2

Lattice Attacks on NTRU and LWE: A History of Refinements

Martin R. Albrecht and Léo Ducas

2.1 Introduction

Since its invention in 1982, the Lenstra–Lenstra–Lovász (LLL) lattice reduction algorithm [380] has found countless applications. In cryptanalysis, the two most prominent applications of LLL and its generalisations, e.g., Slide [205], Block-Korkine–Zolotarev (BKZ) [512, 520] and Self-Dual BKZ (SD-BKZ) [425], are factoring RSA keys with extra information on the secret key via Coppersmith's method [136, 451] (see the chapter by Alexander May) and the cryptanalysis of lattice-based schemes.

After almost 40 years of cryptanalytic applications, predicting and optimising lattice reduction algorithms remains an active area of research. While we do have theorems bounding the worst-case performance of these algorithms, those bounds are asymptotic and not necessarily tight when applied to practical or even cryptographic instances. Reasoning about the behaviour of those algorithms relies on heuristics and approximations, some of which are known to fail for relevant corner cases.

Recently, decades after Arjen Lenstra and his co-authors gave birth to this fascinating and lively research area, this state of affairs became a more pressing issue. Motivated by post-quantum security, standardisation bodies, governments and industries started to move towards deploying lattice-based cryptographic algorithms. This spurred the refinement of those heuristics and approximations, leading to a better understanding of the behaviour of these algorithms over the past few years.

Lattice reduction algorithms, such as LLL and BKZ, proceed with repeated local improvements to the lattice basis, and each such local improvement means solving the short(est) vector problem in a lattice of a smaller dimension. Therefore, two questions arise: how costly is it to find those local

https://doi.org/10.1017/9781108854207.004 Published online by Cambridge University Press

improvements and what is the global behaviour when those improvements are applied.

While these two questions may not be perfectly independent, we will, in this chapter, survey the second one, namely, the global behaviour of such algorithms, given oracle access for finding local improvements. Our focus on the global behaviour is motivated by our intent to draw more of the community's attention to this aspect. We will take a particular interest in the behaviour of such algorithms on a specific class of lattices, underlying the most popular lattice problems to build cryptographic primitives, namely the Learning with Errors (LWE) problem and the NTRU problem. We will emphasise the approximations that have been made, their progressive refinements and highlight open problems to be addressed.

2.1.1 LWE and NTRU

The LWE problem and the NTRU problem have proven to be versatile building blocks for cryptographic applications [104, 218, 274, 493]. For both of these problems, there exist ring and matrix variants. More precisely, the original definition of NTRU is the ring variant [274] and the matrix variant is rarely considered whereas for LWE the original definition is the matrix variant [494] with a ring variant being defined later [401, 561]. In this chapter, we generally treat the matrix variants since our focus is on lattice reduction for general lattices.

Definition 2.1 (LWE [494])**.** Let n, q be positive integers, χ be a probability distribution on $\mathbb{Z}$ and $\mathbf{s}$ be a uniformly random vector in $\mathbb{Z}_q^n$. We denote by $L_{\mathbf{s},\chi}$ the probability distribution on $\mathbb{Z}_q^n \times \mathbb{Z}_q$ obtained by choosing $\mathbf{a} \in \mathbb{Z}_q^n$ uniformly at random, choosing $e \in \mathbb{Z}$ according to χ and considering it in $\mathbb{Z}_q$, and returning $(\mathbf{a}, c) = (\mathbf{a}, \langle \mathbf{a}, \mathbf{s} \rangle + e) \in \mathbb{Z}_q^n \times \mathbb{Z}_q$.
Decision-LWE is the problem of deciding whether pairs $(\mathbf{a}, c) \in \mathbb{Z}_q^n \times \mathbb{Z}_q$ are sampled according to $L_{\mathbf{s},\chi}$ or the uniform distribution on $\mathbb{Z}_q^n \times \mathbb{Z}_q$.
Search-LWE is the problem of recovering $\mathbf{s}$ from pairs $(\mathbf{a}, c) = (\mathbf{a}, \langle \mathbf{a}, \mathbf{s} \rangle + e) \in \mathbb{Z}_q^n \times \mathbb{Z}_q$ sampled according to $L_{\mathbf{s},\chi}$.

We note that the above definition puts no restriction on the number of samples, i.e., LWE is assumed to be secure for any polynomial number of samples. Further, since for many choices of n, q, χ solving Decision-LWE allows solving Search-LWE [105, 494] and vice versa, it is meaningful just to speak of the LWE problem (for those choices of parameters). By rewriting the system in systematic form [23], it can be shown that the LWE problem, where each component of the secret $\mathbf{s}$ is sampled from the error distribution χ, is as secure

https://doi.org/10.1017/9781108854207.004 Published online by Cambridge University Press

as the problem for uniformly random secrets. LWE with such a secret, following the error distribution, is known as normal form LWE. We will consider normal form LWE in this chapter. Furthermore, in this note, the exact specification of the distribution χ will not matter, and we may simply specify an LWE instance by giving the standard deviation σ of χ. We will, furthermore, implicitly assume that χ is centred, i.e., has expectation 0. We may also write LWE in matrix form as $\mathbf{A} \cdot \mathbf{s} + \mathbf{e} \equiv \mathbf{c} \mod q$. The NTRU problem [274] is defined as follows.

Definition 2.2 (NTRU [274])**.** Let n, q be positive integers, $f, g \in \mathbb{Z}_q[x]$ be polynomials of degree n sampled from some distribution χ, subject to f being invertible modulo a polynomial ϕ of degree n, and let $h = g/f \mod (\phi, q)$. The NTRU problem is the problem of finding f, g given h (or any equivalent solution $(x^i \cdot f, x^i \cdot g)$ for some $i \in \mathbb{Z}$).

Concretely, the reader may think of $\phi = x^n + 1$ when n is a power of two and χ to be some distribution producing polynomials with small coefficients. The matrix variant considers $\mathbf{F}, \mathbf{G} \in \mathbb{Z}_q^{n \times n}$ such that $\mathbf{H} = \mathbf{G} \cdot \mathbf{F}^{-1} \mod q$.

2.2 Notation and Preliminaries

All vectors are denoted by bold lower case letters and are to be read as column vectors. Matrices are denoted by bold capital letters. We write a matrix $\mathbf{B}$ as $\mathbf{B} = (\mathbf{b}_0, \ldots, \mathbf{b}_{d-1})$ where $\mathbf{b}_i$ is the ith column vector of $\mathbf{B}$. If $\mathbf{B} \in \mathbb{R}^{m \times d}$ has full-column rank d, the lattice Λ generated by the basis $\mathbf{B}$ is denoted by $\Lambda(\mathbf{B}) = \{\mathbf{B} \cdot \mathbf{x} \mid \mathbf{x} \in \mathbb{Z}^d\}$. A lattice is q-ary if it contains $q\mathbb{Z}^d$ as a sublattice, e.g., $\{\mathbf{x} \in \mathbb{Z}_q^d \mid \mathbf{x} \cdot \mathbf{A} \equiv \mathbf{0}\}$ for some $\mathbf{A} \in \mathbb{Z}^{d \times d'}$. We denote by $(\mathbf{b}_0^\star, \ldots, \mathbf{b}_{d-1}^\star)$ the Gram–Schmidt (GS) orthogonalisation of the matrix $(\mathbf{b}_0, \ldots, \mathbf{b}_{d-1})$. For $i \in \{0, \ldots, d-1\}$, we denote the orthogonal projection to the span of $(\mathbf{b}_0, \ldots, \mathbf{b}_{i-1})$ by π_i; π_0 denotes 'no projection', i.e., the identity. We write $\pi_{\mathbf{v}}$ for the projection orthogonal to the space spanned by $\mathbf{v}$. For $0 \le i < j \le d$, we denote by $\mathbf{B}_{[i:j]}$ the local projected block $(\pi_i(\mathbf{b}_i), \ldots, \pi_i(\mathbf{b}_{j-1}))$, and when the basis is clear from context, by $\Lambda_{[i:j]}$ the lattice generated by $\mathbf{B}_{[i:j]}$. We write $\lg(\cdot)$ for the logarithm to base two.

The Euclidean norm of a vector $\mathbf{v}$ is denoted by $\|\mathbf{v}\|$. The volume (or determinant) of a lattice $\Lambda(\mathbf{B})$ is $\mathrm{vol}(\Lambda(\mathbf{B})) = \prod_i \|\mathbf{b}_i^\star\|$. It is an invariant of the lattice. The first minimum of a lattice Λ is the norm of a shortest non-zero vector, denoted by $\lambda_1(\Lambda)$. We use the abbreviations $\mathrm{vol}(\mathbf{B}) = \mathrm{vol}(\Lambda(\mathbf{B}))$ and $\lambda_1(\mathbf{B}) = \lambda_1(\Lambda(\mathbf{B}))$.

The Hermite constant γ_β is the square of the maximum norm of any shortest

https://doi.org/10.1017/9781108854207.004 Published online by Cambridge University Press

vector in all lattices of unit volume in dimension β:

$$\gamma_\beta = \sup \left\{ \lambda_1^2(\Lambda) \mid \Lambda \in \mathbb{R}^\beta, \mathrm{vol}(\Lambda) = 1 \right\} .$$

Minkowski's theorem allows us to derive an upper bound $\gamma_\beta = O(\beta)$, and this bound is reached up to a constant factor: $\gamma_\beta = \Theta(\beta)$.

2.3 Lattice Reduction: Theory

All lattices of dimension $d \geq 2$ admit infinitely many bases, and two bases $\mathbf{B}, \mathbf{B}'$ generate (or represent) the same lattice if and only if $\mathbf{B} = \mathbf{B}' \cdot \mathbf{U}$ for some unimodular matrix $\mathbf{U} \in \mathrm{GL}_d(\mathbb{Z})$. In other words, the set of (full-rank) lattices can be viewed as the quotient $\mathrm{GL}_d(\mathbb{R}) / \mathrm{GL}_d(\mathbb{Z})$. Lattice reduction is the task of finding a good representative of a lattice, i.e., a basis $\mathbf{B} \in \mathrm{GL}_d(\mathbb{R})$ representing $\Lambda \in \mathrm{GL}_d(\mathbb{R}) / \mathrm{GL}_d(\mathbb{Z})$.

