Provably Secure LWE Encryption with Smallish Uniform Noise and Secret

Daniel Cabarcas, Florian Göpfert, and Patrick Weiden

Technische Universität Darmstadt, Germany
dcabarc@unal.edu.co,{fgoepfert,pweiden}@cdc.informatik.tu-darmstadt.de

Abstract. In this paper we propose the first provably secure public key encryption scheme based on the Learning with Errors (LWE) problem, in which secrets and errors are sampled uniformly at random from a relatively small set rather than from the commonly used discrete Gaussian distribution. Using a uniform distribution, instead of a Gaussian, has the potential of improving computational efficiency a great deal due to its simplicity, thus making the scheme attractive for use in practice. At the same time our scheme features the strong security guarantee of being based on the hardness of worst-case lattice problems. After presenting the construction of our scheme we prove its security and propose asymptotic parameters. Finally, we compare our scheme on several measures to one of the most efficient LWE-based encryption schemes with Gaussian noise. We show that the expected efficiency improvement is debunked, due to the large blow-up of the parameter sets involved.

Keywords Lattice-Based Cryptography; Learning with Errors; Uniform Noise; Provable Security.

1 Introduction

The Learning with Errors (LWE) problem [Reg09] has been the source of great progress in cryptography by providing a link between a variety of cryptographic constructions and classical mathematical problems. It has served as the base for several basic cryptographic primitives including public key encryption (see e.g. [SSTX09,GHV10,LPR10,LP11,SS11,Gal13]) and digital signatures (see e.g. [GPV08,CHKP10,Lyu12,SS13]), as well as other primitives such as lossy-trapdoor functions (see e.g. [PW08,BKPW12,Wee12]), identity- and attribute-based encryption (see e.g. [CHKP10,ABB10,GVW13,GGH$^+$13b]), somewhat and fully homomorphic encryption (see e.g. [BV11a,BV11b,BGV12,GHS12,ASP13,BGH13]), and multilinear maps (see e.g. [GGH13a]). The (decision) LWE problem is to distinguish (A, b) sampled according to the LWE distribution from (A, b) sampled uniformly at random from $\mathbb{Z}_q^{m\times n} \times \mathbb{Z}_q^m$. On input integers n, m, q and distributions $\mathcal{X}, \mathcal{Y}$ on $\mathbb{Z}_q$, the LWE distribution samples a matrix $A \in \mathbb{Z}_q^{m\times n}$ uniformly at random, the entries of a secret vector s according to $\mathcal{X}$, the entries of an error vector e according to $\mathcal{Y}$, computes the noisy linear system $b \equiv As + e \bmod q$ and outputs the tuple (A, b).

In addition to new cryptographic constructions made possible by LWE, there exists a strong security guarantee for schemes based on LWE, the so-called worst-to-average-case reduction. Regev and Peikert [Reg09,Pei09] independently showed that, for certain parameters, LWE is as hard as classical lattice problems, such as the Shortest Independent Vector Problem (SIVP), in the worst case. Thus it follows that LWE-based schemes are provably secure assuming the worst-case hardness of classical lattice problems. Since these

lattice problems have been widely studied and are expected to be hard for most instances, the assumption is considered quite plausible.

When inspecting practical aspects of LWE-based constructions, there has been a continuous efficiency improvement over the past years (e.g. [LPR10,LP11,GFS+12,PG12]). However, the LWE-based cryptosystems have not yet reached the practicality threshold of mainstream cryptosystems such as RSA as used today. Despite the simplicity of the involved operations and progress in tightening reductions to decrease parameters, the error distribution stands as an obstacle to develop a simple and practical cryptosystem. To this date, all provably worst-case secure LWE-based schemes require the error distribution to be a discrete Gaussian, which is problematic in practice. According to Weiden et al. [WHCB13], Gaussian sampling can be very time consuming. Moreover, existing discrete Gaussian samplers require either exact computation of transcendental functions or to store large amounts of data [GPV08,Pei10,GD12], an unsurmountable burden for some constraint devices. It is also not well understood at the moment, how accurate such samplers are or what the impact of a faulty sampler on security is. Furthermore, recent experiments, in which weak RSA keys were exposed [HDWH12,LHA+12], show that schemes can be broken because of biased sampling algorithms. Thus, the complexity of Gaussian samplers adds another pitfall for developers when implementing LWE-based cryptosystems.

One necessary tool to build a worst-case secure scheme without Gaussian sampling was recently provided by Micciancio and Peikert [MP13]. The authors give a reduction from SIVP to LWE with uniform error if the number of LWE samples is limited. They also suggest it might be possible to construct a worst-case secure LWE-based encryption scheme with polynomially bounded uniform error. However, they do not elaborate any further.

Our Contribution In this paper we answer this question affirmatively by proposing the first provably worst-case secure LWE-based encryption scheme with noise and secret drawn from uniform distributions on smallish sets. Our proposed scheme, which we refer to as U-LP, is similar to the LP encryption scheme by Lindner and Peikert [LP11], but it uses uniform distributions over smallish sets for generating keys and for encrypting messages. The security proof of LP falls apart for U-LP because of the use of the uniform distributions. Thus, we prove the security of our construction by relating the security to the hardness of classical worst-case lattice problems. Moreover, we propose asymptotic parameters for U-LP.

The main property of our scheme, instantiating LWE with a uniform distribution instead of a discrete Gaussian, makes it potentially preferable to existing schemes because of the simplicity of sampling errors and secrets. Sampling from a uniform distribution over a set $\{0, \ldots, t-1\}$ for small t is extremely easy and fast. Even in the worst case (if t is slightly bigger than a power of two), it simply requires to sample $2\lceil \log(t) \rceil$ uniformly random bits on average. Moreover, uniform sampling requires no additional storage and no computation of transcendental functions. By these characteristics the uniform distribution on a small set is an ideal candidate for sampling errors and secrets.

From a technical perspective, the challenge of this paper is in designing U-LP in order to obtain an encryption scheme that is both efficient and provably secure. Our main result (summarized in Theorem 5) states that U-LP with correctly chosen parameters is indistinguishable under chosen plaintext attacks (IND-CPA secure) as long as SIVP is hard in the worst case, using a recent result of Micciancio and Peikert [MP13]. The proof can be

summarized as follows:

$$
\begin{aligned}
&\text{U-LP with parameters } n, q, s_k, s_e, \ell \text{ is insecure} \\
&\xRightarrow{\text{Lemma 1}} \text{LWE}(n, n, q, \mathcal{U}_{s_k}, \mathcal{U}_{s_k}) \text{ is easy} \vee \\
&\qquad\qquad \text{LWE}(n, n+\ell, q, \mathcal{U}_{s_e}, \mathcal{U}_{s_e}) \text{ is easy} \\
&\xRightarrow{\text{Lemma 2}} \text{LWE}(n, 2n, q, \mathcal{U}_q, \mathcal{U}_{s_k}) \text{ is easy} \vee \\
&\qquad\qquad \text{LWE}(n, 2n+\ell, q, \mathcal{U}_q, \mathcal{U}_{s_e}) \text{ is easy} \\
&\xRightarrow{\text{Lemma 4}} \text{SIVP}(k, \widetilde{O}(\sqrt{k} \cdot q)) \text{ is easy}
\end{aligned}
$$

Here $\text{LWE}(n, m, q, \mathcal{X}, \mathcal{Y})$ denotes the (decision) LWE problem with $A \in \mathbb{Z}_q^{m \times n}$ sampled uniformly at random, secret $s \in \mathbb{Z}_q^n$ sampled according to the distribution $\mathcal{X}$ and error $e \in \mathbb{Z}_q^m$ according to distribution $\mathcal{Y}$ (written as $e \leftarrow \mathcal{Y}$). By $\text{SIVP}(k, \gamma)$ we denote the Shortest Independent Vector Problem in dimension k with approximation factor γ, and by $\mathcal{U}_t$ we denote the uniform distribution on $\{0, \ldots, t-1\}$. Lemma 1 (Section 2.1) relates the security of U-LP to the hardness of two LWE instances, one that hides the secret key and another one that hides the message, with both smallish uniform noise and secret chosen from the same distribution. Lemma 2 (Section 2.1) shows that these LWE instances, where the secret is chosen according to the error distribution, are not easier than LWE instances with secret chosen uniformly from $\mathbb{Z}_q^n$. Lemma 4 (Section 2.2) proves that a family of LWE instances suitable for U-LP is at least as hard as worst-case SIVP. Finally, Theorem 5 (Section 2.3) puts all these pieces together.

We compare our instantiations of U-LP and LP on several efficiency measures. Unfortunately, the expected advantage does not materialize and instead we obtain opposed results. This is due to the fact that the modulus q and the bound t of the smallish set cannot be chosen too small in order to use the uniform distribution in a worst-case secure way within our proposed scheme. In more detail, the worst-to-average-case reduction from LWE without Gaussian distributions to classical lattice problems requires to restrict the number of samples [MP13]. The maximum number of samples is determined by the size of the secrets and errors—more samples require bigger secrets and errors. Unfortunately, the number of samples provided by the scheme is lower-bounded, which leads to a lower bound for the size of the errors. Bigger average errors, however, require a bigger modulus in order to avoid decryption failures.

