Faster Dual Lattice Attacks for Solving LWE

with applications to CRYSTALS

Qian Guo[1], Thomas Johansson[1]

Dept. of Electrical and Information Technology, Lund University, P.O. Box 118, 221 00 Lund, Sweden
{qian.guo,thomas.johansson}@eit.lth.se

Abstract. Cryptosystems based on the learning with errors (LWE) problem are assigned a security level that relates to the cost of generic algorithms for solving the LWE problem. This includes at least the so-called primal and dual lattice attacks. In this paper, we present an improvement of the dual lattice attack using an idea that can be traced back to work by Bleichenbacher. We present an improved distinguisher that in combination with a guessing step shows a reduction in the overall complexity for the dual attack on all schemes. Our second contribution is a new two-step lattice reduction strategy that allows the new dual lattice attack to exploit two recent techniques in lattice reduction algorithms, i.e., the "dimensions for free" trick and the trick of producing many short vectors in one sieving. Since the incompatibility of these two tricks was believed to be the main reason that dual attacks are less interesting, our new reduction strategy allows more efficient dual approaches than primal attacks, for important cryptographic parameter sets.

We apply the proposed attacks on CRYSTALS-Kyber and CRYSTALS-Dilithium, two of the finalists in the NIST post-quantum cryptography project and present new lower complexity numbers, both classically and quantumly in the core-SVP model. Most importantly, for the proposed security parameters, our new dual attack with refined lattice reduction strategy greatly improves the state-of-the-art primal attack in the classical gate-count metric, i.e., the classical Random Access Machine (RAM) model, indicating that some parameters are really on the edge for their claimed security level. Specifically, the improvement factor can be as large as 15 bits for Kyber1024 with an extrapolation model (Albrecht et al. at Eurocrypt 2019). Also, we show that Kyber768 could be solved with classical gate complexity below its claimed security level. Last, we apply the new attack to the proposed parameters in a draft version of Homomorphic Encryption Standard (see `https://homomorphicencryption.org`) and obtain significant gains. For instance, we could solve a parameter set aiming for 192-bit security in $2^{187.0}$ operations in the classical RAM model. Note that these parameters are deployed in well-known Fully Homomorphic Encryption libraries.

Keywords: Lattice-based cryptography, NIST post-quantum cryptography standardization, dual attacks, CRYSTALS, learning with errors, fast Fourier transform, fully homomorphic encryption.

1 Introduction

The LWE problem was introduced by Regev [51] and has quickly become one of the main problems in cryptography. One reason is the fear of future quantum computers being able to solve the factoring and discrete log problems efficiently. In the search for new future cryptographic schemes not based on the previous standard problems factoring and discrete log, the LWE problem has received a central role. One advantage for LWE is that this problem is claimed to be as hard as worst-case approximation problems in lattices, such as the shortest vector problem (SVP) [49,25]. Another reason for the importance of LWE is its usefulness in a variety of cryptographic constructions and primitives. This, in particular, includes Fully Homomorphic Encryption (FHE), which is a very important primitive that allows operations on encrypted data without decrypting it. The most efficient FHE schemes today are constructed using LWE or some version of the problem as the underlying difficult problem. Examples of such FHE schemes can be found in [27,36,26,2,4].

Returning to the post-quantum scenario, the need for new cryptographic primitives has been identified and in 2015 NIST started the project which we refer to as the NIST PQC standardization process [5]. The goal was to accept candidates for public-key encryption schemes (PKE), key encapsulation mechanisms (KEM), and digital signature schemes, and then to evaluate their security under the assumption that quantum computations can be done. In the end, a few proposals will be selected for possible standardization. The project has now entered the third round, where in the move to each round the number of candidates has been reduced. Many of the candidates in the project as a whole as well as among the remaining round 3 candidates, are based on some LWE-related problem. The round 3 candidates are split in two groups, being the main and the alternate ones.

Proposals are giving parameters in relation to target security categories. Among the 5 defined security categories, category 1,3 and 5 correspond to the complexity of exhaustive key search on AES with key size 128, 192, and 256, respectively. A proposal with parameters given for category 1 thus has to meet the requirement that any attack on the scheme requires a complexity larger than or comparable to the complexity of exhaustive key search on AES with key size 128.

Any cryptosystem based on the learning with errors (LWE) problem can be assigned a security level that corresponds to the lowest cost among any possible attack, which includes generic algorithms for solving the LWE problem. Possible algorithms include at least the so-called primal and dual lattice attacks. These attacks make use of the BKZ lattice reduction algorithm [31], which in turn uses as a subroutine a solver for SVP in projected sublattices (also referred to as blocks). Connected to both the primal and dual lattice attacks is the *cost* of performing them, which relates to the cost of running the BKZ algorithm. Due to the somewhat complicated nature of the BKZ algorithm, there has been several different cost models used in previous literature [11]. The cost model can either be an expression for the asymptotic behaviour of the cost of running BKZ, or

it can be an attempt to express the actual complexity in number of operations of some kind. As we are interested in the actual complexity, we use cost models for this latter case. Another distinguishing factor is the choice of the subroutine for solving SVP in projected sublattices inside BKZ, which can be either enumeration or lattice sieving [44,45]. Sieving gives the better performance but requires more memory. Established cost models for BKZ are used by designers to evaluate the cost of different attacks on their design which in turn gives an indication of the expected security level.

Briefly, we may describe LWE as the problem of recovering a secret vector $\mathbf{s} \in \mathbb{Z}_q^n$ after receiving $(\mathbf{A}, \mathbf{b})$ for which $\mathbf{b} = \mathbf{A}\mathbf{s} + \mathbf{e}$, where $\mathbf{A}$ is an $m \times n$ matrix with entries in $\mathbb{Z}_q$ and $\mathbf{e} \in \mathbb{Z}_q^m$. It is also assumed that both the noise vector $\mathbf{e}$ as well as the secret $\mathbf{s} \in \mathbb{Z}_q^n$ itself are *small*. It means that the entries are small in relation to $\mathbb{Z}_q = \{\frac{-(q-1)}{2}, \ldots, \frac{(q-1)}{2}\}$ (for q odd prime). In the dual attack, the idea is to find short vectors in the dual lattice defined as $\Lambda' = \{(\mathbf{x}, \mathbf{y}) \in \mathbb{Z}^m \times \mathbb{Z}^n : \mathbf{A}^T\mathbf{x} = \mathbf{y} \mod q\}$. For each short vector, denote it $(\mathbf{w}, \mathbf{v})$, we can observe that $\mathbf{w}^T \cdot \mathbf{b} = \mathbf{v}^T \cdot \mathbf{s} + \mathbf{w}^T \cdot \mathbf{e}$ is somewhat small. So with enough short vectors from Λ' we get a distinguisher that can separate whether $\mathbf{b}$ is from the LWE distribution or whether $\mathbf{b}$ is from a uniform distribution. The distinguisher is used in combination with a guessing step where a few entries of the secret $\mathbf{s}$ are guessed and corresponding positions are excluded from $\mathbf{A}$. If the guess is correct, $\mathbf{b}$ will come from the LWE distribution but if the guess is wrong then $\mathbf{b}$ will be from a uniform distribution (or close to). In this way the distinguisher will recover the guessed entries of $\mathbf{s}$ and eventually the full secret is recovered.

1.1 Contributions

In this paper, we present an improvement of the dual lattice attack using an idea that can be traced back to work by Bleichenbacher [23,22] on attacking on ECDSA. We present an improved distinguisher that in combination with a guessing step shows a reduction in the overall complexity for the dual attack on all schemes. This results in a strictly better dual attack than previous ones.

The main idea is to reduce the FFT distinguisher over a very large alphabetic size q to another distinguisher with a very small alphabetic size, say of only size 2 (or 3 for certain proposals where q is not a prime but a power of 2). We design a new mapping technique to map the secret points close to one point in a set of points equally dividing the cycle. Then one can apply the new FFT techniques to accelerate the guessing procedure in a combination of guessing some entries only modulo 2. From an implementation point of view, the transform step can be more efficiently implemented through a Fast Walsh-Hadamard transform (FWHT) instead of the standard FFT approach.

The complexity reduction depends on the attacked scheme. In particular, we apply the proposed dual attack on CRYSTALS-Kyber and CRYSTALS-Dilithium in the NIST post-quantum cryptography project and present new lower complexity numbers measured in the core-SVP model (see the second and the third columns in Table 1).

	Classical core-SVP		Refined classical attacks (gates)		
	Claim [52]	New	Claim [52]	New	NIST [6]
Kyber512	118	115	151	147	143
Kyber768	182	174	215	**205**	207
Kyber1024	256	243	287	272	272
	Claim [46]	New	Claim [46]	New	NIST [6]
Dilithium-II	123	122	159	154	146
Dilithium-III	182	179	217	210	207
Dilithium-V	252	246	285	274	272

Table 1: The complexity comparison on the security parameter sets of the round-3 CRYSTALS. Cost is given in $\log_2$ of operations.

We also investigate the complexity of the new attack in the classic gate-count metric, i.e., the Random Access Machine (RAM) model. This model is more interesting in the NIST Post-Quantum Cryptography Standardization Project because it is difficult to determine if the classical complexity of 2^{174} in the core-SVP model meets the security requirement for NIST-3 defined as 2^{205} classic gates. The official documents of round-3 Kyber and Dilithium set their security parameters by counting the classical gates of primal attacks. One main obstacle is to measure the classic cost in the RAM model of the Nearest Neighbor Search used in lattice sieving, which is addressed by Albrecht et al. in [14]. The designers of Kyber and Dilithium dismiss the dual attack because ".. First, most of those vectors are larger by a factor $\sqrt{4/3}$, secondly the trick of exploiting all those vectors is not compatible with the 'dimension for free' trick.." (cited from [46]).

We show in this paper that dual attacks could be more efficient in the classical gate-count metric even if most of short vectors obtained are larger by a factor $\sqrt{4/3}$. Our novel idea is a new two-step lattice reduction strategy that could exploit both the "dimension for free" (d4f) trick and the "exploiting many short vectors in one sieving" (msv) trick. Furthermore, since BKZ typically includes calling an SVP oracle for many times, we can sieve in the second step with a larger dimension to balance the costs of the two steps. From this perspective, we exploit the d4f trick twice and also produce an exponential number of short vectors. Similar to the official documents of CRYSTALS [52,46], we employ the analysis from [14] to evaluate the sieving cost in the classical RAM model.

The classical complexity comparisons in the gate-count metric for CRYSTALS-Kyber and CRYSTALS-Dilithium are shown in the last columns of Table 1. The gain is generally significant and could be as large as 15 bits for Kyber1024; some parameters, therefore, are really on the edge for their claimed security level. Last, we show that Kyber768 could be solved with complexity below its claimed security level in the gate-count metric.

Lastly, we show that the new dual attack with refined lattice reduction strategy could solve certain parameter sets in a draft version of the Homomorphic Encryption Standard [7] faster than the claimed security levels under the classical RAM model.

