IET Research Journals

Submission Template for IET Research Journal Papers

IET Journals
The Institution of Engineering and Technology

The Impact of the Modulus Switching Technique on Some Attacks against Learning Problems

ISSN 1751-8644
doi: 0000000000
www.ietdl.org

Huy Quoc Le[1] ✉, *Pradeep Kumar Mishra* [1], *Satoshi Nakamura* [1], *Koha Kinjo* [2], *Dung Hoang Duong*[3], *Masaya Yasuda*[4,5]

[1] *Graduate School of Mathematics, Kyushu University, 744 Motooka, Nishi-ku, Fukuoka-shi, Fukuoka-ken, 819-0395, Japan*
[2] *NTT Secure Platform Laboratories, Japan*
[3] *School of Computing and Information Technology, University of Wollongong, Northfields Avenue, Wollongong NSW 2522, Australia*
[4] *Institute of Mathematics for Industry, Kyushu University, 744 Motooka, Nishi-ku, Fukuoka-shi, Fukuoka-ken, 819-0395, Japan*
[5] *JST, CREST, 4-1-8 Honcho, Kawaguchi, Saitama, 332-0012, Japan.*
✉ *E-mail: huyle84@gmail.com*

Abstract: The modulus switching technique has been used in some cryptographic applications as well as in cryptanalysis. For cryptanalysis against the Learning with Errors (LWE) problem and the Learning with Rounding (LWR) problem, it seems that one does not know whether the technique is really useful or not. This work supplies a complete view of the impact of this technique on the decoding attack, the dual attack and the primal attack against both LWE and LWR. For each attack, we give the optimal formula for the switching modulus. The formulas get involved the number of LWE/LWR samples which differs from the known formula in the literature. We also attain the corresponding sufficient conditions saying when we should utilize the technique. Surprisingly, restricted to the LWE/LWR problem that the secret vector is much shorter than the error vector, we also show that performing the modulus switching before using the so-called *rescaling* technique in the dual attack and the primal attack make these attacks worse than only exploiting the rescaling technique as reported by Bai and Galbraith for their attack at ACISP 2014. As an application, we theoretically assess the influence of the modulus switching on the LWE/LWR-based second round NIST PQC submissions.

1 Introduction

Quantum computers that are machines exploiting quantum-mechanical phenomena are supposed to be more powerful than conventional computers. If such a machine is built, despite that the machine will be very useful in many beneficially real applications thanks to its power, it can also be used in compromising the digital world and hence the real world. In fact, if a large-scale quantum computer becomes real, many of the public-key cryptosystems that based on classical hard problems (e.g., the Integer Factorisation Problem or the Discrete Logarithm Problem or the Elliptic Curve Problem) currently in use could be broken due to the quantum computers' capacity in efficiently solving mathematical problems that are difficult or intractable for classical computers [1]. Under the threat of quantum computers, lattice-based cryptography, among others [2], has become a leading candidate for being against the power of these quantum machines.

Among many lattice problems, LWE has been becoming a very important problem in lattice-based cryptography since its introduction in the seminal work of Regev [3] in 2005. The problem uses a noise sampled from some distribution (typically, Gaussian distribution) to hide the secret key. So far, the problem has been well-studied and has been playing a crucial role as an underlying hard problem utilized to built various public-key cryptosystems [3–5]. A disadvantage of the original LWE problem is that the noise is drawn from the discrete Gaussian distribution which is costly to implement in practice. In 2010, a de-randomization variant of LWE using a rounding operation to conceal the secret key instead of a discrete Gaussian noise, the LWR problem, was introduced by Banerjee, Peikert and Rosen [6]. The LWR problem has also several applications in lattice-based cryptography such as pseudorandom functions [6], lossy trapdoor functions, reusable extractors [7], key homomorphic pseudorandom functions [8], etc.

Table 1 The Second-Round NIST PQC Submissions based on LWE and LWR problems

Cryptosystems	LWE/LWR variants
CRYSTALS-Kyber	Module-LWE (MLWE)
FrodoKEM	LWE
LAC	Ring-LWE (RLWE)
NewHope	RLWE
Round5	General LWR
SABER	LWR
ThreeBears	ILWE (Integer version of MLWE)
CRYSTALS-Dilithium	RLWE
qTESLA	RLWE

In order to prepare for the upcoming post-quantum era, National Institute of Standards and Technology (NIST) processed a competition called *NIST Post-Quantum Cryptography (NIST PQC) Standardization* in November 2017. Among 82 candidate packages in total were submitted to NIST at the beginning of the process, 69 submissions were accepted as First-Round Candidates on December 20, 2017. After that from these submissions, 26 packages were chosen to be the *Second-Round NIST PQC Candidates* (we also refer them as *the second round NIST PQC submissions*) of the competition as announced on January 30, 2019, by the NIST organiser [9]. It is noteworthy to know that there are up to 9 Second-Round NIST PQC Candidates that are based on LWE/LWR and their variants (see also Table 1 for a summary), whereas the remaining candidates (17 submissions) are NTRU-based, multivariate-based, code-based, hash-based ones, and so forth. Remark that the security, cost and performance, and algorithm and implementation characteristics are criteria that NIST considered in selecting the second round candidates [9, Subsection 2.3]. Same as other lattice problems, LWE/LWR variants enjoy very strong security proofs based on worst-case hardness, relatively efficient implementations, as well as great simplicity

Table 2 **Choose the optimal switching modulus q' for some typical attacks against LWE and LWR problems**

Attacks	Literature [11]	Our work	
		Optimal modulus	Notations
Decoding		$q' \approx \sqrt{\frac{(m-n)N}{nM}}$	$N = n\sigma_s^2$
Dual	$q' \approx \sqrt{\frac{n\sigma_s^2}{12\sigma^2}} \cdot q$ $\approx \sqrt{N/M}$	$q' \approx \sqrt{\frac{mN}{nM}}$	LWE: $M := M_{lwe} = \frac{12\sigma_{lwe}^2}{q^2}$
Primal		$q' \approx \sqrt{\frac{(m+12)N}{(n+1)M}}$	LWR: $M := M_{lwr} = \frac{q^2+2q}{p^2q^2} \approx \frac{12\sigma_{lwr}^2}{q^2}$, $\sigma_{lwr} = \sqrt{\frac{q^2+2pq}{12p^2}}$

(cf. [10]). These characteristics make LWE/LWE-based candidates important and leading ones for the NIST's consideration.

For more details in the security aspect, the hardness of LWE has been well studied in the literature [3, 11, 12]. Regev [3] showed that the LWE problem is as hard to solve as several worst-case lattice problems. Also, for LWR, there are some analyses of its hardness based on reducing LWR to LWE under certain constraints on parameters such as [6, 7, 13, 14]. And, until now, the only known approach in attacking LWR is to transform an LWR instance to an LWE instance. Most of the strategies for breaking LWR are able to be adapted from attacks against an LWE modulo q instance [15, 16] such as the dual attack, the decoding attack, the primal attack, algebraic attacks [17] and so on (cf. [11] for more details).

In this work, we just focus on the decoding attack, the dual attack and the primal attack. While the first attack was mentioned in our preliminary work [18], the second and the third are attacks often mentioned in security analyses of lattice-based NIST PQC submissions. Compared to the decoding attack, the dual attack and the primal attack are much better, especially in the case that the secret (and the noise, the error) is sampled from (even sparse) small sets and that not many numbers of samples are given. For example, the secret key of Round5, a second round NIST PQC candidate, is sampled according to a fixed Hamming weight distribution of support $\{-1, 0, 1\}$, named $\mathcal{H}_{n,k}(h)$, from which each drawn vector of length $n \cdot k$ has exactly h non-zero components [15].

The modulus switching technique was used for the first time aiming to speed up the homomorphic encryption operations [19]. Then the technique was also used to evaluate the classical hardness of LWE problem [12]. Recently, the technique was modified to combine with the Blum-Kalai-Wasserman (BKW) algorithm on LWE [20]. The technique allows to transform an LWE modulo q instance to an LWE modulo q' instance with q' (called *the switching modulus*) is typically chosen as

$$q' \approx \frac{\sigma_s}{\sigma} \cdot \sqrt{\frac{n}{12}} \cdot q, \tag{1}$$

where n is the length of the secret and σ_s, σ are standard deviations of the secret and the error of the original LWE modulo q instance, respectively (cf. [11, Lemma 2]). Remark that, Eq. (1) does not involve the number of LWE samples m. It is noteworthy to know that the number of LWE samples plays an important role in the success of attacks against the LWE problem. The authors of [21] analyzed the hardness of LWE instances given a restricted number of samples. Since then, they extended the LWE-Estimator of [11] which is a software tool used to estimate the hardness of concrete LWE instances and to choose parameters for lattice-based primitives. Furthermore, they also showed the impact of restricting the number of available samples.

Obviously, Eq. (1) just supplies us a commonly used switching modulus q' not aiming to strengthen the power of known attacks. Also, note that the effect of the modulus switching technique on attacks against the LWE/LWR problems as well as on other techniques, e.g., the so-called rescaling technique, has not been studied carefully so far. The present work is to try to close the gap we have mentioned.

Our contribution: This paper is the extended version of our CANS 2018 paper [18]. In general, the present work extends our approach in [18] to the dual attack and the primal attack for not only LWR but also LWE. We processed a complete consideration of the impact of the modulus switching technique being used to solve both LWE and LWR on these typical attacks. More specifically,

(i) We set up a common framework for both problems. To do that, we scrutinized in details transforming from an LWR instance to a corresponding LWE instance as well as the behavior of the induced LWR error. Furthermore, the successful range in which a search LWR instance can be solved by the decoding attack associated with Babai's Nearest Plane algorithm was also determined. Based on the successful range, we take LWR instances that are consistent with our experiments. This is the completed work of [18].

(ii) We obtained the optimal value for the switching modulus with respect to each attack (see Table 2 for a summary) accompanied with the sufficient condition for exploiting the modulus switching technique in attacking the LWE/LWR problems. Our optimal formulas for the switching modulus get involved the number of LWE/LWR samples which is different from Eq. (1) (see Table 4 for a summary and see also Table 5 for a comparison between the values of the switching moduli according to each attack summarized in Table 2 and the values of q' according to Eq. (1) with some specific parameters).

(iii) We compared the efficacy of the modulus switching and that of the so-called rescaling technique which is mentioned in attacks against LWE/LWR in the case that the secret is much smaller than the error (e.g., the secret is a binary or trinary vector) (see Table 3 for a summary).

(iv) We also assessed the effect of the modulus switching technique on the rescaling technique. Our computation theoretically confirms again the result by Bai and Galbraith at ACISP 2014 [22] that their attack which exploits the rescaling technique is weakened by applying the modulus switching.

(v) Finally, based on our theoretical results, we also evaluated how the LWE/LWR-based NIST PQC submissions are impacted by the modulus switching.

To the best of our knowledge, this work is the first attempt to evaluate carefully the modulus switching's influence on attacks breaking the LWE/LWR problems. We expect that our work will not only provide with a different perspective in exploiting the modulus switching technique to attack LWR/LWE but also gets more attention in other application scenarios.

Organisation: Section 2 gives some background knowledge necessary to our work later on. In Section 3, we first remind the typical way we transform an LWR instance into an LWE instance, then determine the distribution of the induced LWR error. We also estimate the successful range for the decoding attack against LWR in this section. We will review the modulus switching technique applied to LWE and LWR in Section 4. Choosing the optimal switching modulus for the decoding attack, the dual attack and the primal attack will be conducted in Section 5. In Section 6, we compare the modulus switching with the so-called rescaling technique when applied separately to attacks. A valuation on the impact of the modulus switching to the efficacy of the rescaling technique will be

Table 3 The sufficient condition for that the modulus switching technique is better than the rescaling technique

Attacks	Sufficient condition
Dual	$\left(\frac{12\sigma^2}{\sigma_s^2}\right)^n \geq \frac{(m+n)^{m+n}}{m^m} \cdot \frac{\sigma^{2n}}{\sigma_s^{2(m+n)}}$
Primal	$\left(\frac{12(N+m\sigma^2)}{m+n+1}\right)^{m+n+1} \geq \left(\frac{(m+12)N}{n+1}\right)^{n+1} \cdot \left(q^2M\right)^m \cdot \frac{\sigma_s^{2m}}{\sigma^{2m}}$

done in Section 7. Section 8 presents some experimental results relating to using the modulus switching technique in solving the LWR problem. Section 9 is devoted to summarize the second round NIST PQC candidates which rely their security on the hardness of the LWE/LWR variants and evaluate theoretically the impact of the modulus switching on them. Section 10 is to conclude our work.

2 Preliminaries

2.1 Notations

In this work, we represent (column) vectors in lower bold letters, e.g., vector $\mathbf{a}$, matrices in upper bold letters, e.g., matrix $\mathbf{A}$. We write $\mathbf{v}^t$ (resp. $\mathbf{A}^t$) as the transpose of the vector $\mathbf{v}$ (resp. the matrix $\mathbf{A}$). The norm of a vector $\mathbf{v}$ is the standard Euclidean norm computed as $\|\mathbf{v}\| = \sqrt{\langle \mathbf{v}, \mathbf{v} \rangle}$.

If $S = \{a_1, \cdots, a_m\}$ with $a_i \in \mathbb{R}$ then $k \cdot S = \{k \cdot a_1, \cdots, k \cdot a_m\}$ for any $k \in \mathbb{R}$. The logarithm of base 2 of a positive real number x will be written as $\log(x)$. We use $\mathcal{U}(S)$ to indicate the uniform distribution over the set S. The rounding operation $\lfloor a \rceil$ outputs the integer closest to a and in the case of a tie, it outputs the integer next to a. For any positive integer q, we denote by $\mathbb{Z}_q = \{0, 1, \cdots, q-1\}$ the set of integers modulo q. By "$a := b$" we mean defining the new variable a by assigning its value to be b

We write $x \leftarrow \chi$ to say that the random variable x follows the probability distribution χ or x is sampled from the distribution χ. Let $\mathbf{v}$ be a vector, $\mathbf{A}$ be a matrix and f be a polynomial. Notations like $\mathbf{v} \leftarrow \chi^n$, $\mathbf{A} \leftarrow \chi^{m \times n}$, $f \leftarrow \chi^n$, etc., say that $\mathbf{v}$, $\mathbf{A}$ and f are sampled element-wise or coefficient-wise according to the distribution χ. For a real number k, the notation $y \leftarrow k \cdot \chi$ means that $y = k \cdot x$ for some x that follows the probability distribution χ.

2.2 Lattices

The lattice $\mathcal{L} = \mathcal{L}(\mathbf{A})$ generated by the column matrix $\mathbf{A} = [\mathbf{a}_1, \cdots, \mathbf{a}_m] \in \mathbb{R}^{n \times m}$ of m linearly independent vectors is defined to be the set of all linear integral combinations of $\mathbf{a}_i$'s, i.e., $\mathcal{L}(\mathbf{A}) = \{\mathbf{A}.\mathbf{x} : \mathbf{x} \in \mathbb{Z}^m\} = \left\{\sum_{i=1}^m x_i \mathbf{a}_i : x_i \in \mathbb{Z}\right\}$. We call the matrix $\mathbf{A}$ *a basis* of $\mathcal{L}$ and call each $\mathbf{a}_i$ *a basis vector*. The *rank* of the lattice is the number of basis vector (i.e., m). The *dimension* of the lattice is the number of entries in each basis vector (i.e., n). If $m = n$, the lattice is called to be *full-rank*. Notice that, every lattice has infinitely many bases up to a unimodular matrix of determinant ± 1. Hence, if $\mathbf{A}$ and $\mathbf{B}$ are two different bases of the lattice $\mathcal{L}$, then $\det(\mathbf{A}^t\mathbf{A}) = \det(\mathbf{B}^t\mathbf{B})$. We call $\det(\mathcal{L}(\mathbf{A})) := \sqrt{\det(\mathbf{A}^t\mathbf{A})}$ the *determinant* (or *volume*) of the lattice $\mathcal{L}(\mathbf{A})$.

The Gram-Schmidt matrix $\mathbf{A}^* = [\mathbf{a}_1^*, \cdots, \mathbf{a}_m^*]$ for a basis $\mathbf{A} = [\mathbf{a}_1, \cdots, \mathbf{a}_m]$ is defined by setting $\mathbf{a}_1^* = \mathbf{a}_1$ and $\mathbf{a}_i^* = \mathbf{a}_i - \sum_{j=1}^{i-1} \mu_{i,j} \cdot \mathbf{a}_j^*$, where $\mu_{i,j} = \langle \mathbf{a}_i, \mathbf{a}_j^* \rangle / \|\mathbf{a}_j^*\|^2$, for $i = 2, \cdots, m$. We can prove that $\det(\mathcal{L}(\mathbf{A})) = \prod_{i=1}^m \|\mathbf{a}_i^*\|$. The *fundamental parallelepiped* associated with a basis $\mathbf{A} = [\mathbf{a}_1, \cdots, \mathbf{a}_m]$ is $\mathcal{P}_{1/2}(\mathbf{A}) = \left\{\sum_{i=1}^m x_i \mathbf{a}_i : x_i \in \left[-\frac{1}{2}, \frac{1}{2}\right)\right\}$. We define the fundamental parallelepiped $\mathcal{P}_{1/2}(\mathbf{A}^*)$ for the Gram-Schmidt matrix $\mathbf{A}^*$ in the same way.

There always exist non-zero vectors having the smallest norm and non-zero vectors having the second smallest norm in a lattice $\mathcal{L}$. We call these norms *the first minimum* $\lambda_1(\mathcal{L})$ and *the second minimum* $\lambda_2(\mathcal{L})$, respectively.

Let $\mathcal{L}$ be a lattice of rank m and S be a measurable subset of the corresponding space $\mathbb{R}^m$, *Gaussian Heuristic* says that $\#S \cap \mathcal{L} \approx \mathrm{vol}(S)/\det(\mathcal{L})$. Consequently, we have the following estimation for the length of the shortest non-zero vectors of $\mathcal{L}$:

$$\lambda_1(\mathcal{L}) \approx \sqrt{\frac{m}{2\pi\mathfrak{e}}} \det(\mathcal{L})^{1/m},$$

where $\mathfrak{e}$ is the mathematical constant being the base of the natural logarithm.

