# Refined Approx-SVP Rank Reduction Conditions and Adaptive Lattice Reduction for MSIS Security Estimation

Xiaohan Zhang, Zijian Zhou, and Longjing Qu(✉)

College of Science, National University of Defense Technology, Changsha 410073, Hunan, China
ljqu_happy@hotmail.com

**Abstract.** The security of lattice-based cryptography relies critically on the concrete hardness of the approximate shortest vector problem (Approx-SVP). For cryptographic-sized instances, existing Approx-SVP rank reduction conditions may be overly aggressive, as they implicitly assume access to a large number of extremely short lattice vectors. In this work, we systematize and refine Approx-SVP rank reduction conditions from a feasibility perspective. We identify that, in the context of dimension-for-free (D4f) technique, the existence of a single sufficiently short vector is the essential requirement, and we derive two refined and compact rank reduction conditions accordingly. The first condition is based on geometric properties of lattice sieving, while the second incorporates a basis-quality-dependent probabilistic bound. These results are validated through extensive experiments on high-dimensional lattices, where the compact condition outperforms prior methods by up to a factor of 60 in dimensions 850 and 925. To reliably realize these conditions in high dimensions, we present APBKZ, an adaptive Pump-based lattice reduction strategy that dynamically selects the blocksize and dimension-for-free parameters according to the evolving Gram–Schmidt profile. We further introduce HeadAPBKZ, a head-focused execution mode that restricts reduction to a critical prefix once the rank reduction condition is satisfied. Combining these advances, we develop an improved concrete security estimation framework for the MSIS problem. Applied to Dilithium, our analysis indicates that when integrating compact rank reduction behavior with D4f technique, the estimated concrete security margin of Dilithium drops by 9.50–16.63 bits compared to the conservative Core-SVP baseline, offering more accurate security benchmarks for cryptographic standardization.

**Keywords:** Approximate Shortest Vector Problem · Rank Reduction · Lattice Reduction · BKZ Algorithms · MSIS Security Estimation · Post-Quantum Cryptography

## 1 Introduction

Lattice-based cryptography is a cornerstone of post-quantum cryptography, with security mainly based on the hardness of problems such as Learning with Errors (LWE) [24] and Short Integer Solution (SIS)[1] problem. In concrete security evaluations, these problems are typically reduced to instances of the approximate shortest vector problem (Approx-SVP). As a result, the reliability of cryptographic parameter selection critically depends on how accurately the concrete feasibility of Approx-SVP can be assessed under realistic algorithmic assumptions.

*Rank reduction for Approx-SVP.* Modern lattice attacks do not attempt to solve Approx-SVP in the full lattice dimension. Instead, they rely on *rank reduction*: after sufficient basis reduction, the attacker searches for a short vector in a lower-dimensional prefix sublattice [17,2,14,31]. The feasibility of such attacks is governed by a *rank reduction condition*, which specifies how small the prefix dimension $k$ can be while still enabling Approx-SVP with a given approximation factor.

A prominent rank reduction condition for Approx-SVP was proposed by Laarhoven and Mariano [17] in the context of progressive lattice sieving. However, this condition implicitly assumes that the sieving procedure can reliably access an exponentially large set of vectors with norms close to the Gaussian Heuristic. While theoretically meaningful, this assumption is often misaligned with practical high-dimensional lattice reduction, where feasibility is determined by the existence of only a small number of sufficiently short vectors.

*From idealized rank reduction to practical feasibility.* In practice, lattice sieving typically returns vectors whose norms are bounded by a constant factor above the Gaussian Heuristic, most commonly within $\sqrt{4/3}$. The above analysis builds on the geometric modeling of lattice sieving introduced by Nguyen and Vidick [22] and further refined in subsequent works such as [20]. Under this model, short lattice vectors are treated as samples from a thin spherical shell, and the success of sieving is linked to the existence of sufficiently many vector pairs with favorable angular relations. This framework has become standard in the analysis of modern sieving-based SVP algorithms [28,29,7,16,8,18,9,6,17,2,14,31].

In particular, when the dimension-for-free (D4f) technique is employed to reduce the cost of sieving [13], the sieving does not require access to all short vectors, but only to the existence of *at least one* vector below a certain length threshold [4].

This gap between idealized rank reduction and practical feasibility motivates a refined formulation of Approx-SVP rank reduction conditions for cryptographic-sized lattices. In this work, we develop two such conditions that explicitly distinguish what is required for *feasibility* from what is merely sufficient in theory.

*Lattice reduction in high dimensions.* Accurately realizing refined rank reduction conditions in high dimensions requires lattice reduction algorithms that

behave robustly under practical constraints. Existing BKZ variants, such as PnjBKZ within the G6K framework, rely on simulator-driven parameter choices, fixed jump schedules and aggressive D4f estimates [2]. In high dimensions, these strategies frequently lead to unsaturated reduction and unstable basis quality, undermining the feasibility assumptions required for rank reduction.

Motivated by these limitations, we develop *adaptive lattice reduction strategies* that dynamically adjust their behavior based on the evolving quality of the basis. Our proposed algorithm, APBKZ (Adaptive Pump BKZ), adaptively selects blocksizes and dimension-for-free parameters without relying on simulator predictions or jump scheduling. Building on this approach, we further introduce HeadAPBKZ, a head-focused execution mode that restricts reduction to a critical prefix once a compact rank reduction condition is satisfied, reflecting empirical observations that late-stage tail reduction has negligible impact on feasibility [30].

*Security estimation of MSIS.* A refined understanding of Approx-SVP rank reduction has direct implications for the concrete security estimation of SIS and MSIS-based cryptographic schemes. Notably, the Dilithium specification [4] highlights the lack of security estimates that accurately account for modern reduction techniques such as D4f and progressive sieving. Existing estimates based on the conservative Core-SVP model implicitly assume access to vectors at the Gaussian Heuristic bound, which may not reflect realistic attack feasibility.

By combining the compact rank reduction condition with adaptive lattice reduction strategies, we develop an improved security estimation framework for MSIS. Our analysis provides tighter and more realistic lower-bound estimates for the security of Dilithium.

*Our contributions.* The main contributions of this work are summarized as follows:

- We systematize and refine Approx-SVP rank reduction conditions by replacing idealized assumptions with feasibility-oriented criteria that reflect the probabilistic behavior of lattice sieving. In particular, we introduce a practical rank reduction condition derived from geometric sieving bounds, and a compact rank reduction condition that incorporates basis-quality-dependent sieving guarantees.
- We introduce APBKZ, an adaptive lattice reduction strategy designed to robustly realize rank reduction feasibility in high dimensions, and HeadAPBKZ, a head-focused execution mode that exploits the compact rank reduction condition, mitigating early-dimension ineffectiveness and unsaturated reduction.
- We empirically validate the accuracy of the proposed rank reduction conditions and the robustness of the adaptive reduction strategies on high-dimensional $q$-ary lattices from the TU Darmstadt Lattice Challenge. Experiments demonstrate that our compact rank reduction condition outperforms the practical rank reduction baseline by a factor of up to 60 in dimensions 850 and 925, as detailed in Table 1 and Section 5.

– Building on these results, we integrate these results into a refined security estimation framework for MSIS and apply it to Dilithium, showing that accounting for modern reduction techniques significantly affects concrete security estimates. Compared to the conservative Core-SVP model, our security model indicates that Dilithium's security decreases by 9.50–16.63 bits, offering more accurate security benchmarks for cryptographic standardization.

Table 1: Our proposed compact rank reduction against lattice challenges

| Dim | $n$ | Wall time | Previous wall time | Speedup Factor |
|---|---|---|---|---|
| 850 | 230 | $31.2m$ | $1.40d$ | **64.6**× |
| 925 | 295 | $2m$ | $2.07h$ | **62.1**× |
| 1025 | 330 | $1.89h$ | $1.22d$ | 15.5× |

*Organization.* The paper is organized as follows. Section 2 provides the necessary preliminaries of lattice reduction and hard problems. Section 3 revisits its Approx-SVP rank reduction and introduces refined feasibility conditions. Section 4 presents adaptive lattice reduction strategies, including APBKZ and HeadAPBKZ. Section 5 reports experimental results. Section 6 applies our framework to MSIS security estimation and evaluates Dilithium.

## 2 Preliminaries

### 2.1 Notations and Basic Definitions

In this paper, column vectors are denoted by bold lowercase letters (e.g., $\mathbf{b}$), and matrices are denoted by bold capital letters (e.g., $\mathbf{B}$). We write a matrix $\mathbf{B}$ as $\mathbf{B} = (\mathbf{b}_1, \ldots, \mathbf{b}_n)$, where $\mathbf{b}_i$ denotes the $i$-th column vector of $\mathbf{B}$. If $\mathbf{B} \in \mathbb{R}^{m \times n}$ has full column rank $n$, the lattice generated by $\mathbf{B}$ is denoted by $\mathcal{L}(\mathbf{B}) = \{\mathbf{Bx} \mid \mathbf{x} \in \mathbb{Z}^n\}$. Let $(\mathbf{b}_1^*, \ldots, \mathbf{b}_n^*)$ denote the Gram-Schmidt orthogonalization of the basis $(\mathbf{b}_1, \ldots, \mathbf{b}_n)$. For $i \in \{1, \ldots, n\}$, we denote by $\pi_i$ the orthogonal projection onto the orthogonal complement of $\mathrm{span}(\mathbf{b}_1, \ldots, \mathbf{b}_{i-1})$. For $1 \leq i < j \leq n$, we define $\mathbf{B}_{[i:j]} = (\mathbf{b}_i, \ldots, \mathbf{b}_j)$ and denote by $\mathcal{L}_{[i:j]} = \mathcal{L}(\mathbf{B}_{[i:j]})$ the sublattice of $\mathcal{L}(\mathbf{B})$ generated by these basis vectors. In contrast, we define the locally projected block $\mathbf{B}_{\pi[i:j]} = (\pi_i(\mathbf{b}_i), \ldots, \pi_i(\mathbf{b}_j))$ and denote by $\mathcal{L}_{\pi[i:j]} = \mathcal{L}(\mathbf{B}_{\pi[i:j]})$ the lattice generated by the projected vectors. For convenience, we use $\mathbf{B}_{[i]}$ and $\mathcal{L}_{[i]}$ as shorthands for $\mathbf{B}_{[i:n]}$ and $\mathcal{L}_{[i:n]}$, respectively, and similarly $\mathbf{B}_{\pi[i]}$ and $\mathcal{L}_{\pi[i]}$ as shorthands for $\mathbf{B}_{\pi[i:n]}$ and $\mathcal{L}_{\pi[i:n]}$.

