Provably Secure LWE-Encryption with Uniform Secret

Daniel Cabarcas Florian Göpfert Patrick Weiden

Technische Universität Darmstadt
Department of Computer Science
Darmstadt, Germany
{cabarcas,fgoepfert,pweiden}@cdc.informatik.tu-darmstadt.de

Abstract

In this paper we present the (to the best of our knowledge) first LWE-based encryption scheme that removes the need of Gaussian sampling for the error, i.e. the discrete Gaussian distribution is replaced by the uniform distribution on a (small) set, which at the same time preserves the underlying worst-case hardness. This shows that provable security and efficiency do not necessarily have to mutually exclude each other. We give an asymptotic parameter instantiation for our scheme, as well as some hardness results for LWE which might be of independent interest.

Keywords: LWE, Encryption, Lattice-Based Cryptography.

1 Introduction

Lattice-based cryptography has aroused a lot of interest in the last years. From a certain point of view, two main paths have been developed: Provably secure schemes using worst-case to average-case reduction on the one hand and practical schemes on the other hand. The provably secure schemes are mainly based on two problems: Learning With Errors (LWE) and the Small Integer Solution (SIS). We will focus on LWE in this paper, leaving SIS for future work. LWE can be shown to be hard in the average-case as long as certain lattice problems are hard in the worst-case, if a gaussian

*This work was supported by CASED (http://www.cased.de).

sampler is used for noise generation. In practice, however, a significant amount of time is spent on Gaussian sampling (e.g. [WHCB13]). Some implementations solve this problem by using uniform error and pretty much ignoring the provable security (e.g [GLP12], [HPS98]). A recent result from Miccianco and Peikert gives access to another solution of this problem. They showed in [MP13] that LWE with (small) uniform noise can remain its worst-case hardness if the amount of samples is restricted. We use this result to construct the (to the best of our knowledge) first encryption scheme that uses no Gaussian sampling and at the same time remains its worst-case security. For this purpose, we slightly modify the scheme by Lindner and Peikert [LP11].

1.1 Related Work

As mentioned before, we combine results from [MP13] with the scheme presented in [LP11] to construct our scheme. Güneysu et al. present in [GLP12] a highly efficient signature scheme using uniform noise, but without a worst-case hardness proof. Another provably worst-case secure cryptosystem can be found in [Reg09]. Other LWE-based cryptosystems are for example [HPS98, KTX07, PVW08, PW08, Pei09, LPR10, SS11].

1.2 Future Work

Although our scheme is asymptotically provably secure, the hardness of concrete instances remains unclear. This problem needs to be addressed before concrete parameters can be proposed and the scheme can be compared to schemes using Gaussian error. Since our parameters are bigger than parameters for schemes based on Gaussian error, it is unclear which scheme will be more practical. Consequently, improving existing Gaussian sampler remains an interesting problem. Finally, the results of [MP13] can moreover be used to construct other provably secure schemes without Gaussian samples.

1.3 Notation

For integers $n, m, q \in \mathbb{N}$ and probability distributions $\mathcal{X}, \mathcal{Y}$, we denote by $\mathrm{LWE}(n, m, q, \mathcal{X}, \mathcal{Y})$ the decisional variant of the Learning with Errors (LWE) problem (see [Reg09]), where n is the dimension of the underlying lattice, q denotes the modulus, m is the number of given samples, and where secrets are sampled according to the "secret distribution" $\mathcal{X}$ and errors are sampled according to the "error distribution" $\mathcal{Y}$. By $\mathrm{SIVP}(k, \gamma)$, we will furthermore denote the Shortest Independent Vector Problem (SIVP) in dimension k

with approximation factor γ (see [Reg09]), where γ is a function in n. We write $\mathcal{U}_t$ for the uniform distribution on $\{0, \ldots, t-1\}$. Accordingly, since all entries are in $\mathbb{Z}_q$, we denote the uniform distribution on $\mathbb{Z}_q$ as $\mathcal{U}_q$. At last, we write $\tilde{O}(f)$ for $O(f \cdot \log^c(f))$ and some constant $c > 0$.

