsilon; 1 + \epsilon]$.

Using Lemma 3 with an algorithm which tests if the returned secret is equal to the real one, the failure probability is bounded by $\sqrt{m}\epsilon$. Thus, the LWE solving oracle works and gives $\mathbf{s} \bmod q$.

Let $\mathbf{x}' = (\mathbf{x} - \mathbf{A}(\mathbf{s} \bmod q))/q$ so that $\mathbf{s}' = (\mathbf{s} - (\mathbf{s} \bmod q))/q$ and $||\mathbf{v}'|| \leq ||\mathbf{v}||/q$. Therefore, the reduction also works. If we repeat this process n times, we can solve the last closest vector problem with Babai's algorithm, which reveals $\mathbf{s}$. $\qquad\square$

In the previous lemma, we required access to a $D_{\Lambda^*,\sigma}$ oracle. However, for large enough σ, this hypothesis comes for free, as shown by the following lemma, which we borrow from [13].

Lemma 10. *If we have a basis* $\mathbf{A}$ *of the lattice* Λ, *then for* $\sigma \geq \mathcal{O}(\sqrt{\log n}||\widetilde{\mathbf{A}}||)$, *it is possible to sample in polynomial time from* $D_{\Lambda,\sigma}$.

We will also need the following lemma, due to Banaszczyk [10].

Lemma 11. *For a lattice* Λ, $\mathbf{c} \in \mathbb{R}^n$, *and* $t \geq 1$,

$$\frac{\rho\big((\Lambda + \mathbf{c}) \setminus \mathcal{B}\big(0, t\sqrt{\frac{n}{2\pi}}\big)\big)}{\rho(\Lambda)} \leq \exp\big(-n(t^2 - 2\ln t - 1)/2\big) \leq \exp\big(-n(t-1)^2/2\big).$$

Proof. For any $s \geq 1$, using Poisson summation :

$$\frac{\rho_s(\Lambda + \mathbf{c})}{\rho(\Lambda)} = s^n \frac{\sum_{\mathbf{x} \in \Lambda^*} \rho_{1/s}(\mathbf{x}) \exp(2i\pi\langle\mathbf{x},\mathbf{c}\rangle)}{\rho(\Lambda^*)}$$

$$\leq s^n \frac{\sum_{\mathbf{x} \in \Lambda^*} \rho_{1/s}(\mathbf{x})}{\sum_{\mathbf{x} \in \Lambda^*} \rho(\mathbf{x})}$$

$$\leq s^n$$

Then,

$$s^n \rho(\Lambda) \geq \rho_s\left((\Lambda + \mathbf{c}) \setminus \mathcal{B}\left(\mathbf{0}, t\sqrt{\frac{n}{2\pi}}\right)\right)$$

$$\geq \exp(t^2(1 - 1/s^2)n/2)\rho\left((\Lambda + \mathbf{c}) \setminus \mathcal{B}\left(\mathbf{0}, t\sqrt{\frac{n}{2\pi}}\right)\right).$$

And therefore, using $s = t$:

$$\frac{\rho((\Lambda + \mathbf{c}) \setminus \mathcal{B}(\mathbf{0}, t\sqrt{\frac{n}{2\pi}}))}{\rho(\Lambda)} \leq \exp(-n(t^2(1 - 1/s^2) - 2\ln s)/2)$$

$$= \exp(-n(t^2 - 2\ln t - 1)/2)$$

$$\leq \exp(-n(t - 1)^2/2),$$

where the last inequality stems from $\ln t \leq t - 1$. $\qquad \square$

Theorem 8. *Assume we have a* LWE *solving oracle of modulus* $q \geq 2^n$, *parameters* β *and* ξ *which needs* m *samples.*

If we have a basis $\mathbf{A}$ *of the lattice* Λ, *and a point* $\mathbf{x}$ *such that* $\mathbf{As} - \mathbf{x} = \mathbf{v}$ *with* $||\mathbf{v}|| \leq (1 - 1/n)\lambda_1/\beta/t < \lambda_1/2$ *and* $4\exp(-n(t - 1/\beta - 1)^2/2) \leq \xi\exp(-n/2/\beta^2)$, *then with* n^2 *calls to the* LWE *solving oracle with secret* $\mathbf{s}$, *we can find* $\mathbf{s}$ *with probability of failure* $2\sqrt{m}\exp(-n(t^2 - 2\ln t - 1)/2)$ *for any* $t \geq 1 + 1/\beta$.

Proof. Using Lemma 11, we can prove that $\sigma = t\sqrt{n/2/\pi}/\lambda_1 \leq \eta_\epsilon(\Lambda^*)$ for $\epsilon = 2\exp(-n(t^2 - 2\ln t - 1)/2)$ and

$$\rho_{1/\sigma}(\Lambda \setminus \{\mathbf{0}\} + \mathbf{v}) \leq 2\exp\left(-n(t(1 - 1/\beta/t) - 1)^2/2\right).$$

Using LLL, we can find a basis $\mathbf{B}$ of Λ so that $||\widetilde{\mathbf{B}^*}|| \leq 2^{n/2}/\lambda_1$, and therefore, it is possible to sample in polynomial time from $D_{\Lambda, q\sigma}$ since $q \geq 2^n$ for sufficiently large n.

The LLL algorithm also gives a non zero lattice vector of norm $\ell \leq 2^n \lambda_1$. For i from 0 to n^2, we let $\lambda = \ell(1 - 1/n)^i$, we use the algorithm of Lemma 9 with standard deviation $tq\sqrt{n/2/\pi}/\lambda$, which uses only one call to the LWE solving oracle, and return the closest lattice vector of $\mathbf{x}$ in all calls.

Since $\ell(1 - 1/n)^{n^2} \leq 2^n\exp(-n)\lambda_1 \leq \lambda_1$, with $0 \leq i \leq n^2$ be the smallest integer such that $\lambda = \ell(1 - 1/n)^i \leq \lambda_1$, we have $\lambda \geq (1 - 1/n)\lambda_1$. Then the lemma applies since

$$||\mathbf{v}|| \leq (1 - 1/n)\lambda_1/\beta/t \leq \sqrt{1/\pi}\sqrt{n/2}/\beta q/(tq\sqrt{n/2/\pi}/\lambda) = \lambda/t/\beta.$$

Finally, the distance bound makes $\mathbf{As}$ the unique closest lattice point of $\mathbf{x}$. $\qquad \square$

Using self-reduction, it is possible to remove the $1 - 1/n$ factor [36].

Corollary 2. *It is possible to solve* $\mathsf{BDD}_{B,\beta}^{||.||_\infty}$ *in time* $2^{(n/2+o(n))/\ln(1+\log\beta/\log B)}$ *if* $\beta = \omega(1)$, $\beta = 2^{o(n/\log n)}$ *and* $\log B = \mathcal{O}(\log\beta)$.

Proof. Apply the previous theorem and Theorem 4 with some sufficiently large constant for t, and remark that dividing β by some constant does not change the complexity. $\qquad \square$

Note that since we can solve LWE for many secrets in essentially the same time than for one, we have the same property for BDD.

4.2   UniqueSVP

The following reduction was given in [36] and works without modification.

Theorem 9. *Given a* $\mathsf{BDD}_{B,\beta}^{||\cdot||_\infty}$ *oracle, it is possible to solve* $\mathsf{UniqueSVP}_{B,\beta}^{||\cdot||_\infty}$ *in polynomial time of* n *and* β.

Proof. Let $2\beta \geq p \geq \beta$ be a prime number, $\mathbf{A}$ and $\mathbf{s}$ such that $||\mathbf{As}|| = \lambda_1$. If $\mathbf{s} = \mathbf{0} \mod p$ then $\mathbf{A}(\mathbf{s}/p)$ is a non zero lattice vector, shorter than $||\mathbf{As}||$, which is impossible. Let i such that $s_i \neq 0 \mod p$, and

$$\mathbf{A'}_i = [\mathbf{a}_0, \ldots, \mathbf{a}_{i-1}, p\mathbf{a}_i, \mathbf{a}_{i+1}, \ldots, \mathbf{a}_{n-1}],$$

which generates a sublattice Λ' of Λ. If $k \neq 0 \mod p$, then $k\mathbf{As} \notin \Lambda'$ so that $\lambda_1(\Lambda') \geq \lambda_2(\Lambda)$.

If $\mathbf{s'}_i = (s_0, \ldots, \lfloor s_i/p \rfloor, \ldots, s_{n-1})$, then $\mathbf{A'}_i \mathbf{s'}_i + (s_i \mod p)\mathbf{a}_i = \mathbf{As}$ and $||\mathbf{s'}_i||_\infty \leq B$. Therefore, calling BDD to find the point of Λ' closest to $(s_i \mod p)\mathbf{a}_i$ yields $\mathbf{s'}$. By trying every value of i and $(s_i \mod p)$, np calls to $\mathsf{BDD}_{B,\beta}^{||\cdot||_\infty}$ are enough to solve $\mathsf{UniqueSVP}_{B,\beta}^{||\cdot||_\infty}$. $\qquad\square$

The reductions for both $\mathsf{BDD}^{||\cdot||}$ and $\mathsf{UniqueSVP}^{||\cdot||}$ work the same way.