Let BKZ-$\beta$ be an abbreviation of a one-tour BKZ with blocksize $\beta$, and BKZ-$\beta$ reduced basis be the lattice basis after calling sufficient tours of BKZ-$\beta$. For simplification, we use $\beta$ to imply the quality of a BKZ-$\beta$ reduced basis. The Euclidean norm of a vector $\mathbf{v}$ is denoted by $\|\mathbf{v}\|$. The volume of a lattice $\mathcal{L}(\mathbf{B})$

is $\mathrm{Vol}(\mathcal{L}(\mathbf{B})) = \prod_i \|\mathbf{b}_i^*\|$, that is an invariant of the lattice. The first minimum of a lattice $\mathcal{L}$ is the length of a shortest non-zero vector, denoted by $\lambda_1(\mathcal{L})$. We use the abbreviations $\mathrm{Vol}(\mathbf{B}) = \mathrm{Vol}(\mathcal{L}(\mathbf{B}))$ and $\lambda_1(\mathbf{B}) = \lambda_1(\mathcal{L}(\mathbf{B}))$.

**Definition 1 (Gaussian Heuristic [21]).** *Let $S \subset \mathbb{R}^n$ be a measurable body. Then the number of lattice points in $S$ is approximately equal to $\mathrm{Vol}(S)/\mathrm{Vol}(\mathcal{L})$.*

We denote $|S \cap \mathcal{L}|$ as the number of lattice points in $S$. Note that the number of lattice points the Gaussian Heuristic indicates is exactly the expected number of lattice points in a random translation of $S$. When applying the Gaussian Heuristic to an $n$-dimensional ball of volume $\mathrm{Vol}(\mathcal{L})$ we obtain that the minimal length $\lambda_1(\mathcal{L})$ is approximately the radius of this ball, which asymptotically means that $\lambda_1(\mathcal{L}) \approx \sqrt{n/(2\pi e)} \cdot \mathrm{Vol}(\mathcal{L})^{1/n}$. For a lattice $\mathcal{L} \subset \mathbb{R}^n$ we denote this radius by $\mathrm{gh}(\mathcal{L})$, and we also denote $\mathrm{gh}(n)$ for $\mathrm{gh}(\mathcal{L})$ of any $n$-dimensional lattice $\mathcal{L}$ of volume 1: $\mathrm{gh}(n) = \sqrt{n/2\pi e}$. Note that a ball of radius $\omega \cdot \mathrm{gh}(\mathcal{L})$ contains an exponential number of $\omega^d$ lattice vectors not much longer than the minimal length. We say that a list of lattice vectors saturates a volume $S$ if it contains some significant ratio [13], say 50%, of the lattice vectors in $\mathcal{L} \cap S$ as predicted by the Gaussian Heuristic. Here the saturation ratio is defined as $p = N_{\mathrm{list}}/N_{\mathrm{GH}}$, with $N_{\mathrm{list}} = |\mathbf{v} \in \mathrm{list} : \mathbf{v} \in S|$ and $N_{\mathrm{GH}} = \mathrm{Vol}(S)/\mathrm{Vol}(\mathcal{L})$. And we call the list $p$-saturated when $p$ reaches a predefined threshold.

**Definition 2 (Root Hermite Factor [11]).** *Let $\mathbf{B} = (\mathbf{b}_1, \ldots, \mathbf{b}_n)$ be a reduced basis for an $n$-dimensional lattice $L$, we define the root Hermite factor to be*

$$\delta = \left( \frac{\|\boldsymbol{b}_1\|}{\mathrm{Vol}(\mathcal{L})^{\frac{1}{n}}} \right)^{\frac{1}{n}}.$$

**Definition 3 (HKZ reduction and BKZ reduction [26]).** *The basis $\mathbf{B} = (\mathbf{b}_1, \ldots, \mathbf{b}_n)$ of a lattice $\mathcal{L}$ is said to be HKZ reduced if $\|\mathbf{b}_i^*\| = \lambda_1\left(\mathcal{L}\left(\mathbf{B}_{\pi[i:n]}\right)\right)$ for all $i \leq n$. It is said BKZ reduced with blocksize $\beta$ (for short BKZ-$\beta$ reduced) $\|\mathbf{b}_i^*\| = \lambda_1\left(\mathcal{L}\left(\mathbf{B}_{\pi[i:\min(i+\beta,n)]}\right)\right)$ for all $i \leq n$.*

Building upon the Definition 3, the Geometric Series Assumption provides a simplified model for BKZ-reduced bases:

**Definition 4 (Geometric Series Assumption (GSA) [25]).** *Let $\mathbf{B}$ be a BKZ-$\beta$ reduced basis of a lattice $\mathcal{L}$. The GSA states that:*

$$\|\boldsymbol{b}_i^*\| = \|\boldsymbol{b}_1^*\| \cdot \alpha_\beta^{i-1},$$

*where $\frac{\sqrt{3}}{2} \leq \alpha_\beta < 1$ and $\alpha_\beta$ is called GSA constant. This model is reasonably accurate in practice for $\beta > 50$ and $\beta \ll n$.*

Our analysis also relies on fundamental probability distributions that characterize random lattice properties:

**Definition 5 (The Exponential Distribution [5]).** *For $\lambda > 0$, a continuous random variable $X$ follows the exponential distribution* $\mathrm{Expo}(\lambda)$ *if its probability density function is given by*

$$f_X(x) = \lambda e^{-\lambda x}, \quad x \geq 0.$$

*The cumulative distribution function is $F_X(x) = 1 - e^{-\lambda x}$ for $x \geq 0$.*

**Definition 6 (The Gaussian Distribution [23]).** *Let $\sigma, \mu \in \mathbb{R}$ be the standard deviation and the mean value, respectively, of a continuous Gaussian Distribution denoted as $N\left(\mu, \sigma^2\right)$. Its probabilistic density function $\rho_{N(\mu,\sigma^2)} = \frac{e^{-\frac{(x-\mu)^2}{2\sigma^2}}}{\sigma\sqrt{2\pi}}$.*

These distributions play a crucial role in understanding the statistical properties of lattice parameters.

### 2.2 Lattice Hard Problems

We now introduce the fundamental computational problems that form the foundation of lattice-based cryptography. These problems are believed to be hard even for quantum computers, making them central to post-quantum cryptography.

**Definition 7 (Shortest Vector Problem (SVP) [15]).** *The SVP is defined as: given a basis $\mathbf{B}$ of a lattice $\mathcal{L}$, finding the shortest non-zero vector $\boldsymbol{v}$ in $\mathcal{L}$.*

**Definition 8 (Approximate Shortest Vector Problem (Approx-SVP) [15]).** *The Approx-SVP is defined as: let $\gamma \geq 1$, given a basis $\mathbf{B}$ of a lattice $\mathcal{L}$, finding a non-zero vector $\boldsymbol{v} \in \mathcal{L}$ such that $\|\mathbf{v}\| \leq \gamma \cdot \lambda_1(\mathcal{L})$.*

While SVP and Approx-SVP are fundamental lattice problems, modern cryptographic constructions typically rely on structured variants that offer better efficiency and security properties:

**Definition 9 (Module-LWE Problem (MLWE) [4]).** *For integers $m, k$, and a probability distribution $D : R_q \to [0, 1]$, we say that the advantage of algorithm $\mathcal{A}$ in solving the decisional* $\mathrm{MLWE}_{m,k,D}$ *problem over the ring $R_q$ is*

$$\begin{aligned}\mathrm{Adv}^{\mathrm{MLWE}}_{m,k,D}(\mathcal{A}) :=& \left| \Pr\left[b = 1 \mid \mathbf{A} \leftarrow R_q^{m\times k}; \mathbf{t} \leftarrow R_q^m; b \leftarrow \ \mathrm{A}(\mathbf{A}, \mathbf{t})\right]\right. \\ &\left. - \Pr\left[b = 1 \mid \mathbf{A} \leftarrow R_q^{m\times k}; \mathbf{s}_1 \leftarrow D^k; \mathbf{s}_2 \leftarrow D^m; b \leftarrow \ \mathrm{A}\left(\mathbf{A}, \mathbf{A}\mathbf{s}_1 + \mathbf{s}_2\right)\right] \right| .\end{aligned}$$