While there exists a variety of formal definitions for what is a good representative, the general goal is to make the Gram–Schmidt basis $\mathbf{B}^\star$ as small as possible. Using the simple size-reduction algorithm (see [454, Algorithm 3]), it is possible to also enforce the shortness of the basis $\mathbf{B}$ itself.

It should be noted that because we have an invariant $\prod_i \|\mathbf{b}_i^\star\| = \mathrm{vol}(\Lambda)$, we cannot make all GS vectors small at the same time, but the goal becomes to balance their lengths. More pictorially, we consider the log profile of a basis as the graph of $(\ell_i = \lg \|\mathbf{b}_i^\star\|)_{i=0\ldots d-1}$ as a function of i. By the volume invariant, the area under this graph is fixed, and the goal of reduction is to make this graph flatter.

A very strong[1] notion of reduction is the Hermite–Korkine–Zolotarev (HKZ) reduction, which requires each basis vector $\mathbf{b}_i$ to be a shortest non-zero vector of the remaining projected lattice $\Lambda_{[i:d]}$. The Block-Korkine–Zolotarev (BKZ) reduction relaxes HKZ, only requiring $\mathbf{b}_i$ to be close-to-shortest in a local 'block'. More formally, we have the following.

Definition 2.3 (HKZ and BKZ [454])**.** The basis $\mathbf{B} = (\mathbf{b}_0, \ldots, \mathbf{b}_{d-1})$ of a lattice Λ is said to be HKZ reduced if $\|\mathbf{b}_i^\star\| = \lambda_1(\Lambda(\mathbf{B}_{[i:d]}))$ for all $i < d$. It is said BKZ reduced with block size β and $\epsilon \geq 0$ if $\|\mathbf{b}_i^\star\| \leq (1 + \epsilon) \cdot \lambda_1(\Lambda(\mathbf{B}_{[i:\min(i+\beta,d)]}))$ for all $i < d$.

In practice, the BKZ algorithm [512, 520] and its terminated variant [257]

[1] HKZ should nevertheless not be considered to be the strongest notion of reduction. Indeed HKZ is a greedy definition, speaking of the shortness of each vector individually. One could go further and require, for example, $\Lambda_{[0:d/2]}$ to be a densest sublattice of Λ [491].

https://doi.org/10.1017/9781108854207.004 Published online by Cambridge University Press

Algorithm 2.1 High-level description of the BKZ algorithm.

```
Input: LLL-reduced lattice basis $\mathbf{B}$ and block size $\beta$
repeat
    for $i \leftarrow 0$ to $d-2$ do
        LLL on $\mathbf{B}_{[i:\min(i+\beta,d)]}$
        $\mathbf{v} \leftarrow$ find a short vector in $\Lambda\left(\mathbf{B}_{[i:\min(i+\beta,d)]}\right)$
        insert $\mathbf{v}$ into $\mathbf{B}$ at index $i$ and handle linear dependencies with LLL
until until no more change
```

are commonly employed to perform lattice reduction. BKZ is also the algorithm we will focus on in this chapter.

The BKZ algorithm will proceed by enforcing the condition $\|\mathbf{b}_i^\star\| \leq (1+\epsilon) \cdot \lambda_1(\Lambda(\mathbf{B}_{[i:\min(i+\beta,d)]}))$ cyclically for $i = 0, \ldots, d-2, 0, \ldots, d-2, 0 \ldots$, see Algorithm 2.1. However, each modification of $\mathbf{b}_i^\star$ may invalidate the same condition for $j \neq i$. The value of ϵ, which allows to account for numerical instability, is typically chosen very close to 0 (say 0.01); we may sometimes omit it and just speak of a BKZ-β reduced basis. Overall, we obtain the following guarantees for the BKZ algorithm.

Theorem 2.4 (BKZ)**.** *If a basis* $\mathbf{B}$ *is BKZ-*β *reduced with parameter* $\epsilon > 0$ *it satisfies*

- $\|\mathbf{b}_0\| \leq \sqrt{(1+\epsilon)\cdot\gamma_\beta}^{\frac{d-1}{\beta-1}+1} \cdot \mathrm{vol}(\Lambda(\mathbf{B}))^{1/d}$ *(Hermite factor) and*
- $\|\mathbf{b}_0\| \leq \left((1+\epsilon)\cdot\gamma_\beta\right)^{\frac{d-1}{\beta-1}} \cdot \lambda_1(\Lambda(\mathbf{B}))$ *(approximation factor).*

Remark. The approximation factor is established in [517], the Hermite factor bound is claimed in [206]. In [257] a bound of $2 \cdot \sqrt{\gamma_\beta}^{\frac{d-1}{\beta-1}+3}$ is established for the terminating variant. In [258] this bound is improved to $K \cdot \sqrt{\beta}^{\frac{d-1}{\beta-1}+0.307}$ for some universal constant K.

Asymptotically, the lattice reduction algorithm with best, known worst-case guarantees is Slide reduction [205]. We refer to its introduction by Gama and Nguyen [205] for a formal definition, which requires the notion of duality, and only state some of its guarantees concerning Gram–Schmidt length here.

Theorem 2.5 (Slide reduction [205])**.** *If a basis* $\mathbf{B}$ *is Slide reduced for parameters* $\beta \mid d$ *and* $\epsilon > 0$ *it satisfies*

- $\|\mathbf{b}_0\| \leq \sqrt{(1+\epsilon)\cdot\gamma_\beta}^{\frac{d-1}{\beta-1}} \cdot \mathrm{vol}(\Lambda(\mathbf{B}))^{1/d}$ *(Hermite factor) and*
- $\|\mathbf{b}_0\| \leq \left((1+\epsilon)\cdot\gamma_\beta\right)^{\frac{d-\beta}{\beta-1}} \cdot \lambda_1(\Lambda(\mathbf{B}))$ *(approximation factor).*

In practice, BKZ is not implemented as in Algorithm 2.1. Most notably, stronger preprocessing than LLL is applied. A collection of improvements to the algorithm (when enumeration is used to instantiate the SVP oracle) are collectively known as BKZ 2.0 [122] and implemented, e.g., in FPLLL [587] and thus Sage [562]. Slide reduction is also implemented in FPLLL.

2.4 Practical Behaviour on Random Lattices

2.4.1 Shape Approximation

The Gaussian heuristic predicts that the number $|\Lambda \cap \mathcal{B}|$ of lattice points inside a measurable body $\mathcal{B} \subset \mathbb{R}^n$ is approximately equal to $\mathrm{vol}(\mathcal{B}) / \mathrm{vol}(\Lambda)$. Applied to Euclidean d-balls, it leads to the following prediction of the length of a shortest non-zero vector in a lattice.

Definition 2.6 (Gaussian heuristic)**.** We denote by $\mathrm{gh}(\Lambda)$ the expected first minimum of a lattice Λ according to the Gaussian heuristic. For a full-rank lattice $\Lambda \subset \mathbb{R}^d$, it is given by

$$\mathrm{gh}(\Lambda) = \left(\frac{\mathrm{vol}(\Lambda)}{\mathrm{vol}(\mathfrak{B})}\right)^{1/d} = \frac{\Gamma\left(1+\frac{d}{2}\right)^{1/d}}{\sqrt{\pi}} \cdot \mathrm{vol}(\Lambda)^{1/d} \approx \sqrt{\frac{d}{2\pi e}} \cdot \mathrm{vol}(\Lambda)^{1/d},$$

where $\mathfrak{B}$ denotes the d-dimensional Euclidean ball. We also denote by $\mathrm{gh}(d)$ the quantity $\mathrm{gh}(\Lambda)$ of any d-dimensional lattice Λ of volume 1: $\mathrm{gh}(d) \approx \sqrt{d/2\pi e}$. For convenience we also denote $\mathrm{lgh}(x)$ for $\lg(\mathrm{gh}(x))$.

Combining the Gaussian heuristic with the definition of a BKZ reduced basis, after BKZ-β reduction we expect

$$\begin{aligned}\ell_i = \lg\left(\lambda_1(\Lambda(\mathbf{B}_{[i:\min(i+\beta,d)]}))\right) &\approx \mathrm{lgh}(\min(\beta, d-i)) + \frac{\lg\left(\mathrm{vol}(\Lambda(\mathbf{B}_{[i:\min(i+\beta,d)]}))\right)}{\min(\beta, d-i)} \\ &= \mathrm{lgh}(\min(\beta, d-i)) + \frac{\sum_{j=i}^{\min(i+\beta,d)-1} \ell_j}{\min(\beta, d-i)}.\end{aligned}$$

If $d \gg \beta$ this linear recurrence implies a geometric series for the $\|\mathbf{b}_i^\star\|$. Considering one block of dimension β and unit volume, we expect $\ell_i = (\beta - i - 1) \cdot \lg(\alpha_\beta)$ for $i = 0, \ldots, \beta - 1$ and some α_β. We obtain

$$\begin{aligned}\ell_0 = (\beta - 1) \cdot \lg(\alpha_\beta) &\approx \mathrm{lgh}(\beta) + \frac{1}{\beta} \sum_{j=0}^{\beta-1} j \cdot \lg(\alpha_\beta) \\ &= \mathrm{lgh}(\beta) + (\beta-1)/2 \cdot \lg(\alpha_\beta).\end{aligned}$$

https://doi.org/10.1017/9781108854207.004 Published online by Cambridge University Press

Solving for α_β assuming equality we obtain $\alpha_\beta = \mathrm{gh}(\beta)^{2/(\beta-1)}$.

