ModFalcon: compact signatures based on module NTRU lattices

Chitchanok Chuengsatiansup[1], Thomas Prest[2], Damien Stehlé[3], Alexandre Wallet[4], and Keita Xagawa[4]

[1] University of Adelaïde, Australia chitchanok.chuengsatiansup@adelaide.edu.au
[2] PQ Shield, Oxford, UK thomas.prest@pqshield.com
[3] Univ. Lyon, EnsL, UCBL, CNRS, Inria, LIP, F-69342 Lyon Cedex 07, France
damien.stehle@ens-lyon.fr
[4] NTT Secure Platform Laboratories, Tokyo, Japan
[alexandre.wallet.th,keita.xagawa.zv]@hco.ntt.co.jp

Abstract. Lattices lead to promising practical post-quantum digital signatures, combining asymptotic efficiency with strong theoretical security guarantees. However, tuning their parameters into practical instantiations is a delicate task. On the one hand, NIST round 2 candidates based on Lyubashevsky's design (such as DILITHIUM and qTESLA) allow several tradeoffs between security and efficiency, but at the expense of a large bandwidth consumption. On the other hand, the hash-and-sign Falcon signature is much more compact and is still very efficient, but it allows only two security levels, with large compactness and security gaps between them. We introduce a new family of signature schemes based on the Falcon design, which relies on module lattices. Our concrete instantiation enjoys the compactness and efficiency of Falcon, and allows an intermediate security level. It leads to the most compact lattice-based signature achieving a quantum security above 128 bits.

1 Introduction

Many candidates to the NIST call for post-quantum standardization rely on Euclidean lattices. Indeed, lattice problems seem to be quantum-resistant, and at the same time sufficiently malleable to lead to the construction of cryptographic primitives, ranging from basic to advanced (such as homomorphic encryption). Moreover, relying on structured lattices originating from algebraic number theory has led to very efficient schemes, as showcased by the performance of the candidates still running in the second round of the NIST call: LAC [LLJ+19], KYBER [SAB+19], NewHope [PAA+19], NTRU [ZCH+19], NTRU Prime [BCLv19], Round5 [GZB+19], SABER [DKRV19], and Three Bears [Ham19] for public-key encryption and DILITHIUM [LDK+19], Falcon [PFH+19], and qTESLA [BAA+19] for signatures.

For signatures schemes, a well-known approach is Gentry, Peikert and Vaikuntanathan's *hash-and-sign* paradigm upon *collision-resistant preimage sampleable function* ([GPV08] (hereafter denoted as GPV), and its instantiation over the so-called *NTRU lattices* [HPS98]. The GPV framework enjoys tight and strong security proofs in the quantum random oracle model (QROM) [BDF+11], and its security stems from the hardness of computing a short basis of a large rank lattice. At a high level, the idea is to rely on *trapdoor Gaussian sampling* in a lattice using a secret basis composed of short vectors to generate signatures, while the verification key could be any basis. One particularly promising and interesting candidate based on the GPV setting is Falcon [PFH+19], built upon [HHP+03] and [DLP14]. This scheme ranks among

the best in term of efficiency of its operations without sacrificing on its security and while managing compact signatures and verifications keys, which is usually a drawback of lattice-based signature schemes. In a nutshell, three main features of FALCON are:

- short signatures;
- an efficient key generation algorithm to compute a full short basis of an NTRU lattice;
- an efficient and secure Gaussian sampler.

From the point of view of practical security and side-channel attacks, Gaussian samplers have been known to be a potential weak point (e.g., with respect to timing attacks [BDE+18,TW19,FKT+19]). However, the one used in FALCON was provided with a fully constant-time implementation with only minor losses in efficiency [PRR19].

A caveat of FALCON comes from its complicated implementation, as its efficiency relies on several technical routines and the deep exploitation of the structures of the underlying mathematical objects. In particular, the NTRU lattices can be seen as rank 2 *modules* lattices over towers of rings of algebraic integers: this tower structure (reminiscent of the fast Fourier transform) is at the core of FALCON's performance. More precisely, the rings used in practice are *cyclotomic rings* $R = \mathbb{Z}[x]/(x^d+1)$ whose degree d is a power of 2. They are a common choice for structured lattice-based cryptography, as they enjoy well-understood algebraic properties and leads to to efficient implementations. A drawback of this choice is that powers of 2 are sparse and the bit security mostly depends on d. This implies that if the security level of an implemented scheme must be increased, it is very likely that the updated parameters will incur a significant loss in efficiency while becoming an overkill in term of the reached security level. This is best illustrated with FALCON: taking $d = 512$ leads to an estimated quantum security 103 bits, while $d = 1024$ reaches 230 quantum bit-security, without any intermediate step. At the same time, the signature length jumps from 617 Bytes to 1233 Bytes. Should one wish to achieve a better compromise between security and efficiency, then one would need to select an appropriate "intermediate" ring and redo a full implementation from scratch as there are few to no tower structure among number rings as convenient as the power of two case. In fact, the NIST round 1 version of FALCON proposed an implementation over an appropriate intermediate ring with $d = 768$, reaching 172 bits of quantum security for a signature of 994 Bytes [PFH+17]. But it was considered way too technical and was therefore removed from round 2.

For lattice-based public-key encryption and the other signature paradigm based on the lattice adaptation to Schnorr signatures, this issue was successfully addressed by relying on structured lattices of a larger *module* rank, that is, the rank when seen as a module over R instead of a "plain" lattice. This led to the DILITHIUM [DKL+18,LDK+19] signature, and the KYBER [BDK+18,SAB+19] and SABER [DKRV18,DKRV19] encryption schemes, all competitive candidates in the second round of NIST's call. However, no such module variant was known for NTRU schemes either for key encapsulation or signatures, although there were approaches toward the former with the MaTRU design [CG05].

Contributions In this work, we introduce MODFALCON, a family of *efficient* signature schemes. More precisely, our main contributions are the following:

– We provide a general instantiation of the hash-and-sign paradigm to NTRU lattices of larger module ranks, extending the Falcon design to wider ranges of parameter sets;
– We explain how to generalize in an efficient way the key generation and signature algorithms of Falcon (the extension of the verification algorithm is direct);
– We give a complete security analysis against known attack avenues, which encompasses the analysis of Falcon, and we provide light and documented scripts that compute the bit-security levels from a set of parameters following our analysis.

Enjoying the extra flexibility in the choice of parameters, we obtain compact signatures for different security levels than Falcon. In particular, we obtain the smallest lattice-based signature that enjoys at least 128 bits of quantum security: it is about 25% smaller than Falcon-1024, and almost three times smaller than DILITHIUM-III, which moreover only gets close to this level of quantum security. If one wants to minimize the sum of the signature and public key lengths, then the comparison is even more favorable. See Table 1.

Our design only mildly tweaks the existing description of Falcon, as it focuses on module lattices over the same core cyclotomic rings. This eases the task of the implementors since they can rely on the existing building blocks. This argument, already used as a motivation for DILITHIUM, KYBER and SABER, is even more pressing for Falcon, as the key generation and signature algorithms are significantly more involved. To illustrate this modularity, we provide a proof-of-concept python implementation. Since we build upon the existing core features of Falcon without additional manipulation on secret data, our scheme is natively given as much resistance against timing attacks as the implementation of Falcon provides.

Table 1. Comparison between NIST round 2 lattice-based signature schemes and our new proposal ModFalcon. $|\mathsf{vk}|$ and $|\mathsf{sig}|$ are the sizes in Bytes of the public key and signature, respectively. λ_Q and λ_C stand for the quantum and classical bit security estimates, respectively. For Falcon and ModFalcon, `KR` and `SR` refer to the key and signature recovery modes.

	$\lvert\mathsf{vk}\rvert$	$\lvert\mathsf{sig}\rvert$	λ_Q	λ_C
DILITHIUM-I	896	1387	53	58
DILITHIUM-II	1184	2044	91	100
Falcon-512 (`SR`)	897	658	109	120
Falcon-512 (`KR`)	28	1276		
DILITHIUM-III	1472	2701	125	138
qTESLA-p-I	14880	2592	140	151
DILITHIUM-IV	1760	3366	158	174
Falcon-1024 (`SR`)	1793	1274	252	277
Falcon-1024 (`KR`)	63	2508		
qTESLA-p-III	38432	12352	279	305
ModFalcon-2-512 (`SR`)	**1792**	**972**	**174**	**192**
ModFalcon-2-512 (`KR`)	**940**	**1438**		

As an additional contribution, we also describe how to design a public-key encryption scheme similar in spirit to NTRUEncrypt [HPS98], relying again on module lattices of larger ranks. This encryption scheme can be converted into an adaptively secure key encapsulation mechanism by means of the SXY conversion [SXY18], leading to a tight security proof in the QROM. With this description, we could achieve similar performance as the NTRU [ZCH+19] and (streamline) NTRU Prime [BCLv19] round-2 NIST proposals, though not better. For this reason, we chose to limit ourselves to the description of the scheme and provide the interested readers with details on different approaches in the concrete instantiation of the scheme.

Finally, we extend the results of [SS11,SS13] to show that the verification keys is statistically close to uniform, under some parameter constraints. Note that the concrete parameters that we choose do not satisfy these constraints. If the verification key is close to uniform, then one obtains signature scheme that enjoys strong EUF-CMA security in the QROM, under the Module-SIS hardness assumption [LS15]. This result is deduced from a new matrix version of the leftover hash lemma over number fields, and relies on techniques that may be of independent interest. As this requires additional number theory material, and is somewhat disjoint from the above contributions, this contribution is postponed to the appendix.

2 Preliminaries

For a distribution D, we write $x \leftarrow D$ to express that x is sampled from D. For x in the support of D, we write $D(x)$ to denote the probability of $x \leftarrow D$. For a finite set X, we let $U(X)$ denote the uniform distribution over X. All our vectors are row vectors. Vectors and matrices are written in bold letters, and we additionally use upper case letters for matrices. The line concatenation of two matrices $\mathbf{A}, \mathbf{B}$ is denoted by $\left[\mathbf{A}\middle|\mathbf{B}\right]$ and the column concatenation is denoted by $\left[\frac{\mathbf{A}}{\mathbf{B}}\right]$. For any integer n, we write $\mathbf{I}_n$ resp. $\mathbf{0}_n$ for the identity matrix resp. the zero vector of size n.

2.1 Gaussian measures over lattices

A lattice is a discrete subgroup $\mathcal{L}$ of some $\mathbb{R}^n$. All our lattices will be full-rank. In practice, it is described as the set of integer linear combinations of the rows of some basis $\mathbf{B} \in \mathbb{R}^{n\times n}$. The volume of the lattice is $\mathrm{Vol}\,\mathcal{L} := |\det \mathbf{B}|$, for any basis $\mathbf{B}$ of $\mathcal{L}$.

The spherical Gaussian function on $\mathbb{R}^d$ centered at $\mathbf{c}$ and with standard deviation $s > 0$ is defined as $\rho_{s,\mathbf{c}}(\mathbf{x}) = \exp(-\|\mathbf{x}-\mathbf{c}\|^2/(2s^2))$. If $\mathbf{c} = \mathbf{0}$, we often drop $\mathbf{c}$. For any any rank-d lattice $\mathcal{L}$, the discrete Gaussian distribution with support $\mathcal{L}$ and parameters $s > 0$ and $\mathbf{c}$ is defined as

$$\forall\, \mathbf{x} \in \mathcal{L}, D_{s,\mathcal{L},\mathbf{c}}(\mathbf{x}) = \frac{\rho_{s,\mathbf{c}}(\mathbf{x})}{\rho_{s,\mathbf{c}}(\mathcal{L})}.$$

Given $\epsilon > 0$, the smoothing parameter $\eta_\epsilon(\mathcal{L})$ is the smallest real $s > 0$ such that $\rho_{1/s}(\mathcal{L}^*) \leq 1 + \epsilon$, where $\mathcal{L}^*$ is the dual lattice. In a sense, it quantifies the standard deviation parameter needed to "smooth out" the discreteness of the lattice. We omit the definition of the dual, as it is not needed further in this work. Rather, we recall the

following standard upper bound[5] on the smoothing parameter, which we will use for instantiating practical parameters.

Lemma 2.1 (Adapted from [GPV08]). *Let $\mathcal{L}$ be any rank d lattice with basis $\mathbf{B}$, and let $\epsilon > 0$. We have:*

$$\eta_\epsilon(\mathcal{L}) \leq \|\mathbf{B}\|_{\mathrm{GS}} \cdot \frac{1}{\pi}\sqrt{\log(2d(1+1/\epsilon))/2}.$$

The quantity $\|\mathbf{B}\|_{\mathrm{GS}}$ is the norm of the largest vector in the Gram-Schmidt Orthogonalization of $\mathbf{B}$ (see also Section 2.3).

2.2 Cyclotomic fields and NTRU lattices

We let d be a power of 2 and write $K = \mathbb{Q}[x]/(x^d+1)$ the corresponding cyclotomic field. In this setup, $R = \mathbb{Z}[x]/(x^d+1)$ is the ring of integers of K, and for any prime integer q, we define $R_q = R/qR \simeq \mathbb{Z}_q[x]/(x^d+1)$. There are several ways of embedding this number field in a normed vector space. In this work, we use two of them. An element in K can be embedded by its coefficients: $a = \sum_i a_i x^i \in K$ gives $(a_0, \ldots, a_{d-1}) \in \mathbb{Q}^d$. Abusing notations, we will write a to denote both the element in K and its vector of coefficients. We can then consider the ℓ_∞-norm of elements of K as $\|a\|_\infty = \max_{i\in[d-1]} |a_i|$, and their Euclidean norm is $\|a\| = \sqrt{\langle a, a\rangle} = (\sum_i |a_i|^2)^{1/2}$, where $\langle , \rangle$ is the standard inner product over $\mathbb{R}^d$. Observe that $\|x^i a\| = \|a\|$ for all $i \in [d]$. For all $a \in K$ we let $a^\star := a(x^{-1}) = (a_0, -a_{d-1}, \ldots, -a_1)$.

