Journal of Electrical and Computer Engineering Innovations
(JECEI)
Journal homepage: http://www.jecei.sru.ac.ir

Research paper

Revised Estimations for Cost and Success Probability of GNR-Enumeration

G. R. Moghissi*, A. Payandeh

Department of ICT, Malek-Ashtar University of Technology, Tehran, Iran.

Article Info

Article History:
Received 12 January 2023
Reviewed 07 March 2023
Revised 25 April 2023
Accepted 07 May 2023

Keywords:
BKZ simulation
Enumeration cost
Success probability
Optimal enumeration radius
Bounding function generator

*Corresponding Author's Email Address: *fumoghissi@chmail.ir*

Abstract

Background and Objectives: Since exact manner of BKZ algorithm for higher block sizes cannot be studied by practical running, therefore simulation of BKZ is used to predict the total cost of BKZ and quality of output basis. This paper revises some main components of BKZ-simulation for better predictions.
Methods: At first, by definition of full-enumeration success probability, the optimal enumeration radius is formally defined. Next, this paper defines three more pruning types, besides the well-known pruning by bounding function in GNR-enumerations, and consequently uses these four pruning types collectively in revision of success probability estimation. Also, by using these four pruning types and the process of updating-radius, this paper revises the estimation of enumeration cost. Finally, this paper introduces a simple technique to generate partially better bounding functions.
Results: For block sizes of $50 \leq \beta \leq 240$, better domains of radius parameters in GNR enumeration are formally introduced. Also, our revised estimation of success probability (for GNR bounding function) in our test results shows non-negligible gap from former estimations in some main former studies. Moreover, our results show that the cost results by our proposed estimator of GNR-enumeration cost are closer to the cost results determined in experimental running of enumeration, than the cost results by Chen-Nguyen estimator.
Conclusion: This paper revises the estimators of cost and success probability for GNR-Enumeration, and justifies the value of these revised estimators by sufficient test results (in actual running and simulation of BKZ). Also, our novel definition of optimal enumeration radius can be used effectively in actual running and simulation of BKZ.

Introduction

Lattice reduction is the main part of most lattice security attacks. BKZ algorithm is one of the main practical lattice reductions. The security parameters in lattice-based cryptographic primitives are estimated by determining the total cost and output quality of BKZ algorithm 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. There are some claimant BKZ-simulations in former studies, such as the one is introduced by Chen and Nguyen [1], the simulation by Shi Bai et al. [2], and the simulation by Aono et al. [3]; The outputs of BKZ-simulation are divided by two main parts as total cost and output quality which can be used in lattice based security analysis.

The cost of enumeration function on the lattice block of $\mathcal{L}_{[1\ldots\beta]}$ can be estimated by old version of [1] as $N = 2^{0.00405892\,\beta^2 - 0.337913\,\beta + 34.9018}$ or by [4] as $N = 2^{0.000784\,\beta^2 - 0.366\,\beta - 0.9}$; This is obvious that using exact versions of "success probability estimators",

"enumeration cost estimators", "minimum effective (optimal) enumeration radius" and "bounding function generator", which are revised in this paper, can make such these cost models more exact and more close to the practical estimations. To the best of our knowledge, the technique of GNR-enumeration and corresponding concepts studied in [1], [13], is considered yet in current studies and security estimations of Lattice based cryptography, however other techniques such as sieve algorithm, discrete pruning and RSR algorithm may show better results in practical attacks. In fact, the significance of our contributions in this paper for bit-security estimation of lattice-based cryptographic primitives can be more justified by the massive efforts of Albrecht et al., in estimation of the LWE/NTRU schemes [5] (see the user-friendly scripts for these estimations in [6]). For example, the cost of primal attack against "Falcon-1024-2.87-12289" (with claimed bit-security of 230) by using enumeration with four different cost models in Table 10 from [5] is estimated as 2^{418}, 2^{474}, 2^{836} and 2^{1118}! The authors of this paper believe that using non-exact components and definitions on enumeration functions lead to these gaps in bit-security estimations, while our contributions in this paper try to fix the problem of such non-exactness.

It is worthy of noting that the results of [5] are massively used in "Post-Quantum Cryptography Standardization Project" (see the corresponding information in [7]), also in bit-security estimations of current cryptography researches, such as [8]-[11].

In other view on this work, designing a BKZ-simulation with GNR-pruned enumeration needs to some necessary building-blocks which include 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. Our previous study in [12] focuses on design of sampling method for enumeration solution (as solution norm and coefficient vectors). This paper introduces some main revisions for following components: optimal enumeration radius, generation of bounding function, estimation of success probability and GNR enumeration cost. The components which are studied in this paper (except estimation of enumeration cost) can be used in actual running of BKZ algorithm (besides the simulation of BKZ) too! Our contributions in this paper are described briefly as follows:

- By definition of full-enumeration success probability in this paper, the optimal value for radius parameter $\sqrt{\Upsilon}$ (as initial radius factor r_{FAC}) and corresponding bound for solution norm of full-enumeration are defined exactly in average-case. This definition can be used dynamically to compute optimal enumeration radius in BKZ simulation and even actual running of BKZ algorithm. In other sides, former studies on BKZ-simulation [1]-[3] don't use optimal version of the radius parameter of r_{FAC}.
- The former studies [1]-[3] use the efficient idea by [1] to estimate the success probability of GNR-enumerations which only consider the pruning type by cylinder-intersection of bounding function; This paper proposes three more types of pruning in estimation of success probability;
- The former studies [1]-[3] use the efficient idea by [1] to estimate the cost of GNR-enumerations which only consider the pruning type by cylinder-intersection of bounding function; Similar to our revision for success probability, this paper considers all of four types of pruning along with the process of updating enumeration radius in our estimation of GNR-enumeration cost;
- This paper introduces a generator of bounding function including cutting point of $\mathrm{Cut} = d$ [12]; In former studies [1]-[3], if the simulation tries to generate bounding functions with much small success probability, this is possible that the success probability of this bounding functions unintentionally becomes much less than intended value or even zero!

