The ISC Int'l Journal of Information Security
ISeCure

July 2021, Volume 13, Number 2 (pp. 177–208)
http://www.isecure-journal.org

Better Sampling Method of Enumeration Solution for BKZ-Simulation

Gholam Reza Moghissi [1,*] and Ali Payandeh [1]
[1] *Department of ICT, Malek-Ashtar University of Technology, Tehran, Iran.*

ARTICLE INFO.

Article history:
Received: April 8, 2020
Revised: May 8, 2020
Accepted: May 26, 2021
Published Online: June 26, 2021

Keywords:
BKZ Simulation, Coefficient Vector, GNR Enumeration, Optimal Bounding Function, Sampling Method, Solution Norm

Type: Research Article

doi: 10.22042/isecure.2021.225886.531

dor: 20.1001.1.20082045.2021.13.2.8.3

Abstract

The exact manner of BKZ algorithm for higher block sizes cannot be studied by practical running, so simulation of BKZ can be used to predict the total cost and output quality of BKZ algorithm. Sampling method of enumeration solution vector v is one of the main components of designing BKZ-simulation and can be divided into two phases: sampling norm of solution vector v and sampling corresponding coefficient vectors. This paper introduces a simple and efficient idea for sampling the norm of enumeration solution v for any success probability of enumeration bounding functions, while to the best of our knowledge, no such sampling method for norm of enumeration solution is proposed in former studies. Next, this paper analyzes the structure and probability distribution of coefficient vectors (corresponding with enumeration solution v), and consequently introduces the sampling methods for these coefficient vectors which are verified by our test results, while no such a deep analysis for sampling coefficient vectors is considered in design of former BKZ-simulations. Moreover, this paper proposes an approximation for cost of enumerations pruned by optimal bounding functions.

1 Introduction

Lattice reduction is the determinative phase of most lattice security attacks, and BKZ reduction is currently considered as a main practical one [1–4]. In fact, for selecting security parameters in lattice cryptographic primitives, the total cost and output quality of BKZ algorithm should be determined in high block sizes. For predicting the manner of BKZ in higher block sizes, practical running is not the way, therefore BKZ-simulators are introduced. At first, an efficient simulation algorithm is introduced by Chen and Nguyen [2] based on Gaussian heuristic. The simulation by Shi Bai *et al.* [3] is focused on head concavity phenomenon in BKZ and Gaussian heuristic. Also, Aono *et al.* [4] introduces a sharp simulator under geometric series assumption (GSA). There are many studies which show the main role of BKZ algorithm and BKZ-simulation in determining the bit-security of lattice-based cryptographic primitives (see [2, 5–10]).

Two main outputs which are expected to be returned from a BKZ-simulation over an input basis (often in the form of GSO norms) are the output basis quality (often in form of GSO norms) and total cost of BKZ. Total cost and output quality of basis depend to each other in many cases, (e.g., the quality of a lattice block affects the enumeration cost over that block, in other side, the cost of enumerations in optimal progressive-BKZ affects the quality of lattice blocks). Designing a BKZ-simulation with

* Corresponding author.
Email addresses: fumoghissi@chmail.ir, payandeh@mut.ac.ir
ISSN: 2008-2045

GNR-pruned enumeration needs to some necessary building-blocks which includes enumeration radius, generation of bounding function, estimation of success probability, LLL simulation, estimation of GNR enumeration cost, sampling method for enumeration solution, simulation of updating GSO.

One of main components in design of BKZ-simulation with GNR-pruned enumeration as SVP solver are sampling the enumeration solution. Enumeration solution v is often represented in former BKZ-simulations by estimating the expected value of the norm of this solution as $\|v\|$. This paper tries to introduce an approximate sampling method for enumeration solution v which samples both norm and coefficient vectors of enumeration solution. In fact, the GNR enumeration function in BKZ algorithm over a lattice block $\mathcal{L}_{[b_j,\ldots,b_k]}$ returns the coefficient vector y, which can be used to compute the solution vector v by linear combination of this lattice block vectors as $v = \sum_{l=j}^{k} y_l b_l$. To the best of our knowledge, no precise and explicit analysis of sampling the coefficient vector y is considered in former studies on BKZ-simulations. Also to the best of our knowledge, no sampling method for norm of GNR-pruned enumeration solution with any success probability is introduced, however paper [3] introduces a sampling method (see line 14 of Algorithm 4 from paper [3]) by using the probability distribution of solution norm which is stated in Chen's thesis [11] just for full-enumeration (see Theorem 1 in [3]), not for any GNR-pruning with any success probability. Also, paper [4] uses an efficient way to estimate the expected value of solution norm, instead of sampling this norm. Moreover, paper [2] uses only the approximation by Gaussian heuristic expectation of solution norm. In other side, not only this paper introduces a sampling method for solution norm of GNR-pruned enumeration with any success probability, but also it introduces a statistical sampling method for coefficient vector y (and other coefficient vectors). Also by using our analysis on enumerations solution norm and coefficient vectors, this paper proposes an approximation for cost of enumerations pruned by optimal bounding functions.

The remainder of this paper is organized as follows. Section 2 is dedicated to essential background for understanding our contributions in this paper. Our contributions in design of sampling method of GNR-enumeration solution would be introduced in Section 3 (Section 3.1 introduces our sampling method of norm of enumeration solution and Section 3.2 introduces our sampling method of coefficient vectors of enumeration solution). Also our approximation of cost for GNR-enumeration by optimal bounding function is introduced at the end of Section 3. Our test results for verifying our proposed sampling methods are introduced in Section 4. Finally, in Section 5, the conclusions and further studies for this work are expressed.

2 Background

In this section, a sufficient background on BKZ algorithm is introduced to make this work easy to be understood. Also, some related preliminary discussions, propositions, definitions and notations are proposed which help to simply focus on our main contributions in the next sections. To have most traceability among relations, propositions and algorithms, the similar notations would be used in this paper (such as the notations of n and m for rank and dimension of lattice in whole the paper).

2.1 Basic Definitions, Notations and Concepts

Here some basic lattice concepts which are needed in this paper are defined.

Lattices. A lattice is a set of points in the n-dimensional space with a periodic structure [12]. More formally, given n-linearly independent vectors $b_1, \ldots, b_n \in \mathbb{R}^m$, the lattice generated by them is a set of vectors as follows:

$$\mathcal{L}_{[b_1,\ldots,b_n]} = \sum_{i=1}^{n} x_i b_i : x_i \in \mathbb{Z} \qquad (1)$$

Note: The set of vectors of $[b_1, \ldots, b_n]$ are known as the basis of lattice, which is usually shown by the column of matrix B.

Note: Since the lattice in paper is discussed for cryptographic applications, this is assumed to $b_i \in \mathbb{Z}^m$.

Note: The rank and dimension of lattice $\mathcal{L}(B)$ in this paper are respectively n and m.

Note: The vector of $x \in \mathbb{Z}^n$ in relation (1) is named as coefficient basis vector.

Euclidean Norm. The length of a lattice vector $v = (v_1, \ldots, v_m)$ is measured by $\|v\| = \sqrt[2]{v_1^2 + \cdots + v_m^2}$.

Note: In this paper, the phrases of "norm" and "length" refer to Euclidean norm.

Fundamental Domain. For a lattice $\mathcal{L}(B)$, the fundamental domain is defined as following set:

$$\mathcal{F}(\mathcal{L}) = t_1 b_1 + t_2 b_2 + \cdots + t_n b_n : 0 \le t_i < 1 \qquad (2)$$

Volume of Lattices. The volume of a lattice $\mathcal{L}(B)$ is defined by the volume of the parallelepiped of fundamental domain $\mathcal{F}(\mathcal{L})$ which is computed as follows:

$$\mathrm{vol}\left(\mathcal{L}(B)\right) = \mathrm{vol}\left(\mathcal{F}(\mathcal{L}(B))\right) = |\det B| \qquad (3)$$

There are many hard problems in the lattice theory, where the shortest vector problem (SVP) is one of the basic ones. For a given lattice basis B, SVP solvers try to find the shortest nonzero vector in this lattice. In practice, SVP is discussed as an approximate variant, which is defined as follows:

Approximate-SVP (SVP_γ). For a lattice $\mathcal{L}$, the problem of finding a lattice vector whose length is at most some approximation factor γ times the length of the shortest nonzero vector.

Note: The norm of shortest nonzero vector in lattice $\mathcal{L}$ is shown by $\lambda_1(\mathcal{L})$ (which is first successive-minima in lattice $\mathcal{L}$).

Since in practical attacks to solve SVP_γ, the value of $\lambda_1(\mathcal{L})$ is not known, so this is common to use the concept of Hermite-SVP_r which is defined as follows:

Hermite-SVP_r ($HSVP_r$). For a lattice $\mathcal{L}(B)$, the problem of finding a lattice vector whose length is at most some approximation factor r times the length of $\text{vol}\,(\mathcal{L}(B))^{1/n}$ (i.e., $\|b_1\| \leq r|\det B|^{1/n}$).

Root-Hermite factor (δ_r). The common parameter to measure the quality of a reduced basis B is defined as follows

$$\delta_r = r^{1/n} \tag{4}$$

Another basic concept which is needed in our analysis, is the volume of a n-dimensional ball, which can be computed as follows (by using sterling approximation for high dimensional space)

$$\begin{aligned} \text{V}_n(R) &= \text{vol}\,(\text{Ball}_n(R)) = (\pi^{n/2}/\Gamma(n/2+1))R^n \\ &\approx ((2\pi e/n)^{n/2}/\sqrt{n\pi})R^n \end{aligned} \tag{5}$$

Note: In this paper, $\text{V}_l(R)$ refers to volume of a l-dimensional ball with radius R.

The gamma function $\Gamma(x)$ is defined for $x > 0$ by $\Gamma(x) = \int_0^\infty t^{x-1}e^{-t}dt$, where by using sterling approximation, the gamma function $\Gamma(n/2+1)$ is defined as $\Gamma(n/2+1) \approx \sqrt{n\pi}(\frac{n}{2e})^{n/2}$. Also, the beta function $\text{Beta}(x,y)$ which is used later in Lemma 1, is defined as follows

$$\text{Beta}(x,y) = \int_0^1 t^{x-1}(1-t)^{y-1}dt = \tfrac{\Gamma(x)\Gamma(y)}{\Gamma(x+y)} \tag{6}$$

Note: In this paper, the dimension m is assumed to be equal to rank n (i.e., $m = n$), however the main parts of this paper just work on analysis of lattice blocks with rank of β and dimension n.

One of the main heuristic in lattice theory is Gaussian heuristic, which estimates the number of points in a set S. This heuristic is used massively in our analysis and discussion. This heuristic is defined as follows

Heuristic 1 (Gaussian heuristic). "*Given a lattice $\mathcal{L}$ and a set S, the number of points in $S \bigcap \mathcal{L}$ is approximated by $vol(S)/vol(\mathcal{L})$*" [1];

By using Heuristic 1 (Gaussian heuristic), if the lattice $\mathcal{L}$ would be limited to a centred ball with radius length of $R = \lambda_1(\mathcal{L})$, then this is expected that there is at least one lattice vector in $\text{Ball}_n(R)$ with radius R, which is the shortest vector. Therefore, the value of $\lambda_1(\mathcal{L})$ can be estimated by Gaussian heuristic of lattice $\text{GH}(\mathcal{L})$ as follows (by using sterling approximation)

$$\text{GH}(\mathcal{L}) = \left(\tfrac{\text{vol}(\mathcal{L}(B))}{\text{vol}(\text{Ball}_n(1))}\right)^{1/n} \approx \sqrt{\tfrac{n}{2\pi e}}(\det B)^{1/n}. \tag{7}$$

Also, the concepts of Gram-Schmidt Orthogonalization and projected sub-lattices are the fundamental definitions in the structure of BKZ reduction, and they are used massively in our contributions.

Orthogonal projection (π_i). For a given lattice basis $B = (b_1, b_2, \ldots, b_n)$, the orthogonal projection $\pi_i(\cdots)$ is defined as follows

$$\pi_i : \mathbb{R}^m \mapsto \text{span}(b_1, \ldots, b_{i-1})^\perp \quad \text{for} \quad i \in \{1, \ldots, n\}$$

Gram-Schmidt Orthogonal basis (GSO basis). For a given lattice basis $B = (b_1, b_2, \ldots, b_n)$, the Gram-Schmidt Orthogonal basis $B^* = (b_1^*, b_2^*, \ldots, b_n^*)$ is defined as follows

$$\begin{aligned} &\pi_i(b_i) = b_i^* = b_i - \textstyle\sum_{j=1}^{i-1} \mu_{i,j} b_j^*, \\ &\text{where} \quad \mu_{i,j} = (b_i b_j^*)/\|b_j^*\|^2 \quad \text{and} \quad 1 \leq j < i \leq n \end{aligned} \tag{8}$$

Note: For an input lattice basis B, volume of the lattice can be computed by norm of GSO vectors as follows

$$\text{vol}\,(\mathcal{L}(B)) = \textstyle\prod_{i=1}^n \|b_i^*\|. \tag{9}$$

In addition to Heuristic 1 (Gaussian heuristic), other important heuristic in Lattice theory is Geometric Series Assumption (GSA), which is defined as follows

Geometric Series Assumption (GSA). Schnorr's GSA says that for a BKZ-reduced basis, the geometric series of $\|b_i^*\| = \mathfrak{r}^{i-1}\|b_1\|$ for the GSA constant $\mathfrak{r} \in [3/4, 1)$ can be assumed [4].

By using GSA assumption, q-factor can be defined as follows, which measures the quality of basis [13],

$$q \approx 1/\mathfrak{r} = \|b_i^*\|/\|b_{i+1}^*\|. \tag{10}$$

Furthermore, following approximation between root-Hermite factor and q-factor can be assumed

$$\delta_r \approx q^{(n-1)/2n} \approx \sqrt{q}. \tag{11}$$

In other side, some statistical distribution massively is used in our analysis and proofs, such as Normal distribution and Gamma distribution (different from the Gamma function $\Gamma(x)$), and exponential distribution. These distributions are defined as follows.

Normal distribution. The Normal distribution is a bell-shaped, two-parameter and continuous probability distribution which is defined as follows

$$\mathcal{N}(x;\mu,\sigma^2) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2}, \quad \text{where} \quad x \in \mathbb{R}. \tag{12}$$

The mean and variance in Normal distribution respectively are determined by μ and σ^2.

Gamma distribution. The gamma distribution is a two-parameter and continuous probability distribution which is defined as follows (for input shape-parameter of k and scale-parameter of θ)

$$\text{Gamma}(x;k,\theta) = \frac{x^{k-1}e^{-x/\theta}}{\Gamma(k)\theta^k}, \quad \text{where } x > 0. \tag{13}$$

The mean and variance in gamma distribution respectively are determined by $k\theta$ and $k\theta^2$.

Exponential distribution. The exponential distribution is a one-parameter and continuous probability distribution which is defined as follows (for input parameter of ϑ)

$$\text{Expo}(x;\vartheta) = \vartheta e^{-\vartheta x}, \quad \text{where } x > 0 \text{ and } \vartheta > 0. \tag{14}$$

The mean and variance in exponential distribution respectively are determined by $1/\vartheta$ and $1/\vartheta^2$.

Note: In this paper, the expected value and variance of random variable X respectively are shown by $\mathrm{E}[X]$ and $\mathcal{V}[X]$.

At the end, some notations which be used in this paper, are defined as follows. The random function $\text{rand}_{(x,\dots,y)}$ returns a random real number between x and y (except the numbers of x and y). In fact, the notations of $(x,\dots,y)$, $(x,\dots,y]$, $[x,\dots,y)$ and $[x,\dots,y]$ respectively represent the range of x to y except x and y, the range of x to y except x, the range of x to y except y and the full range of x to y. Also, the notation of $\lfloor x \rceil$ returns the nearest integer number to x.

2.2 LLL Reduction

The most well-known and widely used lattice reduction algorithm for lattice problems is LLL, which developed by Lenstra (Arjen Klaas), Lenstra (Hendrik Willem), and LovÃąsz in 1982 [14]. LLL reduction is a polynomial time algorithm for the approximate-SVP within an approximation factor of $\gamma = 2^{O(n)}$ (where n is the dimension of the lattice).

LLL-reduced basis. For a given basis $b_1,\dots,b_n \in \mathbb{Z}^m$, and the parameter of $\delta \in [1/4,1)$, LLL-reduced bases should satisfy the following conditions

- Size-reduction: $|\mu_{i,j}| \leq 1/2$ for $1 \leq j < i \leq n$;
- Lovasz criterion: $\|b^*_{i+1}\|^2 \geq (\delta - \mu^2_{i+1,i})\|b^*_i\|^2$ for $1 \leq i \leq n-1$;

Note: In this paper, the notation of LLL-parameter is shown by δ and notation of root-Hermite factor is shown by δ_r.

Remark 1. Based on Size-reduction and Lovasz criterion, LLL cannot replace the first GSO norm of basis (i.e., $\|b^*_1\|$) with a bigger GSO norm (in fact, LLL always decreases or does not modify the first GSO norm).

2.3 BKZ Algorithm

In 1987, BKZ (Blockwise Korkine-Zolotarev) algorithm was proposed by Schnorr as an extension of LLL algorithm. The main idea behind the design of BKZ is to replace the blocks of 2×2 (which are used in LLL) with the blocks of larger size. Increasing the block size improves the approximation factor at the expense of more running time. There are several variants of Schnorr's BKZ, but all these variants achieve nearly the same exponential approximation factor. Here, BKZ and HKZ (Hermite-Korkine-Zolotarev) is formally defined as follows

HKZ-reduced basis. For every block $\mathcal{L}_{[b_j,\dots,b_n]}$ of input lattice basis $\mathcal{L}_{[b_1,\dots,b_n]}$ where $j = 1\dots n$, the basis should be size-reduced and satisfies $\pi_j(b_j) = \lambda_1(\pi_j(\mathcal{L}_{[b_j,\dots,b_n]}))$.

BKZ_β-reduced basis. For every block $\mathcal{L}_{[b_j,\dots,b_k]}$ of input lattice basis $\mathcal{L}_{[b_1,\dots,b_n]}$ where $1 \leq j < k = \min(j+\beta-1,n)$, then this basis should be size-reduced and satisfies $\pi_j(b_j) = \lambda_1(\pi_j(\mathcal{L}_{[b_j,\dots,b_k]}))$.

Informally, BKZ algorithm starts with the LLL reduction of basis; then, it iteratively performs the following steps

- For an input parameter R (which is defined as enumeration radius), the solution vector $v = \sum_{l=j}^{k} y_l b_l$ is returned from SVP oracle (e.g., lattice enumeration) applied on projected lattice block of $\pi_j(\mathcal{L}_{[b_j,\dots,b_k]})$, when $\|\pi_j(v)\| < R$; at the next, v is inserted between the vectors of b_{j-1} and b_j. The resulted set of vectors is not a basis (because of the linear dependency between vectors), so LLL algorithm is performed on the partial set of $b_1,\dots,b_{j-1},v,b_j,\dots,b_{h=\min(k+1,n)}$.
- Otherwise, LLL algorithm is performed on the partial set of $b_1,\dots,b_{h=\min(k+1,n)}$.

The pseudo-code of Schnorr-Euchner's BKZ algorithm is introduced in Appendix A.1. The BKZ algorithm can use the lattice enumeration for solving SVP in the projected lattice blocks (however some other functions, such as sieve algorithm can be used too) [1]. The norm of first vector of a BKZ_β-reduced basis B is bounded by $\|b_1\| \leq (\beta/(\pi e))^{(n-1)/(\beta-1)}\lambda_1(\mathcal{L}(B))$.

2.4 Enumeration and Pruning

In this paper, for a lattice block of $\mathcal{L}_{[j\dots k]} = \mathcal{L}_{[b_j,b_{j+1},\dots,b_k]}$, the block size β is assumed sufficiently big. Also since these lattice blocks are assumed to be used in BKZ algorithms, the notation $\mathcal{L}_{[b_j,b_{j+1},\dots,b_k]}$ refers to the projected form of $\pi_j(b_j, b_{j+1}, \dots, b_k)$, as a lattice block from the index j to k, whose vectors are projected on the vectors of $(b_1, b_2, \dots, b_{j-1})$. Furthermore, it is clear that the enumeration cost is affected by the choice of the initial radius R [2].

Full-enumeration. For initial radius R, the full-enumeration tree enumerates all lattice points in n-dimensional ball of radius R.

The sound pruning technique, or GNR-pruning, which is introduced by Gama, Nguyen and Regev, uses the concept of cylinder-intersection in pruning the enumeration tree. The cylinder-intersection, bounding function and GNR-pruning formally are defined as follows.

Cylinder-intersection. The l-dimensional cylinder-intersection with radius $(R_1, \dots, R_l)$ is defined as following set [1],

$$\mathcal{C}_{R_1\dots R_l} = \{(x_1, \dots, x_l) \in \mathbb{R}^l | \forall 1 \le i \le l : \Sigma_{t=1}^{i} x_t^2 \le R_i^2\}. \quad (15)$$

Bounding function. For initial radius R, the vector of $\mathcal{R} = [\mathcal{R}_1, \dots, \mathcal{R}_\beta]$ with condition of $0 \le \mathcal{R}_1 \le \mathcal{R}_2 \le \dots \le \mathcal{R}_\beta = 1$, defines a bounded cylinder-intersections with radius $(R_1, \dots, R_l) = (R \times \mathcal{R}_1, \dots, R \times \mathcal{R}_l)$ for $1 \le l \le \beta$, and consequently can be used to prune the enumeration trees [1].

GNR-pruning (Sound pruning). For a lattice block of $B_{[j,k]} = (b_j, b_{j+1}, \dots, b_k)$ and the coefficient vector $x \in \mathbb{Z}^\beta$, GNR pruning replaces the inequalities of $\|\pi_{k+1-i}(xB_{[j,k]})\| \le R$ for $1 \le i \le k - j + 1$ (as a bounded ball in full-enumeration) by $\|\pi_{k+1-i}(xB_{[j,k]})\| \le \mathcal{R}_i R$ as a cylinder-intersection, where $0 \le \mathcal{R}_1 \le \dots \le \mathcal{R}_{k-j+1} = 1$ [2].

The pseudo-code of the sound pruned enumeration function (GNR-enumeration) can be studied in Appendix B from paper [1]. Based on the definition of GNR-pruning, this paper uses the concepts of final solution vector (usually referred as solution vector) as follows.

Final solution vector. For a lattice block of $B_{[j,k]} = (b_j, b_{j+1}, \dots, b_k)$ and the coefficient vector $x \in \mathbb{Z}^\beta$, the projected vector of $\pi_j(xB_{[j,k]})$ where satisfies the condition of $\|\pi_j(xB_{[j,k]})\| \le \mathcal{R}_{k-j+1} R$ is a final solution vector.

Following fact is a clear proposition on updating radius in GNR pruned enumeration.

Fact 1. If there are some vectors in cylinder-intersection of a GNR pruned enumeration over $\mathcal{L}_\beta$, the shortest one is never eliminated by updating radius and finally be returned as the response of enumeration.

The success probability is one of the main features of bounding function, which is defined as follows [1].

Success probability of bounding function. For any lattice block of $[b_j, b_{j+1}, \dots, b_k]$, initial enumeration radius R and bounding function $\mathcal{R}$, if there is just one lattice vector v in n-dimensional ball with radius of R (i.e., $\|v\| \le R$), the probability of finding solution vector v after GNR pruning by $\mathcal{R}$ in enumeration tree is defined as the success probability of $\mathcal{R}$, which is shown by $\mathrm{p}_{\mathrm{succ}}(\mathcal{R})$.

For analysis of the success probability of GNR bounding function, Gama *et al.* uses following heuristic assumptions (in addition to Gaussian heuristic) [1].

Heuristic 2. "The distribution of the coordinates of the target vector v, when written in the normalized Gram-Schmidt basis $(b_1^*/\|b_1^*\|, \dots, b_n^*/\|b_n^*\|)$ of the input basis, look like those of a uniformly distributed vector of norm $\|v\|$".

Heuristic 3. "The distribution of the normalized Gram-Schmidt orthogonalized basis

$$(b_1^*/\|b_1^*\|, \dots, b_n^*/\|b_n^*\|)$$

of a random reduced basis $(b_1, \dots, b_n)$ looks like that of a uniformly distributed orthogonal matrix".

The coefficient of orthonormal basis vector $z = (z_1, z_2, \dots, z_{k-j+1=d})$ in Heuristic 2, which corresponds with the target lattice vector of v, can be formulated as follows (note that, $b_i^*/\|b_i^*\|$ is the ith vector of the orthonormal basis of $(b_1^*/\|b_1^*\|, \dots, b_n^*/\|b_n^*\|)$ [1],

$$v = [z_1, \dots, z_d] \begin{pmatrix} b_k^*/\|b_k^*\| \\ \vdots \\ b_j^*/\|b_j^*\| \end{pmatrix} = [v_1, \dots, v_m]. \quad (16)$$

The coordinates of the coefficient vector z are reversed (i.e., z_i corresponds to $b_{k-i+1}^*/\|b_{k-i+1}^*\|$), while this is clear that $\|z\| = \|v\|$ [1]. Also, the vector $u = (u_1, u_2, \dots, u_{k-j+1=d}) = (z_1/R, z_2/R, \dots, z_d/R)$ is chosen to be uniformly distributed from the d-dimensional ball of the radius 1 (by the notation of $u \sim \mathrm{Ball}_d$). By using these formulations, success probability of a GNR bounding function $\mathcal{R}$ can be formally defined as follows [1]

$$\mathrm{p}_{\mathrm{succ}}(\mathcal{R}) = \mathrm{Pr}_{u \sim \mathrm{Ball}_d}\left(\forall i \in [1, d], \textstyle\sum_{l=1}^{i} u_l^2 \le R_i^2/R_d^2\right)$$

$$= \Pr_{u\sim \text{Ball}_d}\left(\forall i \in [1,d], \sum_{l=1}^{i} u_l^2 \le \mathcal{R}_i^2\right) \quad (17)$$

Note: Since in last block of BKZ, the size of blocks become less than initial block size of β, so this paper usually uses variable size of $d = k-j+1$ to emphasize this fact.

