On the Complexity of the LWR-Solving BKW Algorithm

Hiroki Okada[1], Atsushi Takayasu[2], Kazuhide Fukushima[1], Shinsaku Kiyomoto[1], and Tsuyoshi Takagi[2]

[1] KDDI Research, Inc.
Saitama, Japan
[2] The University of Tokyo
Tokyo, Japan
ir-okada@kddi-research.jp

Abstract. Duc *et al.* applied the Blum-Kalai-Wasserman (BKW) algorithm to the learning with rounding (LWR) problem. The number of blocks is a parameter of the BKW algorithm. By optimizing the number of blocks, we can minimize the time complexity of the BKW algorithm. However, Duc *et al.* did not derive the optimal number of blocks theoretically, but they searched it for numerically. In this paper, we theoretically derive the asymptotically optimal number of blocks and show the minimum time complexity of the algorithm. Furthermore, we derive an equation that relates the Gaussian parameter σ of the LWE problem and the modulus p of the LWR problem. When σ and p satisfy the equation, the asymptotic time complexity of the BKW algorithm to solve the LWE and LWR problems are the same.

Keywords: Lattice, Learning with Errors, Learning with Rounding, Blum-Kalai-Wasserman algorithm

1 Introduction

Background. In December 2016, the National Institute of Standards and Technology (NIST) initiated post-quantum cryptography (PQC) standardization. In the list of the round 1 submissions [33], there are several lattice-based schemes whose security are based on learning with errors (LWE) problem (e.g., [5, 6, 13, 17, 18]) and learning with rounding (LWR) problem (e.g., [9, 21, 24, 34]). Therefore, studies of the algorithm to solve the LWE and LWR problems are important for design and security analysis of post-quantum cryptosystems.

The LWE problem, which is an extension of the learning parity with noise (LPN) problem, is introduced by Regev [42]. An adversary of the LWE problem receives samples $(\boldsymbol{a}_j, \langle \boldsymbol{a}_j, \boldsymbol{s} \rangle + e_j) \in \mathbb{Z}_q^n \times \mathbb{Z}_q$ $(j = 1, 2, \dots)$ from the LWE oracle, where $\boldsymbol{a}_j$ is a uniformly random vector in $\mathbb{Z}_q^n$, $\boldsymbol{s}$ is a fixed secret vector in $\mathbb{Z}_q^n$, and $e_j \in \mathbb{Z}_q$ is a noise (usually, discrete Gaussian noise). The goal of the adversary is to recover secret vector $\boldsymbol{s}$. We note that the LPN problem has a fixed modulus

$q = 2$ and the noise follows the Bernoulli distribution. In [42], Regev presents a reduction from worst-case lattice problems to the average-case LWE problem.

We can classify algorithms for solving the LWE problem into two families. The first family uses lattice reduction techniques, which have been extensively studied (see, *e.g.*, [14, 20, 26, 30, 31, 38–41]). The expected complexity of these algorithms is often considered when parameters for LWE-based schemes are discussed, such as in [3]. The second family is tailor-made for the LPN and LWE problems without lattice reduction, which includes the main subject of this paper: the Blum-Kalai-Wasserman (BKW) algorithm [15]. The BKW algorithm can be described as a "block-wise" and addition-only variant of the standard Gaussian elimination. First, we separate the vector $\boldsymbol{a}_j \in \mathbb{Z}_q^n$ into a blocks: We can write $\boldsymbol{a}_j = (\boldsymbol{a}_{j,1}||\boldsymbol{a}_{j,2}||\dots||\boldsymbol{a}_{j,a})$, where $\boldsymbol{a}_{j,1}, \dots, \boldsymbol{a}_{j,a} \in \mathbb{Z}_q^{n/a}$, and then, by adding the samples together like the Gaussian elimination, we obtain "reduced" samples $\boldsymbol{a}'_j = (\boldsymbol{a}'_{j,1}||\boldsymbol{0}||\dots||\boldsymbol{0})$. As reported in [32], improved variants of the BKW algorithm such as [29,36] are some of the asymptotically fastest algorithms. Although some algorithm [37] based on lattice reduction outperforms these BKW algorithms for some parameter-sets (q, σ), it allows a heuristic [32].

The LWR problem is proposed by Banerjee *et al.* [12] with its reduction from the LWE problem. We can consider the LWR problem as a deterministic variant of the LWE problem in which the noise additions are replaced with deterministic rounding operations. An adversary of the LWR problem receives samples $(\boldsymbol{a}_j, \lceil \frac{p}{q}\langle \boldsymbol{a}_j, \boldsymbol{s}\rangle \rfloor) \in \mathbb{Z}_q^n \times \mathbb{Z}_p$ $(j = 1, 2, \dots)$ from the LWR oracle, where p is a rounding modulus such that $p < q$. Compared with LWE-based cryptographic schemes, LWR-based schemes can be simply implemented because they replace the rich Gaussian error sampling process of the LWE-based schemes with the rounding operations (which can be simply implemented by rounding off the lower-order bits). The LWR problem was initially applied to low-depth pseudorandom functions [10–12, 16], and there have been a number of applications, cf. lossy trapdoor functions [7], public-key cryptosystems [9, 21, 43] and key exchange protocol [34].

However, few studies have been performed on the complexity of the LWR problem, while the complexity of the LWE problem has been extensively studied. The complexity of the LWR problem is often estimated by adopting the LWE-solving algorithms to the LWR problem. In [3], Albrecht *et al.* estimates the cost of running primal and dual lattice attacks, which is based on lattice reduction techniques, against lattice-based schemes including LWE-based schemes and LWR-based schemes in the list of the round 1 submissions for the NIST PQC standardization [33]. They consider that the cost of lattice attacks for the LWR and LWE problem are the same when an equation $\sigma = \frac{q}{2\sqrt{3p}}$ holds, as considered in [22, 24]. This equation is simply derived by comparing the variance of the Gaussian noise of the LWE problem and the "rounding error" of the LWR problem. Note that the equation, which relates the hardness of the LWR problem and the LWE problem, is limited to attacks based on lattice reduction techniques, and it is not shown that the conversion equation can be applied for the BKW algorithm.

Previous Works. The BKW algorithm initially targeted the LPN problem, and its time complexity is sub-exponential in $2^{O(n/\log n)}$. Albrecht *et al.* [2] expanded it to solve the LWE problem whose time complexity is $q^{O(n/\log n)}$. Duc *et al.* improved Albrecht *et al.*'s BKW algorithm and also introduced its variant for the LWR problem, which was the first algorithmic analysis of the LWR problem. They showed that the time complexity of the LWR-solving algorithm is $q^{O(n/\log n)}$ when we choose the number of the block as $a = O(\log n)$. However, they did not show this choice of a is optimal; thus the minimum time complexity of the algorithm is not shown.

After the BKW algorithm proposed by Albrecht *et al.*, new variants of the BKW algorithm [1, 4, 28, 29, 36] for solving a *small-secret* LWE problem, whose secret vector $\boldsymbol{s}$ is extremely small (e.g. $\boldsymbol{s} \in \{0,1\}^n$), are proposed. These algorithms can be applied to the general LWE problem, whose secret vector $\boldsymbol{s}$ is uniform in $\mathbb{Z}_q^n$, by transforming the general LWE problem to *small-secret* LWE problem with a technique called *secret-error switching* [8, 19, 36], and it is shown that some of these algorithms [29, 36] solve the general LWE problem faster. However, it is not shown that these new type of the BKW algorithm can be applied to the LWR problem. In order to apply the *secret-error switching* technique to the LWR problem, we need to convert LWR samples into LWE samples with uniform error by substituting the LWR samples $(\boldsymbol{a}_j, \lceil \frac{p}{q}\langle \boldsymbol{a}_j, \boldsymbol{s}\rangle \rfloor)$ with $(\boldsymbol{a}_j, \frac{q}{p}\lceil \frac{p}{q}\langle \boldsymbol{a}_j, \boldsymbol{s}\rangle \rfloor)$, and, as mentioned in [36], solving this converted LWR problem with their algorithm is out of reach. On the other hand, Duc *et al.*'s LWR-solving BKW algorithm does not need to convert LWR samples into LWE samples; the algorithm is tailor-made for solving the LWR problem.

Our Contribution. We first review Duc *et al.*'s LWR-solving BKW algorithm, and then derive the time complexity in a simpler form. Subsequently, we theoretically derive the optimal choice of the number of blocks a that asymptotically minimize the time complexity, while Duc *et al.* searched numerically for the optimal a in [25]. Thus, an entirely theoretical analysis of the time complexity of the algorithm is shown in this paper: We show that the minimum time complexity of the BKW algorithm is $t = q^{O(n/\log n)}$ and the required number of samples is $m = q^{O(n/\log n)}$. We also confirm that the derived parameter is accurately optimal by showing the results of some concrete instances of the LWR problem, and that they fit the results given by Duc *et al.*

Furthermore, we derive a conversion equation between the Gaussian parameter σ in the LWE problem and the rounding modulus p in the LWR problem, by comparing the time complexity of the BKW algorithm for the LWE and LWR problems: We show that the time complexity of the BKW algorithm to solve the LWE problem and that to solve the LWR problem are the same when σ and p satisfy equation $\sigma = \frac{q}{2\sqrt{3p}}$. This equation coincides with the equation derived from the complexity analysis of the attacks based on lattice reduction techniques. Thus, our result means that the equation is applicable also for the complexity analysis based on the BKW algorithm.

