Revisiting the Expected Cost of Solving uSVP and Applications to LWE

Martin R. Albrecht[1], Florian Göpfert[2,3], Fernando Virdia[1(✉)], and Thomas Wunderer[3(✉)]

[1] Information Security Group, Royal Holloway, University of London, Egham, UK
martin.albrecht@royalholloway.ac.uk, fernando.virdia.2016@rhul.ac.uk
[2] rockenstein AG, Würzburg, Germany
fgoepfert@cdc.informatik.tu-darmstadt.de
[3] TU Darmstadt, Darmstadt, Germany
twunderer@cdc.informatik.tu-darmstadt.de

Abstract. Reducing the Learning with Errors problem (LWE) to the Unique-SVP problem and then applying lattice reduction is a commonly relied-upon strategy for estimating the cost of solving LWE-based constructions. In the literature, two different conditions are formulated under which this strategy is successful. One, widely used, going back to Gama & Nguyen's work on predicting lattice reduction (Eurocrypt 2008) and the other recently outlined by Alkim et al. (USENIX 2016). Since these two estimates predict significantly different costs for solving LWE parameter sets from the literature, we revisit the Unique-SVP strategy. We present empirical evidence from lattice-reduction experiments exhibiting a behaviour in line with the latter estimate. However, we also observe that in some situations lattice-reduction behaves somewhat better than expected from Alkim et al.'s work and explain this behaviour under standard assumptions. Finally, we show that the security estimates of some LWE-based constructions from the literature need to be revised and give refined expected solving costs.

Keywords: Cryptanalysis · Lattice-based cryptography · Learning with Errors · Lattice reduction

1 Introduction

The *Learning with Errors* problem (LWE) has attained a central role in cryptography as a key hard problem for building cryptographic constructions,

The research of Albrecht was supported by EPSRC grant "Bit Security of Learning with Errors for Post-Quantum Cryptography and Fully Homomorphic Encryption" (EP/P009417/1) and EPSRC grant "Multilinear Maps in Cryptography" (EP/L018543/1). The research of Göpfert and Wunderer was supported by the DFG as part of project P1 within the CRC 1119 CROSSING. The research of Virdia was supported by the EPSRC and the UK government as part of the Centre for Doctoral Training in Cyber Security at Royal Holloway, University of London (EP/K035584/1).

T. Takagi and T. Peyrin (Eds.): ASIACRYPT 2017, Part I, LNCS 10624, pp. 297–322, 2017.
https://doi.org/10.1007/978-3-319-70694-8_11

e.g. quantum-safe public-key encryption/key exchange and signatures schemes [Reg09,LP11,ADPS16,BG14a], fully homomorphic encryption [BV11, GSW13] and obfuscation of some families of circuits [BVWW16].

Informally, LWE asks to recover a secret vector $\mathbf{s} \in \mathbb{Z}_q^n$, given a matrix $\mathbf{A} \in \mathbb{Z}_q^{m \times n}$ and a vector $\mathbf{c} \in \mathbb{Z}_q^m$ such that $\mathbf{A}\mathbf{s} + \mathbf{e} = \mathbf{c} \mod q$ for a short error vector $\mathbf{e} \in \mathbb{Z}_q^m$ sampled coordinate-wise from an error distribution χ. The decision variant of LWE asks to distinguish between an LWE instance $(\mathbf{A}, \mathbf{c})$ and uniformly random $(\mathbf{A}, \mathbf{c}) \in \mathbb{Z}_q^{m \times n} \times \mathbb{Z}_q^m$. To assess the security provided by a given set of parameters n, χ, q, two strategies are typically considered: the *dual* strategy finds short vectors in the lattice

$$q\Lambda^* = \left\{\mathbf{x} \in \mathbb{Z}_q^m \mid \mathbf{x} \cdot \mathbf{A} \equiv 0 \bmod q\right\},$$

i.e. it solves the *Short Integer Solutions* problem (SIS). Given such a short vector $\mathbf{v}$, we can decide if an instance is LWE by computing $\langle \mathbf{v}, \mathbf{c} \rangle = \langle \mathbf{v}, \mathbf{e} \rangle$ which is short whenever $\mathbf{v}$ and $\mathbf{e}$ are sufficiently short [MR09]. This strategy was recently revisited for small, sparse secret instances of LWE [Alb17]. The *primal* strategy finds the closest vector to $\mathbf{c}$ in the integral span of columns of $\mathbf{A}$ mod q [LP11], i.e. it solves the corresponding *Bounded Distance Decoding* problem (BDD) directly. Writing $[\mathbf{I}_n | \mathbf{A}']$ for the reduced row echelon form of $\mathbf{A}^T \in \mathbb{Z}_q^{n \times m}$ (with high probability and after appropriate permutation of columns), this task can be reformulated as solving the *unique Shortest Vector Problem* (uSVP) in the $m + 1$ dimensional q-ary lattice

$$\Lambda = \mathbb{Z}^{m+1} \cdot \begin{pmatrix} \mathbf{I}_n & \mathbf{A}' & 0 \\ \mathbf{0} & q\,\mathbf{I}_{m-n} & 0 \\ \mathbf{c}^T & & t \end{pmatrix} \tag{1}$$

by Kannan's embedding [Kan87] with embedding factor t.[1] Indeed, BDD and uSVP are polynomial-time equivalent for small approximation factors up to $\sqrt{n/\log n}$ [LM09]. The lattice Λ has volume $t \cdot q^{m-n}$ and contains a vector of norm $\sqrt{\|\mathbf{e}\|^2 + t^2}$ which is unusually short, i.e. the gap between the first and second Minkowski minimum $\lambda_2(\Lambda)/\lambda_1(\Lambda)$ is large.

Alternatively, if the secret vector $\mathbf{s}$ is also short, there is a second established embedding reducing LWE to uSVP (cf. Eq. (4)). When the LWE instance under consideration is in *normal form*, i.e. the secret $\mathbf{s}$ follows the noise distribution, the geometries of the lattices in (1) and (4) are the same, which is why without loss of generality we only consider (1) in this work save for Sect. 5.

To find short vectors, lattice reduction [LLL82,Sch87,GN08a,HPS11,CN11, MW16] can be applied. Thus, to establish the cost of solving an LWE instance, we may consider the cost of lattice reduction for solving uSVP.

Two conflicting estimates for the success of lattice reduction in solving uSVP are available in the literature. The first is going back to [GN08b] and was developed in [AFG14,APS15,Gö16,HKM17] for LWE. This estimate is commonly

[1] Alternatively, we can perform lattice reduction on the q-ary lattice spanned by $\mathbf{A}^T$, i.e. the lattice spanned by the first m rows of (1), followed by an enumeration to find the closest (projected) lattice point to (the projection of) $\mathbf{c}$ [LP11,LN13].

relied upon by designers in the literature, e.g. [BG14a, CHK+17, CKLS16a, CLP17, ABB+17]. The second estimate was recently outlined in [ADPS16] and is relied upon in [BCD+16, BDK+17]. We will use the shorthand *2008 estimate* for the former and *2016 estimate* for the latter. As illustrated in Fig. 1, the predicted costs under these two estimates differ greatly. For example, considering $n = 1024$, $q \approx 2^{15}$ and χ a discrete Gaussian with standard deviation $\sigma = 3.2$, the former predicts a cost of $\approx 2^{355}$ operations, whereas the latter predicts a cost of $\approx 2^{287}$ operations in the same cost model for lattice reduction.[2]

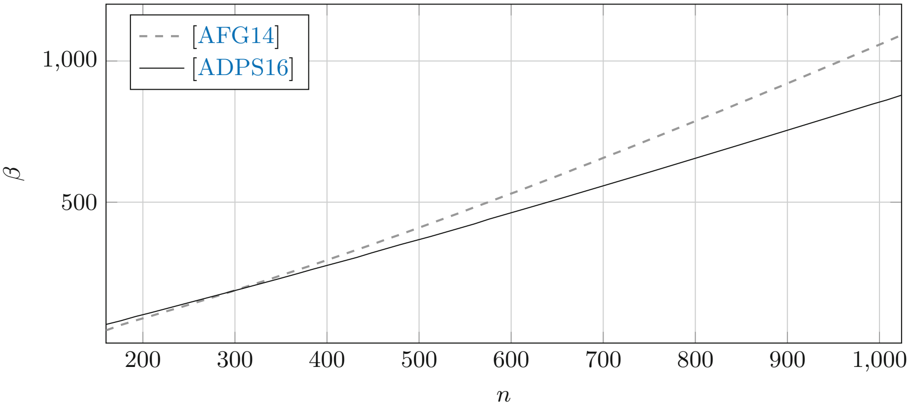

Fig. 1. Required block size β according to the estimates given in [AFG14, ADPS16] for modulus $q = 2^{15}$, standard deviation $\sigma = 3.2$ and increasing n; for [AFG14] we set $\tau = 0.3$ and $t = 1$. Lattice-reduction runs in time $2^{\Omega(\beta)}$.

Our Contribution. Relying on recent progress in publicly available lattice-reduction libraries [FPL17, FPY17], we revisit the embedding approach for solving LWE resp. BDD under some reasonable assumptions about the LWE error distribution. After some preliminaries in Sect. 2, we recall the two competing estimates from the literature in Sect. 3. Then, in Sect. 4, we expand on the exposition from [ADPS16] followed by presenting the results of running 23,000 core hours worth of lattice-reduction experiments in medium to larger block sizes β. Our results confirm that lattice-reduction largely follows the behaviour expected from the 2016 estimate [ADPS16]. However, we also find that in our experiments the attack behaves somewhat better than expected.[3] In Sect. 4.3, we then explain the observed behaviour of the BKZ algorithm under the *Geometric Series Assumption* (GSA, see below) and under the assumption that the unique

[2] Assuming that an SVP oracle call in dimension β costs $2^{0.292\,\beta+16.4}$ [BDGL16, APS15], where $+16.4$ takes the place of $o(\beta)$ from the asymptotic formula and is based on experiments in [Laa14].

[3] We note that this deviation from the expectation has a negligible impact on security estimates for cryptographic parameters.

shortest vector is distributed in a random direction relative to the rest of the basis. Finally, using the 2016 estimate, we show that some proposed parameters from the literature need to be updated to maintain the currently claimed level of security in Sect. 5. In particular, we give reduced costs for solving the LWE instances underlying TESLA [ABB+17] and the somewhat homomorphic encryption scheme in [BCIV17]. We also show that under the revised, corrected estimate, the primal attack performs about as well on SEAL v2.1 parameter sets as the dual attack from [Alb17].

2 Preliminaries

We write vectors in lower-case bold, e.g. $\mathbf{a}$, and matrices in upper-case bold, e.g. $\mathbf{A}$. We write $\langle\cdot,\cdot\rangle$ for the inner products and $\cdot$ for matrix-vector products. By abuse of notation we consider vectors to be row resp. column vectors depending on context, such that $\mathbf{v}\cdot\mathbf{A}$ and $\mathbf{A}\cdot\mathbf{v}$ are meaningful. We write $\mathbf{I}_m$ for the $m \times m$ identity matrix over whichever base ring is implied from context. We write $\mathbf{0}_{m\times n}$ for the $m \times n$ all zero matrix. If the dimensions are clear from the context, we may omit the subscripts.