4.3   GapSVP

The following reduction is a modification of the reduction given in [44], which comes from [23], but doesn't fit to our context.

First, we need a lemma proven in [23] :

Lemma 12. *The volume of the intersection of two balls of radius* 1 *divided by the volume of one ball is at least*

$$\frac{d}{3}(1 - d^2)^{(n-1)/2}\sqrt{n} \geq \frac{d}{3}(1 - d^2)^{n/2}$$

where $d \leq 1$ *is the distance between the centers.*

Lemma 13. *Given access to an oracle solving* $\mathsf{BDD}_{B,\beta}^{||\cdot||_\infty}$ *and let* $\mathcal{D}$ *be an efficiently samplable distribution over points of* $\mathbb{Z}^n$ *whose infinity norm is smaller than* b *and for* $\xi \leq 1$ *and let*

$$\epsilon = \min_{||\mathbf{s}||_\infty \leq R} \Pr_{\mathbf{x} \sim \mathcal{D}} \left[\xi \leq \frac{\mathcal{D}(\mathbf{x} + \mathbf{s})}{\mathcal{D}(\mathbf{x})} \leq 1/\xi \right].$$

Then, we can solve $\mathsf{GapSVP}_{R,\beta/d}^{||\cdot||_\infty}$ *with negligible probability of failure using* $\mathcal{O}(dn/\epsilon/\xi(1 - d^2)^{-n/2})$ *calls to the* BDD *oracle.*

Proof. Let $K = \Theta(dn/\epsilon/\xi(1 - d^2)^{-n/2})$, $\mathbf{A}$ a basis of the lattice Λ. The algorithm consists in testing K times the oracle : sample $\mathbf{x}$ with law $\mathcal{D}$ and check if $\mathsf{BDD}(\mathbf{Ax} + \mathbf{e}) = \mathbf{x}$ for $\mathbf{e}$ sampled uniformly within a ball centered at the origin and of radius $1/d$. If any check is wrong, return that $\lambda_1 \leq 1$, else return $\lambda_1 \geq \beta/d$.

Clearly, the algorithm runs in the given complexity, and if $\lambda_1 \geq \beta/d$, it is correct.

So, let $\mathbf{s} \neq \mathbf{0}$ such that $||\mathbf{As}|| \leq 1$. Let $\mathbf{x}$ and $\mathbf{e}$ be sampled as in the algorithm. With probability over $\mathbf{e}$ greater than $(1 - d^2)^{n/2}d/3$, $||\mathbf{Ax} + \mathbf{e} - \mathbf{A}(\mathbf{x} + \mathbf{s})|| \leq 1/d$. We condition on this event.

Then, with probability at least ϵ, $1/\xi \geq k = \frac{D(\mathbf{x}+\mathbf{s})}{D(\mathbf{x})} \geq \xi$ and we also condition on this event. Let p be the probability that $\mathsf{BDD}(\mathbf{Ax} + \mathbf{e}) = \mathbf{x}$. The probability of failure is at least

$$(1 - p)/(1 + k) + pk/(1 + k) \geq \min(k, 1)/(1 + k) \geq \xi.$$

Therefore, the probability of failure of the algorithm is $2^{-\Omega(n)}$. $\qquad\square$

We could have used the uniform distribution for $\mathcal{D}$, but a discrete gaussian is more efficient.

Lemma 14. *Let Λ be a one dimensional lattice. Then,*

$$\mathbb{E}_{\mathbf{x} \sim D_\Lambda}[||\mathbf{x}||^2] \leq \frac{1}{2\pi}.$$

Proof. Let $f(\mathbf{x}) = ||\mathbf{x}||^2 \rho(\mathbf{x})$. Then, using a Poisson summation :

$$\frac{f(\Lambda)}{\rho(\Lambda)} = \frac{\sum_{\mathbf{x} \in \Lambda^*} \left(\frac{1}{2\pi} - ||\mathbf{x}||^2\right) \rho(\mathbf{x})}{\rho(\Lambda^*)} \leq \frac{1}{2\pi}.$$

$\square$

Lemma 15. *Let $\mathcal{D}$ be such that $\mathcal{D}(\mathbf{x}) \propto \exp(-||\mathbf{x}||^2/(2\sigma^2))$ for all $\mathbf{x}$ with $||\mathbf{x}||_\infty \leq B$. Then $\mathcal{D}$ is polynomially samplable for $B - R \geq 2\sigma$, and using the definitions of [Lemma 13](#), we have for $\xi = \exp(-nR^2/(2\sigma^2) - 2\sqrt{n}R/\sigma)$, $\epsilon \geq \exp(-2n\exp(-((B-R)/\sigma - 1)^2/2))/2$.*

Proof. Using the Banaszczyk lemma , we have :

$$\frac{\sum_{x=R-B}^{B-R} \exp(-x^2/(2\sigma^2))}{\sum_{x=-B}^{B} \exp(-x^2/(2\sigma^2))} \geq \frac{\sum_{x=R-B}^{B-R} \exp(-x^2/(2\sigma^2))}{\sum_{x \in \mathbb{Z}} \exp(-x^2/(2\sigma^2))} \geq 1 - \exp(-((B-R)/\sigma - 1)^2/2).$$

So that :

$$\Pr_{\mathbf{x} \sim \mathcal{D}}[||\mathbf{x}||_\infty \leq B - R] \geq \exp(-2n\exp(-((B-R)/\sigma - 1)^2/2)).$$

Since the discrete Gaussian distribution over $\mathbb{Z}$ is polynomially samplable, this is also the case for $\mathcal{D}$ since $B - R \geq 2\sigma$.

We now condition the distribution over $||\mathbf{x}||_\infty \leq B - R$. For some $\mathbf{s}$ such that $||\mathbf{s}||_\infty \leq R$ and $N = ||\mathbf{s}||$, we study the variable

$$\ell = 2\sigma^2 \ln(\exp(-||\mathbf{x} + \mathbf{s}||^2/(2\sigma^2))/\exp(-||\mathbf{x}||^2/(2\sigma^2))) = N^2 - 2\langle \mathbf{x}, \mathbf{s} \rangle.$$

By symmetry :

$$\mathbb{E}[\ell] = \mathbb{E}[N^2 - 2\langle x, s \rangle] = N^2.$$

Let $\mathcal{D}'$ be the distribution over the first coordinate and $v = s_0$. Then, using the previous lemma :

$$\mathrm{Var}[v^2 - xv] = \mathbb{E}_{x \sim \mathcal{D}'}[(v^2 - xv - v^2)^2] \leq \mathbb{E}_{x \sim D_{\mathbb{Z}, \sigma/\sqrt{2\pi}}}[x^2]v^2 \leq \sigma^2 v^2.$$

Summing over all coordinates, we have :

$$\mathrm{Var}[\ell] \leq 4\sigma^2 N^2.$$

By the Chebyshev inequality, we get :

$$\Pr[|\ell + N^2| \geq 4N\sigma] \leq \frac{1}{2}.$$

And the claim follows. $\square$

Theorem 10. *One can solve any $\mathsf{GapSVP}^{||\cdot||_\infty}_{o(B\sqrt{\log\log\log\beta/\log\log\beta}),\beta}$ in time*

$$2^{(n/2+o(n))/\ln(1+\log\beta/\log B)}$$

for $\beta = 2^{o(n/\log n)}$, $\beta = \omega(1)$, $B \geq 2$.

Proof. Use the previous lemma with $\sigma = B/\sqrt{3\ln\log\log\beta}$ and [Corollary 2](#) with $\beta' = \beta/\log(\beta)$, so that it is sufficient to decode $2^{o(n/\log\log\beta)}$ points. $\square$

Theorem 11. *If it is possible to solve* $\mathsf{BDD}_{B,\beta}^{||\cdot||_\infty}$ *in polynomial time, then it is possible to solve in randomized polynomial time* $\mathsf{GapSVP}_{B/\sqrt{n},\beta\sqrt{n/\log n}}^{||\cdot||_\infty}$.

Proof. Use $\sigma = B/\sqrt{3\ln n}$. $\qquad\square$

We now proves the corresponding theorem in norm L2.

Theorem 12. *Let $\mathcal{D}$ be the uniform distribution over the points $\boldsymbol{x}$ of $\mathbb{Z}^n$ with $||\boldsymbol{x}|| \leq B$. Then, $\mathcal{D}$ is samplable in time $\mathcal{O}(n^2 B^2 \log B)$ and for any $\boldsymbol{s}$ with $||\boldsymbol{s}|| \leq R$ and $\boldsymbol{x}$ sampled according to $\mathcal{D}$, we have for n sufficiently large*

$$\Pr[||\boldsymbol{x} + \boldsymbol{s}|| \leq B] \geq (\sqrt{B^2 - R^2} - \sqrt{n}/2)^{n-1}(R - \sqrt{n}/2)/(B + \sqrt{n}/2)^n.$$

Proof. It is easy to see that a dynamic programming algorithm using $\mathcal{O}(nB^2)$ operations on integers smaller than B^n can determine the i-th lattice point of the ball, ordered lexicographically. Therefore, $\mathcal{D}$ is samplable.