For integers q, m, n $(m \geq n)$, given a random matrix $\mathbf{A} \in \mathbb{Z}_q^{m \times n}$, we consider *q-ary lattices*

$$\Lambda_q(\mathbf{A}) = \left\{\mathbf{u} \in \mathbb{Z}^m : \mathbf{u} = \mathbf{A}\mathbf{s} \bmod q \text{ for some } \mathbf{s} \in \mathbb{Z}^n\right\}$$

and

$$\Lambda_q^\perp(\mathbf{A}) = \left\{\mathbf{x} \in \mathbb{Z}^m : \mathbf{x}^t \cdot \mathbf{A} = 0 \bmod q\right\}.$$

It is well known that $\det(\Lambda_q(\mathbf{A})) = q^{m-n}$ and $\det(\Lambda_q^\perp(\mathbf{A})) = q^n$ with high probability.

In lattice-based cryptography, there are two very important problems: The *Shortest Vector Problem* (SVP) and the *Closest Vector Problem* (CVP).

- SVP is to find a lattice vector of the first minimum given a basis of a lattice.
- CVP is given a basis of a lattice and a target vector to search for a lattice vector that is closest to the target vector.

Almost attacks against LWE and LWR can be reduced to solving these problems over some lattice.

2.3 Lattice Basis Reduction Algorithms and Root Hermite Factor

A basis of a lattice can be reduced using the so-called *lattice basis reduction* (LBR) algorithms to obtain a new basis consisting of short and nearly orthogonal lattice vectors. Among some, the LLL algorithm [23] and the Block-wise Korkine-Zolotarev algorithm (BKZ) [24] are two algorithms typically used in practice. The former is a polynomial-time algorithm while the latter can be considered as a block version of the former with exponential complexity. We briefly remind them in the following.

The LLL Algorithm: The LLL algorithm, named after Lenstra, Lenstra and Lovász, is a polynomial algorithm used to reduce a lattice basis to a δ-LLL-reduced basis $\mathbf{A} = [\mathbf{a}_1, \cdots, \mathbf{a}_m]$ with the reduction factor $\frac{1}{4} < \delta < 1$ satisfying both the following conditions:

- (Size-reduced condition): $|\mu_{i,j}| \leq \frac{1}{2}$ for all $1 \leq j < i \leq m$,
- (Lovász condition): $\delta\|\mathbf{a}_{i-1}^*\|^2 \leq \|\mathbf{a}_i^* + \mu_{i,i-1}\mathbf{a}_{i-1}^*\|^2$ for all $2 \leq i \leq m$.

The BKZ Algorithm: The BKZ algorithm is an algorithm that on input a lattice basis outputs a (δ, β)-BKZ-reduced basis with factor $\frac{1}{4} < \delta < 1$ and blocksize $2 \leq \beta \leq m$. Additionally, we define *the orthogonal projection* over $(\mathbf{b}_1, \cdots, \mathbf{b}_{i-1})^\perp$ by $\pi_i : \mathbb{R}^n \to \mathrm{span}(\mathbf{b}_1, \cdots, \mathbf{b}_{i-1})^\perp$ for $1 \leq i \leq m$. Particularly, π_1 is considered as the identity map. A basis $\mathbf{A} = [\mathbf{a}_1, \cdots, \mathbf{a}_m]$ is called (δ, β)-BKZ-reduced if the following hold:

- It is a δ-LLL reduced basis,

Table 4 The sufficient condition for using the modulus switching makes attacks stronger than without using the technique.

Attacks	Sufficient condition
Decoding	$\left(\frac{12\sigma^2}{\sigma_s^2}\right)^n \geq \frac{m^m}{(m-n)^{m-n}}$
Dual	$\left(\frac{12\sigma^2}{\sigma_s^2}\right)^n \geq \frac{(m+n)^{m+n}}{m^m}$
Primal	$\left(\frac{12(N+m\sigma^2)}{m+n+1}\right)^{m+n+1} \geq \left(\frac{(m+12)N}{n+1}\right)^{n+1} \cdot \left(q^2 M\right)^m$

• The Gram-Schmidt vector $\mathbf{a}_i^*$ is the shortest vector in the projective sublattice generated by $\mathbf{A}_{[i:j]} := [\pi_i(\mathbf{a}_i), \cdots, \pi_i(\mathbf{a}_j)]$, (i.e., $\|\mathbf{a}_i^*\| = \lambda_1(\mathcal{L}(\mathbf{A}_{[i:j]}))$ for all $1 \leq i \leq m$ and $j = \min(i+\beta-1, m)$.

Root Hermite Factor: Let $\mathcal{L}$ be a lattice of rank m and $\mathbf{A}$ be a reduced lattice basis obtained using some LBR algorithm, say $\mathcal{A}$, the *root Hermite factor* (rHF) $\delta_{\mathcal{A}}$ of $\mathcal{A}$ with respect to $\mathbf{A}$ is the constant given by

$$\delta_{\mathcal{A}} = \left(\frac{||\mathbf{u}_1||}{\det(\mathcal{L})^{1/m}}\right)^{\frac{1}{m}}, \tag{2}$$

where $\mathbf{u}_1$ is a shortest non-zero vector in $\mathbf{A}$. Gama and Nguyen in [25] attempted to estimate the rHF of LLL and BKZ for random matrices. Namely, they estimated that the rHF of LLL is $\delta_{\text{LLL}} \approx 1.0219$ on average in high dimension ≥ 100 while that of BKZ with blocksize $\beta = 20$ is $\delta_{\text{BKZ}} \approx 1.0128$.

Unfortunately, however, these experimental results of [25] for random matrices may be not perfectly fit for q-ary lattices. That is the reason why Kudo et al. in [26] conducted intensively an experiment on q-ary lattices to estimate the quantity of $\min_{i=1}^m \|\mathbf{b}_i^*\|$ from which they defined an alternative measure as follows:

$$c_{\mathcal{A}} := \left(\frac{\min_{i=1}^m \|\mathbf{b}_i^*\|}{\det(\Lambda_q(\mathbf{A}))^{1/m}}\right)^{\frac{1}{m}}, \tag{3}$$

where $\mathbf{b}_i^*$'s are Gram-Schmidt vectors of a basis of the q-ary lattice $\Lambda_q(\mathbf{A})$, say $\mathbf{B} = \{\mathbf{b}_1, \cdots, \mathbf{b}_m\}$, that is already reduced by some LBR algorithm $\mathcal{A}$. Clearly, $c_{\mathcal{A}} \leq 1$ since $\min_{i=1}^m \|\mathbf{b}_i^*\| \leq (\prod_{i=1}^m \|\mathbf{b}_i^*\|)^{1/m} = \det(\Lambda_q(\mathbf{A}))^{1/m}$. Especially, Kudo et al. [26] estimated that $c_{\text{LLL}} = 0.9775$ whereas using BKZ with blocksize $\beta = 20$, they got $c_{\text{BKZ}} = 0.9868$ (cf. [26, Table 1]).

If we still denote the rHF for q-ary lattices by $\delta_{\mathcal{A}}$ then it seems that $\delta_{\mathcal{A}} \approx 1/c_{\mathcal{A}}$. For instance, with $c_{\text{LLL}} = 0.9775$ and $c_{\text{BKZ}} = 0.9868$, we have $1/c_{\text{LLL}} = 1.0230$ and $1/c_{\text{BKZ}} = 1.0139$, respectively, that are quite close to the rHF for random matrices mentioned above.

We will use (3) to reach an important heuristic that is useful for our work (see Subsection 5.2).

2.4 Probability

2.4.1 Variance of Random Variables: We denote the variance of a random variable X by σ_X^2. For $a, b \in \mathbb{Z}$, the variance of a random variable X following the discrete uniform distribution $\mathcal{U}(\{a, a+1, \cdots, b-1, b\})$ is $\sigma_X^2 = ((b-a+1)^2 - 1)/12$. If X follows the continuous uniform distribution $\mathcal{U}(a, b)$ then $\sigma_X^2 = (b-a)^2/12$. Assume that $Z = X + Y$ where X, Y are independent random variables then $\sigma_Z^2 = \sigma_X^2 + \sigma_Y^2$. Finally, for every random variable X and for every constant $k \in \mathbb{R}$, let $Y = kX$, then we have $\sigma_Y^2 = k^2\sigma_X^2$.

2.4.2 Gaussian Distribution: The continuous Gaussian distribution $\mathcal{D}_{\mu,\sigma}$ of mean μ and standard deviation $\sigma > 0$ is defined by its probability density function (pdf)

$$\mathcal{D}_{\mu,\sigma}(x) = \frac{1}{\sqrt{2\pi\sigma^2}} \cdot e^{-\frac{(x-\mu)^2}{2\sigma^2}}, \forall x \in \mathbb{R}.$$

In this paper, we will also mention to the discrete Gaussian distribution on $\mathbb{Z}$ centered at 0 and width parameter αq, denoted by $\mathcal{D}_{\mathbb{Z},\alpha q}$. The standard deviation of the distribution is roughly $\sigma = \alpha q/\sqrt{2\pi}$.

2.4.3 Convolution of Two Distributions: Let X and Y be continuously distributed independent random variables with probability density functions f_X and f_Y. Then the pdf of the random variable $Z = X + Y$ is the convolution of f_X and f_Y given by

$$\begin{aligned} f_Z(z) = (f_X * f_Y)(z) &= \int_{-\infty}^{+\infty} f_X(t) f_Y(z-t) dt \\ &= \int_{-\infty}^{+\infty} f_X(z-t) f_Y(t) dt. \end{aligned} \tag{4}$$

2.5 LWE Problem and LWR Problem

The LWE problem proposed by Regev in 2005 [3] has been playing a significant role in lattice-based cryptography. In the original LWE by [3], the given LWE secret vector is chosen uniformly at random over $\mathbb{Z}_q^n$ while the LWE error follows a discrete Gaussian distribution over a lattice. However, there is a reduction from the original LWE problem to a LWE variant whose both secret and error following the same distribution [27]. Thus, in the following, we will generally define the LWE problem whose secret that follows the distribution χ_s and error that follows the distribution χ_e in which these two distributions may be identical.

Definition 1 (LWE Sample)**.** *Let χ_s and χ_e be two distributions over $\mathbb{Z}_q$. Given a secret vector $\mathbf{s} \leftarrow \chi_s^n$ where χ_s has variance σ_s^2, an LWE$_{n,q,\chi_s,\chi_e}$ sample is obtained by choosing a vector $\mathbf{a} \leftarrow \mathbb{Z}_q^n$ uniformly at random, sampling an error term $e \leftarrow \chi_e$ and outputting $(\mathbf{a}, c = \langle \mathbf{a}, \mathbf{s} \rangle + e \ mod \ q) \in \mathbb{Z}_q^n \times \mathbb{Z}_q$.*

If we have such m samples $(\mathbf{a}_i, c_i = \langle \mathbf{a}_i, \mathbf{s} \rangle + e_i)$ for $i = 1, \cdots, m$, we can collect them as $(\mathbf{A}, \mathbf{c} = \mathbf{A} \cdot \mathbf{s} + \mathbf{e})$ in which $\mathbf{A}$ is an $(m \times n)$- matrix whose i-th row is $\mathbf{a}_i$, $\mathbf{c} = (c_1, c_2, \cdots, c_m)^t$, and $\mathbf{e} = (e_1, e_2, \cdots, e_m)^t$. We call $(\mathbf{A}, \mathbf{c} = \mathbf{A} \cdot \mathbf{s} + \mathbf{e})$ an *LWE modulo q instance*.

Definition 2 (LWE Problems)**.** *Given an LWE modulo q instance $(\mathbf{A}, \mathbf{c} = \mathbf{A} \cdot \mathbf{s} + \mathbf{e}) \in \mathbb{Z}_q^{m \times n} \times \mathbb{Z}_q^m$.*

- *The search-LWE (sLWE) problem is to find the secret* $\mathbf{s}$.
- *The decision-LWE (dLWE) problem requires to distinguish the LWE instance from the uniform pair $(\mathbf{A}, \mathbf{c}) \in \mathbb{Z}_q^{m \times n} \times \mathbb{Z}_q^m$.*

Stress that, sLWE and dLWE are equivalent in the sense that if one can solve one problem then one will be able to solve the another by [3, Lemma 4.2].

The LWR problem, a de-randomization variant of LWE, was also introduced by Banerjee, Peikert and Rosen in [6] to avoid using Gaussian distribution which is quite complicated to sample and also suffers side channel attacks. Let p and q be two moduli such that $2 \leq p \leq q$. We define the (q, p)-modulo rounding operation, denoted by $\lfloor . \rceil_{q,p}$, as follows: for $x \in \mathbb{Z}_q$, $\lfloor x \rceil_{q,p} = \lfloor (p/q) \cdot x \rceil \in \mathbb{Z}_p$. As usual, we can extend the operation for vectors, matrices as well as polynomials by taking it component-wise, such as for $\mathbf{x} = (x_1, \cdots, x_n) \in \mathbb{Z}_q^n$, we have $\lfloor \mathbf{x} \rceil_{q,p} = (\lfloor x_1 \rceil_{q,p}, \cdots, \lfloor x_n \rceil_{q,p}) \in \mathbb{Z}_p^n$.

Definition 3 (LWR Sample)**.** *For a secret vector $\mathbf{s} \leftarrow \chi_s^n$ where χ_s is some distribution over $\mathbb{Z}_q$ of variance σ_s^2, an LWR$_{n,q,p,\chi_s}$ sample is obtained by choosing a vector $\mathbf{a} \leftarrow \mathbb{Z}_q^n$ uniformly at random and outputting $(\mathbf{a}, c = \lfloor \langle \mathbf{a}, \mathbf{s} \rangle \rceil_{q,p}) \in \mathbb{Z}_q^n \times \mathbb{Z}_p$.*

Assume that we have such m LWR samples then we can write them as $(\mathbf{A}, \mathbf{c} = \lfloor \mathbf{A}\mathbf{s} \rceil_{q,p}) \in \mathbb{Z}_q^{m \times n} \times \mathbb{Z}_p^m$, where $\mathbf{A}$ is a matrix whose rows are $\mathbf{a}_i$ and $\mathbf{c}$ is a column vector whose elements are c_i. We call $(\mathbf{A}, \mathbf{c} = \lfloor \mathbf{A}\mathbf{s} \rceil_{q,p})$ an LWR modulo (q, p) instance.

Table 5 We compute the value of the optimal switching modulus q' according to literature using Eq. (1) and our formulas summarized in Table 2. We can see that, for the decoding attack, our formulas cannot compute q' when if $m < n$ (we denote that by "–") while for the dual attack and the primal attack, we always have the optimal q' even with $m = 1$. The modulus q' computed via Eq (1) does not depends on the number of samples m.

n	m	bit size of q	bit size of p	σ_s	$\log(q')$ (Eq. (1))	$\log(q')$ (decoding)	$\log(q')$ (dual)	$\log(q')$ (primal)
60	61	15	11	2/3	13.66	10.70	13.67	13.79
60	166	15	11	2/3	13.66	14.07	14.39	14.43
60	1	15	11	2/3	13.66	–	10.70	12.54
80	209	18	13	2/3	15.86	16.21	16.56	16.59
80	30	18	13	2/3	15.86	–	15.16	15.39
80	2	18	13	4	17.16	–	14.49	15.89
80	79	18	13	4	17.16	–	17.15	17.241
100	147	20	15	2/3	18.02	17.48	18.31	18.36
100	120	20	10	14	15.23	14.06	15.36	15.42
100	40	20	10	10	14.98	–	14.32	14.50

Definition 4 (LWR Problems)**.** *Given an LWR modulo* (q, p) *instance* $(\mathbf{A}, \mathbf{c} = \lfloor \mathbf{A}\mathbf{s} \rceil_{q,p}) \in \mathbb{Z}_q^{m\times n} \times \mathbb{Z}_p^m$.

- *The search-LWR (sLWR) problem is to find the secret* $\mathbf{s}$.
- *The decision-LWR (dLWR) problem requires to distinguish the LWR instance from the uniform pair* $(\mathbf{A}, \mathbf{c}) \in \mathbb{Z}_q^{m\times n} \times \mathbb{Z}_p^m$.

3 Reducing the LWR modulo (q, p) to the LWE modulo q

The following reduction, called *q-reduction*, is used to transform an LWR instance consisting of samples of the form $(\mathbf{a}, c = \lfloor \langle \mathbf{a}, \mathbf{s} \rangle \rceil_{q,p}) \in \mathbb{Z}_q^n \times \mathbb{Z}_p$ to an LWE modulo q instance of the form $(\mathbf{a}, c_1) \in \mathbb{Z}_q^n \times \mathbb{Z}_q$ in which

$$\begin{aligned} c_1 &= \left\lfloor \frac{q}{p} \cdot c \right\rceil \bmod q \\ &= \left\lfloor \frac{q}{p} . \left(\frac{p}{q} \cdot (\langle \mathbf{a}, \mathbf{s} \rangle + qu) + e_1 \right) \right\rceil \bmod q \\ &= (\langle \mathbf{a}, \mathbf{s} \rangle + e) \bmod q = \langle \mathbf{a}, \mathbf{s} \rangle \bmod q + e, \end{aligned} \tag{5}$$

where $e_1 \in \left(-\frac{1}{2}, \frac{1}{2}\right], u \in \mathbb{Z}$, and $e := \lfloor (q/p) \cdot e_1 \rceil$. For short, we call the error e *q-error*.

Note that, in the reduction above, we used the following assumption:

Assumption 1: We assume that in our work, the error induced in the process of the reduction from an LWR instance to a corresponding LWE instance is not changed by a modulo operation. Formally, for an LWE sample $(\mathbf{a}, (\langle \mathbf{a}, \mathbf{s} \rangle + e) \bmod q) \in \mathbb{Z}_q^n \times \mathbb{Z}_q$, we assume that

$$(\langle \mathbf{a}, \mathbf{s} \rangle + e) \bmod q = \langle \mathbf{a}, \mathbf{s} \rangle \bmod q + e.$$

This assumption was also used in many previous works relating to LWE such as [19, 26, 28, 29]. We still use this assumption later on.

3.1 Distribution of the q-Error e

The q-reduction above has been typically considered in recent works relating to the LWR problem (see, e.g., [15, 30, 31]). In these works, it is heuristically assumed that e is continuously uniform over $(-q/(2p), q/(2p)]$. If so, the variance is just $\sigma_{lwr}^2 \approx q^2/(12p^2)$. In this section, we show that e is actually distributed according to a discrete uniform distribution over the set $\{-\lfloor \lfloor q/2 \rfloor / p \rceil, \cdots, \lfloor \lfloor q/2 \rfloor / p \rceil\}$, hence its variance is actually $\sigma_{lwr}^2 \approx (q^2 + 2pq)/(12p^2)$ which is significantly greater than $q^2/(12p^2)$ in the case $q \gg p$.

