TRANSCRIPT SECURE SIGNATURES BASED ON MODULAR LATTICES

JEFF HOFFSTEIN, JILL PIPHER, JOHN M. SCHANCK, JOSEPH H. SILVERMAN, WILLIAM WHYTE

Abstract. We introduce the notion of a class of lattice-based digital signature schemes based on modular properties of the coordinates of lattice vectors. We also suggest a method of making such schemes transcript secure via a rejection sampling technique of Lyubashevsky (2009). A particular instantiation of this approach is given, using NTRU lattices. Although the scheme is not supported by a formal security reduction, we present arguments for its security and derive concrete parameters based on the performance of state-of-the-art lattice reduction and enumeration techniques.

1. Introduction

In the GGH and NTRUSign signature schemes [4, Sections 7.4,7.5] a document to be signed is thought of as a point $\boldsymbol{m}$ in $\mathbb{Z}^n$. A lattice L has a private basis, known only to the signer, that is reasonably short and close to orthogonal. The signer uses the private base to solve a CVP and locate a point $\boldsymbol{s} \in L$ that lies reasonably close to $\boldsymbol{m}$. A verifier of the signature checks that $\boldsymbol{s}$ is indeed a point in the lattice L, and that the Euclidean distance between $\boldsymbol{s}$ and $\boldsymbol{m}$ is shorter than some prespecified bound. The security assumption underlying the acceptance of the signature is that it is hard to find a point in L that is close to $\boldsymbol{m}$ unless one knows the private short basis for L.

A major difficulty with these signature schemes is the fact that when the private basis is used to locate $\boldsymbol{s}$, the difference $\boldsymbol{s} - \boldsymbol{m}$ has the form

$$\boldsymbol{s} - \boldsymbol{m} = \sum_{i=1}^{n} \epsilon_i \boldsymbol{v}_i,$$

where $\boldsymbol{v}_1, \dots, \boldsymbol{v}_n$ is the private basis and where each $|\epsilon_i| \le 1/2$. Thus $\boldsymbol{s} - \boldsymbol{m}$ is a point in the interior of the fundamental parallelepiped associated to the private basis. If the signature is obtained by, say, Babai's

Date: June 13, 2014.

2010 *Mathematics Subject Classification.* 94A60.

Key words and phrases. digital signature, NTRU signature, NTRUMLS.

rounding approach, the ϵ_i will be randomly and uniformly distributed in the interval $(-1/2, 1/2)$. A long transcript of signatures then corresponds to a large collection of points randomly and uniformly distributed inside the parallelepiped, and a sufficiently long transcript eventually reveals the vertices of the parallelepiped, and the secret basis. This was demonstrated successfully in [11, 12, 1].

It has been proposed that such an attack could be thwarted by carefully signing in such a way that the distribution of the ϵ_i was controlled, and it was proved that using such methods it is possible to construct signing protocols where the transcript contains no information pertaining to the private basis[2]. The difficulty of this approach is that the process of controlling the distribution of the ϵ_i is computationally expensive.

One contribution of the present work is to show that knowledge of the exact vector $\boldsymbol{m}$ that is close to $\boldsymbol{s}$ can be hidden, making it very difficult to construct the vector $\boldsymbol{s} - \boldsymbol{m}$.

Very roughly, the idea is as follows. Fix a public small prime p, and, rather than taking $\boldsymbol{m}$ to be a point in $\mathbb{Z}^n$, consider it instead to be a point $\boldsymbol{m}_p \in (\mathbb{Z}/p\mathbb{Z})^n$. Fix also a specific public region $\mathcal{R}$ in $\mathbb{Z}^n$. The region $\mathcal{R}$ should be sufficiently large that the volume of $\mathcal{R}$, which we denote by $|\mathcal{R}|$, satisfies

$$\frac{|\mathcal{R}|}{p^n} > C^n,$$

for a sufficiently large C. Precise examples will be given below. A signature on $\boldsymbol{m}_p$ is a point $\boldsymbol{s} \in L \cap \mathcal{R}$, with $\boldsymbol{s} \equiv \boldsymbol{m}_p \pmod{p}$.

Signing is accomplished as follows. To sign $\boldsymbol{m}_p \in (\mathbb{Z}/p\mathbb{Z})^n$, a random point $\boldsymbol{s}_0 \in L \cap \mathcal{R}$ is chosen. Let M be a matrix whose rows are the private basis, and let M_p be the reduction of this basis modulo p. Use M_p to find $\boldsymbol{v}_p \in (\mathbb{Z}/p\mathbb{Z})^n$ such that

$$\boldsymbol{s}_0 + \boldsymbol{v}_p \cdot M_p \equiv \boldsymbol{m}_p \pmod{p}.$$

Let $\boldsymbol{v}$ be the lift of $\boldsymbol{v}_p$ to $\mathbb{Z}^n$ with coefficients chosen from the interval $(-p/2, p/2)$. Then as M is a short basis and p is small, the vector $\boldsymbol{v} \cdot M$ will also be short, and $\boldsymbol{s} = \boldsymbol{s}_0 + \boldsymbol{v} \cdot M$ will satisfy $\boldsymbol{s} \equiv \boldsymbol{m}_p \pmod{p}$. Also, as $\boldsymbol{s}_0$ was chosen to lie in $L \cap \mathcal{R}$, and $\boldsymbol{v} \cdot M$ is short, there is a reasonable chance that $\boldsymbol{s}$ will also lie in $L \cap \mathcal{R}$. The algorithm of choosing $\boldsymbol{s}_0$ and solving for $\boldsymbol{s}$ is repeated until $\boldsymbol{s} \in L \cap \mathcal{R}$.

Any lattice point $\boldsymbol{s}$ satisfying $\boldsymbol{s} \equiv \boldsymbol{m}_p \pmod{p}$ is a valid signature, and such points will, with high probability, be uniformly distributed throughout $\mathcal{R}$. Anyone can use a public basis to find a point in L with the desired properties modulo p, and if $\mathcal{R}$ is sufficiently large it is easy, using a short basis, to find points of $L \cap \mathcal{R}$, but if one does not know a short basis, then it is hard to satisfy both criteria simultaneously.

To create a useful transcript of $\boldsymbol{s} - \boldsymbol{s}_0$, an attacker must also locate the nearby lattice point $\boldsymbol{s}_0$, However, for any $\boldsymbol{s} \in L \cap \mathcal{R}$, there will be many potential $\boldsymbol{s}_0'$ that are close to $\boldsymbol{s}$. In fact, if it is not only required that $\boldsymbol{s} \in L \cap \mathcal{R}$, but also that $\boldsymbol{s}$ lies at least a certain distance inside the boundary of $\mathcal{R}$, then it can be shown that with equal probability any s_0' within a fixed radius of $\boldsymbol{s}$ could have been the actual $\boldsymbol{s}_0$ used in the signing process. This idea can be used to give a proof that the transcript contains no information about the private basis. This aspect of the approach is inspired by a rejection sampling technique of Lyubashevsky [7, 8, 9].

Another contribution of this paper is a particular, efficient, instantiation of this idea using the class of NTRU lattices. We make this choice for two reasons. First, there is a natural dimension doubling: the dimension is $n = 2N$, where N is the number of coordinates needed to determine a point. Second, the lattice can be sufficiently well described using only half of a complete basis, and this half can be made quite short and sufficiently orthogonal. We will refer to this new signature scheme as an *NTRU Modular Lattice Signature Scheme*, or NTRUMLS for short.

2. Description of NTRUMLS

Notation
We work in the ring

$$\mathcal{R} = \mathcal{R}_N = \frac{\mathbb{Z}[x]}{\langle x^N - 1 \rangle}.$$

We implicitly identify each element of $\mathcal{R}$ with the unique polynomial of degree less than N in its congruence class. Having done this, we identify a polynomial with its vector of coefficients in $\mathbb{Z}^N$. Writing an element $\boldsymbol{f} \in \mathcal{R}$ as

$$\boldsymbol{f} = \sum_{i=0}^{N-1} a_i x^i,$$

we set

$$\|\boldsymbol{f}\| = \max_{0 \le i < N} |a_i|,$$
$$\mathcal{R}(k) = \{\boldsymbol{f} \in \mathcal{R} : \|\boldsymbol{f}\| \le k\}.$$

So for example, $\mathcal{R}(3/2)$ is the set of trinary polynomials. We set the convention that if the coefficients of a polynomial are defined modulo q

with q even and we lift to a polynomial in $\mathcal{R}(q/2)$, then the lifted coefficients are chosen to satisfy $-q/2 \le a_i < q/2$.