Let E be the right circular cylinder of center $\boldsymbol{s}/2$, with generating segments parallel to $\boldsymbol{s}$, of length R and radius $r = \sqrt{B^2 - R^2}$, and H the same cylinder with length $R - \sqrt{n}/2$ and radius $r - \sqrt{n}/2$. For the lattice point $\boldsymbol{x} \in \mathbb{Z}^n$, let $F_{\boldsymbol{x}}$ be the axis-aligned cube of length 1 centered on $\boldsymbol{x}$. Let F be the union of all $F_{\boldsymbol{x}}$ such that $F_{\boldsymbol{x}} \subset E$. We have $|E \cap \mathbb{Z}^n| = \mathrm{vol}(F)$ and $H \subset F$. Therefore, $|E \cap \mathbb{Z}^n| \geq V_{n-1}(r - \sqrt{n}/2)^{n-1}(R - \sqrt{n}/2)$. Also, E is a subset of the intersection of the balls of radius B and centers $\boldsymbol{0}$ and $\boldsymbol{s}$. Using Lemma 16 and $V_{n-1} \geq V_n$ for n sufficiently large, the result follows. $\qquad\square$

Corollary 3. *One can solve* $\mathsf{GapSVP}_{\sqrt{n}o(B/\sqrt{\log\log\beta}),\beta}^{||\cdot||}$, $\beta = \omega(1)$ *and* $\beta = 2^{o(n/\log n)}$ *in time*

$$2^{(n/2+o(n))/\ln(1+\log\beta/\log B)}.$$

Proof. Apply Theorem 6 with a reduction to BDD with parameter $\beta' = \beta/\log(\beta)$ and $B' = \max(B, \log\beta)$. Then, apply the previous theorem with Lemma 13. $\qquad\square$

Corollary 4. *It is possible to solve any* $\mathsf{GapSVP}_{\sqrt{n}2^{\sqrt{\log n}},n^c}^{||\cdot||}$ *with $c > 0$ in time $2^{(n+o(n))/\ln\ln n}$.*

Proof. Use the previous corollary with $B = 2^{\sqrt{\log n}}\log\log n$ and $\beta = n^c$. $\qquad\square$

Theorem 13. *If it is possible to solve* $\mathsf{BDD}_{B,\beta}^{||\cdot||_\infty}$ *in polynomial time, then it is possible to solve in randomized polynomial time* $\mathsf{GapSVP}_{B/\sqrt{n},\beta\sqrt{n/\log n}}^{||\cdot||_\infty}$.

5 Other applications

5.1 Low density subset-sum problem

Definition 10. *We are given a vector $\mathbf{a} \in \mathbb{Z}^n$ whose coordinates are sampled independently and uniformly in $[0; M)$, and $\langle\mathbf{a},\mathbf{s}\rangle$ where the coordinates of $\mathbf{s}$ are sampled independently and uniformly in $\{0,1\}$. The goal is to find $\mathbf{s}$. The density is defined as $d = \frac{n}{\log M}$.*

Note that this problem is trivially equivalent to the *modular* subset-sum problem, where we are given $\langle\boldsymbol{a},\boldsymbol{s}\rangle \bmod M$ by trying all possible $\lfloor\langle\boldsymbol{a},\boldsymbol{s}\rangle/M\rfloor$.

In [31,16], Lagarias *et al.* reduce the subset sum problem to UniqueSVP, even though this problem was not defined at that time. We will show a reduction to $\mathsf{BDD}_{1,\Omega(2^d)}^{||\cdot||_\infty}$, which is essentially the same. First, we need two geometric lemmata.

Lemma 16. *Let $\mathcal{B}_n(r)$, the number of points of $\mathbb{Z}^n$ of norm smaller than r, and V_n the volume of the unit ball. Then,*

$$\mathcal{B}_n(r) \leq V_n\left(r + \frac{\sqrt{n}}{2}\right)^n.$$

Proof. For each $\mathbf{x} \in \mathbb{Z}^n$, let $E_{\mathbf{x}}$ be a cube of length 1 centered on $\mathbf{x}$. Let E be the union of all the $E_{\mathbf{x}}$ which have a non empty intersection with the ball of center $\mathbf{0}$ and radius r. Therefore $\mathrm{vol}(E) \geq \mathcal{B}_n(r)$ and since E is included in the ball of center $\mathbf{0}$ and radius $r + \frac{\sqrt{n}}{2}$, the claim is proven. $\qquad\square$

Lemma 17. *For $n \geq 4$ we have*

$$V_n = \frac{\pi^{n/2}}{(n/2)!} \leq (\sqrt{\pi e/n})^n$$

Theorem 14. *Using one call to a $\mathsf{BDD}_{1,c2^{1/d}}^{||\cdot||_\infty}$ oracle with any $c < \sqrt{2/\pi/e}$ and $d = o(1)$, and polynomial time, it is possible to solve a subset-sum problem of density d, with negligible probability of failure.*

Proof. With the matrix :

$$\mathbf{A} = \begin{pmatrix} \mathbf{I} \\ C\mathbf{a} \end{pmatrix}$$

for some $C > c2^{1/d}\sqrt{n}/2$ and $\mathbf{b} = (1/2, \ldots, 1/2, C\langle \mathbf{a}, \mathbf{s}\rangle)$, return $\mathsf{BDD}(\mathbf{A}, \mathbf{b})$. It is clear that $||\mathbf{As} - \mathbf{b}|| = \sqrt{n}/2$. Now, let $\mathbf{x}$ such that $||\mathbf{Ax}|| = \lambda_1$. If $\langle \mathbf{a}, \mathbf{x}\rangle \neq 0$, then $\lambda_1 = ||\mathbf{Ax}|| \geq C$ therefore $\beta \geq c2^{1/d}$. Else, $\langle \mathbf{a}, \mathbf{x}\rangle = 0$. Without loss of generality, $x_0 \neq 0$, we let $y = -(\sum_{i>0} a_i x_i)/x_0$ and the probability over $\mathbf{a}$ that $\langle \mathbf{a}, \mathbf{x}\rangle = 0$ is :

$$\Pr[\langle \mathbf{a}, \mathbf{x}\rangle = 0] = \Pr[a_0 = y] = \sum_{z=0}^{M-1} \Pr[y = z]\Pr[a_0 = z] \leq \frac{1}{M}.$$

Therefore, the probability of failure is at most, for sufficiently large n,

$$\begin{aligned}\mathcal{B}_n(\beta\sqrt{n}/2)/M &\leq (\sqrt{\pi e/n})^n (c2^{1/d}\sqrt{n}/2 + \sqrt{n}/2)^n/2^{n/d} \\ &= \left(\sqrt{\pi e/2}(c + 2^{-1/d})\right)^n = 2^{-\Omega(n)}. \qquad\square\end{aligned}$$

Corollary 5. *For any $d = o(1)$ and $d = \omega(\log n/n)$, we can solve the subset-sum problem of density d with negligible probability of failure in time $2^{(n/2 + o(n))/\ln(1/d)}$.*

The cryptosystem of Lyubashevsky *et al.* [37] uses $2^{1/d} > 10n\log^2 n$ and is therefore broken in time $2^{(\ln 2/2 + o(1))n/\log\log n}$. Current lattice reduction algorithms are slower than this one when $d = \omega(1/(\log n \log\log n))$.

5.2 Sample Expander and application to LWE with binary errors

Definition 11. *Let q be a prime number. The problem Small-DecisionLWE is to distinguish $(\mathbf{A}, \mathbf{b})$ with $\mathbf{A}$ sampled uniformly with n columns and m rows, $\mathbf{b} = \mathbf{As} + \mathbf{e}$ such that $||\mathbf{s}||^2 + ||\mathbf{e}||^2 \leq nk^2$ and $||\mathbf{s}|| \leq \sqrt{n}B$ from $(\mathbf{A}, \mathbf{b})$ sampled uniformly. Also, the distribution $(\mathbf{s}, \mathbf{e})$ is efficiently samplable.*

The problem Small-SearchLWE is to find $\mathbf{s}$ given $(\mathbf{A}, \mathbf{b})$ with $\mathbf{A}$ sampled uniformly and $\mathbf{b} = \mathbf{As} + \mathbf{e}$ with the same conditions on $\mathbf{s}$ and $\mathbf{e}$.

These problems are generalizations of BinaryLWE where $\mathbf{s}$ and $\mathbf{e}$ have coordinates sampled uniformly in $\{0, 1\}$. In this case, remark that each sample is a root of a known quadratic polynomial in the coordinates of $\mathbf{s}$. Therefore, it is easy to solve this problem when $m \geq n^2$. For $m = \mathcal{O}(n)$, a Gröbner basis algorithm applied on this system will (heuristically) have a complexity of $2^{\Omega(n)}$ [3]. For $m = \mathcal{O}(n/\log n)$ and $q = n^{\mathcal{O}(1)}$, it has been shown to be harder than a lattice problem in dimension $\Theta(n/\log n)$ [40].