At the end of this subsection, two families of bounding function including linear-pruning and piecewise-liner pruning, are defined as follows (here, assume that the target vectors have the norm of $\|v\| = R$) [1].

Linear pruning. The linear bounding function is defined as $\mathcal{R}_i^2 = i/\beta$, for $i = 1,\ldots,\beta$ [1]. The uniformly random coefficient vector $z = (z_1, z_2, \ldots, z_{k-j+1=d})$ with length R in Heuristic 2, has the expected squared norm of its projection on the first coordinates of i exactly as $(i/\beta)R^2$ [1]. Also, since for vector u, which is uniformly distributed in the unit sphere Ball_β, paper [1] shows that $\Pr_{u\sim \text{Ball}_\beta}\left(\forall j \in [1,\beta], \sum_{l=1}^{j} u_l^2 \le j/\beta\right) = 1/\beta$ [1].

Piecewise-linear pruning. The piecewise-linear bounding function is defined as $\mathcal{R}_i^2 = 2i\mathbf{a}/\beta$, for $i = 1,\ldots,\beta/2$ and otherwise $\mathcal{R}_i^2 = 2\mathbf{a} - 1 + 2i(1-\mathbf{a})/\beta$, where $\mathbf{a} > 0$ [1]. In paper [1], it is shown that the success probability of $\mathcal{R}$ is roughly $\ge \Omega\left(\beta^{-5/2}(4\mathbf{a}(1-\mathbf{a}))^{\beta/4}\right)$.

Linear pruning is an instance of piecewise-linear pruning for parameter of $\mathbf{a} = 1/2$. In this paper, linear pruning bounding function is represented by $\mathcal{R}_{\text{linear}}$. By using linear pruning, the function of $\text{F}(d)$ (which is frequently used in this paper) is defined as follows

$$\text{F}(d) \approx \frac{1}{\text{psucc}(\mathcal{R}_{\text{linear}})} \in O(d). \quad (18)$$

Note: However success probability of linear pruning bounding function can be estimated by Monte-Carlo method with condition of (25) or by using the efficient technique of Chen-Nguyen in Algorithm 7 from [2], the function $\text{F}(d)$ can be estimated by our following approximation of $\text{F}(d) \approx \frac{d}{3.4+(d-50)/(8d)}$ for $50 < d < 300$.

2.5 Cost Analysis of GNR-enumeration

The estimation of total nodes in sound pruned enumeration tree is the same as the full-enumeration (Schnorr-Euchner enumeration), except that, instead of using balls of radius R, sound pruned enumeration employs the cylinder-intersections of radius $(R_1,\ldots,R_l) = (\mathcal{R}_1 R, \ldots, \mathcal{R}_l R)$ for $1 \le l \le \beta$. The enumeration radius R is defined in [2] as follows (by some partial modification)

$$R = \begin{cases} \min(\sqrt{\Upsilon}\text{GH}(\mathcal{L}_{[j,k]}), \|b_j^*\|), & \text{if } k-j+1 \ge 30 \\ \|b_j^*\|, & \text{otherwise.} \end{cases} \quad (19)$$

where $\text{GH}(\mathcal{L}_{[j,k]})$ is defined by relation (7). In reminder of this paper, the enumeration radius is determined by parameter of r_{FAC}, as follows

$$R = r_{\text{FAC}}\text{GH}(\mathcal{L}) \quad (20)$$

Also by using relation (20) and (19), for block size of $k-j > 30$, the value of r_{FAC} is defined as follows

$$r_{\text{FAC}} = \min(\sqrt{\Upsilon}\text{GH}(\mathcal{L}_{[j,k]}), \|b_j^*\|)/\text{GH}(\mathcal{L}_{[j,k]}). \quad (21)$$

By using Heuristic 1 (Gaussian heuristic), the number of nodes at the level l of the sound pruned enumeration tree can be estimated as follows

$$H_l' = \frac{1}{2}\frac{V_{R_1,\ldots,R_l}}{\prod_{i=k-l+1}^{k}\|b_i^*\|} = \frac{1}{2}\frac{R^l V_{\mathcal{R}_1,\ldots,\mathcal{R}_l}}{\prod_{i=k-l+1}^{k}\|b_i^*\|}. \quad (22)$$

The volume of $\mathcal{C}_{R_1\ldots R_l}$ can be defined as follows

$$\begin{aligned} &\text{V}_{R_1\ldots R_l} = \text{Vol}(\mathcal{C}_{R_1\ldots R_l}) \\ &= \text{V}_l(R)\Pr_{u\sim \text{Ball}_l}\left(\forall j \in [1,l], \sum_{i=1}^{j} u_i^2 \le \mathcal{R}_j^2\right). \end{aligned} \quad (23)$$

Therefore, the total number of nodes in the sound pruned enumeration tree can be estimated as

$$\begin{aligned} &N'(\mathcal{L}_{[j,k]}, \mathcal{R}', R) \approx \sum_{l=1}^{k-j+1} H_l' \approx \\ &\sum_{l=1}^{\beta}\Pr_{u\sim \text{Ball}_l}\left(\forall j \in [1,l], \sum_{i=1}^{j} u_i^2 \le \mathcal{R}_j^2\right) H_l. \end{aligned} \quad (24)$$

The success probability of the bounding function $\mathcal{R}$ can be estimated by Monte-Carlo simulation (see Algorithm 8 in paper [2]) which is used in some tests of this paper, but it is not so efficient, since the number of samples required in this estimation is proportional to $1/\text{p}_{\text{succ}}(\mathcal{R})$ [1]. The Monte-Carlo estimation of success probability is defined by at least $1/\text{p}_{\text{succ}}(\mathcal{R})$ sampling of random vector $u \sim \text{Ball}_d$, and counting the times of success of satisfying the bounding function constraint which is defined as follows [2] (note that $R = R_d$ is the enumeration radius)

$$\begin{aligned} &\left(\forall i \in [1,d], \sum_{l=1}^{i} z_l^2 \le \mathcal{R}_i^2 R_d^2\right) \\ &\equiv \left(\forall i \in [1,d], \sum_{l=1}^{i} u_l^2 \le \frac{R_i^2}{R_d^2} = \mathcal{R}_i^2\right) \\ &\equiv \left(\forall i \in [1,d], \sum_{l=1}^{i} \frac{\omega_{d-l+1}}{\sum_{t=1}^{d}\omega_t} \le \mathcal{R}_i^2\right) \end{aligned} \quad (25)$$

where $\omega_i \leftarrow \text{Gamma}(1/2, 2)$.

2.6 Static Success Probability & Dynamic Success Frequency

The original definition of success probability in Section 2.4, can be applied ideally, when the enumeration radius is nearly equal to the shortest vector length (i.e., $R \approx \lambda_1$). By using relation (20), Roger's theorem [15] determines the approximate number of ${r_{\text{FAC}}}^\beta/2$ solution vector pairs $(v, -v)$ with in the ball of radius $R = r_{\text{FAC}}\text{GH}(\mathcal{L})$ for sufficiently big block size β. Roger's theorem says that, if $r_{\text{FAC}} > 1$ then the success frequency of bounding function $\mathcal{R}$ is more than it's success probability, by factor of $\approx {r_{\text{FAC}}}^\beta/2$.

Note: Since Roger's theorem defines this expectation in average-case, the factor of C_{Roger} is used as an abstract notation (not a real parameter) to emphasize the variance for different lattice blocks and parameter set, however C_{Roger} is set to be 1 in this paper.

The definition of static success probability is the same as the original definition of success probability (which comes in Section 2.4 and is defined at first in [1]) when enumeration radius R is set to be λ_1 (i.e., the norm of shortest vector). This definition can be declared in following forms.

Static success probability of bounding function. For any lattice block of $[b_j, b_{j+1}, \ldots, b_k]$, initial enumeration radius $R = \lambda_1$ and bounding function $\mathcal{R}$, the static success probability of $\mathcal{R}$ is defined as the probability of finding solution vector v (with length of λ_1) after GNR pruning by bounding function $\mathcal{R}$ in enumeration tree.

By the original definition of success probability (which comes in Section 2.4 and at first is defined in [1]), this is clear that, in this definition, there is just one lattice vector v with length of λ_1 in n-dimensional ball with radius of R (i.e., $\|v\| \leq R = \lambda_1$). The static success probability is formulated as follows:

$$p_{succ}^{new0}(\mathcal{L}_{[1,d]}, \mathcal{R}, R) = \Pr_{u \sim \text{Ball}_d}\left(\forall j \in [1,d], \sum_{i=1}^{j} u_i^2 \leq \mathcal{R}_j^2\right). \quad (26)$$

Note: The definition of success probability in Section 2.4 and relation (17) corresponds to static success probability p_{succ}^{new0} which is defined in (26).

In other side, by using Roger's theorem, dynamic success frequency can be defined as follows.

Dynamic success frequency of bounding function. For any lattice block of $\mathcal{L} = [b_j, b_{j+1}, \ldots, b_k]$, initial enumeration radius $R = r_{\text{FAC}}\text{GH}(\mathcal{L})$ and bounding function $\mathcal{R}$ with static success probability $\text{p}_{\text{succ}}(\mathcal{R})$, there are ${r_{\text{FAC}}}^{\beta}/2$ solution vectors in n-dimensional ball with radius of R, consequently the frequency of solution vectors v in enumeration tree (where $\|v\| \leq R$) after GNR-pruning by $\mathcal{R}$ is estimated by $\text{p}_{\text{succ}}(\mathcal{R}){r_{\text{FAC}}}^{\beta}/2$.

The dynamic success frequency is formulated as follows (while p_{succ}^{new0} is defined in (26))

$$f_{succ}^{new0}(\mathcal{L}_{[1,d]}, \mathcal{R}, R) = C_{Roger}\frac{{r_{\text{FAC}}}^{\beta}}{2} p_{succ}^{new0}(\mathcal{L}_{[1,d]}, \mathcal{R}, R). \quad (27)$$

Note: However GNR-enumeration uses updating radius, if this is assumed that there is no updating radius, then the dynamic success frequency of bounding function can be assumed as the exact number of solutions which be visited in enumeration function, else this dynamic success frequency would be more than the exact number of solutions visited in GNR-enumeration.

Note: Dynamic success frequency would be any real number, even much bigger than 1.

2.7 Norm of Full/Pruned Enumeration Solution

For high dimensional lattice basis, it is assumed that the basis tends to be random [2]. In fact, for exact definition of random lattices, it can be shown that as the lattice dimension tends to inanity, the expected value of the best vector of the lattice converges to $\text{GH}(\mathcal{L}_{[j,k]})$ [11, 15]. Currently, just general bounds from a theoretical point of view are known on how small $\lambda_1(\mathcal{L}_{[j,k]})$ should be. Chen and Nguyen performed some experimental tests which show that for the sufficiently large block size β, the norm of the best vector is nearly around $\text{GH}(\mathcal{L}_{[j,k]})$ (see Figure 4 in [2]). Also Chen and Nguyen [2] performed some experimental tests to compare the final solution norm of enumeration with value of $\text{GH}(\mathcal{L}_{[j,k]})$, depending on the starting index j of a local block for one round of BKZ, so that for the first indices j, the final norm is significantly lower than $\text{GH}(\mathcal{L}_{[j,k]})$. This behaviour of solution norm in running of BKZ is "head concavity phenomenon" in BKZ, which is discussed in [3]. However, for the last indices (tail of GSO norms), the GSO norms are significantly larger (which can be named as "tail convexity"). Finally, in the middle indices, which includes the most of the enumeration calls, the solution norms are mostly bounded as follows [2]

$$0.95\ \text{GH}(\mathcal{L}_{[j,k]}) \leq \|v\| \leq 1.05\ \text{GH}(\mathcal{L}_{[j,k]}). \quad (28)$$

This third behaviour of BKZ, can be named as "random manner of middle lattice blocks". To the best of our knowledge, this test in paper [2] is performed with some block sizes ≤ 70. It is believed in this paper that these experiments correspond to the behaviour of GSA, which is not satisfied exactly in the first and last indexes [16]. There are several works modifying the reduction algorithms so that their outputs satisfy the GSA, but it seems difficult to obtain GSA shape easily in practice [4, 13, 16]. Also in the test of Chen and Nguyen (see Figure 3 in [2]), the average norm of λ_1 in the blocks $\mathcal{L}_{[j,k]}$ from the basis B during the BKZ reduction, in the middle indices, is almost $\text{GH}(\mathcal{L}_{[j,k]})$. Besides the blocks of a basis in running of BKZ reduction in the experimental test of [2], also the best norm of solutions for Darmstadt SVP challenges [17, 18] (which are assumed to be close to λ_1) roughly verify the bound of (28) for the random lattice blocks.

Although such experimental results which are pointed at the beginning of this subsection are so useful, some strong theoretical evidence is needed to

define the probability distribution of enumeration solution norms. In fact, the probability distribution of best solution norm for a lattice basis/block is stated in Chen's thesis [11], as following theorem [3].

Theorem 1. *For random lattice $\mathcal{L}_1$ with rank n and unit volume, the distribution of $V_n(1)\lambda_1(\mathcal{L}_1)^n$ converges to distribution of Expo(1/2) as $n \to \infty$.*

By using $\lambda_1(\mathcal{L}) = X^{1/n}\text{GH}(\mathcal{L})$ which X sampled from Expo(1/2), the expected value and variance for $\lambda_1(\mathcal{L})$ are computed as follows [3],

$$\text{E}\left[\lambda_1(\mathcal{L})\right] = 2^{1/n}\Gamma(1+1/n)\text{GH}(\mathcal{L}), \tag{29}$$

$$V\left[\lambda_1(\mathcal{L})\right] = 2^{\frac{2}{n}}\left(\Gamma(1+\tfrac{2}{n}) - \Gamma(1+\tfrac{1}{n})^2\right)\text{GH}(\mathcal{L})^2, \tag{30}$$

also for lattices with large n [3],

$$\text{E}\left[\lambda_1(\mathcal{L})\right] \approx (1 + 0.116/n + o(1/n))\text{GH}(\mathcal{L}),$$

$$\mathcal{V}\left[\lambda_1(\mathcal{L})\right] \approx \tfrac{\pi^2}{6n^2}(1+o(1))\text{GH}(\mathcal{L})^2.$$

The random variable of $\lambda_1(\mathcal{L})$ for lattices of rank d can be sampled by following formula [3],

$$\lambda_1(\mathcal{L}) \leftarrow \left(\frac{X\text{vol}(\mathcal{L})}{V_d(1)}\right)^{1/d}, \text{ where } X \leftarrow \text{Expo}(1/2). \tag{31}$$

Note: Theorem 1 should be considered for full-enumeration (i.e., a GNR enumeration function including a bounding function with static success probability 1 or dynamic success frequency of ${r_{\text{FAC}}}^{\beta}/2$), not for any pruned enumeration with any success probability.

Since for high block sizes, pruned enumeration is usually used, so the shortest vector of each lattice block may not be returned by enumeration function. In paper [4], for the lattice block $\mathcal{L}_{[j,k]}$ (with dimension of β), the expected norm of the solution vector of a pruned enumerations is determined by using Lemma 1 of paper [4], which computes the expected value of the shortest length of vectors from origin to the points uniformly sampled from the β-dimensional unit ball as follows (see proof in [4])

Lemma 1. *For K points $x_1, x_2, \ldots, x_k$ which are uniformly sampled from the β-dimensional unit ball, the expected value of the shortest length of vectors from origin to these points can be estimated as follows [4],*

$$\text{E}\left[\min_{\substack{i\in\{1,\ldots,K\}\\ \|x_i\|\le 1}}\|x_i\|\right] = K\text{Beta}\left(K, \tfrac{\beta+1}{\beta}\right) = K\int_0^1 t^{1/\beta}(1-t)^{K-1}dt. \tag{32}$$

In particular, for $K = 1$, this expected value is $\beta/(\beta+1)$ [4]. Therefore, for $R = r_{\text{FAC}}\text{GH}(\mathcal{L}_\beta)$, the expected norm of the solution vector returned by pruned enumeration with success probability $2/{r_{\text{FAC}}}^{\beta}$ can be computed as follows [4],

$$\text{E}'[\|v\|] = \frac{\beta}{\beta+1}r_{\text{FAC}}\text{GH}(\mathcal{L}_\beta). \tag{33}$$

3 Our Sampling Method of Enumeration Solution

One of main components in design of BKZ-simulation with GNR-pruned enumeration as SVP-solver is sampling the enumeration solution. Enumeration solution v is represented in former BKZ-simulation just by estimating the expected value of the norm of this solution as $\|v\|$. This paper tries to introduce an efficient and exact sampling method for enumeration solution v, in the way that samples both norm and coefficient vectors of enumeration solution. In fact, the GNR enumeration function over each lattice block $\mathcal{L}_{[b_j,\ldots,b_k]}$ in BKZ algorithm returns the coefficient vector y, which can be used to compute the solution vector v by linear combination of this lattice block vectors as $v = \sum_{l=j}^{k} y_l b_l$. To the best of our knowledge, no precise and explicit analysis of sampling the coefficient vector y is considered in former BKZ-simulations. Also, no sampling method for norm of GNR-pruned enumeration solution with any success probability is introduced, however paper [3] introduces a sampling method (see line 14 of Algorithm 4 from paper [3]) which is stated in Chen's thesis [11] just for full-enumeration (see Theorem 1 in Section 2.7), not for any GNR pruning with any success probability. Also our approximation of cost for GNR-enumeration by optimal bounding function is introduced at the end of this section.

3.1 New Sampling Method for Solution Norm

The expected norm of solution vector returned by GNR-pruned enumeration with success probability $2/{r_{\text{FAC}}}^{\beta}$ can be estimated by following limit

$$\text{E}\left[\|v\|\right] = \frac{\text{GH}(\mathcal{L}_\beta)}{{r_{\text{FAC}}}^{\beta}}\lim_{t\to\infty}\sum_{i=1}^{t}\left(r_{\text{FAC}} - i\omega + \tfrac{\omega}{2}\right)\times \left((r_{\text{FAC}} - i\omega + \omega)^{\beta} - (r_{\text{FAC}} - i\omega)^{\beta}\right), \tag{34}$$

where $\omega = (r_{\text{FAC}} - 1)/t$.

As mentioned in Section 2.7, the expected value for (34) is equal to $\text{E}'[\|v\|]$ in formula (33). Also, the median for the norm of these solution vectors (for a pruned enumeration with success probability $2/{r_{\text{FAC}}}^{\beta}$) can be estimated as follows

$$\text{Median}[\|v\|] = \sqrt[\beta]{\tfrac{{r_{\text{FAC}}}^{\beta}+1}{2}}\text{GH}(\mathcal{L}_\beta). \tag{35}$$

The condition of $\text{E}\left[\|v\|\right] < \text{Median}[\|v\|]$ can be considered for enumeration solution v. As mentioned, by using formula (33), this is possible to estimate

the expected norm of the solution vector returned by pruned enumeration with success probability $2/{r_{\text{FAC}}}^{\beta}$, while sampling the norm of this solution vector for success probability $2/{r_{\text{FAC}}}^{\beta}$ can be estimated simply by our proposed lemma, as follows.

Lemma 2. *The norm of solution vector v returned by a pruned enumeration with radius factor of r_{FAC} and success probability $2/{r_{\text{FAC}}}^{\beta}$ over lattice block $\mathcal{L}_\beta$ can be sampled by (36)*

$$\|v\| = \sqrt[\beta]{1+\text{rand}_{[0\ldots1]}({r_{\text{FAC}}}^{\beta}-1)}\,\text{GH}(\mathcal{L}_\beta) \tag{36}$$

See proof in Appendix B.1.

This lemma can be generalized for other success probability of $P \geq 2/{r_{\text{FAC}}}^{\beta}$, by repeating this sampling method and selecting the shortest one, however running time of this technique may be non-tolerable as the number of solution vectors would be increased. Here we generalize Lemma 2 for any success probability $0 < P \leq 1$ by an efficient and simple technique in Lemma 3.

Lemma 3. *If norm of shortest vector in lattice block $\mathcal{L}_\beta$ is less than enumeration radius R, the norm of solution vector v which is returned by a pruned enumeration with radius factor of r_{FAC} and static success probability P over lattice block $\mathcal{L}_\beta$ can be sampled by*

$$\|v\| = \begin{cases} 1: X^{1/\beta}\text{GH}(\mathcal{L}_\beta), \text{ where } X \leftarrow \text{Expo}(1/2), & \text{if } P \approx 1 \\ 2: \sqrt[\beta]{1+\text{rand}_{[0\ldots1]}(\frac{2}{P}-1)}\,\text{GH}(\mathcal{L}_\beta), & \text{if } \frac{2}{{r_{\text{FAC}}}^{\beta}} \leq P < 1 \\ 3: \sqrt[\beta]{1+\text{rand}_{[0\ldots1]}({r_{\text{FAC}}}^{\beta}-1)}\,\text{GH}(\mathcal{L}_\beta), & \text{if } P < \frac{2}{{r_{\text{FAC}}}^{\beta}} \,\&\, \text{rand}_{[0\ldots\frac{2}{{r_{\text{FAC}}}^{\beta}}]} \leq P \\ 4: \text{Un-Successfull}, & \text{if } P < \frac{2}{{r_{\text{FAC}}}^{\beta}} \,\&\, \text{rand}_{[0\ldots\frac{2}{{r_{\text{FAC}}}^{\beta}}]} > P \end{cases} \tag{37}$$

See proof in Appendix B.2.

The pseudo-code of our efficient sampling algorithm by Lemma 3 can be studied in Algorithm 2 from Appendix A.2. Furthermore, our following lemma determines the number of rounds which BKZ_β algorithm with GNR-enumeration and bounding function of $\mathcal{R}$ and any dynamic success frequency of f_{succ} needs to reach the quality of a basis which is reduced by BKZ_β algorithm with full-enumeration.

Lemma 4. *For given block size of β, enumeration radius R defined by (21) and initial radius parameter $r_{\text{FAC}} = \sqrt{\Upsilon} = 1 + 1/C_r$, the Hermite-factor of a basis reduced by the rounds number of $\mathcal{N} \leq C_0/C_r$ from BKZ_β algorithm including GNR-enumeration with dynamic success frequency of f_{succ1} and the expected norm of $0 < E[\|v\|] = \frac{1}{\phi(\beta, f_{succ1})} \times R$ where $\phi(\beta, f_{succ1}) = 1 + 1/C_0 > 1$, is equal to the Hermite-factor of this basis after one round of BKZ_β-reduction with full-enumeration.*

See proof in Appendix B.3.

Note: Full-enumeration corresponds with GNR-enumeration including a bounding function $\mathcal{R}$ with static success probability of $\text{p}_{\text{succ}}(\mathcal{R}) = 1$ and dynamic success frequency of $f_{\text{succ}_0} = {r_{\text{FAC}}}^{\beta}/2$ (see our definitions in Section 2.6).

Note: By using our definitions in Section 2.6, dynamic success frequency of $f_{\text{succ}_1} = 1$ corresponds with static success probability of bounding function $\mathcal{R}$ as $\text{p}_{\text{succ}}(\mathcal{R}) = 2/{r_{\text{FAC}}}^{\beta}$ and consequently $\phi(\beta, f_{\text{succ}_1} = 1) = \beta/(\beta+1)$ (see relation (33) in Section 2.7).

Note: For GNR pruned enumeration including a bounding function with dynamic success frequency of $f_{\text{succ}_1} = 1$ (i.e., static success probability of $\text{p}_{\text{succ}}(\mathcal{R}) = 2/{r_{\text{FAC}}}^{\beta}$), initial enumeration radius with $r_{\text{FAC}} = \sqrt{\Upsilon} = 1.05$ and parameters of $C_r = 20$ and $C_0 = \beta$, the rounds number of $\mathcal{N}$ in Lemma 4 can be estimated as $\mathcal{N} \leq \beta/20$.

Remark 2. Since (to the best of our knowledge) the best time complexity for heuristically sieving algorithm is $O(2^{292\beta})$ with exponential space order [19], which by Grover algorithm, this cost would be lowered to $O(2^{265\beta})$ with exponential space, so the optimal cost SVP-oracles currently have exponential time/space order and consequently for high dimensional lattices, any high block sizes cannot be used in BKZ reduction. This means that, as the lattice dimension tends to in?nity, $\beta \in O(1)$ and consequently, the rounds number of $\mathcal{N}$ in Lemma 4, for dynamic success frequency $f_{\text{succ}} \geq 1$, asymptotically belongs to $O(1)$.