Although we missed the goal of having a truly small set, we can conclude that it is relatively small with respect to the modulus (thus smallish). For security parameter n, the instantiation of U-LP requires a modulus $q = O(n^{3.7})$ and samples drawn from a uniform distribution on $\{0, \ldots, t-1\}$ with $t = O(n^{1.4})$. In comparison, the instantiation of LP with Gaussian noise and comparable parameters requires a modulus of $q = O(n^2)$, and samples drawn from a discrete Gaussian with parameter $\sigma = O(n^{0.5})$. Thus, U-LP requires bigger error and modulus as a trade-off for being able to use uniform noise. In terms of security, we prove that U-LP is secure as long as SIVP in dimension $k = n/8$ with approximation factor $\gamma = \widetilde{O}(n^{4.7})$ is hard, whereas LP with modified parameters is secure as long as SIVP in dimension $k = n$ with approximation factor $\gamma = O(n^{2.5})$ is hard.

Organization The remainder of the paper consists of two sections. In Section 2 we present U-LP and prove its security following the logic of the above sketched proof. Then in Section 3

we analyze several measurements of the scheme's efficiency and we compare them to those of LP with comparable parameters. A short conclusion in Section 4 completes the paper.

2 The Scheme

In this section we introduce U-LP, an LWE-based scheme following the framework of Linder and Peikert [LP11], but with noise and secret drawn form a uniform distribution on a relatively small set. We propose parameters for the scheme and prove that it is correct and worst-case secure.

2.1 Description

The LP scheme introduced by Lindner and Peikert [LP11] is an instance of an abstract cryptosystem described by Micciancio [Mic10]. The scheme uses simple error-tolerant encoding and decoding functions $\mathsf{encode} : \mathbb{Z}_2^\ell \to \mathbb{Z}_q^\ell$ and $\mathsf{decode} : \mathbb{Z}_q^\ell \to \mathbb{Z}_2^\ell$, such that $\mathsf{decode}(\mathsf{encode}(m) + e) = m$ for any error vector e with $\|e\|_\infty < \lfloor q/4 \rfloor$.

KeyGen(n, q, s_k, ℓ)**:** Sample $A \leftarrow \mathcal{U}_q^{n \times n}$, $E \leftarrow \mathcal{U}_{s_k}^{\ell \times n}$ and $S \leftarrow \mathcal{U}_{s_k}^{\ell \times n}$, and let $P = E - SA \in \mathbb{Z}_q^{\ell \times n}$. Return public key (A, P) and secret key S.

Enc$(\mu, (A, P), s_e)$**:** Sample $e_1 \leftarrow \mathcal{U}_{s_e}^n$, $e_2 \leftarrow \mathcal{U}_{s_e}^n$, $e_3 \leftarrow \mathcal{U}_{s_e}^\ell$, compute $\mu' = \mathsf{encode}(\mu) \in \mathbb{Z}_q^\ell$, $c_1 = Ae_1 + e_2$ and $c_2 = Pe_1 + e_3 + \mu'$, and return ciphertext (c_1, c_2).

Dec$((c_1, c_2), S)$**:** Return message $\mathsf{decode}(Sc_1 + c_2) \in \mathbb{Z}_2^\ell$.

Fig. 1. U-LP – our LWE-Based Encryption Scheme with Smallish Uniform Error and Secret

Since we want to avoid Gaussian sampling, our scheme U-LP is derived from LP by replacing the Gaussian distribution by uniform distributions on smallish sets. In Fig. 1 we illustrate the key generation, encryption and decryption algorithms of U-LP. It is parameterized by a lattice dimension $n \in \mathbb{N}$, a prime modulus $q \geq 2$, error bounds $s_k, s_e \in \mathbb{N}$ for key generation and encryption, respectively, and an integer message length $\ell \geq 1$. For decryption to work properly, s_k and s_e must be small with respect to q. Note that in decryption

$$\begin{aligned} Sc_1 + c_2 &= S(Ae_1 + e_2) + Pe_1 + e_3 + \mu' \\ &= SAe_1 + Se_2 + Pe_1 + e_3 + \mu' \\ &= (E - P)e_1 + Se_2 + Pe_1 + e_3 + \mu' \\ &= Ee_1 + Se_2 + e_3 + \mu' \\ &= e + \mu', \end{aligned}$$

where $e = Ee_1 + Se_2 + e_3$. The sizes of E and S depend on s_k, while the sizes of e_1, e_2 and e_3 depend on s_e. As long as s_k and s_e are small enough to guarantee that $\|e\|_\infty < \lfloor q/4 \rfloor$, we have $\mathsf{decode}(Sc_1 + c_2) = \mu$ and hence no decryption failures occur (see Theorem 5).

The security of U-LP relies on the hardness of two instances of the decision LWE problem with uniform noise. The idea is to hide the secret key using a first LWE instance and to hide the message using a second instance.

Lemma 1 *The encryption scheme U-LP as presented in Fig. 1 is IND-CPA secure as long as* $\mathrm{LWE}(n, n, q, \mathcal{U}_{s_k}, \mathcal{U}_{s_k})$ *and* $\mathrm{LWE}(n, n+\ell, q, \mathcal{U}_{s_e}, \mathcal{U}_{s_e})$ *are hard.*

The proof of Lemma 1 is very similar to that of Theorem 3.2 in [LP11]. Intuitively, the idea is to show that an attacker cannot distinguish $c_2 = Pe_1 + e_3 + \mu'$ from a vector chosen uniformly at random, by showing that he cannot distinguish $Pe_1 + e_3$ from uniformly random. This is done in two steps. First, distinguishing P from a uniformly random matrix implies distinguishing (one of) ℓ independent $\mathrm{LWE}(n, n, q, \mathcal{U}_{s_k}, \mathcal{U}_{s_k})$ instances of $P^T = A^T(-S^T) + E^T$ from uniformly random, which is hard by assumption. Second, since P appears random from the attacker's point of view, $\begin{pmatrix} c_1 \\ c_2 \end{pmatrix} = \begin{pmatrix} A \\ P \end{pmatrix} e_1 + \begin{pmatrix} e_2 \\ e_3 \end{pmatrix}$ is an $\mathrm{LWE}(n, n+\ell, q, \mathcal{U}_{s_e}, \mathcal{U}_{s_e})$ instance, which again is hard by assumption. Thus, $Pe_1 + e_3$ is indistinguishable from a uniformly random vector, and hence also is c_2.

Secret Sampled from Error Distribution It is important to notice that the security of U-LP is based on LWE instances not only with small error, but also with small uniform secret. However, the worst-case results for LWE require that the entries of the secret are sampled uniformly over $\mathbb{Z}_q$. To close this gap, we adapt a result from Applebaum et al. [ACPS09], who showed that LWE becomes no easier if the secret is chosen according to the error distribution (instead of chosen uniformly from $\mathbb{Z}_q^n$). Unfortunately, this reduction comes at a loss of n samples. This is not an issue under the "classical" definition of LWE, where arbitrarily many samples are available. However, it does matter if the amount of samples is limited, as it is the case for the small uniform noise LWE instances in U-LP. The following result states the reduction from LWE with uniform secret to LWE with small secret, taking into account the precise loss of samples.

Lemma 2 *Let* $\mathcal{X}$ *be any distribution over* $\mathbb{Z}_q$*. If there is a probabilistic polynomial time (PPT) algorithm that solves* $\mathrm{LWE}(n, m, q, \mathcal{X}, \mathcal{X})$ *with probability* p*, then there exists a PPT algorithm for solving* $\mathrm{LWE}(n, m+n, q, \mathcal{U}_q, \mathcal{X})$ *with probability* $p \cdot \prod_{i=1}^{n}(1 - q^{-i})$*.*

Proof. Suppose $\mathcal{A}$ is a PPT algorithm that solves $\mathrm{LWE}(n, m, q, \mathcal{X}, \mathcal{X})$ with probability p. We now define an algorithm $\mathcal{B}$ with oracle access to $\mathcal{A}$ for solving $\mathrm{LWE}(n, m+n, q, \mathcal{U}_q, \mathcal{X})$. Algorithm $\mathcal{B}$ receives as input $m+n$ samples $(a_i, b_i) \in \mathbb{Z}_q^n \times \mathbb{Z}_q$, where $a_i \leftarrow \mathbb{Z}_q^n$, and the b_i's are either $b_i := \langle a_i, s \rangle + e_i$ with $e_i \leftarrow \mathcal{X}$ (in case of LWE) or $b_i \leftarrow \mathbb{Z}_q$ (in case of uniform). Then $\mathcal{B}$ performs the following steps.

1.) Use the first n samples to define

$$\overline{b} := \begin{pmatrix} b_1 \\ \vdots \\ b_n \end{pmatrix}, \quad \overline{A} := \begin{pmatrix} a_1, a_2, \ldots, a_n \end{pmatrix}.$$

If $\overline{A}$ is not invertible over $\mathbb{Z}_q$, return $\perp$.