2 Preliminaries

Notations. We denote the logarithm of base 2 and the natural logarithm as $\log(\cdot)$ and $\ln(\cdot)$, respectively. We denote the imaginary unit as i, and a real part of $x \in \mathbb{C}$ as $\mathrm{Re}(x)$. We let $\lceil \cdot \rfloor : \mathbb{R} \to \mathbb{Z}$ be the rounding function that rounds to the closest integer. (In the case of equality, we take the floor.) We define $\theta_q := e^{\frac{2\pi i}{q}}$ and also $\theta_p := e^{\frac{2\pi i}{p}}$. We write vectors in bold. By $\boldsymbol{a}_j$ we denote the j-th vector of the list of vectors. We denote a partial vector of a vector $\boldsymbol{a} = (a_1, a_2, \ldots, a_n)$ by $\boldsymbol{a}_{(k,l)} := (a_k, a_{k+1}, \ldots, a_l)$, where $1 \le k \le l \le n$. By $(\boldsymbol{a}||\boldsymbol{b})$ we denote the concatenation of two vectors $\boldsymbol{a}$ and $\boldsymbol{b}$. We denote by $\langle \cdot, \cdot \rangle$ the usual dot product of two vectors, and we define $\langle \cdot, \cdot \rangle_q := \langle \cdot, \cdot \rangle \pmod q$. We write $\boldsymbol{s} \xleftarrow{U} \mathcal{S}$ to denote the process of sampling $\boldsymbol{s}$ uniformly at random over $\mathcal{S}$, and we write $e \leftarrow \chi$ to denote the process of sampling e according to a probability distribution χ.

2.1 LWE and LWR Problem

The LWE Problem. We define the LWE oracle and the LWE problem.

Definition 1. *(LWE oracle). Let n, q be positive integers. Learning with Error (LWE) oracle $\mathsf{LWE}_{\boldsymbol{s},\chi}$ for a fixed vector $\boldsymbol{s} \in \mathbb{Z}_q^n$ and probability distribution χ over $\mathbb{Z}_q$ is an oracle returning $\left\{ (\boldsymbol{a}, c) \;\middle|\; c = \langle \boldsymbol{a}, \boldsymbol{s} \rangle + e \bmod q,\ \boldsymbol{a} \xleftarrow{U} \mathbb{Z}_q^n,\ e \leftarrow \chi \right\}$.*

For the distribution of noise χ, variants of the Gaussian distribution that is discretized into $\mathbb{Z}_q$ are used. In this paper, we consider two types of Gaussian distributions that are considered in [25]; the *rounded Gaussian distribution* $\bar{\Psi}_{\sigma,q}$ and the *discrete Gaussian distribution* $D_{\sigma,q}$. The *rounded Gaussian distribution* $\bar{\Psi}_{\sigma,q}$ is proposed in the initial LWE problem by Regev [42], and is also considered in [2, 27]. Its probability mass function for integer x in the interval $]-\frac{q}{2}, \frac{q}{2}]$, is given by $\Pr[x \leftarrow \bar{\Psi}_{\sigma,q}] = \int_{x-\frac{1}{2}}^{x+\frac{1}{2}} g(\theta; q, \sigma) d\theta$, where $g(\theta; q, \sigma)$ is the probability density function of the *wrapped Gaussian distribution* $\bar{\Psi}_{\sigma,q}$, which is defined by $g(\theta; q, \sigma) := \sum_{l=-\infty}^{\infty} \frac{1}{\sigma\sqrt{2\pi}} \exp\left(\frac{-(\theta+lq)^2}{2\sigma^2}\right)$, for $\theta \in]-\frac{q}{2}, \frac{q}{2}]$. The *discrete Gaussian distribution* $D_{\sigma,q}$ is used in most of the cryptographic applications of the LWE problem and in the classical LWE problem reduction [19]. This distribution is, for x an integer in $]-\frac{q}{2}, \frac{q}{2}]$, $\Pr[x \leftarrow D_{\sigma,q}] = \frac{\exp(-\frac{x^2}{2\sigma^2})}{\sum_{y \in]-\frac{q}{2}, \frac{q}{2}]} \exp(-\frac{y^2}{2\sigma^2})}$.

Definition 2. *(Search-LWE) The Search-LWE problem is the problem of recovering the hidden secret $\boldsymbol{s}$ given m samples $(\boldsymbol{a}_j, c_j) \in \mathbb{Z}_q^n \times \mathbb{Z}_q$ $(j = 1, 2, \ldots, m)$ received from $\mathsf{LWE}_{s,\chi}$.*

The LWR Problem. We define the LWR problem which is the main focus of this paper. For the purpose, we define an LWR Oracle in advance.

Definition 3. *(LWR oracle) Let n, q be natural numbers. Learning with Rounding (LWR) oracle $\mathsf{LWR}_{\boldsymbol{s},p}$ for a hidden vector $\boldsymbol{s} \in \mathbb{Z}_q^n$ and rounding modulus p is an oracle returning $\left\{ (\boldsymbol{a}, c) \;\middle|\; c = \left\lceil \frac{p}{q} \langle \boldsymbol{a}, \boldsymbol{s} \rangle_q \right\rfloor,\ \boldsymbol{a} \xleftarrow{U} \mathbb{Z}_q^n \right\}$.*

Definition 4. *(LWR problem). The LWR problem is the problem of recovering the hidden secret* $\boldsymbol{s}$ *given* m *samples* $(\boldsymbol{a}_j, c_j) \in \mathbb{Z}_q^k \times \mathbb{Z}_q$ $(j = 1, 2, \ldots, m)$ *received from* $\mathsf{LWR}_{\boldsymbol{s},p}$.

The rounding calculation in the LWR sample generates a "rounding error," which is similar to the Gaussian noise added in the LWE sample. Duc *et al.* proved that "rounding error" follows a uniform distribution, in Lemma 19 in [25].

Lemma 1. *(Lemma 19. in [25]) Let* n *and* $q > p \geq 2$ *be positive integers,* q *prime. Let* $(\boldsymbol{a}, c)$ *be a random sample from an LWR oracle* $\mathsf{LWR}_{\boldsymbol{s},p}$. *Then, the "rounding error," given by*

$$\xi = \frac{p}{q}\langle \boldsymbol{a}, \boldsymbol{s} \rangle_q - c, \tag{1}$$

follows the uniform distribution in a discrete subset of $[-\frac{1}{2}, \frac{1}{2}]$ *with mean zero. Furthermore, the characteristic function of* ξ, *for* $t \in \mathbb{R}_{\neq 0}$, *is*

$$\phi_\xi(t) := E[e^{\pm it\xi}] = \frac{\sin(\frac{t}{2})}{q \sin(\frac{t}{2q})}. \tag{2}$$

Banerjee *et al.*, showed a reduction from the LWE problem to the LWR problem, in the paper [12] in which they first introduced the LWR problem. Note that the decision version of the LWE (or LWR) problem can be described as follows: given m samples of the form $(\boldsymbol{a}, c) \in \mathbb{Z}_q^n \times \mathbb{Z}_q$ (or $\mathbb{Z}_p$), where $\boldsymbol{a} \xleftarrow{U} \mathbb{Z}_q^n$, distinguish whether $c \xleftarrow{U} \mathbb{Z}_q$ (or $\mathbb{Z}_p$) or $c = \langle \boldsymbol{a}, \boldsymbol{s} \rangle + e$ (or $c = \left\lfloor \frac{p}{q}\langle \boldsymbol{a}, \boldsymbol{s} \rangle_q \right\rceil$), for a fixed secret $\boldsymbol{s} \in \mathbb{Z}_q^n$.

Theorem 1. *(Theorem 3.2 in [12]) Let* $\beta \in \mathbb{R}_+$, χ *be any efficiently samplable distribution over* $\mathbb{Z}$ *such that* $\Pr_{x \leftarrow \chi}[|x| > \beta]$ *is negligible, and let* $q \geq p \cdot \beta \cdot n^{\omega(1)}$. *Then, solving decision-LWR with secrets of size* n *and parameters* p *and* q *is at least as hard as solving decision-LWE over* $\mathbb{Z}_q^n$ *with noise distribution* χ.

Alwen *et al.* [7] also showed the reduction without the super-polynomial parameters, but it limits the number of samples that the LWR oracle allows the adversary to receive.

Theorem 2. *(Theorem 4.1 from [7]) Let* λ *be the security parameter. Let* n, l, m, p, γ *be positive integers,* p_{max} *be the largest prime divisor of* q, *and* $p_{max} \geq 2\beta\gamma nmp$. *Let* χ *be the probability distribution over* $\mathbb{Z}$ *such that the average absolute value of* $x \leftarrow \chi$ *is less than* β. *Then, if* $n \geq (l+\lambda+1)\log(q)/\log(2\gamma)+2\lambda$ *and if* $\gcd(q, q/p_{max}) = 1$, *the decision-LWR with secrets of size* n, *parameters* p *and* q *and limited to* m *queries is at least as hard as solving decision-LWE over* $\mathbb{Z}_q^l$ *with noise distribution* χ *and limited to* m *queries.*

2.2 Duc *et al.*'s BKW Algorithm for the LWR Problem

We recall Duc *et al.*'s BKW algorithm to solve the LWR problem. The BKW algorithm consists of three stages: 1) Sample reduction, 2) Hypothesis testing,

Algorithm 1 The BKW algorithm to solve the LWR problem [25]

Input: natural numbers a, b $(ab = n), m$, and samples $(\boldsymbol{a}_j, c_j)$, $(j = 1, 2, \ldots, m)$. We represent the set of samples as $\mathcal{S} := \{(\boldsymbol{a}_j, c_j)\}_{j=1}^{m}$.
Output: $\boldsymbol{s}$.

