Finding short integer solutions when the modulus is small

Léo Ducas[1,2], Thomas Espitau[3], and Eamonn W. Postlethwaite[1(✉)]

[1] CWI, Cryptology Group, Amsterdam, The Netherlands
ewp@cwi.nl
[2] Mathematical Institute, Leiden University, Leiden, The Netherlands
[3] PQShield, FR

Abstract. We present cryptanalysis of the inhomogenous short integer solution (ISIS) problem for anomalously small moduli q by exploiting the geometry of BKZ reduced bases of q-ary lattices.
We apply this cryptanalysis to examples from the literature where taking such small moduli has been suggested. A recent work [Espitau–Tibouchi–Wallet–Yu, CRYPTO 2022] suggests small q versions of the lattice signature scheme Falcon and its variant Mitaka. For one small q parametrisation of Falcon we reduce the estimated security against signature forgery by approximately 26 bits. For one small q parametrisation of Mitaka we successfully forge a signature in 15 seconds.

1 Introduction

The Short Integer Solution (SIS) problem is a computational problem that requires one to find a non-zero short vector in a lattice from a specific class of random lattices, known as random q-ary lattices. It was first introduced by Ajtai [1,20] along with reductions to worst-case lattice problems. The SIS problem has emerged as fundamental to lattice-based cryptography in both theory and practice. The above worst-case hardness reductions require the modulus q of the SIS instance to be significantly larger than ν, its Euclidean length bound on solutions. Given that concrete cryptographic design can be thought as a rarefied game of chicken, one sees this requirement on q ignored with parameters pushed towards maximal efficiency, and only constrained by documented cryptanalytic attacks.

The SIS problem has an inhomogeneous variant (ISIS), which often holds greater relevance in cryptanalytic contexts. In particular, forging a signature in lattice-based signature schemes commonly constitutes solving a particular ISIS instance.

One notable difference between the SIS and ISIS problems is that SIS becomes trivial once $q \leqslant \nu$. This can be demonstrated by the solution vector $q \cdot \mathbf{e}_1$ which is non-zero, in all random q-ary lattices and has length not greater than ν. However, $q \cdot \mathbf{e}_1$ is not a solution vector for ISIS provided its target $\mathbf{u}$ is not the zero vector. If $\mathbf{u} = \mathbf{0}$ then SIS and ISIS coincide. Nevertheless, ISIS also eventually becomes trivial as q decreases. For example, consider the ISIS instance $\mathbf{Ax} = \mathbf{u} \bmod q$, where $\mathbf{A} \in \mathbb{Z}_q^{n \times m}$ with $m \geqslant n$ and $\mathbf{u} \in \mathbb{Z}_q^n \setminus \{\mathbf{0}\}$ are chosen uniformly. Assuming that $\mathbf{A} = (\mathbf{A}_1 \,|\, \mathbf{A}_2)$ is such that $\mathbf{A}_1 \in \mathbb{Z}_q^{n \times n}$ is invertible over $\mathbb{Z}_q$, then $\mathbf{x}^t = (\mathbf{u}^t \mathbf{A}_1^{-t} \,|\, \mathbf{0})$ is non-zero and satisfies $\mathbf{Ax} = \mathbf{u} \bmod q$. The first n entries of $\mathbf{x}$ are uniform in $\mathbb{Z}_q^n \setminus \{\mathbf{0}\}$. By reducing the coefficients of $\mathbf{x}$ modulo q around 0 this solution has an expected square length of $nq^2/12$.[1] If this is sufficiently below ν^2 then it is likely that $\mathbf{x}$ is an ISIS solution. As such, the following approximate observations can be made:

1. for $\nu < q$, SIS and ISIS are similar problems and are both subject to lattice reduction attacks,
2. for $\nu \geqslant q$, SIS becomes trivial, but not necessarily ISIS, and
3. for $\nu \geqslant q\sqrt{n/12}$, ISIS also becomes trivial.

This naturally leads to the question of the security of ISIS in the regime where $\nu \in (q, q\sqrt{n/12})$. We do not expect ISIS to be hard as soon as $\nu < q\sqrt{n/12}$. Indeed, if ν is slightly below $q\sqrt{n/12}$ then one can attempt an Information Set Decoding (ISD) style of attack by randomising the columns of $\mathbf{A}_1$ and hoping that after a few trials the solution $\mathbf{x}^t = (\mathbf{u}^t \mathbf{A}_1^{-t} \,|\, \mathbf{0})$ reduced modulo q around $\mathbf{0}$ has a length slightly below its expectation.

While we find this gap in our cryptanalytic knowledge to be motivating in its own right, recent works have proposed ISIS parameters where $\nu > q$. In particular small q parameter sets for the lattice signatures Falcon [22] and its variant Mitaka [15] were proposed in [16], as well as for early parameters of a blind signature scheme [11].[2]

Contributions. For the regime $\nu \in (q, q\sqrt{n/12})$ we give an attack that is essentially a hybrid of the standard lattice reduction attack when $\nu < q$ with the ISD style attack when $\nu \approx q\,\sqrt{n/12}$. We improve this hybrid by exploiting the many short vectors given by a lattice sieve, providing in essence many ISD attempts with a single lattice reduction effort.

The core of the attack lies in noticing that after lattice reduction on a SIS lattice basis, we get a profile often referred to as having a Z-shape. This reduced basis has a number of q vectors as its first columns, $q \cdot \mathbf{e}_1, \dots, q \cdot \mathbf{e}_{\ell-1}$. By performing lattice sieving in the first projected

[1] For simplicity we consider the expected square length of the region $[-q/2, q/2]^n$.

[2] Given early communication with the authors the parameters of [11] were revised.

sublattice after the q vectors and lifting the discovered short vectors to the non-projected lattice, we reduce the first $\ell - 1$ entries of these short vectors modulo q around 0. The square length of a vector lifted in this manner is then the square length of the projected vector that lifted to it, plus the square length of its first $\ell - 1$ entries. These first $\ell - 1$ entries lie in $\lceil -(q-1)/2 \rceil, \ldots, \lfloor q/2 \rfloor$.

On the technical level, our attack requires us to model the Z-shape of a SIS lattice basis after lattice reduction and to count the number of integer points in the intersection of certain hyperballs and hypercubes. We achieve the first by using models that exist in the literature [2,14] and the latter via the efficient convolution of truncated theta series of $\mathbb{Z}$. Our model also assumes that the lifted entries of projected vectors are independently and uniformly distributed in $\lceil -(q-1)/2 \rceil, \ldots, \lfloor q/2 \rfloor$. Both our modelling of the Z-shape after reduction and our assumption on the uniformity of lifted entries are verified experimentally.

As another technical contribution we introduce an intermediate problem between SIS and ISIS that we call SIS*. The SIS* problem is identical to SIS except it disallows solutions that are $\mathbf{0} \bmod q$. In particular, if $\nu < q$ then SIS and SIS* coincide, and if $\nu \geqslant q$ then it allows us to argue about the homogeneous version of our attack. We give a generic reduction from ISIS to SIS* that increases the rank of the instance by one from m to $m+1$ and has a probability loss factor approximately mq. We then give a reduction using the SIS* attack we outline above that performs better: it still increments the rank but has a probability loss factor of $q/2$.

As a final contribution, we present the performance of our attack against several small q parameter sets suggested in [16]. For one parameter set suggested for Falcon we reduce the forgery security in the CoreSVP model [4] from 118 to 92 bits. For another parameter set suggested for Mitaka we reduced the BKZ blocksize required for forgery to $\beta \approx 40$ and implement the attack. We also explicitly state that we believe the attacks presented in this work are far from optimised. As such, we suggest that appealing to the practical security of ISIS instances with $\nu \geqslant q$ is approached with great care and, if possible, not at all. Our code is available at https://github.com/verdiverdiverdi/ISIS-small-q.

Application beyond SIS. An alternative interpretation of our attack can be made directly on SIS in systematic form; $\mathbf{A} = (\mathbf{I}_n \parallel \mathbf{A}_2) \in \mathbb{Z}_q^{n \times m}$ and one searches for a short $\mathbf{x} \in \mathbb{Z}^m \setminus \{\mathbf{0}\}$ such that $\mathbf{A}\mathbf{x} = 0 \bmod q$. The attacker may ignore carefully chosen rows or columns of $\mathbf{A}$. Ignoring columns is standard, and is equivalent to fixing entries of $\mathbf{x}$ to 0. Let $\mathbf{A}'$

denote $\mathbf{A}$ with some rows removed. Assuming solutions to the original SIS instance exist, we may find a short non-zero $\mathbf{x} \in \{0\}^n \times \mathbb{Z}^{m-n}$ such that $\mathbf{A}'\mathbf{x} = \mathbf{0} \bmod q$. Such an $\mathbf{x}$ does not guarantee $\mathbf{Ax} = \mathbf{0} \bmod q$. Due to the systematic form of $\mathbf{A}$ one can choose x_i for $1 \leqslant i \leqslant n$ to ensure $\mathbf{Ax} = \mathbf{0} \bmod q$, but these x_i may not be small. Our attack consists of using the many outputs of a sieve to brute force this approach, hoping that one solution has small enough x_i.

However, we find our geometric description preferable, because it also hints that this attack is not fundamentally limited to SIS type problems. For example, this attack would also be applicable to Hawk [13,9] if the parameter σ_{pk} (η in the specification version) was small. One would need to replace reduction modulo q by Babai lifting [7], and it might no longer be possible to calculate the success probability of the attack via theta series, but the principle remains valid. That is, if the adversary is given vectors that are shorter than what they can find using generic lattice reduction, then these vectors may be abused to improve attacks. The design of Hawk anticipated this, and set σ_{pk} precisely so that the vectors given to the attacker are not too short. To account for the variance in the length of sampled vectors, keys that would be too short are also rejected.

Related Work. The general principle of considering the Z-shape basis structure is not new and is discussed in the cryptanalysis of Dilithium [19]. Due to Dilithium's use of the ℓ_∞ version of ISIS the Z-shape structure did not lead to the best attacks. We take inspiration to consider the Z-shape in q-ary bases from [17,2,14].

Organisation of the paper. Section 2 introduces the necessary preliminaries. Section 3 outlines our attack and our model for it against SIS^*, and Section 3.5 describes how we mount it on ISIS. Section 4 outlines an optimisation to the basic attack. Section 5 provides experimental verification of two of the heuristics in our attack, namely our modelling of the Z-shape of a SIS lattice basis after reduction and the distribution of lifted entries. Section 6 discusses how our attack affects the security of recent cryptosystems in the $\nu \geqslant q$ regime.

Acknowledgements. The authors thank Damien Stehlé and Yang Yu for useful discussions, and the reviewers for their comments. The research of L. Ducas and E.W. Postlethwaite was supported by the ERC-StG-ARTICULATE project (no. 947821).

2 Preliminaries

2.1 Lattices and computational problems

Definition 1 (Lattice)**.** Let $\mathbf{B} \in \mathbb{R}^{d \times m}$ have linearly independent columns. A lattice $\mathbf{\Lambda}$ is the integer span of the columns of $\mathbf{B}$, $\{\mathbf{B} \cdot \mathbf{x} \colon \mathbf{x} \in \mathbb{Z}^m\}$. We say $\mathbf{B}$ is a basis for $\mathbf{\Lambda}$, $\mathbf{\Lambda}$ has dimension d and rank m, and $\mathbf{\Lambda}$ is full rank if $d = m$.

Definition 2 (Lattice Volume)**.** The volume of lattice $\mathbf{\Lambda}$ with basis $\mathbf{B}$ is $\mathrm{vol}(\mathbf{\Lambda}) = \sqrt{\det(\mathbf{B}^t\mathbf{B})}$.

If $\mathbf{\Lambda}$ is full rank then $\mathrm{vol}(\mathbf{\Lambda}) = \det(\mathbf{B})$. Note that the lattices generated by $\mathbf{B}, \mathbf{C} \in \mathbb{R}^{d \times m}$ are equal if and only if there exists $\mathbf{U} \in \mathrm{Gl}_m(\mathbb{Z})$ such that $\mathbf{B} = \mathbf{CU}$, and therefore volume is well defined.

Definition 3 (First minimum)**.** For lattice $\mathbf{\Lambda}$ we define

$$\lambda_1(\mathbf{\Lambda}) = \min_{\mathbf{x} \in \mathbf{\Lambda} \setminus \{\mathbf{0}\}} \|\mathbf{x}\|.$$

We can estimate the first minimum via the Gaussian heuristic which calculates the radius of ball whose volume equals that of the lattice.