3.2 Our Sampling Method for Coefficient Vectors of Enumeration Solution

In this section, the structure and probability distribution of coefficient vectors corresponding with solution vector v returned by GNR-enumeration are defined. Consequently the sampling method of them are designed, while to the best of our knowledge, no such deep discussions for sampling these coefficient vectors are considered in former BKZ-simulations. This section includes following steps. In Section 3.2.1, the structure of coefficient vectors of w and y is analyzed. In Section 3.2.2, the estimation of index of last non-zero coefficient in vector of w is introduced (which is notated by g). In Section 3.2.3, the probability distributions of coefficient vectors of w, z and y are analysed, and sampling methods of them are introduced. Finally some complementary discussions on coefficient vectors are introduced in Section 3.2.4.

3.2.1 Structure of Coefficient Vector w and y

By using Heuristic 2 and Heuristic 3, the uniform randomness of the coefficient vector $z = (z_1, z_2, \ldots, z_d)$ over the normalized Gram-Schmidt basis $(b_d^*/\|b_d^*\|, \ldots, b_1^*/\|b_1^*\|)$ is assumed. Also, for the enumeration radius R, the corresponding vector of $u = (u_1, u_2, \ldots, u_d) = (z_1/R, z_2/R, \ldots, z_d/R)$ can be assumed to be uniformly distributed from the d-dimensional ball with radius 1 (see Section 2.4). For given lattice block $\mathcal{L}_{[1,d]}$, the enumeration over this block returns the solution vector v, where $\|v\| < \|b_1^*\|$. The solution vector v can be written by the coefficient vector $w = (z_d/\|b_1^*\|, \ldots, z_2/\|b_{d-1}^*\|, z_1/\|b_d^*\|)$ on the GSO block basis as follows (corresponding to (16))

$$v = (v_1, \ldots, v_m) = (w_1, \ldots, w_d) \begin{pmatrix} b_1^* \\ \vdots \\ b_d^* \end{pmatrix} \tag{38}$$

Note: The solution vector v from enumeration over lattice block $\mathcal{L}_{[j,k]}$ is a GSO projected vector which is orthogonal over the previous basis vectors in $\mathcal{L}_{[1,j-1]}$ (remember that, here the notation of $\mathcal{L}_{[1,d]}$ represents $\mathcal{L}_{[j,k]}$).

By inserting the solution vector v at first of the lattice block $\mathcal{L}_{[1,d]}$ (which results in the block of $(v, b_1^*, \ldots, b_d^*)$ with $d+1$ vectors), one of the vectors from the GSO block $(v, b_1^*, \ldots, b_d^*)$ should be eliminated after updating GSO norms of these $d+1$ vectors. Lattice enumeration uses the integer coefficients y_i for enumerating over the projected lattice block $\mathcal{L}_{[1,d]}$, therefore the coefficients w_i in vector w depend on integer entries in the vector of y, as follows (remember that, here the projection notation of $\pi_1(\cdot)$ represents $\pi_j(\cdot)$),

Note: The dimension of b_i^* and v is m which differs from the rank of lattice block (i.e., block size of d).

$$v = y \times \begin{pmatrix} \pi_1(b_1) \\ \vdots \\ \pi_1(b_d) \end{pmatrix} = y \times \begin{pmatrix} b_1^* \\ \vdots \\ b_d^* + \sum_{i=1}^{d-1} \mu_{d,i} b_i^* \end{pmatrix} \tag{39}$$

where $y = (y_1, y_2, \ldots, y_d)$.

$$v = \underbrace{(y_1 + \sum_{i=2}^{d} y_i \mu_{i,1}) b_1^*}_{z_d} + \cdots + \underbrace{(y_g + \sum_{i=g+1}^{d} y_i \mu_{i,g}) b_g^*}_{z_{d-g+1}} + \cdots + \underbrace{y_d b_d^*}_{z_1} \Rightarrow$$

$$w = \left[\underbrace{y_1 + \sum_{i=2}^{d} y_i \mu_{i,1}}_{w_1}, \ldots, \underbrace{y_g + \sum_{i=g+1}^{d} y_i \mu_{i,g}}_{w_g}, \ldots, \underbrace{y_d}_{w_d} \right] \tag{40}$$

Consequently the following main theorem can be introduced.

Theorem 2. *The projected vector $b_g^* \in \{b_1^*, \ldots, b_d^*\}$ which is eliminated after inserting the enumeration solution v, has the GSO norm of $\|b_g^*\| \leq \|v\|$, and the coefficient w_g is always the last non-zero coefficient in vector of w in lattice block of $\mathcal{L}_{[1,d]}$, as follows*

$$w_g = y_g = 1. \tag{41}$$

See proof in Appendix B.4.

Based on (40), a zero coefficient of y_i does not always result in $w_i = 0$, except for indices after the last non-zero coefficient of y_g.

3.2.2 Estimation of Last Non-zero Index g

In this section, the last non-zero index for vectors of w and y (corresponding with first non-zero index for vectors of z and u) would be determined statistically.

Lemma 5. *After inserting the enumeration solution v at the first of lattice block $\mathcal{L}_{[1,d]}$, the vector of b_1 is never eliminated in updating GSO.*

Proof. By using (19), the condition of $\|v\| < \|b_1^*\|$ is always true, while by using the fact of $\|b_g^*\| \leq \|v\|$, then $\|b_g^*\| \neq \|b_1^*\|$ and consequently $g > 1$. □

If $\|b_g^*\|^2 > R^2 \mathcal{R}_{d-g+1}^2$ then the bounding function $\mathcal{R}$ prunes all solution vectors of the lattice block with last non-zero vector index g, and returns the solution vectors with other last non-zero vector index, if there is. Also following lemma motivates us for assuming $g = d$ when radius factor of r_{FAC} is not too small

Lemma 6. *For block size d, the condition of $\|b_g^*\|^2 > R^2 \mathcal{R}_{d-g+1}^2$ is never satisfied for a piecewise-linear bounding function $\mathcal{R}$ with parameter $\frac{2(\pi e)^2}{d \times r_{\text{FAC}}^4} \leq \boldsymbol{a}$.*

See proof in Appendix B.5.

In fact, the concept behind the condition of $\|b_g^*\|^2 \leq R^2 \mathcal{R}_{d-g+1}^2$, formally can be defined by cut point index, as follows.

Cutting point. The enumeration cut point index is defined as the last GSO norm index $\text{CUT} \in [1, d]$ where $\|b_{\text{CUT}}^*\|^2 \leq R^2 \mathcal{R}_{d-\text{CUT}+1}^2$ and $\text{CUT} \geq 2$.

Remark 3. For an input lattice block $\mathcal{L}_{[1,d]}$ and bounding function $\mathcal{R}$, the cutting point CUT would be non-negligibly smaller than d, just if this lattice block is preprocessed too much, in the way that the quality of GSO shape is too well (i.e., q-factor is too small) or

the basis is formed by some special structures (such as the one in Darmstadt lattice challenges, which has exactly $\|b_i^*\|=1$ and consequently $q=1$ in many last GSO vector index i), or/and with an extremely small success probability of $\mathcal{R}$.

Surprisingly, this is possible to see some GNR enumerations with cutting point in first block indices, which nearly always does not lead to an enumeration solution (such as in Darmstadt lattice challenges)! The pseudo-code of generating a piecewise-linear bounding function with a specific success probability and determining the cutting point of the generated bounding function is introduced in Algorithm 3 from Appendix A.3. Following lemma introduces an expectation for eliminated vector b_g, in full-enumerations.

Lemma 7. *Under assumption of Heuristic 2 for full enumerations, the statistical expected vector of the lattice block which is eliminated after updating GSO is $b_g \approx b_d$ (i.e., $g \approx \text{CUT} = d$).*

Proof. The proof is trivial, since Heuristic 2 states that, the solution vector v is a uniformly distributed vector of norm $\|v\|$ on the normalized Gram-Schmidt basis $(b_1^*/\|b_1^*\|,\dots,b_d^*/\|b_d^*\|)$, so the chance of every GSO vector of b_i^* to be used in linear combination of v is ≈ 1 (for this end, imagine that a unit vector in two dimensional circle with unit radius is uniformly distributed, now the probability of each two perpendicular vertices in generating this vector in the unit-radius circle is ≈ 1). At result, the last non-zero entry of y would be y_d with probability ≈ 1, thereby the estimation of $b_g \approx b_d$ can be concluded. □

According to Heuristic 2 and Heuristic 3, for sampling g, only this is needed to determine the probability of whether an integer coefficient y_i to be zero or not. If the probability of that an integer coefficient y_i is none-zero, is assumed to be p, then

$$\Pr(g=i)=\tfrac{p(1-p)^{d-i}}{1-(1-p)^d} \Rightarrow \Pr(g=i)=\tfrac{p(1-p)^{d-i}}{1-(1-p)^{d-1}}, \text{ where } i\geq 2. \quad (42)$$

If the probability of whether an integer coefficient y_i to be zero or not, is assumed to be $p\approx 1/2$, then by using (42) and Lemma 5, the vector b_i (where $1\leq i\leq \text{CUT}$) with probability of $\Pr(g=i)=2^{i-2}/(2^{d-1}-1)$ is the eliminated vector, also for block size $d\geq 20$, the expected value of index for the eliminated vector is $\mathrm{E}[g]=\sum_{i=1}^{\text{CUT}}(2^{i-2}i)/(2^{d-1}-1)\approx \text{CUT}-1$. Accordingly, the index of g can be sampled by

$$g=\left\lfloor \log_2\left((2^d-1)\text{rand}_{[2/(2^d-1),\dots,1]}\right)\right\rfloor+1. \quad (43)$$

There are massive observations in Section 4.1 with useful statistical results for index g in lattice enumeration over random lattice blocks which are reduced by $\text{BKZ}_{\beta=45}$ in different settings. For determining PDF (probability distribution function) of g, the experiments in Section 4.1 are not sufficient and needed to be applied for sufficiently big block sizes. Instead of using experimental results for determining this PDF, by using the Roger's theorem for sufficiently big block sizes, the probability distribution of g can be estimated in Lemma 8.

Lemma 8. *For a GNR-enumeration with radius $R=r_{\text{FAC}}\text{GH}(\mathcal{L}_{[1,d]})$ over lattice block of $\mathcal{L}_{[1,d]}$ with quality q, sufficiently big block size d and cut point index* CUT*, the probability distribution of g for the solution vectors v which is returned from this enumeration is estimated as follows*

$$\Pr(g=i)=r_{\text{FAC}}{}^{i-d}(d/i)^{i/2}\times \frac{(\|b_1^*\|\dots\|b_d^*\|)^{\frac{i}{d}}-\|b_i^*\|(\|b_1^*\|\dots\|b_d^*\|)^{\frac{i-1}{d}}\sqrt{\frac{i^i}{(i-1)^{i-1}d\ r_{\text{FAC}}{}^2}}}{(\|b_1^*\|\dots\|b_i^*\|)} \quad (44)$$
$$\approx r_{\text{FAC}}{}^{i-d}\sqrt{(dq^{i-d}/i)^i}\left(1-\sqrt{\frac{i^iq^{d-2i+1}}{(i-1)^{i-1}d\ r_{\text{FAC}}{}^2}}\right)$$
$$\text{for } r_{\text{FAC}}\geq\tfrac{\text{GH}(\mathcal{L}_{[1,i]})}{\text{GH}(\mathcal{L}_{[1,d]})} \text{ and } i\leq \text{CUT},$$
$$\text{else } \Pr(g=i)=0.$$

See proof in Appendix B.6.

For sufficiently big block sizes, the expected value of g is predicted as $\mathrm{E}[g]\approx d$ and corresponding variance of $\mathcal{V}[g]$ is predicted to be negligible. Our experimental tests in Section 4.1 gives some useful information on statistical measures of g in enumeration successes of actual running $\text{BKZ}_{\beta=45}$. Also a simple comparison of formula (44) in Lemma 8 with formula (42) is introduced in Figure 2 from Section 4.1 for dimension 60.

Note: Although the first vectors of block usually are prone to violate the condition of $\|b_i^*\|\leq\|v\|$, for sufficiently high block sizes, the probability of these vectors to be selected as the eliminated vector would be nearly zero, which is consistent with Lemma 8.

Note: The probability distribution of g in Lemma 8, is consistent with Heuristic 2, relation (42), Lemma 7, and our massive observations in Section 4.1.

Note: For an input enumeration radius $R=r_{\text{FAC}}\text{GH}(\mathcal{L}_{[1,d]})$, the enumeration radius factor r_{FAC} is decreased for smaller block sizes i in $\mathcal{L}_{[1,i]}$, since the Gaussian heuristic is increased for this smaller block sizes, i.e., $\text{GH}(\mathcal{L}_{[1,d]})<\text{GH}(\mathcal{L}_{[1,i]})$; Consequently, by using the probability distribution of solution vector norms in Theorem 1, this is probable that there are no solutions in these smaller block sizes! Therefore the condition of $r_{\text{FAC}}>\text{GH}(\mathcal{L}_{[1,i]})/\text{GH}(\mathcal{L}_{[1,d]})$ keeps the use of this probability distribution (44) just for the solutions (with last non-zero index g) whose norms are bigger than $\text{GH}(\mathcal{L}_{[1,g]})$. Accordingly, the relation (44) is found to be consistent with Theorem 1 too.

By use of Lemma 8 (and Lemma 7), the following corollary can introduce the approximate index of CUT

in average-case, if there is an enumeration solution.

Corollary 1. If GNR enumeration returns a solution vector, this is expected to $g \approx \text{CUT} \approx d$ in average-case.

Also Corollary 1 is consistent with Remark 3. After determining the probability distribution of g in Lemma 8, how many times should parameter g be sampled until corresponding constraints in this lemma would be satisfied? The number of samples of index g is K which is the number of expected solutions existing in the polytope of bounding function $\mathcal{R}$ (which is estimated by dynamic success frequency). Our proposed method for sampling the index of g can be studied in Algorithm 4 from Appendix A.4.

3.2.3 Our Sampling Method of Vector w

In this section, the sampling method for coefficient vector w (and consequently vector z) would be introduced. One of the main phase in this step is to sample a uniformly-distributed unit vector on a d-dimensional unit ball, which can be estimated in Remark 4 as follows.

Remark 4. For given random variable $\mathcal{X} = (\mathcal{X}_1, \mathcal{X}_2, \ldots, \mathcal{X}_d)$ iid$\sim \mathcal{N}(0,1)$, by assuming the vector of $\mathcal{X}/\sqrt{\mathcal{X}_1^2 + \cdots + \mathcal{X}_d^2}$ as a uniformly-distributed unit vector on the surface of d-dimensional unit-radius sphere, the formula (45) samples the vector of w for a typical GSO lattice block $[b_1^*, b_2^*, \ldots, b_g^*, \ldots, b_d^*]$ under a full-enumeration

$$w_i \leftarrow \mathcal{X}_i \sqrt{\frac{\|v\|^2 - \|b_g^*\|^2}{\|b_i^*\|^2 \sum_{t=1}^{g-1} \mathcal{X}_t^2}}, \tag{45}$$

where $\mathcal{X}_i \sim \mathcal{N}(0,1)$. It is clear that, the corresponding vector z can be sampled by Remark 4 as follows

$$z_{d-i+1} \leftarrow \mathcal{X}_i \sqrt{\frac{\|v\|^2 - \|b_g^*\|^2}{\sum_{t=1}^{g-1} \mathcal{X}_t^2}}, \tag{46}$$

where $\mathcal{X}_i \sim \mathcal{N}(0,1)$. Also by defining the random variable of $\omega_i = \mathcal{X}_i^2$, where $\omega_i \leftarrow \text{Gamma}(1/2, 2)$, the relations (45) and (46) can be re-defined as follows

$$w_i \leftarrow (-1)^{\lfloor \text{rand}_{[0\ldots 2)} \rfloor} \sqrt{\frac{\omega_i(\|v\|^2 - \|b_g^*\|^2)}{\|b_i^*\|^2 \sum_{t=1}^{g-1} \omega_t}}, \tag{47}$$

where $\omega_i \leftarrow \text{Gamma}(1/2, 2)$,

$$z_{d-i+1} \leftarrow (-1)^{\lfloor \text{rand}_{[0\ldots 2)} \rfloor} \sqrt{\frac{\omega_i(\|v\|^2 - \|b_g^*\|^2)}{\sum_{t=1}^{g-1} \omega_t}}, \tag{48}$$

where $\omega_i \leftarrow \text{Gamma}(1/2, 2)$. The Monte-Carlo estimation of success probability for bounding function in (25) is consistent with Remark 4 (see [2]). The random vector z under Heuristic 2 and Heuristic 3, is a uniformly distributed vector with norm of $\sqrt{\|v\|^2 - \|b_g^*\|^2}$ over the normalized Gram-Schmidt lattice block $(b_d^*/\|b_d^*\|, \ldots, b_1^*/\|b_1^*\|)$, and it's entries as random variable of z_t are dependent whose expected values are estimated in Lemma 9.

Lemma 9. *For a cut point index of* CUT*, also for a solution vector v returned by a full-enumeration over a typical GSO lattice block $[b_1^*, b_2^*, \ldots, b_g^*, \ldots, b_d^*]$, the expected value for all entries of z_x^2 is approximately similar to each other as follows,*

$$E\left[z_x^2\right] \approx \frac{\|v\|^2 - \|b_g^*\|^2}{\text{CUT}}, \tag{49}$$

where $x \in \{d - g + 2, \ldots, d\}$.

See proof in Appendix B.7.

Corollary 2. For an enumeration solution vector v returned by a full-enumeration over a typical GSO lattice block $[b_1^*, b_2^*, \ldots, b_g^*, \ldots, b_d^*]$, the expected value for entries of w_x^2 is approximated with an increasing slope as follows

$$\text{E}\left[w_x^2\right] \approx \frac{\|v\|^2 - \|b_g^*\|^2}{\text{CUT}\|b_x^*\|^2}, \text{ where } x \in \{1, \ldots, g-1\}. \tag{50}$$

The running time of rejection sampling for coefficient vector w (and vector z) with number of $\approx 1/\text{p}_{\text{succ}}(\mathcal{R})$ rejections before one success is not tolerable for bounding functions with asymptotically small success probabilities. Therefore, some efficient techniques for this sampling should be used, however the accuracy of sampling distribution would be lowered a bit. Our idea for sampling coefficient vector w originates from following lemma (note that, $\|v\|$ represents the solution norm and R represents the enumeration radius).

Lemma 10. *Under condition of $\|v\|^2/R^2 \approx 1 - \varepsilon$ where $\varepsilon \approx O(1/d)$ and by assumption of uniform distribution for coefficient vector z on the normalized orthogonal matrix $(b_d^*/\|b_d^*\|, \ldots, b_1^*/\|b_1^*\|)$, when random variable $w_l^2 = z_t^2/\|b_l^*\|^2$ would be sampled by (47) under full-enumeration, the expected value of $X_i = \sum_{t=1}^{i} z_t^2/R^2$ can be closely approximated by entries of linear pruned bounding function $\mathcal{R}_{linear}$ (for $l = d - t + 1$).*

See proof in Appendix B.8. As mentioned in Section 2.4, linear pruning is an instance of piecewise-linear pruning by setting parameter of $\mathbf{a} = 1/2$. Our proposed approximate method of sampling vector w can be generalized for piecewise-linear pruning (instead of linear pruning), which Claim 1 states the main idea behind it

Claim 1. Under the condition of $\|v\|^2/R^2 \approx 1 - \varepsilon$ where $\varepsilon \approx O(1/d)$, for a typical GSO block $(b_1^*, b_2^*, \ldots, b_g^*, \ldots, b_d^*)$ and piecewise-linear bounding function $\mathcal{R}'$ with parameter $\mathbf{a}'$, if random variable of $w_j^2 = z_t^2/\|b_j^*\|^2$ would be sampled by rejection sampling in relation (47), then the expected value of

random variable $X_i = \sum_{t=1}^{i} z_t^2/R^2$ which is bounded by $R_i^{'2}$, can be approximated by a piecewise-linear bounding function $\mathcal{R}$ (with parameter $\mathbf{a}$) and static success probability of $\mathrm{p_{succ}}(\mathcal{R}) = \mathrm{p_{succ}}(\mathcal{R}')/\mathrm{F}(d)$, where function $\mathrm{p_{succ}}$ is defined in (17) and function F is defined in (18).

Claim 1 introduces the approximate expected value for $X_i = \sum_{t=1}^{i} z_t^2/R^2$ and consequently for entries of z_t^2, while this is preferred for BKZ-simulations to use a statistical random sampling from (48) bounded by $\mathcal{R}'$. For this end, a suitable statistical sampling method with sufficiently exact PDF should be used instead of just approximate expected values of z_t^2. Note that, by using (49) and (50), the approximate expected values for entries of z_t^2 can be simply modified into approximate expected values for entries of w_i^2. This is clear that, Claim 1 is proved in Lemma 10 just for piecewise-linear bounding function $\mathcal{R}'$ with parameter $\mathbf{a}' = 1$ (corresponding with full-enumeration). Lemma 11 introduces a sampling technique for coefficient vector z, based on Claim 1, in the way that this sampling method (Lemma 11) claims to be equivalent with rejection sampling by using (48) for coefficient vector z (while random variable $X_i = \sum_{t=1}^{i} z_t^2/R^2$ is bounded by entries of $\mathcal{R}_i'$ from bounding function $\mathcal{R}'$ in Claim 1). Our sampling method by using Claim 1 and Lemma 11 which is referred as "Our sampling method 1" would be verified in Section 4.2.1 (also this sampling method is generalized to "Our sampling method 2" in Section 4.2.2 and Section 4.2.3).

Lemma 11. *Under condition of* $\|v\|^2/R^2 \approx 1-\varepsilon$ *where* $\varepsilon \approx O(1/d)$*, for a typical GSO block* $(b_1^*, b_2^*, \ldots, b_g^*, \ldots, b_d^*)$*, and piecewise-linear bounding function* $\mathcal{R}$*, if the random variable of* $w_l^2 = z_t^2/\|b_l^*\|^2$ *would be sampled by formula (51), then the expected value for random variable* $X_i = \sum_{t=1}^{i} z_t^2/R^2$ *can be closely approximated by* $\mathcal{R}_i^2$

$$w_l^2 = \begin{cases} \dfrac{(1-\mathbf{a})\ \omega_l\ \left(\|v\|^2 - \|b_g^*\|^2\right)}{\|b_l^*\|^2 \left((1-\mathbf{a}) \sum_{t=1}^{\lfloor \frac{d}{2} \rfloor} \omega_t + \mathbf{a} \sum_{t=\lfloor \frac{d}{2} \rfloor + 1}^{g-1} \omega_t\right)}, & \text{for } 1 \le l \le \lfloor \frac{d}{2} \rfloor \\ \dfrac{\mathbf{a}\ \omega_l\ \left(\|v\|^2 - \|b_g^*\|^2\right)}{\|b_l^*\|^2 \left((1-\mathbf{a}) \sum_{t=1}^{\lfloor \frac{d}{2} \rfloor} \omega_t + \mathbf{a} \sum_{t=\lfloor \frac{d}{2} \rfloor + 1}^{g-1} \omega_t\right)}, & \text{for } \lfloor \frac{d}{2} \rfloor < l \le g-1 \\ 1, & \text{for } l = g \\ 0, & \text{for } g < l \le d \end{cases} \tag{51}$$

where $l = d - t + 1$, $\mathcal{R}$ *with parameter of* $\mathbf{a}$ *and* $\omega_i \leftarrow \mathrm{Gamma}(1/2, 2)$.

See proof in Appendix B.9.

Note: By using (47), the sign of entries of w_l in vector w from formula (51), can be set by factor of $(-1)^{\lfloor \mathrm{rand}_{[0\ldots 2)} \rfloor}$.

Lemma 11 by using Claim 1 introduces a sampling method for coefficient vector w which tries to samples the random variable of $X_i = \sum_{t=1}^{i} z_t^2/R^2$ as the same as original sampling method of (48) which is bounded by bounding function $\mathcal{R}'$. Our test results in Test 1 from Section 4.2.1 show that our proposed sampling method by Claim 1 and Lemma 11 is nearly close to the original sampling method by (48) which is bounded by bounding function $\mathcal{R}'$. Note that, the condition of $\|v\|^2/R^2 \approx 1-\varepsilon$ where $\varepsilon \approx O(1/d)$ for our proposed sampling method emphasizes that the enumeration radius should be neared to solution norm. In fact, when the radius factor of r_{FAC} is sufficiently close to 1 together with any success probability of bounding function, or when there is any radius factor of r_{FAC} with small success probability of current bounding function, this condition (the condition of $\|v\|^2/R^2 \approx 1-\varepsilon$ where $\varepsilon \approx O(1/d)$) can be observed. The pseudo-code of this sampling method can be seen in Algorithm 5 from Appendix A.5 (which implements our sampling method 1; also see Test 1 from Section 4.2.1).

In fact, our sampling method by Lemma 11 and Claim 1 is just introduced for piecewise-linear bounding functions. Algorithm 6 in Appendix A.5, which is referred as "Our sampling method 2", tries to generalize Algorithm 5 to work with any type of bounding function, any success probability and any radius factor r_{FAC}. Lines 1-6 in Algorithm 6 do this generalization by use of a simple transformation. Our test results in Test 2 and Test 3 from Section 4.2.2 and Section 4.2.3, show that the accuracy of our sampling method 2 is not acceptable in all settings and should be revised for better exactness in further studies!