**Definition 10 (Short Integer Solution Problem (SIS) [4]).** *To an algorithm $\mathcal{A}$ we associate the advantage function* $\mathrm{Adv}^{\mathrm{SIS}}_{m,k,\gamma}$*, to solve the (Hermite Normal Form)* $\mathrm{SIS}_{m,k,\gamma}$ *problem over the ring $\mathbb{Z}_q$ as*

$$\mathrm{Adv}^{\mathrm{SIS}}_{m,k,\gamma}(\mathcal{A}) := \Pr\left[0 < \|\mathbf{y}\|_\infty \leq \gamma \wedge [\mathbf{I} \mid \mathbf{A}] \cdot \mathbf{y} = \mathbf{0} \mid \mathbf{A} \leftarrow \mathbb{Z}_q^{m\times k}; \mathbf{y} \leftarrow \mathcal{A}(\mathbf{A})\right].$$

**Definition 11 (Module-SIS Problem (MSIS) [4]).** *To an algorithm $\mathcal{A}$ we associate the advantage function* $\mathrm{Adv}^{\mathrm{MSIS}}_{m,k,\gamma}$*, to solve the* (*Hermite Normal Form*) $\mathrm{MSIS}_{m,k,\gamma}$ *problem over the ring $R_q$ as*

$$\mathrm{Adv}^{\mathrm{MSIS}}_{m,k,\gamma}(\mathcal{A}) := \Pr\left[0 < \|\mathbf{y}\|_\infty \leq \gamma \wedge [\mathbf{I} \mid \mathbf{A}] \cdot \mathbf{y} = \mathbf{0} \mid \mathbf{A} \leftarrow R_q^{m\times k}; \mathbf{y} \leftarrow \mathcal{A}(\mathbf{A})\right].$$

### 2.3 BKZ Simulator

The BKZ simulator, proposed by Chen and Nguyen [12], is a polynomial-time algorithm that simulates the evolution of lattice basis quality during BKZ-$\beta$ reduction. Based on the Gaussian Heuristic and properties of BKZ reduction, it estimates the changes in the logarithms of the Gram-Schmidt norms after $T$ tours of BKZ-$\beta$.

## 3 Refined Rank Reduction for Approx-SVP

In this section, we revisit the Approx-SVP rank reduction condition proposed by Laarhoven and Mariano [17], which is commonly used as a feasibility criterion in concrete security analyses, but may be overly aggressive for cryptographic-sized lattice instances. In particular, this classical condition is based on single-vector geometric bounds and does not explicitly account for the probabilistic output behavior of modern sieve-based SVP oracles.

Motivated by this limitation, we introduce two feasibility-oriented rank reduction conditions for Approx-SVP that reflect different levels of geometric modeling fidelity. The *practical* rank reduction condition is derived from the standard geometric model of lattice sieving introduced by Nguyen and Vidick [22], and captures probabilistic feasibility thresholds based on single-vector geometric bounds within this framework. The *compact* rank reduction condition refines this view by explicitly incorporating basis-quality-dependent sieving guarantees, leading to a more conservative and analytically robust feasibility criterion.

Throughout the remainder of the paper, these two conditions play distinct but complementary roles. The practical condition is primarily used for experimental analysis and validation, while the compact condition serves as the operative criterion for security estimation and algorithmic optimization in the subsequent sections.

### 3.1 Revisiting the Original Rank Reduction Condition

In [17], Laarhoven and Mariano provided a preliminary estimation of Approx-SVP rank reduction. Notably, their definition of the root Hermite factor differs from the original Definition 2. To avoid confusion, we denote their version as $\delta'$, where $\delta' = \left(\frac{\|\boldsymbol{b}_1\|}{\mathrm{Vol}(\mathcal{L})^{\frac{1}{n}}}\right)^{\frac{1}{n-1}}$.

Under the Geometric Series Assumption, a BKZ-reduced lattice basis $(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n)$ satisfies $\|\boldsymbol{b}_i^*\| \approx \delta^{n-2i+2} \cdot \mathrm{Vol}(\mathcal{L})^{1/n}$ for all $i = 1, \ldots, n$. The first $k$ basis vectors form a rank-$k$ sublattice $\mathcal{L}_{[1:k]} = \mathcal{L}(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_k) \subset \mathcal{L}$ of volume:

$$\mathrm{Vol}\left(\mathcal{L}_{[1:k]}\right) = \prod_{i=1}^{k} \|\boldsymbol{b}_i^*\| = \mathrm{Vol}(\mathcal{L})^{k/n} \cdot \delta^{k(n-k+1)}. \tag{1}$$

By the Gaussian Heuristic, the shortest vector in $\mathcal{L}_{[1:k]}$ has relative norm:

$$\mathrm{gh}(\mathcal{L}_{[1:k]}) \approx \delta^{(n-k+1)} \sqrt{\frac{k}{n}} \cdot \mathrm{gh}(\mathcal{L}). \tag{2}$$

To solve Approx-SVP with approximation factor $\gamma$, we require $\delta^{(n-k+1)}\sqrt{k/n} \approx \gamma$. This indicates how many dimensions can be effectively "skipped" when seeking approximate solutions.

### 3.2 Geometric Basis for Sieving Output Lengths

The original Approx-SVP rank reduction condition in [17] is based on the assumption that lattice sieving can reliably find vectors with norms close to the Gaussian Heuristic estimate $\mathrm{gh}(\mathcal{L})$. However, this represents a theoretical lower bound that is rarely achieved in practical high-dimensional lattice reduction. In practice, the success of rank reduction depends on the probabilistic behavior of lattice sieving rather than on achieving the Gaussian Heuristic bound exactly. To capture this probabilistic behavior, we adopt a geometric perspective on lattice sieving to derive a more practical rank reduction condition.

Importantly, the geometric properties used in our analysis arise from the standard geometric model employed in the analysis of lattice sieving algorithms. In this model, the analysis of lattice sieving treats short vectors encountered during the algorithm as if they were sampled independently and uniformly from a thin spherical shell, a heuristic introduced by Nguyen and Vidick [22]. Within this geometric framework, Nguyen and Vidick showed that the expected number of vector pairs whose mutual angle is at most $\pi/3$ grows exponentially with the dimension. Equivalently, this implies that the number of vectors of norm at most $\sqrt{4/3} \cdot \mathrm{gh}(\mathcal{L}_{[1:k]})$ produced by lattice sieving is exponential in $k$, provided that the basis of $\mathcal{L}_{[1:k]}$ is sufficiently reduced.

In practice, D4f-enabled sieving does not benefit from enumerating all short vectors returned by the sieve. Instead, feasibility hinges on the existence of a single sufficiently short vector that can initiate the projection-and-lifting procedure. Once such a vector is available, additional short vectors do not further reduce the asymptotic cost [4]. Consequently, rank reduction feasibility should be formulated in terms of a single-vector requirement rather than full list saturation.

Accordingly, we require that lattice sieving on the rank-$k$ prefix sublattice $\mathcal{L}_{[1:k]}$ produces at least one nonzero vector $\mathbf{x}$ satisfying

$$\|\mathbf{x}\| \;\leq\; \sqrt{\frac{4}{3}} \cdot \mathrm{gh}(\mathcal{L}_{[1:k]}).$$

Let $\gamma$ be the approximation factor of the target vector relative to $\mathrm{gh}(\mathcal{L})$, i.e., $\|\mathbf{s}\| = \gamma \cdot \mathrm{gh}(\mathcal{L})$, and since $\mathbf{s} \in \mathcal{L}_{[1:k]}$, it suffices that the following condition holds:

$$\gamma \cdot \mathrm{gh}(\mathcal{L}) \leq \sqrt{\frac{4}{3}} \cdot \mathrm{gh}\left(\mathcal{L}_{[1:k]}\right). \tag{3}$$

Combining the Gaussian Heuristic with inequalities (2) and (3), we derive:

$$\gamma \cdot \sqrt{\frac{n}{2\pi e}} \cdot \mathrm{Vol}(\mathcal{L})^{1/n} \leq \sqrt{\frac{4}{3}} \cdot \sqrt{\frac{k}{2\pi e}} \cdot \mathrm{Vol}(\mathcal{L})^{1/n} \cdot \delta^{(n-k+1)}, \tag{4}$$

The existence of such a vector in $\mathcal{L}_{[1:k]}$ therefore requires:

$$\gamma \leq \sqrt{\frac{4k}{3n}} \cdot \delta^{(n-k+1)}. \tag{5}$$

**Condition 1 (Practical Approx-SVP Rank Reduction Condition).** *Let $\mathcal{L}$ be a lattice of dimension $n$ with a BKZ-$\beta$ reduced basis and root Hermite factor $\delta$. A rank-$k$ prefix sublattice $\mathcal{L}_{[1:k]}$ is said to satisfy the* practical Approx-SVP rank reduction condition *if Inequality* (5) *holds.*

After sufficient BKZ reduction satisfying GSA, the desired short vector can be found by searching only within the first $k$ dimensions. For example, in an 850-dimensional $q$-ary lattice with root Hermite factor 1.668906, a short vector with approximation factor 1.007991 can be obtained by searching only the rank-164 sublattice.

### 3.3 Basis-Quality-Dependent Sieving Output Assumption

In this subsection, we formalize a probabilistic model for the output behavior of lattice sieving. Rather than assuming that a sieve deterministically returns a shortest vector, we model the sieve as a sampling procedure over short lattice vectors, whose output distribution depends on the quality of the reduced basis.

Concretely, we view the sieve as inducing a probability distribution over lattice vectors in $\mathcal{L}_{[1:k]}$, supported on vectors whose norms are bounded by a small constant multiple of the Gaussian Heuristic. Conditioned on the basis quality, a single invocation of the sieve outputs a vector $\mathbf{v} \in \mathcal{L}_{[1:k]}$ satisfying $\|\mathbf{v}\| \leq B \cdot \mathrm{gh}(\mathcal{L}_{[1:k]})$ with non-negligible probability, where the bound $B$ depends only on the reduction quality and the blocksize.