2.1 Learning with Errors

The Learning with Errors (LWE) problem is defined as follows.

Definition 1 (LWE [Reg09]). *Let n, q be positive integers, χ be a probability distribution on $\mathbb{Z}$ and $\mathbf{s}$ be a secret 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 $(\mathbf{a}, c) = (\mathbf{a}, \langle\mathbf{a},\mathbf{s}\rangle + e) \in \mathbb{Z}_q^n \times \mathbb{Z}_q$ sampled according to $L_{\mathbf{s},\chi}$.*

We may write LWE instances in matrix form $(\mathbf{A}, \mathbf{c})$, where rows correspond to samples $(\mathbf{a}_i, c_i)$. In many instantiations, χ is a discrete Gaussian distribution with standard deviation σ. Throughout, we denote the number of LWE samples considered as m. Writing $\mathbf{e}$ for the vector of error terms, we expect $\|\mathbf{e}\| \approx \sqrt{m}\sigma$.

2.2 Lattices

A lattice is a discrete subgroup of $\mathbb{R}^d$. Throughout, d denotes the dimension of the lattice under consideration and we only consider full rank lattices, i.e., lattices $\Lambda \subset \mathbb{R}^d$ such that $\mathsf{span}_{\mathbb{R}}(\Lambda) = \mathbb{R}^d$. A lattice $\Lambda \subset \mathbb{R}^d$ can be represented by a basis $\mathbf{B} = \{\mathbf{b}_1, \ldots, \mathbf{b}_k\}$, i.e., $\mathbf{B}$ is linearly independent and $\Lambda = \mathbb{Z}\mathbf{b}_1 + \cdots + \mathbb{Z}\mathbf{b}_k$. We write $\mathbf{b}_i$ for basis vectors and $\mathbf{b}_i^*$ for the corresponding Gram-Schmidt vectors.

We write $\Lambda(\mathbf{B})$ for the lattice generated by the rows of the matrix $\mathbf{B}$, i.e. all integer-linear combinations of the rows of $\mathbf{B}$. The volume of a lattice $\mathrm{Vol}(\Lambda)$ is the absolute value of the determinant of any basis and it holds that $\mathrm{Vol}(\Lambda) = \prod_{i=1}^{d} \|\mathbf{b}_i^*\|$. We write $\lambda_i(\Lambda)$ for *Minkowski's successive minima*, i.e. the radius of the smallest ball centred around zero containing i linearly independent lattice vectors. The *Gaussian Heuristic* predicts

$$\lambda_1(\Lambda) \approx \sqrt{\frac{d}{2\pi e}} \mathrm{Vol}(\Lambda)^{1/d}.$$

For a lattice basis $\mathbf{B} = \{\mathbf{b}_1, \ldots, \mathbf{b}_d\}$ and for $i \in \{1, \ldots, d\}$ let $\pi_{\mathbf{B},i}(\mathbf{v})$ denote the orthogonal projection of $\mathbf{v}$ onto $\{\mathbf{b}_1, \ldots, \mathbf{b}_{i-1}\}$, where $\pi_{\mathbf{B},1}$ is the identity. We extend the notation to sets of vectors in the natural way. Since usually the basis $\mathbf{B}$ is clear from the context, we omit it in the notation and simply write π_i instead of $\pi_{\mathbf{B},i}$. Since Sect. 4.3 relies heavily on size reduction, we recall its definition and reproduce the algorithm in Algorithm 1.

Definition 2. *Let* $\mathbf{B}$ *be a basis,* $\mathbf{b}_i^*$ *its Gram-Schmidt vectors and*

$$\mu_{i,j} = \langle \mathbf{b}_i, \mathbf{b}_j^* \rangle / \langle \mathbf{b}_j^*, \mathbf{b}_j^* \rangle,$$

then $\mathbf{B}$ *basis is* size reduced *if* $|\mu_{i,j}| \leq 1/2$ *for* $1 \leq j \leq i \leq n$.

```
Data: lattice basis $\mathbf{B}$
Data: top index $i$
Data: start index $1 \leq s < i$
for $j$ from $i-1$ to $s$ do
    $\mu_{ij} \leftarrow \langle \mathbf{b}_i, \mathbf{b}_j^* \rangle / \langle \mathbf{b}_j^*, \mathbf{b}_j^* \rangle$;
    $\mathbf{b}_i \leftarrow \mathbf{b}_i - \lfloor \mu_{ij} \rceil \mathbf{b}_j$;
end
```

Algorithm 1: Size reduction

2.3 Lattice Reduction

Informally, lattice reduction is the process of improving the quality of a lattice basis. To express the output quality of a lattice reduction, we may relate the shortest vector in the output basis to the volume of the lattice in the *Hermite-factor regime* or to the shortest vector in the lattice, in the *approximation-factor regime*. Note that any algorithm finding a vector with approximation-factor α can be used to solve Unique-SVP with a gap $\lambda_2(\Lambda)/\lambda_1(\Lambda) < \alpha$.

The best known theoretical bound for lattice reduction is attained by Slide reduction [GN08a]. In this work, however, we consider the BKZ algorithm (more precisely: BKZ 2.0 [Che13], cf. Sect. 4.2) which performs better in practice. The BKZ-β algorithm repeatedly calls an SVP oracle for finding (approximate) shortest vectors in dimension or *block size* β. It has been shown that after polynomially many calls to the SVP oracle, the basis does not change much more [HPS11]. After BKZ-β reduction, we call the basis *BKZ-β reduced* and in the Hermite-factor regime assume [Che13] that this basis contains a vector of length $\|\mathbf{b}_1\| = \delta_0^d \cdot \mathrm{Vol}(L)^{1/d}$ where

$$\delta_0 = (((\pi\beta)^{1/\beta}\beta)/(2\pi e))^{1/(2(\beta-1))}.$$

Furthermore, we generally assume that for a BKZ-β reduced basis of $\Lambda(\mathbf{B})$ the Geometric Series Assumption holds.

Definition 3 (Geometric Series Assumption [Sch03]**).** *The norms of the Gram-Schmidt vectors after lattice reduction satisfy*

$$\|\mathbf{b}_i^*\| = \alpha^{i-1} \cdot \|\mathbf{b}_1\| \text{ for some } 0 < \alpha < 1.$$

Combining the GSA with the root-Hermite factor $\|\mathbf{b}_1\| = \delta_0^d \cdot \mathrm{Vol}(\Lambda)^{1/d}$ and $\mathrm{Vol}(\Lambda) = \prod_{i=1}^{d} \|\mathbf{b}_i^*\|$, we get $\alpha = \delta_0^{-2d/(d-1)} \approx \delta_0^{-2}$ for the GSA.

3 Estimates

As highlighted above, two competing estimates exist in the literature for when block-wise lattice reduction will succeed in solving uSVP instances such as (1).

3.1 2008 Estimate

A first systematic experimental investigation into the behavior of lattice reduction algorithms LLL, DEEP and BKZ was provided in [GN08b]. In particular, [GN08b] investigates the behavior of these algorithms for solving Hermite-SVP, Approx-SVP and Unique-SVP for families of lattices used in cryptography.

For Unique-SVP, the authors performed experiments in small block sizes on two classes of semi-orthogonal lattices and on Lagarias-Odlyzko lattices [LO83], 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, [GN08b] observed that LLL and BKZ seem to recover a unique shortest vector with high probability whenever $\lambda_2(\Lambda)/\lambda_1(\Lambda) \geq \tau\delta_0^d$, where $\tau < 1$ is an empirically determined constant that depends on the lattice family and algorithm used.

In [AFG14] an experimental analysis of solving LWE based on the same estimate was carried out for lattices of the form (1). As mentioned above, this lattice contains an unusually short vector $\mathbf{v} = (\mathbf{e} \mid t)$ of squared norm $\lambda_1(\Lambda)^2 = \|\mathbf{v}\|^2 = \|\mathbf{e}\|^2 + t^2$. Thus, when $t = \|\mathbf{e}\|$ resp. $t = 1$ this implies $\lambda_1(\Lambda) \approx \sqrt{2m}\sigma$ resp. $\lambda_1(\Lambda) \approx \sqrt{m}\sigma$, with σ the standard deviation of $\mathbf{e}_i \leftarrow_{\$} \chi$. The second minimum $\lambda_2(\Lambda)$ is assumed to correspond to the Gaussian Heuristic for the lattice. Experiments in [AFG14] using LLL and BKZ (with block sizes 5 and 10) confirmed the 2008 estimate, providing constant values for τ for lattices of the form (1), depending on the chosen algorithm, for a 10% success rate. Overall, τ was found to lie between 0.3 and 0.4 when using BKZ.

Still focusing on LWE, in [APS15] a closed formula for δ_0 is given in function of n, σ, q, τ, which implicitly assumes $t = \|\mathbf{e}\|$. In [Gö16] a bound for δ_0 in the [GN08b] model for the case of $t = 1$, which is always used in practice, is given. In [HKM17], a related closed formula is given, directly expressing the asymptotic running time for solving LWE using this approach.

3.2 2016 Estimate

In [ADPS16] an alternative estimate is outlined. The estimate predicts that $\mathbf{e}$ can be found if[4]

$$\sqrt{\beta/d}\,\|(\mathbf{e} \mid 1)\| \approx \sqrt{\beta}\sigma \leq \delta_0^{2\beta-d}\,\mathrm{Vol}(\Lambda(\mathbf{B}))^{1/d}, \tag{2}$$

under the assumption that the Geometric Series Assumption holds (until a projection of the unusually short vector is found). The brief justification for this estimate given in [ADPS16] notes that this condition ensures that the projection of $\mathbf{e}$ orthogonally to the first $d-\beta$ (Gram-Schmidt) vectors is shorter than the expectation for $\mathbf{b}^*_{d-\beta+1}$ under the GSA and thus would be found by the SVP oracle when called on the last block of size β. Hence, for any β satisfying (2), the actual behaviour would deviate from that predicted by the GSA. Finally, the argument can be completed by appealing to the intuition that a deviation from expected behaviour on random instances—such as the GSA—leads to a revelation of the underlying structural, secret information.[5]

4 Solving uSVP

Given the significant differences in expected solving time under the two estimates, cf. Fig. 1, and recent progress in publicly available lattice-reduction libraries enabling experiments in larger block sizes [FPL17,FPY17], we conduct a more detailed examination of BKZ's behaviour on uSVP instances. For this, we first explicate the outline from [ADPS16] to establish the expected behaviour, which we then experimentally investigate in Sect. 4.2. Overall, our experiments confirm the expectation. However, the algorithm behaves somewhat better than expected, which we then explain in Sect. 4.3.