2 Hardness Results for LWE

This section is devided in two parts. In the first, we give an instances of LWE and prove its security. Afterwards, we present the new encryption-scheme with asymptotic parameters and base its security on this LWE instance.

2.1 From Uniform Secret to Secret from Error Distribution

Since [ACPS09] is known that LWE becomes not easier if the secret is chosen according to the error distribution (instead of chosen uniformly from $\mathbb{Z}_q$). Since the errors are small, this leads to smaller secrets (and thus to smaller keys). Unfortunately, the reduction from LWE with small secret to LWE with uniform secret comes with a cost: It loses n samples. This is of course not of interest in the "classical" definition of LWE (since the attacker has access to arbitrary many samples), but does matter if the amount of samples is limited. We now give the reduction, following the proof in [ACPS09].

Theorem 1. *If there is an algorithm that can solve* $\mathrm{LWE}(n, m, q, \mathcal{X}, \mathcal{X})$ *with probability* p *for an arbitrary distribution* $\mathcal{X}$ *over* $\mathbb{Z}_q$*, then there exists an algorithm for solving* $\mathrm{LWE}(n, n+m, q, \mathcal{U}_q, \mathcal{X})$ *with probability* $p \cdot c$*, where* $c = \prod_{i=1}^{n}(1 - q^i)$*.*

Proof. We denote the samples with (a_i, b_i), with $b_i := \langle a_i, s \rangle + e_i$ (in case of LWE) or $b_i \xleftarrow{\$} \mathbb{Z}_q$ (in case of uniform).

1.) Use the first n samples to define

$$\overline{b} := \begin{pmatrix} b_1 \\ \vdots \\ b_n \end{pmatrix}, \quad \overline{e} := \begin{pmatrix} e_1 \\ \vdots \\ e_n \end{pmatrix}, \quad \overline{A} := \begin{pmatrix} a_1, a_2, \ldots, a_n \end{pmatrix}.$$

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

2.) Transform the other samples to (a'_i, b'_i) where

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

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

3.) Query the oracle on the samples $\{(a'_i, b'_i) \mid i \in \{1, \ldots, m\}\}$ and return the output of the oracle (either uniform or LWE).

We now analyse the algorithm for an input from the LWE and the uniform distribution.

- *Uniform input*: If the samples b_{n+i} are uniform, so are the transformed samples b'_i. The oracle in step 3.) is called with uniform input.
- LWE *input:* If the samples have the form $b_i := \langle a_i, s\rangle + e_i$, then

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

  and the oracle indeed is called with LWE-samples where the secret is sampled according to $\mathcal{X}$.

The overall success probability of this algorihm is therefore $p \cdot c$ where

$$\begin{aligned}
c &= \Pr[\overline{A} \text{ is invertible} \mid \overline{A} \xleftarrow{\$} \mathbb{Z}_q^{n\times n}] \\
&= \left(1 - \frac{1}{q^n}\right)\left(1 - \frac{q}{q^n}\right) \cdot \ldots \cdot \left(1 - \frac{q^{n-1}}{q^n}\right) \\
&= \prod_{i=0}^{n-1} \left(1 - q^{i-n}\right).
\end{aligned}$$

□

2.2 Instantiating LWE with Small Uniform Error

The instantiation of our encryption scheme uses a recent result of Micciancio and Peikert in [MP13] about parameter choices for the LWE problem with uniform error. They showed that by limiting the number of samples, one can prove the worst-case hardness of LWE for small non-Gaussian errors. For completeness we give an adapted version here.