Importantly, our analysis does not assume that the sieve outputs all short vectors or succeeds deterministically. Instead, it relies solely on the existence of at least one sufficiently short output per sieve invocation. As a result, the rank reduction condition introduced below depends only on a non-negligible success probability and is independent of implementation-specific details of the sieve.

**Lemma 1 ([11]).** *Let $\Gamma_n = \{\mathcal{L} \in \mathbb{R}^n \mid \mathrm{Vol}(\mathcal{L}) = 1\}$ be the set of all full-rank lattices of rank $n$ with unit volume. For a random lattice $\mathcal{L}$ uniformly sampled from*

*the set $\Gamma_n$ of $n$-dimensional lattices with unit volume, the scaled first minimum $v_n \cdot \lambda_1(\mathcal{L})^n$ converges in distribution to the exponential distribution* $\mathrm{Expo}(1/2)$ *as the dimension $n \to \infty$, where $v_n$ denotes the volume of the $n$-dimensional unit ball.*

**Heuristic 1 (Basis-Quality-Dependent Sieving Output Assumption).** *Given an $n$-dimensional $q$-ary random lattice $\mathcal{L}$ with a BKZ-$\beta$ reduced basis, lattice sieving produces, with overwhelming probability, a nonzero vector $\mathbf{x} \in \mathcal{L}$ satisfying*

$$\|\mathbf{x}\| \leq (8\ln(10))^{1/\beta} \cdot \mathrm{gh}(\mathcal{L}),$$

*for blocksize $\beta > 50$, where the Gaussian Heuristic accurately captures lattice geometry.*

Here, the basis quality is characterized by the BKZ blocksize $\beta$. A larger $\beta$ corresponds to higher basis quality and yields a smaller exponential factor $(8\ln(10))^{1/\beta}$ in the norm upper bound, resulting in shorter output vectors from lattice sieving. This heuristic thus captures how the output quality of sieving depends on the preprocessing quality of the lattice basis in the context of concrete cryptographic instances. We explicitly note that this probabilistic bound is intended to model the behavior of sieving in practice and does not imply a polynomial-time algorithm for Approx-SVP in the asymptotic sense, as the required blocksize $\beta$ for validity grows with $n$. In practical settings considered here, solving Approx-SVP via sieving still incurs exponential computational cost.

We now justify the bound given in Heuristic 1. For a $\beta$-dimensional random lattice $\mathcal{L}$ of unit volume, Lemma 1 shows that the first minimum satisfies:

$$v_\beta \cdot \lambda_1(\mathcal{L})^\beta \sim \mathrm{Expo}(1/2), \tag{6}$$

where $v_\beta$ denotes the volume of the $\beta$-dimensional unit ball.

For the exponential distribution $\mathrm{Expo}(1/2)$, the cumulative distribution function gives:

$$P\left(v_\beta \cdot \lambda_1(\mathcal{L})^\beta \leq t\right) = 1 - e^{-t/2}. \tag{7}$$

Setting a high success probability (e.g., above 0.9999), we obtain:

$$1 - e^{-t/2} = 0.9999 \Rightarrow t = -2\ln(0.0001) = 8\ln(10). \tag{8}$$

Thus, the probabilistic upper bound satisfies:

$$v_\beta \cdot (\lambda_1^{\max})^\beta = 8\ln(10) \Rightarrow \lambda_1^{\max} = \left(\frac{8\ln(10)}{v_\beta}\right)^{1/\beta}. \tag{9}$$

Note that the Gaussian Heuristic indicates $\mathrm{gh}(\mathcal{L}) = v_\beta^{-1/\beta}$ for a unit-volume lattice, we have

$$\lambda_1^{\max} = (8\ln(10))^{1/\beta} \cdot \mathrm{gh}(\mathcal{L}). \tag{10}$$

By applying this bound to each local block of dimension $\beta$ encountered during BKZ-$\beta$ reduction, we obtain, with high confidence, a sieving capability ensuring the existence of a nonzero vector $\mathbf{x}$ satisfying

$$\|\mathbf{x}\| \leq (8\ln(10))^{1/\beta} \cdot \mathrm{gh}(\mathcal{L}). \tag{11}$$

In particular, for a rank-$k$ prefix sublattice $\mathcal{L}_{[1:k]}$, lattice sieving implies the existence of a vector $\mathbf{x} \in \mathcal{L}_{[1:k]}$ such that

$$\|\mathbf{x}\| \leq (8\ln(10))^{1/\beta} \cdot \mathrm{gh}(\mathcal{L}_{[1:k]}). \tag{12}$$

Expressing the Gaussian Heuristic of $\mathcal{L}_{[1:k]}$ in terms of the Gram–Schmidt profile of a BKZ-$\beta$ reduced basis yields the feasibility inequality

$$\gamma \leq (8\ln(10))^{1/\beta} \cdot \sqrt{\frac{k}{n}} \cdot \delta^{(n-k+1)}. \tag{13}$$

**Condition 2 (Compact Approx-SVP Rank Reduction Condition).** *Let $\mathcal{L}$ be a $q$-ary lattice of dimension $n$, and let $\beta$ be the BKZ blocksize with root Hermite factor $\delta$. A rank-$k$ prefix sublattice is said to satisfy the* compact Approx-SVP rank reduction condition *if Inequality* (13) *holds.*

The analysis above is therefore primarily theoretical. In practice, lattice reduction implementations often employ additional heuristics and engineering optimizations to improve empirical performance, which may lead to successful reductions beyond the predicted feasibility thresholds.

*Remark 1.* Re-randomization refers to multiplying the current lattice basis by a random unimodular matrix, typically instantiated as a random permutation matrix, followed by light reduction. This technique is widely adopted in practical lattice reduction algorithms such as BKZ 2.0 [12] and PBKZ [3], especially in the security evaluation of Dilithium [4].

While a formal proof remains open, extensive experiments on cryptographic-sized lattices consistently show that re-randomization aids in escaping local minima, effectively reducing the volume of the projected sublattice $\mathcal{L}_{[1:k]}$. We thus treat this improvement as an experimentally validated heuristic.

*Which rank reduction condition is used later.* Throughout the remainder of this paper, we explicitly distinguish between the roles of the two rank reduction conditions introduced in Section 3.

The *practical* Approx-SVP rank reduction condition (Condition 1) is used to interpret reduction behavior through the geometric lens of lattice sieving, as predicted by the standard spherical-shell model underlying modern sieve analyses. It captures feasibility at the level of single-vector geometric bounds, without explicitly accounting for basis-quality-dependent amplification, as validated in Section 5.2.

The *compact* Approx-SVP rank reduction condition (Condition 2) is adopted as the operative feasibility criterion for theoretical modeling, security estimation,

and optimization procedures in the remainder of this work, including the experimental analysis in Section 5.2 and 5.3, as well as the MSIS security evaluation in Section 6. This choice reflects its conservative nature and its derivation from basis-quality-dependent sieving guarantees.

# 4 Adaptive Lattice Reduction Strategies in High Dimensions

In this section, we present adaptive lattice reduction strategies aimed at improving the stability and robustness of lattice reduction in high dimensions. Unlike classical BKZ variants that rely on simulator-driven parameter choices and fixed reduction schedules, the strategies introduced here adapt their behavior dynamically based on the evolving quality of the lattice basis.

In particular, we introduce APBKZ as a general-purpose adaptive reduction strategy, and a head-focused variant, HeadAPBKZ, which restricts the reduction scope once a compact rank reduction condition has been satisfied. Together, these strategies form a flexible reduction framework suitable for high-dimensional lattice reduction and subsequent security evaluation.

## 4.1 APBKZ: Adaptive Pump BKZ for Stable Lattice Reduction

The PnjBKZ algorithm is among the most efficient lattice reduction methods in high dimensions. Its performance is governed by three key parameters: the blocksize $\beta$, the dimension-for-free parameter $f$, and the jump parameter $j$. At a structural level, these parameters control successive invocations of the core reduction primitive in the G6K framework, namely the $\text{Pump}_{\kappa,f,\beta}$ procedure [2], which applies lattice sieving to the local block $[\kappa : \kappa + \beta]$ and incorporates the dimension-for-free technique by skipping $f$ dimensions to reduce computational cost. In practice, $\beta$ and $j$ are typically selected using a BKZ simulator based on the Gaussian Heuristic and predicted Gram-Schmidt profiles, while $f$ is chosen according to D4f estimates that aim to recover vectors close to $\text{gh}(\mathcal{L})$.

However, the effectiveness of PnjBKZ in high dimensions critically depends on simulator-driven parameter choices and static reduction schedules, both of which may become unreliable when the actual basis quality deviates from Gaussian Heuristic predictions. In particular, inaccurate blocksize selection, overly optimistic dimension-for-free estimates, and aggressive jump strategies often lead to unsaturated reduction, especially in early blocks of high-dimensional $q$-ary lattices.

Compared to classical PnjBKZ, APBKZ differs in three fundamental aspects. First, it eliminates simulator-driven blocksize prediction and instead derives local blocksize directly from the evolving Gram–Schmidt profile of the basis. Second, it replaces static dimension-for-free estimates with basis-quality-dependent rules that adapt to the current reduction state, thereby avoiding reliance on idealized HKZ-reduced assumptions. Third, it enforces full coverage of early blocks during

each reduction tour, preventing unsaturated reduction caused by aggressive jump scheduling.

These design choices reflect the central philosophy of APBKZ: lattice reduction in high dimensions should be guided by the actual quality of the current basis, rather than by fixed schedules or simulator-based predictions.