2.) Transform the remaining m samples to (a'_i, b'_i) by

$$a'_i := -\overline{A}^{-1} a_{n+i} \qquad \text{and} \qquad b'_i := b_{n+i} + \langle \overline{b}, a'_i \rangle$$

for $i \in \{1, \ldots, m\}$.

3.) Query $\mathcal{A}$ with the samples $\{(a'_i, b'_i) \mid i \in \{1, \ldots, m\}\}$ and forward the output (either uniform or LWE) to the challenger.

It is easy to see that if the samples b_{n+i} are uniform, so are the transformed samples b'_i and thus $\mathcal{A}$ in step 3 is queried with uniform input. Now, we look at the case $b_i = \langle a_i, s \rangle + e_i$. Denoting $\overline{e} := (e_1, \ldots, e_n)^T$ we have that $\overline{b} = \overline{A}^T s + \overline{e}$ and then

$$\begin{aligned} b'_i &= b_{n+i} + \langle \overline{b}, a'_i \rangle \\ &= \langle a_{n+i}, s \rangle + e_{n+i} + \langle \overline{A}^T s + \overline{e}, a'_i \rangle \\ &= \langle a_{n+i}, s \rangle + \langle \overline{A}^T s, a'_i \rangle + \langle \overline{e}, a'_i \rangle + e_{n+i} \\ &= \langle a_{n+i}, s \rangle + \langle \overline{A}^T s, -\overline{A}^{-1} a_{n+i} \rangle + \langle a'_i, \overline{e} \rangle + e_{n+i} \\ &= \langle a'_i, \overline{e} \rangle + e_{n+i}, \end{aligned}$$

thus $\mathcal{A}$ is queried with LWE samples where the secret is sampled according to $\mathcal{X}$. Thus $\mathcal{B}$ succeeds whenever $\overline{A}$ is invertible and $\mathcal{A}$ succeeds. So the success probability of $\mathcal{B}$ is $p \cdot c$ with

$$\begin{aligned} c &= \Pr[\overline{A} \text{ is invertible} \mid \overline{A} \leftarrow \mathcal{U}_q^{n \times n}] \\ &= \left(1 - \frac{1}{q^n}\right)\left(1 - \frac{q}{q^n}\right) \cdots \left(1 - \frac{q^{n-1}}{q^n}\right) \\ &= \prod_{i=0}^{n-1} \left(1 - q^{i-n}\right). \end{aligned}$$

□

2.2 LWE with Small Uniform Error

By combining Lemmata 1 and 2 we establish a link between the security of U-LP and the hardness of the two LWE instances $\mathrm{LWE}(n, 2n, q, \mathcal{U}_q, \mathcal{U}_{s_k})$ and $\mathrm{LWE}(n, 2n+\ell, q, \mathcal{U}_q, \mathcal{U}_{s_k})$ with uniform error. In contrast, the original reductions by Regev [Reg09] and Peikert [Pei09] show the hardness of LWE instances with Gaussian error and cannot be applied. In order to prove the worst-case hardness of our LWE instances, we use a recent observation by Micciancio and Peikert. This result holds only under certain conditions, in particular, if the number of LWE samples is limited. In this section we propose a family of LWE instances that satisfies the preconditions of Micciancio and Peikert's result and at the same time is suitable for U-LP. We first give an adapted version of their statement.

Theorem 3 (Theorem 4.6 in [MP13]) *Let*

$$\begin{aligned} 0 < k \leq n \leq m - \omega(\log(k)) \leq k^{O(1)}, \\ s \geq (Cm)^{(m-(n-k))/(n-k)} \end{aligned} \tag{1}$$

for a large enough universal constant C, and q be a prime such that

$$\max\{3\sqrt{k}, (4s)^{m/(m-n)}\} \leq q \leq k^{O(1)}.$$

The $\mathrm{LWE}(n, m, q, \mathcal{U}_q, \mathcal{X})$ *problem is hard with respect to the uniform input distribution* $\mathcal{X} = \mathcal{U}_s$*, under the assumption that* $\mathrm{SIVP}(k, \widetilde{O}(\sqrt{k} \cdot q))$ *is (quantum) hard to approximate in the worst case.*

In order to construct the LWE instance that hides the secret key, we choose the number of samples $m = c(n-k)$ for some constant c so that $\frac{m-(n-k)}{n-k}$ in (1) is constant and hence s is polynomial in n. In order to use the reduction to small secrets from Section 2.1, we need at least n additional samples. This means $m - n \geq n$ and leads to $c(n-k) \geq 2n$ or, equivalently,

$$k \leq \frac{c-2}{c} n.$$

To obtain the biggest possible worst-case dimension k, we choose $k = \frac{c-2}{c} n$. The number of samples is thus given by

$$m = c\left(n - \frac{c-2}{c} n\right) = c\left(n - n + \frac{2}{c} n\right) = 2n.$$

Since k must be an integer, we choose n to be a multiple of $\frac{c}{c-2}$. Obviously, c must be greater than 2 to get a positive worst-case dimension k. Since we want to allow powers of two for the security parameter n, we choose c such that $\frac{c}{c-2} = 2^i$ for some positive integer i, which is equivalent to $c = \frac{2^{i+1}}{2^i - 1}$. Since bigger values of c lead to bigger parameters (s and q) and bigger keys, consequently, i should not be chosen too small. For concreteness, we choose $i = 3$, i.e., n has to be a multiple of 8, which leads to $c = 16/7$ and

$$k = \frac{c-2}{c} n = \frac{2/7}{16/7} n = \frac{1}{8} n.$$

The range of the secret can hence be calculated by

$$s = \left\lceil (Cm)^{(m-(n-k))/(n-k)} \right\rceil = \left\lceil (2Cn)^{9/7} \right\rceil.$$

The LWE instance used to hide the message has to be chosen slightly different. Since we want to have the same worst-case dimension for both LWE instances and since the average-case dimension n stays the same, one would have to choose ℓ linear in $n - k$ in order to choose the amount of samples $2n + \ell$ linear in $n - k$. Instead, we adjust both s_e and q such that Theorem 3 can be applied. We now state the reduction from SIVP to LWE for these particular families of LWE instances. The result accounts for both cases, with $a = 0$ for the LWE instance used to hide the secret key and with $a = \ell$ for the LWE instance used to hide the message.

Lemma 4 *Let* $n = 8k$ *for some* $k \in \mathbb{N}$, $0 \leq a \leq n^{O(1)}$, $m = 2n + a$, $s = \left\lceil (Cm)^{9/7 + (8a)/(7n)} \right\rceil$ *and* $16s^2 \leq q \leq n^{O(1)}$*. Then average case* $\mathrm{LWE}(n, m, q, \mathcal{U}_q, \mathcal{U}_s)$ *is at least as hard as worst case* $\mathrm{SIVP}(k, \widetilde{O}(\sqrt{k} \cdot q))$*.*

Proof. The result follows from Theorem 3. We show that its preconditions are fulfilled:

1.) $s \geq (Cm)^{\frac{m-(n-k)}{n-k}}$: This is clear since

$$\frac{m-(n-k)}{n-k} = \frac{2n+a-n+n/8}{n-n/8} = \frac{9/8 \cdot n + a}{7/8 \cdot n}$$
$$= \frac{9n+8a}{7n} = \frac{9}{7} + \frac{8}{7} \cdot \frac{a}{n}.$$

2.) $k > 0$: trivial
3.) $k \leq n$: trivial
4.) $m - n \geq \omega(\log(k))$:

$$m - n = n + a \geq n = 8k > \omega(\log(k))$$

5.) $m - \omega(\log(k)) \leq k^{O(1)}$:

$$m - \omega(\log(k)) \leq m = 2n + a = 16k + a \leq k^{O(1)}$$

6.) $q \geq 3\sqrt{k}$:

$$q \geq 16s^2 \geq 3n^2 \geq 3\sqrt{k}$$

7.) $q \geq (4s)^{m/(m-n)}$:

$$(4s)^{\frac{m}{m-n}} = (4s)^{\frac{2n+a}{2n+a-n}} \leq (4s)^{\frac{2(n+a)}{n+a}} = (4s)^2 \leq q$$

8.) $q \leq k^{O(1)}$: by hypothesis □

Parameters for U-LP With the worst-to-average-case reduction established, we propose to select the parameters for U-LP as follows. For a given message length ℓ, choose the security parameter n as a multiple of 8, and choose the maximal error sizes and the modulus as

$$\begin{aligned} s_k &= \left\lceil (2Cn)^{\frac{9}{7}} \right\rceil, \\ s_e &= \left\lceil (C(2n+\ell))^{\frac{9}{7}+\frac{8}{7}\cdot\frac{\ell}{n}} \right\rceil, \\ q &\geq \max\left\{(4s_e)^2, 8ns_es_k + 4s_e + 4\right\}, \end{aligned} \tag{2}$$

where q is prime and $C \approx 2\sqrt{2\pi e} \cdot (1 + \sqrt{8} \cdot C^*)$ with $C^* \approx 1/\sqrt{2\pi}$. See Appendix A for how to compute the constant.