We will first prove the following theorem [8], with the coordinates of $\mathbf{x}$ and $\mathbf{y}$ distributed according to a samplable $\mathcal{D}$:

Theorem 15. *Assume there is an efficient distinguisher which uses k samples for Decision-LWE (respectively a solver for Search-LWE) with error distribution $\langle \mathbf{s}, \mathbf{y}\rangle + \langle \mathbf{e}, \mathbf{x}\rangle$ of advantage (resp. success probability) ϵ.*

Then, either there is an efficient distinguisher for Decision-LWE with samples and secret taken uniformly, and error distribution $\mathcal{D}$ in dimension $m-1$ and with $n+m$ samples of advantage $\frac{\xi}{4qk} - q^{-n} - q^{-m}$; or there is an efficient distinguisher of advantage $\epsilon - \xi$ for Small-Decision-LWE (resp. solver of success probability $\epsilon - \xi$ for Small-Search-LWE).

---

[8] In [19], the authors gave a short justification of a similar claim which is far from proven.

Sample Expander The reduction is a generalization and impovement of Döttling's reduction [17], which was made for LPN.

Lemma 18 (Hybrid lemma [22]). *We are given a distinguisher between the distributions $\mathcal{D}_0$ and $\mathcal{D}_1$ of advantage ϵ which needs m samples. Then, there exists a distinguisher between the distributions $\mathcal{D}_0$ and $\mathcal{D}_1$ of advantage ϵ/m which needs one sample, m samples of distribution $\mathcal{D}_0$ and m samples of distribution $\mathcal{D}_1$.*

Proof. The distinguisher samples an integer i uniformly between 0 and $m-1$. It then returns the output of the oracle, whose input is i samples of $\mathcal{D}_0$, the given sample, and $m - i - 1$ samples of $\mathcal{D}_1$.

Let p_i the probability that the oracle outputs 1 if its input is given by i samples of $\mathcal{D}_0$ followed by $m - i$ samples of $\mathcal{D}_1$. Then, if the sample comes from $\mathcal{D}_0$, the probability of outputting 1 is $\frac{1}{m}\sum_{i=1}^{m}p_i$. Else, it is $\frac{1}{m}\sum_{i=0}^{m-1}p_i$. Therefore, the advantage of our algorithm is $\frac{1}{m}|p_m - p_0| = \frac{\epsilon}{m}$. $\qquad\square$

In the following, we will let, for $\mathcal{D}$ some samplable distribution over $\mathbb{Z}/q\mathbb{Z}$, $\mathbf{x}$ be sampled according to $\mathcal{D}^m$, $\mathbf{y}$ be sampled according to $\mathcal{D}^n$, $\mathbf{z}$ according to $(\mathbf{e}|\mathbf{s})$, $\mathbf{u}$ be sampled uniformly in $(\mathbb{Z}/q\mathbb{Z})^m$, v be sampled uniformly in $\mathbb{Z}/q\mathbb{Z}$, $\mathbf{w}$ in $(\mathbb{Z}/q\mathbb{Z})^n$.

Definition 12. *The problem* Knapsack-LWE *is to distinguish between* $(\mathbf{G}, \mathbf{c})$ *where* $\mathbf{G}$ *is always sampled uniformly in* $(\mathbb{Z}/q\mathbb{Z})^{m \times (m+n)}$, *and* $\mathbf{c}$ *is either uniform, or sampled according to* $\mathbf{G}\mathcal{D}^{m+n}$.

The problem Extended-LWE *is to distinguish between* $(\mathbf{A}, {}^t\mathbf{x}\mathbf{A} + \mathbf{y}, \mathbf{z}, \langle(\mathbf{x}|\mathbf{y}), \mathbf{z}\rangle)$ *and* $(\mathbf{A}, \mathbf{w}, \mathbf{z}, \langle(\mathbf{x}|\mathbf{y}), \mathbf{z}\rangle)$.

The problem First-is-errorless-LWE *is to distinguish between* $(\mathbf{A}, {}^t\mathbf{x}\mathbf{A} + \mathbf{y}, \mathbf{u}, \langle\mathbf{x}, \mathbf{u}\rangle)$ *and* $(\mathbf{A}, \mathbf{w}, \mathbf{u}, \langle\mathbf{x}, \mathbf{u}\rangle)$.

The following lemma comes from [39], we give a sketch of their proof.

Lemma 19. *The problems* Knapsack-LWE *and* Decision-LWE *in dimension m with $n + m$ samples where are equivalent: there are reductions in both ways which reduce the advantage by q^{-n-1}.*

Proof. Given the Decision-LWE problem $(\mathbf{A}, \mathbf{b})$, we sample a uniform basis $\mathbf{G}$ of the left kernel of $\mathbf{A}$ and outputs $(\mathbf{G}, \mathbf{G}\mathbf{b})$. If $\mathbf{b} = \mathbf{A}\mathbf{s} + \mathbf{e}$, we have therefore $\mathbf{G}\mathbf{b}$ distributed as $\mathbf{G}\mathbf{e}$.

Given the Knapsack-LWE problem $(\mathbf{G}, \mathbf{c})$, we sample a uniform basis $\mathbf{A}$ of the right kernel of $\mathbf{A}$, and a uniform $\mathbf{b}$ such that $\mathbf{G}\mathbf{b} = \mathbf{c}$ and outputs $(\mathbf{A}, \mathbf{b})$. If $\mathbf{c} = \mathbf{G}\mathbf{e}$, then $\mathbf{b}$ is distributed as $\mathbf{A}\mathbf{s} + \mathbf{e}$ where $\mathbf{s}$ is uniform.

Both reductions map the distributions to their counterparts, except when $\mathbf{A}$ or $\mathbf{G}$ are not full rank, which happens with probability at most q^{-n-1}, hence the result. $\qquad\square$

The lemma is a slight modification of the lemma given in [6].

Lemma 20. *Given a distinguisher of advantage ϵ for* ExtendedLWE*, there is a distinguisher of advantage $\frac{\epsilon(1 - 1/q) - 2q^{-n}}{q}$ for* Decision-LWE *in dimension m, uniform secret, with $n + m$ uniform samples of error distribution $\mathcal{D}$.*

Proof. Using the previous lemma, we can assume that we want to solve a KnapsackLWE problem. Let $(\mathbf{G}, \mathbf{c})$ be its input, with $\mathbf{c} = \mathbf{G}\mathbf{e}$. We start by sampling $\mathbf{z}, \mathbf{t}$ uniformly over $(\mathbb{Z}/q\mathbb{Z})^m$ and $\mathbf{e}'$ according to $\mathcal{D}^{n+m}$.

We then compute $\mathbf{G}' = \mathbf{G} - \mathbf{t}^t\mathbf{z}$ and $\mathbf{c}' = \mathbf{c} - \langle\mathbf{z}, \mathbf{e}'\rangle\mathbf{t}$. Remark that $\mathbf{G}'$ is uniform and $\mathbf{c}' = \mathbf{G}\mathbf{e} - \langle\mathbf{z}, \mathbf{e}'\rangle\mathbf{t} = \mathbf{G}'\mathbf{e} + \langle\mathbf{z}, \mathbf{e} - \mathbf{e}'\rangle\mathbf{t}$. Therefore, if $\langle\mathbf{z}, \mathbf{e}'\rangle = \langle\mathbf{z}, \mathbf{e}\rangle$, which happens with probability at least $\frac{1}{q}$, $(\mathbf{G}', \mathbf{c}')$ comes from the same distribution as $(\mathbf{G}, \mathbf{c})$. Also, $\langle\mathbf{z}, \mathbf{e}\rangle$ is known. Else, since $\mathbf{t}$ is uniform, $(\mathbf{G}', \mathbf{c}')$ is uniform.

We then perform the previous reduction to a Decision-LWE problem, where $\langle\mathbf{z}, \mathbf{e}\rangle$ is known. Finally, we use Theorem 1 to reduce to Extended-LWE. If it doesn't output n samples, which happens with probability at most $1/q$, our distinguisher returns a uniform boolean. Else, remark that we have $(\mathbf{x}|\mathbf{y}) = \mathbf{e}$ so that $\langle(\mathbf{x}|\mathbf{y}), \mathbf{z}\rangle$ is known and we use the given distinguisher. $\qquad\square$

Lemma 21. *Assume there is a distinguisher of advantage ϵ for* First-is-errorless-LWE*. Then, there is a distinguisher of advantage at least $\epsilon(1 - 1/q) - q^{-m}$ and* Decision-LWE *in dimension $m - 1$ with uniform secret and $n + m$ uniform samples of error distribution $\mathcal{D}$.*

Proof. We take a Decision-LWE problem in dimension $m-1$ with $n+m$ samples, and extend it of one coordinate, the corresponding secret coordinate being sampled uniformly. We then switch the error and the secret, as in Theorem 1, and return a random boolean if the reduction do not output n samples.