(Stage 1: Sample reduction.)
for $l = 0$ to $a - 2$ **do**
  $\mathcal{S}' \leftarrow \phi$: empty set
  $\mathcal{T}_l \leftarrow \phi$
  repeat
    extract one sample $(\boldsymbol{a}, c)$ from $\mathcal{S}$.
    if $\boldsymbol{a}_{(b(a-l-1)+1, b(a-l))} = \boldsymbol{0}$ **then**
      $\mathcal{S}' \leftarrow \mathcal{S}' \cup (\boldsymbol{a}, c)$
    else if there is $(\boldsymbol{a}', c') \in \mathcal{T}_l$ such that $(\boldsymbol{a} \pm \boldsymbol{a}')_{(b(a-l-1)+1, b(a-l))} = \boldsymbol{0}$ **then**
      $\mathcal{S}' \leftarrow \mathcal{S}' \cup (\boldsymbol{a} \pm \boldsymbol{a}', c \pm c')$
    else
      $\mathcal{T}_l \leftarrow \mathcal{T}_l \cup (\boldsymbol{a}, c)$
    end if
  until $\mathcal{S} = \phi$
  $\mathcal{S} \leftarrow \mathcal{S}'$
end for
$\mathcal{T}_{a-1} \leftarrow \mathcal{S}$
for $l = a - 1$ to 0 **do**
  (Stage 2: Hypothesis testing.)
  Let $m_l := \#\mathcal{T}_l$, and we denote by $(\overline{\boldsymbol{a}}_j^l, \overline{c}_j^l)$ the samples included in $\mathcal{T}_l$.
  Calculate $f(\boldsymbol{y}) := \sum_{j=1}^{m_l} \mathbb{1}_{\{\overline{\boldsymbol{a}}_j^l = \boldsymbol{y}\}} \theta_p^{\overline{c}_j^l}$ for all $\boldsymbol{y} \in \mathbb{Z}_q^b$
  Calculate the DFT of f, which is $\hat{f}(\boldsymbol{z}) = \sum_{j=1}^{m_l} \theta_p^{-(\frac{p}{q}\langle \overline{\boldsymbol{a}}_j^l, \boldsymbol{z}\rangle - \overline{c}_j^l)}$.
  Calculate $\boldsymbol{s}_{(b(a-l-1)+1, b(a-l)))} \leftarrow \mathrm{argmax}_{\boldsymbol{z}}\ \mathrm{Re}(\hat{f}(\boldsymbol{z}))$.
  (Stage 3: Back substitution.)
  Using the obtained $\boldsymbol{s}_{(bl+1, b(l+1)))}$, update sets $\mathcal{T}_{l'}$ for $0 \le l' < l$.
end for
return $(\boldsymbol{s}_{(1,b)} || \boldsymbol{s}_{(b+1,2b)} || \ldots || \boldsymbol{s}_{((a-1)b+1,ab)})$

and 3) Back substitution. For simplicity, in this paper, we consider only the case that the number of blocks a and the block length b satisfy $ab = n$. Algorithm 1 shows an overview of the algorithm.

Stage 1: Sample reduction. We receive m samples $\{(\boldsymbol{a}_j, c_j)\}_{j=1}^{m}$ from LWR oracle $\mathsf{LWR}_{\boldsymbol{s},p}$, and represent the set of samples as $\mathcal{S} := \{(\boldsymbol{a}_j, c_j)\}_{j=1}^{m}$. We separate the vector $\boldsymbol{a}_j \in \mathbb{Z}_q^n$ into a blocks whose length are b: We can write $\boldsymbol{a}_j = (\boldsymbol{a}_{j\,(1,b)} || \boldsymbol{a}_{j\,(b+1,2b)} || \ldots || \boldsymbol{a}_{j\,((a-1)b+1,ab)})$. In Stage 1, our goal is to produce samples whose elements are all zero except for the first block, with addition or subtraction of pairs of samples. For $l = 0$, we extract a sample $(\boldsymbol{a}, c)$ from $\mathcal{S}$, and search another sample $(\boldsymbol{a}', c')$ such that $(\boldsymbol{a} \pm \boldsymbol{a}')_{((a-1)b+1,ab)} = \boldsymbol{0}$, then we store the sample $(\boldsymbol{a} \pm \boldsymbol{a}', c \pm c')$ in the temporary set $\mathcal{S}'$. If a sample $(\boldsymbol{a}, c)$ already holds $\boldsymbol{a}_{((a-1)b+1,ab)} = \boldsymbol{0}$, we directly store it in $\mathcal{S}'$. If we cannot find the sample

$(\boldsymbol{a}', c')$ such that $(\boldsymbol{a} \pm \boldsymbol{a}')_{((a-1)b+1,ab)} = \boldsymbol{0}$, we store the sample $(\boldsymbol{a}, c)$ in $\mathcal{T}_0$. After we finish extracting samples and empty the set $\mathcal{S}$, we renew $\mathcal{S} \leftarrow \mathcal{S}'$ and move on to the next step for $l = 1$. In this manner, we recursively generate the sets $\mathcal{T}_l$ for $0 \le l \le a-2$, and then we set $\mathcal{T}_{a-1} \leftarrow \mathcal{S}$ in the end. Note that the samples $(\boldsymbol{a}, c)$ in $\mathcal{T}_l$ hold $\boldsymbol{a}_{((a-l)b+1,ab)} = \boldsymbol{0}$ (except for $l = 0$). In particular, the samples $(\boldsymbol{a}, c)$ in $\mathcal{T}_{a-1}$ hold $\boldsymbol{a}_{(b+1,ab)} = \boldsymbol{0}$. We may think of the reduced samples in T_{a-1} as the set of samples of the b-dimensional LWR problem, although the variance of their noise is larger than those of the original samples. Hereinafter, the samples in $\mathcal{T}_l$ are termed "reduced samples", and represent $\mathcal{T}_l = \{(\overline{\boldsymbol{a}}_j^l, \overline{c}_j^l)\}_{j=1}^{m_l}$, where $m_l := \#\mathcal{T}_l$. Note that the maximum number of samples whose $(a-l)$-th block cannot vanish is $\frac{q^b-1}{2}$, and the minimum (worst) number of reduced samples in $\mathcal{T}_{a-1}$ (i.e. minimum value of m_{a-1}) is

$$m' = m - (a-1)\frac{q^b - 1}{2}. \tag{3}$$

Stage 2: Hypothesis testing. For simplicity, we explain Stage 2 and Stage 3 only for $l = a-1$. (In Section 3.1, we only consider the time complexity to recover $\boldsymbol{s}_{(1,b)}$ because the whole time complexity of the algorithm is at most a positive constant multiple of it). For simplicity of notation, we define a b-dimensional vector $\overline{\boldsymbol{a}}_j := (\overline{\boldsymbol{a}}_j^{a-1})_{(1,b)}$, and denote $\overline{c}_j := \overline{c}_j^{a-1}$. The goal of this stage is to estimate the first b elements of $\boldsymbol{s}$, denoted as $\boldsymbol{s}_{(1,b)}$. We define the function $f(\boldsymbol{y}) := \sum_{j=1}^{m'} \mathbb{1}_{\{\overline{\boldsymbol{a}}_j = \boldsymbol{y}\}} \theta_p^{\overline{c}_j}$, where $\boldsymbol{y} \in \mathbb{Z}_q^b$, $\theta_p := e^{\frac{2\pi i}{p}}$, and $\mathbb{1}_{\{\overline{\boldsymbol{a}}_j = \boldsymbol{y}\}}$ is 1 when $\boldsymbol{a}_j = \boldsymbol{y}$ is true and 0 otherwise. The discrete Fourier transform of f is

$$\hat{f}(\boldsymbol{z}) := \sum_{\boldsymbol{y} \in \mathbb{Z}_q^b} f(\boldsymbol{y}) \theta_q^{-\langle \boldsymbol{y}, \boldsymbol{z} \rangle} = \sum_{j=1}^{m'} \theta_p^{-(\frac{p}{q}\langle \overline{\boldsymbol{a}}_j, \boldsymbol{z} \rangle - \overline{c}_j)}. \tag{4}$$