For the rest of this section, let $\mathbf{v}$ be a unique shortest vector in some lattice $\Lambda \subset \mathbb{R}^d$, i.e. in case of (1) we have $\mathbf{v} = (\mathbf{e} \mid t)$ where we pick $t = 1$.

4.1 Prediction

Projected norm. In what follows, we assume the unique shortest vector $\mathbf{v}$ is drawn from a spherical distribution or is at least "not too skewed" with respect

[4] [ADPS16] has $2\beta - d - 1$ in the exponent, which seems to be an error.

[5] We note that observing such a deviation implies solving Decision-LWE.

to the current basis. As a consequence, following [ADPS16], we assume that all orthogonal projections of $\mathbf{v}$ onto a k-dimensional subspace of $\mathbb{R}^d$ have expected norm $(\sqrt{k}/\sqrt{d})\,\|\mathbf{v}\|$. Note that this assumption can be dropped by adapting (2) to $\|\mathbf{v}\| \leq \delta_0^{2\beta-d}\,\mathrm{Vol}(\Lambda)^{\frac{1}{d}}$ since $\|\pi_{d-\beta+1}(\mathbf{v})\| \leq \|\mathbf{v}\|$.

Finding a projection of the short vector. Assume that β is chosen minimally such that (2) holds. When running BKZ the length of the Gram-Schmidt basis vectors of the current basis converge to the lengths predicted by the GSA. Therefore, at some point BKZ will find a basis $\mathbf{B} = \{\mathbf{b}_1, \ldots, \mathbf{b}_d\}$ of Λ for which we can assume that the GSA holds with root Hermite factor δ_0. Now, consider the stage of BKZ where the SVP oracle is called on the last full projected block of size β with respect to $\mathbf{B}$. Note that the projection $\pi_{d-\beta+1}(\mathbf{v})$ of the shortest vector is contained in the lattice

$$\Lambda_{d-\beta+1} := \Lambda\left(\pi_{d-\beta+1}(\mathbf{b}_{d-\beta+1}), \ldots, \pi_{d-\beta+1}(\mathbf{b}_d)\right),$$

since

$$\pi_{d-\beta+1}(\mathbf{v}) = \sum_{i=d-\beta+1}^{d} \nu_i \pi_{d-\beta+1}(\mathbf{b}_i) \in \Lambda_{d-\beta+1}, \text{ where } \nu_i \in \mathbb{Z} \text{ with } \mathbf{v} = \sum_{i=1}^{d} \nu_i \mathbf{b}_i.$$

By (2), the projection $\pi_{d-\beta+1}(\mathbf{v})$ is in fact expected to be the shortest non-zero vector in $\Lambda_{d-\beta+1}$, since it is shorter than the GSA's estimate for $\lambda_1(\Lambda_{d-\beta+1})$, i.e.

$$\|\pi_{d-\beta+1}(\mathbf{v})\| \approx \frac{\sqrt{\beta}}{\sqrt{d}}\,\|\mathbf{v}\| \leq \delta_0^{-2(d-\beta)+d}\mathrm{Vol}(\Lambda)^{\frac{1}{d}}.$$

Hence the SVP oracle will find $\pm\pi_{d-\beta+1}(\mathbf{v})$ and BKZ inserts

$$\mathbf{b}^{\mathsf{new}}_{d-\beta+1} = \pm \sum_{i=d-\beta+1}^{d} \nu_i \mathbf{b}_i$$

into the basis $\mathbf{B}$ at position $d - \beta + 1$, as already outlined in [ADPS16]. In other words, by finding $\pm\pi_{d-\beta+1}(\mathbf{v})$, BKZ recovers the last β coefficients $\nu_{d-\beta+1}, \ldots, \nu_d$ of $\mathbf{v}$ with respect to the basis $\mathbf{B}$.

Finding the short vector. The above argument can be extended to an argument for the full recovery of $\mathbf{v}$. Consider the case that in some tour of BKZ-β, a projection of $\mathbf{v}$ was found at index $d - \beta + 1$. Then in the following tour, by arguments analogous to the ones above, a projection of $\mathbf{v}$ will likely be found at index $d - 2\beta + 2$, since now it holds that

$$\pi_{d-2\beta+2}(\mathbf{v}) \in \Lambda_{d-2\beta+2} := \Lambda\left(\pi_{d-2\beta+2}(\mathbf{b}_{d-2\beta+2}), \ldots, \pi_{d-2\beta+2}(\mathbf{b}^{\mathsf{new}}_{d-\beta+1})\right).$$

Repeating this argument for smaller indices shows that after a few tours $\mathbf{v}$ will be recovered. Furthermore, noting that BKZ calls LLL which in turn calls size reduction, i.e. Babai's nearest plane [Bab86], at some index $i > 1$ size reduction will recover $\mathbf{v}$ from $\pi_i(\mathbf{v})$. In particular, it is well-known that size reduction (Algorithm 1) will succeed in recovering $\mathbf{v}$ whenever

$$\mathbf{v} \in \mathbf{b}^{\mathsf{new}}_{d-\beta+1} + \left\{ \sum_{i=1}^{d-\beta} c_i \cdot \mathbf{b}_i^* : c_i \in \left[-\frac{1}{2}, \frac{1}{2}\right] \right\}. \tag{3}$$

4.2 Observation

The above discussion naturally suggests a strategy to verify the expected behaviour. We have to verify that the projected norms $\|\pi_i(\mathbf{v})\| = \|\pi_i(\mathbf{e} \mid 1)\|$ do indeed behave as expected and that $\pi_{d-\beta+1}(\mathbf{v})$ is recovered by BKZ-β for the minimal $\beta \in \mathbb{N}$ satisfying (2). Finally, we have to measure when and how $\mathbf{v} = (\mathbf{e} \mid 1)$ is eventually recovered.

Thus, we ran lattice-reduction on many lattices constructed from LWE instances using Kannan's embedding. In particular, we picked the entries of $\mathbf{s}$ and $\mathbf{A}$ uniformly at random from $\mathbb{Z}_q$, the entries of $\mathbf{e}$ from a discrete Gaussian distribution with standard deviation $\sigma = 8/\sqrt{2\pi}$, and we constructed our basis as in (1) with embedding factor $t = 1$. For parameters (n, q, σ), we then estimated the minimal pair (in lexicographical order) (β, m) to satisfy (2).

Implementation. To perform our experiments, we used SageMath 7.5.1 [S+17] in combination with the `fplll` 5.1.0 [FPL17] and `fpylll` 0.2.4dev [FPY17] libraries. All experiments were run on a machine with Intel(R) Xeon(R) CPU E5-2667 v2 @ 3.30GHz cores ("strombenzin") resp. Intel(R) Xeon(R) CPU E5-2690 v4 @ 2.60GHz ("atomkohle"). Each instance was reduced on a single core, with no parallelisation.

Our BKZ implementation inherits from the implementation in `fplll` and `fpylll` of BKZ 2.0 [Che13] algorithm. As in BKZ 2.0, we restricted the enumeration radius to be approximately the size of the Gaussian Heuristic for the projected sublattice, apply recursive BKZ-β' preprocessing with a block size $\beta' < \beta$, make use of extreme pruning [GNR10] and terminate the algorithm when it stops making significant progress. We give simplified pseudocode of our implementation in Algorithm 2. We ran BKZ for at most 20 tours using `fplll`'s default pruning and preprocessing strategies and, using `fplll`'s default auto abort strategy, terminated the algorithm whenever the slope of the Gram Schmidt vectors did not improve for five consecutive tours. Additionally, we aborted if a vector of length $\approx \|\mathbf{v}\|$ was found in the basis (in line 15 of Algorithm 2).

Implementations of block-wise lattice reduction algorithms such as BKZ make heavy use of LLL [LLL82] and size reduction. This is to remove linear dependencies introduced during the algorithm, to avoid numerical stability issues and to improve the performance of the algorithm by moving short vectors to the front earlier. The main modification in our implementation is that calls to LLL during preprocessing and postprocessing are restricted to the current block, not

```
   Data: LLL-reduced lattice basis $\mathbf{B}$
   Data: block size $\beta$, preprocessing block size $\beta'$
1  repeat                                                          // tour
2      for $\kappa \leftarrow 1$ to $d$ do                          // step$_\kappa$
3          size reduction from index 1 to $\kappa$ (inclusive);
4          $\ell \leftarrow \|b_\kappa^*\|$;
           // extreme pruning + recursive preprocessing
5          repeat until termination condition met
6              rerandomise $\pi_\kappa(\mathbf{b}_{\kappa+1}, \ldots, \mathbf{b}_{\kappa+\beta-1})$;
7              LLL on $\pi_\kappa(\mathbf{b}_\kappa, \ldots, \mathbf{b}_{\kappa+\beta-1})$;
8              BKZ-$\beta'$ on $\pi_\kappa(\mathbf{b}_\kappa, \ldots, \mathbf{b}_{\kappa+\beta-1})$;
9              $\mathbf{v} \leftarrow$ SVP on $\pi_\kappa(\mathbf{b}_\kappa, \ldots, \mathbf{b}_{\kappa+\beta-1})$;
10             if $\mathbf{v} \neq \perp$ then
11                 extend $\mathbf{B}$ by inserting $\mathbf{v}$ into $\mathbf{B}$ at index $\kappa + \beta$;
12                 LLL on $\pi_\kappa(\mathbf{b}_\kappa, \ldots, \mathbf{b}_{\kappa+\beta})$ to remove linear dependencies;
13                 drop row with all zero entries;
14             end
15         size reduction from index 1 to $\kappa$ (inclusive);
16         if $\ell = \|b_\kappa^*\|$ then
17             yield $\top$;
18         else
19             yield $\perp$;
20         end
21     end
22     if $\top$ for all $\kappa$ then
23         return;
24     end
```

Algorithm 2: Simplified BKZ 2.0 Algorithm

touching any other vector, to aid analysis. That is, in Algorithm 2, LLL is called in lines 7 and 12 and we modified these LLL calls not to touch any row with index smaller than κ, not even to perform size reduction.

As a consequence, we only make use of vectors with index smaller than κ in lines 3 and 15. Following the implementations in [FPL17,FPY17], we call size reduction from index 1 to κ before (line 3) and after (line 15) the innermost loop with calls to the SVP oracle. These calls do not appear in the original description of BKZ. However, since the innermost loop re-randomises the basis when using extreme pruning, the success condition of the original BKZ algorithm needs to be altered. That is, the algorithm cannot break the outer loop once it makes no more changes as originally specified. Instead, the algorithm terminates if it does not find a shorter vector at any index κ. Now, the calls to size reduction ensure that the comparison at the beginning and end of each step κ is meaningful even when the Gram-Schmidt vectors are only updated lazily in the underlying implementation. That is, the call to size reduction triggers an internal update of the underlying Gram-Schmidt vectors and are hence implementation artefacts. The reader may think of these size reduction calls as explicating calls otherwise hidden behind calls to LLL and we stress that our analysis applies to BKZ as commonly implemented, our changes merely enable us to more easily predict and experimentally verify the behaviour.