2.3 Correctness and Security

We now use the collected results to prove the main theorem of this paper which relates the security of U-LP with the worst-case hardness of SIVP.

Theorem 5 *The encryption scheme U-LP as presented in Fig. 1 with parameters as in Equation (2) is correct, i.e., no decryption failures occur. Moreover, with $k \in \mathbb{N}$ big enough and $n = 8k$, U-LP is IND-CPA secure as long as* $\mathrm{SIVP}(k, \widetilde{O}(\sqrt{k} \cdot q))$ *is (quantum) hard in the worst case.*

Proof. We first prove the correctness of the scheme. As explained in Section 2.1, the cryptosystem correctly decrypts the ciphertext and returns the corresponding message if $\|Ee_1 + Se_2 + e_3\|_\infty < \lfloor q/4 \rfloor$. Since every entry of the error vector is at most $2ns_k s_e + s_e$ and since q was chosen so that $2ns_k s_e + s_e < \lfloor q/4 \rfloor$, no decryption failures occur.

The next step is the proof of security. Lemmata 1 and 2 show that U-LP is secure as long as both $\mathrm{LWE}(n, 2n, q, \mathcal{U}_q, \mathcal{U}_{s_k})$ and $\mathrm{LWE}(n, 2n+\ell, q, \mathcal{U}_q, \mathcal{U}_{s_e})$ are hard. To show the hardness of the two LWE instances, we use Lemma 4 twice. Recall that it holds for every polynomially bounded a. Applying Lemma 4 with $a = 0$ and $s = s_k$ shows that $\mathrm{LWE}(n, 2n, q, \mathcal{U}_q, \mathcal{U}_{s_k})$ is hard as long as $\mathrm{SIVP}(k, \widetilde{O}(\sqrt{k} \cdot q))$ is hard. Likewise, applying it with $a = \ell$ and $s = s_e$ shows that $\mathrm{LWE}(n, 2n + \ell, q, \mathcal{U}_q, \mathcal{U}_{s_e})$ is hard as long as worst-case $\mathrm{SIVP}(k, \widetilde{O}(\sqrt{k} \cdot q))$ is hard. Note that a is not a parameter of the scheme, and therefore both LWE instances are hard as long as worst-case $\mathrm{SIVP}(k, \widetilde{O}(\sqrt{k} \cdot q))$ is. □

3 Comparison

In this section we compare the LWE-based scheme with uniform noise U-LP to the Gaussian noise scheme LP by Lindner and Peikert [LP11]. In order to have a fair comparison, we propose parameters for the latter that assure a negligible error rate and a worst-to-average-case reduction. This, as well as proofs of correctness and security is given in Section 3.1. We present the security proof in great detail, not only for completeness, but also because through the proof we uncover two quantities that are important for the comparison, namely the worst-case problem's dimension and approximation factor. Then, in Section 3.2 we compare the two schemes on several measures including worst-case problem's dimension and approximation factor, basic parameters such as modulus q and error distribution, size of secret key, public key and cipher text, and encryption speed.

3.1 Parameter Selection for LP

We propose parameters for Lindner and Peikert's scheme LP with Gaussian noise [LP11] that assure a negligible error rate and a worst- to average-case reduction. Recall that the LP scheme differs from U-LP only in the noise and secret distribution. It uses a discrete Gaussian distribution with parameter σ for key generation and encryption. Remember that the discrete Gaussian distribution D_σ over $\mathbb{Z}$ assigns $x \in \mathbb{Z}$ a probability proportional to $\exp(-\pi x^2/\sigma^2)$.

In order to obtain a negligible error rate, we modify the correctness-result of Lemma 3.1 in [LP11] as follows.

Lemma 6 *Let $n \in \mathbb{N}$, and real $c > 1$ be such that $c \cdot \exp(\frac{1-c^2}{2}) = \sqrt{1/2}$. Then, in LP with parameters n, σ and prime $q > \frac{4c\sigma^2\sqrt{n \ln(2^{n+1})}}{\pi} + 3$, the failure probability per symbol is smaller than 2^{-n}.*

The proof of the above lemma closely resembles that of Lemma 3.1 in [LP11]. Note that for $f(c) := c \cdot \exp(\frac{1-c^2}{2})$, since f is continuous as well as $f(1) = 1$ and $\lim_{c\to\infty} f(c) = 0$, there exists $c > 1$ such that $f(c) = \sqrt{1/2}$.

Unfortunately, the parameters proposed by Lindner and Peikert do not allow a worst-to-average-case reduction. In order to have a fairer comparison to U-LP, we propose parameters

so that the underlying LWE instances are worst-case hard as in [Reg09]. By Theorem 3.2 in [LP11], LP with parameters n, σ and prime q is IND-CPA secure, assuming the hardness of $\mathrm{LWE}(n, \mathrm{poly}(n), q, D_\sigma, D_\sigma)$. Since Regev's worst-to-average-case reduction requires $\sigma > 2\sqrt{n}$, we choose $\sigma = O(\sqrt{n})$. Following Lemma 6, we demand

$$q = O(n\sqrt{n(n+1)\ln(2)}) = O(n^2).$$

Regev's reduction is not for LWE with a discrete Gaussian distribution D_σ, but for a "discretized" Gaussian distribution $\bar{\Psi}_\alpha$ over $\mathbb{Z}_q$. It was noted by Peikert [Pei10] that Regev's reduction is valid for both distributions, thus we use D_σ.

Regev's worst-to-average-case reduction comprises several steps. The first step is a reduction from a problem called Discrete Gaussian Sampling (DGS) to LWE. In order to solve $\mathrm{DGS}_{\varphi(L)}$, one has to sample from a discrete Gaussian distribution over the lattice L with parameter $\varphi(L)$. The next step is a reduction from the Generalized Independent Vector Problem (GIVP) to DGS. For an n-dimensional lattice L, $\mathrm{GIVP}_{\varphi(L)}$ is the problem of finding n linearly independent vectors with length at most $\varphi(L)$. The connection to the more standard SIVP is $\mathrm{GIVP}_{\gamma\lambda_n(L)} = \mathrm{SIVP}(n, \gamma)$. Unfortunately, the result is not for $\mathrm{GIVP}_{\varphi(L)}$ with a function $\varphi(L) = \gamma\lambda_n(L)$, but with $\varphi(L) = \gamma\eta_\varepsilon(L)$, where $\eta_\varepsilon(L)$ denotes the smoothing parameter of L (see [Reg09] for a definition). In order to get a connection to SIVP, we use a result from [MR07] that connects the smoothing parameter η_ε with the n-th successive minimum λ_n.

Theorem 7 *LP with $\sigma > 2\sqrt{n} = O(\sqrt{n})$ and prime $q > \frac{4c\sigma^2\sqrt{n\ln(2^{n+1})}}{\pi} + 3 = O(n^2)$ is secure as long as $\mathrm{SIVP}(n, \gamma)$ with $\gamma = \frac{\sqrt{8}\cdot n^{1+\beta}q}{\sqrt{\pi}\cdot\sigma} = O(n^{2.5+\beta})$ for any $\beta > 0$ is hard.*

Proof. With $\alpha := \sigma/q$, we show that the results can be combined as follows:

$$\begin{aligned}
&\text{LP with parameters } n, q, \sigma \text{ is insecure}\\
&\xRightarrow{\text{Theorem 3.2 in [LP11]}} \mathrm{LWE}(n, \mathrm{poly}(n), q, D_\sigma, D_\sigma) \text{ is easy}\\
&\xRightarrow{\text{Theorem 3.1 in [Reg09]}} \mathrm{DGS}_{\sqrt{2n}\cdot\eta_\varepsilon(L)/\alpha} \text{ is easy}\\
&\xRightarrow{\text{Lemma 3.17 in [Reg09]}} \mathrm{GIVP}_{\sqrt{8}\cdot n\eta_\varepsilon(L)/\alpha} \text{ is easy}\\
&\xRightarrow{\text{Lemma 3.3 in [MR07]}} \mathrm{SIVP}(n, \tfrac{\sqrt{8}\cdot n^{1+\beta}q}{\sqrt{\pi}\cdot\sigma}) \text{ is easy}
\end{aligned}$$

Since $\sigma > 2\sqrt{n}$, we can apply Theorem 3.1 of [Reg09] for $\varepsilon(n) := \frac{2n}{\exp(n^{2\beta})-2n}$ to show that breaking LP is at least as hard as solving $\mathrm{DGS}_{\varphi(L)}$ with $\varphi(L) := \sqrt{2n}\cdot\eta_\varepsilon(L)/\alpha$. Since $\varphi(L) \geq \sqrt{2}\cdot\eta_\varepsilon(L)$, we can apply Lemma 3.17 in [Reg09] to show moreover that breaking LP is at least as hard as solving $\mathrm{GIVP}_{2\sqrt{n}\cdot\varphi(L)} = \mathrm{GIVP}_{\sqrt{8}\cdot n\eta_\varepsilon(L)/\alpha}$. Note that $\varepsilon(n)$ was chosen such that

$$\ln(2n(1+1/\varepsilon(n))) = \ln(2n(1+\frac{\exp(n^{2\beta})-2n}{2n})) = n^{2\beta}.$$