Elements in K can also be represented by their nega-circulant matrix of multiplication, when seeing K as a $\mathbb{Q}$-linear space with basis $1, x, \ldots, x^{d-1}$. In other words, the matrix $\mathrm{M}(a)$ has rows the coefficient vectors of $a, xa, \ldots, x^{d-1}a$, and we have $\mathrm{M}(ab) = \mathrm{M}(a)\,\mathrm{M}(b)$. It can be seen that $\mathrm{M}(a^\star) = \mathrm{M}(a)^t$. Thus for all $a, b \in K$, we have $\langle a, b\rangle = \langle ab^\star, 1\rangle$, so that $\|a\|^2$ is the constant coefficient of $aa^\star$.

These notions can be extended to the linear space K^n by concatenating the coefficient vectors of an element $\mathbf{a} = (a_1, \ldots, a_n)$. The norms over $\mathbb{Q}^d$ extends as $\|\mathbf{a}\|_\infty = \max_{i\in[n]} \|a_i\|_\infty$ and $\|\mathbf{a}\| = (\sum_{i\in[n]} \|a_i\|^2)^{1/2}$. We extend the $\star$ operator component-wise, and consider the K-bilinear form $\langle \mathbf{a}, \mathbf{b}\rangle_K = \sum_i a_i b_i^\star$. The latter corresponds to the Euclidean norm, in the sense that

$$\|\mathbf{a}\|^2 = \sum_i \|a_i\|^2 = \langle \sum_i a_i a_i^\star, 1\rangle = \langle \langle \mathbf{a}, \mathbf{a}\rangle_K, 1\rangle,$$

or, in other words, $\|\mathbf{a}\|^2$ is the constant coefficient of $\langle \mathbf{a}, \mathbf{a}\rangle_K$. We also extend the matrix representation to vectors over K:

$$\mathrm{M}(\mathbf{a}) = [\mathrm{M}(a_1)| \ldots | \mathrm{M}(a_n)] \in \mathbb{Q}^{d\times nd},$$

which can also be used for matrices over K.

[5] Our formulation takes into account a different normalization for the Gaussian function than [GPV08].

NTRU module lattices We call a module in K^m a subset of the form $\mathcal{M}{=}R\mathbf{b}_1{+}\cdots{+}R\mathbf{b}_n$ and such that $\mathrm{Span}_K(\mathbf{b}_1,\ldots,\mathbf{b}_n)$ has dimension n.[6] Observe that for all $x \in R$ and all $\mathbf{a} \in \mathcal{M}$, we have $x\mathbf{a} \in \mathcal{M}$. Let now $\mathbf{F} \in R^{n\times n}$ be invertible modulo some prime integer q, and $\mathbf{g} \in R^n$. We also let $\mathbf{h}^t = \mathbf{F}^{-1}\mathbf{g}^t \bmod q \in R^n$, and define the NTRU module as

$$\mathcal{L}_{\mathrm{NTRU}} := \{(u,\mathbf{v}) \in R^{n+1} \ : \ u + \mathbf{v}\mathbf{h}^t = 0 \bmod q\}.$$

It contains qR^{n+1} so it is in particular of full rank $n+1$. Recall that for any invertible $\mathbf{F} \in R^{n\times n}$, the adjugate of $\mathbf{F}$ is the unique matrix $\mathrm{adj}(\mathbf{F})$ satisfying $\mathrm{adj}(\mathbf{F})\cdot\mathbf{F} = \mathbf{F}\cdot\mathrm{adj}(\mathbf{F}) = (\det_K \mathbf{F})\cdot\mathbf{I}_n$. If there exists $g_0 \in R$ and $\mathbf{f}_0 \in R^n$ such that $g_0 \cdot \det_K \mathbf{F} - \mathbf{f}_0 \cdot \mathrm{adj}(\mathbf{F})\cdot\mathbf{g}^t = q \in R$, then $\mathcal{L}_{\mathrm{NTRU}}$ admits bases in K^{n+1} in the form of

$$\mathbf{B}_{\mathrm{NTRU}} = \begin{pmatrix} -\mathbf{h}^t & \mathbf{I}_n \\ q & \mathbf{0}_n \end{pmatrix} \quad \text{and} \quad \mathbf{B}_{\mathbf{F},\mathbf{g}} = \begin{pmatrix} \mathbf{g}^t & -\mathbf{F} \\ g_0 & -\mathbf{f}_0 \end{pmatrix},$$

since Schur's complement formula shows that $\det_K \mathbf{B}_{\mathbf{F},\mathbf{g}} = q$ and one can check that $\mathbf{B}_{\mathbf{F},\mathbf{g}} \cdot \begin{bmatrix} 1 \\ \mathbf{h}^t \end{bmatrix} = 0 \bmod q$. As any module, $\mathcal{L}_{\mathrm{NTRU}}$ can be seen as a $\mathbb{Z}$-lattice in $\mathbb{Q}^{(n+1)d}$ by concatenation of the coefficient vectors. We can see that $\mathrm{M}(\mathbf{B}_{\mathbf{F},\mathbf{g}}) \in \mathbb{Z}^{(n+1)d\times(n+1)d}$ is a basis of the underlying lattice, so an NTRU lattice has volume q^d.

2.3 Gram-Schmidt orthogonalization

Let $n \le m$, and $\mathbb{F}$ be a field (either $\mathbb{R}$ or $K = \mathbb{Q}[x]/(x^d+1)$). Let $\langle,\rangle$ be a non degenerate $\mathbb{F}$-bilinear form, i.e., such that having $\langle\mathbf{a},\mathbf{a}\rangle = 0$ implies that $\mathbf{a} = \mathbf{0} \in \mathbb{F}^m$. When $\mathbb{F} = \mathbb{R}$ the form $\langle,\rangle$ is the standard inner product, and when $\mathbb{F} = K$, we will consider $\langle\mathbf{a},\mathbf{b}\rangle_K = \sum_{i\in[m]} a_i b_i^\star$. We say that $\mathbf{a},\mathbf{b} \in \mathbb{F}^m$ are orthogonal if $\langle\mathbf{a},\mathbf{b}\rangle = \mathbf{0}$. For any $\mathbf{B} \in \mathbb{F}^{n\times m}$ of rank n with rows $\mathbf{b}_i$'s, the Gram-Schmidt Orthogonalization (GSO) with respect to $\langle,\rangle$ builds $\mathbb{F}$-linearly independent $\widetilde{\mathbf{b}_i}$'s by the formula

$$\widetilde{\mathbf{b}_1} = \mathbf{b}_1 \quad \text{and} \quad \widetilde{\mathbf{b}_i} = \mathbf{b}_i - \sum_{j<i} \frac{\langle\mathbf{b}_i,\widetilde{\mathbf{b}_j}\rangle}{\langle\widetilde{\mathbf{b}_j},\widetilde{\mathbf{b}_j}\rangle}\cdot\widetilde{\mathbf{b}_j} \quad \text{for} \quad i > 1.$$

We have $\mathrm{Span}(\widetilde{\mathbf{b}_1},\cdots,\widetilde{\mathbf{b}_i}) = \mathrm{Span}(\mathbf{b}_1,\cdots,\mathbf{b}_i)$ for all $i \in [n]$. The formula also describes that $\widetilde{\mathbf{b}_i}$ is the orthogonal projection of $\mathbf{b}_i$ onto the space spanned by $\widetilde{\mathbf{b}_1},\cdots,\widetilde{\mathbf{b}_{i-1}}$. The GSO amounts to writing $\mathbf{B} = \mathbf{L}\cdot\widetilde{\mathbf{B}}$, such that $\mathbf{L}$ is lower triangular with 1's on its diagonal, and the rows $\widetilde{\mathbf{b}_i}$'s of $\widetilde{\mathbf{B}}$ are pairwise orthogonal for $\langle\cdot,\cdot\rangle$. This decomposition is unique, and we have $\det(\mathbf{B}\mathbf{B}^\star) = \det(\widetilde{\mathbf{B}}\widetilde{\mathbf{B}}^\star) = \prod_{i\in[n]}\langle\widetilde{\mathbf{b}_i},\widetilde{\mathbf{b}_i}\rangle$. We define the Gram-Schmidt norm of $\mathbf{B} \in \mathbb{R}^{n\times m}$ as $\|\mathbf{B}\|_{\mathrm{GS}} := \max_{i\in[n]}\|\widetilde{\mathbf{b}_i}\|$.

If $\mathbf{B} \in K^{n\times m}$, observe that $\mathrm{M}(\mathbf{B})$ in $\mathbb{Q}^{nd\times md}$ is not the standard GSO of $\mathrm{M}(\mathbf{B})$, as the former matrix has a block structure while the latter one has not in general. However, the operator M allows to relate several interesting properties from the GSO over K to the GSO over $\mathbb{R}$, that we gather in the next lemma. Its proof is inspired from [DLP14,DP16].

Lemma 2.2. *Let* $\mathbf{B} = [\mathbf{b}_1,\ldots,\mathbf{b}_n]$ *of* K^n *and* $\mathrm{M}(\mathbf{B}) = [\mathbf{r}_1,\ldots,\mathbf{r}_{nd}]$. *The following holds for all* $i \in [n]$*:*

[6] Formally, these are free and finitely generated R-modules in K^m.

– *the coefficient vector of* $\widetilde{\mathbf{b}}_i$ *is* $\widetilde{\mathbf{r}}_{(i-1)d+1}$*;*
– $\det(\mathrm{M}(\mathbf{B}\mathbf{B}^\star)) = \prod_{i=1}^{nd} \|\widetilde{\mathbf{r}}_i\|^2$*;*
– $\|\mathrm{M}(\mathbf{B})\|_{\mathrm{GS}} = \max\{\|\widetilde{\mathbf{r}}_1\|, \|\widetilde{\mathbf{r}}_{d+1}\|, \ldots, \|\widetilde{\mathbf{r}}_{(n-1)d+1}\|\}$.

Proof. Let $V_i = \mathrm{Span}_K(\mathbf{b}_1, \ldots, \mathbf{b}_i)$. With an abuse of notation, write $\mathrm{M}(V_i) = \mathrm{RowSpan}([\mathrm{M}(\mathbf{b}_1)|\ldots|\mathrm{M}(\mathbf{b}_i)]$. For every i, we have $\dim_{\mathbb{Q}} \mathrm{M}(V_i) = d \cdot \dim_K V_i$. By definition, we know that $\widetilde{\mathbf{b}}_i$ is orthogonal to V_{i-1}. This means that $\mathrm{M}(\langle \widetilde{\mathbf{b}}_i, \widetilde{\mathbf{b}_j} \rangle_K) = \sum_{k\in[n]} \mathrm{M}(\widetilde{b}_{ik})\,\mathrm{M}(\widetilde{b}_{jk})^t$ is the zero matrix in $\mathbb{Q}^{d\times d}$ for all $j \leq i-1$. Said differently, the space $\mathrm{RowSpan}(\mathrm{M}(\widetilde{\mathbf{b}_i}))$ is orthogonal to $\mathrm{M}(V_{i-1})$. By definition of the GSO, this implies that (the coefficient vector of) $\widetilde{\mathbf{b}}_i$ is orthogonal to $\mathrm{Span}_{\mathbb{Q}}(\mathbf{r}_1, \ldots, \mathbf{r}_{(i-1)d})$. By unicity of the GSO, we see that $\widetilde{\mathbf{b}}_i = \widetilde{\mathbf{r}}_{(i-1)d+1}$.

Next, the orthogonality in K implies that $\mathrm{M}(\widetilde{\mathbf{B}})\,\mathrm{M}(\widetilde{\mathbf{B}})^t$ is a block-diagonal matrix, with blocks $\mathrm{M}(\langle \widetilde{\mathbf{b}}_i, \widetilde{\mathbf{b}}_i \rangle_K)$ for $i \leq n$. This gives

$$\det(\mathrm{M}(\mathbf{B}\mathbf{B}^\star)) = \det(\mathrm{M}(\widetilde{\mathbf{B}}\widetilde{\mathbf{B}^\star})) = \prod_{i\in[n]} \det(\mathrm{M}(\langle \widetilde{\mathbf{b}}_i, \widetilde{\mathbf{b}}_i \rangle_K)) = \prod_{i=1}^{nd} \|\widetilde{\mathbf{r}}_i\|^2.$$

For the last statement, note that for every i and j, the vector $\mathbf{r}_{id+j}$ is a projection of a vector that has the same norm as $\mathbf{r}_{id+1}$.

3 ModFalcon

Our construction of a Module-NTRU signature scheme is based on the GPV framework [GPV08]. The public key is a pseudorandom matrix $\mathbf{A}$, whereas the private key is some trapdoor information about $\mathbf{A}$: typically (and is the case for our scheme too), it consists of a short basis $\mathbf{B}$ of the module lattice "orthogonal modulo q" to the lattice generated by $\mathbf{A}$. Previous works [SS13,DLP14,PFH+19] have instantiated the GPV framework with NTRU lattices for $n = 1$. Here we describe instantiations for $n \geq 2$. This provides us extra flexibility in setting the parameters compared to e.g. Falcon [PFH+19], and allows to reach more levels of security.

During the signing procedure, the signer hashes the message to a point $\mu \in R_q$ and uses the trapdoor information in conjunction with an algorithm called a *trapdoor sampler* in order to compute a short preimage of μ, i.e., a short vector $\mathbf{s}$ such that $\mathbf{s} \cdot \mathbf{A} = \mu$. For the trapdoor sampler, there exist a few possibilities, with different trade-offs in terms of speed, simplicity, and security (the shorter vectors a trapdoor sampler outputs, the more security it provides). We use the fast Fourier sampler used in Falcon [DP16]. This can be done in $O(d \log d)$ arithmetic operations (by exploiting the tower structure of $R = \mathbb{Z}[x]/(x^d + 1)$), while achieving a high level of security. Indeed, the GPV framework enjoys the tight QROM security proof of [BDF+11]. As most of the aspects of our design are inspired from Falcon, we call our module generalization ModFalcon.

To allow for an efficient multiplication in R_q via the Number Theoretic Transform, we choose a prime integer q such that $q = 1 \bmod 2d$. We could alternatively use a prime q satisfying $q = 3 \bmod 8$, which is a parameter condition for the statistical study of the verification key of Section A. Let $n \geq 1$ be an integer, and let D_f and D_g be distributions over $R^{n\times n}$ and R^n, respectively. Let $\mathsf{H}\colon \{0,1\}^* \to R_q$ be a cryptographic hash function modeled as a random oracle. Finally, let $\mathsf{Compress}\colon R^{n+1} \to \{0,1\}^*$ and

$\mathsf{Decompress}\colon \{0,1\}^* \to R^{n+1}$ be efficient maps such that $\mathsf{Decompress} \circ \mathsf{Compress}$ is the identity.