We also fix a hash function

$$\mathsf{Hash} : \mathcal{R}(q/2) \times \{0,1\}^* \longrightarrow \mathcal{R}(p/2)^2.$$

System Parameters

N	dimension parameter
p	a small odd prime
q	an integer larger, and relatively prime to, p
B_s, B_t	norm constraints

The B_s and B_t parameters serve primarily to fine tune the balance between security and performance. Reducing B_s and B_t may, for instance, allow one to choose a smaller q, but this may come at the expense of making it difficult for an honest party to compute a signature. Typical values of B_s and B_t satisfy $B_s = pB_t$, and

$$\|\boldsymbol{a} * \boldsymbol{b}\| \le B_t \quad \text{for all } \boldsymbol{a}, \boldsymbol{b} \in \mathcal{R}\left(\frac{p}{2}\right). \tag{2.1}$$

There will be further conditions on (N, p, q) to prevent search and lattice attacks, while still making it possible to find valid signatures; see Sections 4 and 5 for details.

Private Key Choose polynomials

$$\boldsymbol{f} \xleftarrow{\$} p\mathcal{R}(3/2) \qquad \text{and} \qquad \boldsymbol{g} \xleftarrow{\$} \mathcal{R}(p/2).$$

Writing $\boldsymbol{f} = p\boldsymbol{F}$, so $\boldsymbol{F}$ is trinary, check that both $\boldsymbol{g}$ and $\boldsymbol{F}$ are invertible modulo q and modulo p. Sample a new pair if they are not. (We remark that the probability of $\boldsymbol{g}$ and $\boldsymbol{F}$ being invertible is quite high if $(x^N - 1)/(x - 1)$ does not have low degree factors when reduced modulo p and q.)

The private signing key is the pair $(\boldsymbol{f}, \boldsymbol{g})$.

Public Key The public verification key is the polynomial

$$\boldsymbol{h} \equiv \boldsymbol{f}^{-1} * \boldsymbol{g} \pmod{q}.$$

Also let

$$L_{\boldsymbol{h}} = \left\{(\boldsymbol{s}, \boldsymbol{t}) \in \mathcal{R}^2 : \boldsymbol{t} \equiv \boldsymbol{h} * \boldsymbol{s} \pmod{q}\right\}$$

be the usual NTRU lattice associated to $\boldsymbol{h}$.

We will often consider subsets of $L_{\boldsymbol{h}}$ consisting of vectors of bounded norm. This will be denoted by

$$L_{\boldsymbol{h}}(k_1, k_2) = L_{\boldsymbol{h}} \cap \big(\mathcal{R}(k_1) \times \mathcal{R}(k_2)\big).$$

Document Hashes and Valid Signatures A document hash is a $2N$-vector

$$(\boldsymbol{s}_p, \boldsymbol{t}_p) \in \mathcal{R}(p/2)^2,$$

i.e.,

$$\big\|(\boldsymbol{s}_p, \boldsymbol{t}_p)\big\| = \max\big\{\|\boldsymbol{s}_p\|, \|\boldsymbol{t}_p\|\big\} \le p/2.$$

A valid signature on the document hash $(\boldsymbol{s}_p, \boldsymbol{t}_p)$ for the signing key $\boldsymbol{h}$ is a $2N$-vector $(\boldsymbol{s}, \boldsymbol{t}) \in \mathcal{R}^2$ satisfying:

(a) $(\boldsymbol{s}, \boldsymbol{t}) \in L_{\boldsymbol{h}}\Big(\dfrac{q}{2} - B_s, \dfrac{q}{2} - B_t\Big)$.

(b) $(\boldsymbol{s}, \boldsymbol{t}) \equiv (\boldsymbol{s}_p, \boldsymbol{t}_p) \pmod{p}$.

Signing Algorithm Input $(\mu, \boldsymbol{h})$, where $\mu \in \{0,1\}^*$ is a document to be signed.

```
1: $(\boldsymbol{s}_p, \boldsymbol{t}_p) \longleftarrow \mathsf{Hash}\big(\boldsymbol{h}, \mu\big)$
2: repeat
3:     $\boldsymbol{r} \xleftarrow{\$} \mathcal{R}\left(\left\lfloor \frac{q}{2p} + \frac{1}{2} \right\rfloor\right)$
4:     $\boldsymbol{s}_0 \longleftarrow \boldsymbol{s}_p + p\boldsymbol{r}$
5:     $\boldsymbol{t}_0 \longleftarrow \boldsymbol{h} * \boldsymbol{s}_0 \pmod{q}$ with $\boldsymbol{t}_0 \in \mathcal{R}(q/2)$
6:     $\boldsymbol{a} \longleftarrow \boldsymbol{g}^{-1} * (\boldsymbol{t}_p - \boldsymbol{t}_0) \pmod{p}$ with $\boldsymbol{a} \in \mathcal{R}(p/2)$
7:     $(\boldsymbol{s}, \boldsymbol{t}) \longleftarrow (\boldsymbol{s}_0, \boldsymbol{t}_0) + (\boldsymbol{a} * \boldsymbol{f}, \boldsymbol{a} * \boldsymbol{g})$
8: until $\|\boldsymbol{s}\| \le \frac{q}{2} - B_s$ and $\|\boldsymbol{t}\| \le \frac{q}{2} - B_t$
9: return $(\boldsymbol{s}, \boldsymbol{t}, \mu)$
```

Remark 1. Notice the rejection criterion implicit in Step 8 of the signing algorithm. We compute a potential signature $(\boldsymbol{s}, \boldsymbol{t})$, but then we reject it if it is too big; specifically, we reject $(\boldsymbol{s}, \boldsymbol{t})$ if it falls outside of $L_{\boldsymbol{h}}\big(\frac{q}{2} - B_s, \frac{q}{2} - B_t\big)$.

Verification Algorithm Input $(\boldsymbol{s}, \boldsymbol{t}, \mu, \boldsymbol{h})$

```
1: $\mathit{valid} \longleftarrow$ yes
2: if $(\boldsymbol{s}, \boldsymbol{t}) \notin L_{\boldsymbol{h}}\left(\frac{q}{2} - B_s, \frac{q}{2} - B_t\right)$ then $\mathit{valid} \longleftarrow$ no endif
3: if $(\boldsymbol{s}, \boldsymbol{t}) \not\equiv (\boldsymbol{s}_p, \boldsymbol{t}_p) \pmod{p}$ then $\mathit{valid} \longleftarrow$ no endif
4: return $\mathit{valid}$
```

Proposition 2. *The Signing Algorithm produces signatures that are verified as valid by the Verification Algorithm.*

Proof. This is an easy exercise. □

3. Transcript Security

In this section we prove that under some reasonable assumptions, a transcript of signatures created using the signing algorithm contains no information that is not already available to someone who knows the public verification key $\boldsymbol{h}$. We do this by showing that a person who knows $\boldsymbol{h}$ can produce a transcript of pairs

$$(\text{Valid Signature}_i, \text{Document Hash}_i)_{i=1,2,3,\dots}$$

that is statistically indistinguishable from an analogous transcript produced using the signing algorithm and the private key $(\boldsymbol{f}, \boldsymbol{g})$. We start by analyzing the transcript created using the signing algorithm and $(\boldsymbol{f}, \boldsymbol{g})$. We note that the rejection sampling condition is what allows us to prove that the resulting signatures are uniformly distributed in a certain space of allowable signatures.