3.2.4 Complementary Discussions on Coefficient Vectors

Another measurement which gives some useful information about PDF of coefficient vector z (and vector w) is median of random variable of $X_i = \sum_{t=1}^{i} z_t^2/R^2$, which is defined following lemma.

Lemma 12. *For a GSO block* $(b_1^*, b_2^*, \ldots, b_g^*, \ldots, b_d^*)$ *and enumeration radii* R*, if the random variable of* $w_l^2 = z_t^2/\|b_l^*\|^2$ *would be sampled by (47), the median of random variable* X *which is bounded by bounding function* $\mathcal{R}_i'$ *(where* $X_i = \sum_{t=1}^{i} z_t^2/R^2$*), can be approxi-*

mated by set of vectors in (52)

$$\text{Median}[X] \in \{x| \text{ for } 1 \leq i \leq d: \ x_i \leq \mathcal{R}_i^{'} \ \& \ \text{p}_{\text{succ}}(x) = \tfrac{1}{2}\ \text{p}_{\text{succ}}(\mathcal{R}^{'})\} \tag{52}$$

Proof. This proof is trivial and comes from the original definition of median measurement. □

Note: There are too many medians for random variable X which is bounded by a bounding function $\mathcal{R}^{'}$ (even if $\mathcal{R}^{'}$ is a piecewise-linear bounding function, the shape of entries in one of these medians would be in the form of a piecewise-linear bounding function with dimension d).

At the end of this section, note that the coefficient vector w originally is defined in a discrete way, based on integer vector y. There is no polynomial time method to find such integer vector y precisely, unless there is a polynomial time solver for corresponding problem of approximate-SVP! A simple sampling method of integer vector of y and corresponding discrete vector w can be defined as following remark.

Remark 5. After sampling the continuous vector w by use of (47) for $\mathcal{L}_{[1,d]}$, the integer vector of y and discrete value of entries of vector w as coefficient vector of $w^{''}$ can be redefined as following way (the sequence of operations is important in (53)).

$$[y, w^{''}] = \begin{cases} 1: y_t \leftarrow \left\lfloor w_t - \sum_{i=t+1}^{d} y_i \mu_{i,t} \right\rceil, & \text{for } t = d \text{ down to } 1 \\ 2: w_t^{''} \leftarrow y_t - \sum_{i=t+1}^{d} y_i \mu_{i,t}, & \text{for } t = d \text{ down to } 1 \end{cases} \tag{53}$$

Some propositions in this paper which use the continuous vector w in their reasoning and proofs would be affected by discrete version $w^{''}$, such as, Lemma 9, Corollary 2, Lemma 10, Lemma 11 and Claim 1. The definition in Remark 5 introduces non-exact approximations, so the definitions of w and y are introduced in a continuous way in this paper (instead of discrete ones) as follows

$$y_t \leftarrow w_t - \sum_{i=t+1}^{d} y_i \mu_{i,t}, \ \text{for } t = d \text{ down to } 1 \tag{54}$$

The pseudo-code of rejection sampling method for coefficient vector w and vector y can be studied in Algorithm 5 (referred as "Our sampling method 1") and its generalized version in Algorithm 6 (referred as "Our sampling method 2") from Appendix A.5.

3.3 Approximate Cost of Enumeration by Optimal Bounding Function

The estimation of cost for GNR-enumeration by optimal bounding function can be used to determine the best running time of attacks which use this SVP-solver (i.e., GNR-enumeration), such as BKZ algorithm, and consequently better approximation for bit-security of lattice-based cryptographic primitives against these attacks. A formal definition of optimal bounding function can be declared as follows.

Optimal bounding function. For input lattice block $\mathcal{L}_{[j,k]}$ and the enumeration radius $R \approx \|v\|$ where v is expected to be the final solution vector of GNR-enumeration with an input success probability P, the optimal bounding function $\mathcal{R}_{opt}$ with success probability P can be defined formally as following set

$$\mathcal{R}_{opt} \in \{\mathcal{R}| \ \text{p}_{\text{succ}}(\mathcal{R}) = P \ \& \ \forall \ \mathcal{R}^{'}: N(\mathcal{L}_{[j,k]}, \mathcal{R}, R) \leq N(\mathcal{L}_{[j,k]}, \mathcal{R}^{'}, R)\}, \text{ where } R \approx \|v\|. \tag{55}$$

The function of $N(\mathcal{L}_{[j,k]}, \mathcal{R}, R)$ is defined in (24).

Note: This is possible to have many solutions in the cylinder-intersection by bounding function $\mathcal{R}$, but just one of them which is the shortest one among them is the final solution and returned by GNR-enumeration (see Fact 1).

Note: In our definition of optimal bounding function, this is assumed that the enumeration radius is near to final solution norm returned by GNR-enumeration with an input success probability P, this means that, for using optimal bounding function, the enumeration radius should be forced to be $R \approx \|v\|$!

Following claim introduces an approximation for the cost of GNR-enumeration by optimal bounding function.

Claim 2. For a typical lattice block $\mathcal{L}_{[j,k]}$, the cost of GNR-enumeration which is pruned by optimal bounding function $\mathcal{R}_{opt}$ with static success probability P which is defined in (55), can be approximated by the cost of GNR-enumeration pruned by a bounding function $\mathcal{R}1$ whose entries are defined by expected value of random variable of $X_i = \sum_{t=1}^{i} z_t^2/R^2$ (i.e., $\mathcal{R}1[i] = \text{E}[X_i]$) corresponding with final solution vectors returned by a GNR-enumeration pruned by arbitrary bounding function $\mathcal{R}2$ with static success probability P.

By using Claim 2, the cost of enumeration by optimal bounding function can be approximated by using a bounding function whose entries are equal to the expected value of samples of $X_i = \sum_{t=1}^{i} z_t^2/R^2$ in Our sampling method 1 and Our sampling method 2 (and these expected values of samples X_i refer to $\mathcal{R}_{\mathbf{a}_3}$ with piecewise-linear parameter of $\mathbf{a}_3$ in Algorithm 5 and Algorithm 6). By using following approximations, the reasoning behind Claim 2 would be clear more:

(1) Since Claim 2 is declared based on (55), this

is forced that $R \approx \|v\|$, and the static success probability of enumeration as P is nearly equivalent to dynamic success frequency f_{succ}.

(2) For a typical lattice block $\mathcal{L}_{[j,k]}$, if an enumeration by radius $R \approx \|v\|$ and an arbitrary bounding function $\mathcal{R}2$ with success probability P is applied on that block and returns the final solution vector v with coefficient vector of z and value of $X_i = \sum_{t=1}^{i} z_t^2/R^2$, then the best estimation for optimal bounding function $\mathcal{R}_{opt}$ for that block and success probability P can be equal to $\mathcal{R}_{opt} = X_i + \varepsilon$ (which returns final solution vector v again). Unfortunately this is not possible to find the exact final solution of an enumeration with success probability P simply! So this is needed to use the approximate expected value of $\mathrm{E}[X_i] = \mathrm{E}\left[\sum_{t=1}^{i} z_t^2/R^2\right]$ for each input lattice block (in fact, this is assumed that the variance of $X_i = \sum_{t=1}^{i} z_t^2/R^2$ is near to 0 which is considered as a small approximation gap from best estimation of optimal bounding function in this reasoning).

(3) The probability distribution and expected value of $X_i = \sum_{t=1}^{i} z_t^2/R^2$ for different bounding functions $\mathcal{R}2$ (in Claim 2) with same success probability P is not similar in all settings (as shown in Figure 5 for three different bounding functions), however this is considered as another small approximation gap from best estimation for optimal bounding function in this reasoning.

4 Results for Our Sampling Methods

In this section, sufficient experimental/simulation results are introduced which try to verify our proposed sampling methods of enumeration solution. All the tests in this section are performed on the random instances of SVP lattice challenges [17, 18] and Darmstadt lattice challenges [20, 21].

4.1 Results for Probability Distribution of g

To exhibit the statistical features of parameter of g as last non-zero index in coefficient vectors of w and y, which is defined in Theorem 2, we introduce some experimental tests by actual running of $\text{BKZ}_{\beta=45}$ over some random lattice bases with dimension 100 and 200, in different enumeration radius. Also, just the successful enumerations are considered in results of this test. By using excessive number of enumeration successes in this test, the statistical parameters related to g are shown in Table 1. Table 1 includes following parameters

- The parameter of $\mathrm{E}[g]$ represents the mean value of parameter of g.
- The parameter of SD.$[g]$ represents the standard deviation of parameter of g;
- The parameter of $\mathrm{E}[r_{\text{FAC}}]$ represents the mean value of r_{FAC}.
- The parameter of E [ZeroCount] represents the mean value of number of zero entries in integer coefficient vector y.
- The parameter of E [NoneZeroCount] represents the mean value of number of non-zero entries in integer coefficient vector y.
- The parameter of $\mathrm{E}[|\text{NonZero\_}y_i|]$ represents the mean value for absolute values of non-zero entries in integer coefficient vector y.
- The parameter of $\mathrm{E}[|w_i|]$ represents the mean value for absolute values of the entries in coefficient vector w.

Furthermore, Figure 1 shows the relative frequency distribution of g for local blocks with size of 45 in this test (corresponding with Table 1), and tries to make better sense about our proposed probability distribution of g which is formulated in Lemma 8.

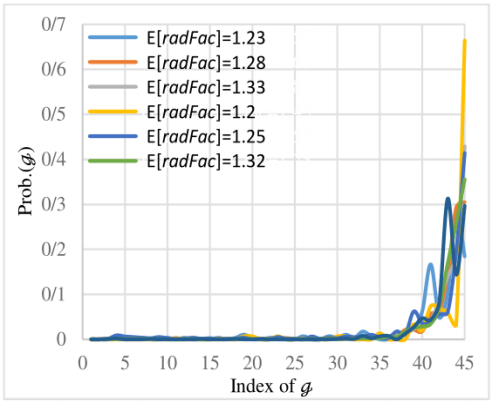


Figure 1. The probability of vector index i in blocks size of 45 to be set as g by experimental running of GNR-enumeration

Our experimental results in this test just make better sense of coefficient vectors and last non-zero index g, since for statistical analysis of coefficient vectors of w, z, y and index of g, this is needed to test sufficiently big block sizes. By using parameter of $\mathrm{E}[g]$, SD.$[g]$, E [ZeroCount], E [NoneZeroCount] and $\mathrm{E}[|\text{NonZero\_}y_i|]$, we try to determine the probability of that an integer coefficient y_i is zero or not, and consequently define a close approximation for probability of $g = i$, but the limitations of this test together with actual complexity of probability distribution for g avoid us for introducing this approximation. In fact, Lemma 8 shows that, probability of $g = i$ depends on different parameters, such as block size of β, enumeration radius factor of r_{FAC}, GSO norms of basis as $(\|b_1^*\|, \|b_2^*\|, \ldots, \|b_\beta^*\|)$. By using parameters

Table 1. Experimental results for g in successful enumerations on blocks of $\text{BKZ}_{\beta=45}$ over random lattices with dim. of 100 and 200

E $[r_{\text{FAC}}]$	E $[g]$	SD.$[g]$	E [ZeroCount]	E [NoneZeroCount]	E $[\lvert\text{NonZero\_}y_i\rvert]$	E $[\lvert w_i\rvert]$
1.2	43.23	4.03	16.93	28.07	1.59	0.25
1.23	40.67	7.2	19.48	25.52	1.61	0.27
1.25	40.67	8.39	21.46	23.54	1.51	0.27
1.28	42.03	6.14	19.41	25.59	1.51	0.27
1.32	42.96	4.04	18.38	26.62	1.48	0.28
1.33	43.05	3.92	18.17	26.83	1.5	0.29

of E $[g]$ and SD.$[g]$ together with Figure 1, an approximate view of the probability distribution of (44) in Lemma 8 can be sensed. Finally, a simple comparison of formula (44) in Lemma 8 as "prob_g2" (with parameters of $\beta = 60$, $r_{\text{FAC}} \approx 1.063$ and GSO norms of random/LLL-reduced/HKZ-reduced bases) and formula (42) as "prob_g1" (with parameters of $p = 0.5$, $p = 0.6$, $p = 0.7$, $p = 0.8$ and $p = 0.9$) is introduced in Figure 2.

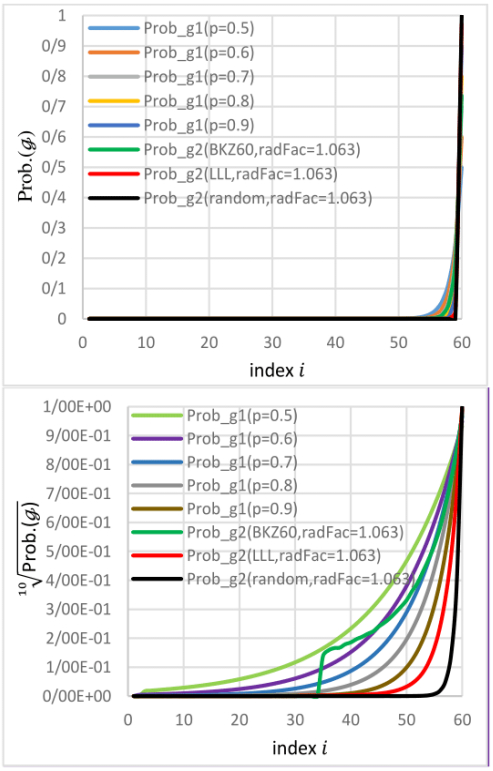


Figure 2. The probability of vector index i in blocks size of 60 to be set as g by using formula (42) and (44)

4.2 Results for Sampling Coefficient Vectors of y, w and z

In this section, three tests are introduced which generalize our idea in this paper for sampling coefficient vectors of w and z in BKZ-simulation with GNR-pruning. For these three tests, the mean value of test

Table 2. Mean value of specifications from 20 random bases in the sense of Darmstadt lattice challenge with dimension of 200

Features	Value
Block size β	200
Squared GSO norm of first vector $\lVert b_1^* \rVert^2$	12795.4
Squared GSO norm of g-th vector $\lVert b_g^* \rVert^2$	0.000692
Gaussian heuristic of block $\text{GH}(\mathcal{L})^2$	33.5495
Quality of input block (reduction quality)	$\text{LLL}_{\delta \approx 0.99}$
Root-Hermite factor of input block	≈ 1.02123
q-factor of input block	≈ 1.043
Number of bases	Up to 20

specifications from 20 random bases in the sense of Darmstadt lattice challenge [20, 21] with dimension of 200 is used (see these specifications in Table 2).

Note: Since this is expected to $g \approx \text{CUT} \approx d$ by Lemma 7, Lemma 8 and Corollary 1, all the tests in this section for simplicity use $g = \text{CUT} = d$.

Note: To compute the norm of solution $\lVert v \rVert$ in all samples of this test, the sampling method of (37) is used.

Note: The bounding function $\mathcal{R} = (\mathcal{R}_1, \ldots, \mathcal{R}_d)$ in all figures of this test is scaled to the vector of $\mathcal{R}^2 R^2 = (\mathcal{R}_1^2 R^2, \ldots, \mathcal{R}_d^2 R^2)$, and consequently the samples of $R^2 X_i = \sum_{t=1}^{i} z_t^2$ are upper bounded by entries of $\mathcal{R}^2 R^2$ (where z_t^2 are squared entries of coefficient vector z).

4.2.1 Test 1 for "Our sampling method 1"

In this test, our sampling method 1 for coefficient vectors of w and z which is introduced in Claim 1 and Lemma 11, would be verified by sufficient number of results. To verify Claim 1, this test is divided in to four parts. Each part compares our sampling method 1 (by Claim 1 and Lemma 11) against the original sampling method by (47) and (48). These four parts of test differs from each other in enumeration radius factors (parameter of r_{FAC}) and piecewise-linear parameter of **a**. The bounding function is labeled by "R_picewise

[$\mathbf{a} = c$]" which means a piecewise-linear bounding function with parameter $\mathbf{a} = c$, and "R_Full [$\mathrm{p_{succ}} = 1$]" which means full-enumeration. This test is done in following way

- The original sampling in this test uses formula of (48) to sample coefficient vector z or formula (47) to sample coefficient vector w, then it selects just the sampled vector of z satisfying the bounding function constraints in (25). The mean value of entries of selected (successful) samples of vector z is used to compute $X_i = \sum_{t=1}^{i} z_t^2/R^2$ as "Original sampling of X_i". Also the trend line equation for "Original sampling of X_i" is presented for each part of test.
- Our sampling method which is labeled as "Our sampling 1 of X_i", uses Claim 1 which says that for each piecewise-linear bounding function $\mathcal{R}_{\mathbf{a}}$ (labeled by "R_picewise [$\mathbf{a} = c$]" or "R_Full [$\mathrm{p_{succ}} = 1$]"), this is only needed to find a piecewise-linear bounding function $\mathcal{R}_{\mathbf{a}'}$ where $\mathrm{p_{succ}}(\mathcal{R}_{\mathbf{a}'}) = \frac{\mathrm{p_{succ}}(\mathcal{R}_{\mathbf{a}})}{\mathrm{F}(d)}$ by function $\mathrm{p_{succ}}$ defined in (17) and function F defined in (18), and consequently we use the entries of $\mathcal{R}_{\mathbf{a}'}$ as the approximate expected values of sampling of $X_i = \sum_{t=1}^{i} z_t^2/R^2$ while bounded by $\mathcal{R}_{\mathbf{a}}$. The parameter $\mathbf{a}'$ from bounding function $\mathcal{R}_{\mathbf{a}'}$ can be used by Lemma 11 to sample $w_t^2 = z_{d-t+1}^2/\|b_t^*\|^2$ and consequently to sample the values of $R^2X_i = \sum_{t=1}^{i} z_t^2$ which is bounded by $\mathcal{R}_{\mathbf{a}}^2R^2$.

Our results for four parts of this test can be observed in Figure 3 and Table 3. Each part uses up to 4×10^8 samples.

Note: In Figure 3, Figure 4 and Figure 5, the label of a typical bounding function $\mathcal{R}$ represents the vector $\mathcal{R}^2R^2 = [\mathcal{R}_1^2R^2, \ldots, \mathcal{R}_d^2R^2]$, also the label of sampled vector X represents the vector R^2X with entries of $R^2X_i = \sum_{t=1}^{i} z_t^2$.

As shown in Figure 3, the original sampling and our sampling method 1 of $R^2X_i = \sum_{t=1}^{i} z_t^2$ are upper-bounded by bounding function of "R_picewise [$\mathbf{a} = c$]" (see the blue dash line as the bounding function).

The parameter $\mathbf{a}$ in Table 3 shows the piecewise-linear parameter of bounding function of $\mathcal{R}_{\mathbf{a}}$. The static success probability and dynamic success frequency of $\mathcal{R}_{\mathbf{a}}$ by (26) and (27) is shown in column 3 and 4. The parameter $\mathbf{a}'$ shows the input parameter of Lemma 11 which computed by Claim 1 as $\mathrm{p_{succ}}(\mathcal{R}_{\mathbf{a}'}) = \mathrm{p_{succ}}(\mathcal{R}_{\mathbf{a}})/\mathrm{F}(d)$ by function $\mathrm{p_{succ}}$ defined in (17) and function F defined in (18) (parameter $\mathbf{a}'$ in Table 3 corresponds with parameter $\mathbf{a}_3$ in Algorithm 5 from Appendix A.5). The expected number of solutions in the cylinder-intersection of radius $(\mathcal{R}_1'R^2, \ldots, \mathcal{R}_l'R^2)$ which is defined by $f_{succ}^{new0}(\mathcal{R}_{\mathbf{a}})$ in (27) is approximated by column 4 in Table 3. In fact, as shown in column 9 and column 10, the condition of $\|v\|^2/R^2 \approx 1 - \varepsilon$ where $\varepsilon \approx O(1/d)$ is satisfied in this test, so Claim 1, Lemma 10 and Lemma 11 can be applied consistently. As shown in Figure 3, our sampling method 1 by Claim 1 and Lemma 11 can be used as an approximation of original sampling by (47), (48) and (25), however as the piecewise-linear parameter of $\mathbf{a}$ nears to 0, our sampling method 1 would be less precise.

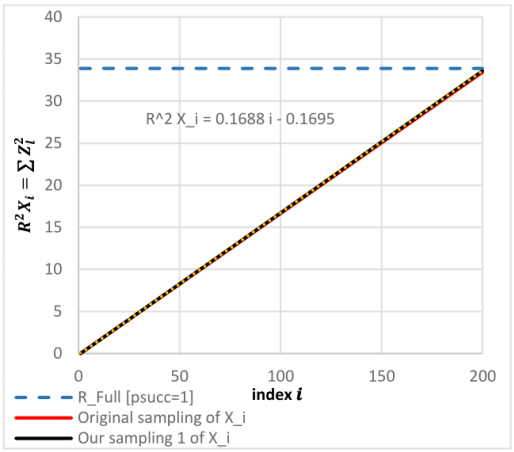
Note: As mentioned, the success probability $\mathrm{p_{succ}}$, which is defined in (17), is equal to static success probability p_{succ}^{new0} which is defined in (26).

Note: If the radius factor r_{FAC} is sufficiently small together with any static success probability of bounding function or the static success probability of current bounding function is small together with any radius factor r_{FAC}, then the condition of $\|v\|^2/R^2 \approx 1 - \varepsilon$ where $\varepsilon \approx O(1/d)$ can be observed. So our proposed sampling method 1 (based on Claim 1 and Lemma 11) just be applied for small radius factor or small static success probability! In other words, if dynamic success frequency $f_{succ}^{new0}(\mathcal{R}_{\mathbf{a}})$ belongs to $\approx O(1)$, the condition of $\|v\|^2/R^2 \approx 1 - \varepsilon$ where $\varepsilon \approx O(1/d)$ can be observed.

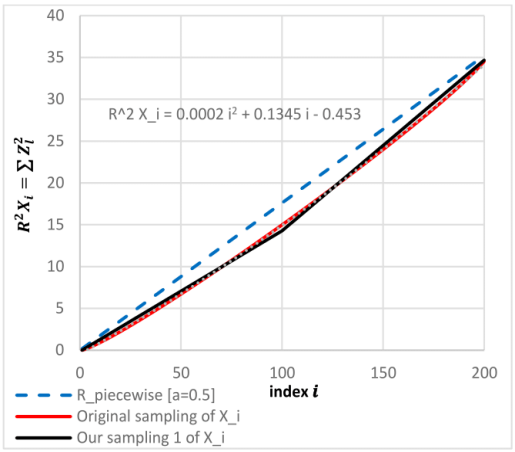
4.2.2 Test 2 for "Our sampling method 2"

As mentioned, Claim 1, Lemma 10 and Lemma 11 are introduced under condition of $\|v\|^2/R^2 \approx 1 - \varepsilon$ where $\varepsilon \approx O(1/d)$. This test focuses on the situations that ε would be close to ≈ 1, so this is not possible to use Claim 1 and Lemma 11 directly to sample the coefficient vector of z. For this case, Algorithm 6 in Appendix A.5 generalizes our sampling method 1 (in Algorithm 5 from Appendix A.5). Lines 1-6 in Algorithm 6 do this generalization by use of a simple transformation. This new sampling method is labeled here as "Our sampling method 2 of X_i". Our results for this test would be observed in Figure 4 and Table 4. The trend line equation for "Original sampling of X_i" is presented for each part of this test in Figure 4.

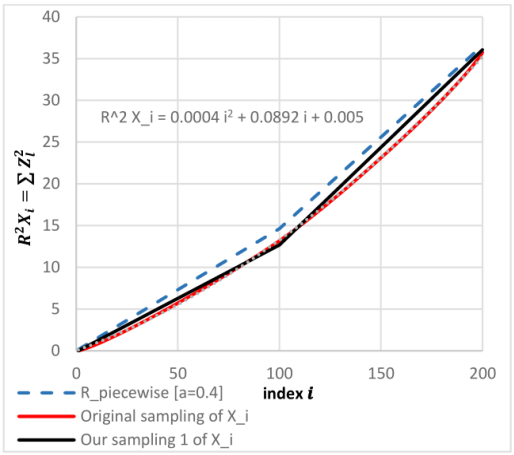
As shown in Figure 4, sampling method 2 of X_i (by using lines 1-6 in Algorithm 6) introduces acceptable approximation for sampling coefficient vector z. Note that, by using these observation, sampling method 2 can be used for sampling coefficient vector z and piecewise-linear bounding function with any enumeration radius.



