# Cryptanalysis of NTRU

Alexander May

Department of Computer Science
University of Frankfurt, Germany

**Abstract.** We present new results on the cryptanalysis of the NTRU Cryptosystem by lattice reduction. The new lattices have smaller dimension than those used in former attacks. In addition, they take advantage of the special structure of NTRU secret keys. A certain class of NTRU keys is especially suitable for these attacks, although the new methods apply to all keys. With these lattices, some instances of NTRU for medium security level can be broken in less than 1 hour. Further, weak keys can be broken for high security levels.

**Keywords:** NTRU, lattice reduction, SVP, polynomial ring.

## 1 Introduction

The NTRU Cryptosystem was first presented by J. Hoffstein, J. Pipher and J.H. Silverman in '96 [3]. It is a ring-based cryptosystem operating in the polynomial ring $\mathbb{Z}_q[X]/(X^n - 1)$ where $n$ is the security parameter. NTRU has achieved considerable attention because of its encryption and decryption speed and the easyness of creating public-key/secret-key pairs, which makes it practical to change keys frequently. There is a licensing agreement with SONY [5]. D. Coppersmith and A. Shamir [1] presented first lattice attacks on the system. In their lattice $L^{cs}$ vectors correspond to factorizations of the public-key in the ring $\mathbb{Z}_q[X]/(X^n - 1)$. They showed that any non-trivial lattice vector at most as long as the original secret key – the target vector – can be used for decryption. Also if one is able to find two vectors not longer than 2.5 times the target vector, then he gets partial information about the plaintext which can be combined to recover the whole message. The lattice has the practical disadvantage that its dimension is $2n$.
The CS attacks have been implemented and analyzed in a recent paper by the NTRU-authors [2]. They did not break the system for the proposed security parameters and made rough estimates of the necessary running time to recover the secret key. It never appeared that smaller or slightly larger keys than the target were discovered. It seems that there is a gap

between useless, longer vectors in the lattice and the target.
To compute short vectors we use lattice reduction algorithms such as the $L^3$-algorithm of Lenstra, Lenstra and Lovasz [4] and the BKZ-algorithm of Schnorr [6, 7]. These algorithms approximate the shortest vector in a lattice $L$ to within some factor. Thus, we are interested in making the gap[1] $c = \frac{\lambda_2(L)}{\lambda_1(L)}$ between the shortest and second-to-shortest vector as big as possible. If we approximate the shortest vector within a factor less than $c$ we get the exact solution. Heuristically, the larger $c$ the higher the probability that the lattice reduction algorithm yields the shortest vector instead of just an approximation. To enlarge $c$ we can shorten the shortest vector or make the second-to-shortest vector longer as it is done in this paper. We show that in the Coppersmith-Shamir lattice $L^{cs}$ the shortest and the target vector are not unique. Then we propose some lattice class where the target is unique if there is a unique longest zero-run in one of the secret polynomials. Next we show that with overwhelming probability this target is also the only vector which has sup-norm 1 for $n$ large enough.
Furthermore, we introduce some lattices which reduce the lattice dimension of the lattice $L^{cs}$ from $2n$ to $(1+\alpha)n,\ 0 < \alpha \leq 1$. With these lattices the time to recover the secret key from the public key drops by about a factor 10 in medium security level.
The paper is organized as follows. In Section 2 we give a short summary of the NTRU Cryptosystem and the lattice of Shamir and Coppersmith. In Section 3 we present the new classes of lattices which make the target vector unique. The target vector is with high probability the shortest vector in sup-norm what is proven in Section 4. Section 5 introduces lattices which reduce the lattice dimension. These methods can be combined with those of Section 3. In Section 6 we give practical results for the security of NTRU in medium security and break some weak instances of NTRU in high security dimension.

## 2 NTRU and the Coppersmith-Shamir Lattice

We denote the ring of integers by $\mathbb{Z}$ and the integers modulo q by $\mathbb{Z}_q$ which are presented by the integers in the interval $(-\frac{q}{2}, +\frac{q}{2})$. The polynomial ring $R_q = \mathbb{Z}_q[X]/(X^n - 1)$ consists of all polynomials with degree less than n and coefficients in $\mathbb{Z}_q$. We identify a polynomial $f \in R_q$ with

[1] The $i^{th}$ successive minimum $\lambda_i(L)$ is the smallest real number $r$ such that there are $i$ linear independent vectors in $L$ of length at most $r$.

its coefficient vector

$$f = (f_0, f_1, \ldots, f_{n-1}) = \sum_{i=0}^{n-1} f_i X^i.$$

Two polynomials $f, g \in R_q$ are multiplied by the ordinary convolution

$$(f * g)_k \equiv \sum_{i+j \equiv k(n)} f_i \cdot g_j \quad (0 \le k < n) \tag{1}$$

which is commutative and associative. The convolution product is presented by the symbol $*$ to distinguish it from multiplication in $\mathbb{Z}$.
The multiplicative group of units in $R_q$ is denoted by $R_q^*$. The inverse polynomial of $f \in R_q^*$ is $f_q^{-1}$.