We have the errorless sample $(\mathbf{e}_{m-1}, s_{m-1})$ with $\mathbf{e}_{m-1} = (0, \ldots, 0, 1)$. If we feed this sample to our reduction, it outputs an errorless sample $(\mathbf{u}, \langle \mathbf{x}, \mathbf{u} \rangle)$, and the vectorial part is distributed as a column of a uniform invertible matrix. Therefore, the statistical distance between the distribution of $\mathbf{u}$ and the uniform distribution is at most q^{-m}. $\qquad\square$

Theorem 16. *Assume there is an efficient distinguisher which uses k samples for* Decision-LWE *(respectively a solver for* Search-LWE*) with error distribution $\langle \mathbf{s}, \mathbf{y} \rangle + \langle \mathbf{e}, \mathbf{x} \rangle$ of advantage (resp. success probability) ϵ.*

Then, either there is an efficient distinguisher for Decision-LWE *with samples and secret taken uniformly, and error distribution $\mathcal{D}$ in dimension $m-1$ and with $n+m$ samples of advantage $\frac{\xi}{4qk} - q^{-n} - q^{-m}$; or there is an efficient distinguisher of advantage $\epsilon - \xi$ for* Small-Decision-LWE *(resp. solver of success probability $\epsilon - \xi$ for* Small-Search-LWE*).*

Proof. Let

- $\mathcal{D}_0$ be the distribution given by $(\mathbf{w}, \langle \mathbf{w}, \mathbf{s} \rangle + \langle \mathbf{s}, \mathbf{y} \rangle + \langle \mathbf{e}, \mathbf{x} \rangle)$
- $\mathcal{D}_1$ be the distribution given by $({}^t\mathbf{x}\mathbf{A} + \mathbf{y}, \langle \mathbf{x}, \mathbf{b} \rangle)$
- $\mathcal{D}_2$ be the distribution given by $({}^t\mathbf{x}\mathbf{A} + \mathbf{y}, \langle \mathbf{x}, \mathbf{u} \rangle)$
- $\mathcal{D}_3$ be the distribution given by $(\mathbf{w}, v)$.

Let ϵ_i be the advantage of D between $\mathcal{D}_i$ and $\mathcal{D}_{i+1}$.

Let $(\mathbf{A}, \mathbf{c}, (\mathbf{e}|\mathbf{s}), \langle (\mathbf{x}|\mathbf{y}), (\mathbf{e}|\mathbf{s}) \rangle)$ be an instance of the Extended-LWE problem. We compute $\mathbf{b} = \mathbf{A}\mathbf{s} - \mathbf{e}$, and use the hybrid lemma with distributions $\mathcal{D}_0$ and $\mathcal{D}_1$, and sample $(\mathbf{c}, \langle \mathbf{c}, \mathbf{s} \rangle + \langle (\mathbf{x}|\mathbf{y}), (\mathbf{e}|\mathbf{s}) \rangle)$. If $\mathbf{c} = {}^t\mathbf{x}\mathbf{A} + \mathbf{y}$, we have $\langle \mathbf{c}, \mathbf{s} \rangle = \langle \mathbf{x}, \mathbf{b} \rangle + \langle \mathbf{x}, \mathbf{e} \rangle + \langle \mathbf{y}, \mathbf{s} \rangle$ so that the sample is distributed according to $\mathcal{D}_1$. Therefore, we have a distinguisher for Extended-LWE of advantage $\frac{\epsilon_0}{k}$.

Clearly, there is a distinguisher of advantage ϵ_1 against Small-Decision-LWE.

Let $(\mathbf{A}, \mathbf{c}, \mathbf{u}, \mathbf{r})$ be an instance of the First-is-errorless-LWE problem. The hybrid lemma with distributions $\mathcal{D}_2$ and $\mathcal{D}_3$, and sample $(\mathbf{c}, \mathbf{r})$ shows that there exist an efficient distinguisher for First-is-errorless-LWE with advantage ϵ_3/k.

Since $\epsilon_0 + \epsilon_1 + \epsilon_2 \geq \epsilon$, using the previous lemmata, and the fact that Decision-LWE is harder in dimension m than $m-1$, the theorem follows.

For the search version, the indistinguishability of $\mathcal{D}_0$ and $\mathcal{D}_1$ is sufficient. $\qquad\square$

Döttling's reduction has a uniform secret, so that a problem solvable in time $2^{\mathcal{O}(\sqrt{n})}$ is transformed into a problem where the best algorithm takes time $2^{(1+o(1))n/\log n}$. We do not have such dramatic loss here.

Applications

Lemma 22. *Let $\mathcal{D} = D_{\mathbb{Z},\sigma}$ for $\sigma \geq 1$.*

Then, the advantage of a distinguisher for Decision-LWE *of dimension m with $m+n$ samples of noise distribution $\mathcal{D}$ is at most $\sqrt{q^n/\sigma^{n+m}}$. Furthermore, the bias of $\langle (\mathbf{s}|\mathbf{e}), (\mathbf{x}|\mathbf{y}) \rangle$, for fixed $\mathbf{s}$ and $\mathbf{e}$, is at least $\exp(-\pi(||\mathbf{s}||^2 + ||\mathbf{e}||^2)\sigma^2/q^2)$.*

Proof. We have $\mathcal{D}^{m+n}(\mathbf{a}) \leq \mathcal{D}(0)^{m+n} = 1/\rho_\sigma(\mathbb{Z})^{m+n}$ and $\rho_\sigma(\mathbb{Z}) = \sigma\rho_{1/\sigma}(\mathbb{Z}) \geq \sigma$ using a Poisson summation. The first property is then a direct application of the leftover hash lemma, since q is prime.

The bias of $\lambda\mathcal{D}$ can be computed using a Poisson summation as :

$$\sum_{a \in \mathbb{Z}} \rho_\sigma(a) \cos(2\pi\lambda a/q) = \rho_{1/\sigma}(\mathbb{Z} + \lambda/q) \geq \exp(-\pi\lambda^2\sigma^2/q^2).$$

Therefore, the second property follows from the independency of the coordinates of $\mathbf{x}$ and $\mathbf{y}$. $\qquad\square$

Corollary 6. *Let q, n and m such that $(m-3)\log q/(n+m)-\log k = \omega(1)$ and $m\log q/(n+m) = o(n/\log n)$. Then, we can solve* Small-Decision-LWE *in time*

$$2^{(n/2+o(n))/\ln(1+((m-3)\log q/(n+m)-\log k)/\log B)}$$

with negligible probability of failure.

Proof. We use the previous lemma with $\sigma = 2q^{(n+2)/(n+m-1)}$, so that we have $\beta = \Omega(q^{(m-3)/(n+m)}/k)$. The algorithm from Theorem 6 needs $2^{o(n)}$ samples, so that the advantage of the potential distinguisher for Decision-LWE is $2^{-n/4+o(n)}/q$ for $\xi = 2^{-n/4}$; while the previous lemma proves it is less than $2^{-n/2}/q$. $\square$

The NTRU cryptosystem [27] is based on the hardness of finding two polynomials f and g whose coefficients are bounded by 1 given $h = f/g \mod (X^n-1, q)$. Since $hg = 0$ with an error bounded by 1, we can apply previous algorithms in this section to *heuristically* recover f and g in time $2^{(n/2+o(1))/\ln\ln q}$. This is the first subexponential time algorithm for this problem since it was introduced back in 1998.

Corollary 7. *Assume we have a* Search-LWE *problem with at least $n\log q + r$ samples and Gaussian noise with $\alpha = n^{-c}$ and $q = n^d$. Then, we can solve it in time $2^{n/(2\ln(d/(d-c))-o(1))}$ for any failure probability in $2^{-n^{o(1)}} + q^{-r}$.*

Proof. First, apply a secret-error switching (Theorem 1) and assume we lose at most r samples. Apply the previous corollary with $B = n^{d-c+o(1)}$ which is a correct bound for the secret, except with probability $2^{-n^{o(1)}}$. Lemma 11 shows that $k^2 \leq \log q\sigma^2$, except with probability $2^{-\Omega(n)}$, so that $\beta = n^{c+o(1)}$. We can then use $\sigma = \Theta(1)$ and apply Theorem 4. $\square$

Note that this corollary can in fact be applied to a very large class of distributions, and in particular to the learning with rounding problem, while the distortion parameter is too large for a direct application of Theorem 4.

Also, if the reduction gives a fast (subexponential) algorithm, one may use $\sigma = 2\sqrt{n}$ and assume that there is no quantum algorithm solving the corresponding lattice problem in dimension m.

Even more heuristically, one can choose σ to be the lowest such that if the reduction does not work, we have an algorithm faster than the best *known* algorithm for the same problem.

References

1. *Proceedings of the 25th Annual IEEE Conference on Computational Complexity, CCC 2010, Cambridge, Massachusetts, June 9-12, 2010.* IEEE Computer Society, 2010.
2. D. Aggarwal, D. Dadush, O. Regev, and N. Stephens-Davidowitz. Solving the shortest vector problem in 2^n time via discrete Gaussian sampling. *arXiv preprint arXiv:1412.7994*, 2015. http://arxiv.org/pdf/1412.7994.
3. M. R. Albrecht, C. Cid, J. Faugère, R. Fitzpatrick, and L. Perret. Algebraic algorithms for LWE problems. *IACR Cryptology ePrint Archive*, 2014:1018, 2014.
4. M. R. Albrecht, C. Cid, J. Faugère, R. Fitzpatrick, and L. Perret. On the complexity of the BKW algorithm on LWE. *Des. Codes Cryptography*, 74(2):325–354, 2015.
5. M. R. Albrecht, J. Faugère, R. Fitzpatrick, and L. Perret. Lazy modulus switching for the BKW algorithm on LWE. In H. Krawczyk, editor, *Public-Key Cryptography - PKC 2014 - 17th International Conference on Practice and Theory in Public-Key Cryptography, Buenos Aires, Argentina, March 26-28, 2014. Proceedings*, volume 8383 of *Lecture Notes in Computer Science*, pages 429–445. Springer, 2014.
6. J. Alperin-Sheriff and C. Peikert. Circular and kdm security for identity-based encryption. In *Public Key Cryptography–PKC 2012*, pages 334–352. Springer, 2012. http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.352.8723&rep=rep1&type=pdf.
7. B. Applebaum, D. Cash, C. Peikert, and A. Sahai. Fast cryptographic primitives and circular-secure encryption based on hard learning problems. In Halevi [26], pages 595–618.