Our key generation, signature and verification algorithms follow closely the description of FALCON, however some steps do not readily generalize. Below, we describe the modifications in details.

3.1 On key generation

Key generation

repeat
  Sample $\mathbf{F} \hookleftarrow D_f$ and $\mathbf{g} \hookleftarrow D_g$
until $\mathbf{F}$ invertible mod q and $\|\mathrm{M}(\mathbf{B}_{\mathbf{F},\mathbf{g}})\|_{\mathrm{GS}} \leq \text{GS\_SLACK} \cdot q^{1/(n+1)}$
Complete $\left[\mathbf{g}^t | -\mathbf{F}\right]$ into a basis $\mathbf{B}_{\mathbf{F},\mathbf{g}}$ of $\mathcal{L}_{\mathrm{NTRU}}$
Compute $\mathbf{h}^t = \mathbf{F}^{-1}\mathbf{g}^t \bmod q$
return $(\mathsf{vk} = \left[\frac{1}{\mathbf{h}^t}\right], \mathsf{sk} = \mathbf{B}_{\mathbf{F},\mathbf{g}})$

The most important differences between FALCON and MODFALCON are in the key generation; since we generate different, more generic lattices, our methods and choice of parameters need to be generalized as well.

The Gram-Schmidt slack. Using Lemma 2.2 and the fact that the NTRU lattice has determinant q^d, one checks that $\|\mathbf{B}\|_{\mathrm{GS}} \geq q^{1/(n+1)}$. In [DLP14], it is experimentally shown that for $n = 1$, one can carefully select D_f and D_g so that one can get close to that optimal lower bound $\sqrt{q}$ by a factor 1.17. By performing our own experiments, we extend this approach to higher values of n. We found that with constant probability, one can get $\|\mathbf{B}\|_{\mathrm{GS}} \leq \text{GS\_SLACK} \cdot q^{1/(n+1)}$, where $\text{GS\_SLACK} \geq 1$ is some "slack" which quantifies the gap between the lower bound of $\|\mathbf{B}\|_{\mathrm{GS}}$ and what we can achieve in practice. Concretely, we take:

$$\begin{aligned} &\text{For } n = 1,\ \text{GS\_SLACK} = 1.17; \\ &\text{For } n = 2,\ \text{GS\_SLACK} = 1.17; \\ &\text{For } n = 3,\ \text{GS\_SLACK} = 1.24. \end{aligned} \tag{1}$$

The distributions D_f and D_g. For the security of trapdoor sampling procedure in the GPV framework, it is needed to sample discrete Gaussians in R with a standard deviation parameter above the GS-norm of the trapdoor basis. To maximize security, we would like this GS-norm to be as small as possible (this makes signature forgery harder). Thanks to Lemma 2.2, we can control $\|\mathbf{B}_{\mathbf{F},\mathbf{g}}\|_{\mathrm{GS}}$ by careful sampling of the rows of $\mathbf{B}_{\mathbf{F},\mathbf{g}}$. In particular, we can try to have all rows of essentially the same norm.

To do so, we notice that orthogonalizing a random vector over a random subspace of dimension $n+1-i$ of K^{n+1} typically shrinks its Euclidean norm by a factor $\sqrt{\frac{n+1}{n+1-i}}$. This effect can be compensated by sampling each row $[\mathbf{f}_{i1}, \ldots, \mathbf{f}_{in}, \mathbf{g}_i]$ according to a discrete Gaussian with standard deviation $\text{GS\_SLACK} \cdot \frac{1}{\sqrt{d(n+2-i)}} \cdot q^{1/(n+1)}$ (the factor d comes from the fact that the Euclidean norms are for vectors in $\mathbb{R}^{d(n+1)}$). Overall,

we expect that $\|\mathbf{B}\|_{\mathrm{GS}} \leq \mathrm{GS\_SLACK} \cdot q^{1/(n+1)}$ with constant probability (if n is bounded from above by a constant).

Completing the trapdoor basis. A full basis of the NTRU lattice is needed to perform trapdoor sampling. Hence we need an extra vector $\left[g \middle| -\mathbf{f}\right] \in R^{n+1}$ to complete $\left[\mathbf{g}^t \middle| -\mathbf{F}\right]$ into $\mathbf{B}_{\mathbf{F},\mathbf{g}}$. When $n = 1$, this boils down to solving a "Bézout"-like equation over R, which can be done efficiently by a recursive approach [PP19] exploiting the tower structure of $R = \mathbb{Z}[x]/(x^d + 1)$. We show how to reduce our current situation to the case $n = 1$. From Schur's complement formula, we have:

$$\begin{aligned}\det_K(\mathbf{B}_{\mathbf{F},\mathbf{g}}) &= \det_K(\mathbf{F}) \cdot \det_K(g - \mathbf{f} \cdot \mathbf{F}^{-1} \cdot \mathbf{g}^t) \\ &= \det_K(\mathbf{F}) \cdot g - \mathbf{f} \cdot \mathrm{adj}(\mathbf{F}) \cdot \mathbf{g}^t.\end{aligned}$$

We let $\mathbf{f} = (f, 0, \ldots, 0)$, so that this equation becomes $g \cdot \det_K(\mathbf{F}) - A \cdot f = q \in R$, where $\det(\mathbf{F}) \in R$, $A \in R$ is the first coordinate of $\mathrm{adj}(\mathbf{F}) \cdot \mathbf{g}^t$ and can be computed once $\mathbf{F}$ and $\mathbf{g}$ have been generated. Then we can solve for f, g using [PP19].

Implicit computation of $\|\mathrm{M}(\mathbf{B}_{\mathbf{F},\mathbf{g}})\|_{\mathrm{GS}}$. The matrix $\mathbf{B}_{\mathbf{F},\mathbf{g}}$ is entirely resampled if its Gram-Schmidt norm is larger than a certain threshold, thus it might be resampled a few times; on the other hand, computing $\left[g \middle| -\mathbf{f}\right]$ (following the trapdoor completion procedure described in the previous paragraph) is somewhat computationally expensive so we only want to do it once. Recall from Lemma 2.2 that $\|\mathrm{M}(\mathbf{B}_{\mathbf{F},\mathbf{g}})\|_{\mathrm{GS}} = \max_{i \in [n]}\{\|\widetilde{\mathbf{b}}_i\|\}$, and that we have

$$q^2 = \det_K(\mathbf{B}_{\mathbf{F},\mathbf{g}} \cdot \mathbf{B}_{\mathbf{F},\mathbf{g}}{}^\star) = \prod_{i=1}^{n+1} \langle \widetilde{\mathbf{b}}_i, \widetilde{\mathbf{b}}_i \rangle_K.$$

Once $\left[\mathbf{g}^t \middle| -\mathbf{F}\right]$ has been sampled, we can apply Lemma 2.2 to compute all the $\langle \widetilde{\mathbf{b}}_i, \widetilde{\mathbf{b}}_i \rangle_K$'s for $i \leq n$ and deduce the remaining $\langle \widetilde{\mathbf{b}}_{n+1}, \widetilde{\mathbf{b}}_{n+1} \rangle_K$, of which $\|\widetilde{\mathbf{b}}_{n+1}\|^2$ is the constant coefficient. We can hence compute $\|\mathrm{M}(\mathbf{B}_{\mathbf{F},\mathbf{g}})\|_{\mathrm{GS}}$ *before* completing $\mathrm{M}(\mathbf{B}_{\mathbf{F},\mathbf{g}})$, allowing us to make only one single call to the trapdoor completion procedure.

3.2 Signature and verification

As in the original FALCON scheme, the signature is a pair (r, S), where r is a hashing salt and S encodes a short vector $\mathbf{s}$ such that $\mathbf{s} \cdot \mathsf{vk} = H(\mathsf{r} \| \mathsf{msg})$. The core technical part of the signing procedure is, once $H(\mathsf{r} \| \mathsf{msg})$ has been computed, to use the secret key $\mathbf{B}_{\mathbf{F},\mathbf{g}}$ to sample a proper $\mathbf{s}$. This is done via a technique known as fast Fourier sampling, developed in [DP16,PFH+19].

Signature: $(\mathsf{sk}, \mathsf{msg}) \to (\mathsf{r}, S)$

Require: A standard deviation parameter σ

1: Get $\mathsf{r} \hookleftarrow U(\{0,1\})^{\lambda_r}$
2: $\mu \leftarrow \mathsf{H}(\mathsf{r} \| \mathsf{msg}) \in R_q$ and let $\mathbf{c} = (\mu, 0, \cdots, 0)$
3: Compute $\mathbf{t} = \mathbf{c} \cdot \mathbf{B}_{\mathbf{F},\mathbf{g}}^{-1}$
4: Compute $\mathbf{z} \in R^{n+1}$ such that $\mathbf{s} := (\mathbf{t} - \mathbf{z}) \cdot \mathbf{B}_{\mathbf{F},\mathbf{g}} \hookleftarrow D_{\sigma, \mathcal{L}_{\mathrm{NTRU}}, \mathbf{c}}$
5: $S = \mathsf{Compress}(\mathbf{s})$
6: **return** the signature (r, S).

Verification: $(\mathsf{vk}, \mathsf{msg}, (\mathsf{r}, S)) \to$ **accept** or **reject**

Require: A fixed bound ρ on the length of the signature

1: $\mathbf{s} \leftarrow \mathsf{Decompress}(S)$
2: If $\|\mathbf{s}\| > \rho$, **return reject**
3: If $\mathbf{s} \cdot \mathsf{vk} \neq \mathsf{H}(\mathsf{r}\|\mathsf{msg})$, **return reject**
4: **return accept**

The procedures Compress *and* Decompress. Our compression and decompression procedures are identical to the ones used in Falcon [PFH+19, Section 3.11.2]: for each integer coefficient, the sign as well as the $\lfloor \log_2(\sigma) \rfloor - 1$ least significant bits are naively encoded (i.e., copy-pasted), whereas the remaining most significant bits are encoded following a unary encoding.

The standard deviation parameter σ. This parameter should be large enough so that the output distribution of the fast Fourier sampler is close to a prefect discrete Gaussian. To apply the Rényi divergence arguments of [Pre17, Section 3.3 and Lemma 6], it suffices to take $\sigma \geq \eta_\epsilon(R^{n+1}) \cdot \|\mathrm{M}(\mathbf{B}_{\mathbf{F},\mathbf{g}})\|_{\mathrm{GS}}$ with $\epsilon = \frac{1}{4\sqrt{\lambda Q_s}}$, where Q_s is an a priori upper bound on the total number of signatures generated using a single key pair, and $\lambda := 256$ is an upper bound of the bit security we are aiming at (we could optimize the value of λ for specific bit-security targets, but this has negligible impact).Concretely, the standard deviation parameter σ is set using the bound on η_ϵ given by Lemma 2.1.

Fast Fourier sampling for module NTRU Lattices An in-depth description of Fourier sampling is outside of the scope of this work, and we refer the interested reader to [PFH+19]. Here, we outline the main operations and how they can be modified to fit our design. The sampler uses a well-designed tree representation of the trapdoor basis $\mathbf{B}_{\mathbf{F},\mathbf{g}}$. Recall that the LDL decomposition of a symmetric positive definite matrix $\mathbf{G}$ writes it uniquely as $\mathbf{G} = \mathbf{L}\mathbf{D}\mathbf{L}^\star$, where $\mathbf{L}$ is lower triangular with 1's on its diagonal, and $\mathbf{D}$ is diagonal with positive entries. The tree is built using successive LDL decompositions over K of the Gram matrix $\mathbf{G} = \mathbf{B}_{\mathbf{F},\mathbf{g}} \cdot \mathbf{B}^\star_{\mathbf{F},\mathbf{g}}$.

The root of the tree is labeled the lower corner of the first $\mathbf{L}$ factor, and the leaves corresponds to the entries in the diagonal matrix $\mathbf{D}$. The next level of the tree is obtained recursively by repeating the procedure on the diagonal blocks, using the fact that K has a tower structure of quadratic extensions over $\mathbb{Q}$. At the bottom of the tree, the leaves are labeled with rationals describing the needed standard deviations for the sampler to output signatures with the correct distribution.

For Falcon, with $n = 1$, the first level of the tree contains two entries, as $\mathbf{G} \in K^{2\times 2}$. In our design, and more generally for an input basis in $K^{n\times n}$, the first level is labeled by the $n(n-1)/2$ entries in the lower corner of $\mathbf{L}$, and there are n leaves each corresponding to an non-zero entry of $\mathbf{D}$. This is the only modification for building the tree as, starting at the next level, the same procedure as in Falcon is used to complete the tree.

The rejection bound ρ. For security, we want ρ to be small. However, the norm of a Gaussian of standard deviation σ will have a median and an expected value of $\sigma\sqrt{d(n+1)}$, so if we do not want to restart the signing procedure too many times, we should take ρ larger than this value. One can explicitly bound the rejection probability using [Lyu12,

Lemma 4.4]; for example taking

$$\rho = \lfloor 1.1 \cdot \sigma \sqrt{d(n+1)} \rceil. \tag{2}$$

ensures, for all parameter sets in this paper, that less than 1 % of the signatures will be rejected. This is clearly sufficient for our purposes.

The salt r. Outputting two different signatures $\mathbf{s} \neq \mathbf{s}'$ for the same hash μ allows to get a short vector in the NTRU lattice and is therefore highly undesirable. If the salt r does not have enough entropy, an adversary may query signatures for the same message msg until $\mathsf{H}(\mathsf{r}\|\mathsf{msg}) = \mathsf{H}(\mathsf{r}'\|\mathsf{msg})$ for $\mathsf{r} \neq \mathsf{r}'$. We require that:

$$\lambda_r \geq \lambda + \log_2 Q_s, \tag{3}$$

where λ is as above. Taking $\mathsf{r} \leftarrow U(\{0,1\}^{\lambda_r})$ and applying the birthday paradox, an adversary making Q_s signature queries will find colliding hashes with a probability upper bounded by about $1 - \exp(-Q_s^2/2^{\lambda_r+1}) \approx Q_s^2/2^{\lambda_r+1}$.