Consequently, Lemma 3.3 of [MR07] reveals

$$\begin{aligned}
\eta_\varepsilon(L) &\leq \sqrt{\frac{\ln(2n(1+1/\varepsilon(n)))}{\pi}}\cdot\lambda_n(L)\\
&= \sqrt{\frac{n^{2\beta}}{\pi}}\cdot\lambda_n(L) = \frac{n^\beta}{\sqrt{\pi}}\cdot\lambda_n(L)
\end{aligned}$$

	Parameters				Security	
	ℓ	q	Noise		k	γ
			Key	Enc		
U-LP	1	$n^{3.7}$	$n^{1.3}$	$n^{1.4}$	$\frac{n}{8}$	$\widetilde{O}(n^{4.7})$
LP	1	n^2	$n^{0.5}$	$n^{0.5}$	n	$O(n^{2.5})$
U-LP	$\frac{n}{2}$	$n^{4.2}$	$n^{1.3}$	$n^{1.9}$	$\frac{n}{8}$	$\widetilde{O}(n^{4.7})$
LP	$\frac{n}{2}$	n^2	$n^{0.5}$	$n^{0.5}$	n	$O(n^{2.5})$

Table 1. Parameter and security comparison of our scheme U-LP with the Gaussian scheme LP for same average-case dimension n and message lengths 1, $\frac{n}{2}$. Columns 2–5 show message length ℓ, modulus q, and noise bounds for key generation and encryption, column 6 shows the worst-case dimension k and column 7 the approximation factor γ of SIVP, respectively. Note that q and noises are given as $O(\cdot)$, and that the key—enc-noise of U-LP is s_k, s_e, respectively, whereas in LP the noises are σ.

and therefore breaking LP is not easier than solving

$$\mathrm{GIVP}_{\frac{\sqrt{8}\cdot n^{1+\beta}}{\sqrt{\pi}\cdot\alpha}\lambda_n(L)} = \mathrm{SIVP}(n,\gamma)$$

with $\gamma = \frac{\sqrt{8}\cdot n^{1+\beta}}{\sqrt{\pi}\cdot\alpha} = \frac{\sqrt{8}\cdot n^{1+\beta} q}{\sqrt{\pi}\cdot\sigma} = O(n^{2.5+\beta})$. □

3.2 U-LP vs. LP

We now compare the uniform noise scheme U-LP to the Gaussian noise scheme LP. We compare the schemes in terms of their parameters, security and efficiency.

We want to note that this is a fair comparison. On one side, we crafted LP by choosing its parameters in a way to be able to apply Regev's worst- to average-case reduction (Theorem 7). Furthermore, the provided parameters entail a negligible error rate for decryption (Lemma 6). On the other side, U-LP is also worst-case secure and has no decryption failures at all (Theorem 5).

Also, note that the message length ℓ affects U-LP more than LP. This is due to the fact that ℓ influences the number of samples of the LWE instance hiding the message. The security reduction of LP does not limit the number of samples, thus q and σ are in this case independent of the number of samples. In the security reduction of U-LP, however, the number of samples is restricted. Consequently, s_e has to increase if longer messages should be encrypted, and with s_e increases q. In order to compare both schemes, we consider $\ell = 1$ and $\ell = n/2$. The second bound is reasonable for applications in hybrid encryption, where the asymmetric scheme is only used to encrypt the key of a symmetric scheme.

The main U-LP parameters are the modulus q and the noise bounds s_k and s_e. For LP the parameters are the modulus q and the Gaussian parameter σ (for the noise distribution). Sizes for the main parameters of the two schemes are given in Table 1. The modulus in U-LP is larger than in LP, even for the encryption of a single bit. In particular q has about twice as many bits in U-LP as in LP. This coincides with the observation above about the influence of ℓ on q. Comparing the parameters for the noise distribution is nontrivial and a direct comparison is misleading. This is because the distributions as well as the meaning

Scheme	ℓ	Sizes [bits]		
		Secret Key	Public Key	Ciphertext
U-LP	1	$1.3 \cdot N$	$3.71 \cdot Nn$	$3.71 \cdot N$
LP	1	$0.5 \cdot N$	$2 \cdot Nn$	$2 \cdot N$
U-LP	$\frac{n}{2}$	$0.64 \cdot Nn$	$6.21 \cdot Nn$	$6.21 \cdot N$
LP	$\frac{n}{2}$	$0.25 \cdot Nn$	$3 \cdot Nn$	$1.5 \cdot N$

Table 2. Efficiency comparison of U-LP and LP for same average-case dimension n and parameters as in Table 1, with $N = n\log(n)$. Columns three to five show the secret key, public key, and ciphertext sizes in bits, respectively. Note that we solely listed the leading term here and left out minor terms that are comparatively small.

of their noise parameters s_k, s_e and σ differ in the two schemes. Thus, a small σ in LP of order $O(\sqrt{n})$ compared to larger s_k, s_e of order $O((\sqrt{n})^3)$ in U-LP does not necessarily lead to smaller keys.

Perhaps more meaningful is to compare key and ciphertext sizes. The public key in both U-LP and LP consists of the matrices A, P over $\mathbb{Z}_q$. This means that the public key has overall $n^2 + n\ell$ entries, each of size $\log(q)$ bits. Similarly, the ciphertext has $n\ell$ entries of size $\log(q)$ bits for both schemes. For the secret key the situation is different. The secret key in U-LP consists of $n\ell$ entries with $\log(s_k)$ bits, while in LP it consists of $n\ell$ entries of $\log(k \cdot \sigma)$ bits, where k is a small constant. Table 2 shows sizes for the secret and public keys as well as for the ciphertexts for both schemes. Note that for the sizes of the keys and ciphertexts we solely write the leading term in order to provide a relatively easy, yet very precise measure which only depends on the security parameter n. Because σ is considerably smaller than s_k, the secret key in LP is smaller than in U-LP. We note that we estimate that the secret key is of maximum size $k\sigma$ for a small constant k. Nonetheless, the entries of the secret key in LP can become as large as q, though the corresponding probability is negligibly small. Thus, a strict upper bound for the secret key in LP is given by $2n\log(n)$ bits for $\ell = 1$ and $n^2\log(n)$ bits for $\ell = n/2$, respectively. Regarding the sizes of public keys and ciphertexts, the situation is quite similar. The public key in U-LP is a factor of 2 larger than in LP, and ciphertexts are even a factor of 4 larger. This increase for all sizes in U-LP is entailed by a larger modulus q and larger bounds s_k and s_e, as mentioned above.

As a measure of security we look at the underlying worst-case problem instances which are determined by the dimension k and the approximation factor γ (see Table 1). Note that the worst-case dimension k for LP is identical to the average-case dimension n. For U-LP the worst-case dimension is smaller by a factor 8 compared to LP. Furthermore, the approximation factor γ of the underlying SIVP in U-LP is roughly the square of the approximation factor in LP. Since the reduction from worst-case SIVP to the security of the two schemes is not tight, we are not confident about the overall impact of a smaller worst-case dimension and a larger approximation factor on the security.

Comparing speed is more involved as it depends on algorithms, hardware and implementations, thus this analysis should be taken with a grain of salt. We look at encryption speed. In both U-LP and LP encryption requires to multiply an $(n + \ell) \times n$-matrix by an n-vector over $\mathbb{Z}_q$ and to sample $2n + \ell$ elements from the corresponding distribution. A

single multiplication in $\mathbb{Z}_q$ takes time proportional to $\log(q)$. Thus, matrix multiplication takes time proportional to $3.7n^2(n+\ell)$ for U-LP, and $2n^2(n+\ell)$ for LP. Sampling from $\mathcal{U}_{s_e}$ in U-LP requires $\log(s_e)$ uniformly random bits, while sampling from D_σ depends on the quality of the distribution. The statistical distance between D_σ and the output of a Gaussian sampler is typically of the order of $\epsilon \cdot \sigma \log(n)$, where ϵ is the fixed point precision of the computation [GD12,BCG+13]. In order to make the statistical distance negligible on n, one must choose ϵ to be of the order of 2^{-n}. The number of uniformly random bits of a typical discrete Gaussian sampler is roughly the number of bits in the fixed point representation, thus $O(n)$.[1] Therefore, U-LP requires $O(\log(n))$ times $2n+\ell$ uniformly random bits, while LP requires $O(n)$ times $2n+\ell$ uniformly random bits in the worst case. In summary, U-LP encryption can be asymptotically faster in sampling, but linearly slower in multiplication. This theoretical analysis is thus not conclusive and the precise speed depends on the constants determined by algorithms, hardware, and implementation.

n	q	σ	Bit Security
256	378353	32	85
320	590921	35.77	116
512	1511821	45.25	228

Table 3. Hardness of LWE with Gaussian Error

n	q	s	Bit Security
488	310027967972291	278420	87
592	615698195236667	356922	118
888	2603483886956573	601141	229

Table 4. Hardness of LWE with Uniform Error

Experiments and Results To assess the practicality of our construction, we implemented both U-LP and LP in C++ using the Number Theory Library (NTL [Sho]). For a fair comparison, we instantiated both schemes with comparable average-case hardness. In order to do this, we estimated the security of the underlying LWE problems. Recall that the security of U-LP reduces to two different LWE instances, and therefore an attacker can choose which instance to attack. Since the number of samples plays only a minor role in the hardness of an LWE instance, the LWE instance with smaller noise (i.e., the instance for the key, or LWE$(n, n, q, \mathcal{U}_{s_k}, \mathcal{U}_{s_k})$) is the easier one.