$m-n$	s	k	$q \geq$	secret
$\frac{1}{2}n - \frac{3}{2}k$	$\sqrt{Cm}$	$< \frac{1}{3}n$	$c \cdot m^{\frac{3}{2}\frac{n-k}{n-3k}} > c \cdot m^{\frac{3}{2}}$	$\mathcal{U}_q$
$n - 2k$	Cm	$< \frac{1}{2}n$	$c \cdot m^{2\frac{n-k}{n-2k}} > c \cdot m^2$	$\mathcal{U}_q$
$\frac{3}{2}n - \frac{5}{2}k$	$(Cm)^{\frac{3}{2}}$	$< \frac{3}{5}n$	$c \cdot m^{\frac{5}{2}\frac{n-k}{n-\frac{5}{3}k}} > c \cdot m^{\frac{5}{2}}$	$\mathcal{U}_q$
$\frac{1}{2}n - \frac{5}{2}k$	$(Cm)^{\frac{3}{2}}$	$< \frac{1}{5}n$	$c \cdot m^{\frac{5}{2}\frac{n-k}{n-\frac{5}{3}k}} > c \cdot m^{\frac{5}{2}}$	$\mathcal{U}_s$
$\frac{1}{2}n - \frac{5}{2}k$	$(Cm)^{\frac{3}{2}}$	$\frac{1}{5}n - \frac{2}{5}\log^2(n)$	$c \cdot m^{\frac{3}{2}(1+\frac{n}{n+\log^2(n)})}$	$\mathcal{U}_s$

Table 1: Parameter sets for provably secure LWE instances. The last column shows the "secret distribution"; errors are chosen according to $\mathcal{U}_s$.

Theorem 2 ([MP13], Theorem 4.6)**.** *Let*

$$0 < k \leq n \leq m - \omega(\log(k)) \leq k^{O(1)}, \tag{1}$$

$$s \geq (Cm)^{(m-(n-k))/(n-k)} \tag{2}$$

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

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

For the set $X = \{0, \ldots, s-1\}^m$ of size $|X| = s^m$, 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* $\mathrm{SIVP}(k, \tilde{O}(\sqrt{k} \cdot q))$ *is (quantum) hard to approximate in the worst-case.*

We note that if we need more than n extra samples (i.e. $m - n > n$, like we require in our instantiation of the LWE-based scheme) this leads to

$$\frac{m-(n-k)}{n-k} > \frac{n+k}{n-k} > 1,$$

which means that the size of the errors is superlinear.

Table 1 gives a short overview of some instantiations of hard LWE-problems. Given $m - n$, we can calculate the smallest s satisfying (2) and, since $m - n > 0$, we get an upper bound for k. We use inequality (3) to get a lower bound for q. The constant C comes from Theorem 2 and the constant c can be derived easily if C is known. We will exemplarily show the calculations for the first row of Table 1: Since $m = \frac{3}{2}(n-k)$ we have $s \geq \sqrt{Cm}$, and by

$$\frac{m}{m-n} = \frac{\frac{3}{2}(n-k)}{\frac{1}{2}n - \frac{3}{2}k} = 3\frac{n-k}{n-3k},$$

we obtain

$$q \geq (4s)^{\frac{m}{m-n}} \geq (4\sqrt{Cm})^{3\frac{n-k}{n-3k}} \geq (4\sqrt{C})^3 m^{\frac{3}{2}\frac{n-k}{n-3k}}. \tag{4}$$

Thus we can choose $c = (4\sqrt{C})^3$.

Please note that in (4), q is a monotonously increasing function in k (for $0 < k < n/3$) and therefore the modulus q increases with increasing worst-case dimension k. Since this is true for all parameter sets presented, there is one more tradeoff to be made. The first two instantiations lead to a sublinear number of extra samples (i.e. $m-n < n$) and cannot be used with a secret generated according to the "error distribution". The third instantiation provides more than n extra samples and can be used for a cryptosystem with small secret, if the worst-case dimension is restricted. This is done in the fourth row: Theorem 1 can be used to show that the LWE-instance presented in row four of Table 1 (using the smaller "error distribution" for the secret) is at least as hard as the LWE-instance presented in row three (with larger secret). The instantiation in the last row is a concrete version of the fourth row which we will use to construct our scheme. In comparison to the parameters proposed in [Reg09] for the LWE-scheme with Gaussian noise (i.e. with $q = O(n^2)$ and $\sigma = \frac{n^{1.5}}{\log^2(n)}$), we have to pay with slightly bigger error and modulus for being able to use uniform noise. We will now give a proof for the worst-case hardness of our concrete instantiation.