APBKZ is governed by three adaptive design rules, which jointly determine its reduction behavior.

*Rule 1: Adaptive blocksize selection.* For a given index $i$, APBKZ selects the local blocksize $\beta_i$ as the smallest value such that $\mathrm{gh}(\mathcal{L}_{\pi[i:i+\beta_i-1]}) < \|\mathbf{b}_i^*\|$. This condition ensures that the current block is sufficiently reduced to allow further improvement, while avoiding unnecessarily large blocksizes that incur excessive computational cost.

*Rule 2: Adaptive dimension-for-free estimation.* Given the selected blocksize $\beta_i$, the dimension-for-free parameter $f_i$ is chosen as the largest value satisfying

$$\sqrt{\frac{4}{3}} \cdot \mathrm{gh}(\mathcal{L}_{\pi[i+f_i:i+\beta_i-1]}) \;\geq\; \|\mathbf{b}_i^*\|.$$

Unlike classical D4f rules that target vectors close to $\mathrm{gh}(\mathcal{L})$, this criterion only requires outperforming the current Gram–Schmidt norm. As a result, multiple vectors may qualify in practice, which significantly mitigates unsaturated reduction. It is worth noting that, in principle, one could attempt to determine the D4f parameter using traditional estimates that assume the existence of vectors close to $\mathrm{gh}(\mathcal{L})$. However, such estimates implicitly rely on the lattice basis being close to HKZ-reduced, an assumption that is rarely met in high-dimensional practical reductions and would require substantially higher computational cost to enforce. In contrast, the adaptive rule adopted here is explicitly aligned with the actual quality of the current basis and provides a lightweight alternative that avoids the overhead of pursuing idealized reduction states.

*Rule 3: Scope control via a runtime upper bound.* Although the D4f value $f_i$ is selected adaptively and may vary across blocks, it is always bounded by the local blocksize. APBKZ therefore maintains a runtime upper bound $f_{\max} = \max_i f_i$ over a reduction tour and restricts subsequent reduction to indices $i \leq n - f_{\max} - 50$. This constraint is empirically motivated to ensure robust saturation behavior. Restricting the pump scope to indices $i \leq n - f_{\max} - 50$ guarantees that the underlying lattice reduction subroutines always operate on projected sublattices of dimension at least 50. In this regime, the behavior of lattice reduction aligns well with GSA, whereas in lower dimensions, boundary effects often render GSA-based predictions unreliable [12].

Algorithm 1 summarizes the APBKZ algorithm and explicitly implements the above adaptive rules. Notably, APBKZ does not rely on any rank reduction condition and can be applied as a general-purpose adaptive reduction strategy for SVP, Approx-SVP.

**Algorithm 1** APBKZ: Adaptive Pump BKZ

**Input:** Lattice basis $\mathbf{B} = (\mathbf{b}_1, \ldots, \mathbf{b}_n)$, maximum blocksize $\beta_{\max}$, maximum number of tours $T_{\max}$.
**Output:** Reduced lattice basis $\mathbf{B}$.

```
tour ← 0
f_max ← 0
while tour < T_max do
    Compute Gram–Schmidt norms ‖b_i^*‖ for all i
    Determine reduction bound i_max ← n − f_max − 50                ▷ Rule 3
    for i = 1 to i_max do
        Select adaptive blocksize β_i ≤ β_max                       ▷ Rule 1
        Select adaptive dimension-for-free value f_i                ▷ Rule 2
        f_max ← max(f_max, f_i)
        B ← Pump_{i,f_i,β_i}(B)
    end for
    tour ← tour + 1
    if no further improvement detected then
        break
    end if
end while
return B
```

### 4.2 HeadAPBKZ: Head-Focused Reduction under Rank Reduction Conditions

Once the compact Approx-SVP rank reduction condition (Condition 2) is satisfied for a prefix of length $k$, further reduction of the lattice tail has negligible impact on feasibility. This observation is consistent with prior empirical findings [30], which indicate that during the final stages of BKZ-style reduction, applying SVP subroutines to the first few indices is typically sufficient.

Our rank reduction framework formalizes this phenomenon by providing an explicit criterion on the required prefix length. Under this condition, it suffices for the lattice basis to satisfy the Geometric Series Assumption within the rank-$k$ sublattice $\mathcal{L}_{[1:k]}$. Restricting reduction to this prefix does not affect correctness, as sufficiently short vectors found in early blocks propagate through subsequent size reduction and basis updates.

HeadAPBKZ exploits this property as a restricted execution mode of APBKZ. It inherits the same adaptive blocksize and dimension-for-free selection rules, but limits the reduction scope to indices

$$i \leq \min\big(k,\ n - f_{\max} - 50\big),$$

where $f_{\max}$ denotes a conservative upper bound on the dimension-for-free parameter encountered during reduction. This joint constraint reflects both the rank reduction condition, which determines the relevant prefix length, and the Gaus-

sian Heuristic, which ensures that the final Pump operation acts on a sublattice of sufficiently large dimension.

In our experiments, the transition from APBKZ to HeadAPBKZ is triggered immediately once the compact rank reduction condition (Condition 2) is met. This condition-driven switch concentrates computational effort on the most impactful part of the basis, while avoiding unnecessary reduction of the tail that would only increase computational cost without improving feasibility.

## 5 Experiments and Analysis

This section presents experimental results and analysis aimed at validating the rank reduction conditions developed in Section 3 and evaluating the performance of the adaptive lattice reduction algorithms proposed in Section 4. Rather than pursuing new performance records, our experiments are designed to assess whether these algorithms can reliably realize the feasibility conditions required by modern Approx-SVP rank reduction analyses on cryptographic-sized lattices.

### 5.1 Experimental Setup

All experiments are implemented on a server equipped with four NVIDIA RTX 4090 GPUs, two Intel Xeon Platinum 8336C CPUs (2.0 GHz), and 1.5 TB of RAM. We consider high-dimensional $q$-ary random lattices from the TU Darmstadt Lattice Challenge [10], which serves as a standard benchmark for evaluating the hardness of SIS instances. For a lattice of dimension $n$, the challenge consists of finding a non-zero vector of norm smaller than the modulus $q$, corresponding to solving SIS in a $q$-ary random lattice.

Following common practice for high-dimensional $q$-ary lattices, we apply the column deletion technique introduced by Micciancio and Regev [19] and refined in [27] to reduce the lattice dimension while preserving the existence of short vectors. All subsequent reduction procedures are applied to these column-deleted sublattices.

### 5.2 Experimental Validation of Approx-SVP Rank Reduction Conditions

This subsection experimentally validates the feasibility thresholds predicted by the rank reduction conditions introduced in Section 3. In particular, we evaluate both the practical rank reduction condition (Condition 1) and the compact Approx-SVP rank reduction condition (Condition 2) by comparing their predicted thresholds with observed attack feasibility in practice.

*Reading guide for rank reduction experiments.* In all tables of this subsection, "MSD" denotes the maximal sieving dimension observed under the rank reduction condition stated in the table caption, combined with the dimension-for-free

technique. "Theory" refers to the feasibility threshold predicted by the corresponding condition, while "Experiments" reports the largest sieving dimension at which the attack succeeded under the same experimental setting. Unless explicitly stated otherwise, reported deviations are measured relative to the theoretical threshold.

Table 2: Experimental validation of the feasibility threshold predicted by Condition 1.

| Dim | $n$ | $\gamma$ | $\delta$ | $\beta$ | **Original rank** | **Our rank** | **MSD** | |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | **Theory** | Experiments |
| 850 | 230 | 1.668906 | 1.007991 | 133 | 130 | 164 | **132** | 133 |
| 925 | 295 | 2.777266 | 1.008458 | 120 | 121 | 153 | **123** | 123 |
| 1025 | 330 | 2.891528 | 1.007766 | 132 | 135 | 170 | **137** | 137 |

| Dim | $n$ | Target norm | Experimental norm | Wall time | Previous wall time |
|---|---|---|---|---|---|
| 850 | 230 | 102.40 | 100.00 | $1.40d$ | - |
| 925 | 295 | 137.56 | 133.36 | $2.07h$ | - |
| 1025 | 330 | 164.99 | 163.43 | $1.22d$ | - |

Table 2 reports the experimental validation of the practical Approx-SVP rank reduction condition (Condition 1) on high-dimensional $q$-ary lattice challenges of dimensions 850, 925, and 1025. These instances are representative of cryptographic-scale lattices and are derived from the Lattice Challenge benchmark.

Across all tested dimensions, the experimentally observed maximal sieving dimensions closely match the theoretical feasibility thresholds predicted by the practical condition. The relative deviation between theory and experiments remains below 1%, confirming that the geometric sieving model underlying Condition 1 accurately captures the feasibility boundary in practice.

The lower sub-table further reports the corresponding target norms, achieved solution norms, and wall-clock running times. While the practical condition successfully predicts feasible sieving dimensions under Condition 1, the associated running times remain substantial at higher dimensions, reflecting the conservative nature of single-vector geometric bounds when applied to large $q$-ary lattices.

Notably, the original Approx-SVP rank reduction condition of [17] fails to yield feasible attacks on these instances within reasonable time, highlighting the necessity of refined feasibility modeling even at the level of practical geometric analysis.