We assume that our hash function outputs document hashes

$$(\boldsymbol{s}_p, \boldsymbol{t}_p) \in \mathcal{R}(p/2)^2$$

that are uniformly distributed on $\mathcal{R}(p/2)^2$. We use Steps 3 through 7 of the Signing Algorithm to define a signing function

$$(\boldsymbol{s}, \boldsymbol{t}) = \sigma'(\boldsymbol{f}, \boldsymbol{g}, \boldsymbol{s}_p, \boldsymbol{t}_p, \boldsymbol{r}).$$

Thus σ' is a map

$$\sigma' : \overbrace{p\mathcal{R}\left(\frac{3}{2}\right) \times \mathcal{R}\left(\frac{p}{2}\right)}^{\text{private key }(\boldsymbol{f},\boldsymbol{g})} \times \overbrace{\mathcal{R}\left(\frac{p}{2}\right) \times \mathcal{R}\left(\frac{p}{2}\right)}^{\text{document hash }(\boldsymbol{s}_p,\boldsymbol{t}_p)} \times \overbrace{\mathcal{R}\left(\left\lfloor \frac{q}{2p} - \frac{1}{2} \right\rfloor\right)}^{\text{random element }\boldsymbol{r}}$$
$$\longrightarrow \underbrace{L_{\boldsymbol{h}}\left(\frac{q}{2} + B_s, \frac{q}{2} + B_t\right)}_{\text{potential signature }(\boldsymbol{s},\boldsymbol{t})}$$

given explicitly by

$$\sigma'(\boldsymbol{f}, \boldsymbol{g}, \boldsymbol{s}_p, \boldsymbol{t}_p, \boldsymbol{r}) = (\boldsymbol{s}_0 + \boldsymbol{a} * \boldsymbol{f}, \boldsymbol{t}_0 + \boldsymbol{a} * \boldsymbol{g}), \tag{3.1}$$

where

$$\boldsymbol{s}_0 = \boldsymbol{s}_p + p\boldsymbol{r}, \tag{3.2}$$

$$\boldsymbol{t}_0 \equiv \boldsymbol{h} * \boldsymbol{s}_0 \pmod{q} \quad \text{with } \boldsymbol{t}_0 \in \mathcal{R}(q/2), \tag{3.3}$$

$$\boldsymbol{a} \equiv \boldsymbol{g}^{-1} * (\boldsymbol{t}_p - \boldsymbol{t}_0) \pmod{p} \quad \text{with } \boldsymbol{a} \in \mathcal{R}(p/2). \tag{3.4}$$

We will write

$$\Omega' = p\mathcal{R}\left(\frac{3}{2}\right) \times \mathcal{R}\left(\frac{p}{2}\right) \times \mathcal{R}\left(\frac{p}{2}\right) \times \mathcal{R}\left(\frac{p}{2}\right) \times \mathcal{R}\left(\left\lfloor \frac{q}{2p} - \frac{1}{2} \right\rfloor\right)$$

for the domain of σ'.

We now introduce rejection sampling by defining

$$\Omega_{B_s,B_t} = \left\{ (\boldsymbol{f},\boldsymbol{g},\boldsymbol{s}_p,\boldsymbol{t}_p,\boldsymbol{r}) \in \Omega' : \begin{array}{r} (\boldsymbol{s},\boldsymbol{t}) := \sigma'(\boldsymbol{f},\boldsymbol{g},\boldsymbol{s}_p,\boldsymbol{t}_p,\boldsymbol{r}) \\ = (\boldsymbol{s}_0 + \boldsymbol{a} * \boldsymbol{f}, \boldsymbol{t}_0 + \boldsymbol{a} * \boldsymbol{g}), \\ \left\|\boldsymbol{s}\right\| \le \frac{q}{2} - B_s, \left\|\boldsymbol{t}\right\| \le \frac{q}{2} - B_t, \\ \left\|\boldsymbol{a} * \boldsymbol{f}\right\| \le B_s, \left\|\boldsymbol{a} * \boldsymbol{g}\right\| \le B_t \end{array} \right\}.$$

The restriction of σ' to Ω_{B_s,B_t}, which we denote by σ, is then a map

$$\sigma : \Omega_{B_s,B_t} \longrightarrow L_{\boldsymbol{h}}\left(\frac{q}{2} - B_s, \frac{q}{2} - B_t\right).$$

To ease notation, we let

$$A = \left\lfloor \frac{q}{2p} + \frac{1}{2} \right\rfloor,$$

so by Step 3 of the Signing Algorithm, the random element $\boldsymbol{r}$ used to generate a signature is chosen uniformly from the set $\mathcal{R}(A)$. The following proposition says that σ gives the uniform distribution on $L_{\boldsymbol{h}}\left(\frac{q}{2} - B_s, \frac{q}{2} - B_t\right)$.

Proposition 3. *The signature function σ has the following property: For a given*

$$\textit{private key} \quad (\boldsymbol{f},\boldsymbol{g}) \in p\mathcal{R} \times \mathcal{R},$$
$$\textit{document hash} \quad (\boldsymbol{s}_p,\boldsymbol{t}_p) \in \mathcal{R}\left(\frac{p}{2}\right) \times \mathcal{R}\left(\frac{p}{2}\right),$$

the output of σ, when queried on uniformly random $\boldsymbol{r} \in \mathcal{R}(A)$, is uniformly distributed over the set

$$\left\{ (\boldsymbol{s},\boldsymbol{t}) \in L_h\left(\frac{q}{2} - B_s, \frac{q}{2} - B_t\right) : (\boldsymbol{s},\boldsymbol{t}) \equiv (\boldsymbol{s}_p,\boldsymbol{t}_p) \pmod{p} \right\}.$$

of valid signatures for $(\boldsymbol{s}_p,\boldsymbol{t}_p)$. Equivalently, the size of the set

$$\{\boldsymbol{r} \in \mathcal{R}(A) : \sigma(\boldsymbol{f},\boldsymbol{g},\boldsymbol{s}_p,\boldsymbol{t}_p,\boldsymbol{r}) = (\boldsymbol{s},\boldsymbol{t})\}$$

is the same for all

$$(\boldsymbol{s},\boldsymbol{t}) \in L_{\boldsymbol{h}}\left(\frac{q}{2} - B_s, \frac{q}{2} - B_t\right) \quad \textit{satisfying} \quad (\boldsymbol{s},\boldsymbol{t}) \equiv (\boldsymbol{s}_p,\boldsymbol{t}_p) \pmod{p}.$$

Proof. Since we know from Proposition 2 that $\sigma(\boldsymbol{f},\boldsymbol{g},\boldsymbol{s}_p,\boldsymbol{t}_p,\boldsymbol{r})$ is congruent to $(\boldsymbol{s}_p,\boldsymbol{t}_p)$ modulo p, it is clear that there is zero probability of generating the signature $(\boldsymbol{s},\boldsymbol{t})$ if $(\boldsymbol{s},\boldsymbol{t}) \not\equiv (\boldsymbol{s}_p,\boldsymbol{t}_p) \pmod{p}$. So we assume henceforth that

$$(\boldsymbol{s},\boldsymbol{t}) \equiv (\boldsymbol{s}_p,\boldsymbol{t}_p) \pmod{p}. \tag{3.5}$$

The random element $\boldsymbol{r}$ used to generate a signature is chosen uniformly from the set $\mathcal{R}(A)$, so there are $(2A+1)^N$ possible choices for r.

Hence the probability of obtaining $(\boldsymbol{s},\boldsymbol{t})$ as a signature on $(\boldsymbol{s}_p,\boldsymbol{t}_p)$ is equal to $(2A+1)^{-N}$ times the number of elements in the set

$$\Sigma(\boldsymbol{f},\boldsymbol{g},\boldsymbol{s},\boldsymbol{t}) = \bigl\{\boldsymbol{r}\in\mathcal{R}(A) : \sigma(\boldsymbol{f},\boldsymbol{g},\boldsymbol{s}_p,\boldsymbol{t}_p,\boldsymbol{r}) = (\boldsymbol{s},\boldsymbol{t})\bigr\}. \tag{3.6}$$

The key to counting the size of the set $\Sigma(\boldsymbol{f},\boldsymbol{g},\boldsymbol{s},\boldsymbol{t})$ is the bijection described in the following lemma.