Key-recovery and signature-recovery modes. The scheme can be instantiated in *signature-recovery mode*. Writing $\mathbf{s} = (s_1, \mathbf{s}') \in R \times R^n$, we have

$$s_1 = \mathsf{H}(\mathsf{r}\|\mathsf{msg}) - \mathbf{s}' \cdot \mathbf{h}^t, \tag{4}$$

therefore s_1 can be deduced from the rest of the signature and does not need to be sent. This optimization is used in [DLP14,PFH+19] to reduce the signature size and it applies here too. This shrinks the signature size by a factor roughly $(n+1)/n$.

The scheme can also be instantiated in *key-recovery mode*. A special case of this idea is proposed in [PFH+19, Section 3.13]. We generalize it here. Observe that if $n-1$ entries of $\mathbf{h}$ and a hash of $\mathbf{h}$ are known, then this is enough to recover $\mathbf{h}$ entirely. Indeed, the public key $\mathbf{h}$ satisfies the linear equation (4). Hence, upon reception of a signature, one can recompute a candidate $\mathbf{h}^*$ from this equation and check whether its hash matches the one in the public key. This allows to replace the public key by a hash thereof.

Asymptotic security of ModFalcon A signature scheme based on the GPV framework can be shown to be enjoy (strongly) existential unforgeability against adaptatively chosen message attacks (also abbreviated as sEUF-CMA security) in the classical random oracle model, when the verification key is statistically close to uniformly random [GPV08]. It was showed in [BDF+11] that sEUF-CMA security also holds in the quantum random oracle model, in which the adversary can make quantum superposition queries to the random oracle. For the case $n = 1$, the distribution of the public key is almost uniform if the entries of $\mathbf{f}, \mathbf{g}$ are discrete Gaussian of standard deviation around $\sqrt{q}$ (see [SS13]). The next statement extends this result to larger ranges of n. It is adapted from a more general new result presented and proved in Appendix A.

Theorem 3.1. *Let $n \geq 1$ be an integer and $q = 3 \bmod 8$ be a prime. Let $s \geq 2dq^{1/(n+1)+2/(d(n+1))}$, and $\mathcal{E}_s$ be the distribution of $\mathbf{F}^{-1}\mathbf{g}^t \bmod q$, when $\mathbf{F} \hookleftarrow D_{s,R^{n\times n}}$ is invertible modulo q and $\mathbf{g} \hookleftarrow D_{s,R^n}$. Then the statistical distance between $\mathcal{E}_s$ and the uniform distribution over R_q^n is $2^{-\Omega(d)}$.*

Note that our concrete scheme parameters are not compatible with the above, because they do not satisfy the assumptions of Theorem 3.1. Nevertheless, we believe they increase our confidence in the soundness of the design rationale.

4 Concrete instantiation

In this section, we explain how to instantiate the various parameters of ModFalcon, to optimize the sizes of the signatures and public keys, under the correctness and security constraints.

4.1 Setting the scheme variables

We briefly summarize how we obtain the scheme variables. The maximal number of signature queries is $Q_s 2^{64}$; this is the number specified in the NIST call for post-quantum cryptography standardization [NIS16]. The value $\text{GS\_SLACK} \geq 1$ is deduced from extensive experiments and given in (1). The salt bitsize λ_r shall verify (3); just like Falcon did, we simply take $\lambda_r = 320$ (for the bit security that we achieve, $\lambda_r = 256$ would actually be sufficient).Finally, the rejection bound ρ on the Euclidean norm of signatures is given in (2). Table 2 lists scheme variables for various parameter sets of Falcon and ModFalcon. All the parameter sets we consider use the same modulus $q = 12289$, as efficient code for the arithmetic in the resulting ring R_q is available.

Table 2. Variables for Falcon and ModFalcon parameter sets

Scheme	n	d	GS_SLACK	ρ	λ_r
Falcon-512	1	512	1.17	6598	320
Falcon-1024	1	1024	1.17	9331	320
ModFalcon-2-512	2	512	1.17	1512	320

4.2 Security analysis of ModFalcon

Our signature scheme follows the design of Falcon [PFH+19], and the applicable attacks are similar. The most notable difference is the use of sublattices to produce signature forgeries: this attack strategy does not seem fruitful in the case of Falcon, but it actually drives the concrete security in the module setup.

The most efficient attacks against schemes based on NTRU-type cryptosystems typically rely on lattice reduction [Sch87,SE94]. The lattices to be reduced correspond to $\mathbb{Z}[x]/(x^d+1)$-modules of (module) rank ≥ 2. There exist algorithms [CDPR16,CDW17,PHS19] to compute short vectors in rank-1 $\mathbb{Z}[x]/(x^d + 1)$-modules (a.k.a. ideal lattices), which outperform algorithms for generic lattices such as [SE94] for some ranges of approximation factors. No such improvement over all-purpose lattice reduction algorithms is known for modules of rank ≥ 2. For example, the recent module-LLL algorithm from [LPSW19] relies on an oracle for solving the Closest Vector Problem for lattices of very high dimensions. It is possible to use automorphisms (multiplying a given short polynomial by x modulo $x^d + 1$ to create another short polynomial), but this is not known to bring more than a small polynomial improvement that is negligible compared to the overall exponential cost. For such module lattices or rank $n \geq 2$, the approach so far is to view them as $\mathbb{Z}$-lattices of dimension nd. This is the approach used to analyze the security of all 11 candidates of the 2nd round of the NIST post-quantum standardization

process that rely on algebraic lattices. Our analysis thus focuses on $\mathbb{Z}$-lattice reduction algorithms, and follows standard works on NTRU schemes [ZCH+19,BCLv19].

Lattice reduction attacks As is now standard in lattice-based cryptography, we follow the *core SVP hardness* methodology put forward in [ADPS16]. The attack is viewed as an instance of a lattice problem, which is solved by the BKZ algorithm [SE94]. At a high level, the BKZ algorithm calls an SVP oracle in dimension β as a subroutine, where β is the selected block-size. The analysis determines the minimal β that allows to break the scheme, and the cost of the attack is bounded from below by the asymptotic cost of the best known algorithm for classically [BDGL16] or quantumly solving the Shortest Vector Problem with this block-size [Laa15]. This strategy is typically viewed as conservative, as BKZ in fact calls the SVP-solver more than once, the asymptotic cost of the SVP-solver hides polynomial factors and SVP-solvers typically require the management of a large amount of (potentially quantum) memory. The algorithm from [BDGL16] (resp. [Laa15]) returns a shortest non-zero vector in a lattice of dimension β in $2^{0.292\beta(1+o(1))}$ classical operations (resp. $2^{0.265\beta(1+o(1))}$ quantum operations). The classical bit-security and quantum bit-security are hence defined as $\lambda_C := 0.292\beta$ and $\lambda_Q := 0.265\beta$, respectively.

Going into more details, BKZ with block-size β is assumed to return a basis of the input lattice whose vectors have Gram-Schmidt norms that decrease geometrically.

Heuristic 1 (Geometric Series Assumption (GSA), [Sch03]) *Let $\mathcal{L}$ be a full-rank lattice with basis $\mathbf{B} \in \mathbb{R}^{r\times r}$ with rows $\mathbf{b}_i$'s. After execution of BKZ with block-size β on $\mathbf{B}$, the norms of the Gram-Schmidt vectors satisfy*

$$\|\widetilde{\mathbf{b}}_i\| = \delta_\beta^{-2(i-1)+r} \cdot \mathrm{Vol}(\mathcal{L})^{1/r} \ , \quad \textit{where} \quad \delta_\beta = \left(\frac{(\pi\beta)^{1/\beta}\cdot\beta}{2\pi\mathrm{e}}\right)^{1/(2(\beta-1))} .$$

The GSA has been backed-up by extensive experimental results [GN08,Che13,AGVW17,YD17,BSW18], and it has been found to be very accurate for large block-sizes.

Key recovery In this scenario, the attacker is given the basis B_{NTRU} of $\mathcal{L}_{\mathrm{NTRU}}$, and aims at finding the first dn rows of $\mathbf{B}_{\mathbf{F},\mathbf{g}}$. For this purpose, it runs BKZ in block-size β on $\mathrm{M}(\mathbf{B}_{\mathrm{NTRU}})$. We consider that it wins if it finds any of these dn rows.

The $\mathcal{L}_{\mathrm{NTRU}}$ has volume q^d and rank $d(n+1)$. Following the specifications of our scheme, we expect all rows of $\widetilde{\mathbf{B}_{\mathbf{F},\mathbf{g}}}$ to have essentially the same Euclidean norm around $\mathrm{GS\_SLACK}\cdot q^{1/(n+1)}$. As this is less than the expected norm $\sqrt{\frac{d(n+1)}{2\pi\mathrm{e}}}q^{1/(n+1)}$ of a shortest non-zero vector of a lattice with the same volume as $\mathcal{L}_{\mathrm{NTRU}}$, we expect these vectors to be the shortest non-zero vectors in $\mathcal{L}_{\mathrm{NTRU}}$. We hence rely on the GSA-based analysis from [ADPS16,AGVW17] to quantify the hardness of finding these unexpectedly short vectors with BKZ. Concretely, BKZ with block-size β is expected to find such a vector when

$$\mathrm{GS\_SLACK}\cdot\sqrt{\frac{\beta}{d(n+1)}} \leq \delta_\beta^{2\beta-d(n+1)}. \tag{5}$$

Signature forgery A signature forgery corresponds to finding a point of $\mathcal{L}_{\mathrm{NTRU}}$ at distance at most ρ from a vector $\mathbf{c}$ of the ambient space derived from the message and the signature salt. As ρ is significantly above the norms $\mathrm{GS\_SLACK} \cdot q^{1/(n+1)}$ of the vectors of $\mathcal{L}_{\mathrm{NTRU}}$ corresponding to the secret key, this is an instance of the Approximate Closest Vector Problem (CVP). To solve it, the first step is to apply BKZ to $\mathrm{M}(\mathbf{B}_{\mathrm{NTRU}})$ with a large block-size β (to be determined below). Then one takes $\mathbf{c}$ and uses Babai's nearest plane algorithm to shorten it, using the obtained BKZ-reduced basis $\mathbf{B}$ of $\mathcal{L}_{\mathrm{NTRU}}$. The resulting vector $\mathbf{t}' := \mathbf{t} - \mathbf{b}$ for some $\mathbf{b} \in \mathcal{L}_{\mathrm{NTRU}}$ can be written $\mathbf{t}' = \sum_i t'_i \widetilde{\mathbf{b}}_i$ with $|t'_i| \leq 1/2$ for every i, and hence, under the GSA:

$$\|\mathbf{t}'\|^2 \leq \frac{1}{4} q^{\frac{2}{n+1}} \cdot \sum_{i \leq d(n+1)} (\delta_\beta^2)^{-2(i-1)+d(n+1)}.$$

This attack strategy can be improved in two ways, and these improvements actually lead to the best known attacks against ModFalcon. The first improvement consists in modifying Babai's nearest-plane algorithm so that it calls an exact CVP solver for the lattice spanned by the first β vectors of the BKZ-reduced basis $\mathbf{B}$. The sieve algorithms [BDGL16,Laa15] can be (heuristically) adapted for this, for the same cost as solving SVP in dimension β. As a result, this adaptation of Babai's algorithm is not more costly than the call to BKZ in block-size β. On the other hand, it allows to find a vector $\mathbf{t}'$ satisfying:

$$\|\mathbf{t}'\|^2 \leq \left((\delta_\beta^2)^{d(n+1)} + \frac{1}{4} \sum_{i=\beta+1}^{d(n+1)} (\delta_\beta^2)^{-2(i-1)+d(n+1)} \right) q^{\frac{2}{n+1}}.$$

The first summand corresponds to the (squared) expected distance between the lattice spanned by the first β vectors of $\mathbf{B}$ and a random vector in its span. It is the same as the expected minimum for that lattice, and we can hence use the GSA to estimate it. For the values of β that we consider, the first term is larger than the second one, and we will just delete the second one, resulting in the inequality

$$\|\mathbf{t}'\| \leq \delta_\beta^{d(n+1)} q^{\frac{1}{n+1}}.$$

The second improvement relies on the observation that one can consider a subset of the rows of $\mathrm{M}(\mathbf{B}_{\mathrm{NTRU}})$ rather than the full matrix $\mathrm{M}(\mathbf{B}_{\mathrm{NTRU}})$. There does not seem to be an advantage considering another subset then those obtained by erasing the first k rows, for $k \leq nd$ (as h is essentially uniform modulo q). The volume remains q^d, but the dimension decreases to $d(n+1) - k$. In the equation above, this allows to decrease the term $\delta_\beta^{d(n+1)}$ to $\delta_\beta^{d(n+1)-k}$, at the expense of increasing the term $q^{1/(n+1)}$ to $q^{d/(d(n+1)-k)}$. Overall, we obtain the following success condition for a signature forgery:

$$\rho \geq \min_{k \leq dn} \left(\delta_\beta^{d(n+1)-k} q^{\frac{d}{d(n+1)-k}} \right). \tag{6}$$

Interestingly, optimizing over k does not help for Falcon but does for ModFalcon.

Combinatorial and hybrid attacks Described in [How07], these attacks combine lattice reduction and a meet-in-the-middle approach, and can be used to recover a line of the trapdoor basis (which we again assume is a win for an attacker). The idea is to decompose a line as $\mathbf{b} = (\mathbf{g}, \mathbf{f}_1, \ldots, \mathbf{f}_n) = \mathbf{s}_1 + \mathbf{s}_2$ (where $\mathbf{s}_1$, resp. $\mathbf{s}_2$, is the vector of first, resp. last, coordinates). The meet-and-the-middle phase makes a guess for $\mathbf{s}_2$ and checks for collision using that plausible candidates for $(\mathbf{s}_1, \mathbf{s}_2)$ should satisfy $\mathbf{s}_1 \cdot \begin{bmatrix} 1 \\ \mathbf{h}^t \end{bmatrix} \approx \mathbf{s}_2 \cdot \begin{bmatrix} 1 \\ \mathbf{h}^t \end{bmatrix}$, as $\mathbf{b}$ is short. For our parameters, it will be less efficient than a direct attack without improvements.