Remarks This algorithmic improvement has very wide applications in lattice-based cryptography—lattice-based proposals need to recheck their security parameter sets for the dual attack. It could lead to a security problem if the original security margin is small. On the other hand, the reported complexity numbers in the classical RAM model assume that the cost of one RAM query is constant. These complexity numbers will increase if a more realistic memory access cost model is taken into consideration. Further research on this is beyond the scope of the paper.

1.2 Related Works

There are a few different classes of algorithms for solving LWE problems, see e.g. [16]. The algebraic method of Arora-Ge [18] and its extension using Gröbner basis techniques [8] is a powerful method when applicable. The combinatorial approach called BKW [24] and its many extensions [10,41,38] is another approach that for some parameter choices can be the most efficient solver for LWE. However, in general both these methods require a larger number of samples than what is available in the cryptanalysis of LWE-based constructions of KEMs, signatures, or FHEs. So the security of such constructions is almost always derived by analyzing the cost of attacks based on lattice reduction. These attacks are either the primal attacks, where one finds the solution by solving a decoding problem in the lattice, or reduce it to solving unique SVP [44,45,13].

The second type of lattice attack is the dual lattice attacks [47]. The basic form of the attack builds a distinguisher from many short vectors in the dual lattice. However, by simply guessing a part of the secret this is turned into a recovery of the secret vector. An efficient guessing procedure can be achieved by use of the Fast Fourier Transform [33]. Various improvements can be achieved if the secret is small and sparse [9,32,28], which is often the case in constructions, in particular for FHE constructions.

To the best of our knowledge, Albrecht [9] firstly studied the problem of efficiently producing many short vectors in the dual lattice attacks. He proposed an amortization approach using re-randomization and lattice reductions with a smaller dimension, but his approach is more heuristic and has worse performance compared with our new two-step lattice reduction approach with sieving.

Independently of this work, the paper [35] was recently posted on eprint. This work also considers the dual attack but in our understanding it uses an idea of generating LWE instances with bigger noise that correspond to a fraction of the secret vector, a different approach to the ideas suggested in this paper. Our approach of reducing the FFT distinguisher over a very large alphabetic size q to another distinguisher with a very small alphabetic size have some similarity to

work by Bleichenbacher [22] on attacking on ECDSA. In [29] a similar but different reduction was used in connection with implementing the BKW algorithm.

Finally, these attacks can sometimes be used in the form of hybrid lattice reduction attacks as introduced in [40]. Such attacks combine a meet-in-the-middle approach and/or guessing with lattice reduction and this can sometimes be the best attack [28,42,53].

Notes. We found another independent work [21] on eprint (posted on Feb 12, 2021) studying dual attacks on round-3 lattice-based primitives in the core-SVP model. Also focusing on the core-SVP model, a first version of our paper was submitted to Eurocrypt 2021 (with deadline on Oct 8th, 2020). Similar to [35], the work [21] studies exhaustive guessing in the dual lattice attacks. We additionally propose a novel FFT distinuisher to further reduce the solving complexity. Our second main contribution, i.e., a new two-step lattice reduction algorithm allowing us to exploit the recent advances in lattice algorithms, and the corresponding complexity results in the classical RAM model are not discussed in [35,21].

1.3 Organization

The remaining of the paper is organized as follows. We first introduce some preliminaries in Section 2, and present the newly proposed FFT distinguisher in Section 3. We then apply this new distinguisher to improve the general dual lattice-reduction approach in Section 4, which is followed by its applications to CRYSTALS in the core-SVP model in Section 5. We then present the new two-step reduction idea and the refined classic attacks beyond the core-SVP estimation in Section 6. Its application to FHE parameters is shown in Section 7. The theory is validated by experimental verification in Section 8. We lastly conclude the paper in Section 9.

2 Preliminaries

We denote vectors in lower-case bold, e.g. $\mathbf{a}$, and matrices in upper-case bold, e.g. $\mathbf{A}$. All vectors are column vectors by default. We denote $\mathbf{a}^T$ (or $\mathbf{A}^T$) its transpose for a vector $\mathbf{a}$ (or matrix $\mathbf{A}$). The matrix $\mathbf{I}_n$ is an identity matrix with dimension $n \times n$. The inner product of two vectors $\mathbf{a}$ and $\mathbf{b}$ with the same dimension is denoted by $\langle \mathbf{a}, \mathbf{b} \rangle$. For a vector $\mathbf{a}$ with dimension n, we denote its i-th entry as a_i, for $0 \le i \le n-1$, and define its norm as

$$\|a\| = \sqrt{\sum_{i=0}^{n-1} a_i^2}.$$

For a complex number $x \in \mathbb{C}$, we denote $\mathfrak{Re}(x)$ its real part. Let θ_q be the q-th root of unity, i.e., the complex number $\exp(2\pi i_0/q)$, where $i_0^2 = -1$. We also write it as θ if there is no ambiguity.

2.1 LWE

The Learning with Errors problem is defined as follows.

Definition 1 ([51]). *Let n be a positive integer, q a prime, and let $\mathcal{X}$ be an error distribution. Fix $\mathbf{s}$ to be a secret vector in $\mathbb{Z}_q^n$, chosen according to a uniform distribution. Denote by $L_{\mathbf{s},\mathcal{X}}$ the probability distribution on $\mathbb{Z}_q^n \times \mathbb{Z}_q$ obtained by choosing $\mathbf{a} \in \mathbb{Z}_q^n$ uniformly at random, choosing an error $e \in \mathbb{Z}_q$ according to $\mathcal{X}$ and returning*

$$(\mathbf{a}, z) = (\mathbf{a}, \langle \mathbf{a}, \mathbf{s} \rangle + e)$$

in $\mathbb{Z}_q^n \times \mathbb{Z}_q$. The (search) LWE *problem is to find the secret vector $\mathbf{s}$ given a fixed number of samples from $L_{\mathbf{s},\mathcal{X}}$.*

The definition above gives the *search* LWE problem, and one could similarly define the *decision* LWE problem to distinguish between samples drawn from $L_{\mathbf{s},\mathcal{X}}$ and a uniform distribution on $\mathbb{Z}_q^n \times \mathbb{Z}_q$.

The error distribution $\mathcal{X}$ is usually selected as the discrete Gaussian distribution on $\mathbb{Z}_q$ with mean 0 and variance σ^2, obtained by assigning a probability proportional to $\exp(\frac{-x^2}{2\sigma^2})$ to each $x \in \mathbb{Z}$ and then accumulating the probability mass function over all integers in each residue class modulo q. The error distribution is also denoted as $\mathcal{X}_\sigma$. One useful heuristic assumption is that the sum of two independent random variables X_1 and X_2 drawn from $\mathcal{X}_{\sigma_1}$ and $\mathcal{X}_{\sigma_2}$ respectively is drawn form $\mathcal{X}_{\sqrt{\sigma_1^2+\sigma_2^2}}$.

It is proven in [25] that LWE with small secrets remains hard, so many cryptosystems base their security on these variants such as LWE with *binary* or *ternary* secrets.

2.2 Dual Lattice Attacks

A *lattice* $\mathcal{L}$ is a discrete subgroup of $\mathbb{R}^d$. Let the columns $\mathbf{b}_0, \ldots, \mathbf{b}_{d-1}$ be linearly independent, and then it is a basis of the lattice $\{\sum v_i \mathbf{b}_i | v_i \in \mathbb{Z}\}$. In lattices, a central hard problem is to find a non-zero shortest vector in this lattice, which is called the shortest vector problem (SVP).

In the dual attack, the aim is to find a short vector $(\mathbf{w}, \mathbf{v})$ in the dual lattice $\mathcal{L}' = \{(\mathbf{x}, \mathbf{y}) \in \mathbb{Z}^m \times \mathbb{Z}^n : \mathbf{A}^T\mathbf{x} = \mathbf{y} \mod q\}$. Thus, given a sequence of LWE instances $(\mathbf{A}, \mathbf{b})$ s.t., $\mathbf{b} = \mathbf{A}\mathbf{s} + \mathbf{e}$, we have that

$$\mathbf{w}^T \cdot \mathbf{b} = \mathbf{w}^T \cdot (\mathbf{A}\ \mathbf{I}) \begin{pmatrix} \mathbf{s} \\ \mathbf{e} \end{pmatrix} = (\mathbf{v}^T\ \mathbf{w}^T) \begin{pmatrix} \mathbf{s} \\ \mathbf{e} \end{pmatrix},$$

which is small and can be distinguished from the uniform. Therefore, the problem is transformed to finding a short column vector in the lattice

$$\mathbf{B} = \begin{pmatrix} \mathbf{I}_m & 0 \\ \mathbf{A}^T & q\mathbf{I}_n \end{pmatrix}$$

The efficiency of the dual lattice attacks highly depends on how short the found vectors are. We have the following lemma to measure the advantage of the distinguishing problem.

Lemma 1 ([44]). *Given an LWE instance characterized by n, q, σ and a vector* $\mathbf{h}$ *with length l such that $\mathbf{h}^T\mathbf{A} = 0 \pmod q$, the advantage of distinguishing $\langle \mathbf{h}, \mathbf{e} \rangle$ from random is close to*

$$\epsilon = \exp\left(-2\pi^2\tau^2\right),$$

where $\tau = \frac{l\sigma}{q}$.

It is also known from statistical theory that if $\mathcal{O}\left(1/\epsilon^2\right)$ independent such samples are available, the success probability for the distinguisher is close to 1.

2.3 Cost Model for BKZ

To achieve high-quality short vectors, we normally use a class of lattice reduction algorithms called BKZ, an iterative, block-wise algorithm for basis reduction. This algorithm solves an SVP problem with a small dimension β and is denoted $\mathrm{BKZ}_{\beta,d}$, where d is the dimension of the lattice. The time complexity of $\mathrm{BKZ}_{\beta,d}$ is denote $T(\mathrm{BKZ}_{\beta,d})$.

For a lattice $\mathcal{L}$, $\mathrm{BKZ}_{\beta,d}$ produces vectors with length

$$\|v\| = \delta_0^d \cdot vol(\mathcal{L})^{\frac{1}{d}}, \tag{1}$$

where $\delta_0 \approx \left(\frac{\beta}{2\pi e}(\pi\beta)^{\frac{1}{\beta}}\right)^{\frac{1}{2(\beta-1)}}$ (see [30] for details), and $vol(\mathcal{L})$ is defined as the volume of the lattice $\mathcal{L}$.

There are several models to estimate the time complexity of $\mathrm{BKZ}_{\beta,d}$, which are generally classified into two categories depending on the method to implement the SVP solver in the BKZ reduction. Here we mainly focus on the sieving approach, the most relevant one for choosing security parameters ([34,12]).