We note that the break condition for the innermost loop at line 5 depends on the pruning parameters chosen, which control the success probability of enumeration. Since it does not play a material role in our analysis, we simply state that some condition will lead to a termination of the innermost loop.

Finally, we recorded the following information. At the end of each step κ during lattice reduction, we recorded the minimal index i such that $\pi_i(\mathbf{v})$ is in $\text{span}(\mathbf{b}_1, \dots, \mathbf{b}_i)$ and whether $\pm\mathbf{v}$ itself is in the basis. In particular, to find the index i in the basis $\mathbf{B}$ of $\pi_i(\mathbf{v})$ given $\mathbf{v}$, we compute the coefficients of $\mathbf{v}$ in basis $\mathbf{B}$ (at the current step) and pick the first index i such that all coefficients with larger indices are zero. Then, we have $\pi_i(\mathbf{b}_i) = c \cdot \pi_i(\mathbf{v})$ for some $c \in \mathbb{R}$. From the algorithm, we expect to have found $\pm\pi_i(\mathbf{b}_i) = \pi_i(\mathbf{v})$ and call i the index of the projection of $\mathbf{v}$.

Results. In Fig. 2, we plot the average norms of $\pi_i(\mathbf{v})$ against the expectation $\sqrt{d-i+1}\,\sigma \approx \sqrt{\frac{d-i+1}{d}}\sqrt{m \cdot \sigma^2 + 1}$, indicating that $\sqrt{d-i+1}\,\sigma$ is a close approximation of the expected lengths except perhaps for the last few indices.

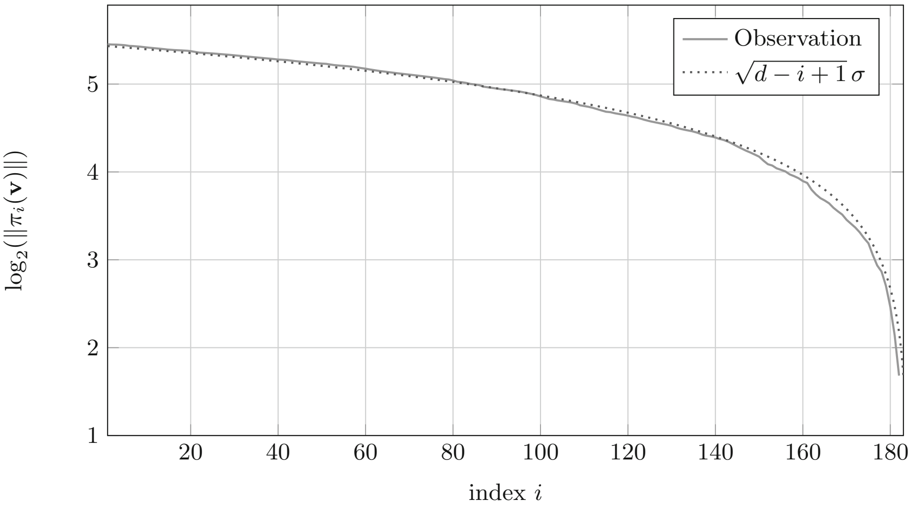

Fig. 2. Expected and average observed norms $\|\pi_i(\mathbf{v})\|$ for 16 bases (LLL-reduced) and vectors $\mathbf{v}$ of dimension $d = m+1$ and volume q^{m-n} with LWE parameters $n = 65, m = 182, q = 521$ and standard deviation $\sigma = 8/\sqrt{2\pi}$.

Recall that, as illustrated in Fig. 3, we expect to find the projection $\pi_{d-\beta+1}(\mathbf{v})$ when (β, d) satisfy (2), eventually leading to a recovery of $\mathbf{v}$, say, by an extension of the argument for the recovery of $\pi_{d-\beta+1}(\mathbf{v})$. Our experiments, summarised in

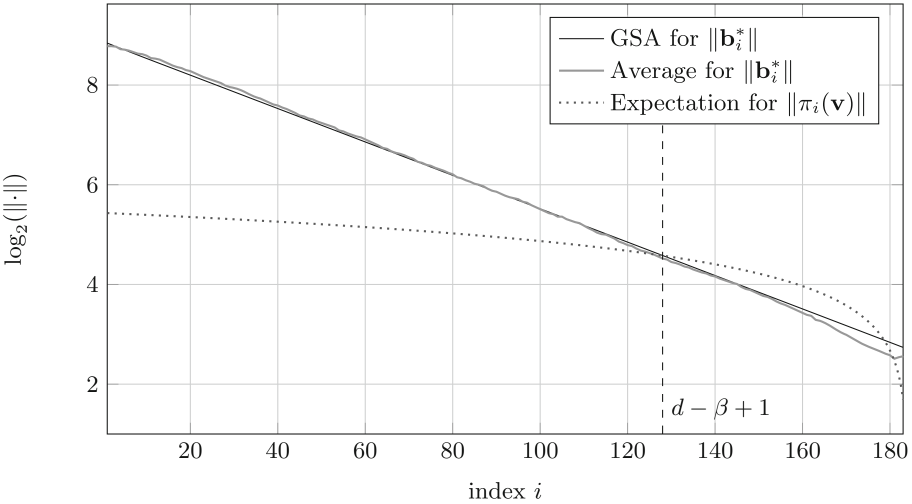

Fig. 3. Expected and observed norms for lattices of dimension $d = m + 1 = 183$ and volume q^{m-n} after BKZ-β reduction for LWE parameters $n = 65, m = 182, q = 521$ and standard deviation $\sigma = 8/\sqrt{2\pi}$ and $\beta = 56$ (minimal (β, m) such that (2) holds). Average of Gram-Schmidt lengths is taken over 16 BKZ-β reduced bases of random q-ary lattices, i.e. *without* an unusually short vector.

Table 1, show a related, albeit not identical behaviour. Defining a cut-off index $c = d - 0.9\beta + 1$ and considering $\pi_\kappa(\mathbf{v})$ for $\kappa < c$, we observe that the BKZ algorithm typically first recovers $\pi_\kappa(\mathbf{v})$ which is immediately followed by the recovery of $\mathbf{v}$ in the same step. In more detail, in Fig. 4 we show the measured probability distribution of the index κ such that $\mathbf{v}$ is recovered from $\pi_\kappa(\mathbf{v})$ in the same step. Note that the mean of this distribution is smaller than $d - \beta + 1$. We explain this bias in Sect. 4.3.

The recovery of $\mathbf{v}$ from $\pi_\kappa(\mathbf{v})$ can be effected by one of three subroutines: either by a call to LLL, by a call to size reduction, or by a call to enumeration that recovers $\mathbf{v}$ directly. Since LLL itself contains many calls to size reduction, and enumeration being lucky is rather unlikely, size reduction is a good place to start the investigation. Indeed, restricting the LLL calls in Algorithm 2 as outlined in Sect. 2.3, identifies that size reduction suffices. That is, to measure the success rate of size reduction recovering $\mathbf{v}$ from $\pi_\kappa(\mathbf{v})$, we observe size reduction acting on $\pi_\kappa(\mathbf{v})$. Here, we consider size reduction to fail in recovering $\mathbf{v}$ if it does not recover $\mathbf{v}$ given $\pi_\kappa(\mathbf{v})$ for $\kappa < c$ with $c = d - 0.9\beta + 1$, regardless of whether $\mathbf{v}$ is finally recovered at a later point either by size reduction on a new projection, or by some other call in the algorithm such as an SVP oracle call at a smaller index. As shown in Table 1, size reduction's success rate is close to 1. Note that the cut-off index c serves to limit underestimating the success rate: intuitively we do not expect size reduction to succeed when starting from a projection with larger index, such as $\pi_{d-\gamma+1}(\mathbf{v})$ with $\gamma < 10$. We discuss this in Sect. 4.3.

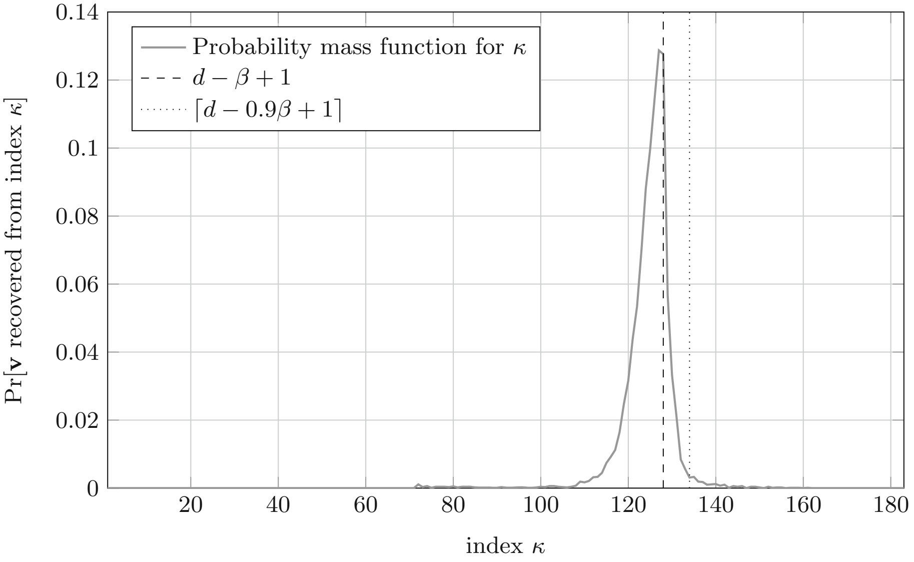

Fig. 4. Probability mass function of the index κ from which size reduction recovers $\mathbf{v}$, calculated over 10,000 lattice instances with LWE parameters $n = 65, m = 182, q = 521$ and standard deviation $\sigma = 8/\sqrt{2\pi}$, reduced using $\beta = 56$. The mean of the distribution is ≈124.76 while $d - \beta + 1 = 128$.

Overall, Table 1 confirms the prediction from [ADPS16]: picking $\beta = \beta_{2016}$ to be the block size predicted by the 2016 estimate leads to a successful recovery of $\mathbf{v}$ with high probability.

4.3 Explaining Observation

As noted above, our experiments indicate that the algorithm behaves better than expected by (2). Firstly, the BKZ algorithm does not necessarily recover a projection of $\mathbf{v}$ at index $d - \beta + 1$. Instead, the index κ at which we recover a projection $\pi_\kappa(\mathbf{v})$ follows a distribution with a centre below $d - \beta + 1$, cf. Fig. 4. Secondly, size reduction usually immediately recovers $\mathbf{v}$ from $\pi_\kappa(\mathbf{v})$. This is somewhat unexpected, since we do not have the guarantee that $|c_i| \leq 1/2$ as required in the success condition of size reduction given in (3).