Lemma 4. *Let*

$$\mathcal{C} = \left\{\boldsymbol{b}\in\mathcal{R}\left(\frac{p}{2}\right) : \bigl\|\boldsymbol{b}*\boldsymbol{f}\bigr\| \le B_s \text{ and } \bigl\|\boldsymbol{b}*\boldsymbol{g}\bigr\| \le B_t\right\},$$

and let

$$(\boldsymbol{s},\boldsymbol{t}) \in L_{\boldsymbol{h}}\left(\frac{q}{2}-B_s,\frac{q}{2}-B_t\right) \quad \text{satisfy} \quad (\boldsymbol{s},\boldsymbol{t}) \equiv (\boldsymbol{s}_p,\boldsymbol{t}_p) \pmod{p}.$$

Then there is a well-defined bijection of sets

$$\begin{aligned} \phi : \mathcal{C} &\longrightarrow \Sigma(\boldsymbol{f},\boldsymbol{g},\boldsymbol{s},\boldsymbol{t}),\\ \boldsymbol{b} &\longmapsto \frac{\boldsymbol{s}-\boldsymbol{s}_p}{p} - \boldsymbol{b}*\frac{\boldsymbol{f}}{p}. \end{aligned} \tag{3.7}$$

Proof. First, since the coefficients of $\boldsymbol{s}-\boldsymbol{s}_p$ are multiples of p, and similarly $\boldsymbol{f}\in p\mathcal{R}(3/2)$ has coefficients divisible by p, we see that the polynomial on the right-hand side of (3.7) has coefficients in $\mathbb{Z}$.

We next need to show that $\phi(\boldsymbol{b})\in\Sigma(\boldsymbol{f},\boldsymbol{g},\boldsymbol{s},\boldsymbol{t})$, which by the definition of $\Sigma(\boldsymbol{f},\boldsymbol{g},\boldsymbol{s},\boldsymbol{t})$ means showing that $\phi(\boldsymbol{b})\in\mathcal{R}(A)$ and

$$\sigma\bigl(\boldsymbol{f},\boldsymbol{g},\boldsymbol{s}_p,\boldsymbol{t}_p,\phi(\boldsymbol{b})\bigr) = (\boldsymbol{s},\boldsymbol{t}).$$

First note that because $\boldsymbol{s}\in\mathcal{R}\left(\frac{q}{2}-B_s\right)$, $\boldsymbol{s}_p\in\mathcal{R}\left(\frac{p}{2}\right)$, and $\boldsymbol{b}\in\mathcal{C}$, the triangle inequality gives

$$\bigl\|\phi(\boldsymbol{b})\bigr\| = \left\|\frac{1}{p}\left(\boldsymbol{s}-\boldsymbol{s}_p-\boldsymbol{b}*\boldsymbol{f}\right)\right\| \le \left\lfloor\frac{\frac{q}{2}-B_s+\frac{p}{2}+B_s}{p}\right\rfloor = A.$$

The use of the floor function is justified by noting that $\phi(\boldsymbol{b})$ has integer coefficients. This establishes that $\phi(\boldsymbol{b})\in\mathcal{R}\,(A)$.

Next we use the four formulas (3.1)–(3.4) to compute the signature $\sigma\bigl(\boldsymbol{f},\boldsymbol{g},\boldsymbol{s}_p,\boldsymbol{t}_p,\phi(\boldsymbol{b})\bigr)$:

$$\begin{aligned} \boldsymbol{s}_0 &= \boldsymbol{s}_p + p\phi(\boldsymbol{b})\\ &= \boldsymbol{s}_p + p\left(\frac{\boldsymbol{s}-\boldsymbol{s}_p}{p} - \boldsymbol{b}*\frac{\boldsymbol{f}}{p}\right)\\ &= \boldsymbol{s}-\boldsymbol{b}*\boldsymbol{f}, \qquad (3.8)\\ \boldsymbol{t}_0 &\equiv \boldsymbol{h}*\boldsymbol{s}_0 \pmod{q}\\ &\equiv \boldsymbol{h}*(\boldsymbol{s}-\boldsymbol{b}*\boldsymbol{f}) \pmod{q} \end{aligned}$$

$$\begin{aligned} &\equiv \boldsymbol{h} * \boldsymbol{s} - \boldsymbol{b} * \boldsymbol{g} \pmod{q} \quad \text{since } \boldsymbol{h} \equiv \boldsymbol{f}^{-1} * \boldsymbol{g} \pmod{q}, \\ &\equiv \boldsymbol{t} - \boldsymbol{b} * \boldsymbol{g} \pmod{q} \quad \text{since } (\boldsymbol{s}, \boldsymbol{t}) \in L_{\boldsymbol{h}}. \end{aligned} \tag{3.9}$$

Since $(\boldsymbol{s}, \boldsymbol{t}) \in L_{\boldsymbol{h}}\bigl(\frac{q}{2} - B_s, \frac{q}{2} - B_t\bigr)$ and $\boldsymbol{b} \in \mathcal{C}$, we have

$$\begin{aligned} \|\boldsymbol{s}_0\| &\le \|\boldsymbol{s}\| + \|\boldsymbol{b} * \boldsymbol{f}\| = \frac{q}{2} - B_s + B_s = \frac{q}{2}, \\ \|\boldsymbol{t}_0\| &\le \|\boldsymbol{t}\| + \|\boldsymbol{b} * \boldsymbol{g}\| = \frac{q}{2} - B_t + B_t = \frac{q}{2}, \end{aligned}$$

i.e. (3.9), similar to (3.8), is an equality, not just a congruence. Continuing with the computation of $\sigma\bigl(\boldsymbol{f}, \boldsymbol{g}, \boldsymbol{s}_p, \boldsymbol{t}_p, \phi(\boldsymbol{b})\bigr)$, we use (3.5) to compute

$$\boldsymbol{a} \equiv \boldsymbol{g}^{-1} * (\boldsymbol{t}_p - \boldsymbol{t}_0) \equiv \boldsymbol{b} \pmod{p}.$$

(Note that $\boldsymbol{t} \equiv \boldsymbol{t}_p \pmod{p}$ from (3.4).) Since both $\boldsymbol{a}$ and $\boldsymbol{b}$ are in $\mathcal{R}(p/2)$, this tells us that $\boldsymbol{a} = \boldsymbol{b}$.

We now use (3.1) to compute the signature

$$\begin{aligned} \sigma\bigl(\boldsymbol{f}, \boldsymbol{g}, \boldsymbol{s}_p, \boldsymbol{t}_p, \phi(\boldsymbol{b})\bigr) &= (\boldsymbol{s}_0 + \boldsymbol{a} * \boldsymbol{f}, \boldsymbol{t}_0 + \boldsymbol{a} * \boldsymbol{g}) \quad \text{definition of } \sigma, \\ &= (\boldsymbol{s} - \boldsymbol{b} * \boldsymbol{f} + \boldsymbol{a} * \boldsymbol{f}, \boldsymbol{t} - \boldsymbol{b} * \boldsymbol{g} + \boldsymbol{a} * \boldsymbol{g}) \\ &\qquad\qquad\qquad \text{from (3.8) and (3.9)}, \\ &= (\boldsymbol{s}, \boldsymbol{t}) \quad \text{since } \boldsymbol{a} = \boldsymbol{b}. \end{aligned}$$

Hence directly from the definition (3.6) of the set $\Sigma(\boldsymbol{f}, \boldsymbol{g}, \boldsymbol{s}, \boldsymbol{t})$, we see that

$$\phi(\boldsymbol{b}) \in \Sigma(\boldsymbol{f}, \boldsymbol{g}, \boldsymbol{s}, \boldsymbol{t}).$$

We next fix an $\boldsymbol{r} \in \Sigma(\boldsymbol{f}, \boldsymbol{g}, \boldsymbol{s}, \boldsymbol{t})$ and compute how many $\boldsymbol{b} \in \mathcal{C}$ satisfy $\phi(\boldsymbol{b}) = \boldsymbol{r}$. Since all coefficients of the polyomials $\boldsymbol{s} - \boldsymbol{s}_p$ and $\boldsymbol{f}$ are divisible by p, to ease notation we write

$$\boldsymbol{s} - \boldsymbol{s}_p = p\boldsymbol{S} \quad \text{and} \quad \boldsymbol{f} = p\boldsymbol{F}.$$