## 2.1 NTRU Cryptosystem

We briefly recall the NTRU system [2]. The subset $S'_d \subset R_q$ consists of all polynomials with exactly $d+1$ coefficients $+1$, d coefficients $-1$ and the others 0. $S_d$ consists of all polynomials with $d$ coefficients $+1$, d coefficients $-1$ and the others 0. Choosing a polynomial $f \in_R S_d$ means randomly choosing a polynomial in $S_d$. Note that the coefficient vector $f$ has sup-norm 1.

**Key creation:**
Choose random $f \in_R S'_d$ and $g \in_R S_d$. Compute $f_q^{-1} \in R_q^*$ and $f_p^{-1} \in R_p^*$ using the extended Euclidean Algorithm for polynomials. If one of these inverses does not exist choose a new $f$. Otherwise $f$ serves as the secret key. Publish the polynomial

$$h \equiv f_q^{-1} * g \pmod q \tag{2}$$

as public key.
**Encryption:**
Choose $\phi \in_R S_e$ and encrypt the plaintext $m \in R_p$ as

$$e \equiv p\phi * h + m \pmod q$$

**Decryption:** Compute

$$\begin{aligned} a &\equiv e * f \pmod q \\ &\equiv p\phi * f_q^{-1} * g * f + m * f \pmod q \\ &\equiv p\phi * g + m * f \pmod q \end{aligned}$$

For appropriate chosen parameters we can ensure that the polynomial $a$ has small coefficients so that is not only in the ring $R_q$ but also in $\mathbb{Z}[X]/(X^n - 1)$. That is, the reduction modulo $q$ does not affect the coefficients. Thus, we can switch to reduction modulo $p$ computing

$$\begin{aligned} a * f &\equiv p\phi * g + m * f * f_p^{-1} \pmod p \\ &\equiv m \pmod p. \end{aligned}$$

We recover the plaintext $m \in R_p$.

| **Security** | n | q | p | f | g | $\phi$ | Priv.Key (bit) | Pub.Key (bit) |
|---|---|---|---|---|---|---|---|---|
| **medium** | 107 | 64 | 3 | $S'_{14}$ | $S_{12}$ | $S_5$ | 340 | 642 |
| **high** | 167 | 128 | 3 | $S'_{60}$ | $S_{20}$ | $S_{18}$ | 530 | 1169 |
| **highest** | 503 | 256 | 3 | $S'_{215}$ | $S_{72}$ | $S_{55}$ | 1595 | 4024 |

**Table 1.** Suggested parameters for NTRU

The suggested parameters are given in Table 1. The security of NTRU is based on the following complexity assumption.

**Assumption 2.2 (Polynomial factorization problem PFP)** *Given a polynomial $h = f_q^{-1} * g \pmod q$ where the coefficients of $f$ and $g$ are small compared to $q$. For suitable parameter settings it is intractable to recover one of the polynomials $f$ or $g$ with knowledge of $h$ or to find two polynomials $f'$ and $g'$ with small coefficients such that $f' * h \equiv g' \pmod q$.*

Note that for each $f \in R_q^*$ there's a factorization of the form $h \equiv f_q^{-1} * g$. Thus, there are $|R_q^*|$ possible factorizations of which only those with small $\ell_2$-norm are useful for decryption.

### 2.3 The lattice $L^{cs}$

Let $v_1, \ldots, v_m \in \mathbb{R}^n$ be linear independent. The lattice $L$ spanned by $\{v_1, \ldots, v_m\}$ consists of all integer linear combinations $L = \sum_{i=1}^m \mathbb{Z} v_i$. Its lattice dimension, denoted by $\dim(L)$ is the dimension of the $\mathbb{R}$-subspace it spans. Each lattice $L$ of dimension $n$ has a basis, i.e., a sequence $[b_1, \ldots, b_n]$ of $n$ elements of $L$ generating $L$.
Coppersmith and Shamir proposed the following lattice which is spanned

by the row vectors of the $(2n \times 2n)$-matrix $L^{cs}$:

$$\mathbf{L}^{\mathbf{cs}} = \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \\ \hline b_{n+1} \\ b_{n+2} \\ \vdots \\ b_{2n} \end{pmatrix} = \left(\begin{array}{cccc|cccc} 1 & 0 & \dots & 0 & h_0 & h_1 & \dots & h_{n-1} \\ 0 & 1 & \dots & 0 & h_1 & h_2 & \dots & h_0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 1 & h_{n-1} & h_0 & \dots & h_{n-2} \\ \hline 0 & 0 & \dots & 0 & q & 0 & \dots & 0 \\ 0 & 0 & \dots & 0 & 0 & q & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 0 & 0 & 0 & \dots & q \end{array}\right)$$

**Definition 2.4** *Let $f = (f_0, f_1, f_2, \dots, f_{n-1}) \in R_q$. The automorphism $\sigma : R_q \to R_q$ with*

$$f(X) \mapsto f(X^{-1}) = (f_0, f_{n-1}, f_{n-2}, \dots, f_1)$$

*is defined as* coefficient-mirror *on $R_q$.*

**Lemma 2.5** *Let $h \equiv f_q^{-1} * g \pmod q$ be any factorization of $h$ in $R_q$ with $f_q^{-1} \in R_q^*$. Then the lattice $L^{cs}$ contains the vector $(\sigma(f), g)$.*