Applying the same argument to a basis in dimension $d \gg \beta$ with $\ell_i = (d - i - 1) \cdot \lg(\alpha_\beta)$ for $i = 0, \ldots, d-1$, we get $\|\mathbf{b}_0\|/\mathrm{vol}(\Lambda)^{1/d} = \alpha_\beta^{d-1}/\alpha_\beta^{(d-1)/2} = \alpha_\beta^{(d-1)/2} = \mathrm{gh}(\beta)^{(d-1)/(\beta-1)}$. This is known as the geometric series assumption (GSA).

Definition 2.7 (GSA [518])**.** Let **B** be a BKZ-β reduced basis of a lattice of volume V. The geometric series assumption states that

$$\lg \|\mathbf{b}_i^\star\| = \ell_i = \frac{d-1-2i}{2} \cdot \lg(\alpha_\beta) + \frac{1}{d} \lg V,$$

where $\alpha_\beta = \mathrm{gh}(\beta)^{2/(\beta-1)}$.

The above assumption is reasonably accurate in the case $\beta \ll d$ (and $\beta \gg 50$), but it ignores what happens in the last $d - \beta$ coordinates. Indeed, the last block is HKZ reduced, and should therefore follow the typical profile of an HKZ reduced basis.

Under the Gaussian heuristic, we can predict the shape $\ell_0 \ldots \ell_{d-1}$ of an HKZ reduced basis, i.e., the sequence of expected norms for the vectors $\mathbf{b}_i^\star$. This, as before, implicitly assumes that all the projected lattices Λ_i also behave as random lattices. The sequence is inductively defined as follows.

Definition 2.8. The (unscaled) HKZ shape of dimension d is defined by the following sequence for $i = 0, \ldots, d-1$:

$$h_i = \mathrm{lgh}(d-i) - \frac{1}{d-i} \sum_{j<i} h_j \,.$$

This leads to the following refinement of the GSA.

Definition 2.9 (Tail-adapted geometric series assumption (TGSA))**.** Let **B** be a BKZ-β reduced basis of a lattice of volume V. The TGSA states that

$$\begin{aligned} \ell_i &= \frac{d-1-2i}{2} \cdot \lg \alpha_\beta + s && \text{if } 0 \le i \le d-\beta\,, \\ \ell_i &= h_{i-(d-\beta)} + \ell_{d-\beta} - h_0 && \text{if } d-\beta \le i < d\,, \end{aligned}$$

where $s \in \mathbb{R}$ is the scaling term such that $\sum \ell_i = \lg V$.

We plot an example for a basis after BKZ reduction under the GSA and the TGSA in Figure 2.1 to illustrate their respective shapes. In Figure 2.1 we chose $d = 2\beta$ to highlight the difference between the two models. As can be seen from that figure, the first few indices of the HKZ shape drop slower than predicted by the GSA and the last indices drop faster.

https://doi.org/10.1017/9781108854207.004 Published online by Cambridge University Press

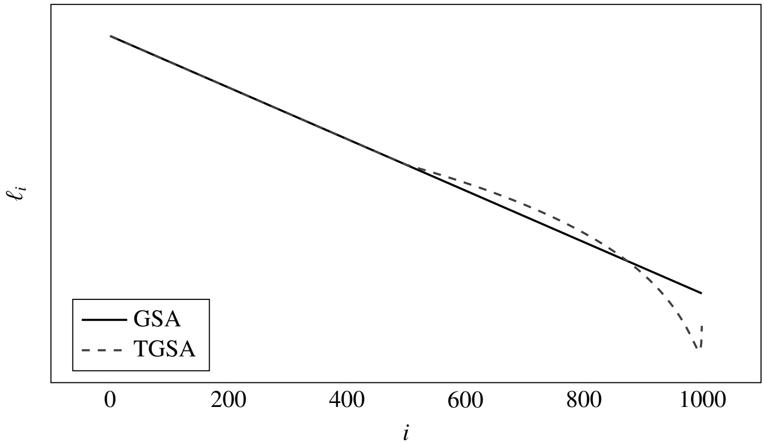

Figure 2.1 GSA and TGSA for $d = 1000$ and $\beta = 500$.

Appealing to the Gaussian heuristic, we may also replace $\sqrt{\gamma_\beta}$, i.e., worst-case bounds, with $\mathrm{gh}(\beta)$, i.e., average-case expectations, in Theorems 2.4 and 2.5. This suggests the following heuristics.

Definition 2.10 (Estimates for block reductions)**.** If a basis $\mathbf{B}$ is BKZ-β reduced for $50 \ll \beta \ll d$ we expect

$$\|\mathbf{b}_0\| \lessapprox \min \begin{cases} \sqrt{\alpha_\beta}^{d-1} \cdot \mathrm{vol}(\Lambda(\mathbf{B}))^{1/d} & \text{(Hermite factor)} \\ \alpha_\beta^{d-1} \cdot \lambda_1(\Lambda(\mathbf{B})) & \text{(approximation factor).} \end{cases}$$

If a basis $\mathbf{B}$ is Slide reduced with parameter β we expect

$$\|\mathbf{b}_0\| \lessapprox \min \begin{cases} \sqrt{\alpha_\beta}^{d-1} \cdot \mathrm{vol}(\Lambda(\mathbf{B}))^{1/d} & \text{(Hermite factor)} \\ \alpha_\beta^{d-\beta} \cdot \lambda_1(\Lambda(\mathbf{B})) & \text{(approximation factor).} \end{cases}$$

The cases over which the minimum is taken define two regimes: the 'Hermite regime' and the 'approximation regime'.

If the lattice is random, then $\lambda_1 \approx \mathrm{gh}(\Lambda)$ and we expect to be in the Hermite regime; the approximation regime is only triggered by the presence of an unusually short vector. In the Hermite regime, we can replace $\lessapprox$ by $\approx$ and we will discuss what happens in the approximation regime further in Section 2.5.4.

We note that the literature usually writes the above approximate equations in terms of the so-called root-Hermite factor $\delta_\beta := \left(\|\mathbf{b}_0\|/\mathrm{vol}(\Lambda)^{1/d}\right)^{1/d}$. We can therefore establish that $\delta_\beta = \sqrt{\alpha_\beta}^{1-1/d} \approx \sqrt{\alpha_\beta}$. We note that making this approximation or not leads to the '-1 discrepancy' blamed on [19] in a footnote

https://doi.org/10.1017/9781108854207.004 Published online by Cambridge University Press

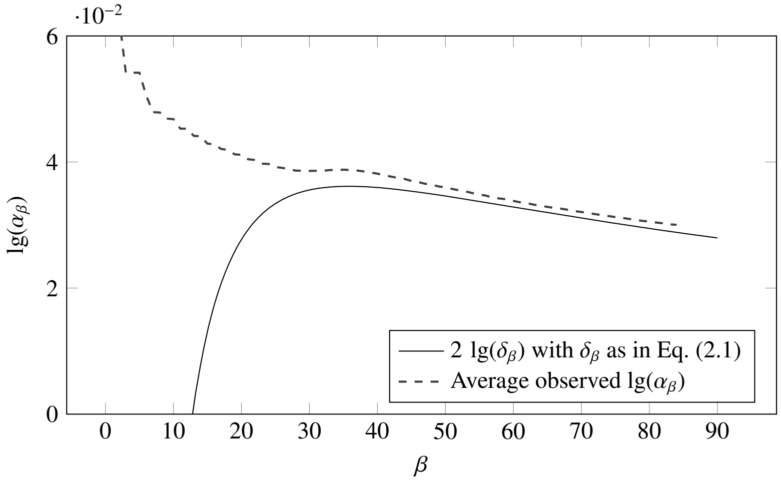

Figure 2.2 Experimentally observed slopes of 16 lattices compared with δ_β as predicted in Eq. (2.1). The input lattices are q-ary lattices in dimension $d = 170$ with $q = 2^{20} - 3$; the experimental $\lg(\alpha_\beta)$ are established using a least-square fit of the log Gram–Schmidt vectors.

of [15]: the analysis of [19] simply did not apply this approximation step. In [121] an expression for δ_β is given as

$$\lim_{\beta\to\infty} \delta_\beta = \left(\frac{\beta}{2\pi e} \cdot (\pi\beta)^{\frac{1}{\beta}}\right)^{\frac{1}{2(\beta-1)}} \tag{2.1}$$

assuming $d \gg \beta$. Experimentally, Eq. (2.1) also holds with good accuracy for $\beta > 50$ and typical d used in cryptography (say, $d \geq c \cdot \beta$ for some $c > 1$). We compare experimentally observed $\lg(\alpha_\beta)$ with the right-hand side of Eq. (2.1) in Figure 2.2.

2.4.2 Simulators

While the (T)GSA provides a first rough approximation of the shape of a basis, it is known to be violated in small dimensions [122]. Indeed, it also does not hold exactly for larger block sizes when d is a small multiple of β, the case most relevant to cryptography. Furthermore, it only models the shape after the algorithm has terminated, leaving open the question of how the quality of the basis improves throughout the algorithm. To address these points, Chen and Nguyen [122] introduced a simulator for the BKZ algorithm which is often referred to as the ‘CN11 simulator’. It takes as input a list of ℓ_i representing the

https://doi.org/10.1017/9781108854207.004 Published online by Cambridge University Press

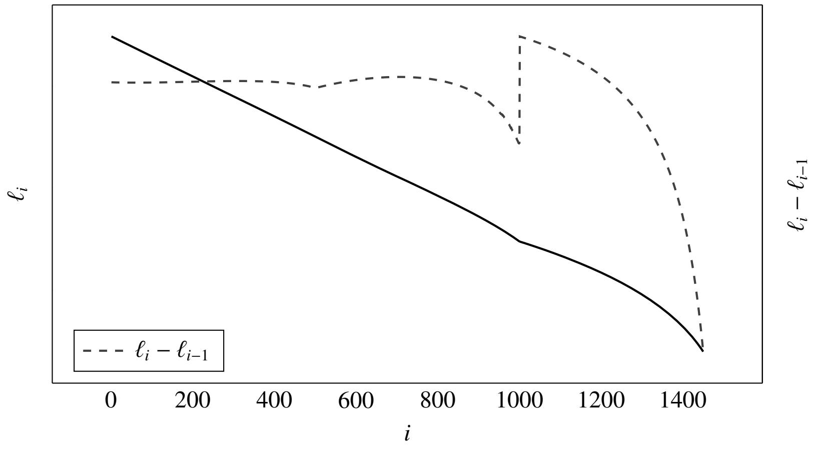

Figure 2.3 CN11 simulator output for $\beta = 500$ on random q-ary lattice in dimension $d = 1500$.

shape of the input basis and a block size β. It then considers blocks $\ell_i, \ldots, \ell_{i+\beta-1}$ of dimension β, establishes the expected norm of the shortest vector in this block using the Gaussian heuristic and updates ℓ_i. To address that the Gaussian heuristic does not hold for $\beta < 50$, the simulator makes use of a precomputed list of the average norms of a shortest vector of random lattices in small dimensions. The simulator keeps on going until no more changes are made or a provided limit on the number of iterations or 'tours' is reached.