To begin with, we state a simple lemma on the rounding operation via the discrete uniform distribution.

Lemma 1. *Given a non-zero real number* b*. Set* $b_0 := -\lfloor b \rceil$, $b_1 := -\lfloor b \rceil + 1, \cdots, b_{t-1} := \lfloor b \rceil - 1$ *and* $b_t := \lfloor b \rceil$ *and define the set* $B := \{b_0, \cdots, b_t\}$*. Let* x *be a real number taken uniformly at random from* $[-b, b]$*. Then*

$$\Pr[\lfloor x \rceil = b_i | b_i \in B] = \Pr[\lfloor x \rceil = b_j | b_j \in B], \text{ for all } i, j \in [t-1].$$

In particular,

$$\Pr[\lfloor x \rceil = b_0] = \Pr[\lfloor x \rceil = b_t] \leq \Pr[\lfloor x \rceil = b_i | b_i \in B], \forall i \in [t-1].$$

Proof: The idea for the proof is easy. Firstly, note that, for $1 \leq i \leq t-1$, we have

$$\Pr[\lfloor x \rceil = b_i] = \Pr[x \in [b_i - 1/2, b_i + 1/2)],$$

which implies the first statement in the lemma. Secondly, since $b_0 = -\lfloor b \rceil$ and $b_t = \lfloor b \rceil$, so $b_0 - 1/2 \leq -b < b_0 + 1/2$ and $b_t - 1/2 \leq b < b_t + 1/2$,

$$\begin{aligned} \Pr[\lfloor x \rceil = b_0] &= \Pr[x \in [-b, b_0 + 1/2)] \\ &\leq \Pr[x \in [b_0 - 1/2, b_0 + 1/2)], \end{aligned}$$

and

$$\begin{aligned} \Pr[\lfloor x \rceil = b_t] &= \Pr[x \in [b_t - 1/2, b]] \\ &\leq \Pr[x \in [b_t - 1/2, b_t + 1/2)]. \end{aligned}$$

□

Now we give the theorem describing the behaviour of the q-error e.

Theorem 1. *Set* $b := \frac{\lfloor q/2 \rfloor}{p}$ *and* $A := \{-\lfloor b \rceil + 1, \cdots, \lfloor b \rceil - 1\}$*. Also let* e *be the q-error defined as in* (5)*. Then we have:*

$$\Pr[e = a | a \in A] = \frac{1}{2b},$$

and

$$\Pr[e = -\lfloor b \rceil] = \Pr[e = \lfloor b \rceil] = \frac{b - \lfloor b \rceil + \frac{1}{2}}{2b} \leq \frac{1}{2b}.$$

However, less precisely, we can say that e *is uniform over*

$$B := \{-\lfloor b \rceil, -\lfloor b \rceil + 1, \cdots, \lfloor b \rceil - 1, \lfloor b \rceil\}.$$

Then the variance of the q-error e *is*

$$\sigma_{lwr}^2 \approx \frac{\left(2\left\lfloor \frac{\lfloor q/2 \rfloor}{p} \right\rceil + 1\right)^2 - 1}{12} \approx \frac{q^2 + 2qp}{12p^2}. \tag{6}$$

Proof: First, we show that the error e_1 appearing in Eq. (5) is distributed uniformly over $\frac{1}{q} \times \{-\lfloor q/2 \rfloor, \cdots, \lfloor q/2 \rfloor\}$. Note that, we fix the secret $\mathbf{s}$ which is sampled from some probability distribution χ_s beforehand. It is true that if we take $\mathbf{a} \in \mathbb{Z}_q^n$ uniformly at random then $(\langle \mathbf{a}, \mathbf{s} \rangle \bmod q)$ is also uniform over $\mathbb{Z}_q$. Hence, $\frac{p}{q} \times (\langle \mathbf{a}, \mathbf{s} \rangle \bmod q)$ is also uniform over

$$\frac{p}{q} \cdot \mathbb{Z}_q = \{0, p/q, \cdots, p \cdot (q-1)/q\}.$$

Suppose that $\langle \mathbf{a}, \mathbf{s} \rangle \bmod q = k$, for some $k \leftarrow \mathcal{U}(\{0, \cdots, q-1\})$. There always exist integers w and v such that $kp = qw + v$, $-\lfloor q/2 \rfloor \leq v \leq \lfloor q/2 \rfloor$, and $0 \leq w \leq p$. Certainly, v is uniform over the set $\{-\lfloor q/2 \rfloor, \cdots, \lfloor q/2 \rfloor\}$. Thus, $(p/q) \cdot (\langle \mathbf{a}, \mathbf{s} \rangle \bmod q) = kp/q = w + v/q$, where

$$-\lfloor q/2 \rfloor / q \leq v/q \leq \lfloor q/2 \rfloor / q.$$

Consequently, $\lfloor (p/q) \cdot (\langle \mathbf{a}, \mathbf{s} \rangle \bmod q) \rceil = w \in \{0, \cdots, p\}$, and hence

$$e_1 := \frac{p}{q}(\langle \mathbf{a}, \mathbf{s} \rangle \bmod q) - \left\lfloor \frac{p}{q}(\langle \mathbf{a}, \mathbf{s} \rangle \bmod q) \right\rceil = \frac{v}{q}$$

is uniform over $(1/q) \cdot \{-\lfloor q/2 \rfloor, \cdots, \lfloor q/2 \rfloor\}$. As a result, $(q/p) \times e_1 = v/p$ is uniform over $\{-\lfloor q/2 \rfloor / p, \cdots, \lfloor q/2 \rfloor / p\}$.

Recall that, the q-error $e = \lfloor (q/p) \cdot e_1 \rceil$. Applying Lemma 1 to $b := \lfloor q/2 \rfloor / p$, $B := \{-\lfloor b \rfloor, -\lfloor b \rfloor + 1, \cdots, \lfloor b \rfloor - 1, \lfloor b \rfloor\}$ and $x := (q/p) \cdot e_1$ and $e := \lfloor x \rceil$, the theorem follows. The variance of e is computed using the discrete uniform distribution over $B \subset \mathbb{Z}$ as in Subsection 2.4.1. □

Remark 1. *In Eq.* (6) *we will have the equality, i.e.,* $\sigma_{lwr}^2 = (q^2 + 2qp)/(12p^2)$ *if* q *and* p *are power of two. This is the case used in Round5 [15], SABER [32] for computational efficiency. In this case, our formula returns the more exact result (and identical to the variance of the discrete uniform distribution over an interval, see Subsection* 2.4*) in comparison with the formula proposed by [33] saying that* σ_{lwr}^2 *should be computed by* $((q/p)^2 - 1)/12$ *which makes* σ_{lwr}^2 *smaller than it is.*

3.2 Estimating the Successful Range for the Decoding Attack in Solving LWR

Our purpose in this section is to find a condition of q so as to the $\text{LWR}_{m,n,q,p,\chi_s}$ instance can be solved by the decoding attack. The condition depends only on the dimension n, the used LBR algorithm $\mathcal{A}$ (through its constant $c_{\mathcal{A}}$ defined as in Eq. (3)) as well as the bit ratio between q and p. According to the q-reduction, we say that a search-$\text{LWR}_{m,n,q,p,\chi_s}$ instance is solvable by the decoding attack if the corresponding LWE modulo q can be solved by the strategy.

Let $(\mathbf{a}, c = \lfloor \langle \mathbf{a}, \mathbf{s} \rangle \rceil_{q,p}) \in \mathbb{Z}_q^n \times \mathbb{Z}_p$ be an $\text{LWR}_{m,n,q,p,\chi_s}$ instance and its corresponding LWE instance computed by (5), the q-error e has variance σ_{lwr} defined as in (6). By Heuristic 1 (see Subsection 5.2), we need $2\sigma_{lwr} \leq c_{\mathcal{A}}^m \cdot q^{(m-n)/m}$ to happen with probability 1. Notice that $\sigma_{lwr} = \sqrt{(q^2 + 2qp)/(12p^2)} \leq q/(2p)$ as $pq \leq q^2$, yielding that $2\sigma_{lwr} \leq q/p$. To estimate the successful range, we should consider the following slightly stronger condition

$$\frac{q}{p} \leq c_{\mathcal{A}}^m \cdot q^{(m-n)/m},$$

from which we obtain $q^n/p^m \leq c_{\mathcal{A}}^{m^2}$. Given $0 < \zeta < 1$ such that $p = q^\zeta$, from the previous equation we get

$$(m\zeta - n)\log(q) \geq -m^2 \log(c_{\mathcal{A}}). \tag{7}$$

It is easy to see that Eq. (7) just makes sense as long as $m > \frac{n}{\zeta}$. So with this condition, we can rewrite (7) as

$$\log(q) \geq -m^2 \log(c_{\mathcal{A}})/(m\zeta - n).$$

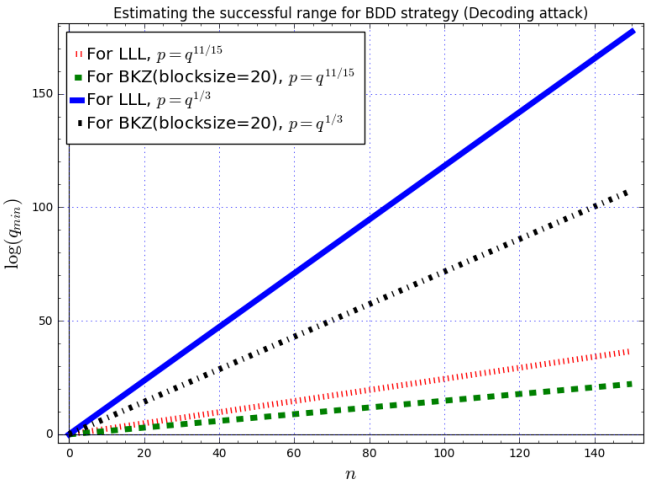

Fig. 1: We plot the graphs of the lines $\log(q_{\text{min}}) = \lceil -4n\log(c_{\mathcal{A}})/\zeta^2 \rceil$ corresponding to $\zeta = 11/15$ and $\zeta = 1/3$. For LLL, we use $c_{\text{LLL}} = 0.9775$ and for BKZ of blocksize 20 we use $c_{\text{BKZ}} = 0.9868$.

Hence we have that

$$\log(q) \geq -4n\log(c_{\mathcal{A}})/\zeta^2,$$

since

$$\min_m \left\{ -m^2 \log(c_{\mathcal{A}})/(m\zeta - n) \right\} = -4n\log(c_{\mathcal{A}})/\zeta^2$$

obtained at $m = 2n/\zeta$. For given ζ, let q_{min} be the integer such that

$$\log(q_{\text{min}}) = \left\lceil \frac{-4n\log(c_{\mathcal{A}})}{\zeta^2} \right\rceil, \tag{8}$$

then $\log(q_{\text{min}})$ is a function in n whose graph is a straight line (called *the boundary line*). The line divides the plane into two half-planes: the upper half-plane indicates the successful range in which LWR instances are solvable by the decoding attack, while the lower half-plane indicates the failure range in which LWR instances are unsolvable by the decoding attack, (see Figure 1).

Remark 2. *It is easy to see from Eq.* (8) *that either* n *grows or/and* ζ *decreases makes the value* $\log(q_{min})$ *increase. This seems to mean that large* n *and/or smaller* ζ *should be chosen for LWR-based cryptosystems. However, as we will see later in our experiments, larger* n *and/or smaller* ζ *provide our modulus switching approach with more advantages.*

The Optimal Number of LWR Samples for the Decoding Attack: The optimal number of LWR samples m should be chosen such that the right-hand side of $2\sigma \leq c_{\mathcal{A}}^m q^{\frac{m-n}{m}}$ is maximum. So the optimal value of m should be:

$$m = \left\lfloor \sqrt{\frac{n\log(q)}{-\log(c_{\mathcal{A}})}} \right\rceil. \tag{9}$$

Remind that, the optimal number of samples typically used in attacking LWE problems (e.g., see [34], [10]) is $m = \left\lfloor \sqrt{n\log(q)/\log(\delta_{\mathcal{A}})} \right\rceil$, which along with Eq. (9) again convince us that $\delta_{\mathcal{A}} \approx 1/c_{\mathcal{A}}$ (see Subsection 2.3).

4 Modulus Switching on LWE and LWR

4.1 Modulus Switching on LWE

Let $(\mathbf{a}, c = \langle \mathbf{a}, \mathbf{s} \rangle + e \bmod q) \in \mathbb{Z}_q^n \times \mathbb{Z}_q$ be an LWE modulo q sample where $\mathbf{s}$ follows some distribution χ_s of variance σ_s^2 and the secret e is sampled from a distribution of variance σ_{lwe}^2. Using the modulus switching, we consider the LWE instance $(\tilde{\mathbf{a}}, \tilde{c}) := (\lfloor (q'/q).\mathbf{a} \rceil, \lfloor (q'/q).c \rceil) \in \mathbb{Z}_{q'}^n \times \mathbb{Z}_{q'}$ where we can write $\tilde{c}$ as $\tilde{c} = \langle \lfloor \tilde{\mathbf{a}} \rceil, \mathbf{s} \rangle \bmod q' + \tilde{e}$, with $\tilde{e} := e_2 + (q'/q).e + e_3$, $e_2 := \langle (q'/q)\mathbf{a} - \lfloor (q'/q)\mathbf{a} \rceil, \mathbf{s} \rangle \bmod q'$ and $e_3 \in (-1/2, 1/2]$. The variance of e_3 is $\sigma_3^2 = 1/12$. The variance σ_2^2 of e_2 can be approximated as sum of n summands in which each summand is uniform on $(-\sigma_s/2, \sigma_s/2]$ where σ_s is the variance of the secret $\mathbf{s}$. Hence $\sigma_2^2 = n\sigma_s^2/12$. Assume that e_2, e_3 and $(q'/q) \cdot e$ are three independent random variables, the variance of $\tilde{e}$ can be estimated as

$$\sigma'^2_{lwe} = \frac{(n\sigma_s^2 + 1)}{12} + \frac{\sigma_{lwe}^2}{q^2} q'^2 \approx \frac{1}{12}(M_{lwe} q'^2 + N), \quad (10)$$

with $M_{lwe} = 12\sigma_{lwe}^2/q^2$ and $N = n\sigma_s^2$. Particularly, if $\chi_e = \mathcal{D}_{\mathbb{Z},\alpha q}$ then $\sigma_{lwe}^2 = \alpha^2 q^2/(2\pi)$ and $M_{lwe} = 6\alpha^2/\pi$.

Recall that, in the literature, one typically chooses q' such that $|e_2| \approx (q'/q)|e|$ yielding

$$q' \approx \frac{\sigma_{\mathbf{s}}}{\sigma_{lwe}} \sqrt{\frac{n}{12}} q$$

(cf. [11]).

4.2 Modulus Switching on LWR

We will analyze the so-called q'*-reduction* which reduces an $\mathrm{LWR}_{m,n,q,p,\chi_s}$ instance to an LWE modulo q' instance. Let $(\mathbf{a}, c = \lfloor \langle \mathbf{a}, \mathbf{s} \rangle \rceil_{q,p}) \in \mathbb{Z}_q^n \times \mathbb{Z}_p$ be an LWR sample. We reduce this LWR sample to the LWE sample modulo q' of the form $(\mathbf{a}', c' = \langle \mathbf{a}', \mathbf{s} \rangle + e') \in \mathbb{Z}_{q'}^n \times \mathbb{Z}_{q'}$ with $c' = \lfloor (q'/q) \cdot c \rceil$ clarified below where $\mathbf{a}' = \lfloor (q'/q) \cdot \mathbf{a} \rceil$. We call the error e' q'*-error*. We now take a closer look into the process of generating e'. Recall that, $c = \lfloor \langle \mathbf{a}, \mathbf{s} \rangle \rceil_{q,p} = (p/q) \cdot \langle \mathbf{a}, \mathbf{s} \rangle + p \cdot u + e_1$ for some $u \in \mathbb{Z}$, where $e_1 \leftarrow (1/q) \times \mathcal{U}(T)$ with $T = \{-\lfloor q/2 \rfloor, \cdots, \lfloor q/2 \rfloor\}$ (see Eq. (5) and the proof of Theorem 1).

Now with the q'-reduction we will obtain

$$c_2 := \frac{q'}{p} \cdot c = \left\langle \left\lfloor \frac{q'}{q} \mathbf{a} \right\rceil, \mathbf{s} \right\rangle + \left\langle \frac{q'}{q}\mathbf{a} - \left\lfloor \frac{q'}{q}\mathbf{a} \right\rceil, \mathbf{s} \right\rangle + q' \cdot u + e_4,$$

where $e_4 := (q'/p) \cdot e_1 \leftarrow q'/(pq) \cdot \mathcal{U}(T)$, with e_1 as in Eq. (5). Hence

$$\begin{aligned} c' :&= \lfloor c_2 \rceil \bmod q' \\ &= \left\langle \left\lfloor \frac{q'}{q}\mathbf{a} \right\rceil, \mathbf{s} \right\rangle \bmod q' + \left\langle \frac{q'}{q}\mathbf{a} - \left\lfloor \frac{q'}{q}\mathbf{a} \right\rceil, \mathbf{s} \right\rangle + e_4 + e_6, \end{aligned}$$

where $e_6 \in [-1/2, 1/2)$. The q'-error is $e' := e_4 + e_5 + e_6$, with $e_5 := \langle (q'/q)\mathbf{a} - \lfloor (q'/q)\mathbf{a} \rceil, \mathbf{s} \rangle$.