**Proof:** Given a factorization $h * f \equiv g \pmod q$ we obtain for each of the $n$ coefficients of the polynomial $g$ an equation of the form

$$g_k \equiv \sum_{i+j \equiv k(n)} h_i \cdot f_j \pmod q \quad \text{for } 0 \le k < n \ .$$

Writing these equations in matrix-vector representation leads to

$$\begin{pmatrix} h_0 & h_1 & \dots & h_{n-1} \\ h_1 & h_2 & \dots & h_0 \\ \vdots & \vdots & \ddots & \vdots \\ h_{n-1} & h_0 & \dots & h_{n-2} \end{pmatrix} \cdot \begin{pmatrix} f_0 \\ f_{n-1} \\ f_{n-2} \\ \vdots \\ f_1 \end{pmatrix} \equiv \begin{pmatrix} g_0 \\ g_1 \\ g_2 \\ \vdots \\ g_{n-1} \end{pmatrix} \quad \text{over } \mathbb{Z}_q$$

This explains the structure of $L^{cs}$. Let $v = \sum c_i b_i$ be any lattice vector. Then $v$ has the form $(c_1, \dots, c_n, v_{n+1}, \dots, v_{2n})$. Thus we obtain in the first $n$ components of a lattice vector the coefficient vector $\sigma(f)$. The addition of row vectors defined by $\sigma(f)$ forces a linear combination on the h vectors which leads according to (2) to the vector $g$ in the second $n$ coefficients. The reduction modulo $q$ is component-wise performed by the $q$-lattice vectors $b_{n+1}, \dots, b_{2n}$. $\square$

The identity matrix can be scaled by a factor of $\lambda = \frac{\|g\|_2}{\|f\|_2}$ to balance the norms of $\sigma(f)$ and $g$ in the target vector. Since there is a computational advantage in using integral lattices we always scaled columns $n+1$ to $2n$ by the factor $\lceil \frac{\|f\|_2}{\|g\|_2} \rfloor$.

# 3 New classes of lattices

## 3.1 The cyclic structure of $L^{cs}$

We point out a cyclic property of $L^{cs}$ which is used to construct new lattices by fixing the target vector to a specially shifted lattice vector.

**Definition 3.2** *Let $f = (f_0, f_1, f_2, \ldots, f_{n-1}) \in R_q$. Then we define*

$$f^{ls(l)} = (f_l, f_{l+1}, f_{l+2}, \ldots, f_{l-1}) \text{ as leftshift of } f \text{ by } l$$

Note that $f^{ls(l)} \equiv x^l \cdot f$ in $R_q$.

**Definition 3.3** *Let $f = (f_0, f_1, f_2, \ldots, f_{n-1}) \in R_q$. Then $f$ is called* periodic with period $t$ *if $f = f^{ls(t)}$ and $t$ divides $n$. If $f$ has smallest period $n$ then $f$ is called* non-periodic.

**Lemma 3.4** *Let $h \equiv f_q^{-1} * g \pmod q$ be the secret factorization of $h$ in $R_q$ with $f_q^{-1} \in S'_{d_f}$ and $g \in S_{d_g}$. Then*

1. *$L^{cs}$ contains the shifted vectors $(\sigma(f)^{ls(l)}, g^{ls(l)})$ for $0 \leq l < n$.*
2. *Let $\|(\sigma(f'), g')\|_p = \lambda_1(L^{cs})$ be the smallest factorization for a given $\ell_p$-norm. If $n > (2d_f + 1) + 2d_g$, $n$ prime, then $(\sigma(f')^{ls(l)}, (g')^{ls(l)})$, $0 \leq l < n$, are pairwise distinct.*

**Proof:** *1.* Observe that multiplying the equation $h * f \equiv q$ by $x^l$ leads to $h * f^{ls(l)} \equiv g^{ls(l)}$ for $0 \leq l < n$. Analogous to the proof of Lemma 2.5 we obtain, that $L^{cs}$ contains the vector $(\sigma(f)^{ls(l)}, g^{ls(l)})$.
*2.* According to *1.* we have that for the smallest factorization $h * f' \equiv g'$ the lattice $L^{cs}$ contains the $n$ vectors $(\sigma(f')^{ls(l)}, (g')^{ls(l)})$ for $0 \leq l < n$ which have identical norm. It remains to show that these $n$ shifted vectors are pairwise distinct. Since $n$ is chosen to be prime, there can be no non-trivial periods in $f', g'$. Assume that $f'$ or $g'$ is the vector $(b, b, \ldots, b)$, $(1 \leq b \leq q-1)$. Then for $p < \infty$ :

$$\|(\sigma(f'), g')\|_p \geq b \cdot \sqrt[p]{n} > \sqrt[p]{(2d_f + 1) + 2d_g}.$$