The most promising attack on LWE is the decoding approach proposed by Lindner and Peikert [LP11]. The idea is to use a basis reduction (typically BKZ) followed by a search algorithm. The decoding approach is quite flexible. It allows the attacker to choose

[1] According to Galbraith and Dwarakanath [GD12] the expected number of uniformly random bits of the Knuth-Yao algorithm is close to the entropy of the distribution, thus proportional to σ and independent of ϵ.

two attack parameters that regulate the trade-off between the running time of the basis reduction, the running time of the search step and the overall success probability of the attack. Lindner and Peikert [LP11] showed how the running time of the attack for given attack parameters can be predicted. We adapted their approach and combined it with a numerical method to optimize the attack parameters (and thereby minimize the running time of the attack). Applying this approach for both schemes leads to the bit security estimates given in Tables 3 and 4 and corresponding parameter sets. One can see that for the same bit security level, the hardness of LWE with uniform error requires much larger values for the dimension n and the modulus q than in the case of LWE with error sampled from a Gaussian distribution. For example, to obtain a bit security level of about 86, one requires a nearly doubled n and a factor $8.2 \cdot 10^8$ larger modulus q. The factors for q increase for larger n, while the factors for n decrease.

Bit	Times U-LP [ms]			Times LP [ms]		
Security	KeyGen	Enc	Dec	KeyGen	Enc	Dec
87	152.3	11.8	0.026	31.1	3.1	0.014
118	209.1	16.8	0.030	49.2	4.9	0.017
229	531.2	4.3	0.052	133.1	12.4	0.026

Table 5. Performance of U-LP and LP for $\ell = 1$

Bit	Secret Key [Bytes]		Public Key [KBytes]	
Security	LP	U-LP	LP	U-LP
87	256	1159	156.26	1461.62
118	360	1406	256.80	2194.10
229	640	2220	689.47	5131.31

Table 6. Key Sizes of U-LP and LP for $\ell = 1$

In order to measure the efficiency of both schemes, i.e., the running times of each algorithm as well as the involved sizes, we conducted the following experiments using the parameters given in Tables 3 and 4.

We let each scheme iteratively and independently generate a key pair, encrypt a randomly chosen message of length ℓ, and decrypt the ciphertext about 10^6 times. For each operation we measured the corresponding time and averaged the results. Table 5 shows the running times for the key generation, encryption, and decryption algorithms of U-LP and LP in milliseconds, for the same bit security level and $\ell = 1$. As we can see, key generation in U-LP is about a factor 4–5 slower than in LP, encryption about a factor of 3.5–3.7, and decryption about a factor 1.7–2 (depending on the bit security level). We note that the factors for key generation and encryption decrease for increasing bit security, whereas the factors for decryption scatter around 1.8.

As for keys, we derived the secret and public key sizes as follows. The public key in both schemes consists of $(A, P) \in \mathbb{Z}_q^{n \times (n+\ell)}$. Therefore, one has to store $n(n+\ell)\lceil \log(q) \rceil$ bits for it. For the secret key, the method is similar for both schemes, but the results are different,

in conformance to the distribution the secret key is sampled from. This means, we obtain a secret key size of $n\ell \lceil \log(s_k) \rceil$ bits in U-LP because the secret key S is chosen from $\mathcal{U}_{s_k}^{\ell \times n}$. In LP, as the secret key is chosen according to a Gaussian distribution with parameter σ, the number of bits to store is $n\ell \lceil \log(13\sigma) \rceil$. Here we choose a maximum threshold of 13σ for the Gaussian sampling such that larger entries of the secret key have a negligibly small probability (less than 2^{-100} according to Lyubashevsky [Lyu12]). Thus, we compute the key sizes as shown in Table 6. One can see that the public key of U-LP is quite large, i.e., megabytes, even for small bit security level. Compared to LP, this is about a factor 7.4–9.4 larger, again with a decrease for increasing bit security level. The factors for the secret key, which itself is much smaller, i.e., only order of kilobytes, are in the range of 3.5–4.5.

The efficiency results are not surprising, since the dimension n and the modulus q are much larger in U-LP than in LP. Thus, the large value of q negatively influences both the key sizes and running times in a direct way. As $\log(q)$ in U-LP is at least a factor of 2.5 larger than in LP, operations get slower due to the roughly doubled bit size of involved operands. The direct influence of n and q to the key sizes is obvious. We want to note that in contrast to the large values of n and q, the noise sizes s_k and s_e are relatively small compared to q (see e.g. Table 4).

4 Conclusion

We introduced a public key encryption scheme based on the LWE problem. As a novelty, secrets and errors are sampled uniformly at random from a relatively small set rather than from the commonly used discrete Gaussian distribution. We proved the scheme secure assuming the worst-case hardness of SIVP. This was made possible by a recent result by Micciancio and Peikert who proved the worst-case hardness of LWE for small non-Gaussian noise [MP13]. We proposed asymptotic parameters for the scheme that offer a compromise between efficiency and security. And we compared the efficiency and security of the scheme with those of one of the most efficient Gaussian noise LWE-based encryption schemes [LP11].

Comparing the uniform noise scheme U-LP to the Gaussian noise scheme LP we obtained negative results. The secret and the public key in U-LP are more than four times larger than in LP, and ciphertexts are a factor of four larger. The security level of U-LP is lower than that of LP. The worst-case dimension of U-LP is smaller by a factor of eight compared to LP, and the approximation factor γ of the underlying SIVP problem in U-LP is roughly the square of the approximation factor in LP. In terms of speed the comparison is leading to similar results. Although in theory U-LP could be faster due to the easy operations involved, the choice of parameters in practice smashes the hope of being comparative. The sampling share of encryption is asymptotically faster for U-LP, but the multiplication share is linearly slower.

Summarizing, we showed that a worst-case secure LWE-based public key encryption scheme with noise and secret sampled uniformly at random from a relatively small set can be constructed. However, the resulting scheme has larger modulus, key and ciphertext sizes than a similar scheme with Gaussian noise. Moreover, key generation, encryption, and decryption are not faster in the case of uniform errors. New hardness results are required to develop efficient worst-case secure LWE-based encryption schemes with uniform noise.

References

[ABB10] Shweta Agrawal, Dan Boneh, and Xavier Boyen. Efficient lattice (H)IBE in the standard model. In Henri Gilbert, editor, *Advances in Cryptology – EUROCRYPT 2010*, volume 6110 of *Lecture Notes in Computer Science*, pages 553–572. Springer Berlin Heidelberg, 2010.

[ACPS09] Benny Applebaum, David Cash, Chris Peikert, and Amit Sahai. Fast cryptographic primitives and circular-secure encryption based on hard learning problems. In Shai Halevi, editor, *Advances in Cryptology – CRYPTO 2009*, volume 5677 of *Lecture Notes in Computer Science*, pages 595–618. Springer Berlin Heidelberg, 2009.

[ASP13] Jacob Alperin-Sheriff and Chris Peikert. Practical bootstrapping in quasilinear time. In Ran Canetti and Juan A. Garay, editors, *Advances in Cryptology – CRYPTO 2013*, volume 8042 of *Lecture Notes in Computer Science*, pages 1–20. Springer Berlin Heidelberg, 2013.

[BCG+13] Johannes Buchmann, Daniel Cabarcas, Florian Göpfert, Andreas Hülsing, and Patrick Weiden. Discrete Ziggurat: A time-memory trade-off for sampling from a Gaussian distribution over the integers. Cryptology ePrint Archive, Report 2013/510, 2013. `http://eprint.iacr.org/2013/510`.

[BGH13] Zvika Brakerski, Craig Gentry, and Shai Halevi. Packed ciphertexts in LWE-based homomorphic encryption. In Kaoru Kurosawa and Goichiro Hanaoka, editors, *Public-Key Cryptography – PKC 2013*, volume 7778 of *Lecture Notes in Computer Science*, pages 1–13. Springer Berlin Heidelberg, 2013.

[BGV12] Zvika Brakerski, Craig Gentry, and Vinod Vaikuntanathan. (Leveled) fully homomorphic encryption without bootstrapping. In *Proceedings of the 3rd Innovations in Theoretical Computer Science Conference*, ITCS'12, pages 309–325, New York, NY, USA, 2012. ACM.

[BKPW12] Mihir Bellare, Eike Kiltz, Chris Peikert, and Brent Waters. Identity-based (lossy) trapdoor functions and applications. In David Pointcheval and Thomas Johansson, editors, *Advances in Cryptology – EUROCRYPT 2012*, volume 7237 of *Lecture Notes in Computer Science*, pages 228–245. Springer Berlin Heidelberg, 2012.