The first model we discuss is the core-SVP model, which was proposed in [17] and is then used in many candidates in the NIST Post-Quantum Cryptography Standardization Project, such as NewHope [50], CRYSTALS-Kyber [52], and CRYSTALS-Dilithium [46]. In the core-SVP model, the classic complexity of BKZ reduction $T(\mathrm{BKZ}_{\beta,d})$ can be estimated as $2^{0.292\beta}$ and the quantum complexity is $2^{0.265\beta}$. This simplified model is definitely useful and allows us to compare the security strength of different lattice-based candidates. However, this model is far from being accurate when considering the security requirement from NIST, which is defined by the gate complexity.

Another model is the gate-count metric, i.e., the cost in the Random Access Machine (RAM) model, which has been studied for the primal lattice attack in the round-3 versions of CRYSTALS-Kyber [52] and CRYSTALS-Dilithium [46]. They use the results from [14] as a black-box to estimate how many gates are required in one operation called '$\mathsf{AllPairSearch}$'. The overall sieving cost can then be estimated according to the current understanding on sieving algorithms [34,12].

2.4 The Classic FFT Distinguisher

We now assume for an LWE problem with reduced dimension t, i.e., we have a list of m LWE samples $(\mathbf{a}_j, b_j)$, where

$$b_j = \sum_{i=0}^{t-1} a_{i,j} s_i + e_j \mod q,$$

for $j = 1..m$.

The normal approach to use the FFT is to classify the samples by $\mathbf{a}_j$ and compute

$$f(\mathbf{a}_j) = \sum_{j_0 \in I(\mathbf{a}_j)} \theta^{b_{j_0}}, \tag{2}$$

where $I(\mathbf{a}_j)$ is the index set such that $\mathbf{a}_{j_0} = \mathbf{a}_j$ for $j_0 \in I(\mathbf{a}_j)$.

Then we compute

$$F(\tilde{\mathbf{s}}) = \sum_{\mathbf{a}_j} f(\mathbf{a}_j) \theta^{-\sum_{i=0}^{t-1} \mathbf{a}_{i,j} \tilde{s_i}}, \tag{3}$$

for all possible $\tilde{\mathbf{s}}$ by using the Fast Fourier Transformation (FFT), and return the guessed secret to be $\mathbf{s_0}$ s.t.,

$$\mathbf{s_0} = \underset{\tilde{\mathbf{s}}}{\operatorname{argmax}} \mathfrak{Re}(F(\tilde{\mathbf{s}})), \tag{4}$$

where $\mathfrak{Re}(F(\tilde{\mathbf{s}}))$ is the real part of $F(\tilde{\mathbf{s}})$.

For the right guess, the computed $F(\mathbf{s})$ is exactly

$$\sum_{j=1}^{m} \theta^{e_j},$$

having a large real part since e_j is sampled from a discrete gaussian distribution.

For a wrong guess, the value

$$\mathfrak{Re}(\sum_{j=1}^{m} \theta^{e'_j}) \to 0,$$

since e'_j is uniformly distributed over $\mathbb{Z}_q$.

Note that the FFT distinguisher has performance close to the optimal distinguisher (see [39]). With the distinguishing advantage ϵ defined in Lemma 1, we could bound the required number of samples by

$$\mathcal{O}\left(\frac{\ln(q^t)}{\epsilon^2}\right),$$

since we need to statistically determine the secret from q^t hypotheses. Similar formulas without the asymptotic notation can be obtained via Hoeffding's bound in [33].

The complexity for the Fast Fourier Transform with size t is $\mathcal{O}\left(q^t \cdot t \cdot \log_2(q)\right)$. This complexity quickly becomes prohibitively high since in lattice-based schemes when increasing the FFT size, as the parameter q is typically chosen as a large integer. Thus, the length of the partial secret vector that can be guessed via the FFT is rather small, highly limiting the gain of applying the FFT technique.

3 A New FFT Distinguisher

In this section, we describe a new distinguisher with the FFT technique, where the underlying idea is similar to that of Bleichenbacher's attacks on ECDSA [23]. We pick an integer γ much smaller than q and attempt to recover one secret entry $(s_i \mod \gamma)$ rather than the exact value of s_i. Thus, the complexity of the FFT with dimension t is reduced from $\mathcal{O}\left(q^t \cdot t \cdot \log_2(q)\right)$ to $\mathcal{O}\left(\gamma^t \cdot t \cdot \log_2(\gamma)\right)$, thereby allowing us to reach a much larger dimension when a certain computational resource is assumed.

If γ is chosen to be 2, then the employed Fast Fourier Transform is actually a Fast Walsh-Hadamard Transform over the complex field. For simplicity, we use the term Fast Fourier Transform (FFT) throughout the paper.

3.1 New Transformation Technique

Let γ be a small element in the ring $\mathbb{Z}_q$ such that $\gamma \cdot \rho = \pm 1 \mod q$, for some element ρ. So γ^{-1} is well-defined, i.e., being ρ or $-\rho$. To be more specific, the field size q is typically chosen as a prime or a power-of-two integer. When q is a prime, we could pick $\gamma = 2$; for the latter case we pick $\gamma = 3$. Now we take the q prime case as an instance to show how this distinguisher works.

We can rewrite the LWE samples as $(\hat{\mathbf{a}}_j, b_j)$ such that,

$$b_j = \sum_{i=0}^{t-1} \hat{a}_{i,j}\hat{s}_i + e_j \mod q,$$

where $\hat{\mathbf{s}} = \gamma^{-1}\mathbf{s} \mod q$ and $\hat{\mathbf{a}}_j = \gamma \mathbf{a}_j \mod q$. Note that we assume $\gamma = 2$.

We then write the equations in the real set $\mathbb{R}$, i.e.,

$$b_j = \sum_{i=0}^{t-1} \hat{a}_{i,j}\hat{s}_i + e_j + l_j \cdot q,$$

for each LWE sample. We could then apply some reduction techniques such as lattice reduction algorithms to make $\hat{a}_{i,j}$ small. We have that

$$\begin{aligned} b_j &= \sum_{i=0}^{t-1} \hat{a}_{i,j}(q+1)/2 \cdot s_i + e_j + l_j \cdot q \\ &= \sum_{i=0}^{t-1} \hat{a}_{i,j} \cdot q/2 \cdot s_i + \sum_{i=0}^{t-1} \hat{a}_{i,j}/2 \cdot s_i + e_j + l_j \cdot q. \end{aligned} \tag{5}$$

Let us compute

$$F(\mathbf{s} \mod 2) = \sum_{j=0}^{m-1} \theta^{b_j - \sum_{i=0}^{t-1} \hat{a}_{i,j} \cdot q/2 \cdot s_i} = \sum_{j=0}^{m-1} \theta^{b_j} \cdot \exp(-\sum_{i=0}^{t-1} \hat{a}_{i,j} \cdot s_i \cdot 2\pi i_0/2).$$

Here we use the notation $F(\mathbf{s} \mod 2)$ to define the above computation, so the function $F(\cdot)$ is different from the one used in Section 2.4.

For the right guess, from Equation (5), the computed value is

$$\sum_{j=0}^{m-1} \theta^{\sum_{i=0}^{t-1} \hat{a}_{i,j}/2 \cdot s_i + e_j} = \sum_{j=0}^{m-1} \exp\left(2\pi i_0/q \cdot (\frac{1}{2} \cdot \sum_{i=0}^{t-1} \hat{a}_{i,j} \cdot s_i + e_j)\right), \quad (6)$$

which is biased if $\hat{a}_{i,j}$ is small. The reason is that the standard deviations of the random variables s_i and e_j are small. Otherwise, the computed value is close to 0 (see Figure 1 for a graphical illustration). Note that the noise for the t positions (for $0 \leq i \leq t-1$) involved in the Fast Fourier Transform, i.e. $\sum_{i=0}^{t-1} \hat{a}_{i,j} \cdot s_i$ is reduced (see Equation (6)).

We next show a smart approach to perform the computation for all possible guesses using the FFT technique. Since $(-1)^2 = 1$, we could further classify θ^{b_j} into 2^t groups according to the vector $\mathbf{c} = (\hat{\mathbf{a}}_j \pmod 2)$ and define

$$f(\mathbf{c}) = \sum_{j_0 \in I(\mathbf{c})} \theta^{b_{j_0}}.$$

Here $I(\mathbf{c})$ is the index set such that $\hat{\mathbf{a}}_{j_0} \pmod 2$ is equal to $\mathbf{c}$ for $j_0 \in I(\mathbf{c})$.

We then have the following equation

$$F(\mathbf{s} \mod 2) = \sum_{\mathbf{c}} f(\mathbf{c})(-1)^{-\langle \mathbf{c}, \mathbf{s} \rangle}. \quad (7)$$

We exhaustively guess all the binary vector $\tilde{\mathbf{s}} \in \mathbb{Z}_2^t$ and compute the corresponding $F(\tilde{\mathbf{s}})$. This procedure can be done in $O(m + t \cdot 2^t)$ via using the Fast Fourier Transform. The guessed vector is a binary vector $\mathbf{s_0} \in \mathbb{Z}_2^t$ s.t.,

$$\mathbf{s_0} = \underset{\tilde{\mathbf{s}} \in \mathbb{Z}_2^t}{\operatorname{argmax}} \mathfrak{Re}(F(\tilde{\mathbf{s}})), \quad (8)$$

where $\mathfrak{Re}(F(\tilde{\mathbf{s}}))$ is the real part of $F(\tilde{\mathbf{s}})$. With sufficient samples, the guessed vector should be $(\mathbf{s} \mod 2)$.

Up to this point, the attacker has recovered t bits of the secret information, which is the most difficult part. If we write $\mathbf{s} = 2 \cdot \mathbf{s}' + \mathbf{s}_0$ and recover the value of $\mathbf{s}_0$, then the norm of $\mathbf{s}'$ is smaller by a factor of almost 2 compared to $\mathbf{s}$. Let $b_j = \langle \mathbf{a}_j, \mathbf{s} \rangle + e_j \mod q$, and we rewrite it as

$$b_j - \langle \mathbf{a}_j, \mathbf{s}_0 \rangle = \langle 2\mathbf{a}_j, \mathbf{s}' \rangle + e_j \mod q.$$

Thus, we have a new LWE problem with secret $\mathbf{s}'$. Since $\|\mathbf{s}'\|$ is much smaller if we recover a sufficient number of bits, which is true for the parameters discussed

later, the cost of recovering the remaining secret by iteratively calling the dual approach is negligible.

The gain. We present a simple example to discuss the pros and cons when comparing the new FFT distinguisher with the classic FFT distinguisher. Let q be a prime of size about 2^{12} and assume that the complexity constraint only allows to perform the classic FFT distinguisher with dimension 2. Thus, two positions are zeroed-out by this distinguisher. Applying the new FFT distinguisher with $\gamma = 2$, we could instead reduce 24 positions, but a certain amount of noise remains in each reduced position.

Another small (or practical) gain is that the new FFT distinguisher allows more flexible parameter selections to meet the time complexity constraint. For the classical FFT distinguisher, the complexity increases by a factor of about q if the FFT size is increased by one, which is much larger than the factor, i.e., γ, increased for the new FFT distinguisher.