Thus, $(\sigma(f'), g')$ cannot be the shortest vector in $L^{cs}$ since we know by construction of $h$ that the target vector (which corresponds to the secret factorization) has $\ell_p$-norm $\sqrt[p]{(2d_f+1)+2d_g}$. In $\ell_\infty$-norm the target has norm 1 and thus is a non-periodic shortest vector in the lattice. So the $n$ cyclic shifted vectors are pairwise distinct. □

It is tempting to state $\lambda_1(L^{cs}) = \lambda_2(L^{cs}) = \cdots = \lambda_n(L^{cs})$ but in general the $n$ cyclic shifted vectors may not be linear independent. The number of linear independent vectors equals the rank of the Toeplitz matrix that they form. Coppersmith, Shamir [1] showed that any vector in $L$ at most as long as the vector corresponding to the factorization of $h$ into the secret components $f, g$ can be equally used in NTRU for decryption. Together with the above Lemma 3.4 this leads to the following Corollary.

**Corollary 3.5** *Let $T$ be the Toeplitz matrix formed by the cyclic shifted vectors $(\sigma(f)^{ls(l)}, g^{ls(l)})$, $(0 \le l < n)$ and let $rank(T)$ be the rank of $T$. Then breaking NTRU can be reduced to computing one of the $n$ vectors with the length of the first $rank(T)$ successive minima $\lambda_1(L^{cs}) = \cdots = \lambda_{rank(T)}(L^{cs})$.*

I.e., for a cryptanalyst it suffices to compute one of the $n$ shortest vectors of $L^{cs}$ by lattice reduction to break NTRU. The shortest vector is not unique in $L^{cs}$ as was shown in Lemma 3.4. Our aim is to construct a lattice $L$ for which there is a unique vector $v$ with $\|v\|_2 = \lambda_1(L)$ which enlarges the fraction $\frac{\lambda_2(L)}{\lambda_1(L)}$ in the Euclidean norm since all practical lattice reduction methods work in $\ell_2$-norm. This makes the NTRU-system more vulnerable for this kind of attack.

### 3.6 The Run-lattices

The class of run-lattices is derived from the lattice $L^{cs}$ by multiplying the columns $n+1$ till $n+r$ by a suitable chosen constant $\theta$.

$$\mathbf{L}_{\mathrm{g}}^{\mathrm{r}}(\theta) = \left(\begin{array}{cccc|ccccc}
1 & 0 & \dots & 0 & \theta\cdot h_0 & \dots & \theta\cdot h_{r-1} & h_r & \dots & h_{n-1} \\
0 & 1 & \dots & 0 & \theta\cdot h_1 & \dots & \theta\cdot h_r & h_{r+1} \dots & h_0 \\
\vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots & & \\
0 & 0 & \dots & 1 & \theta\cdot h_{n-1} & \dots & \theta\cdot h_{r-2} & h_{r-1} \dots & h_{n-2} \\
\hline
0 & 0 & \dots & 0 & \theta\cdot q & \dots & 0 & 0 \dots & 0 \\
0 & 0 & \dots & 0 & 0 & \dots & 0 & 0 \dots & 0 \\
\vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \vdots \ddots & \vdots \\
0 & 0 & \dots & 0 & 0 & \dots & 0 & 0 \dots & q
\end{array}\right)$$

For $\theta = q+1$ we ensure that the smallest vector $v = (v_1, \ldots, v_{2n})$ in the lattice has zero-entries for $v_{n+1}, \ldots, v_{n+r}$. Otherwise it has $\ell_2$-norm at least $\sqrt{\theta}$, but the $q$-vectors $b^r_{n+r+1}, \ldots, b^r_{2n}$ all have norm $\sqrt{q}$. Thus, $v$ cannot be the smallest vector in $L^r_g(\theta)$.
We must determine the size of $r$ for attacking the public key.

**Definition 3.7** *Let $g \in S'_d$. Then $(g_i, g_{i+1}, \ldots, g_j)$ with $g_i, g_{i+1}, \ldots, g_j = 0$ is called a* zero-run in $g$ of length $j-i+1$. *The indices of $g$ are regarded modulo $n$. The longest zero-run in $g$ has length $r$ if*

$$r = \max_{i,j \bmod n}\{j-i+1 \mid g_i, g_{i+1}, \ldots, g_j = 0\}.$$

*We call a zero-run in $g$* unique *if there is only one longest zero-run in $g$.*

Now we are able to formulate the following lemma.

**Lemma 3.8** *Let the coefficient vector $g$ contain a unique, longest zero-run of length $r$ which starts at $g_l$. Then $L^r_g(\theta)$ contains the uniquely determined target vector*

$$\tau^l = ((f_0, f_{n-1}, f_{n-2}, \ldots, f_1)^{ls(l)} |\ (g_0, g_1, \ldots, g_{n-1})^{ls(l)})$$

*with $\ell_2$-norm $\sqrt{\|f\|^2 + \|g\|^2}$. All other shifted vectors $\tau^{ls(m)}$ $(m \neq l)$ have norm at least $\sqrt{\|f\|^2 + \|g\|^2 - 1 + \theta^2}$.*