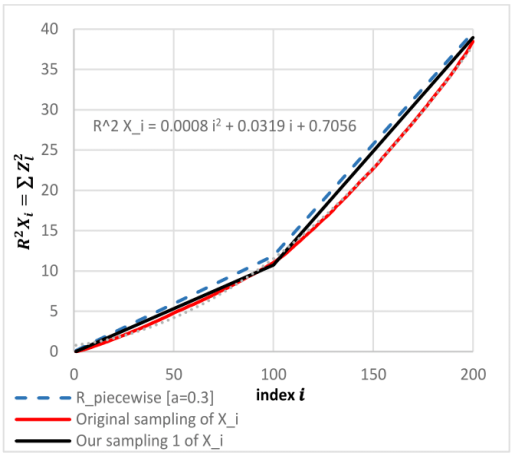
(a) Using full-enumeration (corresponding with $\mathbf{a} = 1$) and radius factor of ${r_{\text{FAC}}}^2 = 1.01$



(b) Using piecewise-linear with parameter of $\mathbf{a} = 0.5$ and radius factor of ${r_{\text{FAC}}}^2 = 1.05$



(c) Using piecewise-linear with parameter of $\mathbf{a} = 0.4$ and radius factor of ${r_{\text{FAC}}}^2 = 1.09$



(d) Using piecewise-linear with parameter of $\mathbf{a} = 0.3$ and radius factor of ${r_{\text{FAC}}}^2 = 1.18$

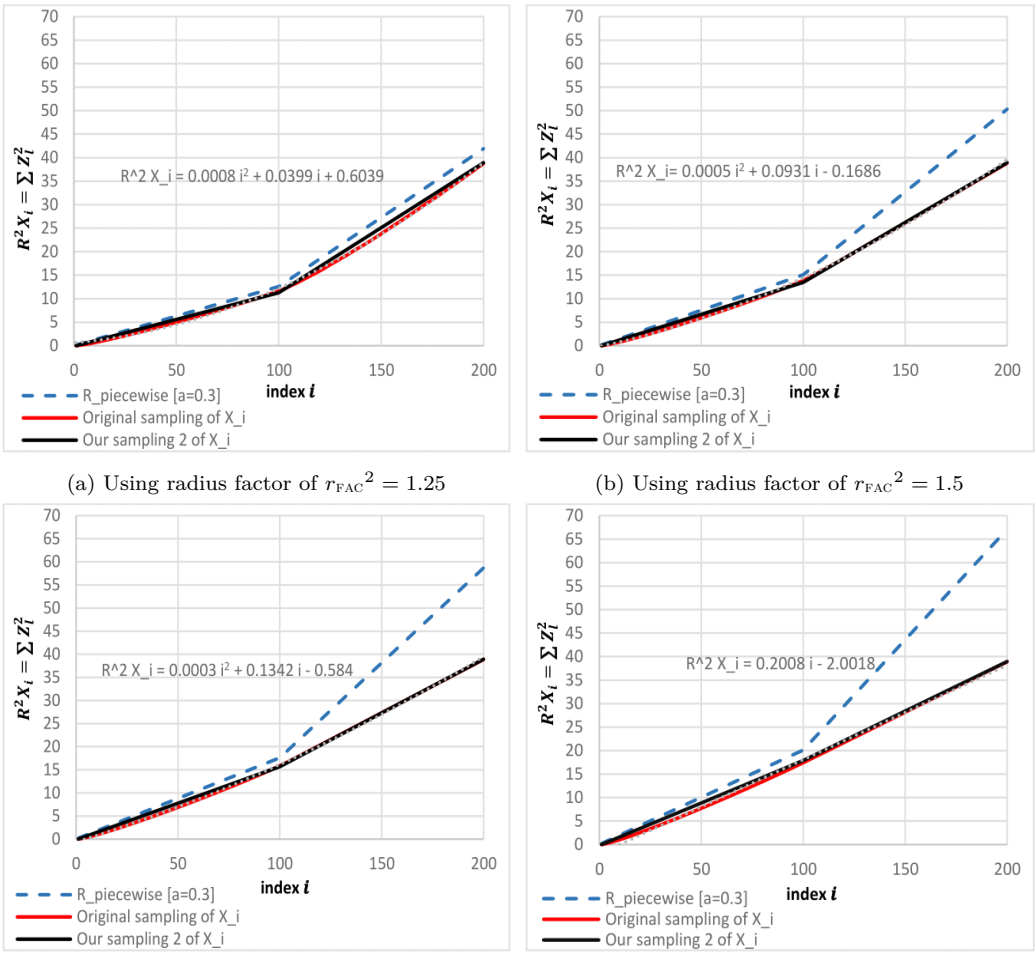
Figure 3. Comparison of original sampling method by (25), (47) and (48) with our sampling method 1 by Claim 1 and Lemma 11

Table 3. Configuration parameters for running Test 1

Test	Param. of $\mathbf{a}$	$p_{succ}^{new0}(\mathcal{R}_{\mathbf{a}})$ by (26)	$f_{succ}^{new0}(\mathcal{R}_{\mathbf{a}})$ by (27)	Param. of $\mathbf{a}'$	$\text{p}_{\text{succ}}(\mathcal{R}_{\mathbf{a}'})$ by (17)	Solution Norm $\|v\|^2$	$\frac{\|v\|^2}{\text{GH}(\mathcal{L})^2}$	$\frac{\|v\|^2}{R^2} = 1-\varepsilon$	Expected value of ε	Radius factor ${r_{\text{FAC}}}^2$	Enum. radius R^2
(1)	Full	1	1.35	0.5	0.0175	33.4313	0.9965	0.987	≈1/77	1.01	33.885
(2)	0.5	0.0175	1.15	0.4134	5.926×10^{-4}	34.54	1.0295	0.98	≈1/50	1.05	35.227
(3)	0.4	2.856×10^{-4}	0.8	0.3524	1.213×10^{-5}	35.742	1.0654	0.977	≈1/44	1.09	36.569
(4)	0.3	1.09×10^{-7}	0.84	0.276	7.407×10^{-9}	38.399	1.1445	0.97	≈1/34	1.18	39.5

Note: By our results in Figure 3 and Figure 4, as the index of i would be neared to d, the distance between the samples of $R^2X_i = \sum_{t=1}^{i} z_t^2$ by original sampling and the samples by our sampling method 2 would be increased (i.e., the accuracy of our sampling method would be less), while as the value of $\|v\|^2/R^2$ would be decreased from $1 - 1/d$ down to 0 (i.e., the value of ε in column 10 from Table 4 would be neared to 1), this distance can be decreased sharply (i.e., the accuracy of our sampling method would be more) and if the piecewise-linear parameter of $\mathbf{a}$ nears to 0, this distance can be increased (i.e., the accuracy of our sampling method would be less).

The last entry of samples as $R^2X_d = \sum_{t=1}^{d} z_t^2$ (by original sampling and our sampling) is equal to $\|v\|^2$, while last entry of scaled bounding function $\mathcal{R}^2R^2$ is equal to R^2. As mentioned in this test (against



(a) Using radius factor of $r_{\text{FAC}}{}^2 = 1.25$

(b) Using radius factor of $r_{\text{FAC}}{}^2 = 1.5$

(c) Using radius factor of $r_{\text{FAC}}{}^2 = 1.75$

(d) Using radius factor of $r_{\text{FAC}}{}^2 = 2$

Figure 4. Comparison of original sampling method by (25), (47) and (48) with our sampling method 2

Test 1), $\|v\|^2$ is not near to R^2, so the condition of $\|v\|^2/R^2 \approx 1-\varepsilon$ where $\varepsilon \approx O(1/d)$ is violated and this is shown in column 9 and column 10 in Table 4.

Note: The parameter $\mathbf{a}'$ in Table 4 corresponds with parameter $\mathbf{a}_3$ in Algorithm 6 from Appendix A.5.

4.2.3 Test 3 for "Our sampling method 2"

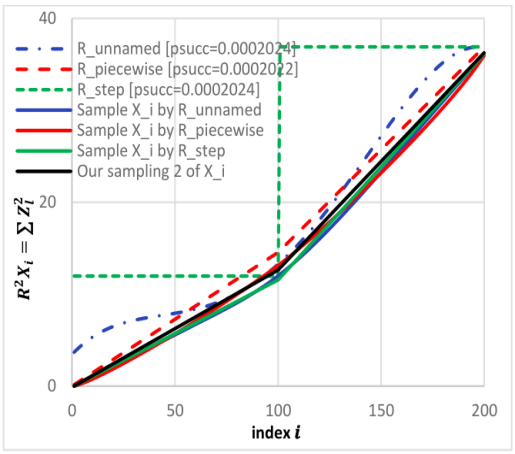
Our proposed sampling method 1 in Test 1 and our sampling method 2 in Test 2 only focuses on the piecewise-linear bounding function, but there are many forms of bounding functions which should be discussed. In this test, our sampling method 2 is used for comparing sampled coefficient vectors of z which is bounded by a piecewise-linear bounding function, a step bounding function and an unnamed bounding function (which is generated by authors just for this test). In fact, this test tries to show that sampling method 2 by Algorithm 6 in Appendix A.5 can be generalized to any bounding function with an input success probability P. Test 3 uses four parts whose samples of $R^2X_i = \sum_{t=1}^{i} z_t^2$ are compared in these three types of bounding function (with same success probability). The success probability of bounding function at each part of this test is different from each other. Figure 5 and Table 5 show our results of this test.

As shown in Figure 5, sampling method 2 of X_i introduces an approximation for sampling coefficient vector z under each of these three types of bounding function (however this is not too precise). Line 1 in Algorithm 6 applies this generalization by transforming any bounding function to a piecewise-linear bounding function with same success probability. In fact, this test should be studied for more types of bounding function with different success probabilities in further studies. Table 5 shows complementary information in Test 3 for comparison of original sampling method with our generalized sampling method 2 for any type of bounding function.

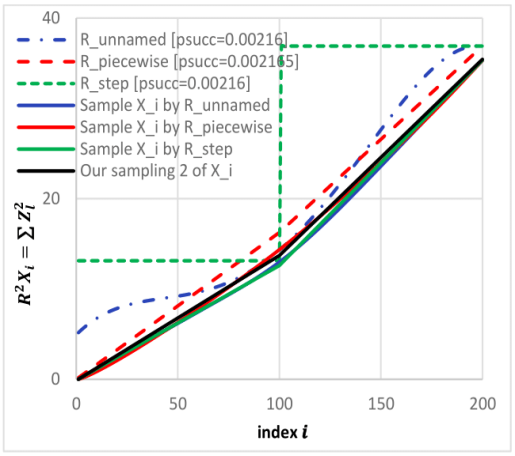
The parameter of $p_{succ}^{new0}(\mathcal{R})$ by (26) in Table 5 shows the static success probability for each of these three

Table 4. Configuration parameters for running Test 2

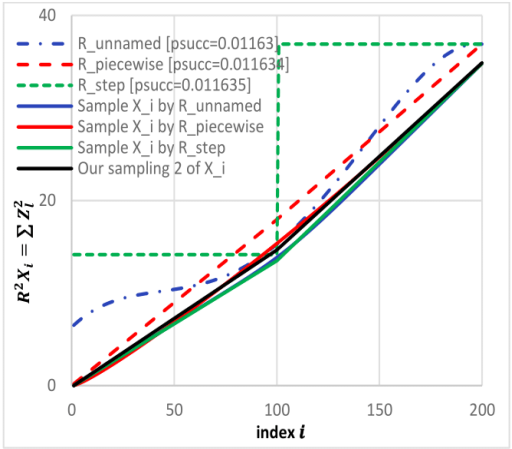
Test	Param. of $\mathbf{a}$	$p_{succ}^{new0}(\mathcal{R}_{\mathbf{a}})$ by (26)	$f_{succ}^{new0}(\mathcal{R}_{\mathbf{a}})$ by (27)	Param. of $\mathbf{a}'$	$\mathrm{p_{succ}}(\mathcal{R}_{\mathbf{a}'})$ by (17)	Solution Norm $\|v\|^2$	$\frac{\|v\|^2}{\mathrm{GH}(\mathcal{L})^2}$	$\frac{\|v\|^2}{R^2} = 1-\varepsilon$	Expected value of ε	Radius factor ${r_{\mathrm{FAC}}}^2$	Enum. radius R^2
(1)	0.3	1.091×10^{-7}	267.8	0.2897	3.58×10^{-8}	38.7656	1.1555	0.9244	$\approx 1/13$	1.25	41.937
(2)	0.3	1.091×10^{-7}	2.22×10^{10}	0.3469	7.92×10^{-5}	38.84	1.1577	0.7718	$\approx 1/4$	1.5	50.324
(3)	0.3	1.091×10^{-7}	1.09×10^{17}	0.4043	3.63×10^{-4}	38.886	1.159	0.6623	$\approx 1/3$	1.75	58.712
(4)	0.3	1.091×10^{-7}	6.91×10^{22}	0.4617	5.05×10^{-3}	38.917	1.16	0.58	$\approx 1/2$	2	67.099



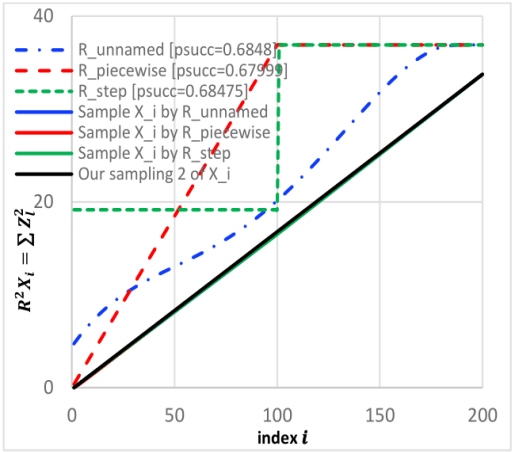
(a) Using three types of bounding functions with success probability ≈ 0.0002022 and radius factor of ${r_{\mathrm{FAC}}}^2 = 1.1$



(b) Using three types of bounding functions with success probability ≈ 0.00216 and radius factor of ${r_{\mathrm{FAC}}}^2 = 1.1$



(c) Using three types of bounding functions with success probability ≈ 0.01163 and radius factor of ${r_{\mathrm{FAC}}}^2 = 1.1$



(d) Using three types of bounding functions with success probability ≈ 0.6848 and radius factor of ${r_{\mathrm{FAC}}}^2 = 1.1$

Figure 5. Comparison of original sampling method by (25), (47) and (48) with our sampling method 2 generalized for three types of bounding function

bounding functions, also the parameter of $f_{succ}^{new0}(\mathcal{R})$ by (27) in Table 5 shows the dynamic success frequency (or expected number of solutions in GNR enumeration) by these three bounding functions. The parameter $\mathbf{a}'$ in Table 5 corresponds with input parameter of $\mathbf{a}$ in Lemma 11 (and corresponds with parameter $\mathbf{a}_3$ in Algorithm 6 from Appendix A.5). Moreover, column 6 shows the mean value of sampled solution norm of enumerations by these three bounding functions.

5 Conclusions and Further Works

BKZ algorithm has a determinative role in security analysis of lattice-based cryptographic primitives,

Table 5. Configuration parameters for running Test 3

Test	$p_{succ}^{new0}(\mathcal{R})$ by (26)	$f_{succ}^{new0}(\mathcal{R}_{\mathbf{a}})$ by (27)	Param. of $\mathbf{a}'$	$\mathrm{p_{succ}}(\mathcal{R}_{\mathbf{a}'})$ by (17)	Average of Solution Norm $\|v\|^2$	Average of $\frac{\|v\|^2}{\mathrm{GH}(\mathcal{L})^2}$	Average of $\frac{\|v\|^2}{R^2} = 1-\varepsilon$	Expected value of ε	Radius factor ${r_{\mathrm{FAC}}}^2$	Enum. radius R^2
(1)	0.0002024	1.39	0.349	9.46×10^{-6}	36.085	1.0756	0.9778	$\approx 1/45$	1.1	36.904
(2)	0.00216	14.88	0.388	0.0001424	35.3452	1.0535	0.95775	$\approx 1/24$	1.1	36.904
(3)	0.01163	80.13	0.422	0.00091	34.824	1.037	0.94363	$\approx 1/18$	1.1	36.904
(4)	0.6848	4718.48	0.5	0.0175	33.73967	1.00567	0.91424	$\approx 1/12$	1.1	36.904

therefore the total cost and output quality of BKZ algorithm should be computed exactly to be used in parameter selection of these primitives. Although the exact manner of BKZ algorithm with small block sizes can be studied by practical running of BKZ, this manner for higher block sizes (e.g., $\beta \geq 100$) should be simulated precisely. Designing a BKZ-simulation with GNR-pruned enumeration needs to some necessary building-blocks which includes enumeration radius, generation of bounding function, estimation of success probability, LLL simulation, estimation of GNR enumeration cost, sampling method for enumeration solution, simulation of updating GSO. This paper tries to introduce an efficient and exact sampling method for enumeration solution v which samples both norm and coefficient vectors of enumeration solution.

To the best of our knowledge, no sampling method for norm of GNR-pruned enumeration with any success probability are introduced. The paper [3] introduces a sampling method for norm of enumeration (see line 14 of Algorithm 4 from paper [3]) by use of the probability distribution of solution norm which is stated in Chen's thesis [11], but that sampling method is defined just for full-enumeration, not any GNR pruning (see Theorem 1 in Section 2.7). Also, paper [4] uses an exact and efficient way to estimate the expected value of the norm of solution norm, instead of sampling this norm (see the formula (32)). Moreover, paper [2] uses only the non-exact estimation of $\mathrm{GH}(\mathcal{L})$ by relation (7) as the expected norm of enumeration solution. In other side, this paper introduces a simple and efficient sampling method for norm of GNR pruned enumeration solution including bounding functions with any success probability in Lemma 3.

GNR enumeration function in BKZ algorithm over a lattice block $\mathcal{L}_{[b_j,\ldots,b_k]}$ returns the coefficient vector y, which can be used to compute the solution vector v by linear combination of this lattice block vectors as $v = \sum_{l=j}^{k} y_l b_l$. To the best of our knowledge, no precise and explicit analysis of sampling the coefficient vector y is considered in former BKZ-simulations. The structure and probability distribution of coefficient vectors corresponding with enumeration solution vector v (i.e., coefficient vectors of z, w and y; see Section 3.2) are discussed deeply in this paper. Consequently, the sampling methods of these vectors are introduced approximately, while no such a deep design of sampling these coefficient vectors are considered in former BKZ-simulations. Precisely, our sampling method for coefficient vectors is developed in three versions, as follows

- Our sampling method 1 (in Algorithm 5) is defined just for piecewise-linear bounding functions with condition of small enumeration radius or small (static) success probability, in other words, the condition that the expected number of solutions in GNR enumeration (i.e., dynamic success frequency) belongs to $\approx O(1)$; This sampling method is verified in Section 4.2.1 (however, for much small success probability, this version is not too accurate).
- Our sampling method 2 (in Algorithm 6) is defined just for piecewise-linear bounding functions with any enumeration radius and any success probability; This sampling method is verified in Section 4.2.2 (however, for enumeration radius of $R \approx \|v\|$ and much small success probability, this version is not too accurate).
- Our sampling method 3 (in Algorithm 6) is adapted for any bounding function with any enumeration radius and any success probability (i.e., with no constraint); Our test results for this sampling method in Section 4.2.3 show that the probability distribution (and expected value) of $X_i = \sum_{t=1}^{i} z_t^2/R^2$ for different bounding functions with same success probability P is not sufficiently similar, unless for big dynamic success frequency (i.e., $f_{\mathrm{succ}} \gg 1$).

Also by using the analysis on enumeration solution norms and coefficient vectors, this paper proposes an approximation for cost of enumerations by optimal bounding functions in Section 3.3. However this paper focuses on BKZ-simulation with GNR-enumeration (as SVP-solver), other SVP-solvers can be considered,

such as lattice enumeration with discrete pruning [22], sieving algorithm (e.g., the variant in [23]), enumeration by integrating sparse orthogonalized integer representations for shortest vectors [24], or even our evolutionary search for solving SVP [25]. Also, although Test 3 in Section 4.2.3 tries to show that sampling method 2 by Algorithm 6 in Appendix A.5 can be generalized to any bounding function, new techniques can be studied in further works to introduce more accurate approximation of sampling coefficient vectors of z, w, and y.

Acknowledgment

The authors would like to thank the anonymous reviewers for their much valuable, constructive and insightful comments which helped us to improve the presentation of this work significantly, also thank the editorial board and editorial staff of the ISeCure journal for their generous help in publishing this study.

References

[1] Nicolas Gama, Phong Q Nguyen, and Oded Regev. Lattice enumeration using extreme pruning. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 257–278. Springer, 2010.

[2] Yuanmi Chen and Phong Q Nguyen. BKZ 2.0: Better lattice security estimates. In *International Conference on the Theory and Application of Cryptology and Information Security*, pages 1–20. Springer, 2011.

[3] Shi Bai, Damien Stehlé, and Weiqiang Wen. Measuring, simulating and exploiting the head concavity phenomenon in BKZ. In *International Conference on the Theory and Application of Cryptology and Information Security*, pages 369–404. Springer, 2018.

[4] Yoshinori Aono, Yuntao Wang, Takuya Hayashi, and Tsuyoshi Takagi. Improved progressive BKZ algorithms and their precise cost estimation by sharp simulator. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 789–819. Springer, 2016.

[5] Joop van de Pol and Nigel P Smart. Estimating key sizes for high dimensional lattice-based systems. In *IMA International Conference on Cryptography and Coding*, pages 290–303. Springer, 2013.

[6] Tancrede Lepoint and Michael Naehrig. A comparison of the homomorphic encryption schemes FV and YASHE. In *International Conference on Cryptology in Africa*, pages 318–335. Springer, 2014.

[7] Mingjie Liu and Phong Q Nguyen. Solving BDD by enumeration: An update. In *Cryptographers' Track at the RSA Conference*, pages 293–309. Springer, 2013.

[8] Yoshinori Aono, Xavier Boyen, Lihua Wang, et al. Key-private proxy re-encryption under LWE. In *International Conference on Cryptology in India*, pages 1–18. Springer, 2013.

[9] Martin R Albrecht, Benjamin R Curtis, Amit Deo, Alex Davidson, Rachel Player, Eamonn W Postlethwaite, Fernando Virdia, and Thomas Wunderer. Estimate all the {LWE, NTRU} schemes! In *International Conference on Security and Cryptography for Networks*, pages 351–367. Springer, 2018.

[10] Jeff Hoffstein, Jill Pipher, John M Schanck, Joseph H Silverman, and William Whyte. Practical signatures from the partial fourier recovery problem. In *International Conference on Applied Cryptography and Network Security*, pages 476–493. Springer, 2014.

[11] Yuanmi Chen. *Réduction de réseau et sécurité concrete du chiffrement completement homomorphe*. PhD thesis, Paris 7, 2013.

[12] Daniele Micciancio and Oded Regev. Lattice-based cryptography. In *Post-quantum cryptography*, pages 147–191. Springer, 2009.

[13] Claus Peter Schnorr. Lattice reduction by random sampling and birthday methods. In *Annual Symposium on Theoretical Aspects of Computer Science*, pages 145–156. Springer, 2003.

[14] Arjen K Lenstra, Hendrik Willem Lenstra, and László Lovász. Factoring polynomials with rational coefficients. *Mathematische annalen*, 261(ARTICLE):515–534, 1982.

[15] CA Rogers. The number of lattice points in a set. *Proceedings of the London Mathematical Society*, 3(2):305–320, 1956.

[16] Johannes Buchmann and Christoph Ludwig. Practical lattice basis sampling reduction. In *International Algorithmic Number Theory Symposium*, pages 222–237. Springer, 2006.

[17] SVP Challenge. Available at: https://www.latticechallenge.org/svp-challenge/index.php.

[18] Daniel Goldstein and Andrew Mayer. On the equidistribution of Hecke points. In *Forum Mathematicum*, volume 15, 2003.

[19] Yang Yu and Léo Ducas. Second order statistical behavior of LLL and BKZ. In *International Conference on Selected Areas in Cryptography*, pages 3–22. Springer, 2017.

[20] Johannes Buchmann, Richard Lindner, and Markus Rückert. Explicit hard instances of the shortest vector problem. In *International Workshop on Post-Quantum Cryptography*, pages 79–94. Springer, 2008.

[21] TU Darmstadt lattice challenge. Available at: www.latticechallenge.org.
[22] Yoshinori Aono and Phong Q Nguyên. Random sampling revisited: lattice enumeration with discrete pruning. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 65–102. Springer, 2017.
[23] Léo Ducas. Shortest vector from lattice sieving: a few dimensions for free. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 125–145. Springer, 2018.
[24] Zhongxiang Zheng, Xiaoyun Wang, Guangwu Xu, and Yang Yu. Orthogonalized lattice enumeration for solving SVP. *Science China Information Sciences*, 61(3):1–15, 2018.
[25] Gholam Reza Moghissi and Ali Payandeh. A parallel evolutionary search for shortest vector problem. *Int. J. Inf. Technol. Comput. Sci.(IJITCS)*, 11(8):9–19, 2019.
[26] Joop van de Pol. Lattice-based cryptography. *Eindhoven University of Technology, Department of Mathematics and Computer Science*, 2011.
[27] Johan Ludwig William Valdemar Jensen et al. Sur les fonctions convexes et les inégalités entre les valeurs moyennes. *Acta mathematica*, 30:175–193, 1906.

Gholam Reza Moghissi received the M.S. degree in department of ICT at Malek-e-Ashtar University of Technology, Tehran, Iran, in 2016. His researches focus on information security.

Ali Payandeh received the M.S. degree in electrical engineering from Tarbiat Modares University in 1994, and the Ph.D. degree in electrical engineering from K.N. Toosi University of Technology (Tehran, Iran) in 2006. He is now an assistant professor in the department of information and communications technology at Malek-e-Ashtar University of Technology, Iran. His research interests include information theory, coding theory, cryptography, security protocols, secure communications, and satellite communications.

Appendices

A. Algorithms Used/Introduced in Paper

In this appendix, the pseudo-codes of essential algorithms in this paper are introduced.

A.1 Schnorr-Euchner's BKZ

Algorithm 1 shows the pseudo-code of original version of BKZ algorithm [26].