Definition 4 (Gaussian heuristic)**.** Let $v_m = \pi^{m/2}/\Gamma(1 + m/2)$ be the volume of the m dimensional unit ball. For rank m lattice $\mathbf{\Lambda}$ we estimate $\lambda_1(\mathbf{\Lambda})$ as $\mathrm{gh}(\mathbf{\Lambda}) = v_m^{-1/m} \cdot \mathrm{vol}\left(\mathbf{\Lambda}\right)^{1/m} \approx \sqrt{m/2\pi e} \cdot \mathrm{vol}\left(\mathbf{\Lambda}\right)^{1/m}$.

Throughout we will consider projected sublattices, for which we need the following projections.

Definition 5 (Projections)**.** Given a lattice basis $\mathbf{B} \in \mathbb{R}^{d \times m}$ and an index $1 \leqslant i \leqslant m + 1$, define $\pi_{\mathbf{B},i} \colon \mathbb{R}^d \to \mathbb{R}^d$ as the orthogonal projection *against* $\mathrm{span}_{\mathbb{R}}(\mathbf{b}_1, \ldots, \mathbf{b}_{i-1})$ (i.e. *onto* $\mathrm{span}_{\mathbb{R}}(\mathbf{b}_1, \ldots, \mathbf{b}_{i-1})^{\perp}$).

For any $\mathbf{B}$ we have $\pi_{\mathbf{B},1} = \mathrm{Id}_{\mathbb{R}^d}$ and, if $m = d$, $\pi_{\mathbf{B},m+1}(\mathbb{R}^d) = \{\mathbf{0}\}$. If $\mathbf{x} \in \mathrm{span}_{\mathbb{R}}(\mathbf{b}_1, \ldots, \mathbf{b}_{i-1})$ and $\mathbf{y} \in \mathbb{R}^d$ then $\langle \mathbf{x}, \pi_{\mathbf{B},i}(\mathbf{y}) \rangle = 0$. If the basis is clear from context we write π_i. To compute these projections one can use the Gram–Schmidt basis related to a lattice basis $\mathbf{B}$.

Definition 6 (Gram–Schmidt)**.** Given a basis $\mathbf{B} \in \mathbb{R}^{d \times m}$ the Gram–Schmidt basis $\mathbf{B}^* \in \mathbb{R}^{d \times m}$ has pairwise orthogonal columns and is related to $\mathbf{B}$ via an upper triangular matrix $\mathbf{M}$ with a unit diagonal as $\mathbf{B} = \mathbf{B}^* {\cdot} \mathbf{M}$.

For $1 \leqslant i \leqslant m$ we have $\operatorname{span}_{\mathbb{R}}(\mathbf{b}_1, \dots, \mathbf{b}_i) = \operatorname{span}_{\mathbb{R}}(\mathbf{b}_1^*, \dots, \mathbf{b}_i^*)$ and for $\mathbf{x} \in \mathbb{R}^d$ one can calculate $\pi_{\mathbf{B},i}(\mathbf{x})$ via

$$\mathbf{x}_1 = \mathbf{x}, \quad \mathbf{x}_{j+1} = \mathbf{x}_j - \frac{\langle \mathbf{b}_j^*, \mathbf{x}_j \rangle}{\langle \mathbf{b}_j^*, \mathbf{b}_j^* \rangle} \mathbf{b}_j^*,$$

for $1 \leqslant j \leqslant i-1$ so that $\pi_{\mathbf{B},i}(\mathbf{x}) = \mathbf{x}_i$. Note that $\pi_i(\mathbf{b}_i) = \mathbf{b}_i^*$ and $\operatorname{vol}(\mathbf{\Lambda}) = \prod_i \|\mathbf{b}_i^*\|$. We use the following shorthand for projected lattices and lattice bases.

Definition 7 (Projected lattices and bases)**.** Given basis $\mathbf{B} \in \mathbb{R}^{d \times m}$ and $1 \leqslant \ell < r \leqslant m+1$ let $\mathbf{B}_{[\ell:r]} = (\pi_\ell(\mathbf{b}_\ell) | \cdots | \pi_\ell(\mathbf{b}_{r-1}))$ and $\mathbf{\Lambda}_{[\ell:r]} = \{\mathbf{B}_{[\ell:r]} \cdot \mathbf{x} \colon \mathbf{x} \in \mathbb{Z}^{r-\ell}\}$. If $r = m+1$ we write $\mathbf{B}_{[\ell]}$ and $\mathbf{\Lambda}_{[\ell]}$.

For example

$$\begin{aligned} \mathbf{B}_{[1]} &= \mathbf{B}, \\ \mathbf{B}_{[1:r]} &= (\mathbf{b}_1 \,|\, \cdots \,|\, \mathbf{b}_{r-1}), \\ \mathbf{B}_{[\ell:r]} &= (\mathbf{b}_\ell^* \,|\, \pi_\ell(\mathbf{b}_{\ell+1}) \,|\, \cdots \,|\, \pi_\ell(\mathbf{b}_{r-1})). \end{aligned}$$

One quantity of a basis we use throughout it its profile.

Definition 8 (Basis profile)**.** Given a basis $\mathbf{B} \in \mathbb{R}^{d \times m}$ its profile is the tuple $(\log\|\mathbf{b}_i^*\|)_{i=1}^m \in \mathbb{R}^m$.

Often we consider lattices of a particular form.

Definition 9 (q-ary lattice)**.** For some $q \in \mathbb{Z}_{>0}$ a rank m lattice $\mathbf{\Lambda}$ is a q-ary lattice if $q\mathbb{Z}^m \subseteq \mathbf{\Lambda} \subseteq \mathbb{Z}^m$.

Solving the following (I)SIS problems can be achieved by performing certain lattice reduction tasks over related q-ary lattices.

Definition 10 ((I)SIS)**.** Let $n \in \mathbb{N}$, m, q, ν be functions with domain $\mathbb{N}$ and $\mathbf{A} \leftarrow \mathsf{U}(\mathbb{Z}_q^{n \times m})$. We supress the dependence of m, q and ν on n.

The $\mathsf{SIS}_{n,m,q,\nu}$ problem is to find a vector $\mathbf{x} \in \mathbb{Z}^m \setminus \{\mathbf{0}\}$ such that $\|\mathbf{x}\| \leqslant \nu$ and $\mathbf{A}\mathbf{x} = \mathbf{0} \bmod q$.

Given also $\mathbf{u} \leftarrow \mathsf{U}(\mathbb{Z}_q^n)$ the $\mathsf{ISIS}_{n,m,q,\nu}$ problem is to find a vector $\mathbf{x} \in \mathbb{Z}^m \setminus \{\mathbf{0}\}$ such that $\|\mathbf{x}\| \leqslant \nu$ and $\mathbf{A}\mathbf{x} = \mathbf{u} \bmod q$.

If $\varphi_{\mathbf{A}} \colon \mathbb{Z}^m \to \mathbb{Z}_q^n$, $\mathbf{x} \mapsto \mathbf{A}\mathbf{x} \bmod q$ then $\ker(\varphi_{\mathbf{A}})$ forms a lattice called the kernel lattice of $\mathbf{A}$. Note that $\mathbf{A}$ is not in general a basis for this lattice.

Definition 11 (Kernel lattice and basis)**.** Let $\mathbf{\Lambda}_q^\perp(\mathbf{A}) = \{\mathbf{x} \in \mathbb{Z}^m \colon \mathbf{A}\mathbf{x} = \mathbf{0} \bmod q\}$ be the kernel lattice of $\mathbf{A}$. If one can permute the columns of $\mathbf{A}$ (which relates to a known entrywise permutation of the lattice $\mathbf{\Lambda}_q^\perp(\mathbf{A})$) to $(\mathbf{A}_1 \mid \mathbf{A}_2)$ such that $\mathbf{A}_1 \in \mathrm{Gl}_n(\mathbb{Z}_q)$ then one can form the basis

$$\mathbf{B}_{\mathbf{A}} = \begin{pmatrix} q\mathbf{I}_n & -\mathbf{A}_1^{-1}\mathbf{A}_2 \\ \mathbf{0} & \mathbf{I}_{m-n} \end{pmatrix},$$

of $\mathbf{\Lambda}_q^\perp(\mathbf{A})$.

Throughout, we will assume that such a permutation of the columns of $\mathbf{A}$ exists and note that for prime q, if $m = 2n$ then one exists with overwhelming probability in n. Note that $\mathbf{\Lambda}_q^\perp(\mathbf{A})$ has (full) rank m and volume q^n. Solving a $\mathsf{SIS}_{n,m,q,\nu}$ instance given by $\mathbf{A}$ is equivalent to finding $\mathbf{x} \in \mathbf{\Lambda}_q^\perp(\mathbf{A})$ with $\|\mathbf{x}\| \leqslant \nu$. Solving an $\mathsf{ISIS}_{n,m,q,\nu}$ instance given by $\mathbf{A}$ and $\mathbf{u}$ is equivalent to finding $\mathbf{b} \in \mathbb{Z}^m$ such that $\mathbf{A}\mathbf{b} = \mathbf{u} \bmod q$ and $\mathbf{x} \in \mathbf{\Lambda}_q^\perp(\mathbf{A})$ with $\|\mathbf{b} - \mathbf{x}\| \leqslant \nu$ since $\mathbf{A}(\mathbf{b} - \mathbf{x}) = \mathbf{u} - \mathbf{0} \bmod q$.

For clarity of exposition we introduce the SIS^* problem, which we give a reduction from ISIS to in Section 3.5. Trivial SIS solutions of the form $q \cdot \mathbf{e}_i$ are disallowed for SIS^*.

Definition 12 (SIS^*)**.** Let $n \in \mathbb{N}$, m, q, ν be functions with domain $\mathbb{N}$ and $\mathbf{A} \leftarrow \mathsf{U}(\mathbb{Z}_q^{n \times m})$. The $\mathsf{SIS}^*_{n,m,q,\nu}$ problem is to find a vector $\mathbf{x} \in \mathbb{Z}^m \setminus q\mathbb{Z}^m$ such that $\|\mathbf{x}\| \leqslant \nu$ and $\mathbf{A}\mathbf{x} = \mathbf{0} \bmod q$.

If $\nu < q$ then SIS and SIS^* are equivalent problems. We make use of a particular instance of theta functions on a lattice.

Definition 13 (Theta function of a lattice)**.** Given a lattice $\mathbf{\Lambda}$ we write

$$\Theta_{\mathbf{\Lambda}}(\tau) = \sum_{\mathbf{x} \in \mathbf{\Lambda}} e^{\pi i \tau \|x\|^2},$$

for any $\tau \in \mathbb{C}$ with $\operatorname{Im} \tau > 0$.

Letting $X = e^{\pi i \tau}$ and suppressing the dependence on τ we see that the coefficient of X^{j^2} in $\Theta_{\mathbf{\Lambda}}$ denotes the number of lattice vectors in $\mathbf{\Lambda}$ with length j. We have

$$\Theta_{\mathbb{Z}} = 1 + 2 \sum_{j \in \mathbb{Z}_{>0}} X^{j^2},$$

and note that $(\Theta_{\mathbb{Z}})^m = \Theta_{\mathbb{Z}^m}$ for $m \in \mathbb{N}$.

2.2 Reduction algorithms

Lattice reduction algorithms take as input a basis $\mathbf{B}_{\text{pre}} \in \mathbb{R}^{d \times m}$ of lattice $\mathbf{\Lambda}$, some control parameters, and upon termination output a pair $(\mathbf{B}, \mathbf{U}) \in \mathbb{R}^{d \times m} \times \mathrm{Gl}_m(\mathbb{Z})$ such that $\mathbf{B} = \mathbf{B}_{\text{pre}}\mathbf{U}$ and $\mathbf{B}$ is a "better" basis of $\mathbf{\Lambda}$. For our cryptanalytic purpose we are interested in the properties of the profile after lattice reduction. We consider the celebrated LLL [18] and BKZ [24] reduction algorithms and heuristics that describe their behaviour on "random" lattices, see [2] for a survey. The relevant information here is that LLL is an efficient form of prereduction, and that BKZ is parametrised by a parameter β, where BKZ-β finds short vectors in rank β lattices. As such, given that lattice sieves (see Section 2.3) are the most efficient method known to achieve this, the cost of BKZ grows exponentially in β.

An important quantity is the root Hermite factor which can be used to determine $\|\mathbf{b}_1\|$ of a basis after BKZ-β reduction.