**Proof:** If the secret $g$ contains a unique longest zero-run of length $r$ then $L^r_g(\theta)$ contains according to Lemma 3.4 the shifted target $(\sigma(f)^{rs(l)}, g^{ls(l)})$, where $g^{ls(l)}$ starts with this zero-run. The multiplier $\theta$ does not affect $\tau^l$'s norm.
Since the longest zero-run is unique all other shifted vectors $\tau^{ls(m)}$ have at least one non-zero entry in the coefficients $n+1, \ldots, n+r$. This entry is scaled by the factor $\theta$. $\square$

Choosing $\theta$ bigger than $q$ forces nearly all of the lattice vector coefficients between the $n+1$ coefficient and the $n+r$ coefficient to be 0, even after LLL reduction. This shortens the search space for small vectors. The larger one can choose $r$ the faster this method works i.e. secret keys $g$ with long zero-runs are especially suitable to this kind of attack.

*Remark 1: Note that we can equally fix zero-runs in $f$ by multiplying rows of the $f$-part of lattice $L^{cs}$ obtaining a lattice $L^r_f(\theta)$. If we want to combine the two lattices $L^r_g(\theta)$ and $L^r_f(\theta)$ we have to guess the correct position of a zero-run in $f$ because we fix the target $\tau^l$ according to Lemma 3.8.*

To get an idea of the fraction of $g$'s having a long zero-run we bound the probabililty of a run of length $r$ when choosing $g$ randomly and uniform according to the probability distribution $S_d$. Remember that $g$ contains $d$ $(+1)$'s, $d$ $(-1)$-coefficients and the others 0.

**Lemma 3.9** *Let* $\Pr(r) = \Pr_{g \in_R S_d}[g$ *contains one run with length* $r]$. *Then we have*

$$\Pr(r) \geq n \cdot \frac{\sum_{i=0}^{2} \binom{n-(r+2)}{d-i}\binom{n-(r+2)-(d-i)}{d-(2-i)} - \sum_{i=0}^{2} \binom{n-2(r+2)}{d-i}\binom{n-2(r+2)-(d-i)}{d-(2-i)}}{\binom{n}{d}\binom{n-d}{d}}$$

**Proof:** The key space of $g$ is $|S_d| = \binom{n}{d}\binom{n-d}{d}$. There are $n$ possibilities for the beginning of a zero-run. The run of length $r$ is either surrounded by two $-1$'s, a $+1$ and $-1$ or two $+1$'s. In the first case the number of keys is $\binom{n-(r+2)}{d}\binom{n-(r+2)-d}{d-2}$ because we have to distribute $d$ $+1$'s on $(n-r-2)$ coefficients and the remaining $(d-2)$ $(-1)$-coefficients on the $(n-r-2-d)$ left places. The other cases are analogous. Note that some $g$'s are counted twice. Namely those, which have a zero-run in the remaining $(n-r-2)$ coefficients. These keys are eliminated by the second summation. □

Thus, we obtain

$$\Pr_{g \in_R S_d}[g \text{ contains a run with length } \geq r] \geq \sum_{j=r}^{n-2d-2} \Pr(j) \qquad (3)$$

We do not know the length $r$ of a longest zero-run in $g$ a priori but as equation (3) suggests the interval of relevant values is small. We may also use a lower bound for $r$.

## 4 The shortest vector in $L_g^r(\theta)$ in sup-norm

In order to enlarge the fraction $\frac{\lambda_2(L_g^r(\theta))}{\lambda_1(L_g^r(\theta))}$ in sup-norm we have to prove that the target vector $\tau^l$ from Lemma 3.8 is the only one with sup-norm 1 assuming that the coefficients of the public-key $h$ are distributed according to the following assumption:

**Assumption 4.1** *Let* $h = (h_0, h_1, \ldots, h_{n-1}) = f_q^{-1} * g$ *with* $f \in_R S_d'^*$, $g \in_R S_d^*$. *Then the* $h_i$ *are distributed independent and uniformly over* $\mathbb{Z}_q$.

For $f \in_R R_q^*$ and $\frac{|R_q^*|}{|R_q|} \approx 1$ we have

$$f \in_R R_q^* \Rightarrow f_q^{-1} \in_R R_q^* \Rightarrow f_q^{-1} * g \equiv h \in_R R_q^*$$

The multiplication with $f_q^{-1}$ is a bijection $R_q^* \times R_q^* \to R_q^*$. Thus the $h$'s are uniformly distributed in $R_q^*$. Because $\frac{|R_q^*|}{|R_q|} \approx 1$ the $h_i$'s are approximately uniformly distributed over $\mathbb{Z}_q$.
We assume that the key $h$ behaves similar if $f$ is chosen from the smaller set $S_d'^*$. With this assumption we are now able to state the following theorem.