Finding the projection. To explain the bias towards a recovery of $\pi_\kappa(\mathbf{v})$ for some $\kappa < d - \beta + 1$, note that if (2) holds then for the parameter sets *in our experiments* the lines for $\|\pi_i(\mathbf{v})\|$ and $\|\mathbf{b}_i^*\|$ intersect twice (cf. Fig. 3). Let $d - \gamma + 1$ be the index of the second intersection. Thus, there is a good chance that $\|\pi_{d-\gamma+1}(\mathbf{v})\|$ is a shortest vector in the lattice spanned by the last projected block of some small rank γ and will be placed at index $d-\gamma+1$. As a consequence,

Table 1. Overall success rate ("$\mathbf{v}$") and success rate of size reduction ("same step") for solving LWE instances characterised by n, σ, q with m samples, standard deviation $\sigma = 8/\sqrt{2\pi}$, minimal (β_{2016}, m_{2016}) such that $\sqrt{b_{2016}}\,\sigma \leq \delta_0^{2\beta_{2016}-(m_{2016}+1)} q^{(m_{2016}-n)/(m_{2016}+1)}$ with δ_0 in function of β_{2016}. The column "β" gives the actual block size used in experiments. The "same step" rate is calculated over all successful instances where $\mathbf{v}$ is found before the cut-off point c and for the instances where exactly $\pi_{d-b+1}(\mathbf{v})$ is found (if no such instance is found, we do not report a value). In the second case, the sample size is smaller, since not all instances recover $\mathbf{v}$ from exactly $\kappa = d - \beta + 1$. The column "time" lists average solving CPU time for one instance, in seconds. Note that our changes to the algorithm and our extensive record keeping lead to an increased running time of the BKZ algorithm compared to [FPL17,FPY17]. Furthermore, the occasional longer running time for smaller block sizes is explained by the absence of early termination when $\mathbf{v}$ is found.

n	q	β_{2016}	m_{2016}	β	#	$\mathbf{v}$	Same step		Time
							$\kappa < c$	$\kappa = d - \beta + 1$	
65	521	56	182	56	10000	93.3%	99.7%	99.7%	1,131.4
				51		52.8%	98.8%	97.3%	1,359.3
				46		4.8%	96.4%	85.7%	1,541.2
80	1031	60	204	60	1000	94.2%	99.6%	100.0%	2,929.0
				55		60.6%	99.3%	96.5%	2,458.5
				50		8.9%	97.6%	100.0%	1,955.0
				45		0.2%	100.0%	—	1,568.1
100	2053	67	243	67	500	88.8%	99.8%	100.0%	28,803.7
				62		39.6%	99.5%	100.0%	19,341.9
				57		5.8%	100.0%	100.0%	7,882.2
				52		0.2%	0.0%	—	3,227.0
108	2053	77	261	77	5	100.0%	100.0%	100.0%	351,094.2
110	2053	78	272	78	5	100.0%	100.0%	100.0%	1,012,634.8

all projections $\pi_i(\mathbf{v})$ with $i > d - \gamma + 1$ will be zero and $\pi_{d-\beta-\gamma+1}(\mathbf{v})$ will be contained in the β-dimensional lattice

$$\Lambda_{d-\beta-\gamma+1} := \Lambda\left(\pi_{d-\beta-\gamma+1}(\mathbf{b}_{d-\beta-\gamma+1}), \ldots, \pi_{d-\beta-\gamma+1}(\mathbf{b}_{d-\gamma+1})\right),$$

enabling it to be recovered by BKZ-β at an index $d - \beta - \gamma + 1 < d - \beta + 1$. Thus, BKZ in our experiments behaves better than predicted by (2). We note that another effect of this second intersection is that, for very few instances, it directly leads to a recovery of $\mathbf{v}$ from $\pi_{d-\beta-\gamma+1}(\mathbf{v})$.

Giving a closed formula incorporating this effect akin to (2) would entail to predict the index γ and then replace β with $\beta+\gamma$ in (2). However, as illustrated in Fig. 3, neither does the GSA hold for the last 50 or so indices of the basis [Che13] nor does the prediction $\sqrt{d-i+1}\,\sigma$ for $\|\pi_{d-1+1}(\mathbf{v})\|$.

We stress that while the second intersection often occurs for parameter sets within reach of practical experiments, it does not always occur for all parameter sets. That is, for many large parameter sets (n, α, q), e.g. those in [ADPS16], a choice of β satisfy (2) does *not* lead to a predicted second intersection at some larger index. Thus, this effect may highlight the pitfalls of extrapolating experimental lattice-reduction data from small instances to large instances.

Finding the short vector. In what follows, we assume that the projected norm $\|\pi_{d-k}(\mathbf{v})\|$ is indeed equal to this expected norm (cf. Fig. 2). We further assume that $\pi_i(\mathbf{v})$ is distributed in a random direction with respect to the rest of the basis. This assumption holds for LWE where the vector $\mathbf{e}$ is sampled from a (near) spherical distribution. We also note that we can rerandomise the basis and thus the relative directions. Under this assumption, we show that size reduction recovers the short vector $\mathbf{v}$ with high probability. More precisely, we show:

Claim 1. *Let $\mathbf{v} \in \Lambda \subset \mathbb{R}^d$ be a unique shortest vector and $\beta \in \mathbb{N}$. Assume that (2) holds, the current basis is $\mathbf{B} = \{\mathbf{b}_1, \dots, \mathbf{b}_d\}$ such that $\mathbf{b}_\kappa^* = \pi_\kappa(\mathbf{v})$ for $\kappa = d - \beta + 1$ and*

$$\mathbf{v} = \mathbf{b}_k + \sum_{i=1}^{k-1} \nu_i \mathbf{b}_i$$

for some $\nu_i \in \mathbb{Z}$, and the GSA holds for $\mathbf{B}$ until index κ. If the size reduction step of BKZ-β is called on $\mathbf{b}_\kappa$, it recovers $\mathbf{v}$ with high probability over the randomness of the basis.

Note that if BKZ has just found a projection of $\mathbf{v}$ at index κ, the current basis is as required by Claim 1. Now, let $\nu_i \in \mathbb{Z}$ denote the coefficients of $\mathbf{v}$ with respect to the basis $\mathbf{B}$, i.e.

$$\mathbf{v} = \mathbf{b}_{d-\beta+1} + \sum_{i=1}^{d-\beta} \nu_i \mathbf{b}_i.$$

Let $\mathbf{b}_{d-\beta+1}^{(d-\beta+1)} = \mathbf{b}_{d-\beta+1}$, where the superscript denotes a step during size reduction. For $i = d-\beta, d-\beta-1, \dots, 1$ size-reduction successively finds $\mu_i \in \mathbb{Z}$ such that

$$\mathbf{w}_i = \mu_i \pi_i(\mathbf{b}_i) + \pi_i(\mathbf{b}_{d-\beta+1}^{(i+1)}) = \mu_i \mathbf{b}_i^* + \pi_i(\mathbf{b}_{d-\beta+1}^{(i+1)})$$

is the shortest element in the coset

$$L_i := \{\mu \mathbf{b}_i^* + \pi_i(\mathbf{b}_{d-\beta+1}^{(i+1)}) | \mu \in \mathbb{Z}\}$$

and sets

$$\mathbf{b}_{d-\beta+1}^{(i)} := \mu_i \mathbf{b}_i + \mathbf{b}_{d-\beta+1}^{(i+1)}.$$

Note that if $\mathbf{b}_{d-\beta+1}^{(i+1)} = \mathbf{b}_{d-\beta+1} + \sum_{j=i+1}^{d-\beta} \nu_j \mathbf{b}_j$, as in the first step $i = d - \beta$, then we have that

$$\pi_i(\mathbf{v}) = \nu_i \mathbf{b}_i^* + \pi_i(\mathbf{b}_{d-\beta+1}^{(i+1)}) \in L_i$$

is contained in L_i and hence

$$L_i = \pi_i(\mathbf{v}) + \mathbb{Z}\mathbf{b}_i^*.$$

If the projection $\pi_i(\mathbf{v})$ is in fact the shortest element in L_i, for the newly defined vector $\mathbf{b}_{d-\beta+1}^{(i)}$ it also holds that

$$\mathbf{b}_{d-\beta+1}^{(i)} = \nu_i \mathbf{b}_i + \mathbf{b}_{d-\beta+1}^{(i+1)} = \mathbf{b}_{d-\beta+1} + \sum_{j=i}^{d-\beta} \nu_j \mathbf{b}_j.$$

Hence, if $\pi_i(\mathbf{v})$ is the shortest element in L_i for all i, size reduction finds the shortest vector

$$\mathbf{v} = \mathbf{b}_{d-\beta+1}^{(1)}$$

and inserts it into the basis at position $d-\beta+1$, replacing $\mathbf{b}_{d-\beta+1}$.

It remains to argue that with high probability p for every i we have that the projection $\pi_i(\mathbf{v})$ is the shortest element in L_i. The success probability p is given by

$$p = \prod_{i=1}^{d-\beta} p_i,$$

where the probabilities p_i are defined as

$$p_i = \Pr\left[\pi_i(\mathbf{v}) \text{ is the shortest element in } \pi_i(\mathbf{v}) + \mathbb{Z}\mathbf{b}_i^*\right].$$

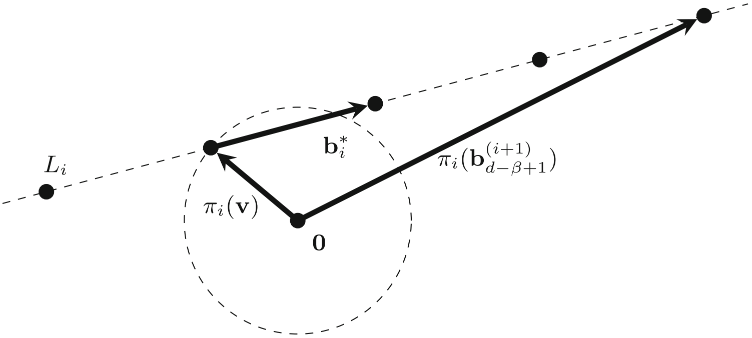

Fig. 5. Illustration of a case such that $\pi_i(\mathbf{v})$ is the shortest element on L_i.