Definition 14 (Root Hermite factor [10])**.** For $\beta \geqslant 50$ let

$$\delta_\beta = \left(\frac{\beta}{2\pi e} (\pi\beta)^{1/\beta} \right)^{1/2(\beta-1)} .$$

For smaller values of β the root Hermite factor δ_β is determined experimentally. After BKZ-β reduction on basis $\mathbf{B} \in \mathbb{R}^{d \times m}$ of lattice $\mathbf{\Lambda}$ we estimate $\|\mathbf{b}_1\| \approx \delta_\beta^{m-1} \cdot \mathrm{vol}\,(\mathbf{\Lambda})^{1/m}$. The other heuristic we use is the Geometric Series Assumption (GSA) [25]. This asserts that after lattice reduction the Gram–Schmidt norms decrease as a geometric series.

Definition 15 (Geometric Series Assumption)**.** After lattice reduction on basis $\mathbf{B} \in \mathbb{R}^{d \times m}$ there exists $\gamma \in (0, 1)$ such that for $1 \leqslant i \leqslant m$ we have $\|\mathbf{b}_i^*\| = \gamma^{i-1}\|\mathbf{b}_1\|$.

Given a basis $\mathbf{B}^{d \times m}$ of lattice $\mathbf{\Lambda}$, and assuming both $\|\mathbf{b}_1\| = \delta_\beta^{m-1} \cdot \mathrm{vol}\,(\mathbf{\Lambda})^{1/m}$ and the GSA after BKZ-β reduction, since $\mathrm{vol}(\mathbf{\Lambda}) = \prod_i \|\mathbf{b}_i^*\| = \|\mathbf{b}_1\|^{m-1} \cdot (\gamma \cdots \gamma^{m-1})$ we have $\gamma(\beta) = 1/\delta_\beta^2$.

For our final assumption we specialise to q-ary lattices, specifically those of the form $\mathbf{\Lambda}_q^\perp(\mathbf{A})$ for $\mathbf{A} \in \mathbb{Z}_q^{n \times m}$ with basis $\mathbf{B}_\mathbf{A}$, recall Definition 11. We note that for $1 \leqslant i \leqslant n$ we have $\pi_i(\mathbf{b}_i) = \mathbf{b}_i^* = q \cdot \mathbf{e}_i$. Under the root Hermite factor and GSA heuristics we assume that after BKZ-β reduction $\|\mathbf{b}_i^*\| = \delta_\beta^{m-1} \cdot \mathrm{vol}\,(\mathbf{\Lambda}) \cdot \gamma(\beta)^{i-1} = \delta_\beta^{m-2i+1} \cdot \mathrm{vol}\,(\mathbf{\Lambda})^{1/m}$ [4, Sec. 6.3]. However in Section 3, we assume that for $1 \leqslant i \leqslant n$, if $q < \delta_\beta^{m-2i+1} \cdot \mathrm{vol}\,(\mathbf{\Lambda})^{1/m}$ then the q vector remains and the decrease in the profile begins only bounded away from the first indices of the basis. This "Z-shape" phenomenon was first observed in [17] and is discussed more in [2] and [19, App. C]; we give more detail on our use of it in Section 3.

2.3 Lattice sieves

In this work a lattice sieve is an algorithm that takes as input a basis $\mathbf{B} \in \mathbb{R}^{d \times m}$ of lattice $\mathbf{\Lambda}$ and outputs in time exponential in m a constant fraction, which we can control, of vectors in $\{\mathbf{x} \in \mathbf{\Lambda} \setminus \{\mathbf{0}\} \colon \|\mathbf{x}\| \leqslant \sqrt{4/3} \cdot \mathrm{gh}(\mathbf{\Lambda})\}$. We call the set of vectors output by a sieve its *database*. One expects $(4/3)^{m/2}$ vectors in a sieve database and for their lengths to concentrate around $\sqrt{4/3} \cdot \mathrm{gh}(\mathbf{\Lambda})$.

In our attack we sieve in projected sublattices $\mathbf{\Lambda}_{[\ell]}$ of $\mathbf{\Lambda}$ determined by some $\mathbf{B}_{[\ell]}$ and then "lift" these vectors from $\mathbf{\Lambda}_{[\ell]}$ to $\mathbf{\Lambda}$ following [12]. Let $1 \leqslant \ell - 1 < m$ and m' be such that $\ell - 1 + m' = m$. If a sieve is performed on $\mathbf{B}_{[\ell]}$ then we have a database of short vectors $L \subset \mathbf{\Lambda}_{[\ell]}$. Let $\mathbf{w} \in \mathbf{\Lambda}$ be such that $\mathbf{w} = \mathbf{B}\mathbf{v}$ for some $\mathbf{v} \in \mathbb{Z}^m$. We may split $\mathbf{B} = (\mathbf{B}' \,|\, \mathbf{B}'')$ with $\mathbf{B}' \in \mathbb{R}^{d \times (\ell-1)}$ and $\mathbf{B}'' \in \mathbb{R}^{d \times m'}$ and $\mathbf{v}$ similarly. Then $\pi_\ell(\mathbf{w}) = \pi_\ell(\mathbf{B}'\mathbf{v}') + \pi_\ell(\mathbf{B}''\mathbf{v}'') = \mathbf{0} + \mathbf{B}_{[\ell]}\mathbf{v}''$. We see therefore that for $\mathbf{w}_{[\ell]} \in L$ with $\mathbf{w}_{[\ell]} = \mathbf{B}_{[\ell]}\mathbf{v}_{[\ell]}$, each lift of $\mathbf{w}_{[\ell]}$ to $\mathbf{\Lambda}$ is of the form $\mathbf{B}'\mathbf{v}' + \mathbf{B}''\mathbf{v}_{[\ell]}$ for $\mathbf{v}' \in \mathbb{Z}^{\ell-1}$.

The shortest $\mathbf{w} \in \mathbf{\Lambda}$ such that $\pi_\ell(\mathbf{w}) = \mathbf{w}_{[\ell]}$ is given by a particular choice of $\mathbf{v}'$. In our case, due to the geometry of our reduced bases, for every $\mathbf{w}_{[\ell]}$ we are able to find this choice of $\mathbf{v}'$. In particular, we consider bases $\mathbf{B}_\mathbf{A}$ of $\mathbf{\Lambda}_q^\perp(\mathbf{A})$ for $\mathbf{A} \in \mathbb{Z}_q^{n \times m}$. Let $\mathbf{B} = \mathbf{B}_\mathbf{A}\mathbf{U}$ be the basis after BKZ-β reduction and ℓ be maximal such that $\mathbf{B}' = (q \cdot \mathbf{e}_1 \,|\, \cdots \,|\, q \cdot \mathbf{e}_{\ell-1})$. In this case we have

$$\mathbf{B}'' = \begin{pmatrix} \mathbf{C} \\ \mathbf{D} \end{pmatrix}, \qquad \mathbf{B}_{[\ell]} = \begin{pmatrix} \mathbf{0} \\ \mathbf{D} \end{pmatrix},$$

with $\mathbf{C} \in \mathbb{Z}^{(\ell-1) \times m'}$ and $\mathbf{D} \in \mathbb{Z}^{m' \times m'}$. If $\mathbf{w}_{[\ell]} = \mathbf{B}_{[\ell]}\mathbf{v}_{[\ell]}$ then its shortest lift is some

$$\mathbf{w} = \mathbf{B}'\mathbf{v}' + \mathbf{B}''\mathbf{v}_{[\ell]} = \begin{pmatrix} qv'_1 \\ \vdots \\ qv'_{\ell-1} \\ 0 \\ \vdots \\ 0 \end{pmatrix} + \begin{pmatrix} \mathbf{C}\mathbf{v}_{[\ell]} \\ \\ \mathbf{B}_{[\ell]}\mathbf{v}_{[\ell]} \end{pmatrix}. \tag{1}$$

To find the shortest lift we reduce $\mathbf{C}\mathbf{v}_{[\ell]}$ modulo q centred around 0.

2.4 Elements of high dimensional geometry

Definition 16. We define the following geometric figures for $n \geqslant 1$,

i. $\mathsf{B}_n(r) = \{\mathbf{x} \in \mathbb{R}^n \colon \|\mathbf{x}\| \leqslant r\}$, the n dimensional ball of radius r, i.e. the dilatation of the ℓ_2 norm unit ball by a factor of r,

ii. $\mathsf{Cube}_n(q) = \begin{cases} \{-(q-1)/2, \ldots, (q-1)/2\}^n, & q \text{ odd}, \\ \{-(q-2)/2, \ldots, q/2\}^n, & q \text{ even}. \end{cases}$

Our $\mathsf{Cube}_n(q)$ represents the region of shortest reductions of $\mathbf{x} \in \mathbb{Z}^n$ modulo q, with an arbitrary choice made in the case of even q.

3 Attack on small modulus SIS

Our attack is based on two main ingredients: on the one hand a precise prediction of the geometry of BKZ reduced bases of q-ary lattices, and on the other calculating the number of integer points in the intersection of hyperballs and hypercubes in high dimensions.

3.1 On the Z-shape of BKZ reduced bases for q-ary lattices

The three zones of the Z-shape. As in the SIS problem we consider a uniform matrix $\mathbf{A} \in \mathbb{Z}_q^{n \times m}$ and its associated kernel lattice $\mathbf{\Lambda}_q^{\perp}(\mathbf{A})$. We refer to its basis $\mathbf{B}_{\mathbf{A}}$ as $\mathbf{B}$ and apply some amount of lattice reduction to it. Initially, the profile $(\ell_i)_{i=1}^m$ of the basis has $\ell_i = \log q$ for $i \in [n]$ and $\ell_i = 0$ for $i > n$, resulting in a "Z-shape", see Figure 1. The profile indices are divided into three distinct zones: Zone I, comprising of the q vectors with $\ell_i = \log q$, Zone II, the "slope", currently empty, and Zone III, the "flat tail", with $\ell_i = 0$.

LLL reduction. As lattice reduction is applied, starting with LLL, the profile may change, with the vector corresponding to the last vector in Zone I potentially having a projection shorter than q and the vector corresponding to the first index in Zone III potentially having a projection longer than 1. These indices are now part of Zone II, where $\ell_i \in (0, \log q)$. Additionally, we assume that the GSA applies to Zone II. The LLL algorithm is partially self-dual, reducing both the basis and the corresponding dual basis, resulting in all ℓ_i falling into these three distinct and ordered zones, as discussed in greater detail in [2, Sec. 4.3].

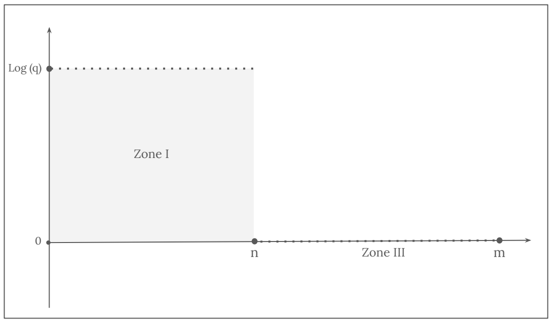


Fig. 1: Initial profile of basis **B**.

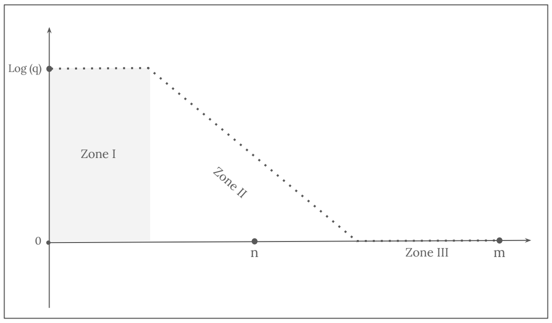


Fig. 2: Profile after some BKZ reduction.

BKZ profile. We then use the stronger lattice reduction algorithm BKZ. Unlike LLL, BKZ does not possess this partially self-dual property. However, if BKZ-β fails to find a vector of length shorter than q within the first β columns of the basis $\mathbf{B}$, it can be asserted that $\ell_1 = \log q$. It has been observed that BKZ-β reduction preserves the Z-shape and its three zones, up to a small "kink" just before Zone III [2], with the slope of Zone II decreasing according to the GSA, see Figure 2. This observation was first documented in [17]. Additionally, randomising $\mathbf{B}$ can remove the q vectors from the first n columns of the basis. Applying BKZ to such randomised bases is depicted in Figure 3. We note that the use of the q vectors is fundamental to our attack, so we do not randomise bases in this manner. The application of BKZ-β to bases of the form $\mathbf{B}_\mathbf{A}$, with and without randomisation, is discussed in more detail in [19, App. C]. In what follows we present a model for BKZ-β reduction on bases of the form $\mathbf{B}_\mathbf{A}$ that captures the aforementioned Z-shape phenomenon, similar to the model presented in [14, Heuristic. 2.8].