Corollary 3. *Let $n \in \mathbb{N}$ be big enough such that $k := \left\lfloor \frac{1}{5}n - \frac{2}{5}\log^2(n) \right\rfloor$ is positive, let $m = \left\lfloor \frac{5}{2}(n-k) \right\rfloor$, $s := \left\lceil (Cm)^{\frac{3}{2}} \right\rceil$, and q be a prime satisfying $q \geq \max\{3\sqrt{k}, (4s)^{\frac{m}{m-n}}\}$. If q is polynomially bounded in k (i.e. $q \leq k^{O(1)}$), then $\mathrm{LWE}(n, m, q, \mathcal{U}_q, \mathcal{U}_s)$ is at least as hard as $\mathrm{SIVP}(k, \tilde{O}(\sqrt{k} \cdot q))$ in the worst case.*

Proof. The theorem will mainly follow from Theorem 2. We show that its preconditions are fulfilled:

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

$$(Cm)^{\frac{m-(n-k)}{n-k}} \leq (Cm)^{\frac{\frac{5}{2}(n-k)-(n-k)}{n-k}} = (Cm)^{\frac{3}{2}} \leq \left\lceil (Cm)^{\frac{3}{2}} \right\rceil = s$$

2.) $k > 0$: precondition

3.) $k \leq n$: trivial

4.) $m - n \geq \omega(\log(k))$:

$$m - n = \left\lfloor \frac{5}{2}(n-k) \right\rfloor - n > \frac{3}{2}n - \frac{5}{2}k - 1 = \frac{3}{2}n - \frac{5}{2}\left\lfloor \frac{1}{5}n - \frac{2}{5}\log^2(n) \right\rfloor - 1$$
$$\geq n + \log^2(n) - 1 \geq n - 1 \geq \omega(\log(n)) \geq \omega(\log(k))$$

5.) $m - \omega(\log(k)) \leq k^{O(1)}$: Since $10k = \left\lfloor 2n - 4\log^2(n) \right\rfloor > n$ for big enough n, we have

$$m - \omega(\log(k)) \leq m \leq 3n < 30k = k^{O(1)}.$$

□

3 Provably Secure Encryption with Uniform Error

In this section we first present the description of our scheme which makes use of small uniform noise (and secret) in contrast to previous schemes which used a discrete Gaussian distribution for noise generation. We also present a concrete parameter instantiation for our scheme which shows that provable security and efficiency do not necessarily have to mutually exclude each other. After this we prove the hardness of our proposed scheme, relying its security to the well-known SIVP using the average-case to worst-case reduction.

3.1 Description of the Scheme and Parameter Instantiation

We will now explain our scheme, which is an adaption of the LWE-encryption scheme presented in [LP11], as well as how to choose parameters. In contrast to this instantiation, our scheme does not require a discrete Gaussian sampler, since all secrets and errors are chosen according to a uniform distribution. According to Lindner and Peikert, we also require 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 $\|e\|_\infty \leq \lfloor \frac{q}{4} \rfloor$ (for a concrete instance see [LP11]). Our scheme makes use of the following parameters: an integer modulus $q \geq 2$, integral dimensions $n_1, n_2 \geq 2$, maximal error sizes s_k, s_e for key generation and encryption, respectively, and an integral message length $\ell \geq 1$.