8. D. F. Aranha, P. Fouque, B. Gérard, J. Kammerer, M. Tibouchi, and J. Zapalowicz. GLV/GLS decomposition, power analysis, and attacks on ECDSA signatures with single-bit nonce bias. In P. Sarkar and T. Iwata, editors, *Advances in Cryptology - ASIACRYPT 2014 - 20th International Conference on the Theory and Application of Cryptology and Information Security, Kaoshiung, Taiwan, R.O.C., December 7-11, 2014. Proceedings, Part I*, volume 8873 of *Lecture Notes in Computer Science*, pages 262–281. Springer, 2014.

9. S. Arora and R. Ge. New algorithms for learning in presence of errors. In L. Aceto, M. Henzinger, and J. Sgall, editors, *Automata, Languages and Programming - 38th International Colloquium, ICALP 2011, Zurich, Switzerland, July 4-8, 2011, Proceedings, Part I*, volume 6755 of *Lecture Notes in Computer Science*, pages 403–415. Springer, 2011.

10. W. Banaszczyk. New bounds in some transference theorems in the geometry of numbers. *Mathematische Annalen*, 296(1):625–635, 1993.

11. D. J. Bernstein and T. Lange. Never trust a bunny. In *Radio Frequency Identification. Security and Privacy Issues*, pages 137–148. Springer, 2013. `https://eprint.iacr.org/2012/355.pdf`.

12. A. Blum, A. Kalai, and H. Wasserman. Noise-tolerant learning, the parity problem, and the statistical query model. *J. ACM*, 50(4):506–519, 2003.

13. Z. Brakerski, A. Langlois, C. Peikert, O. Regev, and D. Stehlé. Classical hardness of learning with errors. In *Symposium on Theory of Computing Conference, STOC'13*, pages 575–584, 2013. `http://perso.ens-lyon.fr/damien.stehle/downloads/LWE.pdf`.

14. Z. Brakerski and V. Vaikuntanathan. Efficient fully homomorphic encryption from (standard) lwe. *SIAM J. Comput.*, 43(2):831–871, 2014.

15. Y. Chen and P. Q. Nguyen. BKZ 2.0: Better lattice security estimates. In *Advances in Cryptology - ASIACRYPT 2011. Proceedings*, pages 1–20, 2011.

16. M. J. Coster, A. Joux, B. A. LaMacchia, A. M. Odlyzko, C. Schnorr, and J. Stern. Improved low-density subset sum algorithms. *Computational Complexity*, 2:111–128, 1992.

17. N. Döttling. Low noise lpn: Kdm secure public key encryption and sample amplification. 2015. `https://eprint.iacr.org/2015/013.pdf`.

18. N. Döttling and J. Müller-Quade. Lossy codes and a new variant of the learning-with-errors problem. In T. Johansson and P. Q. Nguyen, editors, *Advances in Cryptology - EUROCRYPT 2013, 32nd Annual International Conference on the Theory and Applications of Cryptographic Techniques, Athens, Greece, May 26-30, 2013. Proceedings*, volume 7881 of *Lecture Notes in Computer Science*, pages 18–34. Springer, 2013.

19. A. Duc, F. Tramèr, and S. Vaudenay. Better algorithms for lwe and lwr. Cryptology ePrint Archive, Report 2015/056, 2015. `http://eprint.iacr.org/`.

20. N. Gama and P. Q. Nguyen. Finding short lattice vectors within mordell's inequality. In C. Dwork, editor, *Proceedings of the 40th Annual ACM Symposium on Theory of Computing, Victoria, British Columbia, Canada, May 17-20, 2008*, pages 207–216. ACM, 2008.

21. C. Gentry, A. Sahai, and B. Waters. Homomorphic encryption from learning with errors: Conceptually-simpler, asymptotically-faster, attribute-based. In *Advances in Cryptology - CRYPTO 2013 - 33rd Annual Cryptology Conference, Santa Barbara, CA, USA, August 18-22, 2013. Proceedings, Part I*, pages 75–92, 2013.

22. O. Goldreich. *Foundations of cryptography: volume 2, basic applications*. Cambridge university press, 2004.

23. O. Goldreich and S. Goldwasser. On the limits of non-approximability of lattice problems. In *Proceedings of the thirtieth annual ACM symposium on Theory of computing*, pages 1–9. ACM, 1998. `http://groups.csail.mit.edu/cis/pubs/shafi/2000-jcss.pdf`.

24. R. M. Gray and D. L. Neuhoff. Quantization. *Information Theory, IEEE Transactions on*, 44(6):2325–2383, 1998. `http://citeseer.ist.psu.edu/viewdoc/download;jsessionid=BA5DF2EF01ED127A60AB8A3C54DF65A1?doi=10.1.1.116.3824&rep=rep1&type=pdf`.

25. Q. Guo, T. Johansson, and C. Löndahl. Solving lpn using covering codes. In *Advances in Cryptology–ASIACRYPT 2014*, pages 1–20. Springer, 2014.

26. S. Halevi, editor. *Advances in Cryptology - CRYPTO 2009, 29th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 16-20, 2009. Proceedings*, volume 5677 of *Lecture Notes in Computer Science*. Springer, 2009.

27. J. Hoffstein, J. Pipher, and J. H. Silverman. Ntru: A ring-based public key cryptosystem. In *Algorithmic number theory*, pages 267–288. Springer, 1998. `http://www.academia.edu/download/30238047/joe_p_buhler_algorithmic_number_theory_3_sympos.pdf`.

28. P. Kirchner. Improved generalized birthday attack. *IACR Cryptology ePrint Archive*, 2011:377, 2011. `http://eprint.iacr.org/2011/377.pdf`.

29. T. Laarhoven. Sieving for shortest vectors in lattices using angular locality-sensitive hashing. Technical report, Cryptology ePrint Archive, Report 2014/744, 2014. `https://eprint.iacr.org/2014/744.pdf`.

30. T. Laarhoven, M. Mosca, and J. van de Pol. Solving the shortest vector problem in lattices faster using quantum search. *CoRR*, abs/1301.6176, 2013.

31. J. C. Lagarias and A. M. Odlyzko. Solving low-density subset sum problems. *J. ACM*, 32(1):229–246, 1985.

32. A. Lenstra, J. Hendrik Lenstra, and L. Lovász. Factoring polynomials with rational coefficients. *Mathematische Annalen*, 261:515–534, 1982.

33. É. Levieil and P. Fouque. An improved LPN algorithm. In R. D. Prisco and M. Yung, editors, *Security and Cryptography for Networks, 5th International Conference, SCN 2006, Maiori, Italy, September 6-8, 2006, Proceedings*, volume 4116 of *Lecture Notes in Computer Science*, pages 348–359. Springer, 2006.

34. R. Lindner and C. Peikert. Better key sizes (and attacks) for lwe-based encryption. In A. Kiayias, editor, *Topics in Cryptology - CT-RSA 2011 - The Cryptographers' Track at the RSA Conference 2011, San Francisco, CA, USA, February 14-18, 2011. Proceedings*, volume 6558 of *Lecture Notes in Computer Science*, pages 319–339. Springer, 2011.

35. M. Liu and P. Q. Nguyen. Solving BDD by enumeration: An update. In E. Dawson, editor, *Topics in Cryptology - CT-RSA 2013 - The Cryptographers' Track at the RSA Conference 2013, San Francisco,CA, USA, February 25-March 1, 2013. Proceedings*, volume 7779 of *Lecture Notes in Computer Science*, pages 293–309. Springer, 2013.

36. V. Lyubashevsky and D. Micciancio. On bounded distance decoding, unique shortest vectors, and the minimum distance problem. In Halevi [26], pages 577–594.

37. V. Lyubashevsky, A. Palacio, and G. Segev. Public-key cryptographic primitives provably as secure as subset sum. In D. Micciancio, editor, *Theory of Cryptography, 7th Theory of Cryptography Conference, TCC 2010, Zurich, Switzerland, February 9-11, 2010. Proceedings*, volume 5978 of *Lecture Notes in Computer Science*, pages 382–400. Springer, 2010.

38. V. Lyubashevsky, C. Peikert, and O. Regev. On ideal lattices and learning with errors over rings. *J. ACM*, 60(6):43, 2013.