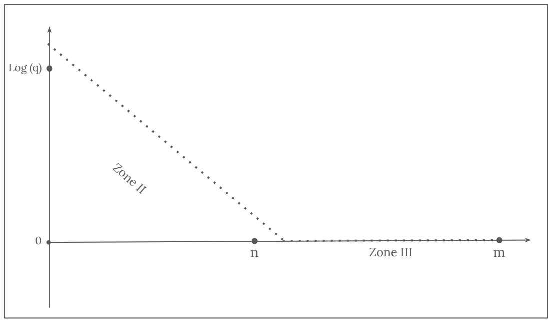


Fig. 3: Profile after rerandomisation and then some BKZ reduction.

A predictive model for BKZ profiles. Our model for predicting the basis profile after BKZ reduction is based on the volume invariance of a lattice under a change of basis, as in [19, App. C]. To determine the output profile, we make two assumptions:

i. the GSA holds in Zone II, with the slope determined solely by the BKZ blocksize β, specifically $\gamma(\beta) = 1/\delta_\beta^2$,
ii. despite not being a self-dual algorithm, upon completion BKZ reduction preserves Zones I, II, and III, and their order.

Using these assumptions, we construct a preliminary "extended" profile that has more indices than the rank of the lattice. Specifically, let n_q, n_{GSA}, and n_1 represent the number of vectors in Zone I, Zone II and Zone III, respectively. The input basis has $n_q = n$ and $n_1 = m - n$ and we use the GSA to determine n_{GSA}. Under the GSA after BKZ-β reduction and in a log scale, Zone II begins at index $n + 1$ with value $\log q - 2\log\delta_\beta$ and decreases by $-2\log\delta_\beta$ per subsequent index. This continues until the profile takes a value in the range $(2\log\delta(\beta), 0]$, which allows us to calculate n_{GSA} as $\lfloor \log q/(2\log\delta_\beta) \rfloor$. On a profile plot, Zone II therefore consists of the points $(n_q + i, \log q - 2i\log\delta_\beta)$ for $i \in \{1, \ldots, n_{\mathsf{GSA}}\}$. The global shape of the profile is quite accurate, but the resulting (logarithm of the) volume, that is to say, the sum of all the values in the three zones

$$\sum_{i=1}^{n_q} \log q + \sum_{i=1}^{n_{\mathsf{GSA}}} (\log q - 2i\log\delta_\beta) + \sum_{i=n_q+n_{\mathsf{GSA}}+1}^{m} 0$$
$$= (n + n_{\mathsf{GSA}})\log q - n_{\mathsf{GSA}}(n_{\mathsf{GSA}} + 1)\log\delta_\beta,$$

is not equal to that of the lattice; $n\log(q)$. What remains is to find the correct starting index of Zone II, i.e. some index smaller than n. To do

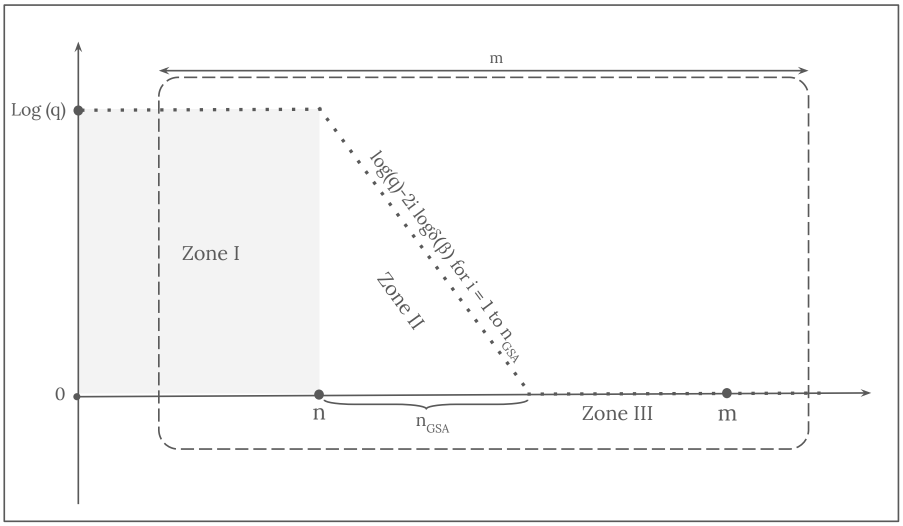


Fig. 4: Illustration of the moving window technique to estimate the profile of a BKZ reduced basis $\mathbf{B}$. The initial profile is constructed by setting the starting index of Zone II as $n + 1$ and letting the subsequent slope be given by the GSA. Zone III continues beyond m. Then a sliding window of length m (the dashed box) moves from its leftmost position to the right until the closest approximation to the lattice volume is found.

so we shift a window of indices of length m, the rank of the lattice, in increments of one, right from $\{1, \ldots, m\}$ to $\{1 + j, \ldots, m + j\}$ for some $j \in \mathbb{N}$. The shift j is chosen such that the volume implied by the profile is as close to the volume of the lattice as the discretisation of indices allows. Finally, we renormalise the profile in Zone II so that the volume of the profile we have constructed equals the volume of the lattice. A schematic of the entire process is given in Figure 4. We note that one can directly compute n_q and n_{GSA} by solving an easy system of equations, but that this requires considering four different cases depending on the existence of Zones I and III.

3.2 Exploiting the Z-shape

Let $\mathbf{B}$ be the output of BKZ-β reduction on some basis $\mathbf{B}_\mathbf{A}$ and let $r = \min\{n_q + \beta + 1, m + 1\}$. Our ability to predict the behaviour of the profile of $\mathbf{B}$ leads to the following observation. When the modulus q is relatively small compared to the length bound ν in $\mathsf{SIS}^*_{n,m,q,\nu}$ instances, the discovery of short vectors in $\mathbf{\Lambda}_{[n_q+1:r]}$ via sieving on $\mathbf{B}_{[n_q+1:r]}$ opens up avenues for new attack strategies through the lifting techniques of [12].

Lifting vectors in q-ary lattices. Recall the notation of (1) and let $\ell = n_q + 1$. We make the slight alteration of considering $\mathbf{B}_{[\ell:r]}$ defining the projected sublattice $\mathbf{\Lambda}_{[\ell:r]}$ rather than $\mathbf{B}_{[\ell]}$ defining $\mathbf{\Lambda}_{[\ell]}$, and so $\mathbf{B}' \in \mathbb{Z}^{m\times(\ell-1)}$ and $\mathbf{B}'' \in \mathbb{Z}^{m\times(r-\ell)}$. Let $\mathbf{w}_{[\ell:r]} = \mathbf{B}_{[\ell:r]}\mathbf{v}_{[\ell:r]} \in \mathbf{\Lambda}_{[\ell:r]}$ have square norm η^2. Each lift $\mathbf{w}$ of $\mathbf{w}_{[\ell:r]}$ is of the form $\mathbf{B}'\mathbf{v}' + \mathbf{B}''\mathbf{v}_{[\ell:r]}$ where $\mathbf{B}' = (q\cdot\mathbf{e}_1 \,|\, \cdots \,|\, q\cdot\mathbf{e}_{\ell-1})$.

Following (1) let $\mathbf{w}$ be the shortest lift of $\mathbf{w}_{[\ell:r]}$. The maximum square length of $\mathbf{w}$ is therefore $\eta^2 + n_q q^2/4$ and the average case length, when the first $\ell - 1$ entries of $\mathbf{w}$ are uniformly distributed mod q around 0, is approximately $\eta^2 + n_q q^2/12$. We note this approach relies on q vectors remaining at the beginning of the basis over which to lift.

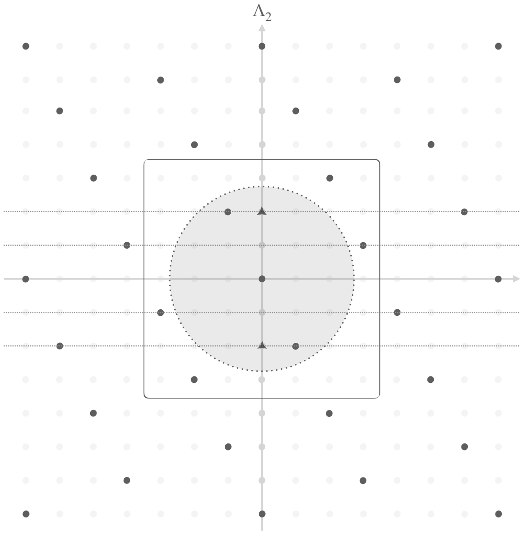


Fig. 5: Illustration of the attack in dimension 2, where we look at the projection $\mathbf{\Lambda}_{[2]}$ of the lattice $\mathbf{\Lambda} = \mathbf{\Lambda}_q^{\perp}(\mathbf{A})$ for $\mathbf{A} \in \mathbb{Z}_q^{1\times 2}$ against the q vector $(q\ 0)^t$. Lifts for the projections within an ℓ_2 ball are depicted by horizontal dotted lines. The only two points of $\mathbf{\Lambda}_{[2]}$ which can be lifted to a vector of $\mathbf{\Lambda}$ in the ℓ_2 ball are highlighted by triangles.

On success probability. The above procedure returns a $\mathsf{SIS}^*_{n,m,q,\nu}$ solution if $\|\mathbf{w}\|^2 \leqslant \nu^2$, i.e. when the lifted entries have square norm less than $\nu^2 - \eta^2$. We make the assumption, which we experimentally verify in Section 5, that the first n_q entries of $\mathbf{w}$ are uniformly distributed in $\mathsf{Cube}_{n_q}(q)$, so this success condition is equivalent to a uniform element of $\mathsf{Cube}_{n_q}(q)$ lying in $\mathsf{B}_{n_q}\left(\sqrt{\nu^2 - \eta^2}\right)$. This probability is precisely

$$p(\nu, \eta, q, n_q) = \frac{\left|\mathsf{Cube}_{n_q}(q) \cap \mathsf{B}_{n_q}\left(\sqrt{\nu^2 - \eta^2}\right)\right|}{|\mathsf{Cube}_{n_q}(q)|}, \tag{2}$$

3.3 On balls and cubes

The denominator of (2) is q^{n_q}, but we require an efficient method to compute the numerator. For this, we appeal to the theta series of $\mathbb{Z}$.

Convolution of truncated theta. We now present a method to calculate the number of lattice points of $\mathbb{Z}^{n_q}$ contained in the intersection of a centered cube and ball. Our method revolves around considering convolutions of the function $\Theta_{\mathbb{Z}}$.

We define $\Theta_{\mathbb{Z},N} = 1 + \sum_{1 \leqslant j \leqslant N} 2X^{j^2}$, and for any polynomial $p(X) = \sum_{i \in \mathbb{N}} \alpha_i X^i \in \mathbb{Z}[X]$, we define $p_N(X) = \sum_{0 \leqslant i \leqslant N^2} \alpha_i X^i$. Note this truncates a polynomial at its degree N^2 term, similar to $\Theta_{\mathbb{Z},N}$. By definition of the product of polynomials, $\Theta_{\mathbb{Z},N} \cdot \Theta_{\mathbb{Z},N}$ is a polynomial whose j^{th} coefficient is the number of integer points of squared norm j and whose coordinates are all smaller than or equal to N in absolute value. Hence, truncating the polynomial at its degree M^2 term and evaluating it at 1 gives exactly the number of points in $\mathbb{Z}^2$ inside the ℓ_2 ball of radius M and with coefficients smaller than or equal to N in absolute value. That is to say, if N is even, the number of points in $\mathsf{Cube}_2(2N+1) \cap \mathsf{B}_2(M)$.[3]

This simple observation leads to a recursive approach that generalises it to arbitrary dimensions. We seek to compute $\Theta^{(n)}$, defined by $\Theta^{(1)} = \Theta_{\mathbb{Z},N}$ and $\Theta^{(i)} = \left(\Theta^{(i-1)} \cdot \Theta^{(1)}\right)_M$ for some n, N, M. In words, this process counts the integer points introduced by increasing the dimension of the cube and removes the points outside of the ball.

One may think of this truncated convolution as a product in the ring $\mathbb{Z}[X]/(X^M)$. It is therefore tempting to accelerate the calculation of $\Theta^{(n)}$ using fast exponentiation (square-and-multiply). It turns out that the naïve iterative approach is also competitive for the parameters at hand if one exploits the fact that $\Theta^{(1)}$ is a rather sparse polynomial with only

[3] Even N relates to odd $q = 2N + 1$. Allowing for even q, where $\mathsf{Cube}_n(q)$ is non-symmetric, requires slightly more care. We are concerned with odd q in this work.