Then, we search the max $\mathrm{Re}(\hat{f}(\boldsymbol{z}))$, and output $\boldsymbol{s}_{(1,b)} = \mathrm{argmax}_{\boldsymbol{z}}\, \mathrm{Re}(\hat{f}(\boldsymbol{z}))$. We explain how the output $\mathrm{argmax}_{\boldsymbol{z}}\, \mathrm{Re}(\hat{f}(\boldsymbol{z}))$ estimates the secret vector. We define the "rounding error" of the reduced samples $\{(\overline{\boldsymbol{a}}_j, \overline{c}_j)\}_{j=1}^{m'}$ by $\overline{\xi}_j := \frac{p}{q}\langle \overline{\boldsymbol{a}}_j, \boldsymbol{s}_{(1,b)} \rangle - \overline{c}_j$, as like (1). Recall that the $\overline{\boldsymbol{a}}_j$ is produced by $a-1$ times of the "tree-like" addition of the original samples in the process of Stage 1, i.e. $\overline{\boldsymbol{a}}_j$ is the sum of the 2^{a-1} original samples, thus we can write $\overline{\boldsymbol{a}}_j = (\boldsymbol{a}_{j,1} \pm \boldsymbol{a}_{j,2} \pm \cdots \pm \boldsymbol{a}_{j,2^{a-1}})_{(1,b)}$, where $\boldsymbol{a}_{j,1}, \ldots, \boldsymbol{a}_{j,2^{a-1}}$ are the original samples. Similarly, we can write $\overline{c}_j = c_{j,1} \pm c_{j,2} \pm \cdots \pm c_{j,2^{a-1}}$, and obtain

$$\begin{aligned}
\overline{\xi}_j &= \frac{p}{q}\langle \boldsymbol{a}_{j,1} \pm \boldsymbol{a}_{j,2} \pm \cdots \pm \boldsymbol{a}_{j,2^{a-1}}, \boldsymbol{s} \rangle - (c_{j,1} \pm c_{j,2} \pm \cdots \pm c_{j,2^{a-1}}) \\
&= \sum_{k=1}^{2^{a-1}} \frac{p}{q}\langle \boldsymbol{a}_{j,k}, \boldsymbol{s} \rangle - c_{j,k} = \sum_{k=1}^{2^{a-1}} \xi_{j,k},
\end{aligned}$$

where the $\xi_{j,k}$ are independent rounding errors from original samples. From the above equation and (4), when $\boldsymbol{z} = \boldsymbol{s}_{(1,b)}$, we obtain $\hat{f}(\boldsymbol{s}_{(1,b)}) =$

$\sum_{j=1}^{m'} \theta_p^{-(\sum_{k=1}^{2^{a-1}} \xi_{j,k})}$. On the other hand, when $\boldsymbol{z} \neq \boldsymbol{s}_{(1,b)}$, $\frac{p}{q}\langle \overline{\boldsymbol{a}}_j, \boldsymbol{z}\rangle - \overline{c}_j$ distribute uniformly in $]0, p]$.

Thus, when we select an appropriate value of parameter a such that the sum of the rounding errors $\sum_{k=1}^{2^{a-1}} \xi_{j,k}$ does not grow too large, $\mathrm{Re}(\hat{f}(\boldsymbol{s}))$ is so much larger than $\mathrm{Re}(\hat{f}(\boldsymbol{z}))$ that the hypothesis test succeeds with high probability.

Stage 3: Back substitution. Using the obtained $\boldsymbol{s}_{(1,b)}$, update the sets $\mathcal{T}_l$ by zeroing-out b elements in each sample: Replace all $(\boldsymbol{a}, c) \in \mathcal{T}_{l'}$ for $0 \leq l' < a-1$ with $(\boldsymbol{a}', c')$, where $\boldsymbol{a}' = (\boldsymbol{0}||\boldsymbol{a}_{(b+1,n)}) \in \mathbb{Z}_q^n$, $c' = c - \frac{p}{q}\langle \boldsymbol{a}_{(1,b)}, \boldsymbol{s}_{(1,b)}\rangle_q \in \mathbb{Z}_p$. Then back to Stage 2 to obtain $\boldsymbol{s}_{(b+1,2b)}$.

Repeating a rounds of Stages 2 to 3, we estimate $\boldsymbol{s}_{(1,b)}$, $\boldsymbol{s}_{(b+1,2b)}$, $\ldots$, $\boldsymbol{s}_{(a(b-1)+1,ab)}$, and obtain $\boldsymbol{s} = (\boldsymbol{s}_{(1,b)}||\boldsymbol{s}_{(b+1,2b)}||\ldots||\boldsymbol{s}_{((a-1)b+1,ab)})$.

3 Analysis of BKW Algorithm for the LWR Problem

We derive the minimum time complexity and the minimum number of required samples, by optimizing the number of blocks a which is a parameter of the BKW algorithm.

As with Duc *et al.*, in this paper, we consider only the case that the block length b satisfy $n = ab$, for simplicity. Therefore, the block length b is determined by the number of blocks a, as $b = n/a$. Note that the complexity of the BKW algorithm for the general case, where $n = (a-1) \cdot b + n'$ and $n' < b$, is asymptotically the same with that for the case where $ab = n$. We always consider q to be a prime, and $q > p > 4$ because we need the condition to prove Lemma 3.

In Section 3.1, we analyze the time complexity of the BKW algorithm for solving the LWR problem, using a as a parameter. In Section 3.2, we derive the optimal value of a that asymptotically minimizes the asymptotic time complexity. In Section 3.3, we calculate the concrete time complexity of the BKW algorithm for several LWR instances, and confirm that the optimal value of a minimizes the time complexity of the algorithm. Furthermore, in Section 4.1, we derive a equation that relates σ of the LWE problem and p of the LWR problem. When σ and p satisfy the equation, the asymptotic time complexity of the BKW algorithm to solve the LWE and LWR problems are the same.

3.1 Complexity Analysis

We analyze the time complexity and required number of samples to solve the LWR problem. We asymptotically analyze the time complexity and make it in a simple form so that we can theoretically derive the optimal number of blocks a in 3.2. We first refer to Lemma 2 (Theorem 23. in [25]), which is the analysis of the minimum number of samples needed to solve the LWR problem.

Lemma 2. *(Theorem 23. in [25]) We define the probability that the algorithm cannot recover the correct answer* $\epsilon := \Pr\left[\mathrm{argmax}_{\boldsymbol{z}} \mathrm{Re}\left(\hat{f}(\boldsymbol{z})\right) \neq \boldsymbol{s}_{(1,b)}\right]$. *Then,*

the number of samples required to solve the LWR problem with oracle $\mathsf{LWR}_{\boldsymbol{s},p}$ *is*

$$m^{\mathsf{LWR}} = \frac{8n}{a} \ln\left(\frac{q}{\epsilon}\right) \left((R_{q,p})^{2^{a-1}} - \left(\frac{3}{p}\right)^{2^{a-1}} \right)^{-2} + (a-1)\frac{q^{n/a}-1}{2}, \tag{5}$$

where $R_{q,p} := \frac{\sin\left(\frac{\pi}{p}\right)}{q\sin\left(\frac{\pi}{pq}\right)}$.

Note that $R_{q,p}$ is derived based on the characteristic function of the "rounding error" given in (2): $R_{q,p} = \phi_\xi(\frac{2\pi}{p})$ holds. As discussed later, this m^{LWR} in (5) is the main term of the time complexity of the algorithm.

In the following Lemma 3, we analyze the asymptotic behavior of the complicated part of the m^{LWR}. We describe it in a simpler form in order to enable the analysis of the minimum time complexity, which will be given later in Section 3.2. Note that we use the error rate of the LWR sample $\alpha_{\mathrm{lwr}} := \frac{1}{p}\sqrt{\frac{\pi}{6}}$ [21] to describe the time complexity for simplicity of notation.

Lemma 3. *Let* $\alpha_{\mathrm{lwr}} := \frac{1}{p}\sqrt{\frac{\pi}{6}}$. *When* $q > p > 4$, *we have*

$$\left((R_{q,p})^{2^{a-1}} - \left(\frac{3}{p}\right)^{2^{a-1}} \right)^{-2} = \exp(\pi\alpha_{\mathrm{lwr}}^2 2^a) + O\left(\frac{1}{p^2q^2}\right). \tag{6}$$

Proof. First, we prove that

$$\left((R_{q,p})^{2^{a-1}} - \left(\frac{3}{p}\right)^{2^{a-1}} \right)^{-2} = (R_{q,p})^{-2^a} + O\left(\frac{1}{p^{2^{a-1}}}\right) \tag{7}$$

holds. Note that $q > p > 4$. We obtain $R_{q,p} = \frac{\frac{p}{\pi}\sin\left(\frac{\pi}{p}\right)}{\frac{pq}{\pi}\sin\left(\frac{\pi}{pq}\right)} \geq \frac{p}{\pi}\sin\left(\frac{\pi}{p}\right)$. Since $\frac{p}{\pi}\sin\left(\frac{\pi}{p}\right)$ is monotonically increasing when $p > 4$, we obtain $R_{q,p} > \frac{4}{\pi}\sin\left(\frac{\pi}{4}\right) = 0.9003\cdots$, and $R_{q,p} > \frac{3}{p}$. Let $x = (R_{q,p})^{2^{a-1}}$ and $y = \left(\frac{3}{p}\right)^{2^{a-1}}$, then we have $\frac{y}{x} < 1$. Using Taylor expansion, we obtain $(x-y)^{-2} = \frac{1}{x^2}\left(1 + O(\frac{y}{x})\right)$. Therefore, (7) holds.

Next, we derive (6) from (7). Using Taylor expansion, we obtain $R_{q,p} = 1 - \frac{\pi^2}{6p^2} + O\left(\frac{1}{p^2q^2}\right) = 1 - \pi\alpha_{\mathrm{lwr}}^2 + O\left(\frac{1}{p^2q^2}\right)$, and $R_{q,p} - \exp\left(-\pi\alpha_{\mathrm{lwr}}^2\right) = O\left(\frac{1}{p^2q^2}\right)$. Consequently, from this equation, we obtain

$$\begin{aligned}
(R_{q,p})^{-2^a} &= \left(\exp\left(-\pi\alpha_{\mathrm{lwr}}^2\right) + O\left(\frac{1}{p^2q^2}\right)\right)^{-2\cdot 2^{a-1}} \\
&= \left(\exp\left(\pi\alpha_{\mathrm{lwr}}^2 \cdot 2\right)\left(1 + O\left(\frac{1}{p^2q^2}\right)\right)\right)^{2^{a-1}} \\
&= \exp\left(\pi\alpha_{\mathrm{lwr}}^2 2^a\right)\left(1 + O\left(\frac{1}{p^2q^2}\right)\right)^{2^{a-1}} \\
&= \exp\left(\pi\alpha_{\mathrm{lwr}}^2 2^a\right) + O\left(\frac{1}{p^2q^2}\right).
\end{aligned} \tag{8}$$

Thus, using (7) and (8), we have (6). □

We can now derive the number of required samples and the time complexity of the algorithm.