The simulator is implemented, for example, in FPyLLL [588] and thus in Sage. In Figure 2.3 we plot the output of the simulator for a basis in dimension 1500 with block size 500 (solid line). We also plot the derivative (dotted line) to illustrate that the GSA also does not hold for $i < d - \beta$. In fact, we observe a ripple effect, with the tail shape exhibiting a damped echo towards the left of the basis. The TGSA is in some sense only a first-order approximation, only predicting the first ripple.

A further simulation refinement was proposed in [27]. Building upon [631], the authors confirmed that the CN11 simulator can be pessimistic about the norm of the first vector output by BKZ. This is because it assumes that the shortest vector in a lattice always has the norm that is predicted by the Gaussian heuristic. By, instead, modelling the norm of the shortest vector as a random variable, the authors were able to model the 'head concavity' behaviour of BKZ as illustrated in Figure 2.4 after many tours and in small block sizes. They also proposed a variant of the BKZ algorithm (pressed-BKZ) that is tailored to exploit this phenomenon. For example, they manage to reach a basis

https://doi.org/10.1017/9781108854207.004 Published online by Cambridge University Press

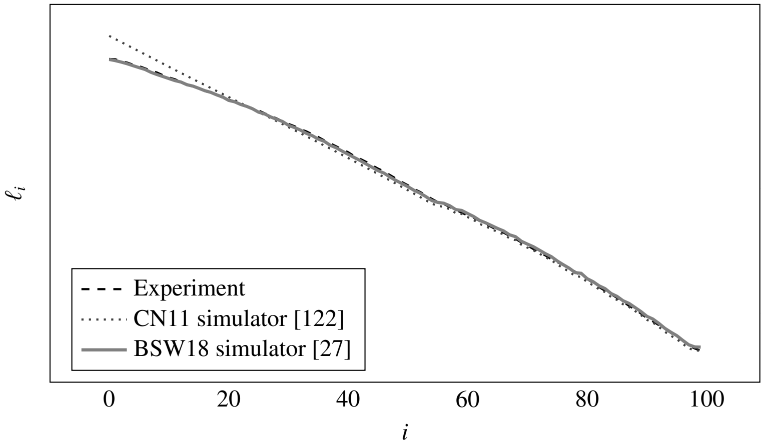

Figure 2.4 Head concavity: dimension $d = 2000$ and block size $\beta = 45$ after 2000 tours, reproduced from [27].

reduction equivalent to BKZ-90 while only using block size 60. The authors note, though, that the head concavity phenomenon does not significantly affect cryptographic block sizes. Indeed, exploiting luck on this random variable seems to be interesting for small block sizes only.

2.4.3 q-ary Lattices and the Z-Shape

Recall that both NTRU and LWE give rise to q-ary lattices. These lattices always contain the vector $(q, 0, \ldots, 0)$ and all its permutations. These so-called 'q-vectors' can be considered short, depending on the parameters of the instance being considered, and might be shorter than what we would expect to obtain following predictions such as the GSA or the TGSA. Furthermore, some of those q-vectors naturally appear in the typical basis construction of q-ary lattices. Even when this is not the case, they can be made explicit by computing the Hermite Normal Form.

To predict lattice reduction on such bases, we may observe that one of the guarantees of the LLL algorithm is that the first vector $\mathbf{b}_0$ never gets longer. For certain parameters this can contradict the GSA. In fact, if $\mathbf{b}_i^*$ does not change for all $i < j$, then $\mathbf{b}_j^*$ cannot become longer either, which means that after the reduction algorithm has completed we may still have many such q-vectors at the beginning of our basis, unaffected by the reduction. It is therefore tempting to predict a piecewise linear profile, with two pieces. It should start with a flat line at $\lg q$, followed by a sloped portion following the predicted GSA slope.

https://doi.org/10.1017/9781108854207.004 Published online by Cambridge University Press

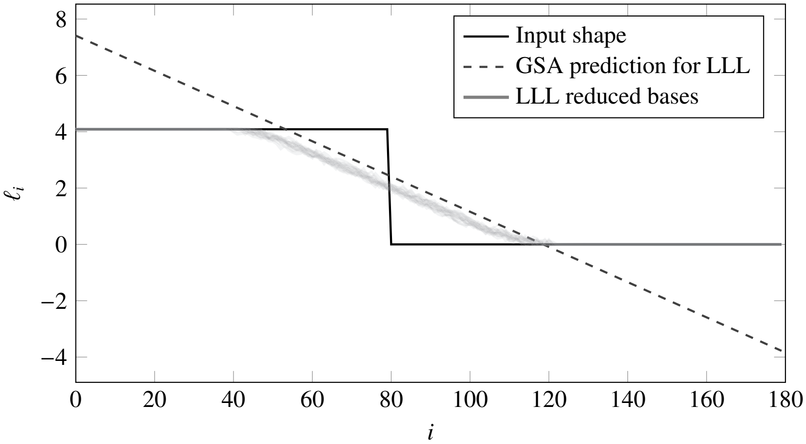

Figure 2.5 GSA and q-ary lattice contradiction. Norms of Gram–Schmidt vectors of 180-dimensional random q-ary lattices with $q = 17$ and volume q^{80}. The grey, blurry lines plot ℓ_i for LLL reduced bases of 16 independent lattices.

In fact, the shape has three pieces, and this is easy to argue for LLL, since LLL is a self-dual algorithm.[2] This means in particular that the last Gram–Schmidt vector cannot get shorter, and following the same argument, we can conclude that the basis must end with a flat piece of 1-vectors. All in all, the basis should follow a Z-shape, and this is indeed experimentally the case [280, 625], as depicted in Figure 2.5, where we picked a small q to highlight the effect. We shall call such a prediction [169, 625] the ZGSA.

It is tempting to extend such a ZGSA model to other algorithms beyond LLL and this has been used for example in [169]. We might also attempt to refine it to a ZTGSA model, where we put an HKZ tail just before the flat section of Gram–Schmidt vectors of norm 1. However, this is a questionable way of reasoning, because BKZ, unlike LLL, is not self-dual. However, it is worth noting that it seems possible to force BKZ to behave in such a way, simply by restricting BKZ to work on the indices up $i < j$, where j is carefully calibrated so that $\|\mathbf{b}_j^\star\| \approx 1$. This is not self-dual, but up to the tail of BKZ, it would produce a Z-shape as well.

Yet, we could also let BKZ work freely on the whole basis, and wonder what would happen. In other words, we may ask whether it is preferable to apply such a restriction to BKZ or not. A natural approach to answering this

[2] This is not entirely true, as the size-reduction condition is not self-dual, but the constraints on the Gram–Schmidt vectors themselves are, which is enough for our purpose.

https://doi.org/10.1017/9781108854207.004 Published online by Cambridge University Press

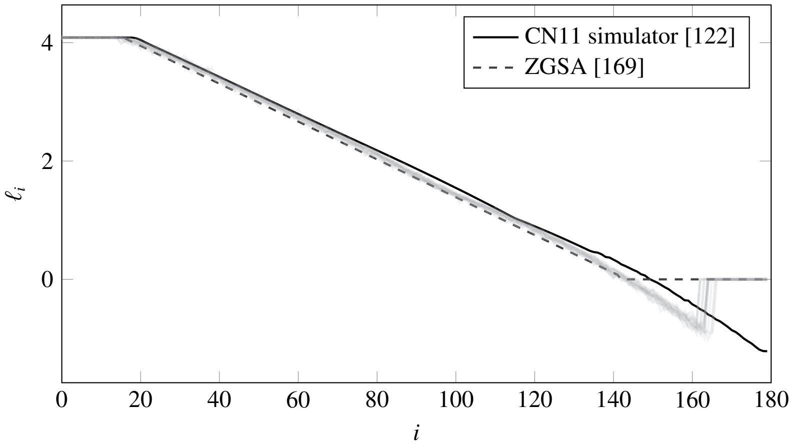

Figure 2.6 BKZ behaviour on q-ary lattice bases with small q. Norms of Gram–Schmidt vectors (grey, blurry lines) after BKZ-65 reduction of 16 180-dimensional q-ary lattices with $q = 17$ and volume q^{80} compared with models from the literature.

question would be to simply use the CN11 simulator, however, it appears that the Z-shape is very poorly simulated. Indeed, while the simulator can easily maintain q-vectors when they are shorter than the one locally predicted by the Gaussian heuristic, the phenomenon on the right end of the Z seems more complicated: some 1-vectors are replaced by Gram–Schmidt vectors of norm strictly less than 1, but not all, see Figure 2.6. Thus, we see the Z-shape known from the literature but with the addition of a kink in the tail block.

Simulating or predicting the behaviour of BKZ on q-ary lattices is still open, but it would allow addressing the question if it can be exploited. A partial answer seems obtainable by defining a specialised variant of the Gaussian heuristic that takes orthogonal sublattices into account. Although we are not certain that a deeper study of this phenomenon would lead to cryptanalytic advances, it is nevertheless quite frustrating to have to resort to Z(T)GSA without a perfect understanding of the behaviour of lattice reduction on this class of lattices.