To further improve the efficiency, the attacker can perform lattice reduction on a suitably chosen sublattice of $\mathcal{L}_{\mathrm{NTRU}}$. The efficiency of the approach is then obtained by assessing the trade-off between the dimension of the lattice and the size of the remaining "guess-space" (see, e.g.,[BCLv19,HPS$^+$17]). However, even with such trade-offs, these types of attacks do not affect our scheme, as we now explain.

Hybrid attacks are mostly considered for secret keys with entries in $\{-1, 0, 1\}$, because their efficiency decreases drastically when the size of the entries increases: indeed, the space of possible guesses grows exponentially with the size of the entries. Also, in some NTRU-based encryption schemes, it can happen that the keys are really "sparse" (with a lot of coefficients equal to 0), to increase the scheme efficiency. The efficiency of hybrid attacks also relies crucially on the sparseness of the vector to be recovered, as it also conditions the size of the guess space.

One could argue that the speed-up due to rotations and the number of lines could play a role. However it can be seen (borrowing for example the analysis in [BCLv19]) that the speed-up is overall negligible due to the lack of sparseness of the keys, which means we can focus on discussing sparseness. In particular, it can be observed in our scheme that the vector which completes $\left[\mathbf{g}^t \middle| -\mathbf{F}\right]$ into a basis is sparser than the others, since $d(n-2)$ entries are 0's while the others rows are Gaussians. Yet the remaining $2d$ entries are not small, nor are the corresponding fields elements sparse in general.

Other attacks As observed in [ABD16,KF17], when the modulus q is sufficiently large compared to the magnitudes of the NTRU secret key coefficients, the attack on the key based on lattice reduction recovers the secret key better than described in Section 4.2. In the case of the NTRU signatures, the magnitudes of the secret key coefficients are of the order of $\sqrt{q}$, which is far too large compared to q for the attack to be applicable. More generally, this attack was considered irrelevant by all the NTRU-based submissions to the NIST standardization process. The same applies to our concrete proposal, which relies on a fairly small modulus q.

Finally, as the signing algorithm of Falcon is admittedly rather complex, an implementation thereof could potentially be vulnerable to timing attacks. Nevertheless, an efficient constant-time implementation was recently proposed [PRR19]. In our case, this existing code can be reused inside the ModFalcon signature algorithm. This would allow to obtain a constant time implementation of the latter.

Conclusion After a detailed analysis of known attacks, the best attacks we found are based on lattice reduction (Section 4.2). The success condition of the best known attacks for key recovery and signature forgery are given by (5) and (6), respectively.

The security levels implied by these best known attacks are given in Table 3. These are computed by light python scripts, available at `https://gofile.io/?c=ANXatH`.

Table 3. Bit security estimates. β is the BKZ blocksize, k is the optimal sublattice dimension in (6) and λ_Q is the *quantum* bit security level.

Scheme	Key recovery (5)			Signature forgery (6)			
	β	λ_Q	λ_C	β	k	λ_Q	λ_C
Falcon-512	504	134	147	411	1024	109	120
Falcon-1024	998	264	291	952	2048	252	277
ModFalcon-2-512	717	190	209	658	1293	174	192

4.3 Implementation and performance

Table 4. Performance comparison between Falcon and ModFalcon. $|\mathsf{vk}|$ and $|\mathsf{sig}|$ denote the size in bytes of the public key (exactly) and signature (on average), respectively. All the schemes use the same modulus $q = 12289$.

The first table is for the signature-recovery mode.

Scheme	n	d	$\lvert\mathsf{vk}\rvert$	$\lvert\mathsf{sig}\rvert$	λ_Q	λ_C
Falcon-512	1	512	897	658	109	120
Falcon-1024	1	1024	1793	1274	252	277
ModFalcon-2-512	2	512	1792	972	174	192

The second table is for the key-recovery mode.

Scheme	n	d	$\lvert\mathsf{vk}\rvert$	$\lvert\mathsf{sig}\rvert$	λ_Q	λ_C
Falcon-512	1	512	28	1276	109	120
Falcon-1024	1	1024	63	2508	252	277
ModFalcon-2-512	2	512	940	1438	174	192

We implemented a complete proof-of-concept implementation of ModFalcon in Python: it can be found at `https://gofile.io/?c=YnCEPM`. The sizes given in the tables above are directly obtained from this implementation, which is provided as supplementary material. As it is an un-optimized Python implementation, the running times are not meaningful. In practice, an optimized implementation would obtain timings close to those of Falcon: the running times of the signature and verification procedures grow with n, but n remains small in our case.

We observe that ModFalcon-2-512 achieves quantum bit security above 128, but has signature size significantly smaller than that of Falcon-1024.

5 Public Key Encryption and Encapsulation

In this section we describe an extension of the well-known NTRU encryption schemes to NTRU lattices of larger ranks. There are several approaches regarding the management of the noise involved in the scheme. However, as ultimately this design did not allow us to improve upon other NTRU-type schemes [ZCH+19,BCLv19]), we merely stay at a high-level description. The following subsections deal with security matters and discuss parameters choices.

5.1 A public key encryption scheme

Let $n \geq m$ be integers, and $q > p$ be coprime odd integers. Let D_F and D_G be distributions over $R^{n\times n}$ and $R^{n\times m}$, respectively. We assume that the infinity norms of the samples are bounded by some constants, i.e., for any $\mathbf{F}=(f_{ij})\hookleftarrow D_F$ and $\mathbf{G}=(g_{ij})\hookleftarrow D_G$, we have $\|f_{ij}\|_\infty \leq B_F$ and $\|g_{ij}\|_\infty \leq B_G$ for some integers B_F and B_G.

Key-Generation

repeat
  Sample $\mathbf{F} \hookleftarrow D_F$
until $\mathbf{F}$ is invertible mod q and mod p
Sample $\mathbf{G} \hookleftarrow D_G$
$\mathbf{H} \leftarrow p\mathbf{F}^{-1}\mathbf{G} \bmod q$ in $R_q^{n\times m}$
return $(\mathsf{pk} = \mathbf{H}, \mathsf{sk} = \mathbf{F})$

Encryption: $(\mathsf{pk} = \mathbf{H}, (\mathbf{r}, \mathbf{e}) \in R^m \times R^n)) \rightarrow \mathbf{c} \in R_q^n$

return $\mathbf{c}^t \leftarrow \mathbf{H}\mathbf{r}^t + \mathbf{e}^t \bmod q$

Decryption: $((\mathsf{pk} = \mathbf{H}, \mathsf{sk} = \mathbf{F}), \mathbf{c}) \rightarrow (\mathbf{r}, \mathbf{e}) \in R_q^m \times R_p^n$

$\mathbf{d}^t \leftarrow \mathbf{F}\mathbf{c}^t \bmod q$
$\mathbf{z}^t \leftarrow \mathbf{d}^t \bmod p$
$\mathbf{e}^t \leftarrow \mathbf{F}^{-1}\mathbf{z}^t \bmod p$
$\mathbf{r}^t \leftarrow (\mathbf{H}^t\mathbf{H})^{-1}\mathbf{H}^t(\mathbf{c}^t - \mathbf{e}^t) \bmod q$
return $(\mathbf{r}, \mathbf{e})$

Key generation is defined in similar to ModFalcon key generation. Encryption and decryption are matrix generalizations of NTRU encryption and decryption. Note that decryption can be accelerated by computing $\mathbf{H}^+ = (\mathbf{H}^t\mathbf{H})^{-1}\mathbf{H}^t \bmod q$ and $\mathbf{F}^{-1} \bmod p$ in the key generation algorithm and storing them along with $\mathbf{F}$.

Arguments for security The security relies on adaptations of two well-known assumptions. The first states that it is impossible for an adversary to distinguish the public key from a uniformly random matrix modulo q. This is sometimes referred to as "the NTRU

assumption" or "the DSPR (decisional short polynomial ratio) assumption" and can actually be shown to hold for certain ranges of parameters [SS11] when $n = m = 1$. Outside these parameter ranges, it is assumed to hold (see for example [LTV12,SHRS17]). The second assumption is known as Module-LWE, which (informally) states that an attacker is unable to distinguish between $(\mathbf{H}, \mathbf{H}\mathbf{r}^t + \mathbf{e}^t \bmod q) \in R_q^{n\times m} \times R_q^n$ and a pair of random elements [BGV12,LS15]. Under these assumptions, the presented scheme provides pseudorandom ciphertexts, for random plaintexts (it is OW-CPA).

5.2 Toward a practical key encapsulation mechanism

By [SXY18], a pseudorandom PKE can be generically converted into an adaptively secure KEM in the QROM, if the PKE decryption never fails and the valid ciphertexts are few in the ciphertext space. Moreover, this conversion is tight. Our scheme enjoys these two properties for suitably chosen parameters. Informally, perfect correctness is obtained by requiring that $\|\mathbf{r}\|_\infty$ and $\|\mathbf{e}\|_\infty$ are no larger than $(p-1)/2$ and that p is small compared to q. Sparseness also follows from the largeness of q compared to $\|\mathbf{r}\|_\infty$ and $\|\mathbf{e}\|_\infty$.

Lemma 5.1 (Correctness). *If* $\|\mathbf{e}\|_\infty, \|\mathbf{r}\|_\infty \leq (p-1)/2$ *and* $(nB_F + pmB_G)d(p-1) < q$*, then the scheme is perfectly correct.*

Proof. The decryption algorithm first computes $\mathbf{d}^t = \mathbf{F}\cdot\mathbf{c}^t \bmod q \in R_q^n$. For $\mathbf{c}$ generated using the encryption algorithm, we have, modulo q:

$$\mathbf{d}^t = \mathbf{F}(\mathbf{e}^t + \mathbf{H}\mathbf{r}^t) = \mathbf{F}\mathbf{e}^t + p\mathbf{G}\mathbf{r}^t.$$

Now, thanks to the assumption, we obtain that the equality above also holds over R since:

$$\begin{aligned}\|\mathbf{F}\mathbf{e}^t + p\mathbf{G}\mathbf{r}^t\|_\infty &\leq \|\mathbf{F}\mathbf{e}^t\|_\infty + p\|\mathbf{G}\mathbf{r}^t\|_\infty \\ &\leq n\cdot dB_F(p-1)/2 + p\cdot m\cdot dB_G(p-1)/2 \\ &= (nB_G + pmB_G)d(p-1)/2 < q/2.\end{aligned}$$

Therefore, modulo p, we have that $\mathbf{z}^t = \mathbf{d}^t = \mathbf{F}\mathbf{e}^t + p\mathbf{G}\mathbf{r}^t = \mathbf{F}\mathbf{e}^t$. From this, we obtain that $\mathbf{e}^t := \mathbf{F}^{-1}\cdot\mathbf{z}^t \bmod p$ in fact holds over R. The vector $\mathbf{r}$ can then be recovered by Gaussian elimination.

Parameter choices There are several ways to instantiate the final KEM. The first consideration is the moduli p and q. With the standard choice of $p = 3$, perfect correctness can be achieved even for q's that are used by other schemes (such as KYBER [SAB+19] and NewHope [PAA+19]) as long as the entries of $\mathbf{F}$ and $\mathbf{G}$ are sufficiently small. The reason to opt for such q's is that they allow a fast multiplication based on the Number Theoretic Transform, as already mentioned when presenting ModFalcon.

A second consideration is the distribution of $\mathbf{F}$ and $\mathbf{G}$. By taking them sufficiently large (and taking a prime q satisfying $q = 3 \bmod 8$, contrarily to the above), one can guarantee that the distribution of the public key is within exponentially small statistical distance from uniform (as showed in appendix). Nevertheless, this forces to take a much larger q, and makes the scheme quite uncompetitive in terms of performance. For

this reason, we would rather recommend taking $\mathbf{F}$ and $\mathbf{G}$ with very small entries, as is typically done for practical NTRU encryption. There is no known attack for such parameter choices, for NTRU-based NIST candidates.

The next concern is the way in which the vectors $\mathbf{r}$ and $\mathbf{e}$ are sampled for each key encapsulation. A possibility is to rely on discrete Gaussians of a large enough standard deviation. However, because of the wide support, using them while guaranteeing perfect correctness via Lemma 5.1 would require a large modulus q. To avoid this caveat, the usual choice of NTRU variants is to have $\mathbf{r}$ and $\mathbf{e}$ take values in a small interval, such as $\{-1, 0, 1\}$, and possibly require that there are few non-zero entries. Another approach is to choose $\mathbf{e}$ deterministically, by rounding the random vector $\mathbf{Hr}^t$ to the "closest multiple" of some other modulus γ, like NTRU Prime [BCLv19] and SABER [DKRV19]. With careful tuning of all parameters, we obtained ciphertexts of bitlengths roughly equivalent to those of NTRU Prime, NTRU [ZCH+19] and several other lattice-based round-2 NIST key encapsulation schemes, for similar security levels. As we did not manage to make them strictly advantageous from some angle, we considered that this did not justify a full description.

References

ABD16. Martin R. Albrecht, Shi Bai, and Léo Ducas. A subfield lattice attack on overstretched NTRU assumptions - cryptanalysis of some FHE and graded encoding schemes. In Matthew Robshaw and Jonathan Katz, editors, *CRYPTO 2016, Part I*, volume 9814 of *LNCS*, pages 153–178. Springer, Heidelberg, August 2016.

ADPS16. Erdem Alkim, Léo Ducas, Thomas Pöppelmann, and Peter Schwabe. Post-quantum key exchange - A new hope. In Thorsten Holz and Stefan Savage, editors, *USENIX Security 2016*, pages 327–343. USENIX Association, August 2016.

AGVW17. Martin R. Albrecht, Florian Göpfert, Fernando Virdia, and Thomas Wunderer. Revisiting the expected cost of solving uSVP and applications to LWE. In Tsuyoshi Takagi and Thomas Peyrin, editors, *ASIACRYPT 2017, Part I*, volume 10624 of *LNCS*, pages 297–322. Springer, Heidelberg, December 2017.

BAA+19. Nina Bindel, Sedat Akleylek, Erdem Alkim, Paulo S. L. M. Barreto, Johannes Buchmann, Edward Eaton, Gus Gutoski, Juliane Kramer, Patrick Longa, Harun Polat, Jefferson E. Ricardini, and Gustavo Zanon. qTESLA. Technical report, National Institute of Standards and Technology, 2019. available at `https://csrc.nist.gov/projects/post-quantum-cryptography/round-2-submissions`.