4.2.1 Distribution of q'-Error e':

The behaviour of the q'-error is mainly affected by that of e_4 (following a uniform distribution (cf. Section 3.1)) and e_5 (following a Gaussian distribution via Central Limit Theorem (cf. [11, Lemma 2])). Then, using the Eq. (4) on the convolution of two distributions, the probability density function of

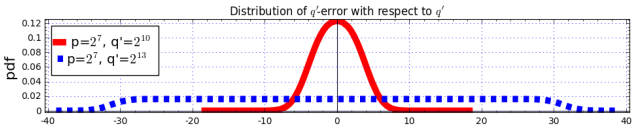

Fig. 2: Distribution of the q'-error with respect to q' given p. We consider two cases: $(p, q') = (2^7, 2^{10})$ and $(p, q') = (2^7, 2^{13})$. In the first case, the graph of pdf is in bell shape like the graph of Gaussian pdf while in the second case the graph consists of a long straight segment in its middle.

e' can be approximated by

$$\begin{aligned} f(y) &= \frac{p\sqrt{6}}{q'\sqrt{\pi n \sigma_s^2}} \int_{-\frac{q'}{2p}}^{\frac{q'}{2p}} \exp\left(-\frac{6(x-y)^2}{n\sigma_s^2}\right) dx \\ &= \frac{p}{q'\sqrt{\pi}} \int_{\frac{\sqrt{6}\left(-\frac{q'}{2p}-y\right)}{\sqrt{n}\sigma_s}}^{\frac{\sqrt{6}\left(\frac{q'}{2p}-y\right)}{\sqrt{n}\sigma_s}} \exp(-\zeta^2) d\zeta \\ &= \frac{p}{2q'} \cdot \left[\mathrm{erf}\left(\frac{\sqrt{6}\left(\frac{q'}{2p} - y\right)}{\sqrt{n}\sigma_s} \right) + \mathrm{erf}\left(\frac{\sqrt{6}\left(\frac{q'}{2p} + y\right)}{\sqrt{n}\sigma_s} \right) \right]. \end{aligned}$$

Its derivative is

$$\begin{aligned} f'(y) = &\frac{\sqrt{6}p}{2\sqrt{\pi n} q' \sigma_s} \times \\ &\times \left[-\exp\left(\frac{-6\left(\frac{q'}{2p} - y\right)^2}{n\sigma_s^2} \right) + \exp\left(\frac{-6\left(\frac{q'}{2p} + y\right)^2}{n\sigma_s^2} \right) \right]. \end{aligned}$$

The function $f(y)$ is symmetric through origin and its graph is a convex bell-shaped curve reaching its highest value $h(q') = (p/q') \times \mathrm{erf}\left((\sqrt{6}q')/(2p\sqrt{n}\sigma_s)\right)$ at $y = 0$. It is easy to see that the functions $f'(y)$ and $h(q')$ tend to 0 as q' increases. Thus, if $q' \gg p$, the error e' will tend to follow a uniform distribution. By contrast, when $q' \approx p$, the error e' will tend to be distributed via a Gaussian distribution (see Figure 2 for an illustration).

The behavior of the q'-error e' is complex, therefore we cannot use Eq. (13) or Eq. (14) to estimate the success probability of the decoding attack in solving the LWE modulo q'. Also, on the other hand, we cannot compare the success probability of the decoding attack on the LWE modulo q' with that on the LWE modulo q using the formula (13) or (14). This is why we need to use Heuristic 1 (see Subsection 5.2).

4.2.2 Variance of q'-Error:

We consider the variances of e_4, e_5 and e_6. The variance of e_4 will be

$$\sigma_4^2 := \left(q'^2/(q^2p^2)\right) \cdot \left(\left((2\lfloor q/2 \rfloor + 1)^2 - 1\right)/12\right).$$

Same as the case of LWE, we assume that e_4, e_5 and e_6 are three independent random variables, then the variance of e' is estimated by

$$\begin{aligned} \sigma'^2_{lwr} &\approx \frac{1}{12}\left(n\sigma_s^2 + \frac{q^2 + 2q}{q^2p^2} \cdot q'^2 + 1 \right) \\ &\approx \frac{1}{12}\left(M_{lwr} \cdot q'^2 + N \right), \end{aligned} \quad (11)$$

where $M_{lwr} := (q^2 + 2q)/(p^2q^2)$ and $N := n\sigma_s^2$. Notice that, we can approximate

$$M_{lwr} \approx \frac{12\sigma_{lwr}^2}{q^2}, \quad (12)$$

with σ_{lwr}^2 as in Eq. (6).

5 Known Attacks and Optimally Choosing the Switching Modulo q'

In this section, we will give the optimal formula for the switching modulus q' according to each attack. Furthermore, we will also provide a corresponding sufficient condition under which one is able to decide whether one should use the modulus switching on the LWE/LWR instances or not. We focus on three typical attacks: the decoding attack, the dual attack and the primal attack.

5.1 Our Framework and Notations

For convenient, we will establish a common framework for both LWE and LWR under the modulus switching. More specifically, we consider two LWE versions: one is the LWE modulo q, the other is the LWE modulo q'. The former is with respect to the original LWE modulo q and the LWE modulo q obtained from reducing LWR modulo q to the LWE modulo q without using the modulus switching. The latter is the LWE instance obtained by applying the modulus switching to the LWE modulo q and to the LWR modulo (q, p). More specifically,

• *Without the modulus switching:* In this case, the LWE modulo q instance we deal with consists of m LWE samples of the form $(\mathbf{a}, c) \in \mathbb{Z}_q^n \times \mathbb{Z}_q$ where $c = \langle \mathbf{a}, \mathbf{s} \rangle \bmod q + e$ and the error e has variance of σ^2. We call each specific attack against this LWE instance *the q-attack*, for instances the q-decoding attack, the q-dual attack and the q-primal attack.
• *With the modulus switching:* In this case, the LWE modulo q' instance we deal with consists of m LWE samples of the form $(\mathbf{a}', c') \in \mathbb{Z}_{q'}^n \times \mathbb{Z}_{q'}$ where $c' = \langle \mathbf{a}', \mathbf{s} \rangle \bmod q' + e'$ and $\mathbf{a}' = \lfloor (q'/q) \cdot \mathbf{a} \rceil$, and the q'-error e' has variance of $\sigma'^2 = \left(Mq'^2 + N\right)/12$. We call each specific attack against this LWE instance *the q'-attack*, for instances the q'-decoding attack, the q'-dual attack and the q'-primal attack.

Remark 3. *From now on, we abuse the notations M, N, σ, σ' without caring about which original problem (the LWE modulo q or the LWR modulo (q, p)) we are dealing with. In particular, for LWR, we will replace $M = 12\sigma^2/q^2$ with $M_{lwr} = 12\sigma_{lwr}^2/q^2$, σ^2 with $\sigma_{lwr}^2 = (q^2 + 2qp)/(12p^2)$ and σ'^2 with σ'^2_{lwr} as in Eq.* (11)*. Likewise, for LWE, M will be replaced with $M_{lwe} = 12\sigma_{lwe}^2/q^2$, σ^2 will be replaced with σ_{lwe}^2, and σ'^2 with σ'^2_{lwe} as in Eq.* (10)*. Still, $N = n\sigma_s^2$ is the same for both LWE and LWR.*

In what follows, according to each attack, we try to choose q' such that the q'-attack is "*optimal*" in the sense that the success probability of this attack on LWE modulo q' is highest. Then, we find a condition under which the q'-attack is more powerful than the q-attack.

We assume that the number of samples m is kept the same for both the q-attack and the q'-attack. Note that, for each attack, the optimal m will be chosen corresponding to the LWE modulo q. Such a choice of m will be theoretically optimal to the q-attack not the q'-attack. However, this implies that the q'-attack can be stronger if one appropriately choose for it the optimal m.

5.2 The Decoding Attack

The decoding attack proposed by Lindner and Peikert [34] is based on the close relation between the search-LWE problem and the BDD problem. Given a lattice and a target vector *unusually close* to the lattice, the BDD problem asks to find the lattice vector closest to the target.

Let $(\mathbf{A}, \mathbf{c} = \mathbf{A}\mathbf{s} + \mathbf{e} \mod q) \in \mathbb{Z}_q^{m \times n} \times \mathbb{Z}_q^m$ be an LWE modulo q instance. We set

$$\Lambda_q(\mathbf{A}) = \left\{ \mathbf{u} \in \mathbb{Z}^m : \mathbf{u} = \mathbf{A}\mathbf{s} \bmod q \text{ for some } \mathbf{s} \in \mathbb{Z}^n \right\}$$

to be the q-ary lattice spanned by columns of $\mathbf{A}$ and we call it *the associated q-ary lattice* of the search-LWE problem $(\mathbf{A}, \mathbf{c})$. If the error $\mathbf{e}$ is sufficiently short then $\mathbf{c}$ is closest to some lattice point $\mathbf{u} = \mathbf{A}\mathbf{s} \bmod q \in \Lambda_q(\mathbf{A})$ since we have $\mathbf{e} = \mathbf{c} - \mathbf{A}\mathbf{s}$. Thus, finding the secret $\mathbf{s}$ is equivalent to finding $\mathbf{u}$, i.e., solving a CVP problem over q-ary lattice $\Lambda_q(\mathbf{A})$. The most basic tools used in solving search-LWE via the decoding attack are some basis reduction algorithm, say $\mathcal{A}$ (e.g., LLL or BKZ), and the Babai's Nearest Plane (BNP) algorithm. The BNP algorithm takes as input the vector $\mathbf{c}$ and a basis $\mathbf{B} = \{\mathbf{b}_1, \cdots, \mathbf{b}_m\}$ of $\Lambda_q(\mathbf{A})$ that is already reduced by $\mathcal{A}$ and outputs the lattice point $\mathbf{u} \in \Lambda_q(\mathbf{A})$ such that $\mathbf{e} = \mathbf{c} - \mathbf{u} \in \mathcal{P}_{1/2}(\mathbf{B}^*)$.

From now on, by "the decoding attack", we mean the solving strategy associated with the BNP algorithm described above. Note also that, the decoding attack needs $m > n$, and the optimal m is approximated by Eq. (9).

The Decoding Attack and Root Hermite Factor: It is conventional that the quality of a reduced basis (which is characterized by the rHF obtained using by some LBR algorithm) has the most significant effect on the success probability of the BNP algorithm (see, e.g., [11, Section 5.4], [35, 36]), hence the decoding attack. Namely, a smaller rHF means that the corresponding basis is reduced better, hence the BNP algorithm may return the closest vector more precisely, so the efficacy of the decoding attack may be higher. Assume for example that the decoding attack deploys the LBR algorithm named $\mathcal{A}$. Also, suppose that we want to compare the efficacy of the decoding attack in solving a search-LWE problem $(\mathbf{A}_1, \mathbf{c}_1)$ with that in solving a search-LWE problem $(\mathbf{A}_2, \mathbf{c}_2)$. Then instead of success probability, we can compare the rHFs of $\mathcal{A}$ with respect to the reduced bases of the associated q-ary lattices $\Lambda_q(\mathbf{A}_1)$ and $\Lambda_q(\mathbf{A}_2)$.

Success Probability of the Decoding Attack: The success probability of the decoding attack in solving search-LWE can be measured by the probability of the event that the error $\mathbf{e}$ lies in $\mathcal{P}_{1/2}(\mathbf{B}^*)$. Depending on which distribution the error $\mathbf{e}$ follows, we have some formulas to compute the probability in the literature: (i) if $\mathbf{e}$ is uniform then we can estimate the probability by

$$\Pr\left[\mathbf{e} \in \mathcal{P}_{1/2}(\mathbf{B}^*)\right] = \prod_{i=1}^{m} \left(\frac{\rceil \|\mathbf{b}_i^*\|}{2\sigma_e\sqrt{3}} \right), \tag{13}$$

(ii) in the case of a Gaussian error $\mathbf{e}$, we can use the formula taken from [34]

$$\Pr\left[\mathbf{e} \in \mathcal{P}_{1/2}(\mathbf{B}^*)\right] = \prod_{i=1}^{m} \mathrm{erf}\left(\frac{\rceil \|\mathbf{b}^*{}_i\|}{2\sigma_e\sqrt{2}} \right), \tag{14}$$

where σ_e^2 is the variance of the error $\mathbf{e}$ according to its distribution and $\mathrm{erf}(\cdot)$ is the Gaussian error function

$$\mathrm{erf}(z) = \frac{2}{\sqrt{\pi}} \int_0^z \exp(-t^2) dt, z \in [0, +\infty].$$

However, Eq. (13) and Eq. (14) are really not helpful for our work. We cannot use either Eq. (13) or Eq. (14) if the error $\mathbf{e}$ has a complex behavior. Such a kind of error is the q'-error that we saw in Section 4. Therefore, we have to look for another way to estimate the success probability for the decoding attack regardless of the error's distribution. Fortunately, we can use a heuristic analysis appeared in [26] as follows:

We have

$$\Pr\left[\mathbf{e} \in \mathcal{P}_{1/2}(\mathbf{B}^*)\right] = \Pr\left[|\langle \mathbf{e}, \mathbf{b}_i^* \rangle| < \frac{\|\mathbf{b}_i^*\|^2}{2}, \forall i = 1, \cdots, m \right].$$

Using the heuristics that $|\langle \mathbf{e}, \mathbf{b}_i^* \rangle| \approx \|\mathbf{e}\| \cdot \|\mathbf{b}_i^*\|/\sqrt{m}$ and $\|\mathbf{e}\| \approx \sigma_e \cdot \sqrt{m}$, we have $2\sigma_e \leq \|\mathbf{b}_i^*\|$ for all $i = 1, \cdots, m$, which is

equivalent to

$$2\sigma_e \leq \min_{i=1,\cdots,m} \|\mathbf{b}_i^*\|. \quad (15)$$

Combining Eq. (15) with Eq. (3) yields the following heuristic that will be very useful for our analysis on the decoding attack:

Heuristic 1: Let $c_{\mathcal{A}}$ is defined as in Eq. (3)*. Heuristically, the success probability for the decoding attack in solving search-LWE problem $(\mathbf{A}, \mathbf{c} = \mathbf{As} + \mathbf{e} \mod q) \in \mathbb{Z}_q^{m\times n} \times \mathbb{Z}_q^m$ can be measured by the probability $\Pr\left[2\sigma_e \leq c_{\mathcal{A}}^m \cdot \det(\Lambda_q(\mathbf{A}))^{1/m}\right]$, namely, $\Pr\left[2\sigma_e \leq c_{\mathcal{A}}^m \cdot q^{(m-n)/m}\right]$, which is equivalent to*

$$\Pr\left[\frac{q^{m-n}}{\sigma_e^m} \geq \frac{2^m}{c_{\mathcal{A}}^{m^2}}\right],$$

where $\mathcal{A}$ is the LBR algorithm used within the decoding attack.

According to the heuristic, the success probability of the q-decoding attack is

$$\Pr\left[\frac{q^{m-n}}{\sigma^m} \geq \frac{2^m}{c_{\mathcal{A}}^{m^2}}\right].$$

Similarly, that of the q'-decoding attack is

$$\Pr\left[\frac{q'^{m-n}}{\sigma'^m} \geq \frac{2^m}{c_{\mathcal{A}}^{m^2}}\right].$$

Set $P' := q'^{m-n}/\sigma'^m$ and $P := q^{m-n}/\sigma^m$. Our goal is to choose q' maximizing P' and with the chosen q', we check the condition when $P' \geq P$. In other words, we choose q' such that

$$P' \text{ is maximum, (denote } P'_{\max}),$$

and $P'_{\max} \geq P$. The condition $P' \geq P$ is equivalent to

$$\frac{q'^{m-n} \cdot \sqrt{12^m}}{\sqrt{(Mq'^2 + N)^m}} \geq \frac{q^{m-n}}{\sigma^m}. \quad (16)$$

Now, it is the time to state the main result for applying the modulus switching to the decoding attack.

Theorem 2. *Let $m, n, M, N, q, q', \sigma, \sigma', \sigma_s$ as mentioned in Subsection* 5.1 *and required that $m > n$. Assume that we apply the modulus switching technique to the decoding attack against the LWE/LWR problems. Then*

(i) The optimal switching modulus q' for the decoding attack is

$$q' \approx \sqrt{\frac{(m-n)N}{nM}}. \quad (17)$$

(ii) The q'-decoding attack is stronger than the q-decoding attack if the following sufficient condition holds:

$$\left(\frac{12\sigma^2}{\sigma_s^2}\right)^n \geq \frac{m^m}{(m-n)^{m-n}}. \quad (18)$$

Proof:

It is easy to see that $q' = \sqrt{(m-n)N/(nM)}$ maximizes $h(q') := q'^{m-n}\sqrt{12^m}/\sqrt{(Mq'^2 + N)^m}$. Then the maximum of $h(q')$ is

$$\sqrt{\frac{(m-n)^{m-n} \cdot n^n \cdot 12^m}{m^m \cdot N^n \cdot M^{m-n}}}.$$

After some calculations, we get from Eq. (16) that

$$\left(\frac{12\sigma^2}{m}\right)^m \geq \left(\frac{N}{n}\right)^n \cdot \left(\frac{q^2 \cdot M}{m-n}\right)^{m-n}. \quad (19)$$

By Remark 3 and some arrangements, Eq. (19) can be rewritten as Eq. (18).

□

Remark 4. *We remark that the statement (ii) of Theorem* 2 *differs from [18, Theorem 2 (ii)]. In the effort of simplifying the Eq.* (19)*, we made a mistake to claim in the proof of [18, Theorem 2 (ii)] that the function $x\ln(x)$ is concave over $(0, +\infty)$. Actually, that claim is not correct and simplifying is unnecessary. In the proof for Theorem* 2 *(ii) of the present paper, we do not use that claim and only transform Eq.* (19) *into Eq.* (18) *using Remark* 3*.*

5.3 The Dual Attack

The dual attack aims to solving the decision version of LWE and LWR. Namely, to distinguish an LWE instance $(\mathbf{A}, \mathbf{c}) \in \mathbb{Z}_q^{m\times n} \times \mathbb{Z}_q^m$ with the uniform $(\mathbf{A}, \mathbf{c}) \in \mathbb{Z}_q^{m\times n} \times \mathbb{Z}_q^m$, the attacker considers the q-ary lattice

$$\Lambda_q^{\perp}(\mathbf{A}) = \left\{(\mathbf{x}, \mathbf{y}) \in \mathbb{Z}^m \times \mathbb{Z}^n : \mathbf{x}^t \cdot \mathbf{A} = \mathbf{y}^t \bmod q\right\},$$

and then uses an LBR algorithm, again named $\mathcal{A}$, to get a reduced basis of the lattice and then get a short vector, let's say $\mathbf{u}_0 = (\mathbf{x}_0, \mathbf{y}_0)$. Notice that if $(\mathbf{A}, \mathbf{c})$ is uniform then

$$z := \langle \mathbf{x}_0, \mathbf{c}\rangle = \langle \mathbf{y}_0, \mathbf{s}\rangle + \langle \mathbf{x}_0, \mathbf{e}\rangle \bmod q$$

is also uniform modulo q. By contrast, if $(\mathbf{A}, \mathbf{c})$ is LWE, i.e., we have $\mathbf{c} = \mathbf{A} \cdot \mathbf{s} + \mathbf{e} \bmod q$ then z tends to be distributed via a Gaussian distribution of mean 0 and its variance $\sigma_z^2 \approx \|\mathbf{u}_0\|^2\sigma^2$ as n increases, where σ^2 is the variance of the error $\mathbf{e}$. This suggests the attacker to check whether the value z is small or not.