2.4.4 Random Blocks?

The heuristic analysis of BKZ is based on the assumption that each sublattice considered by the algorithm ‘behaves like a random lattice’ (strong version), or at least that the expectation or distribution of its shortest vector is the same as for a random lattice (weak version).

More formally, we would have to define the notion of a random lattice,

https://doi.org/10.1017/9781108854207.004 Published online by Cambridge University Press

invoking the Haar measure. However, we can nevertheless interrogate this heuristic without going into those details here. Indeed, as we can see in Figure 2.2, the predicted slopes below dimension 30 are far from the actual behaviour. In fact, the predictions for small block sizes are nonsensical as they predict a flatter slope as β decreases below 30 and even an inversion of the slope below block size ≈ 10.

Although we can observe the prediction and the observation converging for block sizes above 50, what level of precision do we attribute to those predictions? Given the phenomena perturbing the GSA surveyed in this chapter (heads, tails, ripples), how pertinent are the data from Figure 2.2? Pushing experimental evidence a bit further would be reassuring here: although we do not expect surprises, it would be good to replace this expectation with experimental evidence.

But, more conceptually, we note that making the strong version of the heuristic assumption (each block behaves like a random lattice) is self-contradictory. Indeed, the model leads us to conclude that the shape is essentially a line, at least when $\beta \ll d$ and the considered block $\mathbf{B}_{[\kappa:\kappa+\beta]}$ is far from the head and the tail, i.e., $\kappa \gg \beta$, $d - \kappa \gg \beta$. But this block, like all other blocks, is fully HKZ-reduced: since $\mathbf{b}^{\star}_{\kappa+i}$ is a shortest vector of $\Lambda(\mathbf{B}_{[\kappa+i:\kappa+i+\beta]})$, it is also a shortest vector of $\Lambda(\mathbf{B}_{[\kappa+i:\kappa+\beta]})$. Yet, HKZ-reduced bases of random lattices have a concave shape not a straight slope.

We do not mean to discredit the current methodology to predict attacks on lattice-based schemes; current evidence does suggest predictions such as Eq. (2.5) in Section 2.5.4 are reasonably precise. In particular, the above argument does not rule out the weak version of the hypothesis: the shortest vector of those non-random blocks may still have an expected length following the Gaussian heuristic. In fact, for random lattices, it is known that the length of the shortest vector is increasingly concentrated around the Gaussian heuristic; there may be increasingly fewer lattices that fall far from it, which may explain why a bias in the distribution of the lattices themselves does not translate to a bias on the length of its shortest vector.

However, we wish to emphasise that the question of the distribution of those local blocks is at the centre of our understanding of lattice reduction algorithms but remains open. While even formulating specific yet relevant questions seems hard, this phenomenon suggests itself as a challenging but pressing area to study.

2.5 Behaviour on LWE Instances

We can reformulate the matrix form of the LWE equation $\mathbf{c} - \mathbf{A} \cdot \mathbf{s} \equiv \mathbf{e} \bmod q$ as a linear system over the integers as

$$\begin{pmatrix} q\mathbf{I} & -\mathbf{A} \\ 0 & \mathbf{I} \end{pmatrix} \cdot \begin{pmatrix} * \\ \mathbf{s} \end{pmatrix} + \begin{pmatrix} \mathbf{c} \\ \mathbf{0} \end{pmatrix} = \begin{pmatrix} \mathbf{e} \\ \mathbf{s} \end{pmatrix}$$

or homogeneously as

$$\mathbf{B} = \begin{pmatrix} q\mathbf{I} & -\mathbf{A} & \mathbf{c} \\ 0 & \mathbf{I} & 0 \\ 0 & 0 & t \end{pmatrix}, \qquad \mathbf{B} \cdot \begin{pmatrix} * \\ \mathbf{s} \\ 1 \end{pmatrix} = \begin{pmatrix} \mathbf{e} \\ \mathbf{s} \\ t \end{pmatrix}, \tag{2.2}$$

where t is some chosen constant and $*$ stands in for an arbitrary vector. In other words, there exists an element in the lattice spanned by $\mathbf{B}$ with expected norm $\sqrt{(n+m) \cdot \sigma^2 + t^2}$. Let $d = n + m + 1$. If we have $\sqrt{(n+m) \cdot \sigma^2 + t^2} < \mathrm{gh}(\Lambda(\mathbf{B})) \approx \sqrt{\frac{d}{2\pi \cdot e}} \cdot q^{n/d}$ then $\mathbf{B}$ admits an unusually short vector. With a slight abuse of notation, we will refer to the (column) vector $(\mathbf{e}^T, \mathbf{s}^T, t)^T$ simply as $(\mathbf{e}, \mathbf{s}, t)$.

Remark. We note that when $t \nmid q$ then $\Lambda(\mathbf{B})$ is not a q-ary lattice as, in this case, $(0, \ldots, 0, q)^T \notin \Lambda$. The reader may think $t = 1$, which is commonly used in practice albeit being slightly worse compared to $t = \sigma$, which maximises $\lambda_2(\Lambda)/\lambda_1(\Lambda)$ and which makes the problem easier.

2.5.1 Kannan Embedding

More generally, we can consider this approach to solving LWE as solving an instance of the bounded distance decoding problem (BDD) using a solver for the unique shortest vector problem.

Definition 2.11 (α-Bounded Distance Decoding (BDD_α))**.** Given a lattice basis $\mathbf{B}$, a vector $\mathbf{t}$, and a parameter $0 < \alpha < 1/2$ such that the Euclidean distance $\mathrm{dist}(\mathbf{t}, \mathbf{B}) < \alpha \cdot \lambda_1(\mathbf{B})$, find the lattice vector $\mathbf{v} \in \Lambda(\mathbf{B})$ that is closest to $\mathbf{t}$.

Remark. In our definition above we picked $\alpha < 1/2$, which guarantees a unique solution. The problem can be generalised to $1/2 < \alpha \leq 1$ where we expect a unique solution with high probability.

We can view LWE with a fixed number of samples as an instance of BDD (with overwhelming probability over the choice of the samples). Asymptotically, for any polynomially bounded $\gamma \geq 1$ there is a reduction from $\mathrm{BDD}_{1/(\sqrt{2}\gamma)}$ to uSVP_γ [26]. The unique shortest vector problem (uSVP) is defined as follows.

https://doi.org/10.1017/9781108854207.004 Published online by Cambridge University Press

Definition 2.12 (γ-unique Shortest Vector Problem (uSVP_γ))**.** Given a lattice Λ such that $\lambda_2(\Lambda) > \gamma \cdot \lambda_1(\Lambda)$ find a non-zero vector $\mathbf{v} \in \Lambda$ of length $\lambda_1(\Lambda)$.

This reduction is essentially the embedding technique, due to Kannan [311], presented at the beginning of this section, combined with some tricks to improve the parameters of the reduction. For the remaining of this section, we will discuss how strong we require lattice reduction to be to find a unique shortest vector which can then be used to recover the secret values of an LWE instance.

2.5.2 Asymptotic Handwaving

Recall that in Definition 2.10 two regimes are defined, the Hermite regime and the approximation regime. Now, consider decision LWE. On the one hand, when $\mathbf{c}$ is just a random vector then the lattice spanned by $\mathbf{B}$ is a random q-ary lattice and we are in the Hermite regime, i.e., $\lambda_1(\Lambda(\mathbf{B})) \approx \mathrm{gh}(\Lambda(\mathbf{B}))$. On the other hand, when $\mathbf{c}$ is formed as in LWE then $\Lambda(\mathbf{B})$ contains $(\mathbf{e}, \mathbf{s}, t)$ and we expect $\lambda_1(\Lambda(\mathbf{B})) = \|(\mathbf{e}, \mathbf{s}, t)\|$. Now, if this is sufficiently smaller than $\mathrm{gh}(\Lambda(\mathbf{B}))$ then we are in the approximation regime. Thus, one way to distinguish LWE from uniform is to detect the 'phase transition' between the two regimes, the point when the approximation regime 'kicks in', i.e., when

$$\sqrt{\alpha_\beta}^{2d-2} \cdot \lambda_1(\Lambda(\mathbf{B})) < \sqrt{\alpha_\beta}^{d-1} \cdot \mathrm{vol}(\Lambda(\mathbf{B}))^{1/d} \text{ for BKZ and}$$
$$\sqrt{\alpha_\beta}^{2d-2\beta} \cdot \lambda_1(\Lambda(\mathbf{B})) < \sqrt{\alpha_\beta}^{d-1} \cdot \mathrm{vol}(\Lambda(\mathbf{B}))^{1/d} \text{ for Slide reduction.}$$

Rearranging we obtain the following success conditions

$$\lambda_1(\Lambda(\mathbf{B})) < \sqrt{\alpha_\beta}^{1-d} \cdot \mathrm{vol}(\Lambda(\mathbf{B}))^{1/d} \text{ with BKZ and} \tag{2.3}$$
$$\lambda_1(\Lambda(\mathbf{B})) < \sqrt{\alpha_\beta}^{2\beta-d-1} \cdot \mathrm{vol}(\Lambda(\mathbf{B}))^{1/d} \text{ with Slide reduction} \tag{2.4}$$

for solving decision LWE in block size β.