We recall that by assumption, the polynomial $\boldsymbol{F}$ is invertible modulo p. We have

$$\begin{aligned} \phi(\boldsymbol{b}) = \boldsymbol{r} &\iff \boldsymbol{S} - \boldsymbol{b} * \boldsymbol{F} = \boldsymbol{r} \\ &\qquad \iff \boldsymbol{b} \equiv \boldsymbol{F}^{-1} * (\boldsymbol{S} - \boldsymbol{r}) \pmod{p} \quad \text{and} \quad \|\boldsymbol{b}\| \le \frac{p}{2}. \end{aligned}$$

There is thus exactly one value of $\boldsymbol{b}$ in $\mathcal{C}$ satisfying $\phi(\boldsymbol{b}) = \boldsymbol{r}$, namely the unique element of $\mathcal{C}$ that is congruent modulo p to $\boldsymbol{F}^{-1} * (\boldsymbol{S} - \boldsymbol{r})$. This shows that ϕ is bijective, which concludes the proof of Lemma 4. □

Resuming the proof of Proposition 3, we have, for all $(\boldsymbol{s},\boldsymbol{t}) \equiv (\boldsymbol{s}_p,\boldsymbol{t}_p) \pmod{p}$,

$$\mathrm{Prob}_{\boldsymbol{r}\leftarrow\mathcal{R}(A)}\left(\begin{matrix}\text{signature}\\ \text{is } (\boldsymbol{s},\boldsymbol{t})\end{matrix}\;\middle|\;\begin{matrix}\text{private key is } (\boldsymbol{f},\boldsymbol{g}) \text{ and}\\ \text{document hash is } (\boldsymbol{s}_p,\boldsymbol{t}_p)\end{matrix}\right)$$
$$= \frac{\#\Sigma(\boldsymbol{f},\boldsymbol{g},\boldsymbol{s},\boldsymbol{t})}{\#\mathcal{R}(A)} = \frac{\#\mathcal{C}}{\#\mathcal{R}(A)},$$

where the penultimate equality follows from Lemma 4. This completes the proof of Proposition 3. □

We now suppose that Alice has used the private key $(\boldsymbol{f},\boldsymbol{g})$ to create a transcript of valid signatures of the form

$$\big((\boldsymbol{s},\boldsymbol{t}),(\boldsymbol{s}_p,\boldsymbol{t}_p)\big),$$

so according to Proposition 3 and our assumption on hash function, the $(\boldsymbol{s}_p,\boldsymbol{t}_p)$ values are uniformly distributed in $\mathcal{R}(p/2)^2$, and for a given $(\boldsymbol{s}_p,\boldsymbol{t}_p)$, the $(\boldsymbol{s},\boldsymbol{t})$ values are uniformly distributed in the set

$$\left\{(\boldsymbol{s},\boldsymbol{t}) \in L_h\left(\frac{q}{2}-B_s,\frac{q}{2}-B_t\right) : (\boldsymbol{s},\boldsymbol{t}) \equiv (\boldsymbol{s}_p,\boldsymbol{t}_p) \pmod{p}\right\}. \tag{3.10}$$

We now explain how Bob can produce a list of values $\big((\boldsymbol{s},\boldsymbol{t}),(\boldsymbol{s}_p,\boldsymbol{t}_p)\big)$ with the exact same distribution. (The catch, of course, is that Bob will not know what documents he's signing, because for a given triple $(\boldsymbol{h},\boldsymbol{s}_p,\boldsymbol{t}_p)$, he cannot invert the hash function to find a document μ satisfying $\mathsf{Hash}(\boldsymbol{h},\mu) = (\boldsymbol{s}_p,\boldsymbol{t}_p)$.)

Bob first randomly uniformly selects an $\boldsymbol{s} \in \mathcal{R}(q/2-B_s)$. Next he computes

$$\boldsymbol{t} \equiv \boldsymbol{h} * \boldsymbol{s} \pmod{q}.$$

If $\boldsymbol{t} \in \mathcal{R}(q/2-B_t)$, he continues, otherwise he goes back and chooses a different $\boldsymbol{s}$. Finally, he sets

$$\boldsymbol{s}_p = (\boldsymbol{s} \bmod p) \quad\text{and}\quad \boldsymbol{t}_p = (\boldsymbol{t} \bmod p).$$

Then $\big((\boldsymbol{s},\boldsymbol{t}),(\boldsymbol{s}_p,\boldsymbol{t}_p)\big)$ is a valid signature on any digital document μ such that $\mathsf{Hash}(\boldsymbol{h},\mu) = (\boldsymbol{s}_p,\boldsymbol{t}_p)$.

We claim that this procedure produces every $(\boldsymbol{s}_p,\boldsymbol{t}_p)$ value in $\mathcal{R}(p/2)^2$ with equal probability, and that for any such value, it produces every $(\boldsymbol{s},\boldsymbol{t})$ in the set (3.10) with equal probability. There is a subtlety here, because the previous claim is clearly false for some $\boldsymbol{h}$ values. For example, it is false for $\boldsymbol{h} = 1$, since in that case $\boldsymbol{s} = \boldsymbol{t}$, which leads to $\boldsymbol{s}_p = \boldsymbol{t}_p$. The validity of the claim lies in two subsidiary assumptions.

First, we assume that products $\boldsymbol{h} = \boldsymbol{f}^{-1} * \boldsymbol{g} \pmod{q}$ behave like random mod q polynomials as $\boldsymbol{f}$ and $\boldsymbol{g}$ vary over polynomials with small coefficients. This has been much studied experimentally since the original NTRU paper.

Second, we assume that a random mod q polynomial $\boldsymbol{h}$ provides enough mixing so that as $\boldsymbol{s}$ varies over $\mathcal{R}(q/2 - B_s)$, the values of

$$\boldsymbol{s} \bmod p \quad \text{and} \quad (\boldsymbol{h} * \boldsymbol{s} \bmod q) \bmod p$$

behave *independently* and uniformly in $\mathcal{R}(p/2)$. Again this may be verified experimentally, but lack of space precludes the inclusion of a formal proof here.

Hence under reasonable randomness and mixing assumptions, we have shown that Bob can use $\boldsymbol{h}$ to produce a transcript of signature/document pairs that is indistinguishable from the transcript that Alice produced using the private key $(\boldsymbol{f}, \boldsymbol{g})$ and the signing algorithm.

4. Probability of Generating a Valid Signature

To simplify our analysis we let $B = \lceil p^2 N/4 \rceil$ and take

$$B_s = B_t = B.$$

With this assumption there is zero probability of rejecting a candidate signature due to $\|\boldsymbol{a} * \boldsymbol{s}\| > B_s$ or $\|\boldsymbol{a} * \boldsymbol{t}\| > B_t$, but the probability of rejection due to non-inclusion in $\mathcal{R}(q/2-B) \times \mathcal{R}(q/2-B)$ is moderately high. Regardless, we can show that the probability of generating a valid signature is approximately $e^{-8/k}$, which is still practical. Further, the probability of rejection can be made significantly lower by fine-tuning B_s and B_t; our proposed parameters in section 6 reflect this optimization.

For this section we assume that the various parameters satisfy the conditions given in Table 1.