A reasoning in [37, Section 6.4] says that the dual attack has the distinguishing advantage upper bounded by $4\exp(-2\pi^2\sigma_z^2/q^2) = 4\exp(-2\pi^2\|\mathbf{u}_0\|^2\sigma^2/q^2)$. As a result, the success probability of the dual attack depends heavily on the quantity $\|\mathbf{u}_0\|\sigma/q$. The smaller quantity is better for the dual attack. Assume that we try to exploit the modulus switching technique in the dual attack. We should choose the new modulus q' such that the quantity is smallest.

Now we give more details.

- *In the q-dual attack:* The error $\mathbf{e}$ has the variance σ^2. Since we can estimate $\|\mathbf{u}_0\| \approx {\delta_{\mathcal{A}}}^{m+n} q^{n/(m+n)}$ by Eq. (2), we have

$$\frac{\|\mathbf{u}_0\|\sigma}{q} = \frac{{\delta_{\mathcal{A}}}^{m+n} q^{n/(m+n)}\sigma}{q}.$$

- *In the q'-dual attack:* The corresponding error $\mathbf{e}'$ has the variance $\sigma'^2 = (Mq'^2 + N)/12$. Same as above, we can estimate $\|\mathbf{u}'_0\| \approx {\delta_{\mathcal{A}}}^{m+n} q'^{n/(m+n)}$, hence

$$\frac{\|\mathbf{u}'_0\|\sigma'}{q'} = \frac{1}{\sqrt{12}}{\delta_{\mathcal{A}}}^{m+n} q'^{\frac{-m}{m+n}}\sqrt{Mq'^2 + N}.$$

We come to the following theorem for the dual attack giving the optimal switching modulus optimal and a condition to efficiently exploit the modulus switching technique in this attack.

Theorem 3. *Let $m, n, M, N, q, q', \sigma, \sigma', \sigma_s$ as mentioned in Subsection* 5.1*. Assume that we apply the modulus switching technique to the dual attack against the LWE/LWR problems. Then*

(i) The optimal switching modulus q' for the dual attack is

$$q' \approx \sqrt{\frac{mN}{nM}}. \quad (20)$$

(ii) The q'-dual attack is stronger than the q-dual attack if the following sufficient condition holds:

$$\left(\frac{12\sigma^2}{\sigma_s^2}\right)^n \geq \frac{(m+n)^{m+n}}{m^m}. \quad (21)$$

Proof: We should choose q' such that $g(q') := q'^{\frac{-m}{m+n}}\sqrt{Mq'^2+N}$ is minimum. It is easy to see that such a q' is

$$q' = \sqrt{\frac{mN}{nM}}.$$

Then the minimum of $g(q')$ is

$$\sqrt{\left(\frac{mN}{nM}\right)^{-\frac{m}{m+n}} \cdot \frac{(m+n)N}{n}}.$$

Now, we suppose that $\|\mathbf{v'}_0\|\sigma'/q' \leq \|\mathbf{v}_0\|\sigma/q$, which is equivalent to

$$\left(\frac{mN}{nM}\right)^{-\frac{m}{m+n}} \cdot \frac{(m+n)N}{n} \cdot \frac{1}{12} \leq \frac{q^{2n/(m+n)}\sigma^2}{q^2},$$

i.e.,

$$\left(\frac{12\sigma^2}{m+n}\right)^{m+n} \geq \left(\frac{N}{n}\right)^n \cdot \left(\frac{q^2 \cdot M}{m}\right)^m,$$

which can be rewritten as Eq. (21) using Remark 3.

□

Remark 5. *We can see that differing from the decoding attack, we do not need $m > n$ but the arbitrary integer $m > 0$. We also can choose the optimal m for the dual attack by finding m minimizing the quantity ${\delta_{\mathcal{A}}}^{m+n} q^{n/(m+n)}$ given $n, q, \delta_{\mathcal{A}}$. Namely,*

$$m = \left\lfloor \sqrt{\frac{n\log(q)}{\log(\delta_{\mathcal{A}})}} \right\rceil - n. \tag{22}$$

5.4 The Primal Attack

The primal attack solves the search-LWE problem by transforming the problem into an Unique Shortest Vector Problem (uSVP). Let $\gamma > 1$ be a real number, the uSVP$_\gamma$ problem is to find the shortest non-zero vector of a lattice $\mathcal{L}$ given that the gap $\lambda_2(\mathcal{L})/\lambda_1(\mathcal{L}) \geq \gamma$. It is folklore that the bigger the gap $\lambda_2(\mathcal{L})/\lambda_1(\mathcal{L})$ is, the easier the uSVP is. (See more discussions in [25, 38].)

Let $(\mathbf{A}, \mathbf{c} = \mathbf{A}\cdot\mathbf{s}+\mathbf{e}) \in \mathbb{Z}_q^{m\times n} \times \mathbb{Z}_q^m$ be an LWE instance, we have $\mathbf{A}\cdot\mathbf{s}+\mathbf{e}-\mathbf{c} = \mathbf{0} \bmod q$. Consider the q-ary lattice

$$\Lambda = \left\{\mathbf{v} \in \mathbb{Z}^{n+m+1} : (\mathbf{A}|\mathbf{I}_m|-\mathbf{c})\mathbf{v} = \mathbf{0} \bmod q\right\}.$$

Then $\mathbf{v}_0 = (\mathbf{s}|\mathbf{e}|\mathbf{1})$ will be an unusual short vector in the lattice if $\mathbf{s}$ and $\mathbf{e}$ are short. It is known that if the gap between two minima $\lambda_2(\Lambda)$ and $\lambda_1(\Lambda)$ is sufficiently large then the short vector $\mathbf{v}_0$ may be found using an LBR algorithm. Recall that we can estimate $\lambda_2(\Lambda) = \sqrt{(m+n+1)/(2\pi\mathfrak{e})}\cdot q^{\frac{m}{m+n+1}}$ by Gaussian Heuristic, whereas $\lambda_1(\Lambda) = \|(\mathbf{s}|\mathbf{e}|\mathbf{1})\| = \sqrt{\|\mathbf{s}\|^2+\|\mathbf{e}\|^2+1} \approx \sqrt{n\sigma_s^2+1+m\sigma^2} \approx \sqrt{N+m\sigma^2}$.

Now if we use the modulus switching technique on the LWE, we attain a new LWE instance $(\mathbf{A}', \mathbf{c}' = \mathbf{A}'\cdot\mathbf{s}+\mathbf{e}') \in \mathbb{Z}_{q'}^{m\times n} \times \mathbb{Z}_{q'}^m$ where $\mathbf{A}' = \lfloor (q'/q)\cdot\mathbf{A}\rceil$, hence a corresponding q'-ary lattice.

$$\Lambda' = \left\{\mathbf{v}' \in \mathbb{Z}^{n+m+1} : (\mathbf{A}'|\mathbf{I}_m|-\mathbf{c}')\mathbf{v}' = \mathbf{0} \bmod q'\right\}.$$

Same as above, the shortest vector of the q'-ary lattice is $\mathbf{v}'_0 = (\mathbf{s}|\mathbf{e}'|\mathbf{1})$. Accordingly, $\lambda_2(\Lambda') = \sqrt{(m+n+1)/(2\pi\mathfrak{e})}\cdot q'^{\frac{m}{m+n+1}}$ and

$$\begin{aligned}\lambda_1(\Lambda') &= \|(\mathbf{s}|\mathbf{e}'|\mathbf{1})\| = \sqrt{\|\mathbf{s}\|^2+\|\mathbf{e}'\|^2+1} \\ &\approx \sqrt{n\sigma_s^2+m\sigma'^2} = \sqrt{Pq'^2+Q},\end{aligned}$$

where

$$Q = (m+12)N/12, \sigma'^2 = (Mq'^2+N)/12, P = mM/12. \tag{23}$$

We have the following theorem:

Theorem 4. *Let $m, n, M, N, q, q', \sigma, \sigma', \sigma_s$ as mentioned in Subsection 5.1. Assume that we apply the modulus switching technique to the primal attack against the LWE/LWR problems. Then*

(i) The optimal switching modulus q' for the primal attack is

$$q' \approx \sqrt{\frac{(m+12)N}{(n+1)M}}. \tag{24}$$

(ii) The q'-primal attack is stronger than the q-primal attack if the following sufficient condition holds:

$$\left(\frac{12(N+m\sigma^2)}{m+n+1}\right)^{m+n+1} \geq \left(\frac{(m+12)N}{n+1}\right)^{n+1} \cdot \left(q^2M\right)^m. \tag{25}$$

Proof: We will choose q' optimal such that the ratio

$$\frac{\lambda_2(\Lambda')}{\lambda_1(\Lambda')} = \frac{\sqrt{\frac{m+n+1}{2\pi\mathfrak{e}}}\cdot q'^{\frac{m}{m+n+1}}}{\sqrt{Pq'^2+Q}},$$

is maximum, i.e., the function $f(q') := \frac{q'^{\frac{m}{m+n+1}}}{\sqrt{Pq'^2+Q}}$ is maximum. It is easy to obtain that such a q' is

$$q' = \sqrt{\frac{yQ}{(1-y)P}} = \sqrt{\frac{(m+12)N}{(n+1)M}},$$

with $y := \frac{m}{m+n+1}$ and the maximum of $f(q')$ is

$$\sqrt{\frac{y^y Q^{y-1}}{(1-y)^{y-1}P^y}}.$$

Now, we need

$$\max\left(\frac{\lambda_2(\Lambda')}{\lambda_1(\Lambda')}\right) \geq \frac{\lambda_2(\Lambda)}{\lambda_1(\Lambda)},$$

which is

$$\begin{aligned}&\frac{(n+1)^{n+1}\cdot 12^{m+n+1}}{(m+n+1)^{m+n+1}\cdot(m+12)^{n+1}\cdot N^{n+1}\cdot M^m} \\ &\geq \frac{q^{2m}}{(N+m\sigma^2)^{m+n+1}},\end{aligned}$$

equivalently,

$$\left(\frac{12(N+m\sigma^2)}{m+n+1}\right)^{m+n+1} \geq \left(\frac{(m+12)N}{n+1}\right)^{n+1} \cdot \left(q^2M\right)^m.$$

□

Remark 6. *Same as the dual attack, we do not need $m > n$. And we can choose the optimal m satisfying that $m + n + 1 = \left\lfloor \sqrt{\frac{n \log(q)}{\log(\delta_{\mathcal{A}})}} \right\rceil$, i.e.,*

$$m = \left\lfloor \sqrt{\frac{n \log(q)}{\log(\delta_{\mathcal{A}})}} \right\rceil - n - 1, \tag{26}$$

by using the arguments in [22, Section 4.2].

6 The Modulus Switching vesus the Rescaling

In this section, we will revisit the dual attack and the primal attack that were presented in Subsection 5.3 and Subsection 5.4. However, we will focus only on the LWE instances $(\mathbf{A}, \mathbf{c} = \mathbf{A} \cdot \mathbf{s} + \mathbf{e}) \in \mathbb{Z}_q^{m \times n} \times \mathbb{Z}_q^m$ whose $\|\mathbf{s}\| \ll \|\mathbf{e}\|$. As we will see later in Section 9, Round5 [15] and SABER [32] are two NIST PQC submissions having this characteristic. We will compare the efficacy of the so-called rescaling technique to that of the modulus switching on these attacks. The main idea of the rescaling technique that was proposed in [22, Subsection 6.1] and further analyzed in [15, 30] is to re-balance the contributions of summands relating to the secret and the error in the corresponding context of each attack. As we will see below, this technique changes the volume of the lattice (due to the change of the modulus), and also slightly changes the norm of the shortest vector in the corresponding CVP instance. One expects that the Hermite factor of the problem might be increased and if so the successful range of the lattice attack is widened. That is why the rescaling technique is also considered as a technical solution to enhance the power of the lattice attacks. Apparently, the technique looks like the same as the modulus switching technique. Therefore, we will consider in details the rescaling technique applied to the dual attack and the primal attack and then have a comparison between these two techniques to get a condition under which the modulus switching is better than the rescaling. For convenient, we also refer all notations to Section 5.

The Dual Attack: For the dual attack, using the rescaling technique, instead of the lattice as in Subsection 5.3, we construct the following alternative lattice of the form:

$$\Lambda_\omega(\mathbf{A}) = \left\{ \left(\mathbf{x}, \frac{\mathbf{y}}{\omega} \right) \in \mathbb{Z}^m \times \frac{1}{\omega} \cdot \mathbb{Z}^n : \mathbf{x}^t \cdot \mathbf{A} = \mathbf{y}^t \bmod q \right\}.$$

We call ω *the rescaling factor*. Define $q_\omega = q/\omega$, then $\mathrm{vol}(\Lambda_\omega(\mathbf{A})) = q_\omega^n$. Using an LBR algorithm $\mathcal{A}$ of the root Hermite Factor $\delta_{\mathcal{A}}$, we can find the short vector of the form $\mathbf{u}_{0,\omega} = (\mathbf{x}_0, \mathbf{y}_0/\omega) \in \Lambda_\omega(\mathbf{A})$ and $\|\mathbf{u}_{0,\omega}\| \approx {\delta_{\mathcal{A}}}^{m+n} (q_\omega)^{n/(m+n)}$. The rescaling factor ω will be chosen to equalize the contribution of $\mathbf{s}$ and $\mathbf{e}$, namely we can choose $\omega = \sigma/\sigma_s$. Then $z := \langle \mathbf{x}_0, \mathbf{c} \rangle = \langle \mathbf{y}_0, \omega \mathbf{s} \rangle + \langle \mathbf{x}_0, \mathbf{e} \rangle \bmod q$. Again, we pay our attention to the quantity

$$\frac{\|\mathbf{u}_{0,\omega}\| \sigma}{q} = \frac{{\delta_{\mathcal{A}}}^{m+n} q_\omega^{n/(m+n)} \sigma}{q}.$$

Recall that, in the case of using the modulus switching with the modulus q', we have

$$\frac{\|\mathbf{u}_0'\| \sigma'}{q'} = \frac{1}{\sqrt{12}} \cdot {\delta_{\mathcal{A}}}^{m+n} {q'}^{\frac{-m}{m+n}} \sqrt{M {q'}^2 + N},$$

which is minimum of $\frac{1}{\sqrt{12}} \cdot {\delta_{\mathcal{A}}}^{m+n} \sqrt{\left(\frac{mN}{nM}\right)^{-\frac{m}{m+n}} \cdot \frac{(m+n)N}{n}}$ at $q' = \sqrt{mN/(nM)}$.

So as to compare the modulus switching with the rescaling, we compare $\|\mathbf{u}_{0,\omega}\| \sigma/q$ with $\|\mathbf{u}_0'\| \sigma'/q'$: Assume that the former is bigger than the latter (i.e., the modulus switching is better than the rescaling), we have

$$q_\omega^{n/(m+n)} \frac{\sigma}{q} \geq \frac{1}{\sqrt{12}} \sqrt{\left(\frac{mN}{nM}\right)^{-\frac{m}{m+n}} \cdot \frac{(m+n)N}{n}}.$$

Simplifying the above equation, we attain

$$\left(\frac{12}{m+n}\right)^{m+n} \cdot \sigma^{2m} \cdot \sigma_s^{2n} \geq \left(\frac{N}{n}\right)^n \cdot \left(\frac{q^2 \cdot M}{m}\right)^m,$$

i.e.,

$$\left(\frac{12\sigma^2}{\sigma_s^2}\right)^n \geq \frac{(m+n)^{m+n}}{m^m} \cdot \frac{\sigma^{2n}}{\sigma_s^{2(m+n)}}.$$

The Primal Attack: For this attack, using the rescaling technique, we construct the following lattice instead of the lattice as in Subsection 5.4:

$$L_\omega = \left\{ \mathbf{v}_\omega \in \mathbb{Z}^{n+m+1} : (\mathbf{A} | \omega \mathbf{I}_m | -\mathbf{c}) \mathbf{v}_\omega = \mathbf{0} \bmod q \right\},$$

with $\omega = \sigma/\sigma_s$. Then $\mathbf{v}_{0,\omega} = (\mathbf{s} | \omega^{-1} \cdot \mathbf{e} | \mathbf{1})$ will be an unusual short vector in L_ω. Set $q_\omega := \omega^{-1} q$, then we have $\lambda_2(L_\omega) = \sqrt{\frac{m+n+1}{2\pi \mathfrak{e}}} \cdot q_\omega^{\frac{m}{m+n+1}}$ and $\lambda_1(L_\omega) = \|\mathbf{v}_{0,\omega}\| = \sqrt{N + m\sigma_s^2}$.

Again, if we use the modulus switching, we get the q'-ary lattice

$$L' = \left\{ \mathbf{v}' \in \mathbb{Z}^{n+m+1} : (\mathbf{A}' | \mathbf{I}_m | -\mathbf{c}') \mathbf{v}' = \mathbf{0} \bmod q' \right\},$$

with $\mathbf{A}' = \lfloor (q'/q) \cdot \mathbf{A} \rceil$, and $\lambda_2(L') = \sqrt{\frac{m+n+1}{2\pi \mathfrak{e}}} \cdot {q'}^{\frac{m}{m+n+1}}$, $\lambda_1(L') \approx \sqrt{P {q'}^2 + Q}$, where ${\sigma'}^2$, P, and Q as in Eq. (23).

We compare the modulus switching with the rescaling by determining when the condition $\lambda_2(L')/\lambda_1(L') \geq \lambda_2(L_\omega)/\lambda_1(L_\omega)$ holds (i.e., when the modulus switching is better than the rescaling), which is equivalent to

$$\frac{(n+1)^{n+1} \cdot 12^{m+n+1}}{(m+n+1)^{m+n+1} \cdot (m+12)^{n+1} \cdot N^{n+1} \cdot M^m} \geq \frac{q_\omega^{2m}}{(N + m\sigma_s^2)^{m+n+1}}.$$

Simplyfing the equation we obtain,

$$\left(\frac{12(N + m\sigma^2)}{m+n+1}\right)^{m+n+1} \geq \left(\frac{(m+12)N}{n+1}\right)^{n+1} \cdot \left(q^2 M\right)^m \cdot \frac{\sigma_s^{2m}}{\sigma^{2m}}.$$

To summary, we have the following theorem:

Theorem 5 (Modulus Switching vs Rescaling)**.** *Let m, n, M, N, q, q', σ, σ_s as mentioned in Subsection* 5.1.