Table 3 presents the experimental validation of the compact Approx-SVP rank reduction condition (Condition 2) on the same set of lattice challenges. In contrast to the practical condition, the compact condition incorporates basis-

Table 3: Experimental validation of the feasibility threshold predicted by Condition 2.

| Dim | $n$ | $\gamma$ | $\delta$ | $\beta$ | **Original rank** | **Our rank** | **MSD** | |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | **Theory** | Experiments |
| 850 | 230 | 1.668906 | 1.007991 | 133 | 130 | 149 | **119** | 118 |
| 925 | 295 | 2.777266 | 1.008458 | 120 | 121 | 134 | **107** | 107 |
| 1025 | 330 | 2.891528 | 1.007766 | 132 | 135 | 162 | **130** | 130 |

| Dim | $n$ | Target norm | Experimental norm | **Wall time** | Previous wall time |
|---|---|---|---|---|---|
| 850 | 230 | 102.40 | 102.20 | **31.2m** | $1.40d$ |
| 925 | 295 | 137.56 | 136.20 | **2m** | $2.07h$ |
| 1025 | 330 | 164.99 | 163.01 | **1.89h** | $1.22d$ |

quality-dependent sieving guarantees, leading to strictly smaller predicted feasible ranks.

The experimental results confirm that both the maximal sieving dimensions and the achieved solution norms closely follow the theoretical predictions, with relative discrepancies below 2%. This agreement indicates that the compact condition provides a conservative yet accurate feasibility threshold for Approx-SVP attacks in high dimensions.

More importantly, the reduced sieving dimensions translate directly into substantial performance gains. Compared to the practical rank reduction condition, the compact condition achieves speedups of approximately 64.6×, 62.1×, and 15.5× for dimensions 850, 925, and 1025, respectively. These improvements enabled new record solutions for dimensions 850 and 925, while the 1025-dimensional instance remains constrained by the available experimental budget.

The impact of accurate rank estimation is particularly evident for the 850-dimensional instance. Using an overestimated rank would require progressive sieving across a wide range of dimensions, resulting in prohibitive runtimes. In contrast, the compact condition pinpoints a single feasible sieving dimension, demonstrating that precise feasibility thresholds are essential for predictable and efficient lattice reduction.

### 5.3 Experiments for Lattice Basis Reduction

This subsection validates the reduction behavior predicted by the adaptive strategies introduced in Section 4. In particular, we evaluate whether APBKZ can reliably avoid early-dimension ineffectiveness and unsaturated reduction, and whether HeadAPBKZ preserves reduction feasibility once a compact rank reduction condition (Condition 2) is satisfied.

*On the use of fixed blocksizes in the experiments.* In the 1050-dimensional $q$-ary instance considered here, where $q = 170$, the leading basis vectors prior to

solution recovery are $q$-vectors, that is, vectors whose Euclidean norm equals $q$. In this regime, the corresponding prefix sublattices do not exhibit Gaussian Heuristic behavior, and adaptive blocksize selection based on $\mathrm{gh}(\mathcal{L}_{[i:i+\beta-1]})$ is therefore not reliable. Consequently, before a sufficiently short vector is recovered, APBKZ operates with a pre-specified blocksize aligned with the target comparison algorithm, namely PnjBKZ, to ensure a fair and meaningful comparison in the $q$-vector-dominated phase.

Once the reduction progresses beyond this initial prefix, the remaining basis vectors admit nontrivial improvement and the local blocks begin to exhibit Gaussian Heuristic behavior. From this point on, APBKZ applies its adaptive blocksize and dimension-for-free selection rules to all subsequent blocks. This two-phase behavior explains the fixed blocksizes observed at the beginning of the experiments and the fully adaptive behavior thereafter.

**Validation of Adaptive Reduction Behavior** We first evaluate APBKZ on a 340-dimensional sublattice derived from the 1050-dimensional $q$-ary instance of the TU Darmstadt Lattice Challenge. Table 4 compares classical PnjBKZ with APBKZ under comparable parameter settings. All comparisons are conducted using identical input bases, and differences in outcomes are solely attributable to the reduction strategy.

Table 4: Validation of adaptive reduction behavior on a 340-dimensional $q$-ary sublattice

| Algorithm | $(\beta, J)$ | Initial norm | Output norm | Saturation Ratio | Wall time |
|---|---|---|---|---|---|
| PnjBKZ | (118,2) | 275.33 | 267.54 | 0.1% | 6.37d |
| PnjBKZ | (121,2) | 267.54 | 262.14 | 0% | 8.43d |
| APBKZ | (118,1) | 275.33 | 254.92 | **46.2%** | 3.77h |
| APBKZ | (121,1) | 267.54 | 247.94 | 22.1% | 7.02h |

The results confirm that APBKZ consistently avoids early-dimension ineffectiveness. While PnjBKZ can still output valid short vectors as the blocksize increases, it does so under significantly reduced saturation and lower reduction efficiency, leading to output norms that are consistently longer than those achieved by APBKZ under comparable parameter settings. In contrast, APBKZ maintains stable saturation ratios by explicitly reducing every basis vector during each tour, as ensured by the adaptive blocksize and dimension-for-free selection rules described in Section 4.1.

Moreover, the consistently nonzero saturation ratios observed for APBKZ validate the effectiveness of the extended reduction scope $i \leq n - f_{\max} - 50$, which guarantees that all early blocks are processed in a regime where reduction remains effective.

**Validation of HeadAPBKZ Mode Switching** We next evaluate the head-focused execution mode HeadAPBKZ introduced in Section 4.2. In these experiments, HeadAPBKZ is activated only after the compact Approx-SVP rank reduction condition (Condition 2) is satisfied for a prefix of dimension $k$, at which point further reduction of the lattice tail is expected to have negligible impact on feasibility. This behavior provides an algorithmic justification for treating Condition 2 as a stopping criterion rather than merely a theoretical bound.

Table 5: Validation of head-focused reduction after satisfying the compact rank reduction condition

| Algorithm | $(\beta, J)$ | Initial norm | Output norm | Saturation Ratio | Wall time |
|---|---|---|---|---|---|
| APBKZ | (118,1) | 275.33 | 254.92 | **46.2%** | 3.77h |
| APBKZ | (121,1) | 267.54 | 247.94 | 22.1% | 7.02h |
| HeadAPBKZ | (118,1) | 275.33 | 255.34 | **41.8%** | 3.11h |
| HeadAPBKZ | (121,1) | 267.54 | 250.57 | 23.5% | 4.64h |

Table 5 reports reduction outcomes when restricting the reduction scope to the prefix sublattice $\mathcal{L}_{[1:k]}$ with $k = 161$. All HeadAPBKZ runs inherit the adaptive parameter selection rules of APBKZ and differ only in the reduction scope.

The results show that once the compact rank reduction condition is met, restricting reduction to the prefix sublattice preserves reduction quality while substantially reducing computational cost. In particular, HeadAPBKZ achieves reduction quality comparable to full APBKZ, as reflected by similar output norms and saturation behavior, while avoiding unnecessary tail reduction.

These observations validate the design of HeadAPBKZ described in Section 4.2. The transition from APBKZ to HeadAPBKZ is triggered by the satisfaction of the compact rank reduction condition, which indicates that the attack has entered a feasible regime. Beyond this point, further reduction of the tail basis vectors can still yield shorter output norms; however, the marginal improvement per unit of computational cost becomes significantly smaller. HeadAPBKZ therefore restricts reduction to the head region in order to achieve a more favorable cost–quality trade-off, leading to faster norm decrease under comparable computational budgets.

## 6 Refined Security Estimation of MSIS

In this section, we revisit the concrete security estimation of the Module Short Integer Solution (MSIS) problem underlying Dilithium. Our goal is not to propose new attacks, but to refine the security evaluation by instantiating the compact Approx-SVP rank reduction condition (Condition 2) and lattice reduction behavior developed in Sections 3 and 4 within a concrete estimation framework.

### 6.1 Baseline MSIS Security Estimation

Bai et al. [4] estimated the hardness of MSIS within the Core-SVP model by analyzing BKZ reduction in which the local SVP oracle is instantiated by lattice sieving on projected sublattices. Their estimation serves as a conservative baseline and relies on a simplified modeling of reduction feasibility and oracle success.

In this model, the SVP oracle invoked within BKZ-$\beta$ is assumed to return vectors whose lengths are close to the Gaussian Heuristic prediction. Although sieving algorithms are assumed to output an exponential number of short vectors, this multiplicity is incorporated only at the level of expectation. Concretely, if sieving outputs $N = 2^{0.2075\beta}$ candidate vectors and each candidate satisfies the prescribed $\ell_\infty$ constraint with probability $p_{\text{single}}$, the success of the oracle is evaluated via the expected count $N \cdot p_{\text{single}}$, and the attack cost is obtained by scaling the SVP running time by the inverse of this expected value. Once the target blocksize is reached, oracle success is therefore treated as effectively deterministic.

To instantiate this model, Bai et al. employ the **BKZShapeSim** procedure (Algorithm 2) to simulate the Gram–Schmidt profile $(\ell_1, \ldots, \ell_w)$ of a BKZ-$\beta$ reduced basis under the GSA. Following their convention, the index $i$ is defined as the smallest index such that $\ell_i < \log_2 q$, and $j$ as the largest index such that $\ell_j > 0$, thereby partitioning the reduced basis into a head region of length $i-1$ and a middle region of dimension $j-i+1$. The success probability is evaluated on the projected sublattice $\mathcal{L}_{\pi[i:j]}$.

The sieving-based SVP oracle is assumed to output an expected number of $2^{0.2075\beta}$ short vectors of Euclidean norm approximately $2^{\ell_i}$. Each such vector follows a mixed distribution: the head coordinates are modeled as uniformly random modulo $q$, while the middle coordinates are sampled independently from a Gaussian distribution with standard deviation $\sigma \approx 2^{\ell_i}/\sqrt{j-i+1}$. The resulting per-vector success probability is then combined with the expected output count to estimate the overall attack cost, as summarized in Algorithm 3. To mitigate the negligible success probability contributed by initial $q$-vectors (Zone I), re-randomization of the input basis is applied, at the cost of larger initial basis vectors. We refer to [4] for further details.