2.5.3 The 2008 Estimates

Gama and Nguyen [206] performed experiments in small block sizes to establish when lattice reduction finds a unique shortest vector. They considered two classes of semi-orthogonal lattices and Lagarias–Odlyzko lattices [350] which permit to estimate the gap $\lambda_2(\Lambda)/\lambda_1(\Lambda)$ between the first and second minimum of the lattice. For all three families, it was observed in [206] that LLL and BKZ seem to recover a unique shortest vector with high probability whenever $\lambda_2(\Lambda)/\lambda_1(\Lambda) \geq \tau_\beta \cdot \sqrt{\alpha_\beta}^d$, where $\tau_\beta < 1$ is an empirically determined constant that depends on the lattice family, algorithm and block size used.

https://doi.org/10.1017/9781108854207.004 Published online by Cambridge University Press

In [11] an experimental analysis of solving LWE based on the same estimate was carried out for lattices of the form of Eq. (2.2). This lattice contains an unusually short vector $\mathbf{v} = (\mathbf{e}, \mathbf{s}, t)$ of squared norm $\|\mathbf{v}\|^2 = \|\mathbf{s}\|^2 + \|\mathbf{e}\|^2 + t^2$ and we expect $\lambda_1(\Lambda)^2 = \|\mathbf{v}\|^2$. Thus, when $t^2 \approx \|\mathbf{e}\|^2 + \|\mathbf{s}\|^2$ respectively $t = 1$ this implies $\lambda_1(\Lambda) \approx \sqrt{2\,(n+m)} \cdot \sigma$ respectively $\lambda_1(\Lambda) \approx \sqrt{n+m} \cdot \sigma$. The second minimum $\lambda_2(\Lambda)$ is assumed to correspond to the Gaussian heuristic for the lattice (a more refined argument would consider the Gaussian heuristic of $\Lambda' = \pi_{\mathbf{v}}(\Lambda)$, but these quantity are very close for relevant parameters). Experiments in [11] using LLL and BKZ with small block sizes (5 and 10) were interpreted to matched the 2008 estimate, providing constant values for τ_β for lattices of the form of Eq. (2.2), depending on the chosen algorithm, for a 10 per cent success rate. Overall, τ_β was found to lie between 0.3 and 0.4 when using BKZ.

We note that we may interpret this observation as being consistent with Inequality (2.3).

2.5.4 The 2016 Estimate

The 2008 estimates offer no insight into why the algorithm behaves the way it does but only provide numerically established constants that seem to somewhat vary with the algorithm or the block size. In [19] an alternative estimate was outlined. The estimate predicts that $(\mathbf{e}, \mathbf{s}, t)$ can be found if

$$\sqrt{\beta/d} \cdot \|(\mathbf{e}, \mathbf{s}, t)\| \approx \sqrt{\beta \cdot \sigma^2} < \sqrt{\alpha_\beta}^{2\beta-d-1} \cdot \mathrm{Vol}(\Lambda(\mathbf{B}))^{1/d}\,, \qquad (2.5)$$

under the geometric series assumption (until a projection of the unusually short vector is found). The right-hand side of the inequality is the expected norm of the Gram–Schmidt vector at index $d-\beta$ (see Definition 2.7). The left-hand side is an estimate for $\|\pi_{d-\beta}\,((\mathbf{e}, \mathbf{s}, t))\,\|$. If the inequality holds then $\pi_{d-\beta}\,((\mathbf{e}, \mathbf{s}, t))$ is a shortest vector in $\mathbf{B}_{[d-\beta:d]}$ and will thus be found by BKZ and inserted at index $d-\beta$. This is visualised in the top part of Figure 2.7. Subsequent calls to an SVP oracle on $\mathbf{B}_{[d-2\beta+1:d-\beta+1]}$ would insert $\pi_{d-2\beta+1}\,((\mathbf{e}, \mathbf{s}, t))$ at index $d - 2\beta + 1$ etc.

The 2016 estimate was empirically investigated and confirmed in [15]. The authors ran experiments in block sizes up to 78 and observed that a BKZ managed to recover the target vector with good probability as predicted in [19]. An example is given in the bottom part of Figure 2.7. Furthermore, they showed (under the assumption that vectors are randomly distributed in space) that once BKZ has set $\mathbf{b}_i^\star = \pi_{d-\beta}\,((\mathbf{e}, \mathbf{s}, t))$, calls to LLL are expected to suffice to recover $(\mathbf{e}, \mathbf{s}, t)$ itself.

https://doi.org/10.1017/9781108854207.004 Published online by Cambridge University Press

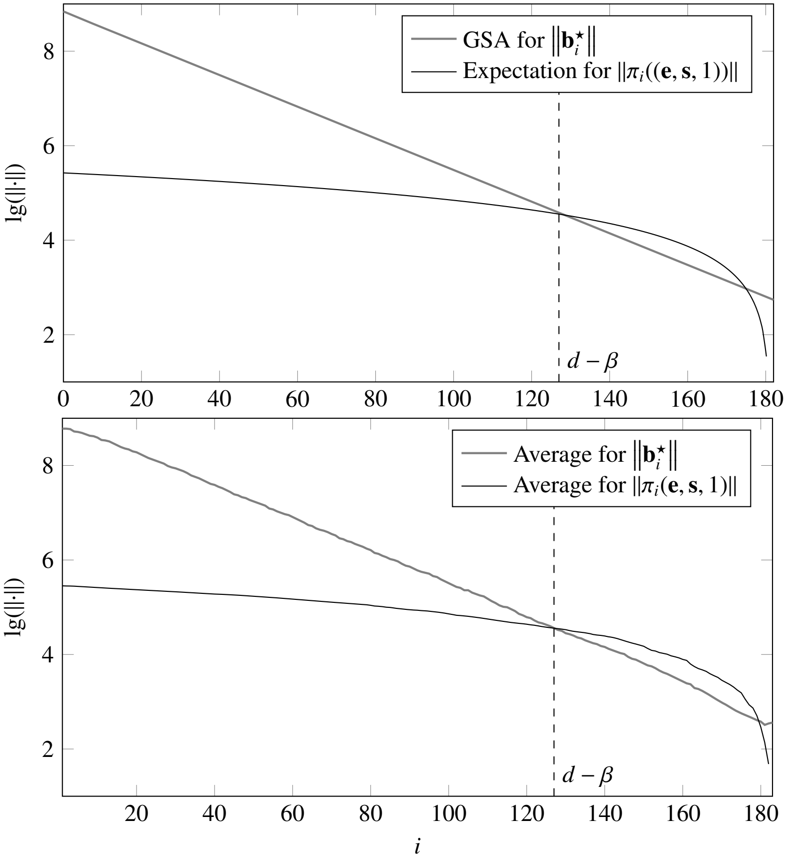

Figure 2.7 (The 2016 estimate.) Expected and observed norms for lattices of dimension $d = 183$ and volume q^{m-n} after BKZ-β reduction for LWE parameters $n = 65, m = 182, q = 521$, standard deviation $\sigma = 8/\sqrt{2\pi}$ and $\beta = 56$ (minimal (β, m) such that Inequality (2.5) holds). Average of Gram–Schmidt lengths is taken over 16 BKZ-β reduced bases of random q-ary lattices, i.e., without an unusually short vector. Reproduced from [15].

Comparing Inequality (2.5) with Inequalities (2.3) and (2.4) we note that it more closely resembles the prediction for Slide reduction rather than for BKZ, despite the rationale and experimental evidence being obtained for BKZ. This suggests that the average behaviour of BKZ and Slide reductions in the approximation factor regime is roughly the same, despite different worst-case bounds being proven. Furthermore, we note that Inequality (2.5) gains an additional factor of $\sqrt{\beta/d}$ compared with Inequality (2.4).

https://doi.org/10.1017/9781108854207.004 Published online by Cambridge University Press

2.5.5 Further Refinements

On the other hand, the authors of [15] also observed that the algorithm behaves somewhat better than predicted. That is, they managed to solve the underlying instances using block sizes somewhat smaller than required to make Inequality (2.5) hold.

This is attributed to a 'double intersection' in [15]. As illustrated in Figure 2.7, the projection of the target vector and the norms of the Gram–Schmidt vectors may intersect twice: once at index $d-\beta$ and once close to index d, say at index $d-o$ for some small o. Applying the same reasoning as above, we expect $\pi_{d-o}((\mathbf{e},\mathbf{s},t))$ to be inserted as $\mathbf{b}^{\star}_{d-o}$. Thus, we expect a subsequent SVP call at index $d-\beta-o$ to recover and insert $\pi_{d-\beta-o}((\mathbf{e},\mathbf{s},t))$. Alternatively, an SVP call in dimension $\beta-o$ at index $d-\beta$ could now recover $\pi_{d-\beta}((\mathbf{e},\mathbf{s},t))$ since this vector is $\in \Lambda_{[d-\beta:d-o]}$. However, it is noted in [15] that this 'double intersection' phenomenon does not occur for typical cryptographic parameters.

Another source of imprecision when applying Inequality (2.5) is that it assumes the GSA (before an unusually short vector is found), replacing this assumption with a BKZ simulator produces refined estimates.

But there seems to be another subtle phenomenon at play. In [148] it is noted that, for very small block sizes β, the prediction of [15] is, on the contrary, too optimistic. The reason is that, while the projected vector $\pi_{d-\beta}((\mathbf{e},\mathbf{s},t))$ may be detected with good probability at position $d-\beta$, we require a bit more luck to lift correctly, i.e., to recover the full vectors $(\mathbf{e},\mathbf{s},t)$ from its projection. Instead, a probabilistic model is proposed, to account for both initial detection and lifting, and this prediction seems to fit very well with experiments; see Figure 2.8.

Balancing Costs It should be mentioned that just running BKZ is not the optimal strategy to solve uSVP instances. Indeed, having spent $O(d)$ many SVP-β calls pre-processing the whole basis, this strategy hopes for the last such SVP call to essentially produce the solution. An improved strategy instead balances the cost of the pre-processing step and the final search step. Therefore, it could, for example, be natural to do a last call to SVP-β' for β' slightly larger than β; this has for example been implemented with sieving in [17] to break Darmstadt LWE challenges [230] and was already standard in the enumeration literature [396].