39. D. Micciancio and P. Mol. Pseudorandom knapsacks and the sample complexity of lwe search-to-decision reductions. In *Advances in Cryptology–CRYPTO 2011*, pages 465–484. Springer, 2011. `https://eprint.iacr.org/2011/521.pdf`.

40. D. Micciancio and C. Peikert. Hardness of SIS and LWE with small parameters. In *Advances in Cryptology - CRYPTO 2013. Proceedings, Part I*, pages 21–39, 2013.

41. D. Micciancio and P. Voulgaris. A deterministic single exponential time algorithm for most lattice problems based on voronoi cell computations. *SIAM Journal on Computing*, 42(3):1364–1391, 2013. `http://cse.ucsd.edu/sites/cse/files/cse/assets/research/theory/MiccancioVoulgaris_2010_stoc.pdf`.

42. E. D. Mulder, M. Hutter, M. E. Marson, and P. Pearson. Using bleichenbacher's solution to the hidden number problem to attack nonce leaks in 384-bit ECDSA: extended version. *J. Cryptographic Engineering*, 4(1):33–45, 2014.

43. J. Neyman and E. S. Pearson. On the problem of the most efficient tests of statistical hypotheses. *Philosophical Transactions of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, 231(694-706):289–337, 1933.

44. C. Peikert. Public-key cryptosystems from the worst-case shortest vector problem: extended abstract. In M. Mitzenmacher, editor, *Proceedings of the 41st Annual ACM Symposium on Theory of Computing, STOC 2009, Bethesda, MD, USA, May 31 - June 2, 2009*, pages 333–342. ACM, 2009.

45. T. Pöppelmann, L. Ducas, and T. Güneysu. Enhanced lattice-based signatures on reconfigurable hardware. *IACR Cryptology ePrint Archive*, 2014:254, 2014. `https://eprint.iacr.org/2014/254.pdf`.

46. O. Regev. On lattices, learning with errors, random linear codes, and cryptography. *Journal of the ACM (JACM)*, 56(6):34, 2009. `http://www.cims.nyu.edu/~regev/papers/qcrypto.pdf`.

47. D. Wagner. A generalized birthday problem. In *Advances in Cryptology - CRYPTO 2002. Proceedings*, pages 288–303, 2002.

A LWE in Small Dimension

If the dimension n is small ($n = \mathcal{O}(\log(\beta))$), and in particular if n is one [9], we can use the same kind of algorithm. Such algorithms have been described in [42,8] and an unpublished paper of Bleichenbacher.

---

[9] This is an interesting case in cryptography for analyzing the security of DSA[42] or proving the hardness of recovering the most significant bits of a Diffie-Hellman.

The principle is to reduce at each step some coordinates and partially reduce another one. If q is too large, then the reduction of the last coordinate has to be changed: we want to have this coordinate *uniformly* distributed in the ball of dimension 1 and radius R (i.e. $[-R; R]$). To this end, for each sample, if the ball of radius R' centered at the sample is included in the ball of radius R and if it contains at least another sample, we will choose uniformly a sample in the ball [10] and then remove it and add the difference in the output list. In the other case, we add the element to our structure. The last coordinate of the outputted samples is clearly uniform over a ball of radius R'.

Finally, we apply a Fourier transform to the samples : for various potential secrets x, if we denote by (a, b) the samples where a is the last coordinate of the vectorial part and b the scalar part, we compute the experimental bias $\Re(\mathbb{E}[\exp(2i\pi(ax - b)/q)])$ and output the s which maximizes them.

We can analyze the Fourier stage of the algorithm with a uniformly distributed in a ball of radius R, an odd integer. The bias for x is $\mathbb{E}[\exp(2i\pi(e + (s - x)a)/q)]$ where $e = as - b$ is the error, which is equal to the bias of the error times $\mathbb{E}[\exp(2i\pi(s - x)a/q)]$. We take x regularly distributed so that the interval between two x is smaller than $q/R/2$, and we can compute all bias in one Fourier transform.

Let us consider an x such that $|s - x| \leq q/R/4$. The introduced bias is the bias of the error times $\frac{\sin \pi(s-x)R/q}{R \sin \pi(s-x)/q}$, which is lower bounded by a universal constant. However, for $|s - x| \geq q/R$, the introduced bias is upper bounded by a constant, strictly smaller than the previous one. Therefore, the Fourier stage determines a x such that $|s - x| < q/R$.

We will in a first time determine the secret up to an approximation factor q/R using a fast Fourier transform on $\approx 2R$ points. We are therefore reduced to a smaller instance where the secret is bounded by q/R and we can choose $R' = R^2$ for instance. Using one more time a Fourier Transform on $2R'/R = 2R$ points, we can recover the secret up to an approximation factor of $q/R' = q/R^2$. We can continue so on, and we get the final coordinate of the secret and repeating this process $n - 1$ times, we have the whole secret.

We define $\beta = \sqrt{\log q}/\alpha$ and express all variables as functions of q.

Theorem 17. *We can solve* Search-LWE *with* $n = 1$, $\beta = \omega(1)$, $\beta = q^{o(1/\log q)}$ *and distortion parameter* $\epsilon \leq \beta^{-4}$ *in time*

$$q^{(1/2 + o(1))/\log \beta}$$

with failure probability $q^{-\Omega(1)}$.

Proof. We analyze here the first iteration of the algorithm.

We set $k = \lfloor \log(\beta^2/\log \beta/3) \rfloor$ the number of iterations, $m = \Theta(\log q 2^k q^x)$ the number of input samples and $R_i = q/R^i$ such that the vectorial part of the input of the ith reduction is sampled uniformly in a ball of the largest odd radius smaller than R_i. If we have N samples, the Hoeffding inequality shows that one reduction step outputs at least $N(1/2 - 1/R - \log q/R^2) - R$ samples, except with probability $q^{-\Omega(1)}$. Therefore, we can use $R = q^x$. For simplicity, we then set $R_k = 3$ so that $R \leq q^{1/a}$. The bias of the error is at least $\exp(-2^k\alpha^2)(1 - \epsilon 3^k) \geq \exp(-\log q/\log \beta/3)/2$ so that taking $x = 1/k$ works.

The overall complexity is $\mathcal{O}(\log^3 q \beta^2 q^{1/k})$ which is $q^{(1/2 + o(1))/\log \beta}$. $\square$

B   Solving LWE with Lattice Reduction

Lattice reduction consists in finding a basis of a lattice with short vectors. The best algorithm[20] uses a polynomial number of times an oracle which returns the shortest non-zero vector of a lattice of dimension d, and finds a vector whose norm is inferior to $(\gamma_d + o(1))^{(n-1)/(d-1)}V^{1/n}$ for a lattice of volume V and dimension n, with $\gamma_d \leq 2B_d^{1/d}$ the square of the Hermite constant.

Heuristically, we use the BKZ algorithm which is stopped after a polynomial number of calls to the same oracle. The i-th vector of the orthonormalized basis has a length proportional to roughly γ^i and the Gaussian heuristic gives $B_d = \gamma^{d(d+1)/2}$ so that $\gamma \approx B_d^{2/d^2}$ and the first vector has a norm $\approx \gamma^{-n/2}V^{1/n} \approx B_d^{n/d^2}V^{1/n} \approx d^{n/2/d}V^{1/n}$.

---

[10] We can do it in logarithmic time using a binary balanced tree which contains in each node the size of the subtree.

The oracle takes time $2^{(c+o(1))d}$ with $c=2$ for a deterministic algorithm[41], $c=1$ for a randomized algorithm[2], $c=0.2972$ heuristically[29] and $c=0.286$ heuristically and quantumly[30].

B.1   Primal algorithm

Given $\mathbf{A}$ and $\mathbf{b}=\mathbf{A}\mathbf{s}+\mathbf{e}+q\mathbf{y}$, with the coordinates of $\mathbf{s}$ and $\mathbf{e}$ being Gaussians of standard deviation $q\alpha=o(q)$, we use the lattice generated by

$$\mathbf{B}=\begin{pmatrix}q\mathbf{I} & \mathbf{A}\\ \mathbf{0} & \mathbf{I}\end{pmatrix}$$

Now,

$$\mathbf{B}\begin{pmatrix}\mathbf{y}\\ \mathbf{s}\end{pmatrix}=\begin{pmatrix}\mathbf{b}\\ \mathbf{0}\end{pmatrix}+\begin{pmatrix}-\mathbf{e}\\ \mathbf{s}\end{pmatrix}.$$

We then heuristically assume that the i-th vector of the orthonormalized basis has a length at least $d^{-(n+m)/2/d}V^{1/(n+m)}$. In the orthogonalized basis, the error vector has also independent gaussian coordinates of standard deviation αq. Therefore, Babai algorithm will work with high probability if $d^{-(n+m)/2/d}V^{1/(n+m)}\geq \alpha q\sqrt{\log n}$.

Thus, we need to maximize $d^{-(n+m)^2/2/d}q^{m/(n+m)}$. So we use $m+n=\sqrt{2dn\log q/\log d}$ and the maximum is

$$2^{-\sqrt{n\log q\log d/2/d}}2^{\log q(1-\sqrt{n\log d/2/d/\log q})}=q2^{-\sqrt{2n\log q\log d/d}}.$$

If $\alpha=n^{-a}$ and $q=n^{b}$, we need $a\log n=\sqrt{2bn\log n\log d/d}$ and we deduce $d/n=2b/a^2+o(1)$, so that the running time is $2^{(2cb/a^2+o(1))n}$.