Theorem 3. *Let n and $q > p > 4$ be positive integers, q prime, and a be a natural number. Fix $\epsilon \in (0,1)$. When at least $m^{\mathsf{LWR}} = \mathrm{poly}(\exp(\pi\alpha_{\mathrm{lwr}}^2 2^a), q^{n/a})$ samples are given by LWR oracle $\mathsf{LWR}_{\boldsymbol{s},p}$, the time complexity of the BKW algorithm to recover secret $\boldsymbol{s}$ with a probability of at least $1-\epsilon$ is $t^{\mathsf{LWR}} = \mathrm{poly}(\exp(\pi\alpha_{\mathrm{lwr}}^2 2^a), q^{n/a})$, where $\alpha_{\mathrm{lwr}} = \frac{1}{p}\sqrt{\frac{\pi}{6}}$.*

Proof. From Lemmas 2 and 3, the number of required samples to solve the LWR problem is $m^{\mathsf{LWR}} = \frac{8n}{a}\ln\left(\frac{q}{\epsilon}\right)\left(\exp(\pi\alpha_{\mathrm{lwr}}^2 2^a) + O\left(\frac{1}{p^2q^2}\right)\right) + (a-1)\frac{q^{n/a}-1}{2}$. Recall that the number of the "reduced" samples we obtain after Stage 1 is $m' = m^{\mathsf{LWR}} - (a-1)\frac{q^{n/a}-1}{2} = \frac{8n}{a}\ln\left(\frac{q}{\epsilon}\right)\left(\exp(\pi\alpha_{\mathrm{lwr}}^2 2^a) + O\left(\frac{1}{p^2q^2}\right)\right)$, which is defined in (3).

In Stage 1, since we apply the addition for $O(m^{\mathsf{LWR}})$ samples in $\mathbb{Z}_q^n$ for $a-1$ times, the time complexity is $t_1 = O(anm^{\mathsf{LWR}})$. In Stage 2, We first calculate $f(\boldsymbol{y}) := \sum_{j=1}^{m'} \mathbb{1}_{\{\overline{\boldsymbol{a}}_j = \boldsymbol{y}\}}\theta_p^{\overline{c}_j}$, for all $\boldsymbol{y} \in \mathbb{Z}_q^b$. Since we need only to calculate $f(\boldsymbol{y})$ for $\boldsymbol{y} \in \{\overline{\boldsymbol{a}}_j\}_{j=1}^{m'}$, the time complexity for calculating $f(\boldsymbol{y})$ is $O(m') = O(\exp(\pi\alpha_{\mathrm{lwr}}^2 2^a)(n/a)\ln q)$. After that, we compute the DFT of f, the complexity of which is $O(q^{n/a}(n/a)\ln q)$. Finally, we search max $\hat{f}(\boldsymbol{z})$ defined in (4) for all $\boldsymbol{z} \in \mathbb{Z}_q^{n/a}$, the time complexity of which is $O(q^{n/a}n/a)$. Therefore, the time complexity of Stage 2 is $t_2 = O(\exp(\pi\alpha_{\mathrm{lwr}}^2 2^a)(n/a)\ln q) + O(q^{n/a}(n/a)\ln q)$. In Stage 3, since we update all samples stored in $\mathcal{T}_{l'}$ $(0 \le l' < a-1)$ (the total number of these samples is $m^{\mathsf{LWR}} - m'$) with inner product calculation of the vectors in $\mathbb{Z}_q^{n/a}$, the time complexity of Stage 3 is $t_3 = O((m^{\mathsf{LWR}} - m')n/a) = O(q^{n/a}n/a)$. Therefore, the time complexity of the BKW algorithm is $t^{\mathsf{LWR}} = t_1 + t_2 + t_3 = O(\exp(\pi\alpha_{\mathrm{lwr}}^2 2^a)(n/a)\ln q) + O(q^{n/a}(n/a)\ln q) = \mathrm{poly}(\exp(\pi\alpha_{\mathrm{lwr}}^2 2^a), q^{n/a})$. □

3.2 Parameter Optimization

We analyze the optimal choice for input parameter a to asymptotically minimize the asymptotic time complexity of the BKW algorithm to solve the LWR problem. Furthermore, we analyze the minimum time complexity.

Theorem 4. *(Optimal choice of a) The optimal parameter a that asymptotically minimizes the asymptotic time complexity of the algorithm to solve the LWR problem is*

$$a = \left\lfloor \frac{1}{\ln 2} W\left(\frac{n\ln q \ln 2}{\pi\alpha_{\mathrm{lwr}}^2}\right)\right\rfloor \tag{9}$$

where W is Lambert W function [23].

Proof. From Theorem 3, we obtain the time complexity $t = O(\exp(\pi\alpha_{\mathrm{lwr}}^2 2^a)(n/a)\ln q) + O(q^{n/a}(n/a)\ln q)$. Note that $\exp(\pi\alpha_{\mathrm{lwr}}^2 2^a)$ monotonically increases and $q^{n/a}$ monotonically decreases, as a increases. Therefore, the time complexity is asymptotically minimized[1] when a satisfies

$$\exp(\pi\alpha_{\mathrm{lwr}}^2 2^a) = q^{n/a}. \tag{10}$$

From (10), by simple arithmetic, we obtain $(\ln 2)ae^{(\ln 2)a} = \frac{n\ln q\ln 2}{\pi\alpha_{\mathrm{lwr}}^2}$. To solve this equation for a, we use the Lambert W function, which satisfies $W(ze^z) = z$. We obtain $W((\ln 2)ae^{(\ln 2)a}) = (\ln 2)a$, and we obtain (9). □

Since the Lambert function $W(x)$ has an asymptotic form as $W(x) = \ln(x) - \ln(\ln(x)) + o(1)$, we can evaluate $a = \frac{1}{\ln 2}\left(\ln\left(\frac{n\ln q\ln 2}{\pi\alpha_{\mathrm{lwr}}^2}\right) - \ln\ln\left(\frac{n\ln q\ln 2}{\pi\alpha_{\mathrm{lwr}}^2}\right)\right) + o(1)$. Furthermore, when we consider q to be at most exponential of n (this range of q includes most of q used in LWE cryptosystems), we obtain $\log q = O(n)$, and $a = O(\log n)$. Using this value, (10), and Theorem 3, we obtain the corollary below.

Corollary 1. *(Minimum time complexity) Let n and $q > p > 4$ be positive integers, q prime. Let $a = \left\lfloor \frac{1}{\ln 2} W\left(\frac{n\ln q\ln 2}{\pi\alpha_{\mathrm{lwr}}^2}\right)\right\rfloor$, where $\alpha_{\mathrm{lwr}} = \frac{1}{p}\sqrt{\frac{\pi}{6}}$. Fix $\epsilon \in (0,1)$. When at least $q^{O(n/\log n)}$ samples are given by LWR oracle $\mathsf{LWR}_{\boldsymbol{s},p}$, the time complexity of the BKW algorithm to recover secret $\boldsymbol{s}$ with a probability of at least $1-\epsilon$ is $q^{O(n/\log n)}$.*

3.3 Concrete Analysis

Table 1 shows the concrete time complexity of the BKW algorithm. We denote the time complexity of the LWR-solving BKW algorithm by $\mathcal{C}^{\mathsf{LWR}}$. Then, similar to Theorem 17 in [25], we obtain

$$\begin{aligned}\mathcal{C}^{\mathsf{LWR}} &= \frac{1}{4}(a-2)(a-1)(2n/a+1)(q^{n/a}-1) + nq^{n/a}\log(q) \\ &\quad + \sum_{j=0}^{a-1} m'^{\mathsf{LWR}}_{j,\epsilon}\left(\frac{a-1-j}{2}(n+2)+2\right),\end{aligned} \tag{11}$$

where $m'^{\mathsf{LWR}}_{j,\epsilon} := \frac{8n}{a}\ln\left(\frac{q}{\epsilon}\right)\left(R_{q,p}^{2^{a-1-j}} - \left(\frac{3}{p}\right)^{2^{a-1-j}}\right)^{-2}$. We use the same parameters n, q and p as in Table 2 in [25], : For type (a), $q = \mathrm{nextprime}(\lceil(2\sigma n)^3\rceil)$, $p = \mathrm{nextprime}(\lceil\sqrt[3]{q}\rceil)$ and for type (b), $p = 13$, $q =$

[1] Let $\tilde{a}$ satisfies $\exp(\pi\alpha_{\mathrm{lwr}}^2 2^{\tilde{a}}) = q^{n/\tilde{a}}$, and Let $t_{\tilde{a}}$ be the time complexity with $a = \tilde{a}$, namely $t_{\tilde{a}} = O(\exp(\pi\alpha_{\mathrm{lwr}}^2 2^{\tilde{a}})(n/a)\ln q)$. If we set $a > \tilde{a}$, then we obtain $t_a = O(\exp(\pi\alpha_{\mathrm{lwr}}^2 2^a)(n/a)\ln q)$, and $t_a > t_{\tilde{a}}$ since $\exp(\pi\alpha_{\mathrm{lwr}}^2 2^a) > \exp(\pi\alpha_{\mathrm{lwr}}^2 2^{\tilde{a}})$. If we set $a < \tilde{a}$, then we obtain $t = O(q^{n/a}(n/a)\ln q)$, and $t_a > t_{\tilde{a}}$ since $q^{n/a} > q^{n/\tilde{a}}$. Therefore, $\tilde{a}$ is asymptotically optimal.