### 6.2 Refined Security Estimation Framework

We now present a refined MSIS security estimation that instantiates the compact Approx-SVP rank reduction condition (Condition 2) developed in Section 3 and explicitly accounts for the feasibility of achieving the assumed basis quality in high dimensions. In this framework, rank reduction is treated as a feasibility condition rather than as an attack primitive: once the compact condition is satisfied for a prefix of length $k$, further reduction of the remaining basis vectors has negligible impact on feasibility.

**Algorithm 2 BKZShapeSim**$(q, h, w, \beta)$

**Input:** $q, h, w, \beta$.
**Output:** $i, j, \{\ell_t\}_{t \le w}$.

$\log_2(\mathrm{V}) \leftarrow h \cdot \log_2(q)$
$slope \leftarrow -2 \cdot \log_2(\mathsf{RHF}(\beta))$
$V_{bkz} \leftarrow 0$
**for** $t = 1$ to $w$ **do**
    $\ell_t \leftarrow -t \cdot slope$
    $V_{bkz} \leftarrow V_{bkz} + \ell_t$
    **if** $V_{bkz} > \log_2(\mathrm{V})$ **then**
        $S_{len} \leftarrow t$
        **break**
    **end if**
**end for**
**for** $t = S_{len} + 1$ to $w$ **do**
    $\ell_t \leftarrow 0$
**end for**
$V_{diff} \leftarrow V_{bkz} - \log_2(\mathrm{V})$
**for** $t = 1$ to $S_{len}$ **do**
    $\ell_t \leftarrow \ell_t - \frac{V_{diff}}{S_{len}}$
**end for**
$i \leftarrow \min\{t \in [1, w] : \ell_t < \log_2(q)\}$ ▷ head: $1, \ldots, i-1$
$j \leftarrow \max\{t \in [1, w] : \ell_t > 0\}$ ▷ middle: $i, \ldots, j$
**return** $i, j, \{\ell_t\}_{t \le w}$

**Algorithm 3 MSISCost** $(i, j, \zeta', \ell_i, q, \beta)$

**Input:** $i, j, \zeta', \ell_i, q, \beta$.
**Output:** $B, p$.

$\sigma \leftarrow 2^{\ell_i}/\sqrt{j - i + 1}$
$p_{mid} \leftarrow \Pr\left[x \leftarrow N\left(0, \sigma^2\right) : -\zeta' \le x \le \zeta'\right]$
$p_{head} \leftarrow \frac{2\zeta'}{q}$
$p_{single} \leftarrow {p_{mid}}^{j-i+1} \cdot {p_{head}}^{i-1}$
$N \leftarrow 2^{0.2075\beta}$
$p\left(\|s\|_\infty < \zeta'\right) \leftarrow N \cdot p_{single}$
$B \leftarrow \frac{1}{p(\|s\|_\infty < \zeta')} \cdot 2^{0.292\beta}$
**return** $B, p\left(\|s\|_\infty < \zeta'\right)$

Let $\mathsf{RHF}(\beta)$ denote the root Hermite factor of a BKZ-$\beta$ reduced basis. For $\beta \geq 50$, we use the standard approximation

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

Solving $\mathrm{MSIS}_{k,\ell,\zeta'}$ reduces to Approx-SVP on the associated $q$-ary lattice induced by the public MSIS matrix $\mathbf{A}$,

$$\mathcal{L}' = \mathcal{L}\left(\begin{bmatrix} q\mathbf{I} & M(\mathbf{A}) \\ \mathbf{0} & -\mathbf{I} \end{bmatrix}\right),$$

with parameters $h = nk$ and $w = nk + n\ell$, where $M(\mathbf{A})$ denotes the standard embedding of $\mathbf{A} \in R_q^{k\times\ell}$ (with $R_q = \mathbb{Z}_q[X]/(X^n+1)$) into an $nk \times n\ell$ integer matrix [1]. Under the Gaussian Heuristic, the expected first minimum satisfies

$$\mathrm{gh}(\mathcal{L}') \approx \sqrt{\frac{w}{2\pi e}} \cdot q^{h/w}.$$

These quantities characterize the shortest-vector threshold that is realistically accessible after BKZ-$\beta$ preprocessing.

*Refined feasibility threshold.* Instead of assuming that vectors of length $\mathrm{gh}(\mathcal{L}')$ can be reliably found, we adopt the practical threshold $(8\ln(10))^{1/\beta} \cdot \mathrm{gh}(\mathcal{L}')$, which captures the length of a single sufficiently short vector required to enable dimension-for-free techniques. Algorithm 4 instantiates the compact rank reduction condition (Condition 2) and outputs the minimal feasible sieving dimension *rank* consistent with $(q, h, w)$ and $\mathsf{RHF}(\beta)$.

**Algorithm 4 PumpRank**$(q, h, w, \ell_i, \beta, \mathsf{RHF}(\beta))$

**Input:** $q, h, w, \ell_i, \beta, \mathsf{RHF}(\beta)$.
**Output:** *rank*.

$\delta \leftarrow \mathsf{RHF}(\beta)$
$\mathrm{gh}(\mathcal{L}') \leftarrow \sqrt{w/(2\pi e)} \cdot q^{h/w}$
$\gamma \leftarrow 2^{\ell_i}/\mathrm{gh}(\mathcal{L}')$
**for** $k = 50$ to $w$ **do**
  **if** $\gamma \leq (8\ln(10))^{1/\beta} \cdot \sqrt{k/w} \cdot \delta^{(w-k+1)}$ **then**
    **return** $k$
  **end if**
**end for**

[1] This embedding is standard in MSIS security analyses and is also used in the Dilithium security evaluation [4].

*Probability-aware modeling.* We further refine the modeling of oracle success by explicitly accounting for probabilistic amplification. If sieving outputs $N$ candidate vectors and each candidate independently satisfies the $\ell_\infty$ constraint with probability $p_{\text{single}}$, then the success probability of a sieving call is modeled as $1-(1-p_{\text{single}})^N$. This refinement does not assume new algorithmic capabilities, but more accurately reflects the behavior of modern sieving-based SVP oracles.

*Two-step optimization.* Let $\mathrm{E}_\beta$ denote the event that BKZ-$\beta$ preprocessing already yields a valid solution, and $\mathrm{E}_{rank}$ the event that a final sieving call in dimension $rank$ succeeds on a BKZ-$\beta$ reduced basis. The overall success probability is modeled as

$$\Pr[\mathrm{W}] = \Pr[\mathrm{E}_\beta] + (1-\Pr[\mathrm{E}_\beta])\Pr[\mathrm{E}_{rank}],$$

and the expected attack cost is minimized over $(\beta, rank)$:

$$\min_{\beta,rank} \frac{T_{\mathrm{BKZ}}(\beta)+T_{\mathrm{sieve}}(rank)}{\Pr[\mathrm{W}]}.$$

Following [4], we ignore constant factors and incorporate the dimension-for-free technique via

$$T_{\mathrm{BKZ}}(\beta)=2^{0.292(\beta-\mathrm{D4f}(\beta))}, \qquad T_{\mathrm{sieve}}(rank)=2^{0.292(rank-\mathrm{D4f}(rank))}.$$

**Algorithm 5 RefinedMSISCost** $(i, j, \zeta', \ell_i, q, \beta)$

**Input:** $i, j, \zeta', \ell_i, q, \beta$.
**Output:** $B, p$.
1: $\sigma \leftarrow 2^{\ell_i}/\sqrt{j-i+1}$
2: $p_{mid} \leftarrow \Pr\left[x \leftarrow N\left(0,\sigma^2\right) : -\zeta' \le x \le \zeta'\right]$
3: $p_{head} \leftarrow \frac{2\zeta'}{q}$
4: $p_{single} \leftarrow {p_{mid}}^{j-i+1} \cdot {p_{head}}^{i-1}$
5: $N \leftarrow 2^{0.2075\beta}$
6: $p\left(\|s\|_\infty < \zeta'\right) \leftarrow 1-(1-p_{single})^N$
7: $B \leftarrow 2^{0.292\beta}$
8: **return** $B, p\left(\|s\|_\infty < \zeta'\right)$

The **BKZShapeSim** procedure is used to simulate basis shapes under the GSA; its output indices $(i, j)$ define the head and middle regions used consistently in Algorithm 5. The role of APBKZ in this estimation pipeline is to ensure that these assumed basis profiles can be stably realized in high dimensions.

*Resulting estimates.* Table 6 reports the optimized parameter choices and resulting estimates for the Dilithium parameter sets. Compared to the Core-SVP baseline, the refined model yields a slightly higher estimated security margin, reflecting the tighter feasibility constraints imposed by the compact rank reduction

condition (Condition 2). When accounting for dimension-for-free techniques, the estimated concrete security margin is reduced by 9.50–16.63 bits relative to the Core-SVP model, highlighting the impact of refined feasibility and probability modeling rather than the existence of fundamentally stronger attacks.