BCLv19. Daniel J. Bernstein, Chitchanok Chuengsatiansup, Tanja Lange, and Christine van Vredendaal. NTRU Prime. Technical report, National Institute of Standards and Technology, 2019. available at `https://csrc.nist.gov/projects/post-quantum-cryptography/round-2-submissions`.

BDE+18. Jonathan Bootle, Claire Delaplace, Thomas Espitau, Pierre-Alain Fouque, and Mehdi Tibouchi. LWE without modular reduction and improved side-channel attacks against BLISS. In Thomas Peyrin and Steven Galbraith, editors, *ASIACRYPT 2018, Part I*, volume 11272 of *LNCS*, pages 494–524. Springer, Heidelberg, December 2018.

BDF+11. Dan Boneh, Özgür Dagdelen, Marc Fischlin, Anja Lehmann, Christian Schaffner, and Mark Zhandry. Random oracles in a quantum world. In Dong Hoon Lee and Xiaoyun Wang, editors, *ASIACRYPT 2011*, volume 7073 of *LNCS*, pages 41–69. Springer, Heidelberg, December 2011.

BDGL16. Anja Becker, Léo Ducas, Nicolas Gama, and Thijs Laarhoven. New directions in nearest neighbor searching with applications to lattice sieving. In Robert Krauthgamer, editor, *27th SODA*, pages 10–24. ACM-SIAM, January 2016.

BDK+18. Joppe W. Bos, Léo Ducas, Eike Kiltz, Tancrède Lepoint, Vadim Lyubashevsky, John M. Schanck, Peter Schwabe, Gregor Seiler, and Damien Stehlé. CRYSTALS - kyber: A cca-secure module-lattice-based KEM. In *2018 IEEE European Symposium on Security and Privacy, EuroS&P 2018, London, United Kingdom, April 24-26, 2018*, pages 353–367, 2018.

BGV12. Zvika Brakerski, Craig Gentry, and Vinod Vaikuntanathan. (Leveled) fully homomorphic encryption without bootstrapping. In Shafi Goldwasser, editor, *ITCS 2012*, pages 309–325. ACM, January 2012.

BSW18. Shi Bai, Damien Stehlé, and Weiqiang Wen. Measuring, simulating and exploiting the head concavity phenomenon in BKZ. In Thomas Peyrin and Steven Galbraith, editors, *ASIACRYPT 2018, Part I*, volume 11272 of *LNCS*, pages 369–404. Springer, Heidelberg, December 2018.

CDPR16. Ronald Cramer, Léo Ducas, Chris Peikert, and Oded Regev. Recovering short generators of principal ideals in cyclotomic rings. In Marc Fischlin and Jean-Sébastien Coron, editors, *EUROCRYPT 2016, Part II*, volume 9666 of *LNCS*, pages 559–585. Springer, Heidelberg, May 2016.

CDW17. Ronald Cramer, Léo Ducas, and Benjamin Wesolowski. Short stickelberger class relations and application to ideal-SVP. In Jean-Sébastien Coron and Jesper Buus Nielsen, editors, *EUROCRYPT 2017, Part I*, volume 10210 of *LNCS*, pages 324–348. Springer, Heidelberg, April / May 2017.

CG05. Michael Coglianese and Bok-Min Goi. MaTRU: A new NTRU-based cryptosystem. In Subhamoy Maitra, C. E. Veni Madhavan, and Ramarathnam Venkatesan, editors, *INDOCRYPT 2005*, volume 3797 of *LNCS*, pages 232–243. Springer, Heidelberg, December 2005.

Che13. Yuanmi Chen. *Réduction de réseau et sécurité concrète du chiffrement complètement homomorphe*. PhD thesis, 2013.

DKL+18. Léo Ducas, Eike Kiltz, Tancrède Lepoint, Vadim Lyubashevsky, Peter Schwabe, Gregor Seiler, and Damien Stehlé. Crystals-dilithium: A lattice-based digital signature scheme. *IACR Trans. Cryptogr. Hardw. Embed. Syst.*, 2018(1):238–268, 2018.

DKRV18. Jan-Pieter D'Anvers, Angshuman Karmakar, Sujoy Sinha Roy, and Frederik Vercauteren. Saber: Module-LWR based key exchange, CPA-secure encryption and CCA-secure KEM. In Antoine Joux, Abderrahmane Nitaj, and Tajjeeddine Rachidi, editors, *AFRICACRYPT 18*, volume 10831 of *LNCS*, pages 282–305. Springer, Heidelberg, May 2018.

DKRV19. Jan-Pieter D'Anvers, Angshuman Karmakar, Sujoy Sinha Roy, and Frederik Vercauteren. SABER. Technical report, National Institute of Standards and Technology, 2019. available at `https://csrc.nist.gov/projects/post-quantum-cryptography/round-2-submissions`.

DLP14. Léo Ducas, Vadim Lyubashevsky, and Thomas Prest. Efficient identity-based encryption over NTRU lattices. In Palash Sarkar and Tetsu Iwata, editors, *ASIACRYPT 2014, Part II*, volume 8874 of *LNCS*, pages 22–41. Springer, Heidelberg, December 2014.

DP16. Léo Ducas and Thomas Prest. Fast fourier orthogonalization. In Sergei A. Abramov, Eugene V. Zima, and Xiao-Shan Gao, editors, *Proceedings of the ACM on International Symposium on Symbolic and Algebraic Computation, ISSAC 2016, Waterloo, ON, Canada, July 19-22, 2016*, pages 191–198. ACM, 2016.

FKT+19. Pierre-Alain Fouque, Paul Kirchner, Mehdi Tibouchi, Alexandre Wallet, and Yang Yu. Uprooting the falcon tree? *IACR Cryptology ePrint Archive*, 2019:1180, 2019.

GN08. Nicolas Gama and Phong Q. Nguyen. Predicting lattice reduction. In Nigel P. Smart, editor, *EUROCRYPT 2008*, volume 4965 of *LNCS*, pages 31–51. Springer, Heidelberg, April 2008.

GPV08. Craig Gentry, Chris Peikert, and Vinod Vaikuntanathan. Trapdoors for hard lattices and new cryptographic constructions. In Richard E. Ladner and Cynthia Dwork, editors, *40th ACM STOC*, pages 197–206. ACM Press, May 2008.

GZB+19. Oscar Garcia-Morchon, Zhenfei Zhang, Sauvik Bhattacharya, Ronald Rietman, Ludo Tolhuizen, Jose-Luis Torre-Arce, Hayo Baan, Markku-Juhani O. Saarinen, Scott Fluhrer, Thijs Laarhoven, and Rachel Player. Round5. Technical report, National Institute of Standards and Technology, 2019. available at `https://csrc.nist.gov/projects/post-quantum-cryptography/round-2-submissions`.

Ham19. Mike Hamburg. Three Bears. Technical report, National Institute of Standards and Technology, 2019. available at `https://csrc.nist.gov/projects/post-quantum-cryptography/round-2-submissions`.

HHP+03. Jeffrey Hoffstein, Nick Howgrave-Graham, Jill Pipher, Joseph H. Silverman, and William Whyte. NTRUSIGN: Digital signatures using the NTRU lattice. In Marc Joye, editor, *CT-RSA 2003*, volume 2612 of *LNCS*, pages 122–140. Springer, Heidelberg, April 2003.

How07. Nick Howgrave-Graham. A hybrid lattice-reduction and meet-in-the-middle attack against NTRU. In Alfred Menezes, editor, *CRYPTO 2007*, volume 4622 of *LNCS*, pages 150–169. Springer, Heidelberg, August 2007.

HPS98. Jeffrey Hoffstein, Jill Pipher, and Joseph H. Silverman. NTRU: A ring-based public key cryptosystem. In *Algorithmic Number Theory, Third International Symposium, ANTS-III, Portland, Oregon, USA, June 21-25, 1998, Proceedings*, pages 267–288, 1998.

HPS+17. Jeffrey Hoffstein, Jill Pipher, John M. Schanck, Joseph H. Silverman, William Whyte, and Zhenfei Zhang. Choosing parameters for NTRUEncrypt. In Helena Handschuh, editor, *CT-RSA 2017*, volume 10159 of *LNCS*, pages 3–18. Springer, Heidelberg, February 2017.

KF17. Paul Kirchner and Pierre-Alain Fouque. Revisiting lattice attacks on overstretched NTRU parameters. In Jean-Sébastien Coron and Jesper Buus Nielsen, editors, *EUROCRYPT 2017, Part I*, volume 10210 of *LNCS*, pages 3–26. Springer, Heidelberg, April / May 2017.

Laa15. Thijs Laarhoven. *Search problems in cryptography*. PhD thesis, 2015.

LDK+19. Vadim Lyubashevsky, Léo Ducas, Eike Kiltz, Tancrède Lepoint, Peter Schwabe, Gregor Seiler, and Damien Stehlé. CRYSTALS-DILITHIUM. Technical report, National Institute of Standards and Technology, 2019. available at `https://csrc.nist.gov/projects/post-quantum-cryptography/round-2-submissions`.

LLJ+19. Xianhui Lu, Yamin Liu, Dingding Jia, Haiyang Xue, Jingnan He, Zhenfei Zhang, Zhe Liu, Hao Yang, Bao Li, and Kunpeng Wang. LAC. Technical report, National Institute of Standards and Technology, 2019. available at `https://csrc.nist.gov/projects/post-quantum-cryptography/round-2-submissions`.

LPR10. Vadim Lyubashevsky, Chris Peikert, and Oded Regev. On ideal lattices and learning with errors over rings. In Henri Gilbert, editor, *EUROCRYPT 2010*, volume 6110 of *LNCS*, pages 1–23. Springer, Heidelberg, May / June 2010.

LPR13. Vadim Lyubashevsky, Chris Peikert, and Oded Regev. A toolkit for ring-LWE cryptography. In Thomas Johansson and Phong Q. Nguyen, editors, *EUROCRYPT 2013*, volume 7881 of *LNCS*, pages 35–54. Springer, Heidelberg, May 2013.

LPSW19. Changmin Lee, Alice Pellet-Mary, Damien Stehlé, and Alexandre Wallet. An LLL algorithm for module lattices. In Steven D. Galbraith and Shiho Moriai, editors, *ASIACRYPT 2019, Part II*, volume 11922 of *LNCS*, pages 59–90. Springer, Heidelberg, December 2019.

LS15. Adeline Langlois and Damien Stehlé. Worst-case to average-case reductions for module lattices. *Designs, Codes and Cryptography*, 75(3):565–599, Jun 2015.

LTV12. Adriana López-Alt, Eran Tromer, and Vinod Vaikuntanathan. On-the-fly multiparty computation on the cloud via multikey fully homomorphic encryption. In Howard J. Karloff and Toniann Pitassi, editors, *44th ACM STOC*, pages 1219–1234. ACM Press, May 2012.

Lyu12. Vadim Lyubashevsky. Lattice signatures without trapdoors. In David Pointcheval and Thomas Johansson, editors, *EUROCRYPT 2012*, volume 7237 of *LNCS*, pages 738–755. Springer, Heidelberg, April 2012.

NIS16. NIST. Submission requirements and evaluation criteria for the post-quantum cryptography standardization process, 2016. `https://csrc.nist.gov/CSRC/media/Projects/Post-Quantum-Cryptography/documents/call-for-proposals-final-dec-2016.pdf`.

PAA+19. Thomas Poppelmann, Erdem Alkim, Roberto Avanzi, Joppe Bos, Léo Ducas, Antonio de la Piedra, Peter Schwabe, Douglas Stebila, Martin R. Albrecht, Emmanuela Orsini, Valery Osheter, Kenneth G. Paterson, Guy Peer, and Nigel P. Smart. NewHope. Technical report, National Institute of Standards and Technology, 2019. available at `https://csrc.nist.gov/projects/post-quantum-cryptography/round-2-submissions`.

PFH+17. Thomas Prest, Pierre-Alain Fouque, Jeffrey Hoffstein, Paul Kirchner, Vadim Lyubashevsky, Thomas Pornin, Thomas Ricosset, Gregor Seiler, William Whyte, and Zhenfei Zhang. FALCON. Technical report, National Institute of Standards and Technology, 2017. available at `https://csrc.nist.gov/projects/post-quantum-cryptography/round-1-submissions`.

PFH+19. Thomas Prest, Pierre-Alain Fouque, Jeffrey Hoffstein, Paul Kirchner, Vadim Lyubashevsky, Thomas Pornin, Thomas Ricosset, Gregor Seiler, William Whyte, and Zhenfei Zhang. FALCON. Technical report, National Institute of Standards and Technology, 2019. available at `https://csrc.nist.gov/projects/post-quantum-cryptography/round-2-submissions`.

PHS19. Alice Pellet-Mary, Guillaume Hanrot, and Damien Stehlé. Approx-SVP in ideal lattices with pre-processing. In Yuval Ishai and Vincent Rijmen, editors, *EUROCRYPT 2019, Part II*, volume 11477 of *LNCS*, pages 685–716. Springer, Heidelberg, May 2019.

PP19. Thomas Pornin and Thomas Prest. More efficient algorithms for the NTRU key generation using the field norm. In Dongdai Lin and Kazue Sako, editors, *PKC 2019, Part II*, volume 11443 of *LNCS*, pages 504–533. Springer, Heidelberg, April 2019.

Pre17. Thomas Prest. Sharper bounds in lattice-based cryptography using the Rényi divergence. In Tsuyoshi Takagi and Thomas Peyrin, editors, *ASIACRYPT 2017, Part I*, volume 10624 of *LNCS*, pages 347–374. Springer, Heidelberg, December 2017.

PRR19. Thomas Prest, Thomas Ricosset, and Melissa Rossi. Simple, fast and constant-time gaussian sampling over the integers for FALCON. *Second PQC Standardization Conference*, 2019.

RSW18. Miruna Rosca, Damien Stehlé, and Alexandre Wallet. On the ring-LWE and polynomial-LWE problems. In Jesper Buus Nielsen and Vincent Rijmen, editors, *EUROCRYPT 2018, Part I*, volume 10820 of *LNCS*, pages 146–173. Springer, Heidelberg, April / May 2018.