(i) For the dual attack, the modulus switching outperforms the rescaling technique if the following requirement holds

$$\left(\frac{12\sigma^2}{\sigma_s^2}\right)^n \geq \frac{(m+n)^{m+n}}{m^m} \cdot \frac{\sigma^{2n}}{\sigma_s^{2(m+n)}}. \tag{27}$$

(ii) For the primal attack, the modulus switching outperforms the rescaling technique if the following requirement holds

$$\left(\frac{12(N + m\sigma^2)}{m+n+1}\right)^{m+n+1} \geq \left(\frac{(m+12)N}{n+1}\right)^{n+1} \cdot \left(q^2 M\right)^m \cdot \frac{\sigma_s^{2m}}{\sigma^{2m}}. \tag{28}$$

Table 6 How to choose parameters and proceed with our experiments in Section 8?

1. First, we choose ζ. The first $\zeta = 11/15$ is inspired from choosing parameters in the work of [15], the last one $\zeta = 1/3$ comes from [16] whilst two middle ones $\zeta = 2/3$ and $\zeta = 1/2$ are additionally suggested by us. Then choose n and compute $\log(q_{\text{min}})$ by (8) (for the decoding attack). We focus on $n = 60, n = 80$ and $n = 100$.
2. Next, choose $\log(q)$ to be around $\log(q_{\text{min}})$ (in the case of the decoding attack), from which we have q and p. After that we compute q' by (17).
3. Choose m: For the decoding attack, we compute m by (9), while for the primal attack we choose m by (26).
4. With chosen parameters, we check the condition (18) or the condition (25) depending on the attack we are considering.
5. For each tuple (ζ, n, q, p, q'), we sample 10 LWR instances. The small secret $\mathbf{s}$ is drawn uniformly at random over $\{-1, 0, 1\}^n$.
6. For each LWR instance, transform it to the LWE modulo q and the LWE modulo q'.
7. Finally, run the attacks on these two LWE instances.

Remark 7. *Considering Eq.* (28) *together with Eq.* (25)*, we can see in the case $\sigma_s < \sigma$ that if Eq.* (25) *holds we will also have Eq.* (28) *to hold. This turns out that for the primal attack against LWE/LWR instances, if we can apply the modulus switching to it (i.e., Eq.* (25) *holds) then the modulus switching is a better than the rescaling. However, for the dual attack, the relation between Eq.* (27) *and Eq.* (21) *is not clear since when $\sigma_s < \sigma$ we still do not whether $\sigma^{2n}/\sigma_s^{2(m+n)}$ is larger than 1 or not. In the case that the ratio is smaller than 1 and Eq.* (21) *holds, for the dual attack against LWE/LWR instances, if we are able to use modulus switching (i.e., Eq.* (21) *holds) then the modulus switching is a better choice rather than the rescaling technique. Take SABER [32] as an example, with the proposed parameters σ_s, σ and $m = n$ (see Table* 9*) we have $\sigma^{2n}/\sigma_s^{2(m+n)} < 1$. Thus using the modulus switching technique is better rather than the rescaling technique in the dual attack against SABER.*

7 The Impact of the Modulus Switching on the Rescaling

Again, in this section, we consider the LWE modulo q instances $(\mathbf{A}, \mathbf{c} = \mathbf{A} \cdot \mathbf{s} + \mathbf{e}) \in \mathbb{Z}_q^{m \times n} \times \mathbb{Z}_q^m$ whose $\|\mathbf{s}\| \ll \|\mathbf{e}\|$ and only focus on the dual attack and the primal attack. For convenient, we also refer all notations to Sections 5 and 6. Assume that we perform the modulus switching with the switching modulus q' first and then we apply the rescaling technique on the LWE instance. We will show that such a combination of these two techniques will make both the dual attack and the primal attack worse. Stress that the same phenomenon was reported by Bai and Galbraith in [22] for the primal attack. However, we will base our work on new theoretical arguments other than that of [22].

The Dual Attack: For the dual attack, first we do the modulus switching to reduce the LWE/LWR problem modulo q to the LWE problem modulo $q' = \sqrt{mN/(nM)}$, and then we do the rescaling on the new LWE modulo q' with the rescaling factor

$$\omega_{\text{mod}} = \frac{\sigma'}{\sigma_s} = \sqrt{\frac{Mq'^2 + N}{12\sigma_s^2}},$$

where σ' is the standard deviation of the q'-error (see Subsection 5.1). Then the resulting lattice $\Lambda_{\text{mod},\omega}(\mathbf{A}')$, which is

$$\left\{ \left(\mathbf{x}, \frac{\mathbf{y}}{\omega_{\text{mod}}} \right) \in \mathbb{Z}^m \times \frac{1}{\omega_{\text{mod}}} \cdot \mathbb{Z}^n : \mathbf{x}^t \cdot \mathbf{A}' = \mathbf{y}^t \bmod q' \right\}$$

with $\mathbf{A}' = \lfloor (q'/q) \cdot \mathbf{A} \rceil$, has volume of q_{mod}^n with

$$q_{\text{mod}} := \omega_{\text{mod}}^{-1} \cdot q' = \frac{q' \cdot \sigma_s}{\sigma'}.$$

Sames as the arguments in Subsection 6, we can find the short vector $\mathbf{u}_{0,\text{mod},\omega} \in \Lambda_{\text{mod},\omega}$ using an LBR algorithm $\mathcal{A}$ of the root Hermite Factor $\delta_{\mathcal{A}}$ and by Gaussian Heuristic we have

$$\begin{aligned} \frac{\|\mathbf{u}_{0,\text{mod},\omega}\|\sigma'}{q'} &= {\delta_{\mathcal{A}}}^{m+n} \cdot \frac{{q_{\text{mod}}}^{\frac{n}{m+n}} \cdot \sigma'}{q'} \\ &= {\delta_{\mathcal{A}}}^{m+n} \cdot \left(\frac{\sigma'}{q'} \right)^{\frac{m}{m+n}} {\sigma_s}^{\frac{n}{m+n}}, \end{aligned}$$

while only exploiting the rescaling without the modulus switching we get the short vector $\mathbf{u}_{0,\omega} = {\delta_{\mathcal{A}}}^{m+n} \cdot q_\omega^{\frac{n}{m+n}}$. Hence

$$\frac{\|\mathbf{u}_{0,\omega}\|\sigma}{q} = {\delta_{\mathcal{A}}}^{m+n} \cdot \frac{q_\omega^{\frac{n}{m+n}} \cdot \sigma}{q} = {\delta_{\mathcal{A}}}^{m+n} \cdot \left(\frac{\sigma}{q} \right)^{\frac{m}{m+n}} {\sigma_s}^{\frac{n}{m+n}}.$$

Again, thus, we just need to compare $\frac{\sigma'}{q'}$ with $\frac{\sigma}{q}$. We have

$$\frac{\sigma'}{q'} = \frac{Mq'^2 + N}{12q'} = \sqrt{\frac{(m+n)M}{12m}} > \sqrt{\frac{M}{12}} = \frac{\sigma}{q},$$

yielding that

$$\frac{\|\mathbf{u}_{0,\text{mod},\omega}\|\sigma'}{q'} \geq \frac{\|\mathbf{u}_{0,\omega}\|\sigma}{q}.$$

This claims that if we use the modulus switching first and we apply the rescaling alter, then the dual attack will be weaker than the dual attack only using the rescaling.

The Primal Attack: For the primal attack, only using the rescaling technique we have

$$\frac{\lambda_2(L_\omega)}{\lambda_1(L_\omega)} = \frac{\sqrt{\frac{m+n+1}{2\pi \mathfrak{e}}} \cdot {q_\omega}^{\frac{m}{m+n+1}}}{\sqrt{N + m\sigma_s^2}},$$

with $\omega = \sigma_s/\sigma$ and $q_\omega = q/\omega = q\sigma_s/\sigma$ as in Section 6.

Now, assume that first we do the modulus switching to reduce the LWE/LWR problem modulo q to the LWE problem modulo $q' = \sqrt{(m+12)N/((n+1)M)}$, and then the rescaling is performed on the new LWE modulo q' with the rescaling factor

$$\omega_{\text{mod}} = \frac{\sigma'}{\sigma_s} = \sqrt{\frac{Mq'^2 + N}{12\sigma_s^2}},$$

where σ' is the standard deviation of the q'-error (cf. Subsection 5.1). Then the resulting lattice $L_{\text{mod},\omega}$, which is

$$\left\{\mathbf{v}_{\text{mod},\omega} \in \mathbb{Z}^{n+m+1} : (\mathbf{A}'|\omega_{\text{mod}}\mathbf{I}_m| - \mathbf{c})\mathbf{v}_{\text{mod},\omega} = \mathbf{0} \bmod q'\right\}$$

with $\mathbf{A}' = \lfloor (q'/q) \cdot \mathbf{A} \rceil$, has volume of q_{mod}^n with

$$q_{\text{mod}} := \omega_{\text{mod}}^{-1} \cdot q' = \sigma_s \cdot \sqrt{\frac{12(m+12)}{(m+n+13)M}}.$$

Since $12/M = q^2/\sigma^2$, we have

$$q_{\text{mod}} = \frac{q \cdot \sigma_s}{\sigma} \cdot \sqrt{\frac{m+12}{m+n+13}} < \frac{q \cdot \sigma_s}{\sigma} = q_\omega. \tag{29}$$

Observe that $\mathbf{v}_{0,\text{mod},\omega} := (\mathbf{s}|\omega_{\text{mod}}^{-1} \cdot \mathbf{e}'|\mathbf{1})$ will be an unusual short vector in $L_{\text{mod},\omega}$. Thus $\lambda_1(L_{\text{mod},\omega}) = \lambda_1(L_\omega) = \sqrt{N + m\sigma_s^2}$. Then

$$\frac{\lambda_2(L_{\text{mod},\omega})}{\lambda_1(L_{\text{mod},\omega})} = \frac{\sqrt{\frac{m+n+1}{2\pi\mathfrak{e}}} \cdot q_{\text{mod}}^{\frac{m}{m+n+1}}}{\sqrt{N + m\sigma_s^2}}.$$

Due to Eq. (29), we obtain

$$\frac{\lambda_2(L_{\text{mod},\omega})}{\lambda_1(L_{\text{mod},\omega})} \leq \frac{\sqrt{\frac{m+n+1}{2\pi\mathfrak{e}}} \cdot q_\omega^{\frac{m}{m+n+1}}}{\sqrt{N + m\sigma_s^2}} = \frac{\lambda_2(L_\omega)}{\lambda_1(L_\omega)}.$$

The result in this subsection again confirms the observation of [22] that using the modulus switching (in which they chose $q'/q \approx 1/8$ (cf. [22, Section 5]) before performing their attack (which is the same as the primal attack accompanied by the rescaling technique) will make the attack worse.

8 Implementation and Experimental Results

We implemented the decoding attack and the primal attack on LWR problem to evaluate the efficacy of the q'-attack in comparison with the q-attack. In our experiments, we used SageMath version 8.1 [39] to implement these two attacks. The LBR algorithm used in our experiments is LLL [23]. We used the function".LLL()" to call the floating point implementation of LLL in the *fplll* library which is included in SageMath with the default reduction parameter 0.99. By using such an LLL algorithm, we have the corresponding constant mentioned in Eq. (3) is $c_{\text{LLL}} = 0.9775$ (see Section 2.3).

The experimental results are summarized in Tables 7 - 8. We refer to Table 6 for generating parameters, sampling LWR instances, as well as how to run attacks on the corresponding LWE instances.

We highlight some noticeable things from our experimental results:

- For the decoding attack, in all cases, the rHF of the q-attack is always bigger than that of the q'-attack. Interestingly, the rHF of the q'-attack becomes smaller once ζ declines while the rHF of the q-attack does not seem to change, namely, rHF(q') ≈ 1.0201 for all considered ζ's. Recall that, smaller root Hermite factor means that the LWE modulo q' instance is more easily solved by attacks than the LWE modulo q instance (see Subsection 5.2). For the primal attack, the case of $\zeta = 11/15$, sometimes we have rHF(q') >rHF(q).
- When ζ is close to 1, such as $\zeta = 11/15$, the q'-attack does not outperform the q-attack much. In contrast, when ζ is closer to 0 than 1, e.g., $\zeta = 1/3$, q' approach is much efficient than the q-attack in terms of success probability, rHF and even running time (we add the runtime data in Table 8) since q' is quite close to p and smaller than q.
- The bit size $\log(q')$ is quite close to $\log(p)$. Namely, in all considered cases, we have $\log(q') - \log(p)$ equals to 3 or 4. It seems that the difference $\log(q') - \log(p)$ increases (but slowly) once either n increases or/and ζ decreases.

9 Accessing the Impact of Modulus Switching on LWE/LWR-based Round 2 NIST PQC Submissions

Among 17 public-key encryption and key-establishment schemes and 9 digital signature schemes that was released as candidates for the second round NIST PQC Standardization, there are up to 9 submissions based on LWE/ LWR variants [9]. Each candidate proposes some tuples of parameters fulfilling one or more target security categories. In this section, we first give some more necessary notions and then review all the LWE/LWR-based second round submissions and try to check whether or not we should apply the modulus switching to them.

First, we give the definition of the centered binomial distribution (see, for example, [40, Section 2.4] or [32, Section 2.1]). The distribution is used in some NIST PQC Submissions as an alternative to the Gaussian sampling which is costly to implement in practice and also impacted by the side channel attacks.

Definition 5 (Centered Binomial Distribution)**.** *The centered binomial distribution of parameter η, denoted as Ψ_η, samples two bits a and b from the binary set $\{0,1\}$ then outputs $a-b$ with the following condition: the probability of the output 0 is $(2-\eta)/2$ and the probability of each of the outputs $1,-1$ is $\eta/4$. The standard deviation of this distribution is $\sqrt{\eta/2}$.*

LAC [40] with its latest version submitted to the second round NIST PQC submission changes to exploit the so-called *fixed weight centered binomial distribution* in which the Hamming weight of a random vector following the distribution is fixed.

Definition 6 (Fixed Weight Centered Binomial Distribution)**.** *The fixed weight centered binomial distribution of parameter (η, n, h) is a n-ary centered binomial distribution, denoted as $\Psi_\eta^{n,h}$ when ($0 < h < n/2$), outputs trinary vectors of length n whose the number of is 1 and of -1 is $h/2$, respectively, and the number of 0 is $n-h$. The standard deviation of this distribution is $\sqrt{h/n}$ and also is equal to $\sqrt{\eta/2}$.*

Besides LWE and LWR, many NIST PQC submissions also base their security on other algebraic variants of LWE and LWR, namely Ring-LWE/Ring-LWR (RLWE/RLWR), Module-LWE/Module-LWR (MLWE/MLWR), or even an integer variant of MLWE called ILWE. In these variants, instead of vectors, one uses a number of polynomials drawn from some polynomial ring [41]. Specifically, we usually consider the ring $R_q := \mathbb{Z}_q[x]/\langle x^n + 1\rangle$, where n is power of two.

So far, there has been no special attacks that exploit the algebraic structure of these variants. The typical way to attack these algebraic instances is to consider them as "standard" LWE/LWR instances in which a polynomial can be considered as a vector of its coefficients and a product of two polynomials can be represented as a matrix-vector multiplication. By "standard" LWE/LWR we mean the LWE/LWR problems defined as in Subsection 2.5. Formally, we will give some more definitions below. Here, let χ_s and χ_e be distributions over R_q.

Definition 7 (RLWE Problems)**.** *Given an RLWE instance $(a, c = a \cdot s + e) \in R_q \times R_q$ where $a \leftarrow \mathcal{U}(R_q)$, $s \leftarrow \chi_s$ and $e \leftarrow \chi_e$:*

- *The search-RLWE (sRLWE) problem is to find the secret s.*
- *The decision-RLWE (dRLWE) problem requires to distinguish the RLWE instance from the uniform pair $(a, c) \in R_q \times R_q$.*

Definition 8 (MLWE Problems)**.** *Given an MLWE instance $(\mathbf{A}, \mathbf{c} = \mathbf{A} \cdot \mathbf{s} + \mathbf{e}) \in R_q^{m\times n} \times R_q^m$ where $\mathbf{A} \leftarrow \mathcal{U}(R_q^{m\times n})$, $\mathbf{s} \leftarrow \chi_s^n$ and $\mathbf{e} \leftarrow \chi_e^m$:*

- *The search-MLWE (sMLWE) problem is to find the secret $\mathbf{s}$.*
- *The decision-MLWE (dMLWE) problem requires to distinguish the MLWE instance from the uniform pair $(\mathbf{A}, \mathbf{c}) \in R_q^{m\times n} \times R_q^m$.*

Table 7 We compare q'-decoding attack with q-decoding attack against LWR: ζ is the bit ratio of p and q, i.e., $\zeta = \log(p)/\log(q)$; n is the dimension of the secret; $\log(q_{\min})$ is the smallest bit size of q computed by (8) given ζ, n; $\log(q')$, $\log(q)$, $\log(p)$ are the bit size of the modulus q' computed by Eq. (17), of moduli q and p, respectively; m is the optimal number of LWR samples for the decoding attack computed by Eq. (9) (we use the same number of samples m in both the q-decoding attack and q'-decoding attack); the columns entitled "succ(q)", "succ(q')", "rHF(q)" and "rHF(q')" represent the success probability of the q-decoding attack and of the q'-decoding attack, the rHF's of the q-attack and of the q'-attack, respectively. Note that, rHF(q) and rHF(q') are computed using Eq. (3). We did not count the running ttime for this experiment. We ran this experiment on a MacBook Pro (Retina, 13-inch, Early 2015) installing macOS High Sierra version 10.13.3 with Memory 8GiB 1867 MHz DDR3, Processor 2.7GHz Intel Core i5, Graphics Intel Iris Graphics 6100 1536 MB.