A description of the scheme is given in Figure 1. As already mentioned, the presented scheme is very similar to Lindner and Peikert's LWE-based encryption scheme in [LP11]. It differs in key generation and encryption due to our use of secrets and errors chosen according to a uniform distribution

KeyGen$(n_1, n_2, q, s_k, s_e, \ell)$**:** Choose $\overline{A} \leftarrow \mathcal{U}_q^{n_1 \times n_2}$, $R_1 \leftarrow \mathcal{U}_{s_k}^{n_1 \times \ell}$ and $R_2 \leftarrow \mathcal{U}_{s_k}^{n_2 \times \ell}$, and let $P = R_1 - \overline{A}R_2 \in \mathbb{Z}_q^{n_1 \times \ell}$. Return public key $(\overline{A}, P)$ and secret key R_2.

Enc$(\mu, (\overline{A}, P))$**:** Choose $e_1 \leftarrow \mathcal{U}_{s_e}^{n_1}$, $e_2 \leftarrow \mathcal{U}_{s_e}^{n_2}$, $e_3 \leftarrow \mathcal{U}_{s_e}^{\ell}$, compute $\overline{\mu} = \mathsf{encode}(\mu) \in \mathbb{Z}_q^{\ell}$, $c_1^t = e_1^t\overline{A} + e_2^t$ and $c_2^t = e_1^t P + e_3^t + \overline{\mu}^t$, and return ciphertext (c_1, c_2).

Dec$((c_1, c_2), R_2)$**:** Return message $\mathsf{decode}(c_1^t R_2 + c_2^t) \in \mathbb{Z}_2^{\ell}$.

Figure 1: LWE-Encryption Scheme with Uniform Error

instead of a discrete Gaussian distribution. In detail, this means we choose R_1, R_2 in key generation and e_1, e_2, e_3 in encryption according to uniform distributions $\mathcal{U}_{s_k}$ and $\mathcal{U}_{s_e}$, respectively.

Regarding the parameters of our scheme, we solely base the security on the single security parameter n_2 which corresponds to the number of LWE-samples in the generation of the keys. The other parameters are chosen depending on n_2, and the sample instantiation of the parameters used in our scheme can be seen in Table 2. As will be shown in the proof of Theorem 5, the conditions $q \geq 3\sqrt{k_k}$ and $q \geq 3\sqrt{k_e}$ will implicitly be fulfilled by the correctness condition $q \geq 4(n_1 + n_2)s_e s_k + 4s_e + 1$.

3.2 Security of the Scheme

We will first sketch the proof of the hardness of our instantiation:

$$
\begin{aligned}
&\text{Scheme with parameters } n_1, n_2, q, s_k, s_e, \ell \text{ easy} \\
\xRightarrow{\text{Thm 4}}\ &\text{LWE}(n_2, n_1, q, \mathcal{U}_{s_k}, \mathcal{U}_{s_k}) \text{ easy} \vee \\
&\text{LWE}(n_1, n_2 + \ell, q, \mathcal{U}_{s_e}, \mathcal{U}_{s_e}) \text{ easy} \\
\xRightarrow{\text{Thm 1}}\ &\text{LWE}(n_2, n_1 + n_2, q, \mathcal{U}_q, \mathcal{U}_{s_k}) \text{ easy} \vee \\
&\text{LWE}(n_1, n_1 + n_2 + \ell, q, \mathcal{U}_q, \mathcal{U}_{s_e}) \text{ easy} \\
\xRightarrow{\text{Cor 3}}\ &\text{SIVP}(k_k, \tilde{O}(\sqrt{k_k} \cdot q)) \text{ easy} \vee \\
&\text{SIVP}(k_e, \tilde{O}(\sqrt{k_e} \cdot q)) \text{ easy}
\end{aligned}
$$

In order to prove the hardness of the scheme, we will need the hardness of two LWE-instances: The first instance will be needed to hide the secret key (i.e. an attacker cannot distinguish the public key from uniform random matrices), the second will be needed to hide the message.