When we have a binary secret and error, since only the size of the vectors counts, it is as if $q\alpha=\mathcal{O}(1)$ and the complexity is therefore $2^{(2c/b+o(1))n}$. When the secret is binary and the error is a Gaussian, the problem is clearly even more difficult[11].

In dimension 1, this gives $\log 1/\alpha=\sqrt{2\log q\log d/d}$ so that $d=(2+o(1))\log q\log(\log q/\log^2\alpha)/\log^2\alpha$, which gives a polynomial time algorithm for $\log^2\alpha=\Omega(\log q\log\log\log q/\log\log q)$ and a subexponential one for $\log^2\alpha=\omega(\log\log q)$.

We can use the same algorithm for subset-sum with density d with the lattice defined in subsection 5.1 : applying the same heuristics gives a complexity of $2^{(c+o(1))nd\log(nd)}n^{\mathcal{O}(1)}$.

B.2   Dual algorithm

The attack consists in using lattice reduction to find sums of samples which are equal to zero on most coordinates, and then use FindSecret to find the secret on the remaining coordinates. It has been described for instance in [42].

More precisely, we select n truncated samples which form an invertible matrix $\mathbf{A}$ of size n; and m others which form the matrix $\mathbf{B}$ with m columns. We then search for a short vector $\mathbf{v}\neq\mathbf{0}$ of the lattice generated by $\begin{pmatrix}q\mathbf{I} & \mathbf{A}^{-1}\mathbf{B}\\ \mathbf{0} & \mathbf{I}\end{pmatrix}$.

We have $\mathbf{v}=\begin{pmatrix}\mathbf{x}\\ \mathbf{y}\end{pmatrix}$ and $\mathbf{A}(-\mathbf{x})+\mathbf{B}\mathbf{y}=\mathbf{A}(-\mathbf{x}+\mathbf{A}^{-1}\mathbf{B}\mathbf{y})=\mathbf{A}\mathbf{0}=\mathbf{0}\mod q$ so we deduce from $\mathbf{v}$ a small sum of samples which is equal to zero on n coordinates. We can then deduce a sample of low dimension with bias $e^{-||\mathbf{v}||^2\alpha^2}$. As $||\mathbf{v}||$ is inferior to $\approx q^{n/(n+m)}2^{(n+m)\log d/d/2}$, so we select $n+m=\sqrt{2nd\log q/\log d}$ and $||\mathbf{v}||^2\approx 2^{\sqrt{8n\log q\log d/d}}$.

Finally, using $\alpha=n^{-a}$, $q=n^{b}$ and $d=\Theta(n)$, we search d so that $2^{\sqrt{8n\log q\log d/d}}=\mathcal{O}(d/\alpha^2)$ so that $d/n=8b/(1+2a)^2+o(1)$.

---

[11] In this case, a standard technique is to scale the coordinates of the lattice corresponding to $\mathbf{e}$, in order to have each coordinate of the distance to the lattice of the same average size.

When the errors are binary, we can apply the same formula with $\alpha = \Theta(1/q)$ and the complexity is therefore $2^{(2cb/(1/2+b)^2+o(1))n}$.
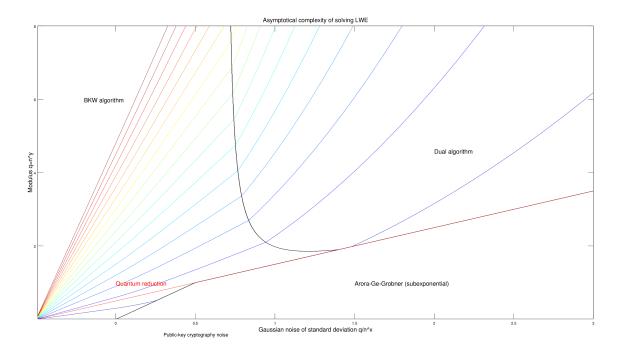
Observe that this algorithm is always asymptotically faster than the previous one, but requires an exponential number of samples. This disadvantage can be mitigated by finding many short vectors in the same lattice, which hopefully will not affect the behaviour of FINDSECRET. Another solution is to use a sample preserving search-to-decision reduction [39] so that the product of the time and the square of the success probability is the same.

In dimension 1, this gives $\log(d/\alpha^2) = \sqrt{8 \log q \log d/d}$ and we deduce

$$d = (8 + o(1)) \log q \log \left(\log q / \log^2(d/\alpha^2) \right) / \log^2(d/\alpha^2)$$

and this is a polynomial time algorithm for $\log^2 \alpha = \Omega(\log q \log \log \log \log q / \log \log q)$, but a subexponential one for any $\log(q)/\alpha^2 = \omega(2^{\sqrt{\log \log q}})$.
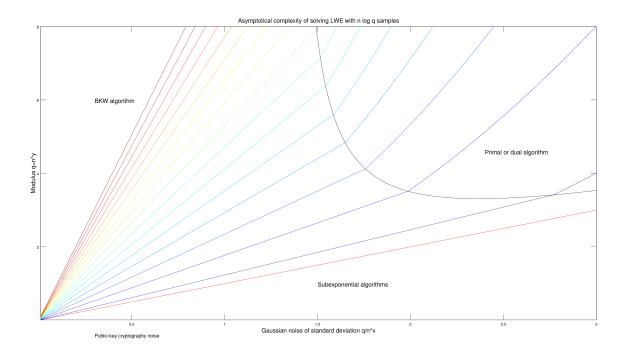
B.3 Graphical Comparison between lattice algorithms and BKW

The horizontal axis represents the standard deviation of the noise, with x such that $\alpha = n^x$. The vertical axis represents the modulus, with y such that $q = n^y$.

In the upper right quadrant delimited by the black line, parameters are such that the dual attack with heuristic classical SVP is more efficient than BKW. Below the black lines, Gröbner basis algorithms are sub-exponential. Above the red line, there is a quantum reduction with hard lattice problems [46].

The rainbow-colored lines are contour lines for the complexity of the best algorithm among BKW and the dual attack. Each curve follows the choice of parameters such that the overall complexity is $2^{(k+o(1))n}$, where k varies from 0 to 5 by increments of 0.3.

Asymptotical complexity of solving LWE with n log q samples

C LPN

Definition 13. *A* LPN *distribution is defined as a* LWE *distribution with $q = 2$. The error follows a Bernoulli distribution of parameter p.*

Theorem 18. *If we have $p = 1/2 - 2^{-o(n)}$ and $p \geq 1/n$, then we solve* Decision-LPN *in time*

$$2^{(n+o(n))/\log(n/\log(1-2p))}$$

with negligible probability of failure.

Proof. We use $d_i = \min(i\lfloor nx \rfloor, n)$, $m = n2^{k+nx}$ and select $k = \lfloor \log(\beta^2/\log \beta/3) \rfloor$ where $\beta^2 = n/2/\log(1 - 2p) = \omega(1)$. The algorithm works if $d_k = n$, so we chose $x = 1/k + 1/n = (1 + o(1))/\log(\beta^2)$. Finally, the bias of the samples given to DISTINGUISH is $2^k \alpha^2 \leq \beta^2/\log \beta/3n/2/\beta^2 = n/6/\log \beta \leq nx/3$ so that DISTINGUISH works with negligible probability of failure. $\square$

In particular, for $p = 1/2 - 2^{-n^{o(1)}}$, we have a complexity of $2^{(n+o(n))/\log n}$.

Theorem 19. *If we have $p \leq 1/4$, then we solve* Search-LPN *in time $\mathcal{O}(n^4(1-p)^{-n})$ with negligible probability of failure.*

Proof. We repeat $n(1-p)^{-n}$ times the following algorithm : apply the secret-error switching algorithm (Theorem 1) over $34n$ new samples, and run DISTINGUISH over the scalar part of the $33n$ samples. If the distinguisher outputs that it is a LPN distribution, then the secret is the zero vector with failure probability $2^{-33/32n}$, and we can recover the original secret from this data. If the new secret is not zero, then the distinguisher outputs that the distribution is uniform, except with probability $2^{-33/32n}$. The probability that no new secret is equal to zero is

$$\left(1 - (1-p)^n\right)^{n(1-p)^{-n}} \leq \exp(-(1-p)^n n(1-p)^{-n}) = 2^{-\Omega(n)}.$$

Using the union bound, the overall probability of failure is at most $n^4 2^n 2^{-33/32n} = 2^{-\Omega(n)}$. $\square$

In particular, for $p = \frac{1}{\sqrt{n}}$, the complexity is $\mathcal{O}(n^4 \exp(\sqrt{n}))$. In fact, for $p = (1 + o(1))/\ln(n) \leq 1/4$, we have a complexity which is $\mathcal{O}(n^4 \exp(n(p + p^2))) = 2^{(1+o(1))n/\log n}$, so that the threshold for p between the two algorithms is in $(1 + o(1))/\ln n$.