ζ	LWR parameters ($p = q^{\zeta}$)				Switching modulus	Success Probability		Root Hermite Factor		Satifying
	$(n, \log(q_{\min}))$	m	$\log(q)$	$\log(p)$	$\log(q')$	succ(q)	succ(q')	rHF(q)	rHF(q')	Eq. (18)
$\frac{11}{15}$	(60,15)	166	**15**	11	**14**	**0%**	**60%**	1.0202	1.0200	Y
	(60,15)	176	**17**	12	**15**	**100%**	**100%**	1.0203	1.0197	Y
	(80,20)	204	**17**	12	**15**	**0%**	**0%**	1.0209	1.0200	Y
	(80,20)	209	**18**	13	**16**	**20%**	**80%**	1.0205	1.0200	Y
	(80,20)	221	**20**	15	**19**	**80%**	**100%**	1.0209	1.0200	Y
	(80,20)	232	**22**	16	**20**	**100%**	**100%**	1.0205	1.0200	Y
	(100,25)	247	**20**	15	**19**	**20%**	**60%**	1.0210	1.0203	Y
	(100,25)	276	**25**	18	**22**	**100%**	**100%**	1.0212	1.0195	Y
$\frac{2}{3}$	(60,18)	171	**16**	11	**14**	**80%**	**80%**	1.0203	1.0199	Y
	(60,18)	176	**17**	11	**14**	**20%**	**60%**	1.0201	1.0188	Y
	(60,18)	181	**18**	12	**15**	**100%**	**100%**	1.0202	1.0191	Y
	(80,24)	226	**21**	14	**18**	**20%**	**100%**	1.0203	1.0187	Y
	(80,24)	237	**23**	15	**19**	**40%**	**100%**	1.0215	1.0180	Y
	(80,24)	242	**24**	16	**20**	**100%**	**100%**	1.0208	1.0183	Y
	(100,30)	270	**24**	16	**20**	**0%**	**80%**	1.0211	1.0184	Y
	(100,30)	281	**26**	17	**21**	**0%**	**100%**	1.0211	1.0179	Y
	(100,30)	292	**28**	19	**23**	**100%**	**100%**	1.0213	1.0182	Y
$\frac{1}{3}$	(60,71)	299	**49**	16	**20**	**0%**	**100%**	1.0215	1.0091	Y
	(80,95)	386	**61**	20	**24**	**0%**	**100%**	1.0214	1.0088	Y
	(100,119)	424	**59**	20	**24**	**0%**	**100%**	1.0213	1.0091	Y
$\frac{1}{2}$	(60,32)	218	**26**	13	**17**	**0%**	**100%**	1.0210	1.0142	Y
	(60,32)	234	**30**	15	**19**	**20%**	**100%**	1.0201	1.0140	Y
	80,43)	256	**27**	14	**18**	**0%**	**80%**	1.0208	1.0146	Y
	80,43)	292	**35**	18	**22**	**0%**	**100%**	1.0209	1.0140	Y
	(100,53)	307	**31**	16	**20**	**0%**	**60%**	1.0214	1.0143	Y
	(100,53)	336	**37**	19	**23**	**0%**	**100%**	1.0213	1.0138	Y

Table 8 We compare the q'-primal attack with the q-primal attack against LWR: Notations are same as in Table 7. We note some more things: $\log(q')$ is the bit size of the modulus q' computed by Eq. (24), m is the optimal number of LWR samples for the primal attack computed by Eq. (22) (we use the same number of samples m in both the q-primal attack and the q'- primal attack); the columns entitled "time(q')" represent the running time of the q-primal attack and of the q'-primal attack (in seconds), respectively. The last column is to check the condition in Eq. (25): "Y" means the condition is satisfied while "N" means not. We ran this experiment on a desktop computer installing Ubuntu 16.04 LTS, with Memory 15.7GiB, Processor Intel Core i7 CPU870@2.93Ghz × 8, Graphics NVA8.

ζ	LWR parameters ($p = q^{\zeta}$)				Switching modulus	Success Probability		Root Hermite Factor		Running time		Satisfying
	n	m	$\log(q)$	$\log(p)$	$\log(q')$	succ(q)	succ(q')	rHF(q)	rHF(q')	time(q)	time(q')	Eq. (25)
$\frac{1}{2}$	60	131	**20**	10	**14**	**0%**	**80%**	0.9869	0.9787	17961 sec	9093 sec	Y
	80	176	**27**	14	**18**	**0%**	**60%**	1.0213	0.9832	146812 sec	68110 sec	Y
	100	202	**30**	15	**19**	**0%**	**40%**	1.0203	0.9854	506745 sec	240675 sec	Y
	100	217	**33**	17	**21**	**0%**	**80%**	1.0201	0.9825	534835 sec	304330 sec	Y
$\frac{2}{3}$	60	111	**16**	11	**14**	**80%**	**100%**	0.9876	0.9839	11436 sec	6397 sec	Y
	80	135	**19**	13	**17**	**60%**	**100%**	0.9869	0.9850	52615 sec	38645 sec	Y
	100	165	**23**	15	**19**	**20%**	**100%**	0.9857	0.9834	336873 sec	245531 sec	Y
$\frac{11}{15}$	60	36	**5**	4	**7**	**0%**	**0%**	1.0185	1.0189	56 sec	76 sec	N
	60	94	**13**	10	**13**	**20%**	**60%**	0.9854	0.9869	3957 sec	3900 sec	Y
	80	117	**16**	12	**15**	**60%**	**80 %**	0.9867	0.9868	26448 sec	24014 sec	Y
	100	23	**5**	4	**6**	**0%**	**0%**	1.0152	1.0159	62 sec	77 sec	N
	100	147	**20**	15	**19**	**0%**	**80%**	0.9859	0.9849	144161 sec	120588 sec	Y
	100	114	**15**	11	**14**	**0%**	**0%**	1.0152	1.0159	62 sec	77 sec	N

Definition 9 (RLWR Problems)**.** *Given an RLWR instance $(a, c = \lfloor a \cdot s \rceil_{q,p}) \in R_q \times R_p$, where $a \leftarrow \mathcal{U}(R_q)$, and $s \leftarrow \chi_s$:*

- *The search-RLWR (sLWR) problem is to find the secret* $\mathbf{s}$.
- *The decision-RLWR (dLWR) problem requires to distinguish the ring-LWR instance from the uniform pair $(a, c) \in R_q \times R_p$.*

Definition 10 (MLWR Problems)**.** *Given an MLWR instance $(\mathbf{A}, \mathbf{c} = \lfloor \mathbf{As} \rceil_{q,p}) \in R_q^{m \times n} \times R_p^m$, where $\mathbf{A} \leftarrow \mathcal{U}(R_q^{m \times n})$, and $\mathbf{s} \leftarrow \chi_s^n$:*

- *The search MLWR (sMLWR) problem is to find the secret* $\mathbf{s}$.
- *The decision MLWR (dMLWR) problem requires to distinguish the ring-LWR instance from the uniform pair $(\mathbf{A}, \mathbf{c}) \in R_q^{m \times n} \times R_p^m$.*

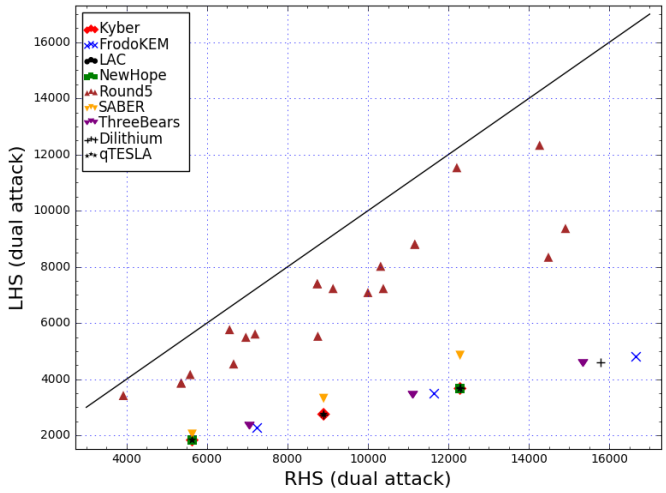

Fig. 3: Using data in Table 9, we evaluate the impact of the modulus switching again the second round NIST PQC Submissions under the dual attack. "LHS", "RHS" mean the left-hand side and the right-hand side, respectively, of Eq. (21). The line represents the case of the equality LHS=RHS. A point which is over the line means that an instance of the corresponding submission will be affected by the modulus switching. We see that all points are under the line, so the modulus switching is not available to apply to the dual attack in breaking the hard underlying problems of the NIST PQC submissions listed in Section 9 with proposed parameters.

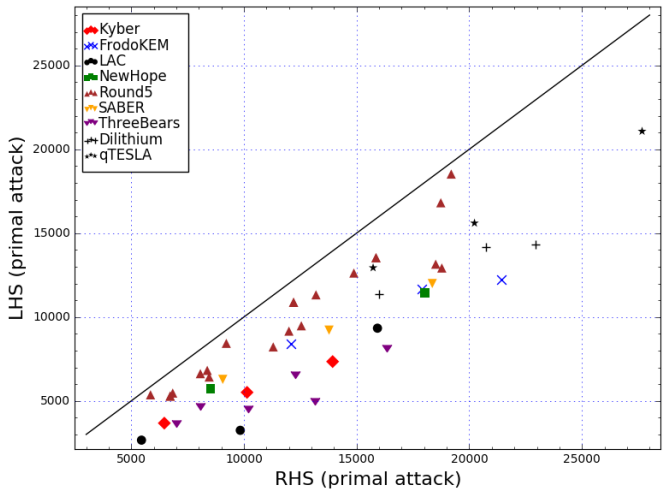

Fig. 4: Same as Figure 3, except that we consider Eq. (25) with respect to the primal attack.

At the moment, we are ready to review the LWE/LWR-based candidates for the second round NIST PQC. Remark that there are some changes in choosing parameters appearing in some candidates submitted to the second round NIST PQC in comparison with they were in the first round. In the following, we also try to update those changes basing on the latest versions of the second round submissions.

CRYSTALS-KYBER [42]: KYBER is a family of key encapsulation mechanisms (KEMs) whose security (in the latest version submitted to the second round NIST PQC) is based on the sMLWE problem over the ring $\mathbb{Z}_{3329}[x]/\langle x^{256}+1\rangle$. The secret and the noise of KYBER are drawn from the centered binomial distribution of parameter 2 for its all proposed version including KYBER512, KYBER768 and KYBER1024. In the version of KYBER submitted to the first round of the NIST PQC competition, KYBER worked on the polynomial ring of a bigger modulus $\mathbb{Z}_{7681}[x]/\langle x^{256}+1\rangle$ and the parameters of the centered binomial distributions from which its secrets and noises are drawn vary from 3 to 5.

FrodoKEM [43]: FrodoKEM is also a family of KEMs that uses algebraically unstructured lattices. The security is based on the LWE problem. The secret and the noise are sampled from a discrete, symmetric distribution on Z, centered at zero and with small support, which approximates a rounded continuous Gaussian distribution (see [43, Definition 2.11]).

LAC [40]: LAC is a cryptosystem based on the RLWE problem. The special point of LAC is that LAC makes use of the very small modulus $q = 251$. In the first round, the scheme used centered binomial distributions of small parameters for secrets and errors to guarantee the difficulty of the RLWE problem with such a small modulus. However, the scheme submitted to the second round NIST PQC has some changes. One of them is that besides the centered binomial distribution, the scheme also mainly uses the so-called fixed weight centered binomial distribution (see Definition 6). The authors of LAC assure that using such the distribution will not affect the security and concrete security of LAC [40, Section 5.2].

NewHope [44]: NewHope is a cryptosystem of KEMs, based its hardness on the RLWE problem. The scheme uses power-of-2 cyclotomic rings with a common modulus $q = 12289$. The secrets and errors are sampled from the centered binomial distribution of parameter 8.

Round5 [15]: The security of Round5 relies on the General Learning with Rounding (GLWR) problem of power-of-two moduli q and p to unify the LWR problem and RLWR problem over prime-order cyclotomic polynomial rings, namely, $x^n + \cdots + x + 1$ with $n + 1$ is a prime number. The secrets of Round5 are sampled according to a fixed Hamming weight distribution of support $\{-1, 0, 1\}$, named $\mathcal{H}_{n,k}(h)$, from which each drawn vector of of length $n \cdot k$ has exactly h non-zero components. Hence, the standard deviation of the secret is computed by $\sigma_s = \sqrt{h/(nk)}$. As mentioned in Remark 1 instead of following [33] to compute the standard deviation of the LWR error as in [45, Section 5] by $\sqrt{(q^2/p^2-1)/12}$, we use the formula in Eq. (6).

SABER [32]: SABER is a family of PKEs and KEMs, whose security based on the quantum hardness of the MLWR problem. The scheme uses modules of varying rank over a fixed power-of-2 cyclotomic ring with fixed dimension, namely $\mathbb{Z}_{1024}/\langle x^{256}+1\rangle$. The MLWR secret distribution is the centered binomial distribution.

ThreeBears [46]: ThreeBears relies its security on an integer variant of the MLWE problem modulo $q = 1024$ (cf. [47] or [46, Section 4.1]). Instead of MLWE over a polynomial ring with an indeterminate x as usual, the indeterminate x is evaluated, yielding instead an integer LWE over a ring modulo an integer N which is a large generalized Mersenne prime and then all computations take place modulo N. The noise modulo N is sampled from a special distribution of fixed variance σ^2 by expanding a seed to one byte per digit, and then converting the digit to an integer with the desired variance (cf. [46, Subsection 2.4.2]).

CRYSTALS-DILITHIUM [48]: Dilithium is a lattice-based signature scheme whose security comes from the hardness of the MLWE problem over the fixed ring $\mathbb{Z}_{8380417}[x]/\langle x^{256}+1\rangle$. The secret and the noise of Dilithium are sampled according to a uniform distribution over the interval $[-\gamma, \gamma]$. For DILITHIUM-II, $\gamma = 6$, for DILITHIUM-III, $\gamma = 5$, and for DILITHIUM-IV, $\gamma = 3$.

qTESLA [49]: qTESLA is a family of post-quantum signature schemes based on the hardness of the dRLWE problem. qTESLA

Table 9 **Parameters of the LWE/LWR-based submissions in the second round NIST PQC**

NIST Submissions	Security Categories	Hard Problems	$\mathbb{Z}_q[x]/\langle\phi(x)\rangle$ $\phi(x)=?$	dimension n	#samples m	modulo q	modulo p	std. of error σ	std. of secret σ_s
CRYSTALS-Kyber									
KYBER512	1	MLWE	$x^{256}+1$	512	512	3329	$--$	1.00	1.00
KYBER768	3			768	768	3329	$--$	1.00	1.00
KYBER1024	5			1024	1024	3329	$--$	1.00	1.00
FrodoKEM									
Frodo640	1	LWE		640	640	2^{15}	$--$	2.8	2.8
Frodo976	3			976	976	2^{16}	$--$	2.3	2.3
Frodo1344	5			1344	1344	2^{16}	$--$	1.4	1.4
LAC									
LAC-128	1,2	RLWE	x^n+1	512	512	251	$--$	0.71	0.71
LAC-192	3,4			1024	1024	251	$--$	0.5	0.5
LAC-256	5			1024	1024	251	$--$	0.71	0.71
NewHope									
NewHope512	1	RLWE	x^n+1	512	512	12289	$--$	2.00	2.00
NewHope1024	5			1024	1024	12289	$--$	2.00	2.00
Round5									
R5ND1KEM0d	1	RLWR	$x^n+\cdots+x+1$	618	618	2^{11}	2^8	2.58	0.41
R5ND3KEM0d	3			786	786	2^{13}	2^9	4.90	0.70
R5ND5KEM0d	5			1018	1018	2^{14}	2^9	9.52	0.65
R5ND1KEM5d	1			490	490	2^{10}	2^7	2.58	0.58
R5ND3KEM5d	3			756	756	2^{12}	2^8	4.90	0.57
R5ND5KEM5d	5			940	940	2^{12}	2^8	4.90	0.66
R5N11KEM0d	1	LWR		594	594	2^{13}	2^{10}	2.58	0.63
R5N13KEM0d	3			881	881	2^{13}	2^{10}	2.58	0.52
R5N15KEM0d	5			1186	1186	2^{15}	2^{12}	2.58	0.78
R5ND0KEM2iot	$--$	RLWR	$x^n+\cdots+x+1$	372	372	2^{11}	2^7	4.90	0.70
R5ND1KEM4longkey	$--$			490	490	2^{10}	2^7	2.58	0.58
R5ND1PKE0d	1			586	586	2^{13}	2^9	4.90	0.56
R5ND3PKE0d	3			852	852	2^{12}	2^9	2.58	0.50
R5ND5PKE0d	5			1170	1170	2^{13}	2^9	4.90	0.44
R5ND1PKE5d	1			508	508	2^{10}	2^7	2.58	0.52
R5ND3PKE5d	3			756	756	2^{12}	2^8	4.90	0.57
R5ND5PKE5d	5			940	940	2^{12}	2^8	4.90	0.66
R5N11PKE0d	1	LWR		636	636	2^{12}	2^9	2.58	0.42
R5N13PKE0d	3			876	636	2^{15}	2^{11}	4.90	0.71
R5N15PKE0d	5			1217	1217	2^{15}	2^{12}	2.58	0.62
R5N13PKE0smallCT	$--$			757	757	2^{14}	2^9	2.58	0.71
SABER									
LightSABER	1	MLWR	$x^{256}+1$	512	512	2^{13}	2^{10}	2.58	2.24
SABER	3			768	768	2^{13}	2^{10}	2.58	2.00
FireSABER	5			1024	1024	2^{13}	2^{10}	2.58	1.73
ThreeBears									
BabyBear (cca 0)	2	ILWE	$q^{312}-q^{156}-1$	624	624	1024	$--$	1.00	1.00
BabyBear (cca1)	2			624	624	1024	$--$	0.75	0.75
MamaBear (cca 0)	5			936	936	1024	$--$	0.94	0.94
MamaBear (cca 1)	4			936	936	1024	$--$	0.64	0.64
PapaBear (cca 0)	5			1248	1248	1024	$--$	0.87	0.87
PapaBear (cca 1)	5			1248	1248	1024	$--$	0.56	0.56
CRYSTALS-Dilithium									
DILITHIUM-II	1	MLWE	$x^{256}+1$	768	768	8380417	$--$	3.74	3.74
DILITHIUM-III	2			1024	1024	8380417	$--$	3.16	3.16
DILITHIUM-IV	3			1280	1280	8380417	$--$	2.00	2.00
qTESLA									
qTESLA-I	1	dRLWE	x^n+1	512	512	4205569	$--$	22.93	22.93
qTESLA-II	2			768	768	8404933	$--$	9.73	9.73
qTESLA-III	3			1024	1024	8404993	$--$	10.2	10.2

is proposed to utilize two different ways in generating parameters. The first way aiming to a heuristic parameter generation results in qTESLA-I, qTESLA-II, qTESLA-III, qTESLA-V and qTESLA-V-size. The second way following a provably-secure parameter generation according to existing security reductions results in qTESLA-p-I and qTESLA-p-IIII[‡]. The secret and the noise of qTESLA are drawn according to the *centered discrete Gaussian distribution* for $c \in \mathbb{Z}$ with standard deviation σ is defined as follows: $\mathcal{D}_\sigma = \rho_\sigma(c)/\rho_\sigma(\mathbb{Z})$, where $\sigma > 0$, $\rho_\sigma(c) = \exp(-c^2/(2\sigma^2))$, and $\rho_\sigma(\mathbb{Z}) = 1 + 2\sum_{c=1}^{\infty} \rho_\sigma(c)$.