n_2	security parameter
k_k	$\left\lfloor \frac{1}{5}(n_2 - 2\log^2(n_2)) \right\rfloor$
n_1	$\left\lfloor \frac{1}{2}(3n_2 - 5k_k) \right\rfloor$
k_e	$\left\lfloor \frac{1}{5}(n_1 - 2\log^2(n_1)) \right\rfloor$
ℓ	$\left\lfloor \frac{1}{2}(3n_1 - 5k_e) \right\rfloor - n_2$
s_k	$\left\lceil (C(n_1+n_2))^{\frac{3}{2}} \right\rceil$
s_e	$\left\lceil (C(n_1+n_2+\ell))^{\frac{3}{2}} \right\rceil$
q	smallest prime $\geq \max\left\{ (4s_k)^{\frac{n_1+n_2}{n_1}}, (4s_e)^{\frac{n_1+n_2+\ell}{n_2+\ell}}, 4(n_1+n_2)s_e s_k + 4s_e + 1 \right\}$

Table 2: Parameter Instantiation for the Scheme

Theorem 4. *The encryption scheme presented in Figure 1 is secure as long as* $\mathrm{LWE}(n_2, n_1, q, \mathcal{U}_{s_k}, \mathcal{U}_{s_k})$ *and* $\mathrm{LWE}(n_1, n_2+\ell, q, \mathcal{U}_{s_e}, \mathcal{U}_{s_e})$ *are hard.*

Proof. Following the proof of Theorem 3.2 in [LP11]. □

We now use the collected results to prove the main theorem of this paper which (quantumly) relies the hardness of the presented encryption scheme onto the hardness of worst-case SIVP.

Theorem 5. *The encryption scheme presented in Figure 1 with parameters as in Table 2 is correct, i.e. no decryption errors occur, and it is secure for big enough* n_2 *as long as* $\mathrm{SIVP}(k_k, \tilde{O}(\sqrt{k_k} \cdot q))$ *and* $\mathrm{SIVP}(k_e, \tilde{O}(\sqrt{k_e} \cdot q))$ *are (quantum) hard in the worst case.*

Proof. We first prove the correctness of our scheme. Let $r_1, \ldots, r_\ell$ be the columns of $R = \begin{pmatrix} R_1 \\ R_2 \\ I_\ell \end{pmatrix} \in \mathbb{Z}_q^{(n_1+n_2+\ell)\times \ell}$ and $e = \begin{pmatrix} e_1 \\ e_2 \\ e_3 \end{pmatrix} \in \mathbb{Z}_q^{n_1+n_2+\ell}$. By the definition of the encode and decode functions, the cryptosystem decrypts the j-th bit of the message correctly if $|\langle e, r_j \rangle| < \lfloor q/4 \rfloor$. For $j = 1, \ldots, \ell$ we have $|\langle e, r_j \rangle| \leq (n_1+n_2)s_e s_k + s_e$, such that from $(n_1+n_2)s_e s_k + s_e < \lfloor q/4 \rfloor$ we obtain $|\langle e, r_j \rangle| < \lfloor q/4 \rfloor$ for $j = 1, \ldots, \ell$, and no decryption errors occur.

We now prove the hardness of our scheme. Theorems 4 and 1 show that the scheme is hard as long as LWE$(n_2, n_1+n_2, q, \mathcal{U}_q, \mathcal{U}_{s_k})$ (i.e. LWE with $m = n_1 + n_2$ and $n = n_2$) and LWE$(n_1, n_1+n_2+\ell, q, \mathcal{U}_q, \mathcal{U}_{s_e})$ (i.e. LWE with $m = n_1 + n_2 + \ell$ and $n = n_1$) are hard. Note that

$$3\sqrt{k_k} < k_k < \frac{1}{5}n_2 < n_2 < 4(n_1+n_2)s_e s_k + 4s_e + 1 \leq q$$

and

$$3\sqrt{k_e} < k_e < \frac{1}{5}n_1 < n_1 < 4(n_1+n_2)s_e s_k + 4s_e + 1 \leq q$$

for large enough n_2. Since

$$n_1 + n_2 = \left\lfloor \frac{1}{2}(3n_2 - 5k_k) \right\rfloor + n_2 = \left\lfloor \frac{5}{2}(n_2 - k_k) \right\rfloor$$

and

$$n_1 + n_2 + \ell = n_1 + n_2 + \left\lfloor \frac{1}{2}(3n_1 - 5k_e) \right\rfloor - n_2 = \left\lfloor \frac{5}{2}(n_1 - k_e) \right\rfloor$$

and $k_k, k_e > 0$ for big enough n_2, we can apply Corollary 3 to prove the hardness of these problems if q is polynomially bounded with respect to k_k and k_e.