$\sqrt{N}$ non zero coefficients out of $M > N$. Indeed, the former approach has a complexity of $O(M^2 \log n)$ arithmetic operations and the latter $O(n \cdot M \cdot \sqrt{N})$. With some implementation effort, the former approach could be accelerated using the Fast Fourier Transform for convolution, leading to $O(M \log M \log n)$ complexity, though this was not necessary for our parameters. Furthermore, to explore the attack parameters, we will generally want to compute $\Theta^{(n)}$ for increasing values of n; the iterative approach with caching perfectly fits this use case.

Alternative Approach. We also mention an alternative approach we considered for computing this numerator, at least approximately. One might forget the discrete aspect of the problem, and simply compute the volume of the intersection between a hyperball and a hypercube. An efficient method exists [23,5] and we implemented it,[4] but found it difficult to use: it requires floating-point computation with high precision and the careful truncation of infinite series. This approach might still be preferable when the modulus q is large.

3.4 Putting it all together

We now give the full attack in Algorithm 1. We then outline how we estimate the success probability of our attack, which is experimentally verified in Section 5, and give its cost.

To estimate the success probability of Algorithm 1, we propose with two conservative assumptions:

i. the maximum length of vectors in the projected sieve database $\mathcal{P}$ is $\sqrt{4/3}\,q$, rather than the slightly smaller $\sqrt{4/3}\,\|\mathbf{b}_\ell^*\|$ that we would approximately expect via the Gaussian heuristic,
ii. all $(4/3)^{(r-\ell)/2}$ vectors in $\mathcal{P}$ are of this maximum length.

Recalling (2) and setting $n_q = \ell - 1$ and $\eta = \sqrt{4/3}\,q$, given that (q, ν) are parameters of our SIS^* instance, we compute $p = p(\nu, \eta, q, n_q)$ via the methods of Section 3.3. We now make the assumption that the first $\ell - 1$ entries of each shortest lift $\mathbf{w}$ of $\mathbf{w}_{[\ell:r]} \in \mathcal{P}$ are independent and identically distributed, implying in particular that each lift has length shorter than ν with probability p. Hence, the expected number of successes of the lifting event over the $4/3^{(r-l)/2}$ candidates of $\mathcal{P}$ corresponds to the expectation of a binomial random variable with $(4/3)^{(r-\ell)/2}$ trials and success probability p. It is therefore $(4/3)^{(r-\ell)/2}p$.

[4] https://github.com/verdiverdiverdi/ball-box

Algorithm 1: Z-attack outline

```
Input: A matrix $\mathbf{A} \in \mathbb{Z}_q^{n \times m}$, a threshold value $\nu > 0$, a
       BKZ blocksize $\beta$.
Output: A solution $\mathbf{x} \in \mathbb{Z}^m \setminus q\mathbb{Z}^m$ such that $\mathbf{Ax} = 0 \bmod q$
        and $\|\mathbf{x}\| \leqslant \nu$, or $\perp$ if the attack is unsuccessful.
Write $\mathbf{A}$ as $(\mathbf{A}_1 \,|\, \mathbf{A}_2) \in \mathbb{Z}_q^{n \times n} \times \mathbb{Z}_q^{n \times (n-m)}$
Assert $\mathbf{A}_1 \in \mathrm{Gl}_n(\mathbb{Z}_q)$
$\mathbf{B}_\mathbf{A} \leftarrow \begin{pmatrix} q\mathbf{I}_n & -\mathbf{A}_1^{-1}\mathbf{A}_2 \\ \mathbf{0} & \mathbf{I}_{m-n} \end{pmatrix}$
Run BKZ-$\beta$ algorithm on $\mathbf{B}_\mathbf{A}$, receive $\mathbf{B}$
Let $\ell$ be maximal such that $\mathbf{B}$ begins $(q \cdot \mathbf{e}_1 \,|\, \cdots \,|\, q \cdot \mathbf{e}_{\ell-1})$
if $\ell = 1$ then return $\mathbf{b}_1$ when $\mathbf{b}_1 \leqslant \nu$ else $\perp$
Let $r \leftarrow \min\{\ell + \beta, m + 1\}$
Let $\mathbf{B} = (\mathbf{B}' \,|\, \mathbf{B}'' \,|\, *) \in \mathbb{Z}^{m \times (\ell-1)} \times \mathbb{Z}^{m \times (r-\ell)} \times \mathbb{Z}^{m \times (m-r+1)}$
$\mathcal{P} \leftarrow$ Sieve $\left(\mathbf{\Lambda}_{[\ell,r]}\right)$ using $\mathbf{B}_{[\ell:r]}$
for $\mathbf{w}_{[\ell:r]} \in \mathcal{P}$ do
    Let $\mathbf{v}_{[\ell:r]} \in \mathbb{Z}^{r-\ell}$ such that $\mathbf{w}_{[\ell:r]} = \mathbf{B}_{[\ell:r]}\mathbf{v}_{[\ell:r]}$
    Find shortest $\mathbf{w} = \mathbf{B}'\mathbf{v}' + \mathbf{B}''\mathbf{v}_{[\ell:r]}$, i.e. reduce the first
      $\ell - 1$ entries of $\mathbf{B}''\mathbf{v}_{[\ell:r]}$ around 0 mod $q$
    if $\|\mathbf{w}\| \leqslant \nu$ then return $\mathbf{w}$
end for
return $\perp$
```

If this expected value is less than one, we rerandomise Zone II of $\mathbf{B}$ (in particular, leaving the q vectors and Zone III unaltered) and repeat once again lattice reduction to retrieve the Z-shape profile and restart the attack. Note that p is non-decreasing if the number of q vectors remaining at the beginning of the basis decreases. We assume that performing the sieving and lifting operation again is independent of previous attempts.

The cost of the attack under consideration is evaluated by adopting the CoreSVP methodology [4]. Specifically, we assume that the total cost of the BKZ reduction and sieve in the projected sublattice can be approximated by a single SVP oracle call. By leveraging lattice sieves, we estimate this cost to be of the order $2^{c\beta + o(\beta)}$, where $c = 0.292$ [8] for a classical lattice sieve. We acknowledge that this estimate is a simplification and underestimate, but we employ it to facilitate comparisons to the security levels of signature schemes proposed in [16]. We note the conventional technique of unbalancing the reduction and sieving dimensions could, in a more precise cost model, optimise our attack.

3.5 Extension to ISIS

For convenience, the attack under consideration has thus far been discussed in the homogeneous setting. Cryptanalysing signature schemes typically requires one to solve ISIS. From a complexity-theoretic perspective, we demonstrate that the inhomogeneous variant is not inherently more difficult. However, our proposed reduction loses a factor of approximately mq in the success probability of the attack. We contend that this loss is largely a consequence of the reduction and posit that we can achieve the significantly smaller factor $q/2$.

A reduction from ISIS to SIS*. While the following reduction from ISIS to SIS in instances where $\nu < q$ seems folklore, it has not been extensively documented beyond a comment by Peikert.[5] Below we give a similar reduction that does not require $\nu < q$.

Lemma 1. *For prime q, if there exists adversary $\mathcal{A}$ solving $\mathsf{SIS}^*_{n,m+1,q,\nu}$ in time T with success probability p, then there exists adversary $\mathcal{B}$ solving $\mathsf{ISIS}_{n,m,q,\nu}$ in time $T + \mathrm{poly}(n, m, \log q)$ with probability at least*

$$\frac{p}{(m+1)(q-1)} - \frac{1}{q^n}.$$

Proof. Set $m' = m+1$. For an $\mathsf{ISIS}_{n,m,q,\nu}$ instance $(\mathbf{A}, \mathbf{u})$, $\mathcal{B}$ proceeds by sampling $f \leftarrow \mathsf{U}(\mathbb{Z}_q^\times)$ and a uniform permutation matrix $\mathbf{P} \in \mathbb{Z}^{m' \times m'}$. Subsequently, $\mathcal{B}$ creates $\mathbf{A}' = [\mathbf{A}|f\mathbf{u}] \cdot \mathbf{P}$ and transmits it to $\mathcal{A}$. Note that the invertibility of f and $\mathbf{P}$ implies that the distribution of $\mathbf{A}'$ remains uniform based on the uniformity of $(\mathbf{A}, \mathbf{u})$. Furthermore, the distribution of $\mathbf{A}'$ is independent of $\mathbf{P}$ and f and follows the correct input distribution for $\mathcal{A}$.

Upon receiving $\mathbf{A}'$, $\mathcal{A}$ produces $\mathbf{x}'$. With probability p, it holds that $\mathbf{A}'\mathbf{x}' = \mathbf{0} \bmod q$, $\|\mathbf{x}'\| \leqslant \nu$, and $\mathbf{x}' \notin q\mathbb{Z}^{m'}$. Specifically, at least one coordinate of $\mathbf{x}'$ must be non-zero modulo q. As such, with probability at least $1/m'$, $\mathbf{P}\mathbf{x}'$ is of the form $(\mathbf{x}, y)$, where $\mathbf{x} \in \mathbb{Z}^m$ and $y \in \mathbb{Z} \setminus q\mathbb{Z}$. It further holds with probability $1/(q-1)$ that $f = -y^{-1}$. Notably, $\mathbf{x}$ has $\|\mathbf{x}\| \leqslant \|\mathbf{P}\mathbf{x}'\| = \|\mathbf{x}'\| \leqslant \nu$, and if $\mathbf{x} \neq \mathbf{0}$, it constitutes a solution to the $\mathsf{ISIS}_{n,m,q,\nu}$ instance. To conclude, remark that $\mathbf{x} = \mathbf{0}$ only when $\mathbf{u} = \mathbf{0}$, which occurs with probability $1/q^n$. □

[5] https://crypto.stackexchange.com/questions/87097/

A heuristic improvement. We note that the above reduction transforms generic adversaries. Our Z-shape attack implements a particular SIS^* solver, with a specific property on the distribution of its output: in our model some number, greater than one, of the first entries of the output solution $\mathbf{x}'$ are uniform mod q around 0. Hence, let us assume that the SIS^* adversary $\mathcal{A}$ above has the same property and design a better reduction. In the notation of Lemma 1 we fix $f = 1$ and $\mathbf{P}$ to a be permutation matrix that sets $\mathbf{u}$ as the first column of $\mathbf{A}'$ and ensures the first n columns are in $\mathrm{Gl}_n(\mathbb{Z}_q)$, in particular let $\mathbf{A}' = (\mathbf{u} \,|\, \bar{\mathbf{A}})$. Let the SIS^* solution be $\mathbf{x}' = (x_1' \,|\, \mathbf{x}'')$. If $x_1' \in \{1, -1\}$ then we have solved our ISIS instance as $\mathbf{A}'\mathbf{x}' = \pm\mathbf{u} + \bar{\mathbf{A}}\mathbf{x}'' = 0 \bmod q$ and one may use the relevant submatrix of $\mathbf{P}$ and potentially negation to recover $\mathbf{x}$ such that $\mathbf{A}\mathbf{x} = \mathbf{u} \bmod q$. Note $\|\mathbf{x}\| = \|\mathbf{x}''\| \leqslant \|\mathbf{x}'\| \leqslant \nu$.

In the above our SIS^* solver must function in one rank higher than the original ISIS instance and the probability that a given $x_1' \in \{1, -1\}$ is $2/q$, i.e. the success probability is $2/q$ rather than approximately $1/mq$ as in Lemma 1.

4 Optimisations

In this section, we introduce an optimisation to enhance the generic attack of Algorithm 1. It employs a technique from the lattice sieving literature referred to as “on-the-fly lifting”. This approach considers more lifts over the q vectors, albeit at slightly longer lengths.

4.1 On the fly lifting

During the execution of a lattice sieve, pairs of vectors are added together to search for new and shorter vectors. There are two main methods for this process: a double loop over the entire current database of vectors [21] or the use of locality-sensitive techniques to consider only pairs of vectors with a high probability of summing to a new, shorter vector [8]. Regardless of the method used, the process is iterative, and the lengths of vectors in the sieve database decrease over many such searches for pairs. This means that many more vectors are considered than ultimately inhabit the terminal sieving database. The on-the-fly lifting technique is introduced to consider lifting some subset of these vectors, as well as those in the terminal database, in the hope that some well-chosen excess computation can improve the sieve’s performance [3].

We model on-the-fly lifting by considering the terminal sieve database and performing one more iteration. Each vector encountered in that iteration, regardless of its length, is lifted. The number and length of these extra vectors will vary depending on the style of sieve used. It is important to note that this surplus iteration is not necessary in practice, as vectors of the appropriate length can simply be lifted during the sieving procedure. However, it is conceptually cleaner.