In Table 9, we summarize main parameters relating to the underlying hard problem of the second round NIST PQC submissions mentioned above: the dimension of the corresponding lattice n, the number of samples m, the moduli q and p, the standard deviation of the error σ_e and the standard deviation of the secret σ_s. Using this data we evaluate the impact of the modulus switching technique on the primal attack and the dual attack against the Second-Round NIST PQC Candidates. Substituting the real values of parameters as in Table 9 into Eqs. (21) and (25), we compute the left-hand side (LHS) and the right-hand side (RHS) of these equations, we then compare these two sides for each equation. The plotted results are shown in Figure 3 and Figure 4. We can see from the figures that, the plotted points for Round5 are closest to the straight lines (*the boundary lines*) than those of the other submissions, especially in the primal attack. This implies that many instances of Round5 might be impacted most by the modulus switching technique. In contrast, FrodoKEM, qTESLA, ThreeBears and Dilithium are submissions whose many instances that are farthest from the boundary lines. And hence the modulus switching may have the least influence on these instances.

10 Conclusion

In this paper, we concentrated on evaluating the effect of the modulus switching technique on some attacks against LWE and LWR problems as well as the impact of this technique on the so-called rescaling technique. We gave the suitable formulas for choosing the best switching modulus for the decoding attack, the dual attack and the primal attack. We also showed the corresponding conditions under which using the modulus switching technique make each attack stronger than without using this technique. Using the conditions, we theoretically assessed the security of the second round LWE/LWR-based NIST PQC submissions under the modulus switching technique.

Although our work does not give any serious warning to the security of the LWE/LWR-based NIST PQC submissions in the second round, it suggests that the modulus switching technique should be carefully considered in choosing parameters and security analyses of prospective LWE/LWR-based cryptosystems.

11 Related own conference publication

The present paper is an extended version of our paper presented at CANS 2018 [18] which originally focused on the BDD attack (another name of the decoding attack) against LWR. This paper significantly expands our approach to the dual attack and the primal attack against both the LWE and the LWR problems. Also, based on some results of [18] we investigated the relation between the modulus switching technique and the rescaling technique. As an application, we use our results to evaluate the influence of the modulus switching to LWE/LWR-based NIST submissions accepted as the second round candidates.

[‡] *We do not include the parameters of qTESLA-V, qTESLA-V-size, qTESLA-p-I and qTESLA-p-IIII in the Table 9 because they are too large to plot in the same ratio as the other parameters.*

12 Acknowledgements

This work was supported by JST CREST Grant Number JPMJCR14D6, Japan. A part of this work was also supported by JSPS KAKENHI Grant Number 16H02830.

13 References

1 Shor, P.W. 'Algorithms for Quantum Computation: Discrete Logarithms and Factoring'. In: Proceedings of the 35th Annual Symposium on Foundations of Computer Science. SFCS '94. (Washington, DC, USA: IEEE Computer Society, 1994). pp. 124–134. Available from: https://doi.org/10.1109/SFCS.1994.365700

2 Bernstein, D.J. 'Introduction to post-quantum cryptography'. In: Bernstein, D.J., Buchmann, J., Dahmen, E., editors. Post-Quantum Cryptography. (Berlin, Heidelberg: Springer Berlin Heidelberg, 2009). pp. 1–14. Available from: https://doi.org/10.1007/978-3-540-88702-7_1

3 Regev, O.: 'On Lattices, Learning with Errors, Random Linear Codes, and Cryptography', **J ACM**, 2009, **56**, (6), pp. 34:1–34:40. Available from: http://doi.acm.org/10.1145/1568318.1568324

4 Peikert, C. 'Public-key Cryptosystems from the Worst-case Shortest Vector Problem: Extended Abstract'. In: Proceedings of the Forty-first Annual ACM Symposium on Theory of Computing. STOC '09. (New York, NY, USA: ACM, 2009). pp. 333–342. Available from: http://doi.acm.org/10.1145/1536414.1536461

5 Gentry, C., Peikert, C., Vaikuntanathan, V. 'Trapdoors for Hard Lattices and New Cryptographic Constructions'. In: Proceedings of the Fortieth Annual ACM Symposium on Theory of Computing. STOC '08. (New York, NY, USA: ACM, 2008). pp. 197–206. Available from: http://doi.acm.org/10.1145/1374376.1374407

6 Banerjee, A., Peikert, C., Rosen, A. 'Pseudorandom Functions and Lattices'. In: Pointcheval, D., Johansson, T., editors. Advances in Cryptology – EUROCRYPT 2012. (Berlin, Heidelberg: Springer Berlin Heidelberg, 2012). pp. 719–737

7 Alwen, J., Krenn, S., Pietrzak, K., et al. 'Learning with Rounding, Revisited'. In: Canetti, R., Garay, J.A., editors. Advances in Cryptology – CRYPTO 2013. (Berlin, Heidelberg: Springer Berlin Heidelberg, 2013). pp. 57–74. Available from: https://doi.org/10.1007/978-3-642-40041-4_4

8 Boneh, D., Lewi, K., Montgomery, H., et al. 'Key Homomorphic PRFs and Their Applications'. In: Canetti, R., Garay, J.A., editors. Advances in Cryptology – CRYPTO 2013. (Berlin, Heidelberg: Springer Berlin Heidelberg, 2013). pp. 410–428. Available from: https://doi.org/10.1007/978-3-642-40041-4_23

9 Alagic, G., Alperin.Sheriff, J., Apon, D., et al.. 'NISTIR 8240 Status Report on the First Round of the NIST Post-Quantum Cryptography Standardization Process'. (NIST Post-Quantum Cryptography Standardization, 2019). Available from https://nvlpubs.nist.gov/nistpubs/ir/2019/NIST.IR.8240.pdf

10 Micciancio, D., Regev, O. 'Lattice-based cryptography'. In: Bernstein, D.J., Buchmann, J., Dahmen, E., editors. Post-Quantum Cryptography. (Berlin, Heidelberg: Springer Berlin Heidelberg, 2009). pp. 147–191

11 Albrecht, M.R., Player, R., Scott, S.: 'On the concrete hardness of Learning with Errors', **Journal of Mathematical Cryptology**, 2015, **9**, (3), pp. 169–203. Available from: https://doi.org/10.1515/jmc-2015-0016

12 Brakerski, Z., Langlois, A., Peikert, C., et al. 'Classical Hardness of Learning with Errors'. In: Proceedings of the Forty-fifth Annual ACM Symposium on Theory of Computing. STOC '13. (New York, NY, USA: ACM, 2013). pp. 575–584. Available from: http://doi.acm.org/10.1145/2488608.2488680

13 Bogdanov, A., Guo, S., Masny, D., et al. 'On the Hardness of Learning with Rounding over Small Modulus'. (Berlin, Heidelberg: Springer Berlin Heidelberg, 2016). pp. 209–224. Available from: https://doi.org/10.1007/978-3-662-49096-9_9

14 Alperin.Sheriff, J., Apon, D.C.: 'Dimension-preserving reductions from lwe to lwr', **IACR Cryptology ePrint Archive**, 2016, **2016**, pp. 589. Available from: https://eprint.iacr.org/2016/589

15 Baan, H., Bhattacharya, S., Garcia.Morchon, O., et al.. 'Round5: KEM and PKE based on (Ring) Learning with Rounding'. (NIST Post-Quantum Cryptography Standardization, 2019). Available from https://csrc.nist.gov/Projects/Post-Quantum-Cryptography/Round-2-Submissions. Accessed on April 19, 2019.

16 Duc, A., Tramèr, F., Vaudenay, S. 'Better algorithms for lwe and lwr'. In: Oswald, E., Fischlin, M., editors. Advances in Cryptology – EUROCRYPT 2015. (Berlin, Heidelberg: Springer Berlin Heidelberg, 2015). pp. 173–202. Available from: https://doi.org/10.1007/978-3-662-46800-5_8

17 Arora, S., Ge, R. 'New Algorithms for Learning in Presence of Errors'. In: Aceto, L., Henzinger, M., Sgall, J., editors. Automata, Languages and Programming. (Berlin, Heidelberg: Springer Berlin Heidelberg, 2011). pp. 403–415

18 Le, H.Q., Mishra, P.K., Duong, D.H., et al. 'Solving LWR via BDD Strategy: Modulus Switching Approach'. In: Camenisch, J., Papadimitratos, P., editors. Cryptology and Network Security. (Cham: Springer International Publishing, 2018). pp. 357–376

19 Brakerski, Z., Vaikuntanathan, V. 'Efficient Fully Homomorphic Encryption from (Standard) LWE'. In: Proceedings of the 2011 IEEE 52Nd Annual Symposium on Foundations of Computer Science. FOCS '11. (Washington, DC, USA: IEEE Computer Society, 2011). pp. 97–106. Available from: http://dx.doi.org/10.1109/FOCS.2011.12

20 Albrecht, M.R., Faugère, J.C., Fitzpatrick, R., et al. 'Lazy Modulus Switching for the BKW Algorithm on LWE'. In: Krawczyk, H., editor. Public-Key Cryptography – PKC 2014. (Berlin, Heidelberg: Springer Berlin Heidelberg, 2014). pp. 429–445

21 Bindel, N., Buchmann, J., Göpfert, F., et al.: 'Estimation of the Hardness of the Learning with Errors Problem with a Restricted Number of Samples', **Journal of Mathematical Cryptology**, 2018, **13**, (1), pp. 47–67. Available from: http://doi.acm.org/10.1515/jmc-2017-0040

22 Bai, S., Galbraith, S.D. 'Lattice Decoding Attacks on Binary LWE'. In: Susilo, W., Mu, Y., editors. Information Security and Privacy. (Cham: Springer International Publishing, 2014). pp. 322–337
23 Lenstra, A.K., Lenstra, H.W., Lovász, L.: 'Factoring Polynomials with Rational Coefficients', **Mathematische Annalen**, 1982, **261**, (4), pp. 515–534. Available from: https://doi.org/10.1007/BF01457454
24 Schnorr, C.P., Euchner, M.: 'Lattice basis reduction: Improved practical algorithms and solving subset sum problems', **Mathematical Programming**, 1994, **66**, (1), pp. 181–199. Available from: https://doi.org/10.1007/BF01581144
25 Gama, N., Nguyen, P.Q. 'Predicting Lattice Reduction'. In: Smart, N., editor. Advances in Cryptology – EUROCRYPT 2008. (Berlin, Heidelberg: Springer Berlin Heidelberg, 2008). pp. 31–51
26 Kudo, M., Yamaguchi, J., Guo, Y., et al. 'Practical Analysis of Key Recovery Attack Against Search-LWE problem'. In: Ogawa, K., Yoshioka, K., editors. Advances in Information and Computer Security. (Cham: Springer International Publishing, 2016). pp. 164–181
27 Applebaum, B., Cash, D., Peikert, C., et al. 'Fast Cryptographic Primitives and Circular-Secure Encryption Based on Hard Learning Problems'. In: Halevi, S., editor. Advances in Cryptology - CRYPTO 2009. (Berlin, Heidelberg: Springer Berlin Heidelberg, 2009). pp. 595–618
28 Brakerski, Z., Gentry, C., Vaikuntanathan, V. '(Leveled) Fully Homomorphic Encryption Without Bootstrapping'. In: Proceedings of the 3rd Innovations in Theoretical Computer Science Conference. ITCS '12. (New York, NY, USA: ACM, 2012). pp. 309–325. Available from: http://doi.acm.org/10.1145/2090236.2090262
29 Brakerski, Z., Vaikuntanathan, V. 'Fully Homomorphic Encryption from Ring-LWE and Security for Key Dependent Messages'. In: Rogaway, P., editor. Advances in Cryptology – CRYPTO 2011. (Berlin, Heidelberg: Springer Berlin Heidelberg, 2011). pp. 505–524
30 Cheon, J.H., Kim, D., Lee, J., Song, Y. 'Lizard: Cut off the Tail! A Practical Post-quantum Public-key Encryption from LWE and LWR'. In: Catalano, D., De.Prisco, R., editors. Security and Cryptography for Networks. (Cham: Springer International Publishing, 2018). pp. 160–177
31 Fang, F., Li, B., Lu, X., et al. '(Deterministic) Hierarchical Identity-based Encryption from Learning with Rounding over Small Modulus'. In: Proceedings of the 11th ACM on Asia Conference on Computer and Communications Security. ASIA CCS '16. (New York, NY, USA: ACM, 2016). pp. 907–912. Available from: http://doi.acm.org/10.1145/2897845.2897922
32 D'Anvers, J.P., Karmakar, A., Roy, S.S., et al.. 'Saber: Mod-LWR based KEM'. (NIST Post-Quantum Cryptography Standardization, 2019). Available from: https://csrc.nist.gov/Projects/Post-Quantum-Cryptography/Round-2-Submissions. Accessed on April 19, 2019.
33 Nguyen, P.. 'Comment on PQC Forum'. (PQC Forum, 2018). Available from: https://groups.google.com/a/list.nist.gov/forum/#!topic/pqc-forum/nZBIBvYmmUI. Accessed on April 19, 2019.
34 Lindner, R., Peikert, C. 'Better Key Sizes (and Attacks) for LWE-Based Encryption'. In: Kiayias, A., editor. Topics in Cryptology – CT-RSA 2011. (Berlin, Heidelberg: Springer Berlin Heidelberg, 2011). pp. 319–339
35 Bischof, C., Buchmann, J., Dagdelen, Ö., et al. 'Nearest Planes in Practice'. In: Cryptography and Information Security in the Balkans. (Cham: Springer International Publishing, 2015). pp. 203–215
36 Göpfert, F., van Vredendaal, C., Wunderer, T. 'A Hybrid Lattice Basis Reduction and Quantum Search Attack on LWE'. In: Lange, T., Takagi, T., editors. Post-Quantum Cryptography. (Cham: Springer International Publishing, 2017). pp. 184–202
37 Alkim, E., Ducas, L., Pöppelmann, T., et al. 'Post-quantum Key Exchange - A New Hope'. In: Proceedings of the 25th USENIX Security Symposium. (USENIX Association, 2016). pp. 327–343
38 Albrecht, M.R., Fitzpatrick, R., Göpfert, F. 'On the Efficacy of Solving LWE by Reduction to Unique-SVP'. In: Lee, H.S., Han, D.G., editors. Information Security and Cryptology – ICISC 2013. (Cham: Springer International Publishing, 2014). pp. 293–310
39 A. Stein, W., et al.. 'Sage Mathematics Software (Version 8.1)'. (SageMath website, 2018). Available from: http://www.sagemath.org
40 Lu, X., Liu, Y., Jia, D., et al.. 'LAC: Lattice-based Cryptosystems'. (NIST Post-Quantum Cryptography Standardization, 2019). Available from: https://csrc.nist.gov/Projects/Post-Quantum-Cryptography/Round-2-Submissions. Accessed on April 19, 2019.
41 Lyubashevsky, V., Peikert, C., Regev, O. 'On Ideal Lattices and Learning with Errors over Rings'. In: Gilbert, H., editor. Advances in Cryptology – EUROCRYPT 2010. (Berlin, Heidelberg: Springer Berlin Heidelberg, 2010). pp. 1–23
42 Avanzi, R., Bos, J., Ducas, L., et al.. 'CRYSTALS-Kyber'. (NIST Post-Quantum Cryptography Standardization, 2017). Available from: https://csrc.nist.gov/Projects/Post-Quantum-Cryptography/Round-2-Submissions. Accessed on April 19, 2019.
43 Alkim, E., Bos, J.W., Ducas, L., et al.. 'FrodoKEM: Learning with Errors Key Encapsulation'. (NIST Post-Quantum Cryptography Standardization, 2019). Available from: https://csrc.nist.gov/Projects/Post-Quantum-Cryptography/Round-2-Submissions. Accessed on April 19, 2019.
44 Alkim, E., Avanzi, R., Bos, J., et al.. 'NewHope: Post-quantum key encapsulation'. (NIST Post-Quantum Cryptography Standardization, 2019). Available at https://csrc.nist.gov/Projects/Post-Quantum-Cryptography/Round-2-Submissions. Accessed on April 19, 2019.
45 Albrecht, M.R., Curtis, B.R., Deo, A., et al. 'Estimate All the LWE, NTRU Schemes!'. In: Security and Cryptography for Networks. (Cham: Springer International Publishing, 2018). pp. 351–367
46 Hamburg, M.. 'Post-quantum cryptography proposal: ThreeBears'. (NIST Post-Quantum Cryptography Standardization, 2019). Available from: https://csrc.nist.gov/Projects/Post-Quantum-Cryptography/Round-2-Submissions. Accessed on April 19, 2019.
47 Gu, C.: 'Integer Version of Ring-LWE and its Applications', **IACR Cryptology ePrint Archive**, 2017, **2017**, pp. 641
48 Ducas, L., Kiltz, E., Lepoint, T., et al.. 'CRYSTALS-Dilithium'. (NIST Post-Quantum Cryptography Standardization, 2019). Available from: https://csrc.nist.gov/Projects/Post-Quantum-Cryptography/Round-2-Submissions. Accessed on April 19, 2019.
49 Bindel, N., Akleylek, S., Alkim, E., et al.. 'Lattice-based digital signature scheme qTESLA'. (NIST Post-Quantum Cryptography Standardization, 2019). Available from: https://csrc.nist.gov/Projects/Post-Quantum-Cryptography/Round-2-Submissions. Accessed on April 19, 2019.