For each i the probability p_i is equal to the probability that

$$\|\pi_i(\mathbf{v})\| < \min\{\|\pi_i(\mathbf{v}) + \mathbf{b}_i^*\|, \|\pi_i(\mathbf{v}) - \mathbf{b}_i^*\|\}$$

as illustrated in Fig. 5. To approximate the probabilities p_i, we model them as follows. By assumption, we have

$$r_i := \|\pi_i(\mathbf{v})\| = (\sqrt{d-i+1}/\sqrt{d})\,\|\mathbf{v}\| \text{ and } R_i := \|\mathbf{b}_i^*\| = \delta_0^{-2(i-1)+d}\mathrm{Vol}(\Lambda)^{\frac{1}{d}},$$

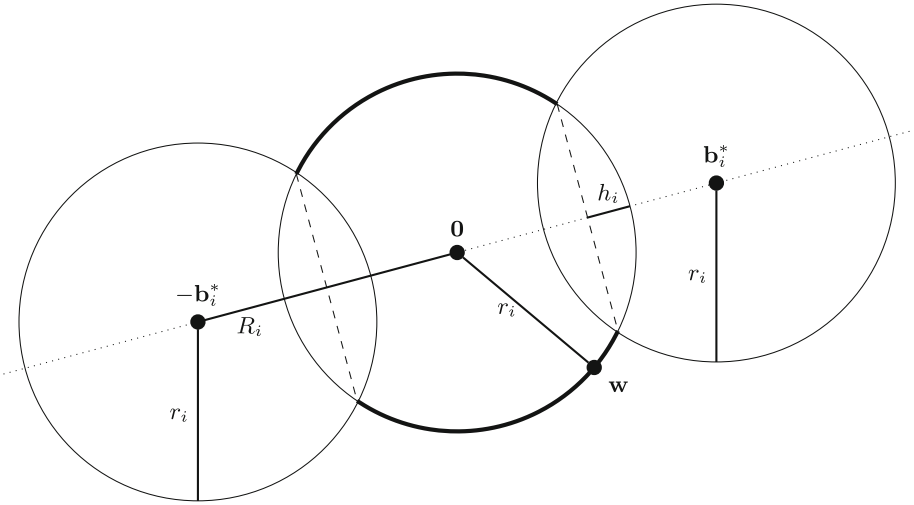

Fig. 6. Illustration of the success probability p_i in $\mathbb{R}^2$. If $\mathbf{w}$ is on the thick part of the circle, step i of size reduction is successful.

and that $\pi_i(\mathbf{v})$ is uniformly distributed with norm r_i. We can therefore model p_i as described in the following and illustrated in Fig. 6.

Pick a point $\mathbf{w}$ with norm r_i uniformly at random. Then the probability p_i is approximately the probability that $\mathbf{w}$ is closer to $\mathbf{0}$ than it is to $\mathbf{b}_i^*$ and to $-\mathbf{b}_i^*$, i.e.

$$r_i < \min\{\|\mathbf{w} - \mathbf{b}_i^*\|, \|\mathbf{w} + \mathbf{b}_i^*\|\}.$$

Calculating this probability leads to the following approximation of p_i

$$p_i \approx \begin{cases} 1 - \frac{2A_{d-i+1}(r_i,h_i)}{A_{d-i+1}(r_i)} & \text{if } R_i < 2r_i \\ 1 & \text{if } R_i \geq 2r_i \end{cases},$$

where $A_{d-i+1}(r_i)$ is the surface area of the sphere in $\mathbb{R}^{d-i+1}$ with radius r_i and $A_{d-i+1}(r_i, h_i)$ is the surface area of the hyperspherical cap of the sphere in $\mathbb{R}^{d-i+1}$ with radius r_i of height h_i with $h_i = r_i - R_i/2$. Using the formulas provided in [Li11], an easy calculation leads to

$$p_i \approx \begin{cases} 1 - \frac{\int_0^{2\frac{h_i}{r_i} - \left(\frac{h_i}{r_i}\right)^2} t^{((d-i)/2)-1}(1-t)^{-1/2} dt}{B(\frac{d-i}{2}, \frac{1}{2})} & \text{if } R_i < 2r_i \\ 1 & \text{if } R_i \geq 2r_i \end{cases},$$

where $B(\cdot, \cdot)$ denotes the Euler beta function. Note that $R_i \geq 2r_i$ corresponds to (3).

Estimated success probabilities p for different block sizes β are plotted in Fig. 7. Note that if we assume equality holds in (2), the success probability p only depends on the block size β and not on the specific lattice dimension, volume of the lattice, or the length of the unique short vector, since then the ratios between the predicted norms $\|\pi_{d-\beta+1-k}(\mathbf{v})\|$ and $\left\|\mathbf{b}^*_{d-\beta+1-k}\right\|$ only depend on β for all $k = 1, 2, \ldots$, since

$$\frac{\|\pi_{d-\beta+1-k}(\mathbf{v})\|}{\left\|\mathbf{b}^*_{d-\beta+1-k}\right\|} = \frac{\frac{\sqrt{\beta}\sqrt{\beta+k}}{\sqrt{\beta}\sqrt{d}}\|\mathbf{v}\|}{\delta_0^{2(\beta+k)-d}\operatorname{Vol}(\Lambda)^{\frac{1}{d}}} = \frac{\frac{\sqrt{\beta+k}}{\sqrt{\beta}}\delta_0^{2\beta-d}\operatorname{Vol}(\Lambda)^{\frac{1}{d}}}{\delta_0^{2(\beta+k)-d}\operatorname{Vol}(\Lambda)^{\frac{1}{d}}} = \frac{\sqrt{\beta+k}}{\sqrt{\beta}}\delta_0^{-2k}$$

and the estimated success probability only depends on these ratios.

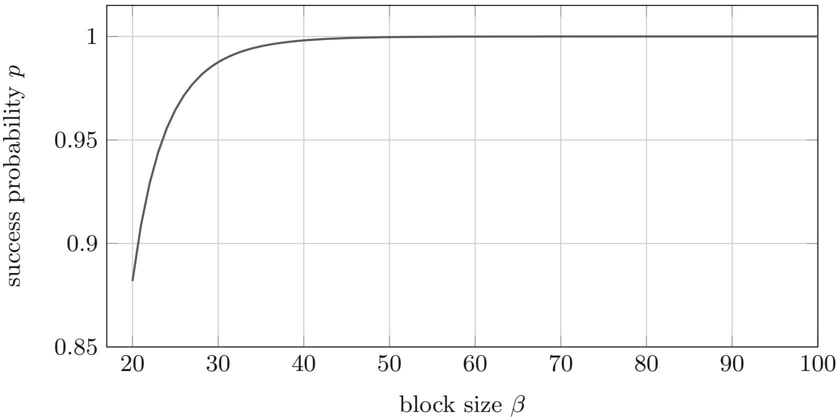

Fig. 7. Estimated success probability p for varying block sizes β, assuming β is chosen minimal such that (2) holds.

The prediction given in Fig. 7 is in line with the measured probability of finding $\mathbf{v}$ in the same step when its projection $\pi_{d-\beta+1}(\mathbf{v})$ is found as reported in Table 1 for $\beta = \beta_{2016}$ and $m = m_{2016}$. Finally, note that by the above analysis we do not expect to recover $\mathbf{v}$ from a projection $\pi_{d-\gamma+1}(\mathbf{v})$ for some small $\gamma \ll \beta$ except with small probability.

5 Applications

Section 4 indicates that (2) is a reliable indicator for when lattice-reduction will succeed in recovering an unusually short vector. Furthermore, as illustrated in Fig. 1, applying (2) lowers the required block sizes compared to the 2008 model which is heavily relied upon in the literature. Thus, in this section we evaluate the impact of applying the revised estimates to various parameter sets from the literature. Indeed, for many schemes we find that their parameters need to be adapted to maintain the currently claimed level of security.

Many of the schemes considered below feature an unusually short secret $\mathbf{s}$ where $s_i \leftarrow_{\$} \{-B, \ldots, B\}$ for some small $B \in \mathbb{Z}_q$. Furthermore, some schemes pick the secret to also be sparse such that most components of $\mathbf{s}$ are zero. Thus, before we apply the revised 2016 estimate, we briefly recall the alternative embedding due to Bai and Galbraith [BG14b] which takes these small (and sparse) secrets into account.

5.1 Bai and Galbraith's Embedding

Consider an LWE instance in matrix form $(\mathbf{A}, \mathbf{c}) \equiv (\mathbf{A}, \mathbf{A} \cdot \mathbf{s} + \mathbf{e}) \in \mathbb{Z}_q^{m \times n} \times \mathbb{Z}_q^m$. By inspection, it can be seen that the vector $(\nu \mathbf{s} \mid \mathbf{e} \mid 1)$, for some $\nu \neq 0$, is contained in the lattice Λ

$$\Lambda = \left\{ \mathbf{x} \in (\nu\mathbb{Z})^n \times \mathbb{Z}^{m+1} \ \mid \mathbf{x} \cdot \left(\frac{1}{\nu}\mathbf{A} \mid \mathbf{I}_m \mid -\mathbf{c} \right)^\top \equiv 0 \bmod q \right\}, \tag{4}$$

where ν allows to balance the size of the secret and the noise. An $(n+m+1) \times (n+m+1)$ basis $\mathbf{M}$ for Λ can be constructed as

$$\mathbf{M} = \begin{pmatrix} \nu \mathbf{I}_n & -\mathbf{A}^\top & \mathbf{0} \\ \mathbf{0} & q\mathbf{I}_m & \mathbf{0} \\ \mathbf{0} & \mathbf{c} & 1 \end{pmatrix}.$$

Indeed, $\mathbf{M}$ is full rank, $\det(\mathbf{M}) = \mathrm{Vol}(\Lambda)$, and the integer span of $\mathbf{M} \subseteq \Lambda$, as we can see by noting that

$$\begin{pmatrix} \nu \mathbf{I}_n & -\mathbf{A}^\top & \mathbf{0} \\ \mathbf{0} & q\mathbf{I}_m & \mathbf{0} \\ \mathbf{0} & \mathbf{c} & 1 \end{pmatrix} \left(\frac{1}{\nu}\mathbf{A} \mid \mathbf{I}_m \mid -\mathbf{c} \right)^\top = (\mathbf{A} - \mathbf{A} \mid q\mathbf{I}_m \mid \mathbf{c} - \mathbf{c})^\top \equiv \mathbf{0} \bmod q.$$

Finally, note that $(\mathbf{s} \mid * \mid 1) \cdot \mathbf{M} = (\nu \mathbf{s} \mid \mathbf{e} \mid 1)$ for suitable values of $*$. If $\mathbf{s}$ is small and/or sparse, choosing $\nu = 1$, the vector $(\mathbf{s} \mid \mathbf{e} \mid 1)$ is unbalanced, i.e. $\frac{\|\mathbf{s}\|}{\sqrt{n}} \ll \frac{\|\mathbf{e}\|}{\sqrt{m}} \approx \sigma$, where σ is the standard deviation of the LWE error distribution. We may then want to rebalance it by choosing an appropriate value of ν such that $\|(\nu \mathbf{s} \mid \mathbf{e} \mid 1)\| \approx \sigma\sqrt{n+m}$. Rebalancing preserves $(\nu \mathbf{s} \mid \mathbf{e} \mid 1)$ as the unique shortest vector in the lattice, while at the same time increasing the volume of the lattice being reduced, reducing the block size required by (2).

If $\mathbf{s} \xleftarrow{\$} \{-1, 0, 1\}^n$ we expect $\|\nu \mathbf{s}\|^2 \approx \frac{2}{3}\nu^2 n$. Therefore, we can chose $\nu = \sqrt{\frac{3}{2}}\sigma$ to obtain $\|\nu \mathbf{s}\| \approx \sigma\sqrt{n}$, so that $\|(\mathbf{s} \mid \mathbf{e} \mid 1)\| \approx \sigma\sqrt{n+m}$. Similarly, if $w < n$ entries of $\mathbf{s}$ are non-zero from $\{-1, 1\}$, we have $\|\nu \mathbf{s}\|^2 = w\,\nu^2$. Choosing $\nu = \sqrt{\frac{n}{w}}\sigma$, we obtain a vector $\nu \mathbf{s}$ of length $\sigma\sqrt{n}$.