[BV11a] Zvika Brakerski and Vinod Vaikuntanathan. Efficient fully homomorphic encryption from (standard) LWE. In Rafail Ostrovsky, editor, *FOCS*, pages 97–106. IEEE, 2011.

[BV11b] Zvika Brakerski and Vinod Vaikuntanathan. Fully homomorphic encryption from ring-LWE and security for key dependent messages. In Phillip Rogaway, editor, *Advances in Cryptology – CRYPTO 2011*, volume 6841 of *Lecture Notes in Computer Science*, pages 505–524. Springer Berlin Heidelberg, 2011.

[CHKP10] David Cash, Dennis Hofheinz, Eike Kiltz, and Chris Peikert. Bonsai trees, or how to delegate a lattice basis. In Henri Gilbert, editor, *Advances in Cryptology – EUROCRYPT 2010*, volume 6110 of *Lecture Notes in Computer Science*, pages 523–552. Springer Berlin Heidelberg, 2010.

[Gal13] Steven D. Galbraith. Space-efficient variants of cryptosystems based on learning with errors, 2013. Preprint, available at `http://www.math.auckland.ac.nz/~sgal018/compact-LWE.pdf`.

[GD12] Steven D. Galbraith and Nagarjun C. Dwarakanath. Efficient sampling from discrete Gaussians for lattice-based cryptography on a constrained device, 2012. Preprint, available at `http://www.math.auckland.ac.nz/~sgal018/gen-gaussians.pdf`.

[GFS+12] Norman Göttert, Thomas Feller, Michael Schneider, Johannes Buchmann, and Sorin Huss. On the design of hardware building blocks for modern lattice-based encryption schemes. In Emmanuel Prouff and Patrick Schaumont, editors, *Cryptographic Hardware and Embedded Systems – CHES 2012*, volume 7428 of *Lecture Notes in Computer Science*, pages 512–529. Springer Berlin Heidelberg, 2012.

[GGH13a] Sanjam Garg, Craig Gentry, and Shai Halevi. Candidate multilinear maps from ideal lattices. In Thomas Johansson and Phong Q. Nguyen, editors, *Advances in Cryptology – EUROCRYPT 2013*, volume 7881 of *Lecture Notes in Computer Science*, pages 1–17. Springer Berlin Heidelberg, 2013.

[GGH+13b] Sanjam Garg, Craig Gentry, Shai Halevi, Amit Sahai, and Brent Waters. Attribute-based encryption for circuits from multilinear maps. In Ran Canetti and Juan A. Garay, editors, *Advances in Cryptology – CRYPTO 2013*, volume 8043 of *Lecture Notes in Computer Science*, pages 479–499. Springer Berlin Heidelberg, 2013.

[GHS12] Craig Gentry, Shai Halevi, and Nigel P. Smart. Fully homomorphic encryption with polylog overhead. In David Pointcheval and Thomas Johansson, editors, *Advances in Cryptology – EUROCRYPT 2012*, volume 7237 of *Lecture Notes in Computer Science*, pages 465–482. Springer Berlin Heidelberg, 2012.

[GHV10] Craig Gentry, Shai Halevi, and Vinod Vaikuntanathan. A simple BGN-type cryptosystem from LWE. In Henri Gilbert, editor, *Advances in Cryptology – EUROCRYPT 2010*, volume 6110 of *Lecture Notes in Computer Science*, pages 506–522. Springer Berlin Heidelberg, 2010.

[GPV08] Craig Gentry, Chris Peikert, and Vinod Vaikuntanathan. Trapdoors for hard lattices and new cryptographic constructions. In *Proceedings of the 40th annual ACM Symposium on Theory of Computing*, STOC'08, pages 197–206, New York, NY, USA, 2008. ACM.

[GVW13] Sergey Gorbunov, Vinod Vaikuntanathan, and Hoeteck Wee. Attribute-based encryption for circuits. In *Proceedings of the 45th annual ACM Symposium on Theory of Computing*, STOC'13, pages 545–554, New York, NY, USA, 2013. ACM.

[HDWH12] Nadia Heninger, Zakir Durumeric, Eric Wustrow, and J. Alex Halderman. Mining your Ps and Qs: Detection of widespread weak keys in network devices. In *Proceedings of the 21st USENIX Security Symposium*, August 2012.

[LHA+12] Arjen K. Lenstra, James P. Hughes, Maxime Augier, Joppe W. Bos, Thorsten Kleinjung, and Christophe Wachter. Public keys. In Reihaneh Safavi-Naini and Ran Canetti, editors, *Advances in Cryptology – CRYPTO 2012*, volume 7417 of *Lecture Notes in Computer Science*, pages 626–642. Springer Berlin Heidelberg, 2012.

[LP11] Richard Lindner and Chris Peikert. Better key sizes (and attacks) for LWE-based encryption. In Aggelos Kiayias, editor, *Topics in Cryptology – CT-RSA 2011*, volume 6558 of *Lecture Notes in Computer Science*, pages 319–339. Springer Berlin Heidelberg, 2011.

[LPR10] Vadim Lyubashevsky, Chris Peikert, and Oded Regev. On ideal lattices and learning with errors over rings. In Henri Gilbert, editor, *Advances in Cryptology – EUROCRYPT 2010*, volume 6110 of *Lecture Notes in Computer Science*, pages 1–23. Springer Berlin Heidelberg, 2010.

[Lyu12] Vadim Lyubashevsky. Lattice signatures without trapdoors. In *Proceedings of the 31st Annual international conference on Theory and Applications of Cryptographic Techniques*, EUROCRYPT'12, pages 738–755, Berlin, Heidelberg, 2012. Springer-Verlag.

[Mic10] Daniele Micciancio. Duality in lattice cryptography, Public Key Cryptography, 2010. Invited talk.

[MP12] Daniele Micciancio and Chris Peikert. Trapdoors for lattices: Simpler, tighter, faster, smaller. In David Pointcheval and Thomas Johansson, editors, *Advances in Cryptology – EUROCRYPT 2012*, volume 7237 of *Lecture Notes in Computer Science*, pages 700–718. Springer Berlin Heidelberg, 2012.

[MP13] Daniele Micciancio and Chris Peikert. Hardness of SIS and LWE with small parameters. In Ran Canetti and Juan A. Garay, editors, *Advances in Cryptology - CRYPTO 2013*, volume 8042 of *Lecture Notes in Computer Science*, pages 21–39. Springer Berlin Heidelberg, 2013.

[MR07] Daniele Micciancio and Oded Regev. Worst-case to average-case reductions based on Gaussian measures. *SIAM Journal on Computing*, 37(1):267–302, 2007.

[Pei09] Chris Peikert. Public-key cryptosystems from the worst-case shortest vector problem: Extended abstract. In *Proceedings of the 41st annual ACM Symposium on Theory of Computing*, STOC'09, pages 333–342, New York, NY, USA, 2009. ACM.

[Pei10] Chris Peikert. An efficient and parallel Gaussian sampler for lattices. In Tal Rabin, editor, *Advances in Cryptology – CRYPTO 2010*, volume 6223 of *Lecture Notes in Computer Science*, pages 80–97. Springer Berlin Heidelberg, 2010.

[PG12] Thomas Pöppelmann and Tim Güneysu. Towards efficient arithmetic for lattice-based cryptography on reconfigurable hardware. In Alejandro Hevia and Gregory Neven, editors, *Progress in Cryptology – LATINCRYPT 2012*, volume 7533 of *Lecture Notes in Computer Science*, pages 139–158. Springer Berlin Heidelberg, 2012.

[PW08] Chris Peikert and Brent Waters. Lossy trapdoor functions and their applications. In *Proceedings of the 40th annual ACM Symposium on Theory of Computing*, STOC'08, pages 187–196, New York, NY, USA, 2008. ACM.

[Reg09] Oded Regev. On lattices, learning with errors, random linear codes, and cryptography. *J. ACM*, 56(6):34:1–34:40, September 2009.

[Sho] Victor Shoup. Number theory library (NTL) for C++.

[SS11] Damien Stehlé and Ron Steinfeld. Making NTRU as secure as worst-case problems over ideal lattices. In Kenneth G. Paterson, editor, *Advances in Cryptology – EUROCRYPT 2011*, volume 6632 of *Lecture Notes in Computer Science*, pages 27–47. Springer Berlin Heidelberg, 2011.

[SS13] Damien Stehlé and Ron Steinfeld. Making NTRUEncrypt and NTRUSign as secure as standard worst-case problems over ideal lattices. Cryptology ePrint Archive, Report 2013/004, 2013. `http://eprint.iacr.org/2013/004`.

[SSTX09] Damien Stehlé, Ron Steinfeld, Keisuke Tanaka, and Keita Xagawa. Efficient public key encryption based on ideal lattices. In Mitsuru Matsui, editor, *Advances in Cryptology – ASIACRYPT 2009*, volume 5912 of *Lecture Notes in Computer Science*, pages 617–635. Springer Berlin Heidelberg, 2009.