The optimal strategy is, therefore, more difficult to predict, and hardness estimates often rely on scripts that numerically optimise the various parameters of the algorithm based on assumptions such as the relative costs of running SVP in slightly larger or smaller dimension, the number of calls to an

https://doi.org/10.1017/9781108854207.004 Published online by Cambridge University Press

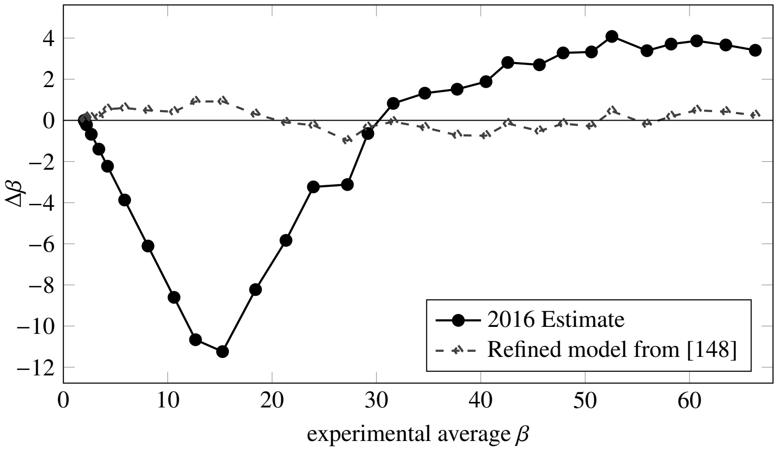

Figure 2.8 The difference $\Delta\beta$ = real − predicted, as a function of the average experimental β. The experiment consists in running a single tour of BKZ-β for $\beta = 2, 3, 4, \ldots$ until the secret short vector is found. This was averaged over 256 many LWE instances per data point, for parameters $q = 3301$, $\sigma = 20$ and $n = m \in \{30, 32, 34, \ldots, 88\}$. Reproduced from [148].

SVP oracle required to achieve a given root-Hermite factor, etc. To avoid this complication, some designers instead opt for accounting only for the cost of a single call to SVP-β when even considering several tours of BKZ-β (a simplification introduced as the 'core SVP hardness' in [19]). In this model, the issue of balancing costs between β' and β does not arise, i.e., $\beta' = \beta$ is optimal and the attack cost is bounded from below by the cost of one call to SVP-β on a BKZ-β reduced basis.

2.6 Behaviour on NTRU Instances

To solve NTRU (Definition 2.2) we may consider the lattice

$$\Lambda^q_{\mathbf{H}} = \left\{(\mathbf{x}, \mathbf{y}) \in \mathbb{Z}^{2n} \text{ s.t. } \mathbf{H} \cdot \mathbf{x} - \mathbf{y} = \mathbf{0} \bmod q\right\}, \tag{2.6}$$

where $\mathbf{H}$ is the matrix associated with multiplication by h modulo ϕ, i.e., the columns of $\mathbf{H}$ are spanned by the coefficients of $x^i \cdot h \bmod \phi$ for $i = 0, \ldots, n-1$. The lattice $\Lambda^q_{\mathbf{H}}$ is spanned by

$$\mathbf{B} = \begin{pmatrix} q\mathbf{I} & \mathbf{H} \\ 0 & \mathbf{I} \end{pmatrix}$$

https://doi.org/10.1017/9781108854207.004 Published online by Cambridge University Press

and contains a short vector $(\mathbf{f}, \mathbf{g})$. This can be observed by multiplying the basis by $(\mathbf{f}, *)$ from the right, where $*$ represents the vector performing modular reduction modulo q and where $\mathbf{f}$ respectively $\mathbf{g}$ is the coefficient vector of f respectively g. If $\|(\mathbf{f}, \mathbf{g})\|$ is much smaller than $\mathrm{gh}(\Lambda_{\mathbf{H}}^{q}) \approx \sqrt{n/(\pi e)} \cdot \sqrt{q}$ then this lattice contains an unusually short vector. Indeed, it also contains all vectors corresponding to 'rotations' of (f, g), i.e., $(x^i \cdot f \bmod \phi, x^i \cdot g \bmod \phi)$ for $i = 0, \ldots, n-1$ and their integral linear combinations. In other words, the NTRU lattice contains a dense sublattice.

2.6.1 NTRU as uSVP

Considering NTRU as the problem of recovering an unusually short vector in the NTRU lattice was already done in the initial NTRU paper [275]. Also, the original NTRU paper [275] discussed an observation from [139] (analysing [274]) that an attacker does not need to recover f, g exactly, but that any sufficiently small multiple of f suffices to break the scheme. For the uSVP case the hardness of the problem was related to $\mathrm{gh}(\Lambda)/\lambda_1(\Lambda)$ where $\lambda_1(\Lambda) = \|(\mathbf{f}, \mathbf{g})\|$. When considering message recovery instead of key recovery, a related quantity is considered. We may a posteriori reinterpret this as framing attacks on NTRU in the framework of the '2008 estimate' (see Section 2.5.3) but replacing $\lambda_2(\Lambda)$ by $\mathrm{gh}(\Lambda)$. This approach became a common way of reasoning about NTRU lattices; see, e.g. [168]. Yet the validity of this approach is doubtful, as in NTRU lattices we have $\lambda_2(\Lambda) = \lambda_1(\Lambda)$ in contrast to the lattices arising for LWE. In this context, we note that the early study of May and Silverman [410] massaged the lattice to decrease the NTRU lattice dimension while also eliminating all but one of the NTRU short vectors.

The 2016 estimate (see Section 2.5.4) sidesteps this discussion on whether $\lambda_2(\Lambda)$ matters, as the heuristic reasoning here does not involve this quantity. This estimate also ended up being used for estimating the hardness of breaking NTRU [633, Section 6.4.2]. More recently the framework proposed in [148] allowed us to revisit the tricks of May and Silverman [410], and it was concluded that this trick was slightly counterproductive. Indeed, the probabilistic model permits to account for the cumulated probabilities of detecting any of those short vectors in the full lattice, and this is slightly easier than finding the (up to signs) unique short vector of the massaged lattice.

Indeed, another line of works showed that the presence of many short vectors can make the problem exponentially easier, at least in some 'overstretched' regimes. These works [14, 123, 217, 324] seem to suggest that simple encryption schemes should not be affected at all, but we will argue that the exact crossover point remains to be determined.

https://doi.org/10.1017/9781108854207.004 Published online by Cambridge University Press

2.6.2 Attacks on Overstretched NTRU

In this last section, we cover an attack that exploits the fact that NTRU lattices hide not one but many unexpectedly short vectors, yielding an unexpectedly dense sublattice. If the right conditions are met then it turns out that this dense sublattice is easier to uncover than the individual vectors spanning it.

This, however, is a fairly a-posteriori view of this discovery. At first, this weakness was associated not primarily with a density property, but more with an algebraic structure property: namely, the presence of subfields in NTRU. The idea of exploiting this structure had been considered as soon as 2002, by Gentry, Szydly, Jonsson, Nguyen and Stern [217, Section 6]; but it was quickly abandoned: yes, NTRU keys can be normed down to a subfield and still yield valid NTRU keys, but this trade-off of dimension versus approximation factor did not seem advantageous for the actual NTRUEncrypt parameters.

When Bai and ourselves explored this idea again [14] (independently, Cheon, Jeong and Lee [123] also explored a closely related idea), the situation was rather different: NTRU was not just a single scheme with a few parameter sets, it was a parameterised assumption with increasing popularity for building homomorphic encryption schemes. In these newly considered regimes the trade-off mentioned above seemed on the contrary quite advantageous. We, therefore, claimed asymptotic improvements over the natural lattice reduction attack, which – depending on the parameters – could decrease the costs of the attacks from exponential to sub-exponential or even polynomial.

This claimed improvement was soon challenged by Kirchner and Fouque [324]. Our mistake was not the complexity of our new algorithm but rather the fact that the complexity of straight-up lattice reduction attacks was much better than expected on such overstretched NTRU instances. They claimed that the old attack should behave as well as the new one, and – with minor performance-enhancing tricks – were able to demonstrate this in practice. In conclusion, the new algorithm we invented was completely useless, and old algorithms performed just as well, if not better, and were more generally applicable. We found solace in the belief that the results of Kirchner and Fouque may not have been discovered without our algebraic detour.

The Subfield Attack

The key idea of this attack is as follows: the relation $h = f/g \bmod q$ between the public key h and the private key (f, g) can be normed down to a smaller field; furthermore, if f and g are short enough, their norms in a smaller field will also be somewhat short. Therefore, one may hope to attack the problem in a subfield and lift back the solution. We note that in the case of cyclotomic

https://doi.org/10.1017/9781108854207.004 Published online by Cambridge University Press

number fields, there is always at least one non-trivial subfield, namely the maximal totally real subfield $\mathbb{K}^+$, of relative rank $r = [\mathbb{K} : \mathbb{K}^+] = 2$. In the case of power of two cyclotomic number fields ($n = 2^k$), one chooses the subfield to tune r to any power of 2 less than n. On the contrary, this approach is not directly applicable to fields as chosen in [55].

In more detail, let $\mathbb{K}$ be a number field ($\mathbb{K} = \mathbb{Q}(x)/(\phi(x))$, where ϕ comes from Definition 2.2), and for simplicity let us assume that $\mathbb{K}$ is a cyclotomic number field. Let $\mathbb{L}$ be a subfield with relative rank $r = [\mathbb{K} : \mathbb{L}]$, and let N denote the relative norm $N : \mathbb{K} \to \mathbb{L}$, defined by $N(x) = \prod_a a(x)$, where a ranges over all the automorphisms of $\mathbb{K}$ that are identity over $\mathbb{L}$. Defining $f' = N(f)$, $g' = N(g)$ and $h' = N(h)$, we note that $h' = f'/g' \bmod q$ still holds over $\mathbb{L}$. Furthermore, if f, g have lengths roughly $\sqrt{n} \cdot \sigma$, we expect f', g' to have lengths roughly $(\sqrt{n} \cdot \sigma)^r$.