**Algorithm 6** Two-Step Rank Reduction MSIS Security Estimation Model

**Input:** $q, h, w, \zeta'$.
**Output:** $\beta_{op}, rank_{op}, T_{op}$.
$T_{op} \leftarrow +\infty$
**for** $\beta = 50$ to $w$ **do**
  $(i, j, \{\ell_t\}_{t \le w}) \leftarrow$ **BKZShapeSim**$(q, h, w, \beta)$
  $rank \leftarrow$ **PumpRank**$(q, h, w, \ell_i, \beta, \mathsf{RHF}(\beta))$
  $(T_{\mathrm{BKZ}}(\beta), p_\beta) \leftarrow$ **RefinedMSISCost**$(i, j, \zeta', \ell_i, q, \beta)$
  $(T_{\mathrm{sieve}}(rank), p_{rank}) \leftarrow$ **RefinedMSISCost**$(i, j, \zeta', \ell_i, q, rank)$
  $p_W \leftarrow p_\beta + (1 - p_\beta) \cdot p_{rank}$
  $T \leftarrow (T_{\mathrm{BKZ}}(\beta) + T_{\mathrm{sieve}}(rank)) / p_W$
  **if** $T < T_{op}$ **then**
    $\beta_{op} \leftarrow \beta$; $rank_{op} \leftarrow rank$; $T_{op} \leftarrow T$
  **end if**
**end for**
**return** $\beta_{op}, rank_{op}, T_{op}$

Table 6: MSIS security estimation for Dilithium (our refined model vs. Core-SVP [4])

| | Dilithium-II | Dilithium-III | Dilithium-V |
|---|---|---|---|
| NIST Level | 2 | 3 | 5 |
| Lattice Dimension $w$ | 2048 | 2816 | 3840 |
| BKZ $\beta$ | 417 | 602 | 868 |
| Core-SVP (bits) | 121.96 | 176.37 | 253.87 |
| $\beta_{op}$ | 418 | 604 | 870 |
| $rank_{op}$ | 420 | 606 | 873 |
| Our Estimation (bits) | 123.29 | 177.69 | 255.38 |
| With D4f (bits) | 112.46 | 163.65 | 237.24 |
| Security Change (bits) | +1.33 | +1.32 | +1.51 |
| With D4f Change (bits) | -9.50 | -12.72 | -16.63 |

## 7 Conclusion

In this work, we revisited Approx-SVP rank reduction from a feasibility-driven perspective and identified fundamental limitations in existing conditions when

applied to cryptographic-sized lattices. By focusing on the existence of a single sufficiently short vector, we derived refined and compact rank reduction conditions that better reflect the requirements of modern D4f-enabled attacks.

We further proposed APBKZ and HeadAPBKZ, two robust lattice reduction variants whose purpose is to ensure that the basis profiles assumed in theoretical analyses can be reliably realized in high dimensions. Experimental results confirm that these strategies significantly improve stability and efficiency without relying on simulator-driven parameter selection.

By integrating refined rank reduction theory with robust reduction behavior, we developed a tighter concrete security estimation framework for MSIS and applied it to Dilithium. Our results suggest that conservative Core-SVP estimates may overstate security margins. Specifically, accounting for compact rank reduction behavior leads to a decrease of 9.50–16.63 bits in the estimated security level when D4f techniques are considered.

**Future work.** An interesting direction for future research is to develop a rigorous theoretical understanding of practical heuristics commonly used in lattice reduction algorithms, such as re-randomization. While such techniques are observed to improve stability in practice, providing formal guarantees remains an open problem and is beyond the scope of this work.

## References

1. Ajtai, M.: Generating hard instances of the short basis problem. In: Automata, Languages and Programming: 26th International Colloquium, ICALP'99 Prague, Czech Republic, July 11–15, 1999 Proceedings. pp. 1–9. Springer (2002)
2. Albrecht, M.R., Ducas, L., Herold, G., Kirshanova, E., Postlethwaite, E.W., Stevens, M.: The general sieve kernel and new records in lattice reduction. In: Annual International Conference on the Theory and Applications of Cryptographic Techniques. pp. 717–746. Springer (2019)
3. Aono, Y., Wang, Y., Hayashi, T., Takagi, T.: Improved progressive bkz algorithms and their precise cost estimation by sharp simulator. In: Annual International Conference on the Theory and Applications of Cryptographic Techniques. pp. 789–819. Springer (2016)
4. Bai, S., Ducas, L., Kiltz, E., Lepoint, T., Lyubashevsky, V., Schwabe, P., Seiler, G., Stehlé, D.: Crystals-dilithium – algorithm specifications and supporting documentation (version 3.1). Submission to the NIST post-quantum project (update from February 2021) (2021)
5. Bai, S., Stehlé, D., Wen, W.: Measuring, simulating and exploiting the head concavity phenomenon in bkz. In: International Conference on the Theory and Application of Cryptology and Information Security. pp. 369–404. Springer (2018)
6. Becker, A., Ducas, L., Gama, N., Laarhoven, T.: New directions in nearest neighbor searching with applications to lattice sieving. In: Proceedings of the twenty-seventh annual ACM-SIAM symposium on Discrete algorithms. pp. 10–24. SIAM (2016)
7. Becker, A., Gama, N., Joux, A.: A sieve algorithm based on overlattices. LMS Journal of Computation and Mathematics **17**(A), 49–70 (2014)
8. Becker, A., Gama, N., Joux, A.: Speeding-up lattice sieving without increasing the memory, using sub-quadratic nearest neighbor search. Cryptology ePrint Archive (2015)

9. Becker, A., Laarhoven, T.: Efficient (ideal) lattice sieving using cross-polytope lsh. In: International Conference on Cryptology in Africa. pp. 3–23. Springer (2016)
10. Buchmann, J., Lindner, R., Rückert, M.: Explicit hard instances of the shortest vector problem. In: International Workshop on Post-Quantum Cryptography. pp. 79–94. Springer (2008)
11. Chen, Y.: Réduction de réseau et sécurité concrete du chiffrement completement homomorphe. Ph.D. thesis, Paris 7 (2013)
12. Chen, Y., Nguyen, P.Q.: Bkz 2.0: Better lattice security estimates. In: International Conference on the Theory and Application of Cryptology and Information Security. pp. 1–20. Springer (2011)
13. Ducas, L.: Shortest vector from lattice sieving: a few dimensions for free. In: Annual International Conference on the Theory and Applications of Cryptographic Techniques. pp. 125–145. Springer (2018)
14. Ducas, L., Stevens, M., van Woerden, W.: Advanced lattice sieving on gpus, with tensor cores. In: Annual International Conference on the Theory and Applications of Cryptographic Techniques. pp. 249–279. Springer (2021)
15. Gama, N., Nguyen, P.Q.: Predicting lattice reduction. In: Annual International Conference on the Theory and Applications of Cryptographic Techniques. pp. 31–51. Springer (2008)
16. Laarhoven, T.: Sieving for shortest vectors in lattices using angular locality-sensitive hashing. In: Annual Cryptology Conference. pp. 3–22. Springer (2015)
17. Laarhoven, T., Mariano, A.: Progressive lattice sieving. In: International Conference on Post-Quantum Cryptography. pp. 292–311. Springer (2018)
18. Laarhoven, T., de Weger, B.: Faster sieving for shortest lattice vectors using spherical locality-sensitive hashing. In: International Conference on Cryptology and Information Security in Latin America. pp. 101–118. Springer (2015)
19. Micciancio, D., Regev, O.: Lattice-based cryptography, post-quantum cryptography (daniel j. bernstein, johannes buchmann, and erik dahmen, eds.) (2009)
20. Micciancio, D., Voulgaris, P.: A deterministic single exponential time algorithm for most lattice problems based on voronoi cell computations. In: Proceedings of the Forty-second ACM Symposium on Theory of Computing. pp. 351–358 (2010)
21. Nguyen, P.Q.: Hermite's constant and lattice algorithms. In: The LLL Algorithm: Survey and Applications, pp. 19–69. Springer (2009)
22. Nguyen, P.Q., Vidick, T.: Sieve algorithms for the shortest vector problem are practical. Journal of Mathematical Cryptology **2**(2), 181–207 (2008)
23. Postlethwaite, E.W., Virdia, F.: On the success probability of solving unique svp via bkz. In: IACR International Conference on Public-Key Cryptography. pp. 68–98. Springer (2021)
24. Regev, O.: On lattices, learning with errors, random linear codes, and cryptography. Journal of the ACM (JACM) **56**(6), 1–40 (2009)
25. Schnorr, C.P.: Lattice reduction by random sampling and birthday methods. In: Annual Symposium on Theoretical Aspects of Computer Science. pp. 145–156. Springer (2003)
26. Schnorr, C.P., Euchner, M.: Lattice basis reduction: Improved practical algorithms and solving subset sum problems. Mathematical programming **66**(1), 181–199 (1994)
27. Sun, M., Wang, S., Chen, H., Qu, L.: A new random sampling method and its application in improving progressive bkz algorithm. Journal of Systems Science and Complexity **37**(5), 2262–2292 (2024)

28. Wang, X., Liu, M., Tian, C., Bi, J.: Improved nguyen-vidick heuristic sieve algorithm for shortest vector problem. In: Proceedings of the 6th ACM Symposium on Information, Computer and Communications Security. pp. 1–9 (2011)
29. Zhang, F., Pan, Y., Hu, G.: A three-level sieve algorithm for the shortest vector problem. In: International Conference on Selected Areas in Cryptography. pp. 29–47. Springer (2013)
30. Zhao, Z., Ding, J.: Practical improvements on bkz algorithm. In: International Symposium on Cyber Security, Cryptology, and Machine Learning. pp. 273–284. Springer (2023)
31. Zhao, Z., Ding, J., Yang, B.Y.: Sieving with streaming memory access. IACR Transactions on Cryptographic Hardware and Embedded Systems **2025**(2), 362–384 (2025)