N	a moderate sized prime, say $200 < N < 5000$
p	a small prime chosen so that $N \log_2(p)$ is greater than the desired bit security
B	$\leq \lceil p^2 N/4 \rceil$
k	a small constant, say $2 \leq k \leq 50$
q	an integer coprime with p and satisfying $q \approx kNB \approx kp^2N^2/4$

Table 1. Parameter guidelines

The rejection criterion says that we only accept signatures whose norm is smaller than $q/2 - B$, so we want q to be a lot larger than B, or it will be too hard to find an acceptable signature. We consider the sup norm of a potential signature

$$(\boldsymbol{s}, \boldsymbol{t}) = (\boldsymbol{s}_0, \boldsymbol{t}_0) + (\boldsymbol{a} * \boldsymbol{f}, \boldsymbol{a} * \boldsymbol{g})$$

produced in Step 7 of the signing algorithm. The coefficients of $\boldsymbol{s}_0$ and $\boldsymbol{t}_0$ are in $\mathcal{R}(q/2)$, the coefficents of $\boldsymbol{a} * \boldsymbol{f}$ are in $\mathcal{R}(p^2N/4)$, and the coefficients of $\boldsymbol{a} * \boldsymbol{g}$ are in $\mathcal{R}(pN/2)$. Hence the coefficients of an $(\boldsymbol{s}, \boldsymbol{t})$ pair produced by Step 7 satisfy

$$\big\|(\boldsymbol{s}, \boldsymbol{t})\big\| \le \frac{q}{2} + \frac{p^2N}{4} \approx \frac{q}{2} + B, \tag{4.1}$$

where we recall that $B = \lceil p^2N/4 \rceil$. We will make the simplifying assumption[1] that the coefficients of $\boldsymbol{s}$ and $\boldsymbol{t}$ are equally likely to take on each of the values in the interval (4.1). The rejection criterion says that we only accept signatures whose coefficents are at most $q/2 - B$. Since we need all $2N$ of the coefficients of $(\boldsymbol{s}, \boldsymbol{t})$ to satisfy this condition, we find that

$$\operatorname{Prob}\big((\boldsymbol{s}, \boldsymbol{t}) \text{ is accepted}\big) \approx \left(\frac{q/2 - B}{q/2 + B}\right)^{2N}.$$

Using the chosen value

$$q \approx \frac{kp^2N^2}{4} \approx kNB$$

from Table 1, we find that

$$\begin{aligned}\operatorname{Prob}\big((\boldsymbol{s}, \boldsymbol{t}) \text{ is accepted}\big) &\approx \left(\frac{1 - 2B/q}{1 + 2B/q}\right)^{2N} \\ &\approx \left(\frac{1 - 2/kN}{1 + 2/kN}\right)^{2N} \\ &\approx e^{-8/k},\end{aligned}$$

where for the last equality we use the estimate $(1 + t/n)^n \approx e^t$, valid when t is small and n is large.

5. Lattice Problems Associated to NTRUMLS

In this section we consider the lattice problems underlying signature keys and signature forgery. We note that shortest and closest vector

[1] In actuality, the coefficients of the products $\boldsymbol{a} * \boldsymbol{f}$ and $\boldsymbol{a} * \boldsymbol{g}$ tend to cluster more towards 0, since they are more-or-less hypergeometrically distributed.

problems (SVP and CVP) are analyzed using the L^2-norm, not the L^∞-norm. We write

$$\|\boldsymbol{v}\|_2 = \sqrt{v_1^2 + v_2^2 + \cdots}$$

for the L^2-norm of the vector $\boldsymbol{v} = (v_1, v_2, \ldots)$.

We will use the following elementary lattice result, whose proof we defer to Section A of the appendix.

Proposition 5. *Let $L_1 \subset \mathbb{Z}^r$ and $L_2 \subset \mathbb{Z}^r$ be lattices of rank r, let $\boldsymbol{t}_1, \boldsymbol{t}_2 \in \mathbb{Z}^r$ be arbitrary vectors, and let*

$$M = (L_1 + \boldsymbol{t}_1) \cap (L_2 + \boldsymbol{t}_2)$$

be the intersection of the indicated translations of L_1 and L_2. We make the following assumptions:

(i) $\gcd\bigl(\det(L_1), \det(L_2)\bigr) = 1$.
(ii) *Either $\boldsymbol{t}_1 \notin L_1$ or $\boldsymbol{t}_2 \notin L_2$ (or both), so in particular $M \neq L_1 \cap L_2$.*

Then the following are true:

(a) $\det(L_1 \cap L_2) = \det(L_1) \cdot \det(L_2)$.
(b) $M \neq \emptyset$.
(c) *For every $\boldsymbol{w}_0 \in M$, the map*

$$L_1 \cap L_2 \longrightarrow M, \qquad \boldsymbol{v} \longmapsto \boldsymbol{v} + \boldsymbol{w}_0 \tag{5.1}$$

is a bijection.
(d) *Let $\boldsymbol{w}_0 \in M$, and let $\boldsymbol{w}' \in M$ be a shortest non-zero vector in M. Then $\boldsymbol{w}_0 - \boldsymbol{w}'$ solves the the closest vector problem in $L_1 \cap L_2$ for the vector $\boldsymbol{w}_0$. (This is true for any norm on $\mathbb{Z}^r$, so in particular it is true for both the L^∞ norm and the L^2 norm.)*

We recall two key quantities associated to lattice problems.

Heuristic. The *Gaussian heuristic* says that the likely L^2-size of a solution to SVP or CVP in a "random" lattice L of reasonably large dimension is approximately

$$\gamma(L) = \sqrt{\frac{\dim L}{2\pi e}} \cdot \det(L)^{1/\dim(L)}.$$

In other words, for "most" lattices L and "most" target vectors $\boldsymbol{v}_0$,

$$\min_{\boldsymbol{v} \in L \smallsetminus \boldsymbol{0}} \|\boldsymbol{v}\|_2 \approx \gamma(L) \quad \text{and} \quad \min_{\boldsymbol{v} \in L} \|\boldsymbol{v} - \boldsymbol{v}_0\|_2 \approx \gamma(L).$$

Heuristic. Let $L \subset \mathbb{Z}^n$ be a lattice for which we want to solve either τ-appr-SVP or τ-appr-CVP. In other words, let $\boldsymbol{v}_0 \in \mathbb{Z}^n$, and suppose that we want to find a vector $\boldsymbol{v} \in L$ satisfying either

$$0 < \|\boldsymbol{v}\|_2 \leq \tau \quad \text{or} \quad \|\boldsymbol{v} - \boldsymbol{v}_0\|_2 \leq \tau.$$

We call τ the *target length* of the problem. The *Gaussian defect* of the problem is the ratio

$$\rho(L,\tau) = \frac{\tau}{\gamma(L)}.$$

Let $0 < \delta < 2$. The *δ-LLL heuristic*, which has been confirmed in numerous experiments, says that solving the τ-appr-SVP or τ-appr-CVP problem is (exponentially) hard as a function of $\dim(L)$, provided that the Gaussian defect $\rho(L,\tau)$ is no more than a small multiple of $\dim(L)^{\delta}$.

We consider the problem of forging a signature. The forger needs to find a vector $(\boldsymbol{s},\boldsymbol{t}) \in L_{\boldsymbol{h}}$ satisfying:

$$\text{Congruence Condition :} \qquad (\boldsymbol{s},\boldsymbol{t}) = (\boldsymbol{s}_p,\boldsymbol{t}_p) \pmod{p}. \tag{5.2}$$

$$\text{Norm Condition :} \qquad \|\boldsymbol{s}\| \le \frac{q}{2} - B_s \tag{5.3}$$

$$\|\boldsymbol{t}\| \le \frac{q}{2} - B_t. \tag{5.4}$$

N.B. The norm condition (5.3) is an L^∞-norm condition.

The vectors $\boldsymbol{s}_p,\boldsymbol{t}_p \in R(p/2)$ are given, so the congruence condition (5.2) may be rephrased as saying that the target vector $(\boldsymbol{s},\boldsymbol{t})$ is in the translation of the lattice $p\mathbb{Z}^{2N}$ by the vector $(\boldsymbol{s}_p,\boldsymbol{t}_p)$. Thus the forger is looking for an L^∞-short vector in the intersection

$$(\boldsymbol{s},\boldsymbol{t}) \in L_{\boldsymbol{h}} \cap \bigl(p\mathbb{Z}^{2N} + (\boldsymbol{s}_p,\boldsymbol{t}_p)\bigr).$$

The determinants

$$\det(L_{\boldsymbol{h}}) = q^N \quad \text{and} \quad \det(p\mathbb{Z}^{2N}) = p^{2N}$$

are relatively prime, so we can use Proposition 5(a) to conclude that

$$\det(L_{\boldsymbol{h}} \cap p\mathbb{Z}^{2N}) = p^{2N}q^N.$$

Then Proposition 5(d) tells us that finding a short vector in the intersection $L_{\boldsymbol{h}} \cap \bigl(p\mathbb{Z}^{2N} + (\boldsymbol{s}_p,\boldsymbol{t}_p)\bigr)$ is equivalent to solving an appr-CVP problem in the lattice $L_{\boldsymbol{h}} \cap p\mathbb{Z}^{2N}$. Since the Gaussian heuristic of $L_{\boldsymbol{h}} \cap p\mathbb{Z}^{2N}$ is

$$\gamma(L_{\boldsymbol{h}} \cap p\mathbb{Z}^{2N}) = \sqrt{\frac{N}{\pi e}}(p^{2N}q^N)^{1/2N} = \sqrt{\frac{p^2qN}{\pi e}},$$

it only remains to estimate the target length.