3.2 The Distinguishing Property

We show the visual explanation of the distinguishing property in Figure 1. The FFT distinguisher computes the value

$$\mathfrak{Re}(\sum_{j=1}^{m} \theta_{2q}^{X_j}),$$

where $X_j = Y_j + E_j$ and E_j is drawn from a discrete Gaussian $\mathcal{X}_{2\sigma}$ over $\mathbb{Z}_{2q}$. The random variable $Y_j = \lambda \cdot q$ is 0 for the right guess. For the wrong guess, the variable λ is uniformly distributed over $\mathbb{Z}_2$ and the FFT distinguisher computes $\mathfrak{Re}(\sum_{j=1}^{m} \theta_{2q}^{X_j}) \to 0$, due to the symmetry. We verified numerically that this is true for the relevant parameters in this paper since $\hat{a}_{i,j}$ drawn from a reasonably small discrete Gaussian still ensures that $\hat{a}_{i,j} \pmod 2$ is very close to the uniform.

Both the new distinguisher and the classic FFT distinguisher are estimating $\frac{1}{m}\sum_{j=1}^{m} \cos(\theta^{X_j})$, with $\mathbb{E}[\cos(\theta^{X_j})]$ having the same value away from 0 for the right guess and $\mathbb{E}[\cos(\theta^{X_j})] = 0$ for the wrong guess. Hoeffding's bound could then be applied, implying that the data complexity of the new distinguisher can be estimated in a similar manner to the classic FFT distinguisher.

4 Improving the Dual Lattice-Reduction Approach

The new distinguisher is now put into a framework of a dual attack to present a full LWE solving algorithm. The general steps of the new algorithm are described in Algorithm 1.

(a) The right guess case. (b) The wrong guess case.

Fig. 1: Graphical representation of the new distinguishing property ($\gamma = 2$).

Algorithm 1 New dual algorithm for solving LWE.

Input: The m LWE samples.

Output: A partial secret vector.

1: Map the entries in the matrix $\mathbf{A}$ as described in Equation (9).
2: Find sufficiently many short vectors in the lattice $\mathcal{L}$ via lattice reductions, where $\mathcal{L}$ is the lattice formed in Equation (10).
3: Guess the last t_1 positions of $\mathbf{s}$ exhaustively.
4: Use the new FFT procedure to guess the last t unknown entries in $\mathbf{s} \mod \gamma$.

Assume that the secret variables are distributed as a discrete gaussian distribution with standard deviation σ and the noise variables is distributed as a discrete gaussian distribution with standard deviation $c \cdot \sigma$. The general idea is that we assume for $t_a = t + t_1$ positions to be determined partially or fully in one run of the algorithm. Once the secret is partially determined, the problem of recovering the remaining positions of the secret is of much lower complexity and hence this part is discarded in the analysis [1].

We exhaustively guess the last t_1 positions in the secret. This may become beneficial if e.g. the secret variables take values in a very small alphabet. We write

$$\mathbf{A} = \left(\mathbf{A}_0 \ \hat{\mathbf{A}}_1 \ \mathbf{A}_2\right),$$

where $\hat{\mathbf{A}}_1$ is an $m \times t$ matrix, and $\mathbf{A}_2$ an $m \times t_1$ matrix, respectively. Here $\mathbf{A}_0$ is the matrix that corresponds to the remaining positions that are not directly affected by our procedure.

We perform the transformation and obtain

$$\mathbf{A}_1 = \gamma \hat{\mathbf{A}}_1 \mod q, \tag{9}$$

[1] For all parameter choices used in this paper (where t_1 and t are somewhat large), the statement is true as knowledge of t_1 entries and t bits then reduces the difficulty of the remaining problem considerably. For example, considering the parameters for solving Kyber768 in the classical RAM model (see Table 5), the cost of solving the remaining problem can be bounded by 2^{188}, which is negligible compared to the main cost of 2^{205}.

so we have a new matrix of $\begin{pmatrix}\mathbf{A}_0 & \mathbf{A}_1 & \mathbf{A}_2\end{pmatrix}$. The contribution from the t_1 positions that are exhaustively guessed, corresponding to the $\mathbf{A}_2$, is just computed and subtracted from $\mathbf{b}$ and can thus be removed.

According to the analysis in the previous section, if we use the FFT to guess the t values that are secret $s_i \pmod{\gamma}$, then the noise from these t positions are reduced by a factor of γ. We can thus search for a short vector $(\mathbf{w}, \mathbf{v}_0, \mathbf{v}_1)$ in the lattice $\mathcal{L}$ constructed as

$$\{(c\mathbf{x}, \mathbf{y}_0, \mathbf{y}_1/\gamma) \in c\mathbb{Z}^m \times \mathbb{Z}^{n-t_a} \times \frac{1}{\gamma}\mathbb{Z}^t : (\mathbf{A}_0 \ \mathbf{A}_1)^T\mathbf{x} = \begin{pmatrix}\mathbf{y}_0\\ \mathbf{y}_1\end{pmatrix} \mod q\} \quad (10)$$

to balance the noise from each position. The lattice has dimension $d = m+n-t_1$ and volume $c^m \cdot \frac{q^{n-t_1}}{\gamma^t}$ with high probability. This scaling trick is similar to [19].

If we compute $(\mathbf{w}/c)^T \cdot \mathbf{b}$, then the final noise after partial guessing and FFT is formed as

$$e = (\mathbf{w}/c)^T(\mathbf{A}_0 \ \mathbf{A}_1)\begin{pmatrix}\mathbf{s}_0\\ \frac{1}{\gamma}\mathbf{s}_1\end{pmatrix} + \langle \mathbf{w}/c, \mathbf{e}\rangle = \langle \mathbf{w}/c, \mathbf{e}\rangle + \langle \mathbf{v}_0, \mathbf{s}_0\rangle + \langle \gamma \cdot \mathbf{v}_1, \mathbf{s}_1\rangle \cdot \frac{1}{\gamma}.$$

Assume that the norm of the short vector $(\mathbf{w}, \mathbf{v}_0, \mathbf{v}_1)$ is l. The noise size is estimated as $\sigma \cdot l$, since the standard deviation of each entry in $\mathbf{e}$ ($\mathbf{s}$) is $c\sigma$ (σ).

For a BKZ reduction $\mathrm{BKZ}_{\beta,d}$, the shortest vector produced is expected to be of size $l = \delta_0^d \cdot (\frac{c^m \cdot q^{n-t_1}}{\gamma^t})^{\frac{1}{d}}$, where $\delta_0 \approx \left(\frac{\beta}{2\pi e}(\pi\beta)^{\frac{1}{\beta}}\right)^{\frac{1}{2(\beta-1)}}$. For the decision-LWE problem, the advantage is estimated as

$$\epsilon = \exp(-2\pi^2\tau^2),$$

where $\tau = \frac{\sigma \cdot l}{q}$.

Note that this is a general setting and for the schemes studied in this paper, i.e. CRYSTALS-Kyber and CRYSTALS-Dilithium, the constant c is always set to be 1.

4.1 Complexity Analysis

We present the complexity analysis of the new algorithm.

Bounded secret distribution. In many lattice-based primitives, the secret vector entries are chosen from an bounded alphabet of size B. For instance, the value B is 3 if the secret is ternary. We also assume that a lattice reduction algorithm could produce many (say $N(\beta)$) short vectors simultaneously with length $l = c_s \cdot \delta_0^d \cdot (\frac{c^m \cdot q^{n-t_1}}{\gamma^t})^{\frac{1}{d}}$, where c_s is a small constant. If $c_s = 1$, we assume that all the short vectors are as short as the shortest vector obtained from a BKZ reduction, which is definitely optimistic. Let the required number of samples for successful distinguishing be

$$N \geq \frac{c_0 \cdot \ln(\gamma^t \cdot B^{t_1})}{\epsilon^2}, \quad (11)$$

where c_0 is a constant factor[2] chosen as 4 and the factor $\ln(\gamma^t \cdot B^{t_1})$ comes from the fact that the FFT distinguisher finds the secret among $\gamma^t \cdot B^{t_1}$ hypotheses. To count the overall complexity of the new algorithm, we accumulate the complexity of different steps.

- The first step is a mapping for a small matrix with negligible cost.
- The second step involves $\max(1, \frac{N}{N(\mathsf{RED})})$ lattice reduction steps to produce sufficiently many short vectors, where $N(\mathsf{RED})$ denotes the number of short vectors produced via one lattice reduction. Thus, the complexity is $T(\mathsf{RED}) \cdot \max(1, \frac{N}{N(\mathsf{RED})})$, where $T(\mathsf{RED})$ denotes the complexity for one reduction procedure.
- We then guess t_1 positions with B^{t_1} possibilities in total. For each guess, the inner product of the guessed partial secret key and the corresponding coefficient vector needs to be subtracted for N samples outputted from the previous lattice reduction procedure and a large FFT transform with size t needs to be performed. For each FFT transform, the complexity is $t \cdot \gamma^t \cdot \log_2 \gamma$.

The exhaustive guessing approach could be done using a trick of storing intermediate values in memory. Now assuming that we need to compute $b - \langle \mathbf{a}, \mathbf{s} \rangle$, where $\mathbf{s}$ run through all the vectors of length t_1 and each entry in $\mathbf{s}$ has B choices. Notice that this computation needs to be done for N times since we have N short vectors from the previous lattice reductions.

We first build a table by computing $B \cdot t_1$ vectors of length N with entry $a_i s_i$ for $0 \leq i \leq t_1 - 1$ and s_i runs through all B choices. The cost $B \cdot t_1 \cdot N$ is much smaller than $B^{t_1} \cdot N$ for our targeted parameters; we, therefore, omit this cost in the complexity formula.

We could then enumerate all the possible $\mathbf{s}$ and build an enumeration tree of depth t_1. The starting point is an all-zero vector and the computation is trivial. The output $\mathbf{b}$ of the all-zero guess of $\mathbf{s}$ is placed in a leaf node and all the nodes in the path from the root to this all-zero leaf store the same vector $\mathbf{b}$ of length N. Afterwards, the computation of a new guess is only to add the vector stored in the parent node and a vector from a look-up table, which costs roughly N operations. Note that it is unnecessary to store all the enumeration tree, since only the vector in its parent node is needed to compute the vector in the new node. The memory cost of this enumeration is at most $O(t_1 \cdot N)$.

This technique is a general method used in different scenarios such as Information Set Decoding. As the complexity of the exhaustive guessing procedure can be bounded by the size of the guessing tree, the overall complexity is then estimated as

$$C = T(\mathsf{RED}) \cdot \max(1, \frac{N}{N(\mathsf{RED})}) + B^{t_1} \cdot (N + t \cdot \gamma^t \cdot \log_2 \gamma). \tag{12}$$

[2] For solving the Learning Parity with Noise (LPN) problem, this constant is chosen to be 4, which is verified in [37]. We adopt this setting and verify it via experiments in Sect. 8. Theoretical results [33,35] from Hoeffding's inequality bounds this value by roughly 8 multiplying some other terms related to the success probability.