**Theorem 4.2** *Under assumption 4.1 we have for $q > 9$*

$$\lim_{n\to\infty} \Pr_h[\exists f', g' \,|\, f' * h \equiv g' \pmod q), \|f'\|_\infty, \|g'\|_\infty = 1, (\sigma(f'), g') \not\equiv \tau^l] = 0.$$

**Proof:** Let $r$ be the length of the longest zero-run in $g$. Consider the lattice $L_g^r(2)$. For a vector $v \in L_g^r(2)$ having $\ell_\infty$-norm 1, the linear combination of columns $n+1, \dots, n+r$ must be zero. Let the longest zero-run of $g$ start with $g_i$. Then we have:

$$\begin{pmatrix} h_0 & h_1 \dots & h_{n-1} \\ h_1 & h_2 \dots & h_0 \\ \vdots & \vdots \ddots & \vdots \\ h_{n-1} & h_0 \dots & h_{n-2} \end{pmatrix} \cdot \begin{pmatrix} f_i \\ f_{i-1} \\ f_{i-2} \\ \vdots \\ f_{i+1} \end{pmatrix} \equiv \begin{pmatrix} 0 \\ \vdots \\ 0 \\ g_{i+r} \\ \vdots \\ g_{i-1} \end{pmatrix} \quad \text{over } \mathbb{Z}_q$$

According to Assumption 4.1 we have $h_j \in_R \mathbb{Z}_q$. We look at the probability that equation (2) has other solutions $f', g'$ with sup-norm 1.

$$\Pr_h\left[\exists f', g' \left| \begin{array}{l} f' * h \equiv g', \\ \|f'\|_\infty, \|g'\|_\infty = 1, \\ (\sigma(f'), g') \not\equiv \tau^l \end{array}\right.\right] = \Pr_h\left[\exists f' \left| \begin{array}{l} f' * h \equiv g', \\ \|f'\|_\infty = 1, \\ f' \neq f^{ls(l)}, g' \text{ fixed} \end{array}\right.\right]$$
$$\cdot |\{g \in \{\pm 1, 0\}^n \,|\, g \text{ starts with r zeros}\}|$$

According to Assumption 4.1 we have $h_j \in_R \mathbb{Z}_q$. Thus

$$\Pr_h\left[\exists f', g' \left| \begin{array}{l} f' * h \equiv g', \\ \|f'\|_\infty, \|g'\|_\infty = 1, \\ (\sigma(f'), g') \not\equiv \tau^l \end{array}\right.\right] \leq \frac{3^n}{q^n} \cdot 3^{n-r} = \frac{3^{2n-r}}{q^n}$$

The claim follows. $\square$

So for sufficiently large $n$ the target is the only lattice vector $v$ with $\|v\|_\infty = 1$. It is open if there is a similar result for the $\ell_2$-norm.

## 5 Dimension-reducing lattices

### 5.1 The lattice $L_\alpha^{gred}$

Experimental expectations suggest that there are no other vectors in $L^{cs}$ with Euclidean length comparable to that of the target – except the $n$ shifted versions. Thus the method of CS [1] to construct spurious keys is not applicable in practice. On the other hand, if there are only a few small lattice vectors corresponding to small factorizations it suffices that the $n$ equations (2) are not fulfilled for all coefficients $g_k$ $(0 \le k < n)$ but for most of them. If there are no other vectors with small entries in these coefficients we will nevertheless recover the target. This idea gives rise to the following lattice $L_\alpha^{gred}$.

$$\mathbf{L}_\alpha^{\mathbf{gred}} = \left(\begin{array}{cccc|cccc}
\lambda & 0 & \dots & 0 & h_0 & h_1 & \dots & h_{\lceil \alpha n \rceil - 1} \\
0 & \lambda & \dots & 0 & h_1 & h_2 & \dots & h_{\lceil \alpha n \rceil} \\
\vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\
0 & 0 & \dots & \lambda & h_{n-1} & h_0 & \dots & h_{\lceil \alpha n \rceil - 2} \\
\hline
0 & 0 & \dots & 0 & q & 0 & \dots & 0 \\
0 & 0 & \dots & 0 & 0 & q & \dots & 0 \\
\vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\
0 & 0 & \dots & 0 & 0 & 0 & \dots & q
\end{array}\right)$$

The practical advantage is that the lattice-dimension drops from $2n$ to $n(1+\alpha)$, $0 < \alpha \le 1$.

### 5.2 The lattice $L_\beta^{fred}$

Analogous to the method of Section 5.1 we can consider only a fraction of the $f$-coefficients. This leads to the lattice $L_\beta^{fred}$ generated by the row vectors of the following matrix.

$$
\mathbf{L}_\beta^{\mathbf{fred}} = \left(\begin{array}{cccc|ccc}
\lambda & 0 & \dots & 0 & h_0 & \dots\dots & h_{n-1} \\
0 & \lambda & \dots & 0 & h_1 & \dots\dots & h_0 \\
\vdots & \vdots & \ddots & \vdots & \vdots & \dots\dots & \vdots \\
0 & 0 & \dots & \lambda & h_{\lceil\beta n\rceil-1} & \dots\dots & h_{\lceil\beta n\rceil-2} \\
0 & 0 & \dots & 0 & h_{\lceil\beta n\rceil} & \dots\dots & h_{\lceil\beta n\rceil-1} \\
\vdots & \vdots & & \vdots & \vdots & & \vdots \\
0 & 0 & \dots & 0 & h_{n-1} & \dots\dots & h_{n-2} \\
\hline
0 & 0 & \dots & 0 & q & 0\ \dots & 0 \\
0 & 0 & \dots & 0 & 0 & q\ \dots & 0 \\
\vdots & \vdots & \ddots & \vdots & \vdots & \vdots\ \ddots & \vdots \\
0 & 0 & \dots & 0 & 0 & 0\ \dots & q
\end{array}\right)
$$