Nguyen–Vidick style sieves. In Nguyen–Vidick style sieves [21] each iteration of the sieve is a double loop over the database where all distinct pairs of vectors are added and the shortest sums kept.[6] Given our assumption that a terminal sieving database on a rank β lattice $\mathbf{\Lambda}$ has size $(4/3)^{\beta/2}$ and maximum length $\sqrt{4/3}\,\mathrm{gh}(\mathbf{\Lambda})$, performing a final sieving iteration visits $(4/3)^{\beta}$ vectors of length less than $\sqrt{2}\cdot\sqrt{4/3}\,\mathrm{gh}(\mathbf{\Lambda})$. Specialising to Algorithm 1 this means performing the lifting during the sieve operation, and altering our conservative assumptions on the success probability to stating that each vector we attempt to lift has length $\sqrt{2}\cdot\sqrt{4/3}\,q$ and that there are $(4/3)^{r-\ell}$ of them. We note that while, when not considering on the fly techniques, we took complexity exponent $c = 0.292$ because there was no reason to not consider the fastest lattice sieve, Nguyen–Vidick sieves have asymptotic time complexity given by the exponent $c = 0.415$ [21].

The Becker–Ducas–Gama–Laarhoven sieve. Sieves that use locality sensitive techniques achieve lower time complexities by considering fewer pairs of vectors in an iteration [8]. This means that in our model that considers repeating the final iteration of sieving on the terminal sieve database, such sieves give fewer opportunities for on the fly lifting. On the positive side, the ability of such sieves to forego trying so many pairs of vectors comes from mechanisms to select only pairs that are more likely to have a short sum. In particular, the distribution of the lengths of sums of pairs that *are* selected is concentrated on shorter values than for Nguyen–Vidick style sieves. The following lemma examines the time optimal parameters of [8]. Here a vector $\mathbf{u}$ is compared only with vectors $\mathbf{w}$ that take angular distance not more than $\pi/3$ with some vector $\mathbf{r}$, which itself takes angular distance not more than $\pi/3$ with $\mathbf{u}$.

[6] For simplicity, one may think of including $\mathbf{0}$ in the database to allow the iteration to keep short vectors already present in the database.

Lemma 2. *Following the notation of [8, Sec. 2] let $\alpha = 1/2$ and $\mathbf{u} \in S^{n-1}$ be the centre of spherical cap $\mathcal{C}_{\mathbf{u},\alpha}$. Let $\mathbf{r} \in \mathcal{C}_{\mathbf{u},\alpha}$ be the centre of another spherical cap $\mathcal{C}_{\mathbf{r},\alpha}$. The probability that uniform $\mathbf{w} \in \mathcal{C}_{\mathbf{r},\alpha}$ is such that $\|\mathbf{u} - \mathbf{w}\| \leqslant \sqrt{3/2}$ is at least one half.*

Proof. Rotate such that $\mathbf{r} = \mathbf{e}_1$ and decompose $\mathbf{u} = \mathbf{u}' + u_1\mathbf{r}$, $\mathbf{w} = \mathbf{w}' + w_1\mathbf{r}$ such that $\langle \mathbf{u}', \mathbf{r}\rangle = \langle \mathbf{w}', \mathbf{r}\rangle = 0$. We have $\|\mathbf{u} - \mathbf{w}\|^2 = \|\mathbf{u}' - \mathbf{w}'\|^2 + (u_1 - w_1)^2$ with $u_1 \in [1/2, 1]$ and $\|\mathbf{u}'\|^2 = 1 - u_1^2$, and similarly for $(w_1, \mathbf{w}')$. Then $\|\mathbf{u} - \mathbf{w}\|^2 = \|\mathbf{u}'\|^2 + \|\mathbf{w}'\|^2 - 2\langle \mathbf{u}', \mathbf{w}'\rangle + (u_1 - w_1)^2 \leqslant 3/2 - 2\langle \mathbf{u}', \mathbf{w}'\rangle$. We project $\mathbf{u}, \mathbf{w}$ in the cap $\mathcal{C}_{\mathbf{r},\alpha}$ onto the ball of one less dimension $\mathsf{B}_{n-1}\left(\sqrt{3/4}\right) \subset \{0\} \times \mathbb{R}^{n-1}$. By the rotational symmetry of $\mathcal{C}_{\mathbf{r},\alpha}$ around the axis $\mathbf{r}$, for any $\mathbf{u}$ and uniform $\mathbf{w} \in \mathcal{C}_{\mathbf{r},\alpha}$, the angle between $\mathbf{u}'$ and $\mathbf{w}'$ is uniform in $[0, 2\pi)$, and therefore the inner product above is non negative with probability one half. In this case, $\|\mathbf{u} - \mathbf{w}\| \leqslant \sqrt{3/2}$. □

By scaling onto the sphere of radius $\sqrt{4/3}\,\mathrm{gh}(\mathbf{\Lambda})$ we have $\sqrt{3/2} \cdot \sqrt{4/3}\,\mathrm{gh}(\mathbf{\Lambda}) = \sqrt{2}\,\mathrm{gh}(\mathbf{\Lambda})$. We therefore change our assumptions on the success probability of Algorithm 1, when using the sieve of [8], to each vector we attempt to lift having length $\sqrt{2}\,q$ and there being $(3/2)^{(r-\ell)/2}$ of them. This number of vectors comes from the $\alpha = \beta = 1/2$ case of [8, Sec. 7] and is less than the $(4/3)^{r-\ell}$ of the Nguyen–Vidick sieves. Here we have complexity exponent $c = 0.292$.

A possible improvement. We note that when considering on the fly lifting it is not necessarily the case that the lengths of vectors are concentrated around their maximum, as opposed to the terminal database of a sieve. For example, in a Nguyen–Vidick style sieve, if the lengths of the vectors considered during on the fly lifting have lengths concentrated below $\sqrt{2} \cdot \sqrt{4/3}\,\mathrm{gh}(\mathbf{\Lambda})$ then our model is pessimistic; taking a shorter length as an upper bound for our projected vectors would better match reality and lower the cost of the attack.

5 Experimental verification

In this section we experimentally verify two heuristics used in our attack. The first is on the behaviour of the lengths of lifted entries of vectors from a projected sieve database, which are expected to follow the uniform distribution over $\mathsf{Cube}_{n_q}(q)$. As an extension, we also verify that the total lengths of lifted vectors match our expectations. The second heuristic of our model relates to the simulation of the Z-shape after BKZ reduction.

We note that we are considering the lengths of lifted vectors without considering on-the-fly lifting, as introduced in Section 4.

Experimental toolkit. The `fpylll` library [26] is used for BKZ reduction algorithm and the general sieve kernel [27,3] is used for sieving in projected sublattices. We use the progressive BKZ algorithm [6], in which a subset of block sizes $(\beta'_i)_i \subset \{3, \ldots, \beta - 1\}$ are used in some number of BKZ-β' tours as a prereduction process prior to BKZ-β.

5.1 The lengths of lifts

In these experiments we use progressive BKZ-β with a decreasing number of iterations as $\beta' < \beta$ increases. Specifically, we denote the process of running t iterations of BKZ-β' for $\beta' \in \{l, \ldots, u\}$ by the pair $(t, [l, u])$, resulting in the following sequence: $(8, [3, 5])$, $(4, [6, 10])$, $(2, [11, 20])$, $(1, [40, \infty))$.

Uniformity in $\mathsf{Cube}_n(q)$. Following the selection of parameters n, m, q, and a reduction parameter β for BKZ reduction such that we expect q vectors to remain at the beginning of the basis, we perform BKZ-β reduction on a uniform $\mathbf{B_A}$, resulting in $\mathbf{B}$. The number of remaining q vectors in $\mathbf{B}$ is denoted by n_q. Next, by setting $\ell - 1 = n_q$ and $r = \max\{\ell + \beta, m + 1\}$, we sieve in $\mathbf{\Lambda}_{[\ell:r]}$ using $\mathbf{B}_{[\ell:r]}$, following Algorithm 1. We ensure that $r \leqslant m+1$ so that we sieve in a rank β projected sublattice. The number of vectors having length less than or equal to $4/3\, q$ is denoted as N, and we ensure that this quantity is greater than $(4/3)^{\beta/2}$. Subsequently, the sieve database is lifted to $\mathbf{B}$ over the q vectors, and the length of each lifted vector, limited to its first n_q entries, is recorded as $\{L_i\}_{i=1}^N$. As in Section 3.3, we can calculate the fraction of $\mathsf{Cube}_{n_q}(q)$ having length less than a given radius R, i.e.

$$p'(R, q, n_q) = \frac{|\mathsf{Cube}_{n_q}(q) \cap \mathsf{B}_{n_q}(R)|}{|\mathsf{Cube}_{n_q}(q)|},$$

similarly to (2). Further, $\{L_i\}_{i=1}^N$ is sorted as $L_1 \leqslant \cdots \leqslant L_n$, and for any i, if there exists $j < i$ such that $L_j = L_i$, then we remove L_j. Ultimately, for the remaining L_i the coordinates $(L_i, p'(L_i, q, n_q))$ and $(L_i, i/N)$ are plotted. These represent the proportion of lifts with length that is less than or equal to L_i according to our model ("Modelled proportion" in Figure 6), and observed in our experiments ("Experimental proportion" in Figure 6).

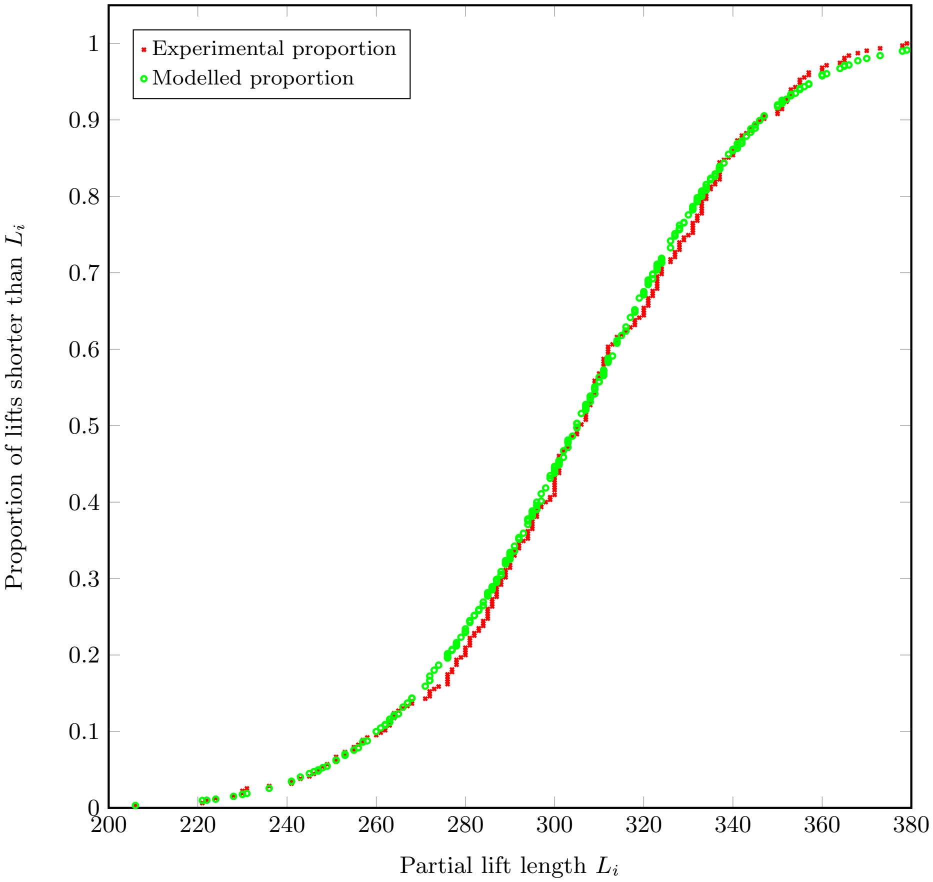


Fig. 6: For $(n, m, q, \beta) = (120, 240, 257, 40)$ we check the distribution of the length of only the lifted entries of lifted vectors, following the description of the first experiment in Section 5.1.