Another optimization trick. One general optimization trick is to guess a fixed number of most probable choices in the alphabet and take into account the probability P_0 that the partial secret is one of the guessed vectors. Thus, in such an approach the overall complexity can be estimated as $\frac{C}{P_0}$.

In lattice research, we usually pick the secret pattern with bounded Euclidean distance. Now assuming the number of guessed patterns is $N(\mathsf{guess})$, we have the following theorem to bound the complexity of the new algorithm.

Theorem 1. *Let n, q, σ, c be the parameters for the LWE problem and m be the number of LWE samples used. Let t_1 be the guessing positions and t be the FFT size. Let the constants c_0, c_s and γ be as defined before. Assume that the lattice reduction algorithms include BKZ reductions $\mathrm{BKZ}_{\beta,d}$ to produce a reduced basis with good quality and one reduction procedure can produce $N(\mathsf{RED})$ short vectors with norm $l = c_s \cdot \delta_0^d \cdot (\frac{c^m \cdot q^{n-t_1}}{\gamma^t})^{\frac{1}{d}}$, $d = m+n-t_1$ and $\delta_0 \approx \left(\frac{\beta}{2\pi e}(\pi\beta)^{\frac{1}{\beta}}\right)^{\frac{1}{2(\beta-1)}}$. Let $T(\mathsf{RED})$ denote the complexity for one reduction procedure and N the required number of short vectors for the distinguisher. Let $N(\mathsf{guess})$ be the number of guessed patterns in the exhaustively guessing step.*

The time complexity of the new algorithm can be estimated as $\frac{C}{P_0}$, where P_0 is the probability that the partial secret is one of the guessed vector and,

$$C = T(\mathsf{RED}) \cdot \max(1, \frac{N}{N(\mathsf{RED})}) + N(\mathsf{guess}) \cdot (N + t \cdot \gamma^t \cdot \log_2 \gamma), \tag{13}$$

supposing that

$$N \geq \frac{c_0 \cdot \ln(\gamma^t \cdot N(\mathsf{guess}))}{\epsilon^2},$$

where $\epsilon = \exp(-2\pi^2\tau^2)$ and $\tau = \frac{\sigma \cdot l}{q}$.

Remarks. We describe a general formula for estimating the complexity of the new dual attack in Theorem 1. We mainly discuss two types of cost models in this paper, i.e., the core-SVP model and the classical RAM model (also called the gate-count model in the official documents of CRYSTALS). In different models, many functions, such as $T(\mathsf{RED})$, $N(\mathsf{RED})$, and l, need to be specified. For instance, a typical assumption in the core-SVP model is that a BKZ procedure $\mathrm{BKZ}_{\beta,d}$ could produce $2^{0.2075\beta}$ short vectors that are as short as the shortest one. In the RAM model, we use more realistic settings where $T(\mathsf{RED})$ and $N(\mathsf{RED})$ are studied in [14], and the produced short vectors are larger by a factor of $\sqrt{4/3}$ than the shortest one. Note that the latter assumption is suggested by theoretical analysis in [48], and is extensively verified in recent works [34,12,15].

5 Application to CRYSTALS

In this section we discuss the application of the algorithm to two of the seven finalists, i.e., CRYSTALS-Kyber [52] and CRYSTALS-Dilithium [46], in the NIST

Post-Quantum Cryptography Standardization Project, under the core-SVP estimation model. In the core-SVP model, the lattice reduction procedure is one BKZ reduction, denoted $\mathrm{BKZ}_{\beta,d}$, with time complexity $2^{0.292\beta}$ for a classic computer and $2^{0.265\beta}$ for a quantum computer. Such a reduction step is supposed to output $2^{0.2075\beta}$ short vectors with size as short as the shortest one. Thus, in this model $T(\mathsf{RED})$ is $2^{0.292\beta}$ for a classic computer and is $2^{0.265\beta}$ for a quantum computer, $N(\mathsf{RED}) = 2^{0.2075\beta}$, and $l = c_s \cdot \delta_0^d \cdot (\frac{c^m \cdot q^{n-t_1}}{\gamma^t})^{\frac{1}{d}}$ with $c_s = 1$.

These two cryptosystems are both from the "Cryptographic Suite for Algebraic Lattices" (CRYSTALS) [1], thus sharing similar designs. We fix γ to be 2 since in CRYSTALS the parameter q is always selected as an odd prime. We also know $c = 1$ since the secret distributions are the same as the noise ones.

The security of Kyber and Dilithium is related to solving LWE problems with different parameters. We numerically investigate the concrete complexity for solving the transformed LWE problems in the core-SVP model and show the estimation in Table 2 and 4.

	Kyber512	Kyber768	Kyber1024
Claimed security level	NIST-1	NIST-3	NIST-5
n	512	768	1024
q	3329	3329	3329
η	3	2	2
Classical core-SVP			
Claim [52]	118	182	256
Sect. 5	115	174	243
BKZ block-size β	394	595	829
Guessing size t_1	10	23	32
FFT size t	75	113	163
Quantum core-SVP			
Claim [52]	107	165	232
Sect. 5	105	160	223
BKZ block-size β	397	602	840
Guessing size t_1	7	15	21
FFT size t	72	117	163

Table 2: The complexity estimation on the security parameters of CRYSTALS-Kyber in the core-SVP model. Here n is the dimension when transforming the key-recovery problem to an LWE problem and q is the alphabetic size. Cost is given in $\log_2$ of operations. Here $\gamma = 2$.

Kyber. CRYSTALS-Kyber [52] is an IND-CCA2-secure KEM in the finalists of the NIST Post-Quantum Cryptography Standardization Project. We describe

the detailed parameter sets of Kyber in Table 2. The scheme fixes the alphabetic size to 3329. In the round-3 specification, each secret/noise entry is sampled from a centered binomial distribution B_η, where B_η is implemented as

1. Sample $(a_1, \ldots, a_\eta, b_1, \ldots, b_\eta) \leftarrow\!\!\$\ \{0,1\}^{2\eta}$;
2. Output $\sum_{i=1}^{\eta}(a_i - b_i)$.

For Kyber768 and Kyber1024, the secret and noise distributions are set to be B_2, while the distributions are B_3 for Kyber512. The distribution of B_2 is shown in Table 3.

	0	± 1	± 2
Probabilities	$\frac{3}{8}$	$\frac{1}{4}$	$\frac{1}{16}$

Table 3: The distribution of B_2.

We could have an efficient approach to guess a vector of dimension t_1 with entries sampled from B_2 for Kyber768 and Kyber1024. For such a vector, we numerically compute the distribution of the norm of the vector. We pick a bound R to ensure that the probability that the norm of the vector is smaller than R is larger than P_0. In our estimation, we fix P_0 to be 0.9 to reduce the cost of searching for the optimal parameter. We could then count $N(\mathsf{guess})$, the number of patterns that the norm is no larger than R. This optimization trick could offer a small gain of less than 1 bit for the targeted parameter sets.

We see from Table 2 that our new attack could have a gain in the core-SVP model as large as 13 bits classically and 9 bits with quantum computers, for Kyber1024. The gain is also significant for Kyber768.

Dilithium. CRYSTALS-Dilithium [46] is an EUF-CMA-secure digital signature algorithm in the finalists of the NIST Post-Quantum Cryptography Standardization Project. It has a large alphabetic size $q = 8380417$ and a larger dimension (than Kyber) for the same security level. For Dilithium-2 and Dilithium-5, the secret/noise distributions are set to be S_2, while they are S_4 for Dilithium-3. Here S_η is the uniform distribution over integers in $[-\eta, \eta]$. Under the core-SVP model, we describe in detail the attack complexity on parameter settings of Dilithium in Table 4. This table shows that we could improve the state-of-the-art attacks for all the three parameter sets, though the gain is smaller than that for Kyber.

6 Beyond Core-SVP Estimation

In the previous section, we have shown the improvement from the new dual attacks in the core-SVP model. However, it is unclear to compare these numbers

	Dilithium-2	Dilithium-3	Dilithium-5
Claimed security level	NIST-2	NIST-3	NIST-5
n	1024	1280	1792
q	8380417	8380417	8380417
η	2	4	2
Classical core-SVP			
Claim [46]	123	182	252
Sect. 5	122	179	246
BKZ block-size β	417	613	842
Guessing size t_1	13	15	29
FFT size t	75	116	163
Quantum core-SVP			
Claim [46]	112	165	229
Sect. 5	111	163	225
BKZ block-size β	419	616	848
Guessing size t_1	9	10	19
FFT size t	76	116	164

Table 4: The complexity of the new attack on the security parameters of CRYSTALS-Dilithium in the core-SVP model. The value n is the dimension of the transformed LWE problem, q the alphabetic size and η the parameter in the noise generation of S_η. Cost is given in $\log_2$ of operations. Here $\gamma = 2$.

with the security requirements from NIST. In the official documents of round-3 Kyber and Dilithium, the designers also presented security numbers in the gate-count metric. They, however, excluded the analysis against dual attacks since ".. First, most of those vectors are larger by a factor $\sqrt{4/3}$, secondly the trick of exploiting all those vectors is not compatible with the 'dimension for free' trick of [34].." (cited from [46]).

We in this section investigate the complexity of our new dual attacks in the gate-count metric and show that dual attacks could be more efficient even if most of short vectors obtained are larger by a factor $\sqrt{4/3}$. The novel idea is rather simple – we propose a new two-step lattice reduction algorithm where the first and second steps exploit the "dimension for free" (d4f) gain and the "many short vectors" (msv) gain, respectively. Also, a BKZ procedure typically includes calling an SVP oracle for many times. Thus, in the second step we could perform a sieving algorithm with a larger dimension to balance the costs of the two steps. From this perspective, we exploit the d4f trick twice and also produce an exponential number of short vectors.

6.1 A New Lattice Reduction Strategy

We describe the new two-step lattice reduction algorithm. The framework is shown in Algorithm 2. The first step is just a BKZ reduction where the d4f

Algorithm 2 Two-step Lattice Reduction

Input: A lattice.

Output: A list of short vectors.

1: Do BKZ reductions with size β. Then we obtain a reduced basis with a short vector $\mathbf{b}_0$ as the first vector in the basis.

2: For the lattice $\mathcal{L}'$ generated by the first β_0 vectors in the reduced basis, we perform a sieving step and get a list of $N(\beta_0)$ short vectors with size no larger than $\sqrt{4/3} \cdot \lambda_1(\mathcal{L}')$, where $\lambda_1(\mathcal{L}')$ is the shortest vector in the lattice $\mathcal{L}'$.

gain could be exploited, meaning that for a BKZ reduction $\mathrm{BKZ}_{\beta,d}$, the actual costs correspond to a smaller β'. We use this step to improve the quality of the reduced basis.