We will now show that this is true. First we note that $k_k^{O(1)} = k_e^{O(1)}$ since $n_2 < n_1 < 2n_2$. We will therefore in the following only write $k^{O(1)}$ for this class. Note that s_k is in this class since for big enough n_2 we have $10k_k > n_2$ such that

$$s_k \approx C(n_1+n_2)^{\frac{3}{2}} \approx C\frac{5}{2}(n_2-k_k)^{\frac{3}{2}} < C\frac{5}{2}(10k_k - k_k)^{\frac{3}{2}} \leq k^{O(1)}.$$

A similar calculation gives us $s_e < k^{O(1)}$. We will furthermore need that n_1 and n_2 are polynomially bounded in k. This is the case since

$$\begin{aligned} n_1 &\approx \frac{3}{2}n_2 - \frac{5}{2}k_k \approx \frac{3}{2}n_2 - \frac{1}{2}n_2 + \log^2(n_2) \\ &= n_2 + \log^2(n_2) < 10k_k + \log^2(10k_k) \leq k^{O(1)}. \end{aligned}$$

Again, a similar calculation results in $n_2 \leq k^{O(1)}$. In order to bound the exponent, we note that

$$\frac{n_1+n_2}{n_1} \approx \frac{2n_2 + \log^2(n_2)}{n_2 + \log^2(n_2)} = 1 + \frac{n_2}{n_2 + \log^2(n_2)} < 2$$

and (similarly) $\frac{n_1+n_2+\ell}{n_2+\ell} < 2$. Putting this together, we obtain $q \leq k^{O(1)}$. $\square$

References

[ACPS09] Benny Applebaum, David Cash, Chris Peikert, and Amit Sahai. Fast cryptographic primitives and circular-secure encryption based on hard learning problems. In *Proceedings of the 29th Annual International Cryptology Conference on Advances in Cryptology*, CRYPTO '09, pages 595–618, Berlin, Heidelberg, 2009. Springer-Verlag.

[GLP12] Tim Güneysu, Vadim Lyubashevsky, and Thomas Pöppelmann. Practical lattice-based cryptography: A signature scheme for embedded systems. *Cryptographic Hardware and Embedded Systems–CHES 2012*, pages 530–547, 2012.

[HPS98] Jeffrey Hoffstein, Jill Pipher, and Joseph H. Silverman. Ntru: A ring-based public key cryptosystem. In JoeP. Buhler, editor, *Algorithmic Number Theory*, volume 1423 of *Lecture Notes in Computer Science*, pages 267–288. Springer Berlin Heidelberg, 1998.

[KTX07] Akinori Kawachi, Keisuke Tanaka, and Keita Xagawa. Multi-bit cryptosystems based on lattice problems. In Tatsuaki Okamoto and Xiaoyun Wang, editors, *Public Key Cryptography – PKC 2007*, volume 4450 of *Lecture Notes in Computer Science*, pages 315–329. Springer Berlin Heidelberg, 2007.

[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.

[MP13] Daniele Micciancio and Chris Peikert. Hardness of SIS and LWE with small parameters. Cryptology ePrint Archive, Report 2013/069, 2013. `http://eprint.iacr.org/`.

[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.

[PVW08] Chris Peikert, Vinod Vaikuntanathan, and Brent Waters. A framework for efficient and composable oblivious transfer. *Advances in Cryptology–CRYPTO 2008*, pages 554–571, 2008.

[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.

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

[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/`.