Distribution of total lengths. Given the above approach we can also consider, given a particular distribution of lengths of projected vectors in the database of $\mathrm{Sieve}\big(\mathbf{\Lambda}_{[\ell:r]}\big)$, the distribution of lengths of the full lifted vectors expected by our model. Note that we must consider the distribution of the lengths of the projected vectors since, under our model, the distributions of the lifts of projected vectors of different lengths are themselves different. In this case we let $\{L_{proj,i}\}_{i=1}^{N}$ represent the lengths of the vectors in the projected sieve database, and for each index i let $\{L_{total,i}\}_{i=1}^{N}$ represent the length of the respective entire lifted vector. We sort $\{L_{total,i}\}_{i=1}^{N}$ as $L_{total,1} \leqslant \cdots \leqslant L_{total,N}$ and for all i if there exists

$j < i$ such that $L_{total,j} = L_{total,i}$ we remove $L_{total,j}$. For given $\{L_{proj,i}\}_{i=1}^{N}$ and length $L_{total,k}$, the expected number of lifted vectors of length less than or equal to $L_{total,k}$ is given by

$$E_k = \sum_{i=1}^{N} p'\left(\sqrt{L_{total,k}^2 - L_{proj,i}^2}\,, q, n_q\right).$$

We plot coordinates $(L_{total,k}, E_k)$ and $(L_{total,k}, k)$ as "Modelled number" and "Experimental number" respectively in Figure 7, the latter of which represents the number of lifts from our experiments which have length less than or equal to $L_{total,k}$.

5.2 The Z-shape basis

To achieve the Z-shape of Section 3.1 is unfortunately not a matter of simply applying lattice reduction to a basis $\mathbf{B_A}$. As described in [2, Fig. 6] there is a phenomenon whereby, rather than Zone III consisting of Gram–Schmidt vectors of length 1, a kink appears with vectors of Gram–Schmidt norm strictly less than 1. These shorter than expected Gram–Schmidt vectors introduce, in the log scale, negative terms to the sum $\sum_i \log\|\mathbf{b}_i^*\| = n \log q$. This in turn, due to the invariance of the sum, means some $\log\|\mathbf{b}_i^*\|$ must be greater, potentially leading to more q vectors than expected. Having a larger Zone I means that on average more length is added during the lifting process, lowering the efficacy of our attack. To avoid this we take the number of indices we expect to be in Zone III according to our model of Section 3.1 and perform no lattice reduction on them. These indices are then unchanged, and since lattice reduction preserves the real span of the vectors of Zone I and Zone II, their Gram–Schmidt norms remain 1. We also perform slightly heavier progressive BKZ in these experiments, denoted by the single pair $(8, [3, \infty))$. With the above, slightly artificial, alterations we are able to experimentally achieve the number of q vectors expected by our model, see Table 1. In Figure 8 we also plot the average profile of the same experiments against the Z-shape profile expected by our model in Section 3.1.

Our modified BKZ reduction process is seemingly capable of producing bases with the expected number of q vectors, as predicted by our model. However, the aforementioned kinks are still to some degree present for $\beta \in \{20, 30, 40\}$. The accurate modelling of Z-shape bases remains an open question. Nevertheless, we maintain that it would be unsatisfactory to rely on the presence of such kinks in a basis profile for the practical security of a cryptographic scheme.

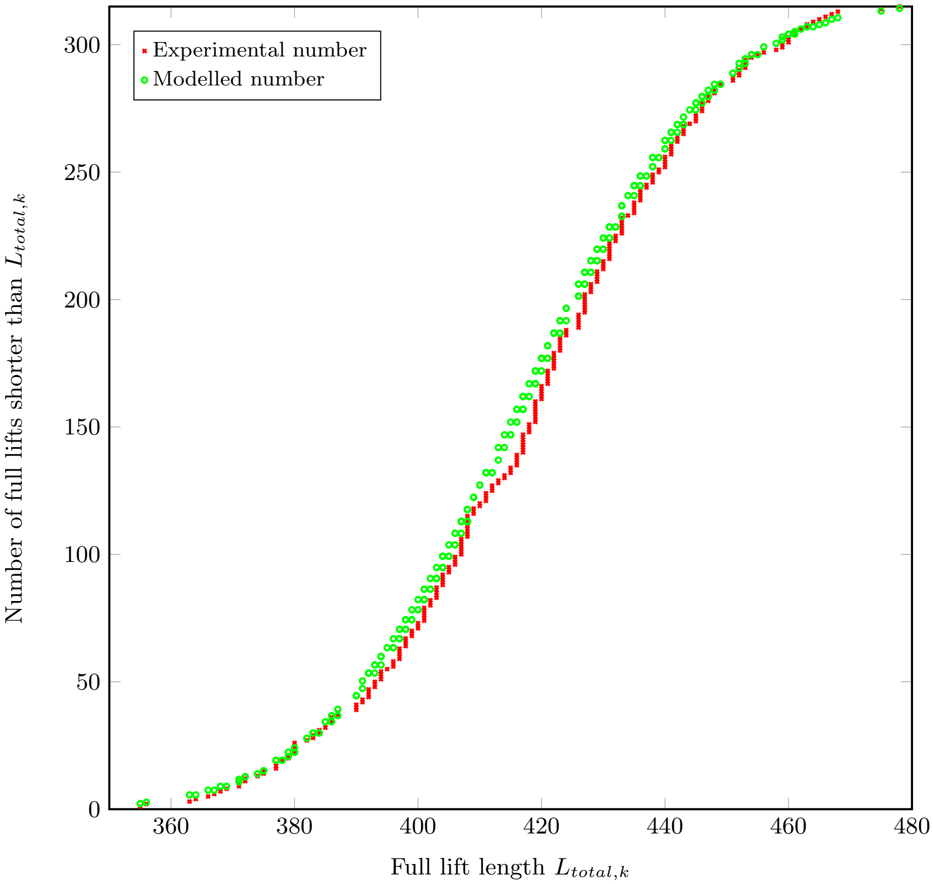
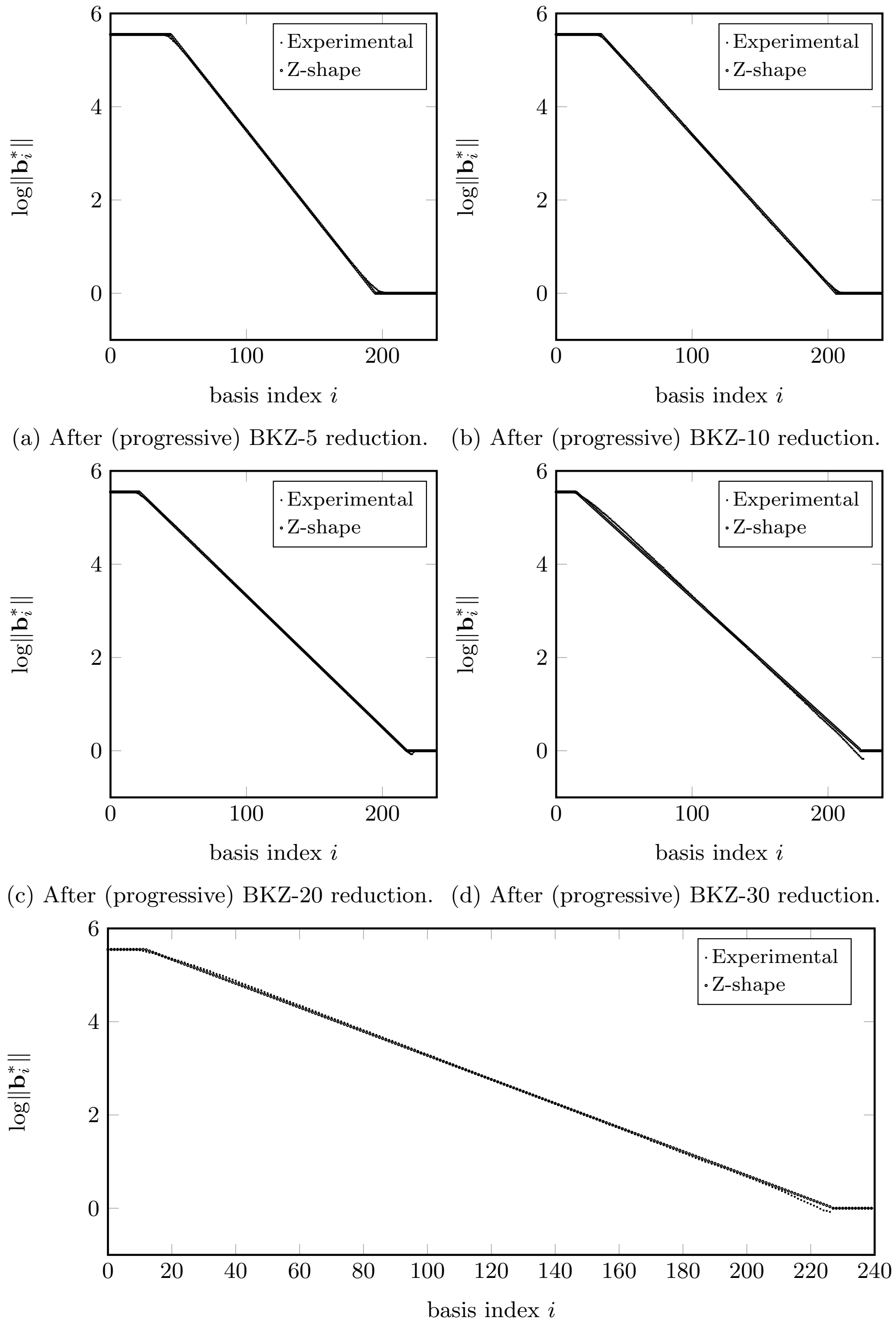


Fig. 7: For $(n, m, q, \beta) = (120, 240, 257, 40)$ we check the distribution of the length of the entire lifted vectors, following the description of the second experiment in Section 5.1.

6 Application and practical cryptanalysis

6.1 Small *q* hash and sign signatures

In [16] a simple technique to reduce the bandwidth of hash and sign based signatures such as Falcon [22] and Mitaka [15] was proposed: reduce the size of the modulus q. Even though this technique is simple and effective, care must be taken with regards to the choice of q, as the best attacks are dependent on q. By framing Falcon and Mitaka as particular ISIS instances, we propose a revision of the cryptanalysis of [16] with the attack and optimisations introduced in Section 3 and Section 4.



(a) After (progressive) BKZ-5 reduction.

(b) After (progressive) BKZ-10 reduction.

(c) After (progressive) BKZ-20 reduction.

(d) After (progressive) BKZ-30 reduction.

(e) After (progressive) BKZ-40 reduction.

Fig. 8: For $(n, m, q, \beta) = (120, 240, 257, 40)$ we run the altered progressive BKZ-β reduction described in Section 5.2 and plot the profile expected by our Z-shape model against the average of 60 experimental profiles.

β	$\mathbb{E}[X]$	$\sqrt{\mathbb{V}[X]}$	Section 3.1
5	41.4	2.2	45
10	31.6	1.7	33
20	19.4	1.4	22
30	13.9	1.4	15
40	9.8	1.5	12

Table 1: For $(n, m, q, \beta) = (120, 240, 257, \beta)$ we run the altered progressive BKZ-β reduction described in Section 5.2. After $\beta' \in \{5, 10, 20, 30, 40\}$ we compute the average number of q vectors and the standard deviation of this number over 60 experimental trials and denote these quantities as $\mathbb{E}[X]$ and $\sqrt{\mathbb{V}[X]}$. In the final column we give the number of q vectors expected by the model of Section 3.1.

On hash and sign signatures as ISIS. We review the underlying principles of hash and sign signatures such as Falcon and Mitaka. The scheme involves two keys: the signing key, which acts as a trapdoor, and enables one to solve the approximate closest vector problem via discrete Gaussian sampling over the lattice, and the verification key $\mathbf{H}$, which can only verify whether a point belongs to the lattice.

We provide a high-level, but incomplete, description of the signing process for both Falcon and Mitaka. In both schemes, the public key $\mathbf{H}$ can be expressed as an integer matrix with specific structure. While we do not delve into the details of the matrix construction, the security argument crucially relies on the decisional NTRU assumption, which loosely states that $\mathbf{H}$ can be viewed as a random matrix.[7]

Let $m = 2n$. The signing algorithm hashes a message to $\mathbf{c} \in \mathbb{Z}_q^n$ and employs the signing key to sample $\mathbf{s}_1, \mathbf{s}_2 \in \mathbb{Z}^n$ such that $\mathbf{s}_1 + \mathbf{H}\mathbf{s}_2 = \mathbf{c} \bmod q$, as described in [22, Sec. 3.9.1]. If we concatenate $\mathbf{s}_1$ and $\mathbf{s}_2$ to form $\mathbf{s}$, the signature is valid if $\|\mathbf{s}\| \leqslant \nu$ for some length bound ν. This can be viewed as an $\mathsf{ISIS}_{n,m,q,\nu}$ instance with added structure, where $\mathbf{A} = (\mathbf{I}_n \,|\, \mathbf{H})$. If we can differentiate this ISIS instance from a uniform $\mathbf{A}$, we can break the decisional NTRU assumption discussed earlier. Note that if we express $\mathbf{A} = (\mathbf{A}_1 \,|\, \mathbf{A}_2)$, where $\mathbf{A}_1 \in \mathbb{Z}_q^{n \times n}$ and $\mathbf{A}_1 \in \mathrm{Gl}_n(\mathbb{Z}_q)$, we can transform this ISIS instance into one that is semantically similar to those implied by Falcon and Mitaka via $\mathbf{A}_1^{-1}\mathbf{A} = (\mathbf{I}_n \,|\, \mathbf{A}_1^{-1}\mathbf{A}_2)$. With this view of Falcon and Mitaka as ISIS instances, we can apply our attack.