Although $L_\beta^{fred}$ is generated by $2n$ vectors the lattice dimension is below $2n$ because the lattice vectors are linear dependent. The $L^3$-algorithm discovers these linear dependencies and discards the arising zero-vectors. Therefore, the lattice dimension drops to $n(1+\beta)$. Because the target contains the vector $g$ we are able to recover all coefficients of $f$ by solving equation (2).

## 6 Experimental results

We used the lattice-reduction algorithms LLL [4] and BKZ [6–9] with reduction parameter $\delta = 0.95$. The parameter $\gamma$ for deep insertions was chosen to be 0.95, too. We tested several pruning parameters $p$ - which cut the search trees for shortest vectors in the sublattices of BKZ - for optimality. For $n > 95$ the best choice is $p \in [6, 8]$.

The following results were obtained on a HP-Unix workstation model 9000 with 200 Mhz. The NTRU parameters were chosen according to Table 1 for medium security level. The first column represents the time it took to find the secret factorization of $h$ when using the Coppersmith-Shamir lattice $L^{cs}$. The second column gives running times for the dimension-reducing lattices presented in Section 5 with several different parameters. Note that if $\alpha$ and $\beta$ are both smaller than 1 the remaining coefficients have to be guessed - for instance by the meet-in-the-middle algorithm of Odlyzko [10]. The third column shows the running times when using the lattice $L_g^r(q)$ of Section 3.6. We have chosen several keys until the zero-run in $g$ was at least $\frac{21}{107}n$. The probability of having such a run can be computed by equation (3). It is at least 5% for $n = 107$. The last column shows the running time when fixing zero-runs in both keys $g$ and $f$ (see Remark 1 in Section 3.6). The sum of those runs was chosen to be at least $1.8 \cdot \frac{21}{107}n$. The last two columns also combine the methods of Section 5 and 3.6 in using $\alpha = 0.7, \beta = 1.0$. The running time is given in hh:mm:ss.

| | $\mathbf{L^{cs}}$ | $\mathbf{L_\alpha^{gred}}$ & $\mathbf{L_\beta^{fred}}$ | | | $\mathbf{L_g^r(q)}$&$\mathbf{L_{0.7}^{gred}}$ | | $\mathbf{L_{g+f}^r(q)}$ & $\mathbf{L_{0.7}^{gred}}$ | | |
|---|---|---|---|---|---|---|---|---|---|
| N | Time | Time | $\alpha$ | $\beta$ | Time | run | Time | grun | frun |
| 75 | 000:15:12 | 00:06:47 | 0.5 | 1.0 | 00:02:56 | 17 | 00:02:39 | 14 | 16 |
| 80 | 000:24:30 | 00:18:23 | 0.8 | 0.8 | 00:06:39 | 20 | 00:04:41 | 14 | 15 |
| 85 | 000:47:26 | 00:32:36 | 0.5 | 1.0 | 00:06:32 | 17 | 00:06:16 | 17 | 16 |
| 90 | 001:05:10 | 00:37:04 | 0.5 | 1.0 | 00:07:44 | 18 | 00:07:40 | 17 | 17 |
| 92 | 003:02:42 | 01:16:03 | 0.8 | 1.0 | 00:10:08 | 22 | 00:08:37 | 33 | 11 |
| 94 | 003:05:11 | 02:40:36 | 0.8 | 1.0 | 00:13:39 | 22 | 00:13:12 | 24 | 10 |
| 96 | 010:17:12 | 04:01:36 | 0.5 | 1.0 | 00:19:27 | 21 | 00:15:49 | 27 | 12 |
| 98 | 006:41:19 | 03:55:53 | 0.8 | 1.0 | 00:25:07 | 23 | 00:21:36 | 15 | 10 |
| 100 | 024:58:12 | 02:15:14 | 0.77 | 1.0 | 00:28:50 | 20 | 00:23:46 | 16 | 20 |
| 102 | 070:19:49 | 09:16:14 | 0.8 | 1.0 | 00:42:40 | 25 | 00:17:31 | 23 | 18 |
| 104 | 129:37:37 | - | | | 00:24:21 | 24 | 00:28:20 | 26 | 12 |
| 107 | 663:08:52 | 66:40:25 | 0.8 | 0.8 | 01:28:53 | 21 | 00:40:51 | 12 | 27 |

**Table 2.** Timing comparison of the different lattices for medium security

For a graphical representation see Figure 1. We also broke some weak keys $h$ in high security level with long zero-runs in both $f$ and $g$ with the combined lattice methods $L_g^r(500), L_f^r(500)$ & $L_{0.7}^{gred}$. For these reductions we used another implementation of the LLL-algorithm computing the Gram-Schmidt coefficients by householder transformation which is more stable.

| run($f$)+run($g$) | 173 | 155 | 135 | 116 | 97 | 82 |
|---|---|---|---|---|---|---|
| time | 02:00:06 | 01:58:51 | 02:56:22 | 03:18:28 | 05:24:16 | 15:35:44 |

**Table 3.** Running times in high security level $n = 167$

## 7 Discussion and Open Problem

The NTRU-Cryptosystem needs small norms for the secret polynomials to ensure correct decryption. Thus, the problem of zero-runs is inherent.