SAB+19. Peter Schwabe, Roberto Avanzi, Joppe Bos, Léo Ducas, Eike Kiltz, Tancrède Lepoint, Vadim Lyubashevsky, John M. Schanck, Gregor Seiler, and Damien Stehlé. CRYSTALS-KYBER. Technical report, National Institute of Standards and Technology, 2019. available at `https://csrc.nist.gov/projects/post-quantum-cryptography/round-2-submissions`.

Sch87. Claus-Peter Schnorr. A hierarchy of polynomial time lattice basis reduction algorithms. *Theor. Comput. Sci.*, 53:201–224, 1987.

Sch03. Claus-Peter Schnorr. Lattice reduction by random sampling and birthday methods. In *STACS 2003, 20th Annual Symposium on Theoretical Aspects of Computer Science, Berlin, Germany, February 27 - March 1, 2003, Proceedings*, pages 145–156, 2003.

SE94. Claus-Peter Schnorr and M. Euchner. Lattice basis reduction: Improved practical algorithms and solving subset sum problems. *Math. Program.*, 66:181–199, 1994.

SHRS17. John M. Schanck, Andreas Hulsing, Joost Rijneveld, and Peter Schwabe. NTRU-HRSS-KEM. Technical report, National Institute of Standards and Technology, 2017. available at `https://csrc.nist.gov/projects/post-quantum-cryptography/round-1-submissions`.

SS11. Damien Stehlé and Ron Steinfeld. Making NTRU as secure as worst-case problems over ideal lattices. In Kenneth G. Paterson, editor, *EUROCRYPT 2011*, volume 6632 of *LNCS*, pages 27–47. Springer, Heidelberg, May 2011.

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

SXY18. Tsunekazu Saito, Keita Xagawa, and Takashi Yamakawa. Tightly-secure key-encapsulation mechanism in the quantum random oracle model. In Jesper Buus Nielsen and Vincent Rijmen, editors, *EUROCRYPT 2018, Part III*, volume 10822 of *LNCS*, pages 520–551. Springer, Heidelberg, April / May 2018.

TW19. Mehdi Tibouchi and Alexandre Wallet. One bit is all it takes: A devastating timing attack on bliss's non-constant time sign flips. *IACR Cryptology ePrint Archive*, 2019:898, 2019.

YD17. Yang Yu and Léo Ducas. Second order statistical behavior of LLL and BKZ. In Carlisle Adams and Jan Camenisch, editors, *SAC 2017*, volume 10719 of *LNCS*, pages 3–22. Springer, Heidelberg, August 2017.

ZCH+19. Zhenfei Zhang, Cong Chen, Jeffrey Hoffstein, William Whyte, John M. Schanck, Andreas Hulsing, Joost Rijneveld, Peter Schwabe, and Oussama Danba. NTRUEncrypt. Technical report, National Institute of Standards and Technology, 2019. available at `https://csrc.nist.gov/projects/post-quantum-cryptography/round-2-submissions`.

A Almost uniformity of ModFalcon's verification keys

The purpose of this section is to extend the proof of uniformity of NTRU public keys from [SS11] to larger n. We show that $\mathbf{F}^{-1}\mathbf{g}^t \bmod q$ is pseudorandom as long as the entries are of standard deviation essentially $q^{1/(n+1)}$, which is the case in our scheme. In fact, we obtain a more general result, as we are able to also handle matrices for the "$\mathbf{g}$ component". For $n \geq m$, $\mathbf{F} \in R^{n\times n}$ invertible modulo q and $\mathbf{G} \in R^{n\times m}$, our result essentially states that if the entries in $\mathbf{F}, \mathbf{G}$ are discrete Gaussians of standard deviation essentially $q^{m/(n+m)}$, then $\mathbf{F}^{-1}\mathbf{G} \bmod q$ is pseudorandom. This can be seen as a general Leftover Hash Lemma over number rings handling also matrices.

More precisely, the statement is proved only for primes $q = 3 \bmod 8$. The reason is that our proof technique relies on an inclusion-exclusion argument to handle sublattices of some $\mathcal{L}_{\mathrm{NTRU}}$; these sublattices come in two layers, one corresponding to the ideal factors of q and one to enumerate all the possible nullspaces of the matrix $\mathbf{F}^{-1}\mathbf{G}$ in $R_q^{n\times m}$. Such q's have a small splitting pattern in R, which means that the "ideal factor" layer of the inclusion-exclusion can be managed by a probability overestimate. This overestimate becomes too loose when the splitting pattern involves more ideals. We leave it for further work to overcome the increased technicality of the proof technique to handle all q's.

For $s > 0$, we let $D_{R^{n\times m},s}$ denote the distribution over $R^{n\times m}$ whose entries are distributed from $D_{R,s}$, and $D_{\mathrm{GL}_n(R,q),s}$ be the restriction of $D_{R^{n\times n},s}$ to the set $\mathrm{GL}_n(R,q)$ of matrices in $R^{n\times n}$ that are invertible modulo q. Lastly, we define the distribution $\mathcal{E}_s$ as the distribution of $\mathbf{F}^{-1}\mathbf{G} \bmod q$ when $\mathbf{F}$ is sampled from $D_{\mathrm{GL}_n(R,q),s}$ and G is sampled from $D_{R^{n\times m},s}$. The following theorem is the main result of this section.

Theorem A.1. *Let K be a cyclotomic number field of degree d and maximal order R. Let $n \geq m \geq 1$. Let q be a prime integer which factors as $qR = \mathfrak{p}_1\mathfrak{p}_2$, where the $\mathfrak{p}_i$'s have algebraic norm $q^{d/2}$. For $s \geq 2dq^{m/(n+m)+2/(d(n+m))}$, we have:*

$$\Delta\left(\mathcal{E}_s, U(R_q^{n\times m})\right) \leq 2^{-\Omega(d)}.$$

A.1 Additional notations and lemmas

The expectation of a (function of a) random vector X is denoted by $\mathbb{E}[f(X)]$. If two distributions D_1 and D_2 are over the same countable support Ω, their statistical distance is

$$\Delta(D_1, D_2) = \frac{1}{2}\sum_{\omega\in\Omega} |D_1(\omega) - D_2(\omega)| = \sum_{\omega: D_1(\omega) > D_2(\omega)} [D_1(\omega) - D_2(\omega)].$$

For any countable set S and function f defined over S, we let $f(S) = \sum_{s\in S} f(s)$. For any lattice $\mathcal{L}$, the Poisson summation formula gives

$$\rho_s(\mathcal{L}) = (\mathrm{Vol}\,\mathcal{L})^{-1} \cdot s^d \cdot \rho_{1/s}(\mathcal{L}^*).$$

We will need the next lemma, which essentially motivates the definition of the smoothing parameter.

Lemma A.2. *Let $\mathcal{L}$ be a rank d lattice, and $\epsilon \in (0,1)$. For any $s \geq \eta_\epsilon(\Lambda)$, we have $\rho_s(\mathcal{L}) \in [1 \pm \epsilon] \cdot s^d (\mathrm{Vol}\,\mathcal{L})^{-1}$.*

The proof will use results and tools from algebraic number theory that are now considered as standard. We refer to e.g. [LPR10] and [RSW18] for further details on ideals in number rings, and provide below only some notation and lemmas that will be used. The discriminant of a number field K is written Δ_K. The algebraic norm of an ideal I is denoted by $\mathrm{N}(I)$.

Lemma A.3 ([LPR13, Th. 7.2]). *For any ideal $I \subset R$ and $s > 0$, we have:*

$$\rho_{1/s}(I) \leq \max(1, \mathrm{N}(I)^{-1} s^{-d})(1 + 2^{-2d}).$$

The proof also uses some results on (finitely generated) modules over number rings (see also [LS15]). The dual of a R-module $\mathcal{M} \subset K^n$ is $\mathcal{M}^\vee := \{\mathbf{y} \in K^n : \forall \mathbf{x} \in \mathcal{M}, \mathrm{Tr}(\langle \mathbf{x}, \mathbf{y} \rangle_K) \in \mathbb{Z}\}$, where Tr denotes the field trace (equivalently, the trace of the multiplication matrix). By linearity of the trace, we see that $\mathrm{Tr}(\langle \mathbf{x}, \mathbf{y} \rangle_K) = d\langle \mathbf{x}, \mathbf{y}^\star \rangle$, where we implicitely consider coefficient vectors in the right inner product. This shows that $\mathcal{L}(\mathcal{M})^* = \mathcal{L}((d\mathcal{M}^\vee)^\star)$.

For any $\mathbf{v} \in R^n$, any ideal I of R and any $k \geq 1$, we will consider modules lattices of the form

$$\Lambda_I^{\perp k}(\mathbf{v}) = \{X \in R^{k \times n} : X \cdot \mathbf{v}^t \equiv 0 \bmod I\},$$

The next result is known for other types of module lattices ([SS13,RSW18]). Its proof is standard and given for the sake of completeness.

Lemma A.4. *Let K be a number field with maximal order R, and I be an ideal in R. Let $k, n \geq 1$. Then, for any $\mathbf{v} \in (R/I)^n \setminus \{0\}$, we have*

$$\Lambda_I^{\perp k}(\mathbf{v})^\vee = \big((I^\vee/R^\vee) \cdot \mathbf{v} + (R^\vee)^n\big)^k.$$

Proof. It suffices to prove the result for $k = 1$, as $\Lambda_I^{\perp k}(\mathbf{v})$ is the direct sum of k copies of $\Lambda_I^{\perp 1}(\mathbf{v})$. Let $L = (I^\vee/R^\vee) \cdot \mathbf{v} + (R^\vee)^{1 \times n}$. We proceed by double inclusion, starting with $L \subseteq \Lambda_I^{\perp 1}(\mathbf{v})^\vee$. Let $\mathbf{x} \in \Lambda_I^{\perp 1}(\mathbf{v})$ and $\mathbf{y} \in L$. There are $\lambda \in I^\vee/R^\vee$ and $\mathbf{r} \in (R^\vee)^{1 \times n}$ such that $\mathbf{y} = \lambda \cdot \mathbf{v} + \mathbf{r}$. Therefore, we have $\langle \mathbf{x}, \mathbf{y} \rangle = \lambda \langle \mathbf{x}, \mathbf{v} \rangle + \langle \mathbf{x}, \mathbf{r} \rangle$. By definition of $\Lambda_I^{\perp 1}(\mathbf{v})^\vee$, we have $\langle \mathbf{x}, \mathbf{v} \rangle \in I$ so that the trace of the first term is an integer. The second term is in $R^\vee$ so it also has an integer trace, and the first inclusion is proven.

By duality, the second inclusion is equivalent to $L^\vee \subseteq \Lambda_I^{\perp 1}(\mathbf{v})$. Let $\mathbf{x} \in L^\vee$. A vector $\mathbf{y}$ in its dual is of the form $\mathbf{y} = \lambda \cdot \mathbf{v} + \mathbf{r}$ with $\lambda \in I^\vee/R^\vee$ and $\mathbf{r} \in (R^\vee)^{1 \times n}$. Consider $\lambda = 0$ and let $\mathbf{r}$ vary across vectors with all but one entries being 0: the fact that the trace of $\langle \mathbf{x}, \mathbf{y} \rangle$ is an integer implies that $\mathbf{x} \in R^n$. Now, consider $\mathbf{r} = \mathbf{0}$ and let λ vary in $I^\vee/R^\vee$: the integrality of the trace implies that $\langle \mathbf{x}, \mathbf{v} \rangle$ is in I.

A.2 Gaussian mass of matrices invertible modulo q

We now show that the Gaussian mass of $\mathrm{GL}_n(R, q)$ is essentially the full Gaussian mass of $R^{n \times n}$, for q prime such that the prime factor ideals of qR have large algebraic norms. This will prove useful as the trapdoor matrix $\mathbf{F}$ is sampled in $R^{n \times n}$ conditioned on being invertible modulo q, More precisely, we obtain the following result.

Theorem A.5. *Let K be a cyclotomic number field of degree d and maximal order R. Let $n \geq 1$ and let q an unramified prime integer such that each prime factor $\mathfrak{p}$ of qR has algebraic norm $2^{\Omega(d)}$. Assume that $s \geq 2(\Delta_K \mathrm{N}(\mathfrak{p})^{\frac{n-1}{2n-1}})^{1/d}$. Then*

$$\rho_s(R^{n \times n} \setminus \mathrm{GL}_n(R, q)) \leq 2d \frac{s^{n^2 d}}{\Delta_K^{n^2} \mathrm{N}(\mathfrak{p})} \leq \frac{4d}{\mathrm{N}(\mathfrak{p})} \rho_s(R^{n \times n}).$$

Observe that being non-invertible modulo q is equivalent to being non-invertible modulo at least one prime factor of qR. Let $\mathfrak{p}$ be such a prime ideal. By the union bound and the fact that for our choice of R the prime factors of qR are isometric, we have

$$\rho_s(R^{n \times n} \setminus \mathrm{GL}_n(R, q)) \leq d \cdot \rho_s(R^{n \times n} \setminus \mathrm{GL}_n(R, \mathfrak{p})).$$

At this stage, it is worth recalling that $R/\mathfrak{p}$ is a finite field of characteristic q. Being non-invertible in $(R/\mathfrak{p})^{n\times n}$ is hence equivalent to having a non-zero vector in the kernel. Since any two non-zero colinear vectors generate the same line, we can write

$$\begin{aligned}\rho_s(R^{n\times n}\setminus \mathrm{GL}_n(R,q)) &\le d\cdot\frac{1}{\mathrm{N}(\mathfrak{p})-1}\cdot\sum_{\mathbf{v}\in(R/\mathfrak{p})^n\setminus\mathbf{0}}\rho_s(\Lambda_{\mathfrak{p}}^{\perp n}(\mathbf{v}))\\ &\le d\cdot\frac{\mathrm{N}(\mathfrak{p})^n-1}{\mathrm{N}(\mathfrak{p})-1}\cdot\mathbb{E}_{\mathbf{v}\leftarrow U((R/\mathfrak{p})^n\setminus\mathbf{0})}\left[\rho_s(\Lambda_{\mathfrak{p}}^{\perp n}(\mathbf{v}))\right].\end{aligned}$$

We are hence reduced to studying $\mathbb{E}[\rho_s(\Lambda_{\mathfrak{p}}^{\perp n}(\mathbf{v}))]$ for $\mathbf{v}$ uniformly distributed in $(R/\mathfrak{p})^n\setminus\mathbf{0}$. For this, we will use the Poisson summation formula, which will involve the dual of $\Lambda_{\mathfrak{p}}^{\perp n}(\mathbf{v})$. The following technical lemma holds for an arbitrary number field K and an arbitrary prime ideal $\mathfrak{p}$. Note that it implies Theorem A.5.