Exploiting the d4f gain. It is observed in [34] that the SVP in dimension β could be solved using a sieve in dimension $\beta' = \beta - d_{4f}$, where $d_{4f} = \Theta(\beta/\log\beta)$. Actually, this d4f gain comes from the fact that one sieving procedure could produce many short vectors. In [34], an "optimistic" estimation for d_{4f} is given as

$$d_{4f} = \frac{\beta \log(4/3)}{\log(\beta/(2\pi e))}. \tag{14}$$

This estimation is asymptotic and denoted by the *Asymptotic Model*. However, it is shown in [12] that the G6K sieve framework can achieve a larger dimension for free via a technique called "on the fly" lifting. By extrapolating from experimental data, they set d_{4f} as

$$d_{4f} = 11.46 + 0.0757 \cdot \beta. \tag{15}$$

We denote the latter extrapolated estimation the *G6K Model*.

Exploiting the msv gain. The second step is just one sieving procedure on the lattice $\mathcal{L}'$ generated by the first β_0 vectors in the reduced basis outputted by the previous step. We could then get a list of $N(\beta_0)$ short vectors with size no larger than $\sqrt{4/3} \cdot \lambda_1(\mathcal{L}')$, where $\lambda_1(\mathcal{L}')$ is the shortest vector in the lattice $\mathcal{L}'$. One important problem is thus to estimate the value of $\lambda_1(\mathcal{L}')$.

As we already know a short vector $\mathbf{b}_0$ in the lattice $\mathcal{L}'$, we could use $\|\mathbf{b}_0\|$ to upper-bound the value of $\lambda_1(\mathcal{L}')$. One could also use Gaussian Heuristics to estimate the value of $\lambda_1(\mathcal{L}')$. Note that the two approaches lead to quite close complexity numbers (see Section 6.3 for details). The number of short vectors produced is denoted $N(\beta_0)$, where $N(\beta_0) = 1/\mathsf{Caps}(\beta_0, \pi/3)$ and $\mathsf{Caps}(\beta_0, \pi/3)$ is the probability that a vector randomly drawn from the unit sphere of dimension $(\beta_0 - 1)$ has angle at most $\pi/3$ with some fixed vector. The number $N(\beta_0)$ can be concretely estimated from the source code in the appendix of [14].

6.2 Complexity Analysis

We analyze the complexity of the algorithm in the gate-metric count model. Let the lattice dimension be $d = m + n - t_1$. Theorem 1 could also apply since the structure of the dual algorithm is unchanged, but the terms $T(\mathsf{RED})$ and $N(\mathsf{RED})$ and the length of the short vectors need to be updated.

We use a similar approach to that in round-3 Kyber [52] and Dilithium [46] for analyzing the cost of sieving and BKZ in the gate-metric count. To be more specific, we employ the analysis in [14] of the gate count of a '$\mathsf{AllPairSearch}$' operation for different sieving dimensions. We build a table with table entry $GT(\beta)$ storing this cost in gate count metric for dimension β.

Also, similar to [52], we assume that the '$\mathsf{AllPairSearch}$' operation needs to be called only once using progressive sieving [43,34], and define the *progressivity overhead* $c_{\mathsf{po}} = 1/(1 - 2^{-0.292}) = 5.46$, i.e., the limit of ratio between $\sum_{i \leq b} 2^{0.292i+o(i)}$ and $2^{0.292b+o(b)}$ as b grows. We estimate $T(\mathsf{RED})$ as follows.

- For the first BKZ size β, we compute the sieving dimension $\beta' = \beta - d_{4f}$ and could check the GT table to have the complexity $GT(\beta')$. Similar to the analysis in [52], the complexity for BKZ is $(d - \beta)c_{\mathsf{po}}^2 GT(\beta')$.
- For the second step of the reduction, as the basis has been well-reduced by the first BKZ reduction and the d4f gain is no longer achieved, we do progressive sieving and the sieving complexity is estimated as $c_{\mathsf{po}} \cdot GT(\beta_0)$ for a sieving dimension β_0. So we set $(d-\beta)c_{\mathsf{po}} \cdot GT(\beta') \approx GT(\beta_0)$ to balance the cost, and produce $N(\beta_0)$ short vectors. We could achieve a slightly larger dimension of β_0 than β'. Also, the term $N(\mathsf{RED})$ is equal to $N(\beta_0)$, estimated with the concrete analysis from [14].

Thus, one new two-step reduction algorithm will cost

$$T(\mathsf{RED}) = (d - \beta)c_{\mathsf{po}}^2 \cdot GT(\beta') + c_{\mathsf{po}} \cdot GT(\beta_0).$$

The short vectors are as short as $\sqrt{4/3} \cdot \lambda_1(\mathcal{L}')$, where $\lambda_1(\mathcal{L}')$ can be estimated using the Gaussian Heuristic or be upper-bounded by

$$\|\mathbf{b}_0\| = \delta_0^d \cdot (\frac{c^m \cdot q^{n-t_1}}{\gamma^t})^{\frac{1}{d}}.$$

We have $\delta_0 \approx \left(\frac{\beta}{2\pi e}(\pi\beta)^{\frac{1}{\beta}}\right)^{\frac{1}{2(\beta-1)}}$. For Kyber and Dilithium, the constant c is always 1 and γ is 2 since the secret and noise distributions are the same and q is a prime.

When analyzing the primal attacks in the official documents of CRYSTALS, the designers use BKZ simulators to replace the simple geometric-series assumption, mainly due to its inaccuracy caused by the "tail" phenomenon. The situation is different in our dual lattice attack where the "head" phenomenon of BKZ reduction is the most important. Bai et al. in [20] stated that "Our simulator, which accurately predicts the head phenomenon, suggests that the head phenomenon

vanishes when the block-size becomes large .. Quantitatively, the phenomenon has almost fully disappeared for $\beta \approx 200$".

The focus of the paper is to assess the strength of the security parameters proposed in various cryptographic primitives with the BKZ block-size $\beta \gg 200$. Thus, the geometric-series assumption is accurate, and we stick with it mainly due to its simplicity. It could be more accurate to instead use BKZ simulators when discussing the complexity of solving smaller LWE instances (with the BKZ block-size $\beta \ll 200$).

	Kyber512	Kyber768	Kyber1024
Claimed security level	NIST-1	NIST-3	NIST-5
n	512	768	1024
q	3329	3329	3329
η	3	2	2
Claim [52]	151.5	215.1	287.3
Sect. 6			
Asymptotic Model	148.3	207.3	275.4
BKZ block-size β	398	604	848
BKZ sieving dimension $\beta' = \beta - d_{4f}$	361	555	785
Second sieving dimension β_0	400	596	827
Guessing size t_1	20	36	45
FFT size t	78	118	166
G6K Model [12]	147.1	**205.2**	272.3
BKZ block-size β	399	606	850
BKZ sieving dimension $\beta' = \beta - d_{4f}$	357	548	774
Second sieving dimension β_0	396	589	816
Guessing size t_1	20	36	45
FFT size t	77	116	164
Required by NIST	143	207	272

Table 5: The gate complexity comparison on the security parameters of CRYSTALS-Kyber. Here n is the dimension when transforming the key-recovery problem to an LWE problem and q is the alphabetic size. Cost is given in $\log_2$ of operations. Here $\gamma = 2$.

6.3 Results

We show in Table 5 and 6 the estimated complexity in the classical RAM model (also called gate-count metric in the official documents of CRYSTALS) for the security parameter sets of round-3 Kyber and Dilithium. The gain compared with the primal lattice attack is generally significant, ranging from 4 bits to 15 bits in the G6K Model (and from 3 bits to 12 bits in the Asymptotic Model),

	Dilithium-2	Dilithium-3	Dilithium-5
Claimed security level	NIST-2	NIST-3	NIST-5
n	1024	1280	1792
q	8380417	8380417	8380417
η	2	4	2
Claim [46]	158.6	216.7	285.4
Sect. 6			
Asymptotic Model	155.4	212.9	278.1
BKZ block-size β	418	620	853
BKZ sieving dimension $\beta' = \beta - d_{4f}$	380	570	790
Second sieving dimension β_0	424	616	837
Guessing size t_1	25	24	41
FFT size t	81	126	167
G6K Model [12]	153.8	210.4	274.4
BKZ block-size β	418	621	854
BKZ sieving dimension $\beta' = \beta - d_{4f}$	374	562	774
Second sieving dimension β_0	418	608	824
Guessing size t_1	26	24	41
FFT size t	80	120	165
Required by NIST	146	207	272

Table 6: The gate complexity of the new attack on the security parameters of round-3 CRYSTALS-Dilithium. The value n is the dimension of the transformed LWE problem, q the alphabetic size and η the parameter in the noise generation of S_η. Cost is given in $\log_2$ of operations. Here $\gamma = 2$.

and the gain in the gate-count metric is larger than that in the core-SVP model, since in the prior model we could have a larger guessing size and also a larger FFT size.

These two tables show that some parameter sets such as Kyber512, Dilithium-2 and Dilithium-3 have a rather limited security margin, some such as Kyber1024 and Dilithium-5 are really on the edge, and the parameter set Kyber768 fails[3] to achieve the security requirement from NIST. In these tables, we use $\|\mathbf{b}_0\|$ to upper-bound the value of $\lambda_1(\mathcal{L}')$. We also employ the Gaussian Heuristics to estimate $\lambda_1(\mathcal{L}')$ and obtain similar complexity numbers. For instance, the complexity of solving Kyber768 increases from 205 bits to 206 bits, but is still below its claimed security level.

[3] One may argue that the extrapolated G6K Model could be optimistic when the dimension is large. As the $\log_2$ of the gate count in the Asymptotic Model is so close to the NIST requirement (207.3 v.s. 207) for Kyber768, however, a small number of extra dimensions for free could make the scheme insufficient for its claimed security level.

Security Level	n	$\log_2(q)$	uSVP	dec (from [7])	dual	New Dual (RAM model)
128	1024	27	131.6	160.2	138.7	131.6
192	1024	19	193.0	259.5	207.7	187.0
256	1024	14	265.6	406.4	293.8	251.1

Table 7: The complexity comparison for the security parameters in the Homomorphic Encryption Standardization draft aiming for classic security. Here n is the dimension when transforming the key-recovery problem to an LWE problem and q is the alphabetic size. Cost is given in $\log_2$ of operations. The secret distribution is a uniform distribution from $\{-1, 0, 1\}$. The columns of uSVP, dec, and dual represent the complexity of the methods of uSVP, decoding, and dual, respectively, stated in the official documents of the Homomorphic Encryption Standard [7]. Here we pick $\gamma = 3$.

This new method can be partially understood as a time-memory trade-off trick since we use a sieving procedure with larger dimension (i.e, β_0) to produce more short vectors. We also have some other memory costs such as the cost for the FFT procedure. However, these costs are negligible compared with the cost of the main sieving step.