[7] Strictly speaking, the NTRU assumption only states that the number ring element from which the matrix $\mathbf{H}$ can be reconstructed is indistinguishable from uniform.

$(n, q, \nu$, scheme)	no otf		otf		[16, Tab. 2]
	(SIS*)	ISIS	(SIS*)	ISIS	
(512, 257, 801, F)	(95)	98	(90)	**92**	118
(512, 257, 1470, M)	(8)	14	(8)	**12**	94
(512, 521, 1141, F)	(113)	115	(110)	**111**	121
(512, 521, 2094, M)	(55)	58	(51)	**54**	97
(512, 1031, 1606, F)	(117)	**119**	(120)	121	122
(512, 1031, 2945, M)	(81)	84	(78)	**80**	99

Table 2: Classical complexities of variants of our attack against signature forgery on small q parameter sets for Falcon and Mitaka. Our n relates to d in [16], we fix $m = 2n$, and F and M denote Falcon and Mitaka respectively. We report two pairs of complexities, one without on the fly lifting and one with, denoted "no otf" and "otf" respectively. In each pair we report in order the cost of the SIS* attack and the cost of the ISIS attack, accounting for the reduction loss factor of $q/2$ on success probability. The former is given between parenthesis. The final column is the suggested bit security of the parameter set in the CoreSVP model according to [16, Tab. 2]. The lowest ISIS attack cost is in boldface in each row.

Attack costs. In Table 2 when considering on the fly lifting we only consider the faster sieve of [8], and recall that this optimisation was *not* subject to experimental validation in Section 5. For the second entry of each pair of estimated costs we incorporate the probability loss factor into our script by assumpting each lift is short enough with probability reduced by a factor $q/2$.

An overestimated loss. In the next paragraph we report on experiments that mount the above attack. In these experiments we did not multiply by the randomising scalar f in Lemma 1 for the reasons discussed below the lemma. The success probability of the attack appears to be higher than even the heuristic $2/q$ we suggest. This can be explained by the fact that vectors output by SIS* solvers are short, hence their coordinates are biased toward smaller values such as 1 and -1.

We therefore reiterate our warning that the concrete results given in this paper are only meant as a cautionary tale, and certainly not as definitive cost estimates usable for claiming concrete security. Most likely our attack and its analysis can be further improved. This optimisation effort is left to whoever dares venture into the low modulus ISIS regime.

Practical attack on Mitaka with small q. As seen in Table 2 the cost of the attack on the small q variant of Mitaka with $n = 512$ and $q = 257$ appears very low, so low that not mounting the attack in practice would be indefensible. Without on the fly lifting our script proposes as the optimal attack a blocksize of 25 repeated $2^{6.5}$ times. Due to various overheads, one might prefer to choose a blocksize of 45 and repeat only once.

In practice, it is generally preferable to run BKZ with a smaller blocksize and run a final sieve on a projected sublattice of a larger rank to better balance the cost of the two procedures. We chose (by trial and error) a BKZ blocksize of 12 and a sieving rank of 60 and did not perform any on the fly lifting techniques, rather we lifted every vector in our terminal sieve database. We also restricted the sloped portion of the Z-shape on which we ran lattice reduction to dimension 160 to avoid having to resort to high precision floating point arithmetic in LLL.

We implemented this attack on ISIS with parameters derived from the small q parameters for Mitaka of [16]: $m = 1024$, $n = 512$, $q = 257$ and $\nu = 1470$. This implementation is provided in the sage script `attack.sage`, and relies on the libraries fpylll and g6k [26,27,3]. It ran successfully on all 20 random instances we launched, each taking less than 15 seconds on a single core (Intel(R) Core(TM) i7-4790 CPU, 3.60GHz).

References

1. Ajtai, M.: Generating hard instances of lattice problems (extended abstract). In: 28th ACM STOC. pp. 99–108. ACM Press (May 1996). https://doi.org/10.1145/237814.237838
2. Albrecht, M.R., Ducas, L.: Lattice Attacks on NTRU and LWE: A History of Refinements, pp. 15–40. London Mathematical Society Lecture Note Series, Cambridge University Press (2021). https://doi.org/10.1017/9781108854207.004
3. Albrecht, M.R., Ducas, L., Herold, G., Kirshanova, E., Postlethwaite, E.W., Stevens, M.: The general sieve kernel and new records in lattice reduction. In: Ishai, Y., Rijmen, V. (eds.) EUROCRYPT 2019, Part II. LNCS, vol. 11477, pp. 717–746. Springer, Heidelberg (May 2019). https://doi.org/10.1007/978-3-030-17656-3_25
4. Alkim, E., Ducas, L., Pöppelmann, T., Schwabe, P.: Post-quantum key exchange - A new hope. In: Holz, T., Savage, S. (eds.) USENIX Security 2016. pp. 327–343. USENIX Association (Aug 2016)
5. Aono, Y., Nguyen, P.Q.: Random sampling revisited: Lattice enumeration with discrete pruning. In: Coron, J.S., Nielsen, J.B. (eds.) EUROCRYPT 2017, Part II. LNCS, vol. 10211, pp. 65–102. Springer, Heidelberg (Apr / May 2017). https://doi.org/10.1007/978-3-319-56614-6_3

6. Aono, Y., Wang, Y., Hayashi, T., Takagi, T.: Improved progressive BKZ algorithms and their precise cost estimation by sharp simulator. In: Fischlin, M., Coron, J.S. (eds.) EUROCRYPT 2016, Part I. LNCS, vol. 9665, pp. 789–819. Springer, Heidelberg (May 2016). https://doi.org/10.1007/978-3-662-49890-3_30
7. Babai, L.: On lovász' lattice reduction and the nearest lattice point problem. Combinatorica **6**(1), 1–13 (1986), https://doi.org/10.1007/BF02579403
8. Becker, A., Ducas, L., Gama, N., Laarhoven, T.: New directions in nearest neighbor searching with applications to lattice sieving. In: Krauthgamer, R. (ed.) 27th SODA. pp. 10–24. ACM-SIAM (Jan 2016). https://doi.org/10.1137/1.9781611974331.ch2
9. Bos, J.W., Bronchain, O., Ducas, L., Fehr, S., Huang, Y.H., Pornin, T., Postlethwaite, E.W., Prest, T., Pulles, L.N., van Woerden, W.: HAWK. Tech. rep., National Institute of Standards and Technology (2023), to appear. https://csrc.nist.gov/projects/pqc-dig-sig
10. Chen, Y.: Réduction de réseau et sécurité concrète du chiffrement complètement homomorphe. Ph.D. thesis, Paris 7 (2013), http://www.theses.fr/2013PA077242, thèse de doctorat dirigée par Nguyen, Phong-Quang Informatique Paris 7 2013
11. Devevey, J., Fawzi, O., Passelègue, A., Stehlé, D.: On rejection sampling in lyubashevsky's signature scheme. In: Agrawal, S., Lin, D. (eds.) Advances in Cryptology – ASIACRYPT 2022. pp. 34–64. Springer Nature Switzerland, Cham (2022)
12. Ducas, L.: Shortest vector from lattice sieving: A few dimensions for free. In: Nielsen, J.B., Rijmen, V. (eds.) EUROCRYPT 2018, Part I. LNCS, vol. 10820, pp. 125–145. Springer, Heidelberg (Apr / May 2018). https://doi.org/10.1007/978-3-319-78381-9_5
13. Ducas, L., Postlethwaite, E.W., Pulles, L.N., Woerden, W.v.: Hawk: Module lip makes lattice signatures fast, compact and simple. In: Advances in Cryptology–ASIACRYPT 2022: 28th International Conference on the Theory and Application of Cryptology and Information Security, Taipei, Taiwan, December 5–9, 2022, Proceedings, Part IV. pp. 65–94. Springer (2023)
14. Ducas, L., van Woerden, W.P.J.: NTRU fatigue: How stretched is overstretched? In: Tibouchi, M., Wang, H. (eds.) ASIACRYPT 2021, Part IV. LNCS, vol. 13093, pp. 3–32. Springer, Heidelberg (Dec 2021). https://doi.org/10.1007/978-3-030-92068-5_1
15. Espitau, T., Fouque, P.A., Gérard, F., Rossi, M., Takahashi, A., Tibouchi, M., Wallet, A., Yu, Y.: Mitaka: A simpler, parallelizable, maskable variant of falcon. In: Dunkelman, O., Dziembowski, S. (eds.) EUROCRYPT 2022, Part III. LNCS, vol. 13277, pp. 222–253. Springer, Heidelberg (May / Jun 2022). https://doi.org/10.1007/978-3-031-07082-2_9
16. Espitau, T., Tibouchi, M., Wallet, A., Yu, Y.: Shorter hash-and-sign lattice-based signatures. In: Dodis, Y., Shrimpton, T. (eds.) CRYPTO 2022, Part II. LNCS, vol. 13508, pp. 245–275. Springer, Heidelberg (Aug 2022). https://doi.org/10.1007/978-3-031-15979-4_9
17. Howgrave-Graham, N.: A hybrid lattice-reduction and meet-in-the-middle attack against NTRU. In: Menezes, A. (ed.) CRYPTO 2007. LNCS, vol. 4622, pp. 150–169. Springer, Heidelberg (Aug 2007). https://doi.org/10.1007/978-3-540-74143-5_9
18. Lenstra, A.K., Lenstra, H.W., Lovász, L.: Factoring polynomials with rational coefficients. Mathematische Annalen **261**(4), 515–534 (1982). https://doi.org/10.1007/BF01457454, https://doi.org/10.1007/BF01457454

19. Lyubashevsky, V., Ducas, L., Kiltz, E., Lepoint, T., Schwabe, P., Seiler, G., Stehlé, D., Bai, S.: CRYSTALS-DILITHIUM. Tech. rep., National Institute of Standards and Technology (2022), available at https://csrc.nist.gov/Projects/post-quantum-cryptography/selected-algorithms-2022
20. Micciancio, D., Regev, O.: Worst-case to average-case reductions based on gaussian measures. SIAM Journal on Computing **37**(1), 267–302 (2007). https://doi.org/10.1137/S0097539705447360, https://doi.org/10.1137/S0097539705447360
21. Nguyen, P.Q., Vidick, T.: Sieve algorithms for the shortest vector problem are practical. J. Mathematical Cryptology **2**(2), 181–207 (2008). https://doi.org/10.1515/JMC.2008.009, https://doi.org/10.1515/JMC.2008.009
22. Prest, T., Fouque, P.A., Hoffstein, J., Kirchner, P., Lyubashevsky, V., Pornin, T., Ricosset, T., Seiler, G., Whyte, W., Zhang, Z.: FALCON. Tech. rep., National Institute of Standards and Technology (2022), available at https://csrc.nist.gov/Projects/post-quantum-cryptography/selected-algorithms-2022
23. Rousseau, C.C., Ruehr, O.G.: Problems and solutions. SIAM Review **39**(4), 761–789 (1997). https://doi.org/10.1137/SIREAD000039000004000761000001, https://doi.org/10.1137/SIREAD000039000004000761000001
24. Schnorr, C.P., Euchner, M.: Lattice basis reduction: Improved practical algorithms and solving subset sum problems. Mathematical Programming **66**(1), 181–199 (Aug 1994). https://doi.org/10.1007/BF01581144, https://doi.org/10.1007/BF01581144
25. Schnorr, C.P.: Lattice reduction by random sampling and birthday methods. In: Alt, H., Habib, M. (eds.) STACS 2003. pp. 145–156. Springer Berlin Heidelberg, Berlin, Heidelberg (2003)
26. development team, T.F.: fpylll, a Python wrapper for the fplll lattice reduction library, Version: 0.5.9 (2023), https://github.com/fplll/fpylll, available at https://github.com/fplll/fpylll
27. development team, T.G.: The general sieve kernel, Version: 0.1.2 (2023), https://github.com/fplll/g6k, available at https://github.com/fplll/g6k