Algorithm 1 Block Korkin-Zolotarev (BKZ)

Input: $B = (b_1, \dots, b_n) \in \mathbb{Z}^{n \times m}$, GSO Coef Mat μ, $2 \leq \beta \leq n$, $1/4 \leq \delta < 1$.
Output: BKZ$_\beta$ reduced basis B.

1: LLL(B, μ, δ); //LLL reduce the basis and update μ
2: **for** $(z = 0, j = 0; z < n - 1;)$ **do**
3: $\quad j = (j \mod (n-1)) + 1$;
4: $\quad k = \min(j + \beta - 1, n)$;
5: $\quad h = \min(k + 1, n)$;
6: $\quad y \leftarrow \text{Enum}(\|b_j^*\|^2, \|b_{j+1}^*\|^2, \dots, \|b_k^*\|^2, \mu_{[j,k]})$;
7: $\quad$ **if** $y \neq (1, 0, \dots, 0)$ **then**
8: $\quad\quad \text{LLL}([b_1, \dots, b_{j-1}, \sum_{l=j}^{k} y_l b_l, b_j, \dots, b_h], \mu, \delta)$at stage j;
9: $\quad\quad z = 0$;
10: $\quad$ **else**
11: $\quad\quad \text{LLL}([b_1, \dots, b_h], \mu, \delta)$ at stage $h - 1$;
12: $\quad\quad z++$;
13: $\quad$ **end if**
14: **end for**

A.2 Our Sampling Algorithm of Solution Norm

The pseudo-code of the sampling method of solution norm by (37) in Lemma 3 can be studied in Algorithm 2. Note that, sampling x from exponential distribution where PDF $=$ Expo$(x; \vartheta)$ (see relation (14)), for $\vartheta = 0.5$ can be done by uniformly random selection of Y from corresponding CDF as $Y \leftarrow \text{CDF} = 1 - e^{-0.5x}$, and computing the corresponding sample of x (see line 2 in Algorithm 2).

A.3 Generator of Piecewise-Linear Bounding Func.

Here, we introduce the pseudo-code of generating piecewise-linear bounding function in Algorithm 3. The function of PiecewiceLinear$(\mathbf{a}, d)$ makes a piecewise-linear bounding function with length of d and piecewise-linear parameter of $\mathbf{a}$. The search approach for suitable parameter of $\mathbf{a}$ in this algorithm can be used for step bounding function too. Also function $\text{p}_{\text{succ}}(\mathcal{R})$ comes from relation (17) or (26) as the static success probability which is implemented in Algorithm 6 or Algorithm 7 from Appendix A in paper [2].

Algorithm 2 Our sampling method of solution norm ($\text{SAMPLE}_{\|v\|}$)

Input: block size d, static success probability p, Gaussian heuristic of local block $\mathfrak{gh}$, radii factor of r_{FAC}, Determinant of local block V, Volume of d_dimensional ball with unit radii as v_d.
Output: [succ, ℓ_{new}]. /*boolean variable of SUCC which determines whether a solution is sampled or not and norm of solution by parameter of ℓ_{new}*/

1: succ = true;
2: $x = -2\ln(1-\text{rand}_{(0\ldots1)})$;/*$x$ is sampled from Expo($x;\vartheta = 0.5$) where $0 < \text{rand}_{(0\ldots1)} < 1$*/
3: $\ell_{\min} \leftarrow (xV/v_d)^{1/d}$; /*see (31)*/
4: **if** ($\ell_{\min} \geq R$) **then** succ = false;
5: **else**
6: **if** ($p \approx 1$) **then** $\ell_{new} = \ell_{\min}$;
7: **else**
8: **if** ($\frac{2}{{r_{\text{FAC}}}^d} \leq p < 1$) **then**
9: $\ell_{new} = \sqrt[d]{1+\text{rand}_{[0\ldots1]}(\frac{2}{p}-1)} \times \mathfrak{gh}$;
10: **else**
11: **if** ($p < \frac{2}{{r_{\text{FAC}}}^d}$ and $\text{rand}_{[0\ldots\frac{2}{{r_{\text{FAC}}}^d}]} \leq p$) **then**
12: $\ell_{new} = \sqrt[d]{1+\text{rand}_{[0\ldots1]}({r_{\text{FAC}}}^d-1)} \times \mathfrak{gh}$;
13: **else**
14: **if** ($p < \frac{2}{{r_{\text{FAC}}}^d}$ and $\text{rand}_{[0\ldots\frac{2}{{r_{\text{FAC}}}^d}]} > p$) **then**
15: succ = false;
16: **end if**
17: **end if**
18: **end if**
19: **end if**
20: **end if**

Algorithm 3 $\text{GET}_{\mathbf{a}\_\text{CUT}\_1}$

Input: GSO norms of block given as $B^*_{[1,d]} = [\|b_1^*\|\ldots\|b_d^*\|]$, enumeration radii R, success probability p_{succT},BKZ specific structure S.
Output: piecewise linear parameter $\mathbf{a}$ and cutting point CUT.

1: $\text{step}_{\mathbf{a}} = 0.5$;//$\varepsilon^+ > 0$
2: **for** ($\mathbf{a} = \text{step}_{\mathbf{a}}; |\text{step}_{\mathbf{a}}| > \varepsilon^+; \mathbf{a} += \text{step}_{\mathbf{a}}$) **do**
3: $\mathcal{R} \leftarrow \text{PIECEWICELINEAR}(\mathbf{a}, d)$;
4: $p_0 = \text{p}_{\text{succ}}(\mathcal{R})$;
5: **if** ($p_0 - \text{p}_{\text{succT}} > 0$) **then**
6: $\text{step}_{\mathbf{a}} = -|\text{step}_{\mathbf{a}}/2|$;
7: **else**
8: $\text{step}_{\mathbf{a}} = |\text{step}_{\mathbf{a}}/2|$;
9: **end if**
10: **end for**
11: CUT $= -1$;
12: **for** ($i = d; i \geq 2; i--$) **do**
13: **if** ($\|b_i^*\| \leq R\mathcal{R}_{d-i+1}$) **then**
14: CUT $= i$; BREAK;
15: **end if**
16: **end for**

If terminating condition of "for"-loop (i.e., the condition which breaks the loop) in line 2 in Algorithm 3 would be replaced with the finishing condition of $|\text{p}_{\text{succ}}(\mathcal{R}) - \text{p}_{\text{succT}}| \leq \varepsilon^+$ then ε^+ is determined as a function of p_{succT}, while for finishing condition of $|\text{step}_{\mathbf{a}}| \leq \varepsilon^+$ (which is preferred by us), ε^+ is simply the minimum bound for steps in searching the suitable value of $\mathbf{a}$ in time of $O(\log_2(1/\varepsilon^+))$ iterations, where $\varepsilon^+ = \frac{1}{\text{number of step}}$.

A.4 Sampling Method for Index of g

Algorithm 4 shows the pseudo-code of sampling g. The lines 4-8 in Algorithm 4 determine CUT_{down} which is the minimum possible cutting point satisfying the radius factor constraint in Lemma 8. In fact, for sampling g, just one sample is selected randomly among the number of K solution samples. Note that, since the input parameter K shows the number of solution vectors in enumeration tree (without updating radius), so the probability of $\Pr(g = i)$ must be changed into $K\Pr(g = i)$. In lines 1-3 from Algorithm 4, this is determined that whether there is at least one solution or not, and if there is at least one solution to be returned by GNR-enumeration, then the probability of failure in enumeration for $g >$ CUT is not considered by this algorithm. Therefore, the probability of $\Pr(g = i)$ must be changed into $\Pr(g = i)/\text{CDF}(g \leq \text{CUT})$ for $g \leq$ CUT. Furthermore, since $p \leftarrow \text{rand}_{[0\ldots1]}$ in line 9 must be multiplied with K and then checked just by variable of cdf in line 12, the factor K is eliminated from these lines of Algorithm 4 for simplicity. Note that, the returned index in this algorithm showed by symbol of g_{idx}.

Algorithm 4 GET_g

Input: GSO norms of block given as $B^*_{[1,d]} = [\|b_1^*\|\ldots\|b_d^*\|]$, enumeration radii R, solution norm ℓ_{new},bounding function $\mathcal{R}$, radii factor r_{FAC}, K as the integer number of enumeration solutions.
Output: last non-zero index of g_{idx}.

1: **if** ($K < 1$) **then**
2: $g_{\text{idx}} = -1$; return g_{idx};
3: **end if**
4: **for** ($i = d-1; i \geq 2; i--$) **do**
5: **if** $\left(r_{\text{FAC}} < \frac{\text{GH}(\mathcal{L}_{[1,i]})}{\text{GH}(\mathcal{L}_{[1,d]})}\right)$ **then**
6: $\text{CUT}_{down} = i+1$; BREAK;
7: **end if**
8: **end for**
9: $p \leftarrow \text{rand}_{[0\ldots1]}; g_{\text{idx}} = -1$;
10: **for** (cdf $= 0, i =$ CUT$; i \geq \text{CUT}_{down}; i--$) **do**
11: cdf+ $= \frac{\Pr(g=i)}{\text{CDF}(g\leq\text{CUT})}$;/*CDF($g \leq$ CUT) can use relations (42), (43), (44), etc., also for relation (43) CDF($g \leq$ CUT) $= \frac{1}{2^{d-\text{CUT}}}$*/
12: **if** ($p \leq$ cdf) **then**
13: **if** ($\|b_i^*\| \leq \ell_{new}$ and $\|b_i^*\| \leq \mathcal{R}_{d-i+1}R$) **then** /*implicit condition which is applied automatically as follows ($1 < g_{\text{idx}}$ and $\text{CUT}_{down} \leq g_{\text{idx}} \leq$ CUT)*/
14: $g_{\text{idx}} = i$; BREAK;
15: **end if**
16: **end if**
17: **end for**

A.5 Sampling Method for Vector w

The pseudo-code of rejection sampling of coefficient vector w (proposed in Claim 1 and Lemma 11) is

shown in Algorithm 5. Our test results for Algorithm 5 can be studied in Test 1 from Section 4.2.1.

Algorithm 5 RejectionSample$_w$

Input: GSO norms of block given as $B_{[1,d]}^* = (\|b_1^*\| \dots \|b_d^*\|)$ at stage j, piecewise_linear parameter $\mathbf{a}_1$, enumeration radii R, Sampled norm of enumeration solution as ℓ_{new}, last nonzero index of g, boolean parameter of RandBlock, GSO Coefficient Matrix of μ.

Output: Sampled coefficient vectors w and y.

1: $p_0 = \frac{\mathrm{p_{succ}}(\mathcal{R}_{\mathbf{a}_1})}{\mathrm{F}(d)}$;
2: $\mathbf{a}_3 = \mathrm{Get}_{\mathbf{a}}(p_0, d)$;/*if(RandBlock=true) then $\mathbf{a}_3 = 0.5$; also $\mathrm{Get}_{\mathbf{a}}$ is simpler than $\mathrm{Get}_{\mathbf{a}\_\mathrm{cut}\_1}$, so that just searches for $\mathbf{a}$ where $\mathrm{p_{succ}}(\mathcal{R}_{\mathbf{a}}) \approx p_0$*/
3: **for** (SampleIsDone←false; SampleIsDone=false;) **do**
4: **for** $(t = 1; t \le g-1; t++)$ **do**
5: $\omega_t \leftarrow \mathrm{Gamma}(1/2, 2)$;
6: **end for**
7: **for** $(i = 1; i \le \lfloor d/2 \rfloor; i++)$ **do**
8: $\mathrm{sign} \leftarrow (-1)^{\lfloor \mathrm{rand}_{[0\dots 2)} \rfloor}$;
9: $w_i = \mathrm{sign} \sqrt{\frac{(1-\mathbf{a}_3)\omega_i(\ell_{new}^2 - \|b_g^*\|^2)}{\|b_i^*\|^2 \left((1-\mathbf{a}_3)\sum_{t=1}^{\lfloor \frac{d}{2} \rfloor} \omega_t + \mathbf{a}_3 \sum_{t=\lfloor \frac{d}{2} \rfloor + 1}^{g-1} \omega_t\right)}}$;
10: **end for**
11: **for** $(i = \lfloor d/2 \rfloor + 1; i \le g-1; i++)$ **do**
12: $\mathrm{sign} \leftarrow (-1)^{\lfloor \mathrm{rand}_{[0\dots 2)} \rfloor}$;
13: $w_i = \mathrm{sign} \sqrt{\frac{\mathbf{a}_3\omega_i(\ell_{new}^2 - \|b_g^*\|^2)}{\|b_i^*\|^2 \left((1-\mathbf{a}_3)\sum_{t=1}^{\lfloor \frac{d}{2} \rfloor} \omega_t + \mathbf{a}_3 \sum_{t=\lfloor \frac{d}{2} \rfloor + 1}^{g-1} \omega_t\right)}}$;
14: **end for**
15: $w_g = 1$;//for $i = g$
16: **for** $(i = g+1; i \le d; i++)$ **do**
17: $w_i = 0$;
18: **end for**
19: **for** $(t = d; t \ge 1; t--)$ **do**
20: $y_t \leftarrow w_t - \sum_{i=t+1}^{d} y_i \mu_{i,t}$;//see relation (54)
21: **end for**
22: **if** (RandBlock=true) **then**
23: BREAK;
24: **else**
25: SampleIsDone=true;
26: **for** $(t = d-g+2; t \le d; t++)$ **do**
27: **if** $\left(\sum_{l=d-g+1}^{t} w_{d-l+1}^2 \|b_{d-l+1}^*\|^2 > R^2 \mathcal{R}_t^2\right)$ **then**
28: SampleIsDone=false; BREAK;
29: **end if**
30: **end for**
31: **end if**
32: **end for**

All operations in line 7 from Algorithm 5 implements our proposed formula (51) in Lemma 11. Following remark describes the features of sampling coefficient vector of w in Algorithm 5 by considering whether the input block is re-randomized or not.

Remark 6. The boolean variable of "RandBlock" in Algorithm 5 (and Algorithm 6) determines that whether the input lattice block is randomized or not. If the lattice block is randomized, the parameter $\mathbf{a}$ in Lemma 11 is set to be $\mathbf{a} = 0.5$, similar to full-enumeration (the parameter $\mathbf{a}$ in Lemma 11 corresponds with $\mathbf{a}_3$ in Algorithm 5). Also if the lattice block is randomized, this is assumed that the solution vector with input norm of ℓ_{new} and last non-zero index of g (for its coefficient vectors) automatically satisfies the constraint of corresponding bounding function over randomized lattice block (see this constraint in (25)), and consequently this constraint is not checked for the lattice block (before being randomized) in line 24 from Algorithm 5.

Algorithm 6 generalizes our sampling method in Algorithm 5, in the way that the input bounding function $\mathcal{R}$ is not limited to piecewise-linear bounding function, also is not limited to short enumeration radius factor or even small success probability of bounding function (i.e., Algorithm 6 allows to work with any length of enumeration radius, any success probability and any bounding function). Our test results for Algorithm 6 can be studied in Test 2 (in Section 4.2.2) and Test 3 (in Section 4.2.3).

For these two algorithms, following remark introduces an important approximate technique to speed-up their sampling

Remark 7. The way proposed in this paper for sampling coefficient vector of w (or vector of z) by Claim 1 and Lemma 11 (see Algorithm 5) and our generalized sampling technique in Algorithm 6 are novel ideas which approximates reasonably the original sampling in (47) and (48) bounded by input bounding function. In fact, for BKZ-simulation, this is believed in this paper that the cost results and output quality of BKZ-simulation by using precise sampling for coefficient vector w (and vector z) differ from some approximate one negligibly. Therefore, there is no need to force the constraint of $X_i = \sum_{t=1}^{i} z_t^2/R^2 \le \mathcal{R}_i'^2$ (see relation (17)) over sampled vector z (and vector w) in BKZ-simulations, while this constraint (in line 24 of Algorithm 5 and line 28 of Algorithm 6) may lead to a long time loops, while the satisfaction of this condition (constraint) may cause no sensible effect on the BKZ-simulation outputs. Accordingly, line 24 in Algorithm 5 and line 28 in Algorithm 6 can be eliminated.

B. Proof of Lemmas and Theorems

In this appendix, our lemmas and theorems which are introduced in this paper are proved (some short proofs are included in the main text of paper).

B.1 Proof of Lemma 2

To prove this lemma, for given lattice block $\mathcal{L}_\beta$ and an input enumeration radius $R_H = r_{\mathrm{FAC}\,H}\mathrm{GH}(\mathcal{L}_\beta)$, this

Algorithm 6 GeneralizedRejectionSample$_w$

Input: GSO norms of block given as $B^*_{[1,d]} = (\|b_1^*\| \dots \|b_d^*\|)$ at stage j, any type of bounding function $\mathcal{R}$, enumeration radii R, Sampled norm of enumeration solution as ℓ_{new}, last nonzero index of g, boolean parameter of RandBlock, GSO Coefficient Matrix of μ.

Output: Sampled coefficient vectors w and y.

$p_0 = \mathrm{p_{succ}}(\mathcal{R})$; $\mathbf{a}_0 = \mathrm{GET}_{\mathbf{a}}(p_0, d)$; /*$\mathrm{GET}_{\mathbf{a}}(p_0, d)$ is simpler than $\mathrm{GET}_{\mathbf{a\_CUT\_1}}$ so that just searches for $\mathbf{a}_0$ where $\mathrm{p_{succ}}(\mathcal{R}_{\mathbf{a}_0}) \approx p_0$ */
$p_1 = \frac{p_0}{\mathrm{F}(d)}$; $\mathbf{a}_1 = \mathrm{GET}_{\mathbf{a}}(p_1, d)$;
$A = \frac{\mathbf{a}_1 R^2 - \|b_g^*\|^2}{\frac{d}{2} - (d-g+1)}$; $\mathbf{a}_3 = \frac{dA}{2\ell_{new}^2}$;
if $(\mathbf{a}_3 > 0.5)$ **then**
  $\mathbf{a}_3 = 0.5$;
end if//if(RandBlock=true) then $\mathbf{a}_3 = 0.5$;
for (SampleIsDone←false; SampleIsDone=false;) **do**
  for $(t = 1; t \le g-1; t++)$ **do**
    $\omega_t \leftarrow \mathrm{Gamma}(1/2, 2)$;
  end for
  for $(i = 1; i \le \lfloor d/2 \rfloor; i++)$ **do**
    $\mathrm{sign} \leftarrow (-1)^{\lfloor \mathrm{rand}_{[0\dots 2)} \rfloor}$;
    $w_i = \mathrm{sign} \sqrt{\frac{(1-\mathbf{a}_3)\omega_i(\ell_{new}^2 - \|b_g^*\|^2)}{\|b_i^*\|^2 \left((1-\mathbf{a}_3) \sum_{t=1}^{\lfloor \frac{d}{2} \rfloor} \omega_t + \mathbf{a}_3 \sum_{t=\lfloor \frac{d}{2} \rfloor + 1}^{g-1} \omega_t \right)}}$;
  end for
  for $(i = \lfloor d/2 \rfloor + 1; i \le g-1; i++)$ **do**
    $\mathrm{sign} \leftarrow (-1)^{\lfloor \mathrm{rand}_{[0\dots 2)} \rfloor}$;
    $w_i = \mathrm{sign} \sqrt{\frac{\mathbf{a}_3\omega_i(\ell_{new}^2 - \|b_g^*\|^2)}{\|b_i^*\|^2 \left((1-\mathbf{a}_3) \sum_{t=1}^{\lfloor \frac{d}{2} \rfloor} \omega_t + \mathbf{a}_3 \sum_{t=\lfloor \frac{d}{2} \rfloor + 1}^{g-1} \omega_t \right)}}$;
  end for
  $w_g = 1$;//for $i = g$
  for $(i = g+1; i \le d; i++)$ **do**
    $w_i = 0$;
  end for
  for $(t = d; t \ge 1; t--)$ **do**
    $y_t \leftarrow w_t - \sum_{i=t+1}^{d} y_i \mu_{i,t}$;//see relation (54)
  end for
  if (RandBlock=true) **then**
    BREAK;
  else
    SampleIsDone=true;
    for $(t = d-g+2; t \le d; t++)$ **do**
      if $\left(\sum_{l=d-g+1}^{t} w_{d-l+1}^2 \|b_{d-l+1}^*\|^2 > R^2 \mathcal{R}_t^2 \right)$ **then**
        SampleIsDone=false; BREAK;
      end if
    end for
  end if
end for

is needed to show that the probability distribution of the norm of returned solution vector by a pruned enumeration which includes a bounding function with static success probability $2/r_{\mathrm{FAC}}{}_H^{\beta}$ is equal to the probability distribution of sampled norm by formula (36). This goal for proof can be modified into a well-defined equivalent proposition, as follows.

Goal. For an input enumeration radius R_H, the probability of that the returned solution vector from an actual pruned enumeration function belongs to the range of enumeration radius $[R_1, R_2]$, where $R_1 \le R_2 \le R_H$, is the same as the probability of that the sampled norm by Lemma 2 belongs to this range.

Note: For input lattice block $\mathcal{L}_\beta$ with block size β: $\mathrm{GH}(\mathcal{L}_\beta) \le R_1 = r_{\mathrm{FAC}1}\mathrm{GH}(\mathcal{L}_\beta) \le R_2 = r_{\mathrm{FAC}2}\mathrm{GH}(\mathcal{L}_\beta) \le R_H = r_{\mathrm{FAC}H}\mathrm{GH}(\mathcal{L}_\beta)$.

For actual enumeration function on lattice block $\mathcal{L}_\beta$, the number of solution vectors in a ball with each enumeration radius of R_y, where $\mathrm{GH}(\mathcal{L}_\beta) \le R_y \le R_H$, is determined simply by using Roger's theorem as follows $n_y = r_{\mathrm{FAC}}{}_y^{\beta}/2$. Therefore, the probability of that the returned solution vector from an actual full-enumeration function belongs to the range of enumeration radius $[R_1, R_2]$, can be determined as follows.

$$P1 = (n_2 - n_1)/n_H = (r_{\mathrm{FAC}}{}_2^{\beta} - r_{\mathrm{FAC}}{}_1^{\beta})/r_{\mathrm{FAC}}{}_H^{\beta}.$$

In other side, the probability of whether the sampled norm in this lemma belongs to the range of enumeration radius $[R_1, R_2]$ can be computed by determining the corresponding distance of $[x_1, \dots, x_2]$ from the range of $[0, \dots, 1]$ in random number generation in formula (36), as follows

$$r_{\mathrm{FAC}1}\mathrm{GH}(\mathcal{L}_\beta) \le \|v\| \le r_{\mathrm{FAC}2}\mathrm{GH}(\mathcal{L}_\beta) \Rightarrow$$
$$r_{\mathrm{FAC}1}\mathrm{GH}(\mathcal{L}_\beta) \le \sqrt[\beta]{1 + \mathrm{rand}_{[x_1,\dots,x_2]}(r_{\mathrm{FAC}}{}_H^{\beta} - 1)}\mathrm{GH}(\mathcal{L}_\beta) \le r_{\mathrm{FAC}2}\mathrm{GH}(\mathcal{L}_\beta) \Rightarrow$$
$$r_{\mathrm{FAC}}{}_1^{\beta} \le 1 + \mathrm{rand}_{[x_1,\dots,x_2]}(r_{\mathrm{FAC}}{}_H^{\beta} - 1) \le r_{\mathrm{FAC}}{}_2^{\beta} \Rightarrow$$
$$r_{\mathrm{FAC}}{}_1^{\beta} - 1 \le \mathrm{rand}_{[x_1,\dots,x_2]}(r_{\mathrm{FAC}}{}_H^{\beta} - 1) \le r_{\mathrm{FAC}}{}_2^{\beta} - 1 \Rightarrow$$
$$x_1 = \frac{r_{\mathrm{FAC}}{}_1^{\beta} - 1}{r_{\mathrm{FAC}}{}_H^{\beta} - 1} \text{ and } x_2 = \frac{r_{\mathrm{FAC}}{}_2^{\beta} - 1}{r_{\mathrm{FAC}}{}_H^{\beta} - 1} \Rightarrow$$
$$P2 = x_2 - x_1 = \frac{r_{\mathrm{FAC}}{}_2^{\beta} - r_{\mathrm{FAC}}{}_1^{\beta}}{r_{\mathrm{FAC}}{}_H^{\beta} - 1}$$

Because of the equality of these two probabilities (i.e., $P1 \approx P2$), this proof is completed.

B.2 Proof of Lemma 3

Since Lemma 3 is introduced under the condition that the norm of shortest vector of lattice block $\mathcal{L}_\beta$ is less than enumeration radius R, so there is at least one solution which can be sampled or cannot be successfull in sampling. Under this condition, four parts of sampling method of (37) can be proved respectively as follows.