7 Application to the Homomorphic Encryption Standard

The Homomorphic Encryption Standard [7] was initiated by several famous researchers/research groups in this area during the Homomorphic Encryption Standardization Workshop [3], hosted at Microsoft Research in Redmond. It suggests security parameters at security level of 128, 192, and 256, respectively.

In the suggested parameter settings, the standard deviation of the noise variable is chosen to be 3.2. The secret distribution could be uniform, the same as noise, or the bounded size in $\{-1, 0, 1\}$. We focus on the bounded secret case since it is the main parameter choice of many important implementations (e.g., the default parameters in the Microsoft SEAL [4]) that could lead to preferable performance.

We set $\gamma = 3$, and the complexity comparison for the security parameters aiming for classic security is shown in Table 7. We only consider the classic gate complexity, i.e., the cost in the Random Access Machine (RAM) model and we fix n to be 1024. The improvement factors vary for different parameter sets. For a parameter set designed for 256-bit security, the new dual approach with the refined lattice reduction strategy could lead to a security loss of about 5 bits.

8 Experimental Verification

In this section we experimentally verify the theoretical complexity estimation. The assumptions in lattice reduction algorithms, such as the d4f gain and the msv gain, have been verified in previous research [34,12,15]. Thus, we mainly perform experimental validation of the success rate of the new FFT distinguisher.

We have generated the samples in $\mathbb{Z}_q$ s.t.,

$$b_j = \sum_{i=0}^{t-1} \hat{a}_{i,j} \cdot \frac{q+1}{2} \cdot s_i, + e_j,$$

in the simulation, where each $\hat{a}_{i,j}$ was generated from a discrete Gaussian distribution $\mathcal{X}_{\sigma_1}$ and e_j was from another discrete Gaussian distribution $\mathcal{X}_{\sigma_2}$. We then implemented the new distinguisher to recover the secret vector of length t. These experiments simulate the processing steps after receiving many short vectors from the BKZ reduction algorithms. The alphabetic size q is set to be 3329, the same value as that in CRYSTALS-Kyber. For simplicity, we generated s_i from a uniform distribution in $\mathbb{Z}_2$. Note that, for parameter sets in public key encryption primitives, the secret s_i is usually set to be small and the variables $\hat{a}_{i,j}$ and e_j are (a sum of) entries from reduction algorithms, thus being wide.

We aim to verify in the experiments that

1. the sample complexity estimation in Equation (11) is correct;
2. and it is sufficient to choose c_0 to be 4 to ensure a high success probability.

For the first purpose, we designed two types of experiments with different values of σ_1 and σ_2, since different noise parts contribute to the final noise with different weights (scales) according to our theoretical analysis. For the second purpose, we ran experiments with sample complexity computed by Equation (11) where c_0 is set to $1, 2$ and 4, respectively.

In each experiment, we chose a typical key with length t and weight $\frac{t}{2}$ and ran the simulation test for 1000 times. The success probabilities in simulation are shown in Table 8. The experimental data match the theoretical prediction from Equation (11) very well. To be more specific, the success probabilities are always 100% in our experiments when the value c_0 is set to 4. We already ensure a high success probability (of 95%) when setting $c_0 = 2$.

In addition, we have simulated the success probability when generating a new key in each run of the test. The secret entry s_i, as before, was generated from a uniform distribution in $\mathbb{Z}_2$, but the weight of the secret vector was not controlled. Thus, the error probability could be slightly higher if the weight of the secret is high. We have only run **Type-II experiments** with the FFT dimension 8 and 16, respectively, and performed 10000 tests in each setting. The coefficient c_0 is set to 4, the value in the theoretical prediction. We succeeded 9979 times for $t = 8$ and 9975 times for $t = 16$, strongly supporting our theoretical estimation.

t	c_0	#(samples) $\log_2(\cdot)$	#(success)	#(test)	success rate
Type-I experiments ($\sigma_1 = 700, \sigma_2 = 1350$)					
8	4	16.36	1000	1000	100%
	2	15.36	976	1000	97.6%
	1	14.36	701	1000	70.1%
12	4	18.20	1000	1000	100%
	2	17.20	990	1000	99.0%
	1	16.20	741	1000	74.1%
16	4	19.87	1000	1000	100%
	2	18.87	999	1000	99.9%
	1	17.87	770	1000	77.0%
Type-II experiments ($\sigma_1 = 500, \sigma_2 = 1500$)					
8	4	17.32	1000	1000	100%
	2	16.32	956	1000	95.6%
	1	15.32	677	1000	67.7%
12	4	18.55	1000	1000	100%
	2	17.55	979	1000	97.9%
	1	16.55	686	1000	68.6%
16	4	19.60	1000	1000	100%
	2	18.60	991	1000	99.1%
	1	17.60	651	1000	65.1%

Table 8: Experimental success probabilities with the novel FFT distinguisher. Here $\gamma = 2$ and the prime field size q is 3329. The value t is the FFT dimension and c_0 is the coefficient in Equation (11). The rows with $c_0 = 4$ correspond to the experiments with number of samples predicted by our theory.

9 Concluding Remarks

We have presented a novel fast dual-type lattice attack for solving the LWE problem, based on two main contributions. Firstly, we have proposed a new efficient distinguisher using the FFT technique with a small alphabetic size. Secondly, we have described a new two-step reduction strategy that first uses a BKZ reduction for a high-quality lattice basis and then employs a progressive sieving step to produce many short vectors. This new reduction framework allows us to take into account the recent advances in lattice algorithms, such as the "dimensions for free" trick and more precise gate estimations on nearest neighbor search. The proposed new algorithm improves the complexity of solving the security parameter sets in the round-3 submissions of CRYSTALS-Kyber and CRYSTALS-Dilithium in both the core-SVP model and the gate-count metric. This new algorithm could recover the secret key of Kyber768 with classical gate complexity below its claimed security level under a model in [12] extrapolated from experimental data. Also, this new algorithm could improve the best-known

attacks on certain FHE parameters. This new dual attack has rather wide applications and could affect many lattice-based primitives.

Acknowledgements

The authors would like to thank the anonymous reviewers for their helpful comments. This work was supported in part by the Swedish Research Council (Grant No. 2019-04166), by the Swedish Foundation for Strategic Research (Grant No. RIT17-0005), and by the Wallenberg Autonomous Systems and Software Program (WASP). The computations/simulations were enabled by resources provided by LUNARC.

References

1. Cryptographic Suite for Algebraic Lattices. `https://pq-crystals.org/index.shtml`, accessed: 2020-08-31
2. HElib. `https://github.com/homenc/HElib`, accessed: 2020-08-31
3. Homomorphic Encryption Standardization Workshop. `https://www.microsoft.com/en-us/research/event/homomorphic-encryption-standardization-workshop/`, accessed: 2020-10-07
4. Microsoft SEAL. `https://www.microsoft.com/en-us/research/project/microsoft-seal`, accessed: 2020-08-31
5. NIST Post-Quantum Cryptography Standardization. `https://csrc.nist.gov/Projects/Post-Quantum-Cryptography/Post-Quantum-Cryptography-Standardization`, accessed: 2018-09-24
6. Submission Requirements and Evaluation Criteria for the Post-Quantum Cryptography Standardization Process. `https://csrc.nist.gov/CSRC/media/Projects/Post-Quantum-Cryptography/documents/call-for-proposals-final-dec-2016.pdf`, accessed: 2021-02-18
7. Albrecht, M., Chase, M., Chen, H., Ding, J., Goldwasser, S., Gorbunov, S., Halevi, S., Hoffstein, J., Laine, K., Lauter, K., Lokam, S., Micciancio, D., Moody, D., Morrison, T., Sahai, A., Vaikuntanathan, V.: Homomorphic encryption security standard. Tech. rep., HomomorphicEncryption.org, Toronto, Canada (November 2018)
8. Albrecht, M., Cid, C., Faugere, J.C., Fitzpatrick, R., Perret, L.: Algebraic algorithms for lwe problems (2014)
9. Albrecht, M.R.: On dual lattice attacks against small-secret LWE and parameter choices in HElib and SEAL. In: Coron, J.S., Nielsen, J.B. (eds.) Advances in Cryptology – EUROCRYPT 2017, Part II. Lecture Notes in Computer Science, vol. 10211, pp. 103–129. Springer, Heidelberg, Germany, Paris, France (Apr 30 – May 4, 2017)
10. Albrecht, M.R., Cid, C., Faugere, J.C., Fitzpatrick, R., Perret, L.: On the complexity of the bkw algorithm on lwe. Designs, Codes and Cryptography 74(2), 325–354 (2015)
11. Albrecht, M.R., Curtis, B.R., Deo, A., Davidson, A., Player, R., Postlethwaite, E.W., Virdia, F., Wunderer, T.: Estimate all the {LWE, NTRU} schemes! In: International Conference on Security and Cryptography for Networks. pp. 351–367. Springer (2018)

12. Albrecht, M.R., Ducas, L., Herold, G., Kirshanova, E., Postlethwaite, E.W., Stevens, M.: The general sieve kernel and new records in lattice reduction. In: Ishai, Y., Rijmen, V. (eds.) Advances in Cryptology – EUROCRYPT 2019, Part II. Lecture Notes in Computer Science, vol. 11477, pp. 717–746. Springer, Heidelberg, Germany, Darmstadt, Germany (May 19–23, 2019)
13. Albrecht, M.R., Fitzpatrick, R., Göpfert, F.: On the efficacy of solving lwe by reduction to unique-svp. In: International Conference on Information Security and Cryptology. pp. 293–310. Springer (2013)
14. Albrecht, M.R., Gheorghiu, V., Postlethwaite, E.W., Schanck, J.M.: Estimating quantum speedups for lattice sieves. In: Moriai, S., Wang, H. (eds.) Advances in Cryptology – ASIACRYPT 2020, Part II. Lecture Notes in Computer Science, vol. 12492, pp. 583–613. Springer, Heidelberg, Germany, Daejeon, South Korea (Dec 7–11, 2020)
15. Albrecht, M.R., Heninger, N.: On bounded distance decoding with predicate: Breaking the "lattice barrier" for the hidden number problem. IACR Cryptol. ePrint Arch. 2020, 1540 (2020), https://eprint.iacr.org/2020/1540
16. Albrecht, M.R., Player, R., Scott, S.: On the concrete hardness of learning with errors. Journal of Mathematical Cryptology 9(3), 169–203 (2015)
17. Alkim, E., Ducas, L., Pöppelmann, T., Schwabe, P.: Post-quantum key exchange - A new hope. In: Holz, T., Savage, S. (eds.) USENIX Security 2016: 25th USENIX Security Symposium. pp. 327–343. USENIX Association, Austin, TX, USA (Aug 10–12, 2016)
18. Arora, S., Ge, R.: New algorithms for learning in presence of errors. In: Aceto, L., Henzinger, M., Sgall, J. (eds.) ICALP 2011: 38th International Colloquium on Automata, Languages and Programming, Part I. Lecture Notes in Computer Science, vol. 6755, pp. 403–415. Springer, Heidelberg, Germany, Zurich, Switzerland (Jul 4–8, 2011)
19. Bai, S., Galbraith, S.D.: Lattice decoding attacks on binary LWE. In: Susilo, W., Mu, Y. (eds.) ACISP 14: 19th Australasian Conference on Information Security and Privacy. Lecture Notes in Computer Science, vol. 8544, pp. 322–337. Springer, Heidelberg, Germany, Wollongong, NSW, Australia (Jul 7–9, 2014)
20. Bai, S., Stehlé, D., Wen, W.: Measuring, simulating and exploiting the head concavity phenomenon in BKZ. In: Peyrin, T., Galbraith, S. (eds.) Advances in Cryptology – ASIACRYPT 2018, Part I. Lecture Notes in Computer Science, vol. 11272, pp. 369–404. Springer, Heidelberg, Germany, Brisbane, Queensland, Australia (Dec 2–6, 2018)
21. Bi, L., Lu, X., Luo, J., Wang, K., Zhang, Z.: Hybrid dual attack on lwe with arbitrary secrets. Cryptology ePrint Archive, Report 2021/152 (2021), https://eprint.iacr.org/2021/152
22. Bleichenbacher, D.: On the generation of dsa one-time keys. presentation at cryptography research. Inc., San Francisco, CA (2007)
23. Bleichenbacher, D.: On the generation of one-time keys in DL signature schemes. Presentation at IEEE P1363 working group meeting (2000)
24. Blum, A., Kalai, A., Wasserman, H.: Noise-tolerant learning, the parity problem, and the statistical query model. Journal of the ACM (JACM) 50(4), 506–519 (2003)
25. Brakerski, Z., Langlois, A., Peikert, C., Regev, O., Stehlé, D.: Classical hardness of learning with errors. In: Boneh, D., Roughgarden, T., Feigenbaum, J. (eds.) 45th Annual ACM Symposium on Theory of Computing. pp. 575–584. ACM Press, Palo Alto, CA, USA (Jun 1–4, 2013)