In the case of sparse secrets, combinatorial techniques can also be applied [HG07, BGPW16, Alb17]. Given a secret $\mathbf{s}$ with at most $w < n$ non-zero entries, we guess k entries of $\mathbf{s}$ to be 0, therefore decreasing the dimension of the lattice to consider. For each guess, we then apply lattice reduction to recover the remaining components of the vector $(\mathbf{s} \mid \mathbf{e} \mid 1)$. Therefore, when estimating the overall complexity for solving such instances, we find $\min_k \{1/p_k \cdot C(n-k)\}$ where $C(n)$ is the cost of running BKZ on a lattice of dimension n and p_k is the probability of guessing correctly.

5.2 Estimates

In what follows, we assume that the geometry of (4) is sufficiently close to that of (1) so that we transfer the analysis as is. Furthermore, we will denote applying (2) from [ADPS16] for Kannan's embedding as "Kannan" and applying (2) for Bai and Galbraith's embedding [BG14b] as "Bai-Gal". Unless stated otherwise, we will assume that calling BKZ with block size β in dimension d costs $8\,d\,2^{0.292\,\beta+16.4}$ operations [BDGL16, Alb17].

Lizard [CKLS16b, CKLS16a] is a PKE scheme based on the Learning With Rounding problem, using a small, sparse secret. The authors provide a reduction to LWE, and security parameters against classic and quantum adversaries, following their analysis. In particular, they cost BKZ by a single call to sieving on a block of size β. They estimate this call to cost $\beta\,2^{c\,\beta}$ operations where $c = 0.292$ for classical adversaries, $c = 0.265$ for quantum ones and $c = 0.2075$ as a lower bound for sieving ("paranoid"). Applying the revised 2016 cost estimate for the primal attack to the parameters suggested in [CKLS16b] (using their sieving cost model as described above) reduces the expected costs, as shown in Table 2. We note that in the meantime the authors of Lizard have updated their parameters in [CKLS16a].

Table 2. Bit complexity estimates λ for solving Lizard PKE [CKLS16b] as given in [CKLS16b] and using Kannan's resp. Bai and Galbraith's embedding under the 2016 estimate. The dimension of the LWE secret is n. In all cases, BKZ-β is estimated to cost $\beta\,2^{c\,\beta}$ operations.

	Classical			Quantum			Paranoid		
n, $\log_2 q$, σ	386, 11, 2.04			414, 11, 2.09			504, 12, 4.20		
Cost	β	d	λ	β	d	λ	β	d	λ
[CKLS16b]	418	—	130.8	456	—	129.7	590	—	131.6
Kannan	372	805	117.2	400	873	114.6	567	1120	126.8
Bai-Gal	270	646	88.5	297	692	86.9	372	833	85.9

HElib [GHS12a, GHS12b] is an FHE library implementing the BGV scheme [BGH13]. A recent work [Alb17] provides revised security estimates for HELib by employing a dual attack exploiting the small and sparse secret, using the same cost estimate for BKZ as given at the beginning of this section. In Table 3 we provide costs for a primal attack using Kannan's and Bai and Galbraith's embeddings. Primal attacks perform worse than the algorithm described [Alb17], but, as expected, under the 2016 estimate the gap narrows.

Table 3. Solving costs for LWE instances underlying HELib as given in [Alb17] and using Kannan's resp. Bai and Galbraith's embedding under the 2016 estimate. The dimension of the LWE secret is n. In all cases, BKZ-β is estimated to cost $8d\,2^{0.292\,\beta+16.4}$ operations.

80 bit security															
n	1024			2048			4096			8192			16384		
$\log_2 q$, σ	47, 3.2			87, 3.2			167, 3.2			326, 3.2			638, 3.2		
Cost	β	d	λ	β	d	λ	β	d	λ	β	d	λ	β	d	λ
[Alb17] $\text{SILKE}_{\text{sparse}}$	105	—	61.3	111	—	65.0	112	—	67.0	123	—	70.2	134	—	73.1
Kannan	156	2096	76.0	166	4003	79.8	171	7960	82.3	176	15606	84.7	180	31847	86.9
Bai-Gal	137	1944	70.3	152	3906	75.9	163	7753	79.9	169	16053	82.9	173	32003	85.9
128 bit security															
n	1024			2048			4096			8192			16384		
$\log_2 q$, σ	38, 3.2			70, 3.2			134, 3.2			261, 3.2			511, 3.2		
Cost	β	d	λ	β	d	λ	β	d	λ	β	d	λ	β	d	λ
[Alb17] $\text{SILKE}_{\text{sparse}}$	138	—	73.2	145	—	77.4	151	—	81.2	163	—	84.0	149	—	86.4
Kannan	225	2076	96.1	238	4050	100.9	245	8011	103.9	250	16017	106.4	257	31635	109.4
Bai-Gal	189	1901	86.6	211	3830	94.4	204	7348	99.3	185	13543	102.8	204	28236	105.9

SEAL [CLP17] is an FHE library by Microsoft, based on the FV scheme [FV12]. Up to date parameters are given in [CLP17], using the same cost model for BKZ as mentioned at the beginning of this section. In Table 4, we provide complexity estimates for Kannan's and Bai and Galbraith's embeddings under the 2016 estimate. Note that the gap in solving time between the dual and primal attack reported in [Alb17] is closed for SEAL v2.1 parameters.

Table 4. Solving costs for parameter choices in SEAL v2.1 as given in [CLP17], using [Alb17] as implemented in the current [APS15] estimator commit `84014b6` ("[Alb17]+"), and using Kannan's resp. Bai and Galbraith's embedding under the 2016 estimate. In all cases, BKZ-β is estimated to cost $8d\,2^{0.292\,\beta+16.4}$ operations.

n, $\log_2 q$, σ	1024, 35, 3.19			2048, 60, 3.19			4096, 116, 3.19			8192, 226, 3.19			16384, 435, 3.19		
Cost	β	d	λ	β	d	λ	β	d	λ	β	d	λ	β	d	λ
[CLP17]	230	—	97.6	282	—	115.1	297	—	119.1	307	—	123.1	329	—	130.5
[Alb17]+	255	—	104.9	298	—	118.4	304	—	121.2	310	—	124.0	328	—	130.2
Kannan	257	2085	105.5	304	4041	120.2	307	8047	122.0	312	15876	124.5	328	31599	130.1
Bai-Gal	237	1984	99.6	288	4011	115.5	299	8048	119.7	309	15729	123.6	326	31322	129.5

TESLA [BG14a, ABBD15] is a signature scheme based on LWE. Post-quantum secure parameters in the quantum random oracle model were recently proposed in [ABB+17]. In Table 5, we show that these parameters need to be increased to maintain the currently claimed level of security under the 2016 estimate. Note that [ABB+17] maintains a gap of $\approx \log_2 n$ bits of security between the best known attack on LWE and claimed security to account for a loss of security in the reduction.

Table 5. Bit complexity estimates for solving TESLA parameter sets [ABB+17]. The entry "[ABB+17]+" refers to reproducing the estimates from [ABB+17] using a current copy of the estimator from [APS15] which uses $t = 1$ instead of $t = \|\mathbf{e}\|$, as a consequence the values in the respective rows are slightly lower than in [ABB+17]. We compare with Kannan's embedding under the 2016 estimate. Classically, BKZ-β is estimated to cost $8d\,2^{0.292\,\beta+16.4}$ operations; quantumly BKZ-β is estimated to cost $8d\,\sqrt{\beta^{0.0225\,\beta} \cdot 2^{0.4574\,\beta}/2^{\beta/4}}$ operations in [ABB+17].

	TESLA-0			TESLA-1			TESLA-2		
n, $\log_2 q$, σ	644, 31, 55			804, 31, 57			1300, 35, 73		
Cost	β	d	λ	β	d	λ	β	d	λ
Classical									
[ABB+17]	—	—	110.0	—	—	142.0	—	—	204.0
[ABB+17]+	255	—	110.0	358	—	140.4	563	—	200.9
Kannan	248	1514	102.4	339	1954	129.3	525	3014	184.3
Post-Quantum									
[ABB+17]	—	—	71.0	—	—	94.0	—	—	142.0
[ABB+17]+	255	—	68.5	358	—	90.7	563	—	136.4
Kannan	248	1415	61.5	339	1954	81.1	525	3014	122.4

BCIV17 [BCIV17] is a somewhat homomorphic encryption scheme obtained as a simplification of the FV scheme [FV12] and proposed as a candidate for enabling privacy friendly energy consumption forecast computation in smart grid settings. The authors propose parameters for obtaining 80 bits of security, derived using the estimator from [APS15] available at the time of publication. As a consequence of applying (2), we observe a moderate loss of security, as reported in Table 6.

Table 6. Solving costs for proposed Ring-LWE parameters in [BCIV17] using Kannan's resp. Bai and Galbraith's embedding under the 2016 estimate. In both cases, BKZ-β is estimated to cost $8d\,2^{0.292\,\beta+16.4}$ operations.

80 bit security							
$n = 4096$, $\log_2 q = 186$, $\sigma = 102$							
Embedding	β	d	λ	Embedding	β	d	λ
Kannan	156	8105	77.9	Bai-Gal	147	7818	75.3

Acknowledgements. We thank Léo Ducas and Rachel Player for helpful discussions.