- If $P \approx 1$, then corresponding enumeration is not pruned (i.e., this full-enumeration), and therefore relation (31) from Theorem 1 can be used.
- As mentioned in Section 2.6, dynamic success frequency of bounding function $\mathcal{R}$ with static success probability $P = \mathrm{p_{succ}}(\mathcal{R})$ is defined as

$f_{\text{succ}} = P^{\frac{r_{\text{FAC}}{}^{\beta}}{2}}$, therefore if $P \geq \frac{2}{r_{\text{FAC}}{}^{\beta}}$ then $f_{\text{succ}} \geq 1$, and the solution number is nearly $\approx \lfloor f_{\text{succ}} \rceil$ in average-case; If $P = \frac{2}{r_{\text{FAC}}{}^{\beta}}$, then the number of solution is only $\lfloor f_{\text{succ}} \rceil = 1$ and consequently the sampling method (37) is the same as (36) in Lemma 2; If $\frac{2}{r_{\text{FAC}}{}^{\beta}} < P < 1$, then the number of solution is $1 \leq \lfloor f_{\text{succ}} \rceil \leq \frac{r_{\text{FAC}}{}^{\beta}}{2}$, but for solution number > 1, one solution should be returned finally as the response of GNR enumeration after updating radius; By using Fact 1, the shortest solution vector is never eliminated by updating radius and finally is returned by GNR-enumeration. Therefore the norm of shortest solution only should be estimated. To sample the norm of shortest solution in this case, the minimum radius for GNR-enumeration with static success probability P is needed, so that this GNR-enumeration returns just one solution (i.e., a radius factor r_0 is searched to force the success probability of corresponding GNR-pruned enumeration to $P^{\frac{r_0^{\beta}}{2}} = 1$). Since the β-dimensional ball for full-enumeration with radius $R = \sqrt[\beta]{2/P}\text{GH}(\mathcal{L}_\beta) < r_{\text{FAC}}\text{GH}(\mathcal{L}_\beta)$ includes $1/P$ number of solutions, the cylinder-intersection of corresponding GNR-pruned enumeration with success probability P has one solution in average-case (i.e., $P \times \frac{\left(\sqrt[\beta]{2/P}\right)^{\beta}}{2} = 1$) which can be sampled by Lemma 2:

$$\|v\| = \sqrt[\beta]{1 + \text{rand}_{[0\ldots1]}\left(\left(\sqrt[\beta]{2/P}\right)^{\beta} - 1\right)}\text{GH}(\mathcal{L}_\beta) = \sqrt[\beta]{1 + \text{rand}_{[0\ldots1]}(2/P - 1)}\text{GH}(\mathcal{L}_\beta)$$

Note: Other solution vectors v with norms in range of $\sqrt[\beta]{2/P}\text{GH}(\mathcal{L}_\beta) < \|v\| \leq r_{\text{FAC}}\text{GH}(\mathcal{L}_\beta)$ are eliminated by updating radii.

- For static success probability of $P < \frac{2}{r_{\text{FAC}}{}^{\beta}}$ and dynamic success frequency of $f_{\text{succ}} = P^{\frac{r_{\text{FAC}}{}^{\beta}}{2}} < 1$ (see the concepts of static success probability and dynamic success frequency in Section 2.6), there is one solution with norm $\|v\| < r_{\text{FAC}}\text{GH}(\mathcal{L}_\beta)$ to be returned by enumeration with probability of f_{succ}. In this case, the condition of "$\text{rand}_{[0\ldots\frac{2}{r_{\text{FAC}}{}^{\beta}}]} \leq P$" check that whether there is a solution to be sampled or not, as follows

$$\text{rand}_{[0\ldots1]} \leq f_{\text{succ}} \Rightarrow \text{rand}_{[0\ldots1]} \leq P^{\frac{r_{\text{FAC}}{}^{\beta}}{2}} \Rightarrow \text{rand}_{[0\ldots\frac{r_{\text{FAC}}{}^{\beta}}{2}]} \leq P$$

Therefore if the condition of "$\text{rand}_{[0\ldots\frac{2}{r_{\text{FAC}}{}^{\beta}}]} \leq P$" is true, then there is one solution with norm $\|v\| < r_{\text{FAC}}\text{GH}(\mathcal{L}_\beta)$, and consequently the norm of this solution can be sampled by Lemma 2.
- By using our reasoning in previous item, for static success probability of $P < \frac{2}{r_{\text{FAC}}{}^{\beta}}$, if condition of "$\text{rand}_{[0\ldots\frac{2}{r_{\text{FAC}}{}^{\beta}}]} \leq P$" is false, then there is no solution to be sampled.

This proof is completed.

B.3 Proof of Lemma 4

For simplicity in this proof, without losing generality, Gaussian heuristic is assumed as the minimum expected norm of solution (which is approximately consistent with Theorem 1). By this assumption, the condition of $\phi(\beta, f_{\text{succ}1}) \leq r_{\text{FAC}}$ should be satisfied to result in the condition of $C_0 \geq C_r$. This proof divide into two steps. In Step 1, Lemma 4 is proved for $\beta = n$. At next, in Step 2, this is proved that for $\beta < n$, after less number of rounds than the rounds count in Step 1, the quality of BKZ_β-reduced basis with dynamic success frequency of $f_{\text{succ}1}$ is similar to one round of BKZ_β-reduced basis with full-enumeration.

Note: However this paper defines the concept of cutting point in Section 3.2.2, the bounding functions is assumed to be forced in this proof for cutting point of $\text{CUT} = d$ (there is some techniques to force this).

***Step 1*:** In this step, when an enumeration solution is added at the first of lattice block and consequently GSO vectors of b_i^* in this block are updated for $\beta = n$, the determinant of the first block would be the same as old one (i.e., Gaussian heuristic of the block is not changed after each round). The GSO norm of first vector in block of $\mathcal{L}_{[b_1,\ldots,b_{\beta=n}]}$, after each round of $\text{BKZ}_{\beta=n}$ with dynamic success frequency of $f_{\text{succ}1}$, is computed by using (19), as follows.

Note: In this proof, the notations of $l_j^{[i]}$, $\text{GH}_{[j,k]}^{[i]}$ and $R_{[j,k]}^{[i]}$ respectively represent the GSO norm of $\|b_j^*\|$ in round i, Gaussian heuristic and enumeration radius R based on (19) for block $\mathcal{L}_{[j,k]}$ in round i, also for simplicity, the notation of $\phi(\beta, f_{\text{succ}1})$ is shown by ϕ.

Round 0: $l_1^{[0]} = \|b_1^*\|$, with $\text{GH}_{[1,\beta]}^{[0]} = \sqrt{\frac{\beta}{2\pi e}}(l_1^{[0]} \times l_2^{[0]} \times \ldots \times l_\beta^{[0]})^{1/\beta}$ where $R_{[1,\beta]}^{[0]} = r_{\text{FAC}}\text{GH}_{[1,\beta]}^{[0]}$ by using (21);

Round 1: $l_1^{[1]} = \frac{1}{\phi} r_{\text{FAC}}\text{GH}_{[1,\beta]}^{[0]}$, with $\text{GH}_{[1,\beta]}^{[1]} = \text{GH}_{[1,\beta]}^{[0]}$ where $R_{[1,\beta]}^{[1]} = \frac{1}{\phi} r_{\text{FAC}}\text{GH}_{[1,\beta]}^{[0]}$ by (21);

Round 2: $l_1^{[2]} = \left(\frac{1}{\phi}\right)^2 r_{\text{FAC}}\text{GH}_{[1,\beta]}^{[0]}$, with $\text{GH}_{[1,\beta]}^{[2]} = \text{GH}_{[1,\beta]}^{[0]}$ where $R_{[1,\beta]}^{[2]} = \left(\frac{1}{\phi}\right)^2 r_{\text{FAC}}\text{GH}_{[1,\beta]}^{[0]}$ by (21);

$\vdots$

Round $\mathcal{N}$: $l_1^{[N]} = \left(\frac{1}{\phi}\right)^{\mathcal{N}} r_{\text{FAC}}\text{GH}_{[1,\beta]}^{[0]}$, with $\text{GH}_{[1,\beta]}^{[\mathcal{N}]} = \text{GH}_{[1,\beta]}^{[0]}$ where $R_{[1,\beta]}^{[N]} = \left(\frac{1}{\phi}\right)^{\mathcal{N}} r_{\text{FAC}}\text{GH}_{[1,\beta]}^{[0]}$.

To reach the solution norm of $\mathrm{GH}^{[0]}_{[1,\beta]}$ in this step, the rounds count of $\mathcal{N}$ is computed as follows.
$\frac{1}{\phi^{\mathcal{N}}} r_{\mathrm{FAC}} = 1 \Rightarrow \mathcal{N}\ln(1+1/C_0) = \ln(1+1/C_r) \Rightarrow$
$\mathcal{N} = \frac{\ln\left(\frac{C_r+1}{C_r}\right)}{\ln\left(\frac{C_0+1}{C_0}\right)} = \frac{\ln(C_r+1)-\ln(C_r)}{\ln(C_0+1)-\ln(C_0)} = \frac{\frac{1}{C_r}}{\frac{1}{C_0}} = \frac{C_0}{C_r}$

***Step 2*:** In this step, when an enumeration solution is added at the first of lattice block with size of $\beta < n$ and consequently GSO norms of vectors in this block are updated, the determinant of the first block is preserved. After updating GSO, when partial-LLL (see line 7 in Algorithm 1) applied on the basis, the determinant of the block and the corresponding Gaussian heuristic can be modified.

Note: Since updating GSO just changes the GSO norm of lattice block vectors (i.e., norms of $\|b_i^*\|$), so the determinant of current block is not changed by updating GSO, but the determinant of previous blocks are changed. In other side, partial-LLL (see line 7 in Algorithm 1) always can modify the determinant of all lattice blocks in the basis.

By using Remark 1, after each round of $\mathrm{BKZ}_{\beta<n}$ with dynamic success frequency of f_{succ1}, the first GSO norm is computed as follows.

Round 0: $l_1^{[0]} = \|b_1^*\|$, with $\mathrm{GH}^{[0]}_{[1,\beta]} = \sqrt{\frac{\beta}{2\pi e}}(l_1^{[0]} \times l_2^{[0]} \times \ldots \times l_\beta^{[0]})^{1/\beta}$ where $R^{[0]}_{[1,\beta]} = r_{\mathrm{FAC}}\mathrm{GH}^{[0]}_{[1,\beta]} \leq \sqrt{\Upsilon}\mathrm{GH}^{[0]}_{[1,\beta]}$, since $R^{[0]}_{[1,\beta]} = \min(\sqrt{\Upsilon}\mathrm{GH}^{[0]}_{[1,\beta]}, l_1^{[0]})$ by using (21).

Round 1: $l_1^{[1]} = \frac{1}{\phi} r_{\mathrm{FAC}}\mathrm{GH}^{[0]}_{[1,\beta]}$, where $R^{[1]}_{[1,\beta]} \leq \frac{1}{\phi} r_{\mathrm{FAC}}\mathrm{GH}^{[0]}_{[1,\beta]}$ because of following three states:

(1) For $\mathrm{GH}^{[1]}_{[1,\beta]} > \mathrm{GH}^{[0]}_{[1,\beta]}$, then $R^{[1]}_{[1,\beta]} = \min(\sqrt{\Upsilon}\mathrm{GH}^{[1]}_{[1,\beta]}, l_1^{[1]}) = l_1^{[1]} = \frac{1}{\phi} r_{\mathrm{FAC}}\mathrm{GH}^{[0]}_{[1,\beta]}$, since $l_1^{[1]} < \sqrt{\Upsilon}\mathrm{GH}^{[0]}_{[1,\beta]} < \sqrt{\Upsilon}\mathrm{GH}^{[1]}_{[1,\beta]}$ and $l_1^{[1]} < l_1^{[0]}$,

(2) For $\mathrm{GH}^{[1]}_{[1,\beta]} = \mathrm{GH}^{[0]}_{[1,\beta]}$, then $R^{[1]}_{[1,\beta]} = l_1^{[1]} = \frac{1}{\phi} r_{\mathrm{FAC}}\mathrm{GH}^{[0]}_{[1,\beta]}$ (similar to our reasoning in Step 1),

(3) For $\mathrm{GH}^{[1]}_{[1,\beta]} < \mathrm{GH}^{[0]}_{[1,\beta]}$, then $R^{[1]}_{[1,\beta]} = \min(\sqrt{\Upsilon}\mathrm{GH}^{[1]}_{[1,\beta]}, l_1^{[1]}) \leq l_1^{[1]} = \frac{1}{\phi} r_{\mathrm{FAC}}\mathrm{GH}^{[0]}_{[1,\beta]}$, since if $\sqrt{\Upsilon}\mathrm{GH}^{[1]}_{[1,\beta]} < l_1^{[1]}$ then $R^{[1]}_{[1,\beta]} = \sqrt{\Upsilon}\mathrm{GH}^{[1]}_{[1,\beta]} < l_1^{[1]}$ else $R^{[1]}_{[1,\beta]} = l_1^{[1]}$.

Note: By using these three states, in average case, this can be concluded that $R^{[1]}_{[1,\beta]} \leq l_1^{[1]}$.

Round 2: $l_1^{[2]} = \frac{1}{\phi} R^{[1]}_{[1,\beta]} \leq \left(\frac{1}{\phi}\right)^2 r_{\mathrm{FAC}}\mathrm{GH}^{[0]}_{[1,\beta]}$, where $R^{[2]}_{[1,\beta]} \leq \left(\frac{1}{\phi}\right)^2 r_{\mathrm{FAC}}\mathrm{GH}^{[0]}_{[1,\beta]}$ for following states

(1) For $\mathrm{GH}^{[2]}_{[1,\beta]} > \mathrm{GH}^{[1]}_{[1,\beta]}$, then $R^{[2]}_{[1,\beta]} = \min(\sqrt{\Upsilon}\mathrm{GH}^{[2]}_{[1,\beta]}, l_1^{[2]}) = l_1^{[2]} \leq \left(\frac{1}{\phi}\right)^2 r_{\mathrm{FAC}}\mathrm{GH}^{[0]}_{[1,\beta]}$, since $l_1^{[2]} < \sqrt{\Upsilon}\mathrm{GH}^{[1]}_{[1,\beta]} < \sqrt{\Upsilon}\mathrm{GH}^{[2]}_{[1,\beta]}$ and $l_1^{[2]} < l_1^{[1]}$,

(2) For $\mathrm{GH}^{[2]}_{[1,\beta]} = \mathrm{GH}^{[1]}_{[1,\beta]}$, then $R^{[2]}_{[1,\beta]} = l_1^{[2]} \leq \left(\frac{1}{\phi}\right)^2 r_{\mathrm{FAC}}\mathrm{GH}^{[0]}_{[1,\beta]}$,

(3) For $\mathrm{GH}^{[2]}_{[1,\beta]} < \mathrm{GH}^{[1]}_{[1,\beta]}$, then $R^{[2]}_{[1,\beta]} = \min(\sqrt{\Upsilon}\mathrm{GH}^{[2]}_{[1,\beta]}, l_1^{[2]}) \leq l_1^{[2]} \leq \left(\frac{1}{\phi}\right)^2 r_{\mathrm{FAC}}\mathrm{GH}^{[0]}_{[1,\beta]}$, since if $\sqrt{\Upsilon}\mathrm{GH}^{[2]}_{[1,\beta]} < l_1^{[2]}$ then $R^{[2]}_{[1,\beta]} = \sqrt{\Upsilon}\mathrm{GH}^{[2]}_{[1,\beta]} < l_1^{[2]}$ else $R^{[2]}_{[1,\beta]} = l_1^{[2]}$.

Note: By using these three states, in average case, this can be concluded that $R^{[2]}_{[1,\beta]} \leq l_1^{[2]}$.

Round 3: $l_1^{[3]} \leq \left(\frac{1}{\phi}\right)^3 r_{\mathrm{FAC}}\mathrm{GH}^{[0]}_{[1,\beta]}$, where $R^{[3]}_{[1,\beta]} \leq \left(\frac{1}{\phi}\right)^{[3]} r_{\mathrm{FAC}}\mathrm{GH}^{[0]}_{[1,\beta]}$ with the same reasoning.

$\vdots$

Round $\mathcal{N}$: $l_1^{[\mathcal{N}]} \leq \left(\frac{1}{\phi}\right)^{[\mathcal{N}]} r_{\mathrm{FAC}}\mathrm{GH}^{[0]}_{[1,\beta]}$, where $R^{[\mathcal{N}]}_{[1,\beta]} \leq \left(\frac{1}{\phi}\right)^{[\mathcal{N}]} r_{\mathrm{FAC}}\mathrm{GH}^{[0]}_{[1,\beta]}$ with the same reasoning.

To reach to solution norm of $\mathrm{GH}^{[0]}_1$ in this step, the rounds count of $\mathcal{N}$ can be computed as follows
$\frac{1}{\phi^{\mathcal{N}}} r_{\mathrm{FAC}} \geq 1 \Rightarrow \mathcal{N}\ln(1+1/C_0) \leq \ln(1+1/C_r) \Rightarrow$
$\mathcal{N} \leq \frac{\ln(\frac{C_r+1}{C_r})}{\ln(\frac{C_0+1}{C_0})} = \frac{\ln(C_r+1)-\ln(C_r)}{\ln(C_0+1)-\ln(C_0)} = \frac{\frac{1}{C_r}}{\frac{1}{C_0}} = \frac{C_0}{C_r}$.

B.4 Proof of Theorem 2

By inserting a vector v at the beginning of the linear independent GSO vectors of b_1^*, b_2^*, …, b_d^*, while vector v is generated by the linear combination of these GSO vectors as $v = \sum_{i=1}^{d} w_i b_i^*$, GSO process just eliminates the last vector b_g^*, where $w_g \neq 0$. Because of eliminating one of the solution pairs $(v, -v)$ in the enumeration tree (because of the symmetry of enumeration tree), the enumeration algorithms just can use $w_g > 0$ or $w_g < 0$. The enumeration algorithm usually uses the positive values for w_g (however use of negative w_g leads to a symmetric solution $-v$; see line 29 in Algorithm 2 from paper [1] and line 32 in Algorithm 9 from paper [2], which use $w_g > 0$). Therefore

$0 < w_g = y_g + \sum_{i=g+1}^{d} y_i \mu_{i,g} \leq \frac{\|v\|}{\|b_g^*\|}$
$(y_{g+1}, y_{g+2}, \ldots, y_d) = (0, 0, \ldots, 0) \Rightarrow$
$\sum_{i=g+1}^{d} y_i \mu_{i,g} = 0 \Rightarrow 0 < w_g = y_g \leq \frac{\|v\|}{\|b_g^*\|} \Rightarrow w_g \in \mathbb{N}.$

Now there are two states which can be assumed as

follows.

State 1: For $w_g = y_g = 1$ then $\|v\| = \ell_1$.

State 2: For $w_g = y_g > 1$ then $\ell_2 = \ell_1 + (y_g - 1)\|b_g^*\| \Rightarrow$ State 1 always leads to shorter solution of $v \Rightarrow$

$$w_g = y_g = 1. \tag{56}$$

So this is concluded that $\|b_g^*\| \leq \|v\|$. Now by adding v at the first of block, the GSO vector b_g^* is updated as follows (here, the old vector of b_g^* is shown by b_g^* and the updated vector is shown by $b_{g_{new}}^*$):

The old orthogonal block of $(b_1^*, b_2^*, \ldots, b_g^*, \ldots, b_d^*)$ is updated to $[\pi_1(v), \pi_2(b_1), \ldots, \pi_{g+1}(b_g), \ldots, \pi_{d+1}(b_d)]$. Also by using the relation of $\text{SPAN}(v, b_1, \ldots, b_g)^{\perp} = \text{SPAN}(b_1, \ldots, b_g, v)^{\perp}$, the updated orthogonal block can be represented by $\mathcal{L}_{[1,d+1]_{new}} = [\pi_1(b_1), \pi_2(b_2), \ldots, \pi_g(v), \pi_{g+1}(b_g), \ldots, \pi_{d+1}(b_d)]$. Now by using $\mathcal{L}_{[1,d+1]_{new}}$, the vector of $b_{g_{new}}^*$ would be computed as follows.

$$b_{g_{new}}^* = \pi_{g+1}(b_g) = b_g - \sum_{j=1}^{\overbrace{g-1}^{b_g^*}} \mu_{g,j} b_j^* - \mu_{g,v}\pi_g(v)$$
$$= b_g^* - \mu_{g,v}\pi_g(v) \tag{57}$$

where for $1 \leq i \leq g-1$, GSO coefficient of $\mu_{v,i}$ is defined as follows.

$$\mu_{v,i} = \frac{v b_i^*}{\|b_i^*\|^2} = \frac{w_i \|b_i^*\|^2}{\|b_i^*\|^2} = w_i. \tag{58}$$

By using (58), (56) and (38)

$$\pi_g(v) = v - \sum_{j=1}^{g-1} \mu_{v,j} b_j^* = v - (w_1 b_1^* + w_2 b_2^* + \ldots + w_{g-1} b_{g-1}^*) = b_g^*. \tag{59}$$

Also, by using (38) and (59), the GSO coefficient of $\mu_{g,v}$ for b_g over solution v at index of g can be computed as

$$\mu_{g,v} = \frac{b_g \pi_g(v)}{\|\pi_g(v)\|^2} = \frac{b_g b_g^*}{\|b_g^*\|^2} = \frac{\|b_g^*\|^2}{\|b_g^*\|^2} = 1. \tag{60}$$

By using (60), (57) and (59) $\pi_{g+1}(b_g) = b_{g_{new}}^* = 0$; The proof is completed.

B.5 Proof of Lemma 6

GSA assumption leads to $\|b_d^*\|^2 = \frac{\|b_1^*\|^2}{q^{2d-2}}$.

By using the definition of piecewise-linear bounding function with parameter of $\mathbf{a}$ and condition of $\|b_g^*\|^2 \leq R^2 \mathcal{R}_{d-g+1}^2 \Rightarrow \|b_d^*\|^2 \leq \frac{2\mathbf{a}}{d} R^2$.

By using the worst-case assumption of $\|b_1^*\| = R = r_{\text{FAC}}\text{GH}(\mathcal{L}_{[1,d]})$ in satisfying the condition of $\|b_g^*\|^2 \leq R^2 \mathcal{R}_{d-g+1}^2 \Rightarrow \|b_d^*\|^2 \leq \frac{2\mathbf{a}}{d}\|b_1^*\|^2 \Rightarrow$

$$\frac{d}{2q^{2d-2}} \leq \mathbf{a} \tag{61}$$

By using (11) and worst case assumption of $\|b_1^*\| = R$ (the notation δ_r shows the root-Hermite factor)

$$q \approx {\delta_r}^2 = \left(\frac{\|b_1^*\|}{\det(\mathcal{L}_{[1,d]})^{1/d}}\right)^{2/d} = \left(\frac{r_{\text{FAC}}\text{GH}(\mathcal{L}_{[1,d]})}{\det(\mathcal{L}_{[1,d]})^{1/d}}\right)^{2/d} \Rightarrow$$
$$q \approx \left(\frac{r_{\text{FAC}}\sqrt{\frac{d}{2\pi e}}\det(\mathcal{L}_{[1,d]})^{1/d}}{\det(\mathcal{L}_{[1,d]})^{1/d}}\right)^{2/d} \Rightarrow$$
$$q \approx \left(\frac{d\,{r_{\text{FAC}}}^2}{2\pi e}\right)^{1/d} \tag{62}$$

By using (61) and (62), $\frac{2(\pi e)^2}{d\,{r_{\text{FAC}}}^4} \leq \mathbf{a}$.

B.6 Proof of Lemma 8

By using the definition of cutting point in Section 3.2.2, Roger's theorem says that, there are the number of $\frac{{r_{\text{FAC}}}^{\text{CUT}}}{2}$ solution vector pairs $(v, -v)$ with in the ball of radius $R = r_{\text{FAC}}\text{GH}(\mathcal{L}_{[1,d]})$ for the sufficiently big block size d. Accordingly, the probability distribution for last non-zero index of g can be computed as follows,

$\Pr(g = i) \approx \frac{({}^{R}/_{\text{GH}(\mathcal{L}_{[1,i]})})^i/2 - ({}^{R}/_{\text{GH}(\mathcal{L}_{[1,i-1]})})^{i-1}/2}{{r_{\text{FAC}}}^d/2}$.

By using (21):$\Pr(g = i) \approx$

$$\frac{\left(r_{\text{FAC}}\frac{\text{GH}\left(\mathcal{L}_{[1,d]}\right)}{\text{GH}\left(\mathcal{L}_{[1,i]}\right)}\right)^i - \left(r_{\text{FAC}}\frac{\text{GH}\left(\mathcal{L}_{[1,d]}\right)}{\text{GH}\left(\mathcal{L}_{[1,i-1]}\right)}\right)^{i-1}}{{r_{\text{FAC}}}^d}.$$

By using (7) and (9),

$$\Pr(g = i) \approx \frac{\left(r_{\text{FAC}}\sqrt{\frac{d}{i}}\frac{\left(\|b_1^*\|\ldots\|b_d^*\|\right)^{1/d}}{\left(\|b_1^*\|\ldots\|b_i^*\|\right)^{1/i}}\right)^i}{{r_{\text{FAC}}}^d} -$$
$$\frac{\left(r_{\text{FAC}}\sqrt{\frac{d}{i-1}}\frac{\left(\|b_1^*\|\ldots\|b_d^*\|\right)^{1/d}}{\left(\|b_1^*\|\ldots\|b_{i-1}^*\|\right)^{1/(i-1)}}\right)^{i-1}}{{r_{\text{FAC}}}^d}.$$

By using (10) and some simplifications: $\Pr(g = i) \approx {r_{\text{FAC}}}^{i-d}(d/i)^{i/2}q^{i(i-d)/2}\left(1 - \frac{1}{r_{\text{FAC}}}\sqrt{\frac{i^i q^{d-2i+1}}{d\,(i-1)^{i-1}}}\right)$.