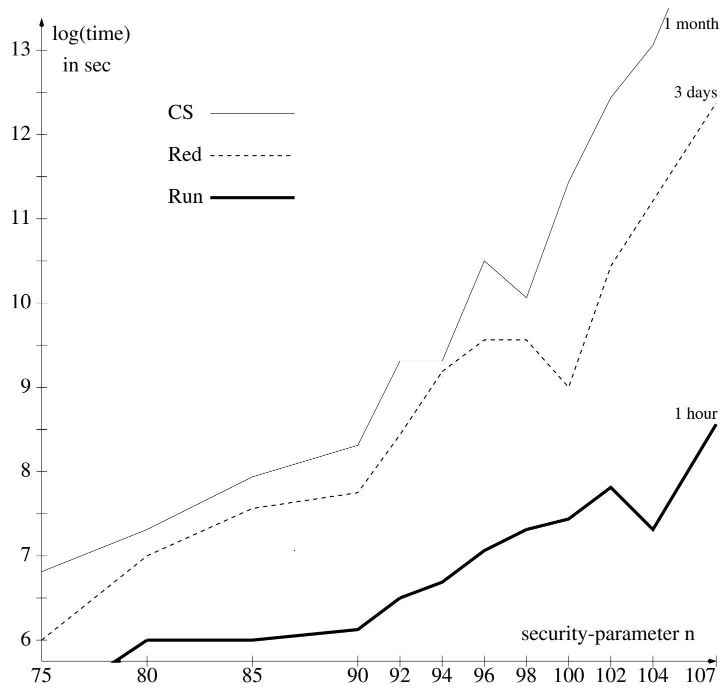


**Fig. 1.** Timing comparison of $L^{cs}$, $L_{\alpha}^{gred}$ & $L_{\beta}^{fred}$ and $L_g^r(q)$ & $L_{0.7}^{gred}$

A counter-measure guarding against the run lattices is to reject weak keys with long zero-runs. This cuts down the key space a little. Thus, one has to increase the security parameter $n$. For the case of message attacks by lattice reduction the situation is even better, because the randomization polynomial $\phi$ is suggested to have a few non-zero coefficients less than $g$. Therefore, we have longer zero-runs in $\phi$ making the described methods also vulnerable for lattice attacks on a single message $m$.
It is still open if there is a result analogous to Theorem 4.2 for the $\ell_2$-norm. If this is the case: What is the fraction $\frac{\lambda_2(L_g^r(\theta))}{\lambda_1(L_g^r(\theta))}$ in the Euclidean norm? This fraction is interesting for optimatization of the parameters $\alpha$ and $\beta$.

## 8 Acknowledgement

The author wants to thank Henrik Koy for providing him the stable $L^3$ with householder transformation making attacks on NTRU in high security level possible.

## References


1. D. Coppersmith, A. Shamir, "Lattice Attacks on NTRU", Eurocrypt '97, Springer LNCS 1233
2. J. Hoffstein, J. Pipher, J.H. Silverman, "NTRU: A new high speed public key cryptosystem", Algorithmic Number Theory (ANTS III), Portland, OR, June 1998, LNCS 1423, J.P. Buhler (ed.), Springer-Verlag, Berlin, 1998, 267-288
3. J. Hoffstein, J. Pipher and J.H. Silverman, "NTRU: A new high speed public key cryptosystem", manuscript, Rump Session Crypto '96, August 1996
4. A.K. Lenstra, H.W. Lenstra and L. Lovasz, "Factoring Polynomials with Integer Coefficients", Mathematische Annalen 261 (1982), 513-534
5. Press Releases, available under "http://www.ntru.com/newscenter.htm"
6. C.P. Schnorr, "A hierarchy of polynomial time lattice basis reduction algorithms", Theoretical Computer Science 53 (1987), 201-224
7. C.P. Schnorr, "Block reduced lattice basis and successive minima", Combinatorics, Probability and Computing 3 (1994), 507-522
8. C.P. Schnorr, M. Euchner, "Lattice basis reduction: Improved practical algorithms and solving subset sum problems", Mathematical Programming 66 (1994), 181-199
9. C.P. Schnorr, H.H. Hoerner, "Attacking the Chor Rivest cryptosystem by improved lattice reduction", Proc. Eurocrypt 1995, Lecture Notes in Computer Science 921, Springer-Verlag, 1995, 1-12
10. J.H. Silverman, A. Odlyzko, "A Meet-In-The-Middle Attack on a NTRU private key", NTRU Cryptosystem Technical Report No.4 (1997), available under "http://www.ntru.com/documentcenter.htm"