Table 1. The worst case time complexity ($\mathcal{C}^{\mathsf{LWR}}$) and the number of required samples (m^{LWR}) for the LWR-solving BKW algorithm. We also provide the value of a theoretically derived in (9), which asymptotically approaches the optimal value that minimizes the complexity. In this table, "*" means the value is optimal, and "†" means the value is not optimal. The optimal values are shown in parenthesis.

(type)	n	q	p	a	$\log(\mathcal{C}^{\mathsf{LWR}})$	$\log(m^{\mathsf{LWR}})$
	32	6318667	191	19^*	51.00	42.70
	40	23166277	293	20^*	60.66	52.18
	64	383056211	733	$24^\dagger$ (23)	$92.70^\dagger$ (92.10)	$83.08^\dagger$ (82.80)
(a)	80	1492443083	1151	25^*	110.82	101.11
	96	4587061889	1663	26^*	132.17	122.15
	112	11942217841	2287	28^*	148.00	137.68
	128	27498355153	3023	29^*	167.44	156.88
	32	2411	13	11^*	44.53	37.00
	40	3709	13	11^*	53.24	45.44
	64	9461	13	12^*	81.48	72.92
(b)	80	14867	13	12^*	103.76	94.86
	96	21611	13	12^*	126.83	117.66
	112	29717	13	13^*	140.08	130.63
	128	39241	13	13^*	162.50	152.84

nextprime($\lceil 2\sigma np \rceil$), where $\sigma = \frac{n^2}{\sqrt{2\pi n}(\log(n))^2}$. These parameters are selected based on Corollary 4.2 in [7], which follows Theorem 2. Type (a) parameters maximize the efficiency, and type (b) parameters minimize the modulus to error ratio (q/σ). Note that we also ignored the constraint on the number of samples m as Duc *et al.* did. We set $a = \left\lfloor \frac{1}{\ln 2} W\left(\frac{n \ln q \ln 2}{\pi \alpha_{\mathrm{lwr}}^2}\right) \right\rfloor$ and calculate m^{LWR} and $\mathcal{C}^{\mathsf{LWR}}$ in (5) and (11), respectively. We also set $\epsilon = 0.01$ in Table 1, following the setting given in Table 2 of [25].

In Table 1, we can observe that our choice of the number of blocks a asymptotically (but almost completely) minimizes the time complexity of the algorithm.

4 Comparison between the LWE and LWR problems

4.1 Relation between σ and p

In this section, we compare the time complexity of the BKW algorithm to solve the LWE and LWR problem, and then derive a relation between p in the LWR problem and σ in the LWE problem.

In Theorem 3, we showed that the time complexity of the BKW algorithm to solve the LWR problem is $\mathrm{poly}(\exp(\pi\alpha_{\mathrm{lwr}}^2 2^a), q^{n/a})$. On the other hand, based on Theorem 16 in [25], Kaminakaya *et al.* [35] analyzed the time complexity of the BKW algorithm to solve the LWE problem, and showed that the complexity is $\mathrm{poly}(\exp(\pi\alpha_{\mathrm{lwe}}^2 2^a), q^{n/a})$, where $\alpha_{\mathrm{lwe}} := \frac{\sqrt{2\pi}\sigma}{q}$. We will describe the result later

in Lemma 5 and refer to the proof given in [35]. As a preparation, we refer to the Theorem 16 in [25], which shows the number of samples required to solve the LWE problem:

Lemma 4. *(Theorem 16. in [25]) Let* $\epsilon := \Pr\left[\mathrm{argmax}_{\boldsymbol{z}}\mathrm{Re}\left(\hat{f}(\boldsymbol{z})\right) \neq \boldsymbol{s}_{(1,b)}\right]$ *be the probability that the algorithm does not recover the correct answer. Then, the number of samples required to solve the LWE problem with oracle* $\mathsf{LWE}_{\boldsymbol{s},\chi}$ *is*

$$m^{\mathsf{LWE}} = \frac{8n}{a}\ln\left(\frac{q}{\epsilon}\right)(R_{q,\sigma,\chi})^{-2^a} + (a-1)\frac{q^{n/a}-1}{2}, \tag{12}$$

where

$$R_{q,\sigma,\chi} = \begin{cases} \frac{q}{\pi}\sin\left(\frac{\pi}{q}\right)e^{-2\pi^2\sigma^2/q^2} & \text{when } \chi = \bar{\Psi}_{q,\sigma}, \\ 1 - \frac{2\pi^2\sigma^2}{q^2} & \text{when } \chi = D_{q,\sigma}. \end{cases}$$

Based on this Lemma 4, we can show the time complexity of the BKW algorithm for the LWE problem:

Lemma 5. *([35]) Let* a *and* b *be natural numbers such that* $ab = n$*. There is an algorithm to solve the LWE problem whose oracle is* $\mathsf{LWE}_{\boldsymbol{s},\chi}$*, with the number of samples* $m = \mathrm{poly}(\exp(\pi\alpha_{\mathrm{lwe}}^2 2^a), q^{n/a})$*, and the time complexity* $t = \mathrm{poly}(\exp(\pi\alpha_{\mathrm{lwe}}^2 2^a), q^{n/a})$*, where* $\alpha_{\mathrm{lwe}} := \frac{\sqrt{2\pi}\sigma}{q}$*, both when* $\chi = D_{\sigma,q}$ *and* $\chi = \bar{\Psi}_{\sigma,q}$*.*

Proof. Here, we refer the proof given in [35]. Similar to the proof of Theorem 3, using Lemma 4, we can prove that there is an algorithm to solve the LWE problem whose oracle is $\mathsf{LWE}_{\boldsymbol{s},\chi}$, with number of samples $m = \mathrm{poly}((R_{q,\sigma,\chi})^{-2^a}, q^{n/a})$, and time complexity $t = \mathrm{poly}((R_{q,\sigma,\chi})^{-2^a}, q^{n/a})$. Thus, we need only prove that

$$R_{q,\sigma,\chi} = O(\exp(-\pi\alpha_{\mathrm{lwe}}^2)) \tag{13}$$

holds, both when $\chi = \bar{\Psi}_{\sigma,q}$ and when $\chi = D_{\sigma,q}$. When $\chi = \bar{\Psi}_{\sigma,q}$, since $\sin\left(\frac{\pi}{q}\right) < \frac{\pi}{q}$, we obtain $R_{q,\sigma,\chi} < e^{-2\pi^2\sigma^2/q^2} = \exp(-\pi\alpha_{\mathrm{lwe}}^2)$, which means (13) holds. Next, we prove that (13) holds when $\chi = D_{\sigma,q}$. Using Taylor expansion, we obtain $R_{q,\sigma,\chi} - \exp(-\pi\alpha_{\mathrm{lwe}}^2) = 1 - \pi\alpha_{\mathrm{lwe}}^2 - \exp(-\pi\alpha_{\mathrm{lwe}}^2) = -\frac{\alpha_{\mathrm{lwe}}^4}{2} + O(\alpha_{\mathrm{lwe}}^6)$, thus we obtain $R_{q,\sigma,\chi} = \exp(-\pi\alpha_{\mathrm{lwe}}^2) + O(\alpha_{\mathrm{lwe}}^4)$. □

We now can derive the relation between the parameters of the LWE problem and the LWR problem.

Corollary 2. *The time complexity of the BKW algorithm to solve the LWE problem over* $\mathbb{Z}_q^n$ *with Gaussian parameter* σ *and that to solve the LWR problem over* $\mathbb{Z}_q^n$ *with rounding modulus* p *are asymptotically the same, when* q*,* p *and* σ *satisfy*

$$\sigma = \frac{q}{2\sqrt{3}p}. \tag{14}$$

Proof. The time complexity of the BKW algorithm to solve the LWE and LWR problems are given in Theorem 3 and Lemma 5, respectively. Solving the equation $\pi\alpha_{\mathrm{lwe}} = \pi\alpha_{\mathrm{lwr}}$ for σ, we obtain (14).