[Wee12] Hoeteck Wee. Dual projective hashing and its applications – lossy trapdoor functions and more. In David Pointcheval and Thomas Johansson, editors, *Advances in Cryptology – EUROCRYPT 2012*, volume 7237 of *Lecture Notes in Computer Science*, pages 246–262. Springer Berlin Heidelberg, 2012.

[WHCB13] Patrick Weiden, Andreas Hülsing, Daniel Cabarcas, and Johannes Buchmann. Instantiating treeless signature schemes. Cryptology ePrint Archive, Report 2013/065, 2013. `http://eprint.iacr.org/2013/065`.

A Computation of Constant C

We show here how to compute the value given for the constant C in the parameter selection in Equation (2). Since the constant C in (2) is the same as in Theorem 4.6 of [MP13], a look at its proof reveals $C = 4C'$, furthermore relying on Theorem 4.5 in [MP13]. In the latter, C' is given as "the universal constant hidden by the $O(\cdot)$ notation from Lemma 4.4". So, we have to determine C' as the constant hidden by $O(\sigma ms/\sqrt{\ell})$ by a closer look to the proof of Lemma 4.4 in [MP13] and unveil several estimations covered by $O(\cdot)$ notations.

Throughout the analysis we make use of (tuples of) the set $X \subseteq \{-s, \ldots, s\}$ and the function $f : \mathbb{Z}^m \to \mathbb{Z}^\ell, x \mapsto (I_\ell, Y) \cdot x$, where I_ℓ is the $\ell \times \ell$ identity matrix and Y is a $\ell \times m-\ell$ discrete Gaussian matrix with parameter σ, i.e., the entries of Y are chosen according to $D_{\mathbb{Z},\sigma}$.

By Lemma 2.9 of [MP12] we obtain that for any $t \geq 0$ the largest singular value $s_1(Y)$ of Y is at most

$$s_1(Y) \leq c^* \sigma(\sqrt{\ell} + \sqrt{m-\ell} + t)$$

except with probability at most $2\exp(\delta)\exp(-\pi t^2)$, and c^* is very close to $1/\sqrt{2\pi}$ for discrete Gaussians. We set $t := \sqrt{\ell} + \sqrt{m-\ell}$, and obtain

$$s_1(Y) \leq 2c^* \sigma(\sqrt{\ell} + \sqrt{m-\ell}) \leq 2\sqrt{2} \cdot c^* \sigma \sqrt{m}$$

except with probability at most

$$\begin{aligned} &2\exp(\delta)\exp(-\pi(\sqrt{\ell} + \sqrt{m-\ell})^2) \leq \\ &\qquad \leq 2\exp(\delta)\exp(-\pi(m + \sqrt{\ell(m-\ell)}) = 2^{-\Omega(m)}. \end{aligned}$$

This is correct since $t = \sqrt{\ell} + \sqrt{m-\ell}$ is indeed ≥ 0, as the function $g(\ell) := \sqrt{\ell} + \sqrt{m-\ell}$ satisfies $g(\ell) \in [\sqrt{m}, \sqrt{2m}]$ for $\ell \in [0, m]$: Using its derivative one can show that $g(\ell)$ has its maximum value $\sqrt{2m}$ for $\ell = m/2$, has a gradient greater or equal zero for $\ell \in [0, m/2]$ and a gradient smaller or equal zero for $\ell \in [m/2, m]$. This means that $g(\ell)$ is monotonically increasing in $[0, m/2]$ and monotonically decreasing in $[m/2, m]$.

To proceed with the analysis, for $u_1 \in X^\ell$ we have $\|u_1\| \leq s\sqrt{\ell}$ and for $u_2, x \in X^{m-\ell}$

$$\|Yu_2\| \leq \max_{0 \neq x \in X^{m-\ell}} \|Yx\| = s_1(Y) \cdot \|x\| \leq s_1(Y) \cdot s\sqrt{m-\ell}.$$

With this we can bound the size of the images of f with preimages from X^m as follows:

$$\begin{aligned}
\|f(u)\| &= (I_\ell, Y)(u_1, u_2)^T \\
&\leq (\sqrt{\ell} + s_1(Y) \cdot \sqrt{m-\ell})s \\
&\leq (\sqrt{\ell} + 2\sqrt{2} \cdot c^* \sigma\sqrt{m} \cdot \sqrt{m-\ell})s \\
&\overset{(*)}{\leq} (\sigma m + 2\sqrt{2} \cdot c^* \sigma\sqrt{m}\sqrt{m})s \\
&= (1 + 2\sqrt{2} \cdot c^*)\sigma m s,
\end{aligned}$$

where in $(*)$ we have $\sqrt{\ell} \leq \sqrt{m} \leq m \leq \sigma m$ for $\sigma > 1$ and $\sqrt{m-\ell} \leq \sqrt{m}$. Using $R := (1+2\sqrt{2}\cdot c^*)\sigma m s$ and the fact $\sqrt{\ell}/2 \leq R$, we can bound the number of integer points in the ℓ-dimensional zero-centered ball with radius R, and thus the maximal number of images of X^m under f, as

$$|f(X^m)| \leq (R + \sqrt{\ell}/2)^\ell V_\ell \leq (2R)^\ell V_\ell.$$

Here $V_\ell = \pi^{\ell/2}/(\ell/2)!$ is the volume of the ℓ-dimensional unit ball. "Tweaking" Stirling's formula

$$(\ell/2)! \geq (\ell/(2e))^{\ell/2}\sqrt{2\pi\ell/2} \overset{\ell \geq 1, \sqrt{\pi} \geq 1}{\geq} (\sqrt{\ell}/\sqrt{2e})^\ell$$

we can thus upper bound

$$|f(X^m)| \leq (2R\sqrt{2\pi e}/\sqrt{\ell})^\ell = (2\sqrt{2\pi e} \cdot (1 + 2\sqrt{2} \cdot c^*)\sigma m s/\sqrt{\ell})^\ell.$$

Finally, the constant C is at most $2\sqrt{2\pi e} \cdot (1 + 2\sqrt{2} \cdot c^*)$ with $c^* \approx 1/\sqrt{2\pi}$. So, we have

$$C = 2\sqrt{e} \cdot (\sqrt{2\pi} + 2\sqrt{2}) \approx 17.59.$$

B Ring Variant of U-LP

In this section we present the ring variant of our scheme U-LP. We note that the security proof and the parameters chosen for U-LP do not necessarily hold for the ring variant. This means more investigation about the ring variant has to be rolled out in order to estimate its security. Nonetheless, we present the scheme here for completeness.

All operations are performed in the ring $\mathcal{R}_q = \mathbb{Z}_q[x]/\langle x^n + 1\rangle$. Norms of polynomials correspond to norms of vectors that can be obtained using the coefficient embedding, i.e., for a polynomial $\mathbf{a} = \sum_{i=0}^{n} a_i x^i$ we write its coefficient vector as $(a_0, \ldots, a_{n-1})^T$.

Note that the encode and decode functions also have to be changed. In particular, $\mathsf{encode} : \mathbb{Z}_2^\ell \to \mathcal{R}_q$ and $\mathsf{decode} : \mathcal{R}_q \to \mathbb{Z}_2^\ell$ such that $\mathsf{decode}(\mathsf{encode}(\mu) + \mathbf{e}) = \mu$ for any message $\mu \in \mathbb{Z}_2^\ell$ of length ℓ and error polynomial $\mathbf{e}$ with $\|\mathbf{e}\|_\infty < \lfloor q/4 \rfloor$.

With everything replaced by the corresponding polynomial version, we obtain the ring variant of our U-LP scheme as shown in Fig. 2.

KeyGen(n, q, s_k)**:** Sample $\mathbf{a} \leftarrow \mathcal{U}_q^n$, $\mathbf{e} \leftarrow \mathcal{U}_{s_k}^n$ and $\mathbf{s} \leftarrow \mathcal{U}_{s_k}^n$, and let $\mathbf{p} = \mathbf{e} - \mathbf{s}\mathbf{a} \in \mathcal{R}_q$. Return public key $(\mathbf{a}, \mathbf{p})$ and secret key $\mathbf{s}$.

Enc$(\mu, (\mathbf{a}, \mathbf{p}), s_e)$**:** Sample $\mathbf{e}_1 \leftarrow \mathcal{U}_{s_e}^n$, $\mathbf{e}_2 \leftarrow \mathcal{U}_{s_e}^n$, $\mathbf{e}_3 \leftarrow \mathcal{U}_{s_e}^n$, compute $\mu' = \mathsf{encode}(\mu) \in \mathcal{R}_q$, $\mathbf{c}_1 = \mathbf{a}\mathbf{e}_1 + \mathbf{e}_2$ and $\mathbf{c}_2 = \mathbf{p}\mathbf{e}_1 + \mathbf{e}_3 + \mu'$, and return ciphertext $(\mathbf{c}_1, \mathbf{c}_2)$.

Dec$((\mathbf{c}_1, \mathbf{c}_2), \mathbf{s})$**:** Return message $\mathsf{decode}(\mathbf{s}\mathbf{c}_1 + \mathbf{c}_2) \in \mathbb{Z}_2^\ell$.

Fig. 2. Ring Variant of U-LP