References

[ABB+17] Alkim, E., Bindel, N., Buchmann, J., Dagdelen, Ö., Eaton, E., Gutoski, G., Krämer, J., Pawlega, F.: Revisiting TESLA in the quantum random oracle model. In: Lange, T., Takagi, T. (eds.) PQCrypto 2017. LNCS, vol. 10346, pp. 143–162. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-59879-6_9

[ABBD15] Alkim, E., Bindel, N., Buchmann, J., Dagdelen, Ö.: TESLA: tightly-secure efficient signatures from standard lattices. Cryptology ePrint Archive, Report 2015/755 (2015). http://eprint.iacr.org/2015/755

[ADPS16] Alkim, E., Ducas, L., Pöppelmann, T., Schwabe, P.: Post-quantum key exchange - a new hope. In: Holz, T., Savage, S. (eds.) 25th USENIX Security Symposium, USENIX Security 16, pp. 327–343. USENIX Association (2016)

[AFG14] Albrecht, M.R., Fitzpatrick, R., Göpfert, F.: On the efficacy of solving LWE by reduction to unique-SVP. In: Lee, H.-S., Han, D.-G. (eds.) ICISC 2013. LNCS, vol. 8565, pp. 293–310. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-12160-4_18

[Alb17] Albrecht, M.R.: On dual lattice attacks against small-secret LWE and parameter choices in HElib and SEAL. In: Coron, J.-S., Nielsen, J.B. (eds.) EUROCRYPT 2017. LNCS, vol. 10211, pp. 103–129. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-56614-6_4

[APS15] Albrecht, M.R., Player, R., Scott, S.: On the concrete hardness of learning with errors. J. Math. Cryptol. **9**(3), 169–203 (2015)

[Bab86] Babai, L.: On lovász' lattice reduction and the nearest lattice point problem. Combinatorica **6**(1), 1–13 (1986)

[BCD+16] Bos, J.W., Costello, C., Ducas, L., Mironov, I., Naehrig, M., Nikolaenko, V., Raghunathan, A., Stebila, D.: Frodo: take off the ring! Practical, quantum-secure key exchange from LWE. In: Weippl, E.R., Katzenbeisser, S., Kruegel, C., Myers, A.C., Halevi, S. (eds.) ACM CCS 16, pp. 1006–1018. ACM Press, October 2016

[BCIV17] Bos, J.W., Castryck, W., Iliashenko, I., Vercauteren, F.: Privacy-friendly forecasting for the smart grid using homomorphic encryption and the group method of data handling. In: Joye, M., Nitaj, A. (eds.) AFRICACRYPT 2017. LNCS, vol. 10239, pp. 184–201. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-57339-7_11

[BDGL16] 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, January 2016

[BDK+17] Bos, J., Ducas, L., Kiltz, E., Lepoint, T., Lyubashevsky, V., Schanck, J.M., Schwabe, P., Stehlé, D.: CRYSTALS - kyber: a CCA-secure module-lattice-based KEM. Cryptology ePrint Archive, Report 2017/634 (2017). http://eprint.iacr.org/2017/634

[BG14a] Bai, S., Galbraith, S.D.: An improved compression technique for signatures based on learning with errors. In: Benaloh, J. (ed.) CT-RSA 2014. LNCS, vol. 8366, pp. 28–47. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-04852-9_2

[BG14b] Bai, S., Galbraith, S.D.: Lattice decoding attacks on binary LWE. In: Susilo, W., Mu, Y. (eds.) ACISP 2014. LNCS, vol. 8544, pp. 322–337. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-08344-5_21

[BGH13] Brakerski, Z., Gentry, C., Halevi, S.: Packed ciphertexts in LWE-based homomorphic encryption. In: Kurosawa, K., Hanaoka, G. (eds.) PKC 2013. LNCS, vol. 7778, pp. 1–13. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-36362-7_1

[BGPW16] Buchmann, J., Göpfert, F., Player, R., Wunderer, T.: On the hardness of LWE with binary error: revisiting the hybrid lattice-reduction and meet-in-the-middle attack. In: Pointcheval, D., Nitaj, A., Rachidi, T. (eds.) AFRICACRYPT 2016. LNCS, vol. 9646, pp. 24–43. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-31517-1_2

[BV11] Brakerski, Z., Vaikuntanathan, V.: Efficient fully homomorphic encryption from (standard) LWE. In: Ostrovsky, R. (ed.) 52nd FOCS, pp. 97–106. IEEE Computer Society Press, October 2011

[BVWW16] Brakerski, Z., Vaikuntanathan, V., Wee, H., Wichs, D.: Obfuscating conjunctions under entropic ring LWE. In: Sudan, M. (ed.) ITCS 2016, pp. 147–156. ACM, January 2016

[Che13] Chen, Y.: Réduction de réseau et sécurité concrète du chiffrement complètement homomorphe. Ph.D. thesis, Paris 7 (2013)

[CHK+17] Cheon, J.H., Han, K., Kim, J., Lee, C., Son, Y.: A practical post-quantum public-key cryptosystem based on spLWE. In: Hong, S., Park, J.H. (eds.) ICISC 2016. LNCS, vol. 10157, pp. 51–74. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-53177-9_3

[CKLS16a] Cheon, J.H., Kim, D., Lee, J., Song, Y.: Lizard: cut off the tail! Practical post-quantum public-key encryption from LWE and LWR. Cryptology ePrint Archive, Report 2016/1126 (2016). http://eprint.iacr.org/2016/1126

[CKLS16b] Cheon, J.H., Kim, D., Lee, J., Song, Y.: Lizard: cut off the tail! Practical post-quantum public-key encryption from LWE and LWR. Cryptology ePrint Archive, Report 2016/1126 (20161222:071525) (2016). http://eprint.iacr.org/2016/1126/20161222:071525

[CLP17] Chen, H., Laine, K., Player, R.: Simple encrypted arithmetic library - SEAL v2.1. Cryptology ePrint Archive, Report 2017/224 (2017). http://eprint.iacr.org/2017/224

[CN11] Chen, Y., Nguyen, P.Q.: BKZ 2.0: better lattice security estimates. In: Lee, D.H., Wang, X. (eds.) ASIACRYPT 2011. LNCS, vol. 7073, pp. 1–20. Springer, Heidelberg (2011). https://doi.org/10.1007/978-3-642-25385-0_1

[FPL17] The FPLLL development team: FPLLL, a lattice reduction library (2017). https://github.com/fplll/fplll

[FPY17] The FPYLLL development team: FPYLLL, a Python (2 and 3) wrapper for FPLLL (2017). https://github.com/fplll/fpylll

[FV12] Fan, J., Vercauteren, F.: Somewhat practical fully homomorphic encryption. Cryptology ePrint Archive, Report 2012/144 (2012). http://eprint.iacr.org/2012/144

[GHS12a] Gentry, C., Halevi, S., Smart, N.P.: Homomorphic evaluation of the AES circuit. Cryptology ePrint Archive, Report 2012/099 (2012). http://eprint.iacr.org/2012/099

[GHS12b] Gentry, C., Halevi, S., Smart, N.P.: Homomorphic evaluation of the AES circuit. In: Safavi-Naini and Canetti [SNC12], pp. 850–867

[GN08a] Gama, N., Nguyen, P.Q.: Finding short lattice vectors within Mordell's inequality. In: Ladner, R.E., Dwork, C. (ed.) 40th ACM STOC, pp. 207–216. ACM Press, May 2008

[GN08b] Gama, N., Nguyen, P.Q.: Predicting lattice reduction. In: Smart, N. (ed.) EUROCRYPT 2008. LNCS, vol. 4965, pp. 31–51. Springer, Heidelberg (2008). https://doi.org/10.1007/978-3-540-78967-3_3

[GNR10] Gama, N., Nguyen, P.Q., Regev, O.: Lattice enumeration using extreme pruning. In: Gilbert, H. (ed.) EUROCRYPT 2010. LNCS, vol. 6110, pp. 257–278. Springer, Heidelberg (2010). https://doi.org/10.1007/978-3-642-13190-5_13

[GSW13] Gentry, C., Sahai, A., Waters, B.: Homomorphic encryption from learning with errors: conceptually-simpler, asymptotically-faster, attribute-based. In: Canetti, R., Garay, J.A. (eds.) CRYPTO 2013. LNCS, vol. 8042, pp. 75–92. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-40041-4_5

[Gö16] Göpfert, F.: Securely instantiating cryptographic schemes based on the learning with errors assumption. Ph.D. thesis, Technische Universität Darmstadt (2016). http://tuprints.ulb.tu-darmstadt.de/5850/

[HG07] 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 (2007). https://doi.org/10.1007/978-3-540-74143-5_9

[HKM17] Herold, G., Kirshanova, E., May, A.: On the asymptotic complexity of solving LWE. Des. Codes Crypt. (2017). https://link.springer.com/article/10.1007/s10623-016-0326-0

[HPS11] Hanrot, G., Pujol, X., Stehlé, D.: Analyzing blockwise lattice algorithms using dynamical systems. In: Rogaway, P. (ed.) CRYPTO 2011. LNCS, vol. 6841, pp. 447–464. Springer, Heidelberg (2011). https://doi.org/10.1007/978-3-642-22792-9_25

[Kan87] Kannan, R.: Minkowski's convex body theorem and integer programming. Math. Oper. Res. **12**(3), 415–440 (1987)

[Laa14] Laarhoven, T.: Sieving for shortest vectors in lattices using angular locality-sensitive hashing. Cryptology ePrint Archive, Report 2014/744 (2014). http://eprint.iacr.org/2014/744

[Li11] Li, S.: Concise formulas for the area and volume of a hyperspherical cap. Asian J. Math. Stat. **4**(1), 66–70 (2011)

[LLL82] Lenstra, A.K., Lenstra Jr., H.W., Lovász, L.: Factoring polynomials with rational coefficients. Math. Ann. **261**(4), 515–534 (1982)

[LM09] Lyubashevsky, V., Micciancio, D.: On bounded distance decoding, unique shortest vectors, and the minimum distance problem. In: Halevi, S. (ed.) CRYPTO 2009. LNCS, vol. 5677, pp. 577–594. Springer, Heidelberg (2009). https://doi.org/10.1007/978-3-642-03356-8_34

[LN13] Liu, M., Nguyen, P.Q.: Solving BDD by enumeration: an update. In: Dawson, E. (ed.) CT-RSA 2013. LNCS, vol. 7779, pp. 293–309. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-36095-4_19

[LO83] Lagarias, J.C., Odlyzko, A.M.: Solving low-density subset sum problems. In: 24th FOCS, pp. 1–10. IEEE Computer Society Press, November 1983

[LP11] Lindner, R., Peikert, C.: Better key sizes (and attacks) for LWE-based encryption. In: Kiayias, A. (ed.) CT-RSA 2011. LNCS, vol. 6558, pp. 319–339. Springer, Heidelberg (2011). https://doi.org/10.1007/978-3-642-19074-2_21

[MR09] Micciancio, D., Regev, O.: Lattice-based cryptography. In: Bernstein, D.J., Buchmann, J., Dahmen, E. (eds.) Post-Quantum Cryptography, pp. 147–191. Springer, Heidelberg (2009). https://doi.org/10.1007/978-3-540-88702-7_5

[MW16] Micciancio, D., Walter, M.: Practical, predictable lattice basis reduction. In: Fischlin, M., Coron, J.-S. (eds.) EUROCRYPT 2016. LNCS, vol. 9665, pp. 820–849. Springer, Heidelberg (2016). https://doi.org/10.1007/978-3-662-49890-3_31

[Reg09] Regev, O.: On lattices, learning with errors, random linear codes, and cryptography. J. ACM **56**(6), 1–40 (2009)

[S+17] Stein, W., et al.: Sage Mathematics Software Version 7.5.1. The Sage Development Team (2017). http://www.sagemath.org

[Sch87] Schnorr, C.-P.: A hierarchy of polynomial time lattice basis reduction algorithms. Theoret. Comput. Sci. **53**, 201–224 (1987)

[Sch03] Schnorr, C.P.: Lattice reduction by random sampling and birthday methods. In: Alt, H., Habib, M. (eds.) STACS 2003. LNCS, vol. 2607, pp. 145–156. Springer, Heidelberg (2003). https://doi.org/10.1007/3-540-36494-3_14

[SNC12] Safavi-Naini, R., Canetti, R. (eds.): CRYPTO 2012. LNCS, vol. 7417. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-32009-5