The rejection criterion in the signature algorithm says that a valid signature $(\boldsymbol{s},\boldsymbol{t})$ has sup norm at most $q/2 - \min(B_s,B_t)$. Hence in particular a valid signature satisfies the L^2-norm bound

$$\bigl\|(\boldsymbol{s},\boldsymbol{t})\bigr\|_2 \le \Bigl(\frac{q}{2} - \min(B_s,B_t)\Bigr)\sqrt{2N}, \tag{5.5}$$

but not every vector in $L_{\boldsymbol{h}}$ satisfying the L^2-norm condition (5.5) and the congruence condition (5.2) will be a valid signature. We are going to simplify the life of a potential forger and assume that she only needs to satisfy the L^2-norm condition (5.5), rather than the more stringent L^∞-norm condition (5.3). Furthermore we will assume, again in the forger's favor, that $B_s = B_t = 0$, so that the she need only find a vector in $\mathcal{R}(\frac{q}{2}) \times \mathcal{R}(\frac{q}{2})$. This gives a target length

$$\tau = q\sqrt{N/2}.$$

Hence the Gaussian defect for our appr-CVP problem is

$$\rho = \frac{q\sqrt{N/2}}{\sqrt{p^2qN/2\pi e}},$$

and using the relations in Table 1 between the various parameters, a little bit of algebra yields

$$\rho = N\sqrt{\frac{k\pi e}{8}}.$$

Thus ρ is a small multiple of $\dim(L_{\boldsymbol{h}} \cap p\mathbb{Z}^{2N})$, so the LLL-heuristic says that solving the associated appr-CVP is a hard problem provided that the dimension is chosen appropriately. Of course, in practice one needs to do experiments with current LLL technology to obtain extrapolated estimates for the actual running time when N is moderately large, say in the range from 500 to 5000.

We next briefly consider the problem of finding the private key $(\boldsymbol{f}, \boldsymbol{g})$ from the public key $\boldsymbol{h}$. The attacker knows that $\boldsymbol{f} = p\boldsymbol{F}$, and standard methods allow him to reduce to the problem of finding the shorter vector $(\boldsymbol{F}, \mathbf{g})$. Then, since on average we have

$$\|\boldsymbol{F}\|_2 \approx \sqrt{N} \quad \text{and} \quad \|\boldsymbol{g}\|_2 \approx \frac{1}{2}p\sqrt{N},$$

the corresponding lattice problem needs to be balanced, also a well-known procedure. See for example [3, 5, 10] for details. For all of the proposed parameter sets in Section 6, the parameters have been chosen so that the difficulty of the private key lattice problem is roughly equal to that of the lattice forgery problem, taking into account the heuristic fact that solving unique-SVP tends to be a bit easier in practice than it is in theory.

6. Proposed Parameter Sets and Implementation

We have implemented NTRUMLS and made it available at `https://github.com/NTRUOpenSourceProject/NTRUMLS`. The parameter sets we have implemented are listed in Tables 2 and 3.

The only feature of our implementation not documented above is the use of product form polynomials for $\boldsymbol{f}$ and $\boldsymbol{g}$. Precisely we specify three small integers d_1, d_2, and d_3 and take

$$\begin{aligned}\boldsymbol{f} &= p(\boldsymbol{F}_1 * \boldsymbol{F}_2 + \boldsymbol{F}_3 + 1), \text{and}\\ \boldsymbol{g} &= \boldsymbol{G}_1 * \boldsymbol{G}_2 + \boldsymbol{G}_3 + 1\end{aligned}$$

where the polynomials $\boldsymbol{F}_i$ and $\boldsymbol{G}_i$ have exactly d_i coefficients equal to $+1$ and d_i coefficients equal to -1. The extra constant terms are to ensure that $\boldsymbol{f}(1) \neq 0$ and $\boldsymbol{g}(1) \neq 0$. Product form keys were introduced to NTRUEncrypt in [6].

	Set #1	Set #2	Set #3	Set #4
N	401	439	593	743
p	3	3	3	3
$\log_2 q$	18	19	19	20
B_s	240	264	300	336
B_t	80	88	100	112
d_1, d_2, d_3	8,8,6	9, 8, 5	10, 10, 8	11, 11, 15
Key & signature size (bytes)	853	988	1335	1765
≈ Prob[accept]	38%	55%	41%	53%
≈ bit security	112	128	192	256

Table 2. Sample NTRUMLS Parameters

	Set #1	Set #2	Set #3	Set #4
KeyGen (μs)	2431	2928	5183	7855
Sign (μs)	575	436	1033	1000
Verify (μs)	92	102	179	231

Table 3. Preliminary performance results. Average times, in microseconds, over 10000 iterations. Code was run on an Intel(R) Core(TM) i7-2640M. Further benchmarks will be available at `http://bench.cr.yp.to/` in the near future.

References

[1] Léo Ducas and Phong Q. Nguyen, *Learning a zonotope and more: Cryptanalysis of NTRUSign countermeasures*, Advances in Cryptology ASIACRYPT 2012 (Xiaoyun Wang and Kazue Sako, eds.), Lecture Notes in Computer Science, no. 7658, Springer Berlin Heidelberg, January 2012, pp. 433–450.

[2] Craig Gentry, Chris Peikert, and Vinod Vaikuntanathan, *Trapdoors for hard lattices and new cryptographic constructions*, Proceedings of the 40th annual ACM symposium on Theory of computing (New York, NY, USA), STOC '08, ACM, 2008, p. 197206.

[3] Jeffrey Hoffstein, Jill Pipher, and Joseph H. Silverman, *NTRU: a ring-based public key cryptosystem*, Algorithmic number theory (Portland, OR, 1998), Lecture Notes in Comput. Sci., vol. 1423, Springer, Berlin, 1998, pp. 267–288. MR 1726077

[4] ———, *An Introduction to Mathematical Cryptography*, Undergraduate Texts in Mathematics, Springer, New York, 2008. MR 2433856 (2009m:94051)

[5] Jeffrey Hoffstein and Joseph Silverman, *Optimizations for NTRU*, Public-key cryptography and computational number theory (Warsaw, 2000), de Gruyter, Berlin, 2001, pp. 77–88. MR 1881629 (2003f:94060)

[6] Jeffrey Hoffstein and Joseph H. Silverman, *Random small Hamming weight products with applications to cryptography*, Discrete Applied Mathematics **130** (2003), no. 1, 37–49.