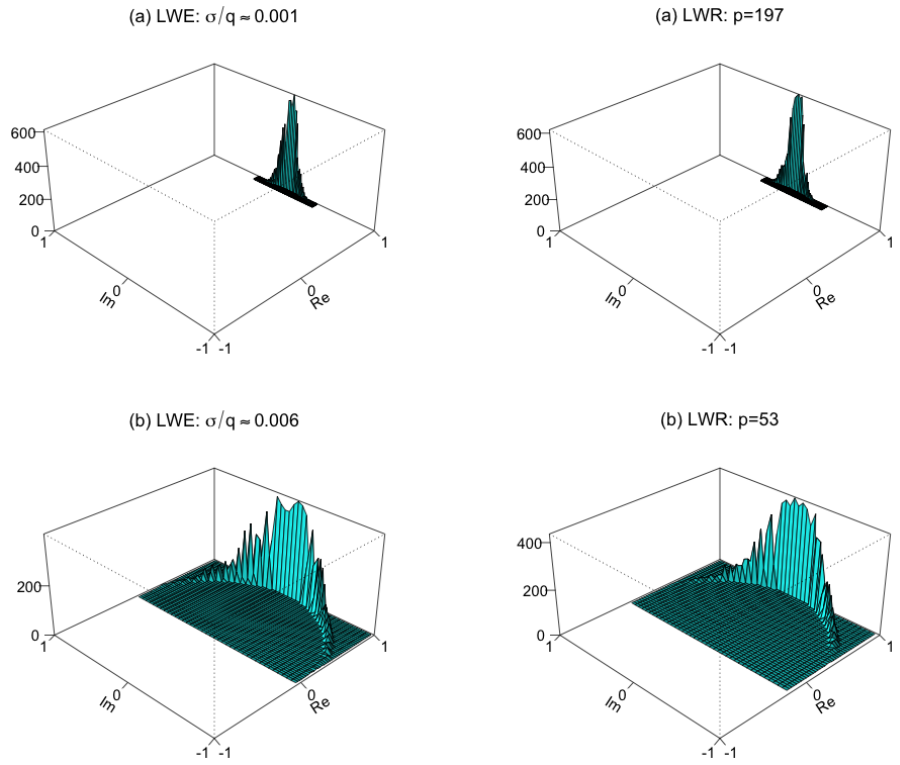

Fig. 1. Distribution of the noises of the LWE and LWR samples obtained after the sample reduction stage. Parameter p for the LWR sample is calculated from σ according to (14), which relates the complexity of the BKW algorithm for the LWE problem and the LWR problem.

4.2 Noise Distribution of Concrete Instances

We confirm that, when σ of the LWE problem and p of the LWR problem satisfy (14), the distribution of the Gaussian noise of LWE samples and the rounding error of LWR samples after sample reduction are similar by showing concrete examples. From the similarity of the LWR and LWE problems, the LWE-solving BKW algorithm in [25] is almost the same as the LWR-solving algorithm: Only the hypothesis testing stage is different. In the LWE-solving algorithm, (4) is replaced by $\hat{f}(\boldsymbol{z}) = \sum_{j=1}^{m'} \theta_q^{-(\langle \overline{\boldsymbol{a}}_j, \boldsymbol{z} \rangle - \overline{c}_j)}$, where $\theta_q := e^{\frac{2\pi i}{q}}$. We define $\overline{e}_j := \langle \overline{\boldsymbol{a}}_j, \boldsymbol{s} \rangle - \overline{c}_j$, and denote $\overline{\boldsymbol{a}}_j = \boldsymbol{a}_{j,1} \pm \boldsymbol{a}_{j,2} \pm \cdots \pm \boldsymbol{a}_{j,2^{a-1}}$, $\overline{c}_j = c_{j,1} \pm c_{j,2} \pm \cdots \pm c_{j,2^{a-1}}$, then we obtain

$$\begin{aligned}\overline{e}_j &= \langle \boldsymbol{a}_{j,1} \pm \boldsymbol{a}_{j,2} \pm \cdots \pm \boldsymbol{a}_{j,2^{a-1}}, \boldsymbol{s} \rangle - (c_{j,1} \pm c_{j,2} \pm \cdots \pm c_{j,2^{a-1}}) \\ &= \sum_{k=1}^{2^{a-1}} \langle \boldsymbol{a}_{j,k}, \boldsymbol{s} \rangle - c_{j,k} = \sum_{k=1}^{2^{a-1}} e_{j,k},\end{aligned}$$

Table 2. The time complexity and the required number of samples of the LWE-solving BKW algorithm and the LWR-solving BKW algorithm, when $p = \frac{q}{2\sqrt{3}\sigma}$.

		LWE (Regev [42])				LWR ($p = \frac{q}{2\sqrt{3}\sigma}$)			
n	q	σ	a	$\log(\mathcal{C}^{\mathsf{LWE}})$	$\log(m^{\mathsf{LWE}})$	p	a	$\log(\mathcal{C}^{\mathsf{LWR}})$	$\log(m^{\mathsf{LWR}})$
64	4099	5.67	19	49.74	43.61	211	19	49.70	43.60
80	6421	7.14	20	60.22	53.85	263	20	60.20	53.84
96	9221	8.65	21	71.72	63.79	311	21	71.03	63.65
112	12547	10.20	21	82.73	75.94	359	21	82.73	75.94
128	16411	11.79	22	91.84	84.86	409	22	91.84	84.86

where $e_{j,k}$ is the independent Gaussian noise from the original LWE samples. When $\boldsymbol{z} = \boldsymbol{s}$, we obtain $\hat{f}(\boldsymbol{s}) = \sum_{j=1}^{m'} \theta_q^{-(\langle \overline{\boldsymbol{a}}_j, \boldsymbol{s} \rangle - \overline{c}_j)} = \sum_{j=1}^{m'} \theta_q^{-(\sum_{k=1}^{2^{a-1}} e_{j,k})}$.

Figure 1 shows examples of the distribution of the Gaussian noise of LWE samples and the rounding error of LWR samples after sample reduction. The two figures on the left show histograms of $\theta_q^{-(\langle \overline{\boldsymbol{a}}_j, \boldsymbol{s} \rangle_q - \overline{c}_j)}$ on a complex plane, where $(\overline{\boldsymbol{a}}_j, \overline{c}_j), j \in \{1, 2, \ldots, m'\}$ are LWE samples obtained after the sample reduction stage. The two figures on the right show histograms of $\theta_p^{-(\frac{p}{q}\langle \overline{\boldsymbol{a}}_j, \boldsymbol{s} \rangle - \overline{c}_j)}$, where $(\overline{\boldsymbol{a}}_j, \overline{c}_j), j \in \{1, 2, \ldots, m'\}$ are LWR samples obtained after the sample reduction stage. In these figures, we used $n = 128, q = 16411, a = 8, m = 2^{20}, m' = 2^{12}$ and $l = 0$. In type (a) figure, we used $\sigma = q/(\sqrt{2\pi n}(\log(n))^2)$, which is for Regev cryptosystem [42]. For type (b), we used $\sigma = 4q/(\sqrt{2\pi n}(\log(n))^2)$. Parameter p is calculated from σ according to (14). From Figure 1, we can observe that those distributions are similar when parameter σ and p satisfy (14).

4.3 Time Complexity of Concrete Instances

We denote the time complexity of the LWE-solving BKW algorithm by $\mathcal{C}^{\mathsf{LWE}}$. Then, similar to Theorem 17 in [25], we obtain

$$\mathcal{C}^{\mathsf{LWE}} = \frac{1}{4}(a-2)(a-1)(2n/a+1)(q^{n/a}-1) + nq^{n/a}\log(q) + \sum_{j=0}^{a-1} m'^{\mathsf{LWE}}_{j,\epsilon}\left(\frac{a-1-j}{2}(n+2)+2\right)$$

where $m'^{\mathsf{LWE}}_{j,\epsilon} := \frac{8n}{a}\ln\left(\frac{q}{\epsilon}\right)\cdot\left(R_{q,\sigma,\chi}\right)^{-2^{a-j}}$. The number of samples of the LWE-solving BKW algorithm m^{LWE} is given in (12). Table 2 shows the time complexity of the LWE problem for various parameters of the Regev cryptosystem [42], and the time complexity of the LWR problem whose parameter p is calculated by (14). Concretely, in Table 2, $q = \text{nextprime}(n^2)$, $\sigma = \frac{n^2}{\sqrt{2\pi n}(\log(n))^2}$, $p = \text{nextprime}\left(\frac{q}{2\sqrt{3}\sigma}\right)$, $\epsilon = 0.01$. From this table, we can confirm that the complexity of the LWE problem and the LWR problem whose parameters satisfy (14) are almost the same.

5 Conclusion

We analyzed the time complexity of the BKW algorithm for the LWR problem and theoretically derived the optimal number of blocks a that asymptotically (but almost completely) minimizes the time complexity of the algorithm, while Duc *et al.* numerically searched for the optimal value of a [25].

Furthermore, we derived the relation between the parameters of the LWE and LWR problems with the same time complexity of the BKW algorithm, which is $\sigma = \frac{q}{2\sqrt{3p}}$. This equation coincides with the equation derived by the complexity analysis of the lattice attacks: We showed that the conversion equation can also be applied for complexity analysis based on the BKW algorithm.