26. Brakerski, Z., Vaikuntanathan, V.: Efficient fully homomorphic encryption from (standard) LWE. In: Ostrovsky, R. (ed.) 52nd Annual Symposium on Foundations of Computer Science. pp. 97–106. IEEE Computer Society Press, Palm Springs, CA, USA (Oct 22–25, 2011)
27. Brakerski, Z., Vaikuntanathan, V.: Fully homomorphic encryption from ring-LWE and security for key dependent messages. In: Rogaway, P. (ed.) Advances in Cryptology – CRYPTO 2011. Lecture Notes in Computer Science, vol. 6841, pp. 505–524. Springer, Heidelberg, Germany, Santa Barbara, CA, USA (Aug 14–18, 2011)
28. Buchmann, J., Göpfert, F., Player, R., Wunderer, T.: On the hardness of lwe with binary error: Revisiting the hybrid lattice-reduction and meet-in-the-middle attack. In: International Conference on Cryptology in Africa. pp. 24–43. Springer (2016)
29. Budroni, A., Guo, Q., Johansson, T., Mårtensson, E., Wagner, P.S.: Making the BKW algorithm practical for LWE. In: Bhargavan, K., Oswald, E., Prabhakaran, M. (eds.) Progress in Cryptology - INDOCRYPT 2020: 21st International Conference in Cryptology in India. Lecture Notes in Computer Science, vol. 12578, pp. 417–439. Springer, Heidelberg, Germany, Bangalore, India (Dec 13–16, 2020)
30. Chen, Y.: Réduction de réseau et sécurité concrète du chiffrement complètement homomorphe. Ph.D. thesis, Paris 7 (2013)
31. Chen, Y., Nguyen, P.Q.: BKZ 2.0: Better lattice security estimates. In: Lee, D.H., Wang, X. (eds.) Advances in Cryptology – ASIACRYPT 2011. Lecture Notes in Computer Science, vol. 7073, pp. 1–20. Springer, Heidelberg, Germany, Seoul, South Korea (Dec 4–8, 2011)
32. Cheon, J.H., Hhan, M., Hong, S., Son, Y.: A hybrid of dual and meet-in-the-middle attack on sparse and ternary secret lwe. IEEE Access 7, 89497–89506 (2019)
33. Duc, A., Tramèr, F., Vaudenay, S.: Better algorithms for LWE and LWR. In: Oswald, E., Fischlin, M. (eds.) Advances in Cryptology – EUROCRYPT 2015, Part I. Lecture Notes in Computer Science, vol. 9056, pp. 173–202. Springer, Heidelberg, Germany, Sofia, Bulgaria (Apr 26–30, 2015)
34. Ducas, L.: Shortest vector from lattice sieving: A few dimensions for free. In: Nielsen, J.B., Rijmen, V. (eds.) Advances in Cryptology – EUROCRYPT 2018, Part I. Lecture Notes in Computer Science, vol. 10820, pp. 125–145. Springer, Heidelberg, Germany, Tel Aviv, Israel (Apr 29 – May 3, 2018)
35. Espitau, T., Joux, A., Kharchenko, N.: On a hybrid approach to solve small secret lwe. Cryptology ePrint Archive, Report 2020/515 (2020), `https://eprint.iacr.org/2020/515`
36. Gentry, C., Sahai, A., Waters, B.: Homomorphic encryption from learning with errors: Conceptually-simpler, asymptotically-faster, attribute-based. In: Canetti, R., Garay, J.A. (eds.) Advances in Cryptology – CRYPTO 2013, Part I. Lecture Notes in Computer Science, vol. 8042, pp. 75–92. Springer, Heidelberg, Germany, Santa Barbara, CA, USA (Aug 18–22, 2013)
37. Guo, Q., Johansson, T., Löndahl, C.: Solving LPN using covering codes. Journal of Cryptology 33(1), 1–33 (Jan 2020)
38. Guo, Q., Johansson, T., Stankovski, P.: Coded-BKW: Solving LWE using lattice codes. In: Gennaro, R., Robshaw, M.J.B. (eds.) Advances in Cryptology – CRYPTO 2015, Part I. Lecture Notes in Computer Science, vol. 9215, pp. 23–42. Springer, Heidelberg, Germany, Santa Barbara, CA, USA (Aug 16–20, 2015)
39. Guo, Q., Mårtensson, E., Wagner, P.S.: On the sample complexity of solving LWE using bkw-style algorithms. In: IEEE International Symposium on Information Theory, ISIT 2021, Melbourne, Australia, July 12-20, 2021. pp. 2405–2410. IEEE (2021), `https://doi.org/10.1109/ISIT45174.2021.9518190`

40. Howgrave-Graham, N.: A hybrid lattice-reduction and meet-in-the-middle attack against NTRU. In: Menezes, A. (ed.) Advances in Cryptology – CRYPTO 2007. Lecture Notes in Computer Science, vol. 4622, pp. 150–169. Springer, Heidelberg, Germany, Santa Barbara, CA, USA (Aug 19–23, 2007)
41. Kirchner, P., Fouque, P.A.: An improved BKW algorithm for LWE with applications to cryptography and lattices. In: Gennaro, R., Robshaw, M.J.B. (eds.) Advances in Cryptology – CRYPTO 2015, Part I. Lecture Notes in Computer Science, vol. 9215, pp. 43–62. Springer, Heidelberg, Germany, Santa Barbara, CA, USA (Aug 16–20, 2015)
42. Kirchner, P., Fouque, P.A.: Revisiting lattice attacks on overstretched ntru parameters. In: Annual International Conference on the Theory and Applications of Cryptographic Techniques. pp. 3–26. Springer (2017)
43. Laarhoven, T., Mariano, A.: Progressive lattice sieving. In: Lange, T., Steinwandt, R. (eds.) Post-Quantum Cryptography - 9th International Conference, PQCrypto 2018. pp. 292–311. Springer, Heidelberg, Germany, Fort Lauderdale, Florida, United States (Apr 9–11 2018)
44. Lindner, R., Peikert, C.: Better key sizes (and attacks) for LWE-based encryption. In: Kiayias, A. (ed.) Topics in Cryptology – CT-RSA 2011. Lecture Notes in Computer Science, vol. 6558, pp. 319–339. Springer, Heidelberg, Germany, San Francisco, CA, USA (Feb 14–18, 2011)
45. Liu, M., Nguyen, P.Q.: Solving BDD by enumeration: An update. In: Dawson, E. (ed.) Topics in Cryptology – CT-RSA 2013. Lecture Notes in Computer Science, vol. 7779, pp. 293–309. Springer, Heidelberg, Germany, San Francisco, CA, USA (Feb 25 – Mar 1, 2013)
46. Lyubashevsky, V., Ducas, L., Kiltz, E., Lepoint, T., Schwabe, P., Seiler, G., Bai, D.S.S.: CRYSTALS-DILITHIUM. Tech. rep., National Institute of Standards and Technology (2020), available at `https://csrc.nist.gov/projects/post-quantum-cryptography/round-3-submissions`
47. Micciancio, D., Regev, O.: Lattice-based cryptography. In: Post-quantum cryptography, pp. 147–191. Springer (2009)
48. Nguyen, P.Q., Vidick, T.: Sieve algorithms for the shortest vector problem are practical. J. Math. Cryptol. 2(2), 181–207 (2008), `https://doi.org/10.1515/JMC.2008.009`
49. Peikert, C.: Public-key cryptosystems from the worst-case shortest vector problem: extended abstract. In: Mitzenmacher, M. (ed.) 41st Annual ACM Symposium on Theory of Computing. pp. 333–342. ACM Press, Bethesda, MD, USA (May 31 – Jun 2, 2009)
50. Poppelmann, T., Alkim, E., Avanzi, R., Bos, J., Ducas, L., de la Piedra, A., Schwabe, P., Stebila, D., Albrecht, M.R., Orsini, E., Osheter, V., Paterson, K.G., Peer, G., Smart, N.P.: NewHope. Tech. rep., National Institute of Standards and Technology (2019), available at `https://csrc.nist.gov/projects/post-quantum-cryptography/round-2-submissions`
51. Regev, O.: On lattices, learning with errors, random linear codes, and cryptography. In: Gabow, H.N., Fagin, R. (eds.) 37th Annual ACM Symposium on Theory of Computing. pp. 84–93. ACM Press, Baltimore, MA, USA (May 22–24, 2005)
52. Schwabe, P., Avanzi, R., Bos, J., Ducas, L., Kiltz, E., Lepoint, T., Lyubashevsky, V., Schanck, J.M., Seiler, G., Stehlé, D.: CRYSTALS-KYBER. Tech. rep., National Institute of Standards and Technology (2020), available at `https://csrc.nist.gov/projects/post-quantum-cryptography/round-3-submissions`
53. Son, Y., Cheon, J.H.: Revisiting the hybrid attack on sparse and ternary secret lwe. IACR Cryptol. ePrint Arch. 2019, 1019 (2019)