On the other hand, the dimension of the normed-down NTRU lattice is $2n/r$ and its volume is $q^{n/r}$. The original article [14] reasons more formally, using the approximate factor bound of lattice reduction; however, here we will give a simplified and more heuristic exposition. Roughly, using either the 2008 estimate or the 2016 estimate, we expect to solve this instance using a block size β such that

$$(\sqrt{n} \cdot \sigma)^r \cdot \delta_\beta^{2n/r} \le \sqrt{q}.$$

For $\sigma = \mathrm{poly}(n)$, the subfield attack [14] obtains the asymptotic success condition

$$\frac{\beta}{\lg \beta} = \Theta\left(\frac{n}{r \lg q - r^2 \lg n}\right)$$

assuming $r \lg q - r^2 \lg n > 0$.

Parameterising the attack to not use a subfield ($r = 1$) should therefore require $\beta = \tilde{\Theta}(n/\log q)$, while choosing a relative rank $r = \Theta(\log q/\log n)$ leads to $\beta = \tilde{\Theta}(n/\log^2 q)$. For schemes that use large moduli such as fully homomorphic schemes [94, 399] or candidate cryptographic multi-linear maps [208], this therefore makes a significant difference; both in practice and in theory.

Full Secret Reconstruction It should be noted that finding f', g' does not lead to a full recovery of the original secret. However, we can still reconstruct a small multiple $\alpha(f, g)$ of the original secret key (f, g), by constructing $(f', g' \cdot h/h')$. This is typically enough to break encryption schemes. If we insist on recovering the original key (f, g), this intermediate information is still helpful. For example, repeating the attack with a rerandomised initial basis, we may

https://doi.org/10.1017/9781108854207.004 Published online by Cambridge University Press

recover the exact lattice generated by the secret key $(f, g)^T \cdot \mathcal{O}_{\mathbb{K}}$. Recovering (f, g) is now much easier; it can be done with an algorithm for the Principal Ideal Problem, and this is classically sub-exponential time [60], and quantumly polynomial time [59].

The Dense Sublattice Attack

We will now explain why the above subfield attack was a detour to the discovery of a much more general result by Kirchner and Fouque [324]. In a sense, LLL and BKZ are rather clever algorithms and what we can try to make more visible to them via algebraic massaging of the lattice at hand was already geometrically obvious to them: there is a particularly dense sublattice to be found inside NTRU instances. This version of the attack is therefore not prevented by choosing a number field as in [55], or even by going for a matrix version of NTRU without any underlying number field.

To prove that LLL can indeed uncover this hidden dense sublattice, let us first go back to the (worst-case) argument to prove that LLL can solve a unique-SVP instance when $\lambda_2(\Lambda)/\lambda_1(\Lambda) > (4/3 + \epsilon)^{d/2}$.

It follows from the inequality $\lambda_1(\Lambda) \geq \min_i \|\mathbf{b}_i^\star\|$, which is obtained by writing a shortest vector $\mathbf{v}$ as $\mathbf{v} = \sum v_i \mathbf{b}_i^\star$ and noting that $\mathbf{v}$ must be longer than $\mathbf{b}_j^\star$ where j is the largest index such that $v_j \neq 0$. From there, we argue that

$$\|\mathbf{b}_1\| \leq (4/3 + \epsilon)^{d/2} \min_i \|\mathbf{b}_i^\star\| \leq (4/3 + \epsilon)^{d/2} \lambda_1(\Lambda) < \lambda_2(\Lambda)\ .$$

Recall that we can make an even simpler case that LLL or BKZ must distinguish this lattice from random without having to go through the full argument. Indeed, let us simply note that, for a random lattice, we expect a particular shape for the basis, say following ZGSA or ZTGSA. But for a large enough β, the prediction for the shape becomes incompatible with the constraint that $\lambda_1(\Lambda) \geq \min_i \|\mathbf{b}_i^\star\|$. In such cases, LLL and BKZ must, therefore, behave differently, and this is easily seen by just looking at the shape: the NTRU lattice has been distinguished from random.

The analysis of Kirchner and Fouque follows essentially from the same kind of argument, generalising the invariant $\lambda_1(\Lambda) \geq \min_i \|\mathbf{b}_i^\star\|$. Here, we can read '$\lambda_1(\Lambda)$' as the determinant of the densest one-dimensional sublattice; a k-dimensional variant of the inequality was given by Pataki and Tural.

Lemma 2.13 ([469, Lemma 1]). *Let Λ be a d-dimensional lattice, and $\mathbf{b}_0, \ldots, \mathbf{b}_{d-1}$ be any basis of Λ, and let $k \leq d$ be a positive integer. Then, for any k-dimensional sublattice $\Lambda' \subset \Lambda$, it holds that*

$$\mathrm{vol}\,(\Lambda') \geq \min_J \prod_{j \in J} \|\mathbf{b}_j^\star\|,$$

https://doi.org/10.1017/9781108854207.004 Published online by Cambridge University Press

where J ranges over all subsets of $\{0, \ldots, d-1\}$ *of size k.*

We will now apply this to the dense sublattice Λ' generated by the n short vectors out of the $d = 2n$ dimensions of the NTRU lattice. This gives a (log) left-hand side of $\log \mathrm{vol}(\Lambda') \leq n \log R$, where $R = \|(\mathbf{f}, \mathbf{g})\| \approx \sqrt{d}\sigma$ (and in fact we can argue that $\log \mathrm{vol}(\Lambda') \approx n \log R$). For the right-hand side, the minimum is reached by the n last indices $J = \{n, n+1, \ldots, 2n-1\}$.

Pictorially, the usual one-dimensional argument forbids the last Gram–Schmidt vector to go above R; if the heuristically predicted shape contradicts this rule, then the shortest vector must have been detected somehow. The multi-dimensional version of Pataki and Tural instead forbids the black-hashed region to have a surface larger than the grey-filled region in Figure 2.9.

We make our prediction under the Z-shape model, denoting $s = \lg \alpha_\beta$ the slope of the middle section, between indices $n - z$ and $n + z$. The inaccuracies of this model discussed in Section 2.4.3 should be asymptotically negligible, as we will be interested in regimes for which $\beta = o(z)$. The picture also makes it easy to compute the right-hand side of the inequality. It is given by the surface of a right-angled triangle of height $h = \frac{1}{2} \lg q$. Its surface is given by $S = \frac{1}{2} hz = \frac{1}{2} h^2/s = (\lg q)^2/(8 \lg \alpha_\beta)$. We therefore predict that the Pataki-Tural inequality would be violated when $nR = S$ that is: $\lg \alpha_\beta = \lg^2 q/(8nR)$. Noting that $\lg \alpha_\beta = \Theta\left(\frac{\lg \beta}{\beta}\right)$, we conclude that the lattice reduction is going to detect the dense sublattice when

$$\frac{\beta}{\lg \beta} = \Theta\left(\frac{nR}{\lg^2 q}\right).$$

The required block size is therefore $\beta = \tilde{\Theta}(n/\lg^2 q)$ as it was for the subfield attack, however a more careful analysis of the hidden constants [324] reveals that going to the subfield is slightly unfavourable.

Concrete Behaviour

Although we kept the above development asymptotic for simplicity, it is not hard to keep track of the hidden constants – or even to run simulations – and to predict precisely when the Pataki–Tural lemma would be violated. However, even such a methodology would only lead to an upper bound on the cost of this attack and not an estimate. Indeed, this methodology would essentially correspond to the one of Section 2.5.2 for LWE-uSVP; it is based on an impossibility argument, but it does not explain or predict the phenomenon, unlike the 2016 estimate.

We therefore emphasise this gap as our last and foremost open problem: give a more detailed explanation of how BKZ detects the hidden sublattice,

https://doi.org/10.1017/9781108854207.004 Published online by Cambridge University Press

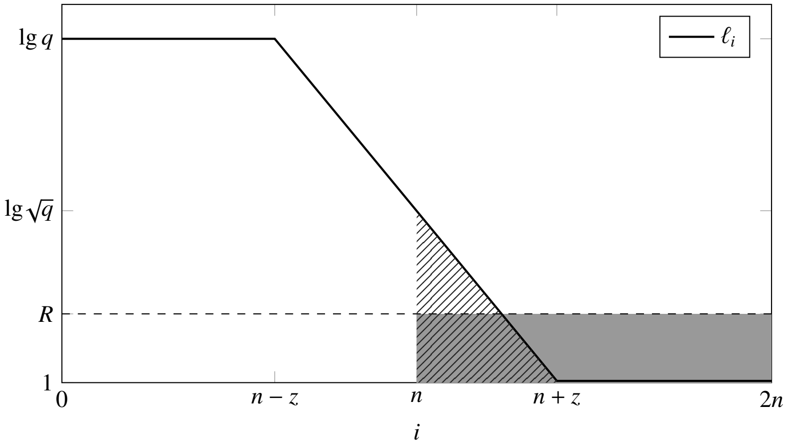

Figure 2.9 The Pataki–Tural constraint on reduced NTRU bases.

leading to a heuristic estimate on when the phenomenon happens, confirmed by extensive experiments. A possible answer may be found by extending the probabilistic analysis of [148], this time accounting for more than the n shortest vectors $(x^i \cdot f, x^i \cdot g)$ for $0 \leq i < n$. Indeed, one could instead consider all the vectors $(p \cdot f, p \cdot g)$ for elements p up to a certain length. These vectors are longer and therefore the probability of finding a given one of them is smaller. Yet, it might be that, in some regimes of parameters, their number outgrows this decrease in probability. When considering multiple vectors from the same dense sublattice the events of finding each of them may not be independent, which might require some care when modelling.

Acknowledgments We are indebted to Damien Stehlé, Rachel Player and Benjamin Wesolowski for their careful review of this chapter. Part of this work was done while the authors were visiting the Simons Institute for the Theory of Computing. The research of Albrecht was supported by EPSRC grants EP/S020330/1 and EP/S02087X/1. The research of Albrecht and Ducas was supported by the European Union Horizon 2020 Research and Innovation Program Grant 780701 (Project PROMETHEUS).

https://doi.org/10.1017/9781108854207.004 Published online by Cambridge University Press