References

1. Albrecht, M.R.: On Dual Lattice Attacks against Small-Secret LWE and Parameter Choices in HElib and SEAL. In Coron, J.S., Nielsen, J.B., eds.: Advances in Cryptology – EUROCRYPT 2017, Springer (2017) 103–129
2. Albrecht, M.R., Cid, C., Faugère, J.C., Fitzpatrick, R., Perret, L.: On the complexity of the BKW algorithm on LWE. Designs, Codes and Cryptography **74**(2) (Feb 2015) 325–354
3. Albrecht, M.R., Curtis, B.R., Deo, A., Davidson, A., Player, R., Postlethwaite, E.W., Virdia, F., Wunderer, T.: Estimate All the {LWE, NTRU} Schemes! In Catalano, D., De Prisco, R., eds.: Security and Cryptography for Networks. SCN '18, Springer (2018) 351–367
4. Albrecht, M.R., Faugère, J.C., Fitzpatrick, R., Perret, L.: Lazy Modulus Switching for the BKW Algorithm on LWE. In Krawczyk, H., ed.: Public-Key Cryptography – PKC 2014, Springer (2014) 429–445
5. Albrecht, M.R., Orsini, E., Paterson, K.G., Peer, G., Smart, N.P.: Tightly Secure Ring-LWE Based Key Encapsulation with Short Ciphertexts. In Foley, S.N., Gollmann, D., Snekkenes, E., eds.: Computer Security – ESORICS 2017, Springer (2017) 29–46
6. Alkim, E., Ducas, L., Pöppelmann, T., Schwabe, P.: Post-quantum Key Exchange - A New Hope. In: USENIX Security Symposium. (2016) 327–343
7. Alwen, J., Krenn, S., Pietrzak, K., Wichs, D.: Learning with Rounding, Revisited. In Canetti, R., Garay, J.A., eds.: Advances in Cryptology – CRYPTO 2013, Springer (2013) 57–74
8. Applebaum, B., Cash, D., Peikert, C., Sahai, A.: Fast Cryptographic Primitives and Circular-Secure Encryption Based on Hard Learning Problems. In Halevi, S., ed.: Advances in Cryptology - CRYPTO 2009, Springer (2009) 595–618
9. Baan, H., Bhattacharya, S., Garcia-Morchon, O., Rietman, R., Tolhuizen, L., Torre-Arce, J.L., Zhang, Z.: Round2: KEM and PKE based on GLWR. Cryptology ePrint Archive, Report 2017/1183 (2017) https://eprint.iacr.org/2017/1183.
10. Banerjee, A., Fuchsbauer, G., Peikert, C., Pietrzak, K., Stevens, S.: Key-Homomorphic Constrained Pseudorandom Functions. In Dodis, Y., Nielsen, J.B., eds.: Theory of Cryptography Conference. TCC '15, Springer (2015) 31–60
11. Banerjee, A., Peikert, C.: New and Improved Key-Homomorphic Pseudorandom Functions. In Garay, J.A., Gennaro, R., eds.: Advances in Cryptology – CRYPTO 2014, Springer (2014) 353–370

12. Banerjee, A., Peikert, C., Rosen, A.: Pseudorandom Functions and Lattices. In Pointcheval, D., Johansson, T., eds.: Advances in Cryptology – EUROCRYPT 2012, Springer (2012) 719–737
13. Bansarkhani, R.E.: LARA - A Design Concept for Lattice-based Encryption. Cryptology ePrint Archive, Report 2017/049 (2017) https://eprint.iacr.org/2017/049.
14. Becker, A., Gama, N., Joux, A.: A sieve algorithm based on overlattices. LMS Journal of Computation and Mathematics **17**(A) (2014) 49–70
15. Blum, A., Kalai, A., Wasserman, H.: Noise-Tolerant Learning, the Parity Problem, and the Statistical Query Model. J. ACM **50**(4) (July 2003) 506–519
16. Boneh, D., Lewi, K., Montgomery, H., Raghunathan, A.: Key Homomorphic PRFs and Their Applications. In Canetti, R., Garay, J.A., eds.: Advances in Cryptology – CRYPTO 2013, Springer (2013) 410–428
17. Bos, J., Ducas, L., Kiltz, E., Lepoint, T., Lyubashevsky, V., Schanck, J.M., Schwabe, P., Seiler, G., Stehle, D.: CRYSTALS - Kyber: A CCA-Secure Module-Lattice-Based KEM. In: 2018 IEEE European Symposium on Security and Privacy (EuroS&P). (April 2018) 353–367
18. Bos, J., Costello, C., Ducas, L., Mironov, I., Naehrig, M., Nikolaenko, V., Raghunathan, A., Stebila, D.: Frodo: Take off the Ring! Practical, Quantum-Secure Key Exchange from LWE. In: Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security. CCS '16, ACM (2016) 1006–1018
19. Brakerski, Z., Langlois, A., Peikert, C., Regev, O., Stehlé, D.: Classical Hardness of Learning with Errors. In: Proceedings of the Forty-fifth Annual ACM Symposium on Theory of Computing. STOC '13, ACM (2013) 575–584
20. Chen, Y., Nguyen, P.Q.: BKZ 2.0: Better Lattice Security Estimates. In Lee, D.H., Wang, X., eds.: Advances in Cryptology – ASIACRYPT 2011, Springer (2011) 1–20
21. 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., eds.: Security and Cryptography for Networks. SCN '18, Springer (2018) 160–177
22. Cheon, J.H., Park, S., Lee, J., Kim, D., Song, Y., Hong, S., Kim, D., Kim, J., Hong, S.M., Yun, A., Kim, J., Park, H., Choi, E., Kim, K., Kim, J.S., Lee, J.: Lizard. Technical report, National Institute of Standards and Technology, 2017. available at https://csrc.nist.gov/.
23. Corless, R.M., Gonnet, G.H., Hare, D.E., Jeffrey, D.J., Knuth, D.E.: On the Lambert W Function. Advances in Computational Mathematics **5** (1996) 329–359
24. D'Anvers, J.P., Karmakar, A., Sinha Roy, S., Vercauteren, F.: Saber: Module-LWR Based Key Exchange, CPA-Secure Encryption and CCA-Secure KEM. In Joux, A., Nitaj, A., Rachidi, T., eds.: Progress in Cryptology – AFRICACRYPT 2018, Springer (2018) 282–305
25. Duc, A., Tramèr, F., Vaudenay, S.: Better Algorithms for LWE and LWR. In Oswald, E., Fischlin, M., eds.: Advances in Cryptology – EUROCRYPT 2015, Springer (2015) 173–202
26. Gama, N., Nguyen, P.Q., Regev, O.: Lattice Enumeration Using Extreme Pruning. In Gilbert, H., ed.: Advances in Cryptology – EUROCRYPT 2010, Springer (2010) 257–278
27. Goldwasser, S., Kalai, Y.T., Peikert, C., Vaikuntanathan, V.: Robustness of the Learning with Errors Assumption. In: Innovations in Computer Science (ICS 2010), Tsinghua University Press (2010)

28. Guo, Q., Johansson, T., Mårtensson, E., Stankovski, P.: Coded-BKW with sieving. In Takagi, T., Peyrin, T., eds.: Advances in Cryptology – ASIACRYPT 2017, Springer (2017) 323–346
29. Guo, Q., Johansson, T., Stankovski, P.: Coded-BKW: Solving LWE Using Lattice Codes. In Gennaro, R., Robshaw, M., eds.: Advances in Cryptology – CRYPTO 2015, Springer (2015) 23–42
30. Hanrot, G., Pujol, X., Stehlé, D.: Algorithms for the Shortest and Closest Lattice Vector Problems. In Chee, Y.M., Guo, Z., Ling, S., Shao, F., Tang, Y., Wang, H., Xing, C., eds.: Coding and Cryptology, Springer (2011) 159–190
31. Hanrot, G., Pujol, X., Stehlé, D.: Analyzing Blockwise Lattice Algorithms Using Dynamical Systems. In Rogaway, P., ed.: Advances in Cryptology – CRYPTO 2011, Springer (2011) 447–464
32. Herold, G., Kirshanova, E., May, A.: On the asymptotic complexity of solving LWE. Designs, Codes and Cryptography **86**(1) (Jan 2018) 55–83
33. Information Technology Laboratory, National Institute of Standards and Technology: Post-Quantum Cryptography. https://csrc.nist.gov/Projects/Post-Quantum-Cryptography Accessed: January 31, 2018.
34. Jin, Z., Zhao, Y.: Optimal Key Consensus in Presence of Noise. CoRR **abs/1611.06150** (2016)
35. Kaminakaya, K., Kunihiro, N., Takayasu, A.: BKW Algorithm for Solving LWE Problem. In: Symposium on Cryptography and Information Security. SCIS '16, IEICE (2016) in Japanese.
36. Kirchner, P., Fouque, P.A.: An Improved BKW Algorithm for LWE with Applications to Cryptography and Lattices. In Gennaro, R., Robshaw, M., eds.: Advances in Cryptology – CRYPTO 2015, Springer (2015) 43–62
37. Laarhoven, T.: Sieving for Shortest Vectors in Lattices Using Angular Locality-Sensitive Hashing. In Gennaro, R., Robshaw, M., eds.: Advances in Cryptology – CRYPTO 2015, Springer (2015) 3–22
38. Lindner, R., Peikert, C.: Better Key Sizes (and Attacks) for LWE-Based Encryption. In Kiayias, A., ed.: Topics in Cryptology – CT-RSA 2011, Springer (2011) 319–339
39. Liu, M., Nguyen, P.Q.: Solving BDD by Enumeration: An Update. In Dawson, E., ed.: Topics in Cryptology – CT-RSA 2013, Springer (2013) 293–309
40. Nguyen, P.Q.: Lattice Reduction Algorithms: Theory and Practice. In Paterson, K.G., ed.: Advances in Cryptology – EUROCRYPT 2011, Springer (2011) 2–6
41. Nguyen, P.Q., Stehlé, D.: Low-Dimensional Lattice Basis Reduction Revisited. In Buell, D., ed.: Algorithmic Number Theory, Springer (2004) 338–357
42. Regev, O.: On Lattices, Learning with Errors, Random Linear Codes, and Cryptography. J. ACM **56**(6) (September 2009) 34:1–34:40
43. Xie, X., Xue, R., Zhang, R.: Deterministic Public Key Encryption and Identity-Based Encryption from Lattices in the Auxiliary-Input Setting. In Visconti, I., De Prisco, R., eds.: Security and Cryptography for Networks. SCN '12, Springer (2012) 1–18