Lemma A.6. *Let K be a number field of degree d and maximal order R. Let $n,k\ge 1$ and $q\ge 2$ be an unramified prime integer. Let $\mathfrak{p}\subseteq R$ be a prime factor of qR. For any $s\ge 2(\Delta_K\,\mathrm{N}(\mathfrak{p})^{\frac{k-1}{n+k-1}})^{1/d}$, we have:*

$$\left|\frac{\mathbb{E}_{\mathbf{v}\leftarrow U((R/\mathfrak{p})^n\setminus\mathbf{0})}\left[\rho_s(\Lambda_{\mathfrak{p}}^{\perp k}(\mathbf{v}))\right]}{\left(s^{nd}/\,\mathrm{N}(\mathfrak{p})\right)^k}\;-\;1\right|\;\le\;8nk\cdot 2^{-d}.$$

Proof. Let $\mathbf{v}\in(R/\mathfrak{p})^n\setminus\mathbf{0}$. The lattice $\Lambda_{\mathfrak{p}}^{\perp k}(\mathbf{v})$ is the direct sum of k copies of $\Lambda_{\mathfrak{p}}^{\perp 1}(\mathbf{v})$. The latter has index $\mathrm{N}(\mathfrak{p})$ in R^n (using the fact that $\mathfrak{p}$ is prime and $\mathbf{v}$ is non-zero). Hence $\det(\Lambda_{\mathfrak{p}}^{\perp k}(\mathbf{v}))=\mathrm{N}(\mathfrak{p})^k$. By the Poisson summation formula and Lemma A.4, we have:

$$\rho_s(\Lambda_{\mathfrak{p}}^{\perp k}(\mathbf{v}))=\Big(\frac{s^{nd}}{\mathrm{N}(\mathfrak{p})}\Big)^k\cdot\rho_{1/s}\big((\mathfrak{p}^\vee/R^\vee)^{k\times 1}\cdot\mathbf{v}^t+(R^\vee)^{k\times n}\big).$$

We focus on the expectation of the latter Gaussian sum, and aim at showing that it is very close to $\rho_{1/s}(R^\vee)^{nk}\approx 1$. We have:

$$\begin{aligned}&\left|\mathbb{E}_{\mathbf{v}}\left[\rho_{1/s}\big((\mathfrak{p}^\vee/R^\vee)^{k\times 1}\cdot\mathbf{v}^t+(R^\vee)^{k\times n}\big)\right]-\rho_{1/s}(R^\vee)^{nk}\right|\\ &\qquad=\sum_{\mathbf{x}\in(\mathfrak{p}^\vee/R^\vee)^k\setminus\mathbf{0}}\mathbb{E}_{\mathbf{v}}\Big[\prod_{\substack{i\in[k]\\ j\in[n]}}\rho_{1/s}(x_iv_j+R^\vee)\Big]\\ &\qquad\le\frac{nk}{\mathrm{N}(\mathfrak{p})^n-1}\sum_{\substack{\mathbf{x}\in(\mathfrak{p}^\vee/R^\vee)^k\\ \mathbf{v}\in(R/\mathfrak{p})^n\\ x_1,v_1\neq 0}}\prod_{\substack{i\in[k]\\ j\in[n]}}\rho_{1/s}(x_iv_j+R^\vee).\end{aligned}$$

For the inequality, we used the facts that that $(\mathfrak{p}^\vee/R^\vee)^k\setminus\mathbf{0}=\bigcup_{i\in[k]}\{\mathbf{x}\in(\mathfrak{p}^\vee/R^\vee)^k\,:\,x_i\neq 0\}$, $(R/\mathfrak{p})^n\setminus\mathbf{0}=\bigcup_{j\in[n]}\{\mathbf{v}\in(R/\mathfrak{p})^n\,:\,v_j\neq 0\}$, and that each set in these unions has the same Gaussian mass.

We now provide some intuition on how the sum above will be handled. We aim at separating the variables andswapping the order of the sum and the product. However, the function being summed is not a product of functions of independent variables. Indeed, in the sum over $\mathbf{x}$ and $\mathbf{v}$, there are $n+k$ independent variable (over $\mathfrak{p}^\vee/R^\vee$

and $R/\mathfrak{p}$, which are two representations of the same finite field $\mathbb{F}_{\mathrm{N}(\mathfrak{p})}$). On the other hand, the product consists of nk terms involving the non-independent variables $x_i v_j$. In what follows, we restrict the product to $i = 1$ or $j = 1$, to have $n + k - 1$ independent quadratic terms. Concretely, we write:

$$\begin{aligned}\prod_{\substack{i\in[k]\\ j\in[n]}} \rho_{1/s}(x_i v_j + R^\vee) &= \prod_{i=1 \text{ or } j=1} \rho_{1/s}(x_i v_j + R^\vee) \cdot \prod_{i,j>1} \rho_{1/s}(x_i v_j + R^\vee) \\ &\le \rho_{1/s}(R^\vee)^{(n-1)(k-1)} \cdot \prod_{i=1 \text{ or } j=1} \rho_{1/s}(x_i v_j + R^\vee).\end{aligned}$$

The inequality holds because the Gaussian sum of a lattice coset is maximized for the zero coset.Next, we apply a change of variable over the summand $(\mathbf{x}, \mathbf{v})$. Concretely, we (bijectively) map $(\mathbf{x}, \mathbf{v}) \in (\mathfrak{p}^\vee/R^\vee)^k \times (R/\mathfrak{p})^n$ with $x_1, v_1 \neq 0$ to $(\mathbf{x}', \mathbf{v}')$ with $x'_1 = x_1$, $x'_i = x_i v_1 \in \mathfrak{p}^\vee/R^\vee$ for $i > 1$ and $v'_j = v_j x_1 \in \mathfrak{p}^\vee/R^\vee$ for $j \ge 1$. Overall, we have

$$\begin{aligned}&\sum_{\substack{\mathbf{x}\in(\mathfrak{p}^\vee/R^\vee)^k\\ \mathbf{v}\in(R/\mathfrak{p})^n\\ x_1,v_1\neq 0}} \prod_{\substack{i\in[n]\\ j\in[k]}} \rho_{1/s}(x_i v_j + R^\vee) \\ &\le \rho_{1/s}(R^\vee)^{(n-1)(k-1)} \cdot \sum_{\substack{\mathbf{x}'\in(\mathfrak{p}^\vee/R^\vee)^n\\ \mathbf{v}'\in(\mathfrak{p}^\vee/R^\vee)^k\\ x'_1,v'_1\neq 0}} \prod_{1<i\le k} \rho_{1/s}(x'_i + R^\vee) \prod_{1\le j\le n} \rho_{1/s}(v'_j + R^\vee) \\ &= \rho_{1/s}(R^\vee)^{(n-1)(k-1)} \cdot \Big(\sum_{x\in\mathfrak{p}^\vee/R^\vee} \rho_{1/s}(x + R^\vee)\Big)^{n+k-2} \cdot \sum_{x\in\mathfrak{p}^\vee/R^\vee\setminus 0} \rho_{1/s}(x + R^\vee) \\ &\le \rho_{1/s}(R^\vee)^{(n-1)(k-1)} \cdot \rho_{1/s}(\mathfrak{p}^\vee)^{n+k-2} \cdot (\rho_{1/s}(\mathfrak{p}^\vee) - 1).\end{aligned}$$

From Lemma A.3, we have

$$\rho_{1/s}(\mathfrak{p}^\vee) \le \max(1, \Delta_K \,\mathrm{N}(\mathfrak{p}) s^{-d})(1 + 2^{-2d}).$$

Similarly, since $s \ge \eta_{2^{-2d}}(R)$, we have $\rho_{1/s}(R^\vee) \le 1 + 2^{-2d}$.We hence obtain:

$$\sum_{\substack{\mathbf{x}\in(\mathfrak{p}^\vee/R^\vee)^k\\ \mathbf{v}\in(R/\mathfrak{p})^n\\ x_1,v_1\neq 0}} \prod_{\substack{i\in[k]\\ j\in[n]}} \rho_{1/s}(x_i v_j + R^\vee) \;\le\; 2 \cdot \max\left(2^{-2d}, (\Delta_K \,\mathrm{N}(\mathfrak{p}) s^{-d})^{n+k-1}\right).$$

This leads to the following bound:

$$\begin{aligned}&\left|\mathbb{E}_{\mathbf{v}}\left[\rho_{1/s}((\mathfrak{p}^\vee/R^\vee)^{k\times 1} \cdot \mathbf{v}^t + (R^\vee)^{k\times n})\right] - 1\right| \\ &\qquad \le \frac{4nk}{\mathrm{N}(\mathfrak{p})^n} \cdot \max\left(2^{-2d}, (\Delta_K \,\mathrm{N}(\mathfrak{p}) s^{-d})^{n+k-1}\right) + 2nk2^{-d} \\ &\qquad \le 4nk \cdot \left(2^{-d} + (\Delta_K s^{-d})^{n+k-1} \,\mathrm{N}(\mathfrak{p})^{k-1}\right).\end{aligned}$$

The assumption on s gives the result.

A.3 Proof of Theorem A.1

For $\mathbf{H} \in R_q^{n\times m}$, the q-ary orthogonal lattice associated to $\mathbf{H}$ is

$$\Lambda_q^{\perp}(\mathbf{H}) := \left\{ \mathbf{x} \in R^{n+m} \,:\, \mathbf{x} \cdot \left[\frac{\mathbf{I}_m}{\mathbf{H}}\right] = \mathbf{0}_m \bmod q \right\},$$

and NTRU lattices corresponds to $m = 1$. We have $\operatorname{Vol} \Lambda_q^{\perp}(\mathbf{H}) = q^{md}$. In terms of lattices, taking the direct sum of n copies of this module amounts to considering

$$\Lambda_q^{\perp n}(\mathbf{H}) := \left\{ \mathbf{X} \in R^{n\times(n+m)} \,:\, \mathbf{X} \cdot \left[\frac{\mathbf{I}_m}{\mathbf{H}}\right] = \mathbf{0}_{n\times m} \bmod q \right\}$$

for which we have $\operatorname{Vol} \Lambda_q^{\perp,n}(\mathbf{H}) = q^{mnd}$. The next result holds for any finite number of copies.

Lemma A.7 ([LPR13, Th. 7.4]). *Let K be a cyclotomic number field of degree d and maximal order R. Let $m, n \geq 1$ and $q \geq 2$.*
Then $\eta_{2^{-\Omega(d)}}\big(\Lambda_q^{\perp}(\mathbf{H})\big) < 2dq^{m/(n+m)+2/(d(n+m))}$, except with probability at most $2^{-\Omega(d)}$ over the choice of $\mathbf{H} \leftarrow U(R_q^{n\times m})$.

We now show that for a fraction $1 - 2^{-\Omega(d)}$ of $\mathbf{H} \in R_q^{n\times m}$, the quantity $\mathcal{E}_s[\mathbf{H}] - |R_q^{n\times m}|^{-1}$ is smaller than $q^{-mnd} \cdot 2^{-\Omega(d)}$.We have:[7]

$$\begin{aligned}
\mathcal{E}_s[\mathbf{H}] &= \mathbb{P}_{(\mathbf{F},\mathbf{G})\leftarrow D_{\mathrm{GL}_n(R,q),s}\times D_{R^{n\times m},s}} \left[\big[\, \mathbf{G} | \mathbf{F} \,\big] \cdot \left[\frac{\mathbf{I}_m}{\mathbf{H}}\right] = 0 \bmod q \right] \\
&= \mathbb{P}_{(\mathbf{F},\mathbf{G})\leftarrow D_{R^{n\times(n+m)},s}} \left[\big[\, \mathbf{G} | \mathbf{F} \,\big] \cdot \left[\frac{\mathbf{I}_m}{\mathbf{H}}\right] = 0 | \mathbf{F} \in \mathrm{GL}_n(R,q) \right] \\
&\leq \frac{\mathbb{P}_{(\mathbf{F},\mathbf{G})\leftarrow D_{R^{n\times(n+m)},s}} \left[\big[\, \mathbf{G} | \mathbf{F} \,\big] \in \Lambda_q^{\perp n}(\mathbf{H}) \right]}{\mathbb{P}_{\mathbf{F}\leftarrow D_{R^{n\times n},s}} \left[\mathbf{F} \in \mathrm{GL}_n(R,q) \right]} \\
&= \frac{\rho_s\big(\Lambda_q^{\perp n}(\mathbf{H})\big)}{\rho_s(\mathrm{GL}_n(R,q)) \cdot \rho_s(R)^{n\cdot m}}.
\end{aligned}$$

We first consider the term $\rho_s(\Lambda_q^{\perp,n}(\mathbf{H}))$. To handle it, we use Lemma A.2 and Lemma A.7. We obtain that for a fraction $1 - 2^{-\Omega(d)}$ of $\mathbf{H} \in R_q^{n\times m}$, we have:

$$\big|\rho_s\big(\Lambda_q^{\perp n}(\mathbf{H})\big) - \frac{s^{d(m+n)}}{q^{mnd}}\big| \leq \frac{s^{d(m+n)}}{q^{mnd}} 2^{-\Omega(d)}.$$

Similarly, we have with Lemma A.2 again:

$$\big|\rho_s(R) - s^d\big| \leq s^d 2^{-\Omega(d)}$$

Finally, by Theorem A.5, we have:

$$\rho_s(\mathrm{GL}_n(R,q)) \geq s^{n^2 d}\left(1 - \frac{4d}{\mathrm{N}(\mathfrak{p})}\right).$$

This provides the result.

[7] Note that not much is lost in the inequality only if most matrices in $R^{n\times n}$ are invertible modulo q. This is the case for example when the rational integer q does not split too much in R.