[7] Vadim Lyubashevsky, *Lattice-based identification schemes secure under active attacks*, Public key cryptography—PKC 2008, Lecture Notes in Comput. Sci., vol. 4939, Springer, Berlin, 2008, pp. 162–179. MR 2570228 (2010m:94142)

[8] ———, *Fiat-Shamir with aborts: applications to lattice and factoring-based signatures*, Advances in cryptology—ASIACRYPT 2009, Lecture Notes in Comput. Sci., vol. 5912, Springer, Berlin, 2009, pp. 598–616. MR 2593089

[9] ———, *Lattice signatures without trapdoors*, Advances in cryptology—EUROCRYPT 2012, Lecture Notes in Comput. Sci., vol. 7237, Springer, Heidelberg, 2012, pp. 738–755. MR 2972929

[10] Alexander May and Joseph H. Silverman, *Dimension reduction methods for convolution modular lattices*, Cryptography and lattices (Providence, RI, 2001), Lecture Notes in Comput. Sci., vol. 2146, Springer, Berlin, 2001, pp. 110–125. MR 1903891 (2003d:11097)

[11] Phong Q. Nguyen and Oded Regev, *Learning a parallelepiped: cryptanalysis of GGH and NTRU signatures*, Advances in cryptology—EUROCRYPT 2006, Lecture Notes in Comput. Sci., vol. 4004, Springer, Berlin, 2006, pp. 271–288. MR 2423548 (2009f:94054)

[12] ———, *Learning a parallelepiped: cryptanalysis of GGH and NTRU signatures*, J. Cryptology **22** (2009), no. 2, 139–160. MR 2496387 (2010j:94046)

Appendix A. Short Vectors in Intersections of Translated Lattices

In this appendix we prove Proposition 5, which relates the problem of finding short vectors in intersections of translated lattices to the problem of finding close vectors in the associated intersection of lattices. We applied this result in Section 5 to the intersection of an NTRU lattice $L_{\boldsymbol{h}}$ and the lattice $p\mathbb{Z}^{2N}$.

Proof of Propostion 5. (a) The fact that the determinants multiply is a standard fact from the theory of lattices.
(b) We let $D_i = \det(L_i)$ for $i = 1, 2$. We use the fact that for any lattice $L \subset \mathbb{Z}^r$ of determinant D, we have $D\mathbb{Z}^r \subset L$. The assumption that $\gcd(D_1, D_2) = 1$ means that we can find $(x, y) \in \mathbb{Z}$ such that

$$xD_1 + yD_2 = 1.$$

We let

$$e_1 = yD_2 = 1 - xD_1, \qquad e_2 = xD_1 = 1 - yD_2.$$

We now consider the vector

$$\boldsymbol{t} = e_1\boldsymbol{t}_1 + e_2\boldsymbol{t}_2.$$

Then

$$\boldsymbol{t} - \boldsymbol{t}_1 = (e_1 - 1)\boldsymbol{t}_1 + e_2\boldsymbol{t}_2 = -xD_1\boldsymbol{t}_1 + xD_1\boldsymbol{t}_2 \in D_1\mathbb{Z}^r \subset L_1,$$

and similarly,

$$\boldsymbol{t} - \boldsymbol{t}_2 = e_1\boldsymbol{t}_1 + (e_2 - 1)\boldsymbol{t}_2 = yD_2\boldsymbol{t}_1 - yD_2\boldsymbol{t}_2 \in D_2\mathbb{Z}^r \subset L_2.$$

Hence $\boldsymbol{t}$ is in M, so $M \neq \emptyset$.
(c) In order to prove that (5.1) is a bijection, we will show that

$$\boldsymbol{v} \in L_1 \cap L_2 \quad \Longrightarrow \quad \boldsymbol{v} + \boldsymbol{w}_0 \in M \tag{A.1}$$

and

$$\boldsymbol{w} \in M \quad \Longrightarrow \quad \boldsymbol{w} - \boldsymbol{w}_0 \in L_1 \cap L_2. \tag{A.2}$$

For (A.1), we know that $\boldsymbol{w}_0 \in M$, so by definition of M,

$$\boldsymbol{w}_0 = \boldsymbol{v}_1 + \boldsymbol{t}_1 = \boldsymbol{v}_2 + \boldsymbol{t}_2 \quad \text{with } \boldsymbol{v}_i \in L_1 \text{ and } \boldsymbol{v}_2 \in L_2.$$

Then

$$\boldsymbol{v} + \boldsymbol{w}_0 = \underbrace{(\boldsymbol{v} + \boldsymbol{v}_1)}_{\text{in } L_1} + \boldsymbol{t}_1 = \underbrace{(\boldsymbol{v} + \boldsymbol{v}_2)}_{\text{in } L_2} + \boldsymbol{t}_2,$$

so $\boldsymbol{v} + \boldsymbol{w}_0 \in M$. For (A.2), we write the given $\boldsymbol{w} \in M$ as

$$\boldsymbol{w} = \boldsymbol{v}_1' + \boldsymbol{t}_1 = \boldsymbol{v}_2' + \boldsymbol{t}_2 \quad \text{with } \boldsymbol{v}_i' \in L_1 \text{ and } \boldsymbol{v}_2' \in L_2.$$

Then

$$\boldsymbol{w} - \boldsymbol{w}_0 = \underbrace{\boldsymbol{v}_1' - \boldsymbol{v}_1}_{\text{in } L_1} = \underbrace{\boldsymbol{v}_2' - \boldsymbol{v}_2}_{\text{in } L_2},$$

so $\boldsymbol{w} - \boldsymbol{w}_0 \in L_1 \cap L_2$.
(d) We are given that $\boldsymbol{w}_0, \boldsymbol{w}' \in M$ and that

$$\|\boldsymbol{w}'\|_2 = \min_{\boldsymbol{w} \in M \smallsetminus \boldsymbol{0}} \|\boldsymbol{w}\|_2.$$

To ease notation, we set

$$\boldsymbol{v}' = \boldsymbol{w}_0 - \boldsymbol{w}'.$$

We know from (c) that $\boldsymbol{w}' - \boldsymbol{w}_0 \in L_1 \cap L_2$, and $L_1 \cap L_2$ is a lattice, so $\boldsymbol{v}' \in L_1 \cap L_2$. We estimate

$$\begin{aligned} &\|\boldsymbol{v}' - \boldsymbol{w}_0\|_2 \\ &\qquad = \|\boldsymbol{w}'\|_2 \quad \text{by definition of } \boldsymbol{v}', \\ &\qquad = \min_{\boldsymbol{w} \in M \smallsetminus \boldsymbol{0}} \|\boldsymbol{w}\|_2 \quad \text{by definition of } \boldsymbol{w}', \\ &\qquad = \min_{\boldsymbol{v} \in (L_1 \cap L_2) \smallsetminus \boldsymbol{w}_0} \|-\boldsymbol{v} + \boldsymbol{w}_0\|_2 \quad \text{since (c) says } M = (L_1 \cap L_2) + \boldsymbol{w}_0. \end{aligned}$$

Hence if $\boldsymbol{w}_0 \notin L_1 \cap L_2$, then we have shown that

$$\|\boldsymbol{v}' - \boldsymbol{w}_0\|_2 = \min_{\boldsymbol{v} \in (L_1 \cap L_2)} \|\boldsymbol{v} - \boldsymbol{w}_0\|_2,$$

which is the desired result.

Finally, suppose that $\boldsymbol{w}_0 \in L_1 \cap L_2$. Since also

$$\boldsymbol{w}_0 \in M = (L_1 + \boldsymbol{t}_1) + (L_2 + \boldsymbol{t}_2),$$

we can write

$$\boldsymbol{w}_0 = \boldsymbol{v}_1 + \boldsymbol{t}_1 \quad \text{and} \quad \boldsymbol{w}_0 = \boldsymbol{v}_2 + \boldsymbol{t}_2 \quad \text{with } \boldsymbol{v}_1 \in L_1 \text{ and } \boldsymbol{v}_2 \in L_2.$$

But then $\boldsymbol{t}_1 = \boldsymbol{w}_0 - \boldsymbol{v}_1 \in L_1$ and $\boldsymbol{t}_2 = \boldsymbol{w}_0 - \boldsymbol{v}_2 \in L_2$, contradicting the initial assumption on $\boldsymbol{t}_1$ and $\boldsymbol{t}_2$. Hence $\boldsymbol{w}_0 \notin L_1 \cap L_2$, which completes the proof of Proposition 5. □

Jeff Hoffstein/Jill Pipher/Joseph H. Silverman, Mathematics Department, Box 1917, Brown University, Providence, RI 02912 USA

John M. Schanck/William Whyte, Security Innovation, Wilmington, MA 01887

E-mail address: jhoff@math.brown.edu, jpipher@math.brown.edu, jhs@math.brown.edu, jschanck@securityinnovation.com, wwhyte@securityinnovation.com