This is clear that the index of $g = i$ should be lower than cutting point (i.e., the condition of $i \leq \text{CUT}$). Also the condition of $r_{\text{FAC}} \geq \text{GH}(\mathcal{L}_{[1,i]})/\text{GH}(\mathcal{L}_{[1,d]})$ emphasizes that whether a solution vector v exists in lattice block $\mathcal{L}_{[1,i]}$ to use this formula in estimating the probability of index of $g = i$ or not. In this condition, without losing generality, this is assumed that the shortest norm of vector in a lattice block is not lower than Gaussian heuristic of that lattice block.

B.7 Proof of Lemma 9

Let $1 < x \leq g-1$ and $\mathcal{Y} = \frac{z_x^2}{\|v\|^2 - \|b_g^*\|^2} = \frac{\omega_x}{\sum_{t=1}^{g-1}\omega_t}$.

Our goal is computing $\text{E}[\mathcal{Y}]$ where $\omega_i \leftarrow \text{Gamma}(1/2, 2)$.

For random variable $\mathcal{Y}$, since $\mathcal{Y}>0$, the function of $\phi(\mathcal{Y}) = 1/\mathcal{Y}$ is convex, therefore by using Jensen's inequality [27], following inequality is proved as follows

$$\mathcal{Y} > 0 \Rightarrow 1/\text{E}[\mathcal{Y}] \leq \text{E}[1/\mathcal{Y}];$$
$$1/\mathcal{Y} = \frac{\sum_{t=1}^{x-1}\omega_t + \sum_{t=x+1}^{g-1}\omega_t}{\omega_x} + 1,$$

where $\omega_i \leftarrow \text{Gamma}(1/2, 2)$;

$$Y_1 = \sum_{t=1}^{x-1} \omega_t + \sum_{t=x+1}^{g-1} \omega_t \text{ and } Y_2 = \omega_x;$$

Fact 2. Let $1/\mathcal{Y} = Y_1/Y_2 + 1$, if Y_1 is independent from Y_2 then $\mathrm{E}[Y_1/Y_2] = \mathrm{E}[Y_1]\,\mathrm{E}[1/Y_2]$, and by using Jensen's inequality [27]: $\mathrm{E}[Y_1]/\mathrm{E}[Y_2] + 1 \leq \mathrm{E}[Y_1]\,\mathrm{E}[1/Y_2] + 1 \Rightarrow \mathrm{E}[Y_1]/\mathrm{E}[Y_2] + 1 \leq \mathrm{E}[1/\mathcal{Y}]$.

Approximation 1. Following with Fact 2, by using $1/\mathrm{E}[\mathcal{Y}] \leq \mathrm{E}[1/\mathcal{Y}]$ and $\mathrm{E}[Y_1]/\mathrm{E}[Y_2] + 1 \leq \mathrm{E}[1/\mathcal{Y}]$, this is assumed the approximation of $\mathrm{E}[Y_1]/\mathrm{E}[Y_2] + 1 \approx 1/\mathrm{E}[\mathcal{Y}]$.

By using Fact 2 and Approximation 1:

$\mathrm{E}[Y_1]/\mathrm{E}[Y_2] + 1 \approx \mathrm{E}\left[\sum_{t=1}^{g-1} \omega_t\right]/\mathrm{E}[\omega_x] \approx g - 1 \Rightarrow \mathrm{E}[\mathcal{Y}] \approx \frac{1}{g-1}$.

It is expected to $g \approx \text{CUT}$ by using Lemma 7, so it is expected $\mathrm{E}[\mathcal{Y}] \approx 1/\text{CUT}$ and therefore, it is expected that each entries of z approximately are similar to each other as follows $\mathrm{E}[z_x^2] \approx \frac{\|v\|^2 - \|b_g^*\|^2}{\text{CUT}}$.

B.8 Proof of Lemma 10

To prove this lemma (and Lemma 11), the expected value of $\|v\|^2/R^2$ is approximated in Remark 8.

Remark 8. For full-enumeration with success probability $P = 1$, by using (19) and (29), this is expected to $\mathrm{E}[\|v\|^2/R^2] = \mathrm{E}[\lambda_1^2/R^2] = \frac{2^{2/d}\Gamma^2(1+1/d)\,\text{GH}^2(\mathcal{L}_{[j,k]})}{\min(\Upsilon_{\text{GH}^2}(\mathcal{L}_{[j,k]}), \|b_j^*\|^2)}$ which can be simplified as $\frac{2^{2/d}\Gamma^2(1+1/d)}{\Upsilon} \leq \mathrm{E}[\|v\|^2/R^2] \leq 1$. For other success probability $0 < P < 1$, with number of solution as $K \approx \lfloor P^{\frac{r_{\text{FAC}}{}^{\beta}}{2}} \rceil$, this is expected to $\mathrm{E}[\|v\|^2/R^2] = K^2 \times \text{Beta}^2(K, (\beta+1)/\beta)$, and specially for success probability of $P = 2/r_{\text{FAC}}{}^{\beta}$, this is expected to $\mathrm{E}[\|v\|^2/R^2] = d/(d+1)$. At result, for success probability $2/r_{\text{FAC}}{}^{\beta} \leq P \leq 1$, the expectation of $\mathrm{E}[\|v\|^2/R^2]$ can be approximated by $1 - \varepsilon$ where ε nearly varied from $1/(d+1)$ up to $\frac{\Upsilon - 2^{2/d}\Gamma^2(1+1/d)}{\Upsilon}$. For much long enumeration radius, the condition of $\varepsilon \approx 1$ may be observed, while if the condition of $\|v\|^2/R^2 \approx 1 - \varepsilon$ where $\varepsilon \approx O(1/d)$ is satisfied, then this is expected to $\mathrm{E}[\|v\|^2/R^2] \approx 1 - \varepsilon$ where $\mathrm{E}[\varepsilon] \approx O(1/d)$, also the enumeration radius R is limited to be near to solution norm $\|v\|$.

Note: The cutting point in full-enumeration with bounding function $\mathcal{R} = [1, 1, \ldots, 1]$ is always $\text{CUT} = d$. also by using Lemma 7 for full-enumeration, this is expected to $g \approx d$, therefore this can be assumed to $g \approx \text{CUT}$.

Goal. For linear pruned bounding function $\mathcal{R}$, this proof tries to show the correctness of following approximation $\mathrm{E}\left[\sum_{t=1}^{i} z_t^2/R^2\right] \approx \mathcal{R}_i^2$.

For simplicity, here we introduce Approximation 2 without violating the correctness of this proof.

Approximation 2. By using Lemma 7 (also Corollary 1), this is assumed to $g \approx \text{CUT} = d$, so the expected value of $\|b_g^*\|^2$ in this lemma can be approximated by $\mathrm{E}[\|b_g^*\|^2] \approx R^2\mathcal{R}_{d-\text{CUT}+1}^2$ where $\mathcal{R}$ is the linear pruning bounding function, and consequently following estimation is used

$$\mathrm{E}[\|b_g^*\|^2/R^2] \approx \mathcal{R}_{d-\text{CUT}+1}^2 = \tfrac{d-\text{CUT}+1}{d};$$

Note: The value of $\|b_g^*\|^2$ is smaller than the solution norm $\|v\|^2$ and nearly smaller than other GSO norms $\|b_i^*\|^2$ where $1 \leq i \leq g$.

Note: For more simplicity in this proof, the variance of $\|b_g^*\|$ in Approximation 2 and $\|v\|$ in Remark 8 is assumed to be nearly 0, this means that, instead of $\mathrm{E}[\|b_g^*\|^2]$ and $\mathrm{E}[\|v\|^2]$, this proof uses the approximations of $\|b_g^*\|^2 \approx R^2\mathcal{R}_{d-\text{CUT}+1}^2$ and $\|v\|^2 \approx (1-\varepsilon)R^2$ as invariants.

The mentioned approximation in Goal of this proof can be simplified as follows

$$\mathrm{E}\left[\sum_{t=1}^{i} z_t^2/R^2\right] \overset{?}{\approx} \mathcal{R}_i^2 \Rightarrow \mathrm{E}\left[\sum_{t=d-g+2}^{i} z_t^2\right] \overset{?}{\approx} R^2\mathcal{R}_i^2 - \|b_g^*\|^2 \Rightarrow \mathrm{E}\left[\sum_{t=d-g+2}^{i} \frac{z_t^2}{\|v\|^2 - \|b_g^*\|^2}\right] \overset{?}{\approx} \frac{R^2\mathcal{R}_i^2 - \|b_g^*\|^2}{\|v\|^2 - \|b_g^*\|^2}$$

For $1 < g \leq d$ and $1 \leq x \leq g-1$,

$$X = \sum_{t=d-g+2}^{i} \frac{z_t^2}{\|v\|^2 - \|b_g^*\|^2} = \frac{\sum_{t=x}^{g-1} \omega_t}{\sum_{t=1}^{g-1} \omega_t},$$

where $\omega_i \leftarrow \text{Gamma}(1/2, 2)$.

It should be proved that whether $\mathrm{E}[X] \overset{?}{\approx} \frac{R^2\mathcal{R}_{d-x+1}^2 - \|b_g^*\|^2}{\|v\|^2 - \|b_g^*\|^2}$ for $i = d - x + 1$.

Let c_0 for linear pruned bounding function $\mathcal{R}$

$c_0(i) = \frac{\|v\|^2 - \|b_g^*\|^2}{R^2\mathcal{R}_i^2 - \|b_g^*\|^2} = \frac{\|v\|^2/R^2 - \|b_g^*\|^2/R^2}{\mathcal{R}_i^2 - \|b_g^*\|^2/R^2} \approx \frac{\text{CUT} - \varepsilon d - 1}{\text{CUT} + i - d - 1} \approx O\left(\frac{d}{i} - \frac{1}{i}\right)$,

where $\mathrm{E}[\varepsilon] \approx O(1/d)$ by Remark 8.

In other side, for random variable X, since $X > 0$, the function $\phi(X) = 1/X$ is convex, therefore by using Jensen's inequality [27].

$X > 0 \Rightarrow 1/\mathrm{E}[X] \leq \mathrm{E}[1/X]$;

$1/X = \frac{\sum_{t=1}^{x-1} \omega_t}{\sum_{t=x}^{g-1} \omega_t} + 1$. $Y_1 = \sum_{t=1}^{x-1} \omega_t$ and $Y_2 = \sum_{t=x}^{g-1} \omega_t$ are independent random variables.

By using Fact 2, Approximation 1 and Approximation 2:

$c_1(i) = 1/\mathrm{E}[X] \approx \frac{\mathrm{E}[Y_1]}{\mathrm{E}[Y_2]} + 1 \approx \frac{d-i}{\text{CUT}+i-d-1} + 1 \approx$

$O(\frac{d}{i}-1)\Rightarrow$
To complete the proof, this is needed to check whether the value of $c_0(i)$ is nearly equal to the value of $c_1(i)$ or not, as follows
$c_1(i)\approx c_0(i)-1+\text{err}$, where $\text{err}\approx O(1/i)\Rightarrow$
For $1\leq i\leq d$: $c_1(i)\approx O(c_0(i))\approx O(d/i)\Rightarrow$

Finally, by assuming that the variance of $X=\frac{\sum_{t=x}^{g-1}\omega_t}{\sum_{t=1}^{g-1}\omega_t}$ is small enough (which can be checked in the same way that be done for expected value of X), the main approximation in Goal of this proof (i.e., $\mathrm{E}\left[\sum_{t=1}^{i}z_t^2/R^2\right]\approx\mathcal{R}_i^2$) can be proved as follows

$\mathrm{E}\left[\sum_{t=1}^{i}z_t^2/R^2\right]\approx\mathrm{E}\left[\frac{(\|v\|^2-\|b_g^*\|^2)X+\|b_g^*\|^2}{R^2}\right]\approx$
$\frac{(\|v\|^2-\|b_g^*\|^2)\mathrm{E}[X]}{R^2}+\frac{\|b_g^*\|^2}{R^2}\approx\frac{\|v\|^2-\|b_g^*\|^2}{R^2c_1(i)}+\frac{\|b_g^*\|^2}{R^2}\approx$
$\frac{\|v\|^2-\|b_g^*\|^2}{R^2c_0(i)}+\frac{\|b_g^*\|^2}{R^2}\approx\mathcal{R}_i^2$.

B.9 Proof of Lemma 11

To prove this lemma, the expected value of $\|v\|^2/R^2$ is approximated in Remark 8.

Note: It is expected to $g\approx\text{CUT}\approx d$ by using Lemma 7, Lemma 8, Corollary 1.

Goal. For piecewise-linear pruned bounding function $\mathcal{R}$, this proof tries to show the correctness of following approximation: $\mathrm{E}\left[\sum_{t=1}^{i}z_t^2/R^2\right]\approx\mathcal{R}_i^2$.

For simplicity, here we introduce Approximation 3 without violating the correctness of this proof.

Approximation 3. By using Corollary 1, this is assumed to $g\approx\text{CUT}\approx d$, so the expected value of $\|b_g^*\|^2$ in this lemma can be approximated by $\mathrm{E}\left[\|b_g^*\|^2\right]\approx R^2\mathcal{R}_{d-\text{CUT}+1}^2$ where $\mathcal{R}$ is the piecewise-linear bounding function with parameter $\mathbf{a}$, and consequently following estimation is used

$$\mathrm{E}\left[\|b_g^*\|^2/R^2\right]\approx\mathcal{R}_{d-\text{CUT}+1}^2=\frac{2\mathbf{a}(d-\text{CUT}+1)}{d};$$

Note: The value of $\|b_g^*\|^2$ is smaller than the solution norm $\|v\|^2$ and nearly smaller than other GSO norms $\|b_i^*\|^2$ where $1\leq i\leq g$.

Note: For more simplicity in this proof (similar to proof of Lemma 10), the variance of $\|b_g^*\|$ in Approximation 3 and $\|v\|$ in Remark 8 is assumed to be nearly 0, this means that, instead of $\mathrm{E}\left[\|b_g^*\|^2\right]$ and $\mathrm{E}\left[\|v\|^2\right]$, this proof uses the approximations of $\|b_g^*\|^2\approx R^2\mathcal{R}_{d-\text{CUT}+1}^2$ and $\|v\|^2\approx(1-\varepsilon)R^2$ as invariants.

The mentioned approximation in Goal of this proof can be simplified as follows

$\mathrm{E}\left[\sum_{t=1}^{i}z_t^2/R^2\right]\overset{?}{\approx}\mathcal{R}_i^2\Rightarrow\mathrm{E}\left[\sum_{t=d-g+2}^{i}z_t^2\right]\overset{?}{\approx}R^2\mathcal{R}_i^2-\|b_g^*\|^2\Rightarrow\mathrm{E}\left[\sum_{t=d-g+2}^{i}\frac{z_t^2}{\|v\|^2-\|b_g^*\|^2}\right]\overset{?}{\approx}\frac{R^2\mathcal{R}_i^2-\|b_g^*\|^2}{\|v\|^2-\|b_g^*\|^2}$.

Note: The reasonable and effective piecewise-linear parameter $\mathbf{a}$ even for extreme-pruning is nearly assumed to be $\mathbf{a}>0.1$ (this note is used in approximating the expressions in this proof).

The reminder of this proof can be proceeded in following three steps.

***Step 1*:** For $d/2<g\leq d$ and $d/2<x\leq g-1$ let
For $i=d-x+1$, following approximation should be proved

$$X=\sum_{t=d-g+2}^{i}\frac{z_t^2}{\|v\|^2-\|b_g^*\|^2}=\frac{\mathbf{a}\sum_{t=x}^{g-1}\omega_t}{(1-\mathbf{a})\sum_{t=1}^{d/2}\omega_t+\mathbf{a}\sum_{t=d/2+1}^{g-1}\omega_t},$$

where $\omega_i\leftarrow\text{Gamma}(1/2,2)$.
It should be proved whether $\mathrm{E}[X]\overset{?}{\approx}\frac{R^2\mathcal{R}_{d-x+1}^2-\|b_g^*\|^2}{\|v\|^2-\|b_g^*\|^2}$.
Let c_0 for piecewise-linear bounding function $\mathcal{R}$

$$c_0(i)=\frac{\|v\|^2-\|b_g^*\|^2}{R^2\mathcal{R}_i^2-\|b_g^*\|^2}=\frac{\|v\|^2/R^2-\|b_g^*\|^2/R^2}{\mathcal{R}_i^2-\|b_g^*\|^2/R^2}\approx\frac{d-\varepsilon d-2\mathbf{a}d+2\mathbf{a}\text{CUT}-2\mathbf{a}}{2\mathbf{a}i-2\mathbf{a}d+2\mathbf{a}\text{CUT}-2\mathbf{a}},$$

where $\mathrm{E}[\varepsilon]\approx O(1/d)$, and consequently by Remark 8,

$$c_0(i)\approx\frac{d-\varepsilon d-2\mathbf{a}}{2\mathbf{a}i-2\mathbf{a}}\approx O(\tfrac{d}{i}-\tfrac{1}{i})\qquad(63)$$

In other side, for random variable X, since $X>0$, the function $\phi(X)=1/X$ is convex, therefore by using Jensen's inequality [27]
$X>0\Rightarrow 1/\mathrm{E}[X]\leq\mathrm{E}[1/X]$;

$$1/X=\frac{(1-\mathbf{a})\sum_{t=1}^{d/2}\omega_t+\mathbf{a}\sum_{t=d/2+1}^{x-1}\omega_t}{\mathbf{a}\sum_{t=x}^{g-1}\omega_t}+1;$$

$Y_1=(1-\mathbf{a})\sum_{t=1}^{d/2}\omega_t+\mathbf{a}\sum_{t=d/2+1}^{x-1}\omega_t$ and $Y_2=\mathbf{a}\sum_{t=x}^{g-1}\omega_t$ are independent random variables.
By using Fact 2, Approximation 1, Approximation 3
$c_1(i)=\frac{1}{\mathrm{E}[X]}\approx\frac{\mathrm{E}[Y_1]}{\mathrm{E}[Y_2]}+1\approx\frac{d-2\mathbf{a}i}{2\mathbf{a}i-2a}\approx O(\frac{d}{i}-1)$
To complete the proof of this step, this is needed to check whether the value of $c_0(i)$ is nearly equal to the value of $c_1(i)$ or not, as follows
$c_1(i)\approx c_0(i)-1+\text{err}$, where $\text{err}\approx O(1/i)\Rightarrow$
For $1\leq i<d/2+1$ in this step $c_1(i)\approx O(c_0(i))\approx O(d/i)$.
Finally, by assuming that the variance of $X=$

$\frac{\mathbf{a}\sum_{t=x}^{g-1}\omega_t}{(1-\mathbf{a})\sum_{t=1}^{d/2}\omega_t+\mathbf{a}\sum_{t=d/2+1}^{g-1}\omega_t}$ is small enough (which can be checked in the same way that be done for expected value of X), the main approximation in Goal of this proof, as $\mathrm{E}\left[\sum_{t=1}^{i} z_t^2/R^2\right] \approx \mathcal{R}_i^2$, for $1 \le i < d/2+1$, can be proved as follows

$\mathrm{E}\left[\sum_{t=1}^{i} z_t^2/R^2\right] \approx \mathrm{E}\left[\frac{(\|v\|^2-\|b_g^*\|^2)X+\|b_g^*\|^2}{R^2}\right] \approx \frac{(\|v\|^2-\|b_g^*\|^2)\mathrm{E}[X]}{R^2} + \frac{\|b_g^*\|^2}{R^2} \approx \frac{\|v\|^2-\|b_g^*\|^2}{R^2 c_1(i)} + \frac{\|b_g^*\|^2}{R^2} \approx \frac{\|v\|^2-\|b_g^*\|^2}{R^2 c_0(i)} + \frac{\|b_g^*\|^2}{R^2} \approx \mathcal{R}_i^2;$

The proof of this step is completed.

Step 2: For $d/2 < g \le d$ and $1 \le x \le d/2$ let

For $i = d-x+1$, following approximation should be proved $X' = \sum_{t=d-g+2}^{i} \frac{z_t^2}{\|v\|^2-\|b_g^*\|^2} = \frac{\mathbf{a}\sum_{t=d/2+1}^{g-1}\omega_t+(1-\mathbf{a})\sum_{t=x}^{d/2}\omega_t}{(1-\mathbf{a})\sum_{t=1}^{d/2}\omega_t+\mathbf{a}\sum_{t=d/2+1}^{g-1}\omega_t}$, where $\omega_i \leftarrow \mathrm{Gamma}(1/2,2)$

It should be proved that whether $\mathrm{E}\left[X'\right] \stackrel{?}{\approx} \frac{R^2\mathcal{R}_{d-x+1}^2-\|b_g^*\|^2}{\|v\|^2-\|b_g^*\|^2}$ or not.

Let c_0 for piecewise-linear bounding function $\mathcal{R}$

$c_0(i) = \frac{\|v\|^2-\|b_g^*\|^2}{R^2\mathcal{R}_i^2-\|b_g^*\|^2} = \frac{\|v\|^2/R^2-\|b_g^*\|^2/R^2}{\mathcal{R}_i^2-\|b_g^*\|^2/R^2} \approx \frac{d-\varepsilon d-2\mathbf{a}d+2\mathbf{a}\mathrm{CUT}-2\mathbf{a}}{-d+2i-2i\mathbf{a}+2\mathbf{a}\mathrm{CUT}-2\mathbf{a}},$

where $\mathrm{E}[\varepsilon] \approx O(1/d)$ by Remark 8.

$$c_0(i) \approx \frac{d-\varepsilon d-2\mathbf{a}}{-d+2i-2i\mathbf{a}+2\mathbf{a}\mathrm{CUT}-2\mathbf{a}} \approx O\left(\frac{d-\varepsilon d}{-d+2i+2\mathbf{a}d}\right) \quad (64)$$

In other side, by using Jensen's inequality [27]

$X' > 0 \Rightarrow 1/\mathrm{E}\left[X'\right] \le \mathrm{E}\left[1/X'\right];$

$1/X' = \frac{(1-\mathbf{a})\sum_{t=1}^{x-1}\omega_t}{(1-\mathbf{a})\sum_{t=x}^{d/2}\omega_t+\mathbf{a}\sum_{t=d/2+1}^{g-1}\omega_t} + 1;$

$Y_3 = (1-\mathbf{a})\sum_{t=1}^{x-1}\omega_t$ and $Y_4 = (1-\mathbf{a})\sum_{t=x}^{d/2}\omega_t + \mathbf{a}\sum_{t=d/2+1}^{g-1}\omega_t$ are independent random variables.

By using Fact 2, Approximation 1, Approximation 3,

$c_1(i) = 1/\mathrm{E}\left[X'\right] \approx \frac{\mathrm{E}[Y_3]}{\mathrm{E}[Y_4]} + 1 \approx \frac{d}{-d+2i-2i\mathbf{a}+2\mathbf{a}\mathrm{CUT}-2\mathbf{a}} \approx O\left(\frac{d}{-d+2i+2\mathbf{a}d}\right)$

To complete the proof of this step, this is needed to check whether the value of $c_0(i)$ is nearly equal to the value of $c_1(i)$ or not, as follows.

$c_1(i) \approx c_0(i) + \mathrm{err}$, where $\mathrm{err} \approx O(\frac{1}{-d+2i+2\mathbf{a}d}) \Rightarrow$

For $d/2+1 \le i \le d$ in this step: $c_1(i) \approx O(c_0(i)) \approx O(\frac{d}{-d+2i+2\mathbf{a}d})$.

Finally, by assuming that the variance of $X' = \frac{\mathbf{a}\sum_{t=d/2+1}^{g-1}\omega_t+(1-\mathbf{a})\sum_{t=x}^{d/2}\omega_t}{(1-\mathbf{a})\sum_{t=1}^{d/2}\omega_t+\mathbf{a}\sum_{t=d/2+1}^{g-1}\omega_t}$ is small enough (which can be checked in the same way that be done for expected value of X'), the main approximation in Goal of this proof, as $\mathrm{E}\left[\sum_{t=1}^{i} z_t^2/R^2\right] \approx \mathcal{R}_i^2$, for $d/2+1 \le i \le d$, can be proved as follows

$\mathrm{E}\left[\sum_{t=1}^{i} z_t^2/R^2\right] \approx \mathrm{E}\left[\frac{(\|v\|^2-\|b_g^*\|^2)X'+\|b_g^*\|^2}{R^2}\right] \approx \frac{(\|v\|^2-\|b_g^*\|^2)\mathrm{E}\left[X'\right]}{R^2} + \frac{\|b_g^*\|^2}{R^2} \approx \frac{\|v\|^2-\|b_g^*\|^2}{R^2 c_1(i)} + \frac{\|b_g^*\|^2}{R^2} \approx \frac{\|v\|^2-\|b_g^*\|^2}{R^2 c_0(i)} + \frac{\|b_g^*\|^2}{R^2} \approx \mathcal{R}_i^2;$

The proof of this step is completed.

Step 3: For $1 < g \le d/2$ and $1 \le x \le d/2$ let

By using Lemma 8, the probability of the condition of $1 < g \le d/2$ is zero! Therefore, this step refers to a special case which is rarely expected to be observed (and occurred), and consequently the effect of this step can be ignored in checking the accuracy of our approximation of the random variable $X_i = \sum_{t=1}^{i} z_t^2/R^2$ in Lemma 11. Therefore this step is not considered in the proof;

At results, by using these three steps, for $1 \le i \le d$, this is proved that the expected value of $X_i = \sum_{t=1}^{i} z_t^2/R^2$ is closely approximated by $\mathcal{R}_i^2$.