The remainder of this paper is organized as follows. Second section is dedicated to essential background for our contributions in this paper. In third section, we describe our contributions as follows:

- In third section (Part A), the optimal enumeration radius is defined exactly;
- In third section (Part B), our estimation of success probability is introduced;
- In third section (Part C), our estimation of GNR-enumeration cost is introduced;
- In third section (Part D), a simple technique for forcing $\mathrm{Cut} = \beta$ in generation of bounding function is introduced.

Also, our test results for these contributions are introduced in fourth section. Finally, in fifth section, the conclusion for this work is expressed.

Background

In this section, the needed preliminaries on theory of lattice, BKZ-reduction and other corresponding concepts for this work are introduced.

A. Basic Definitions and Notations

In this section, some basic concepts, needed in this paper, are defined.

Lattices. For n-linearly independent vectors of $b_1, \dots, b_n \in \mathbb{R}^m$, the lattice generated by these vectors is defined as following set:

$$\mathcal{L}(b_1, \dots, b_n) = \{\sum_{i=1}^{n} x_i b_i : x_i \in \mathbb{Z}\}. \quad (1)$$

The set of vectors $[b_1, \dots, b_n]$ is known as a lattice basis which is usually shown by a column-matrix B where $b_i \in \mathbb{Z}^m$ for cryptographic applications. Also, the rank and dimension of lattice $\mathcal{L}(B)$ are respectively shown by n and m. In this paper, the notation of $\mathcal{L}_i$ is defined as follows:

$$\mathcal{L}_i = [b_1, b_2, \dots, b_i]. \quad (2)$$

Euclidean norm. The length of a lattice vector $v = (v_1, \dots, v_m)$ is measured by $\|v\| = \sqrt[2]{{v_1}^2 + \cdots + {v_m}^2}$. In this paper, the phrases of "norm" and "length" refer to Euclidean norm.

Volume of Lattices. The– volume of a lattice $\mathcal{L}(B)$ is defined by determinant of basis matrix:

$$\mathrm{Vol}\big(\mathcal{L}(B)\big) = |\det B|. \quad (3)$$

First Successive-Minima of lattice $\mathcal{L}$. The norm of shortest nonzero vector in lattice $\mathcal{L}$ is first successive-minima of that lattice and is shown by $\lambda_1(\mathcal{L})$.

In the worst-case of the SVP solver, the optimal (smallest) value of Hermite-factor for all n-dimensional input lattice bases are defined formally as follows:

Hermite's constant. Hermite's constant $\gamma_n$is supremum of the ratio $\big(\lambda_1(\mathcal{L})/\mathrm{Vol}(\mathcal{L})^{1/n}\big)^2$ over all n-dimensional lattices.

By sterling approximation for high dimensional space, volume of a n-dimensional sphere (ball) is computed as follows:

$$V_n(R) = \mathrm{Vol}\big(Ball_n(R)\big) = \frac{\pi^{\frac{n}{2}}}{\Gamma\left(\frac{n}{2}+1\right)} R^n \approx \frac{1}{\sqrt{n\,\pi}}\left(\frac{2\pi e}{n}\right)^{\frac{n}{2}} R^n. \quad (4)$$

In this paper, $V_l(R)$ refers to the 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 $\Gamma(n/2+1) \approx \sqrt{n\pi}(\frac{n}{2e})^{n/2}$;

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

Heuristic 1 (Gaussian Heuristic). *"Given a lattice* $\mathcal{L}$ *and a set* S*, the number of points in* $S \cap \mathcal{L}$ *is approximated by* $Vol(S)/Vol(\mathcal{L})$*"* [13];

By using Gaussian Heuristic, if a lattice $\mathcal{L}$ is limited in a centered ball with radius of $R = \lambda_1(\mathcal{L})$, then it is expected that there is at least one lattice vector in $Ball_n(R)$ with radius R, which is the shortest vector. Therefore, the value of $\lambda_1(\mathcal{L})$ can be estimated by Gaussian Heuristic of this lattice as follows (by using sterling approximation):

$$\mathrm{GH}(\mathcal{L}) = \left(\frac{\mathrm{Vol}(\mathcal{L}(B))}{\mathrm{Vol}(Ball_n(1))}\right)^{\frac{1}{n}} \approx \sqrt{\frac{n}{2\pi e}}\,(\det B)^{\frac{1}{n}}. \quad (5)$$

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

$$\pi_i(b_i) = b_i^* = b_i - \textstyle\sum_{j=1}^{i-1} \mu_{i,j} b_j^*, \quad (6)$$

where $\mu_{i,j} = \frac{b_i b_j^*}{\|b_j^*\|^2}$ *and* $1 \le j < i \le n$.

The parameter of $\mu_{i,j} \in \mathbb{R}$ is named a GSO coefficient and b_i^* refers to i-th vector of GSO basis of B^*. For an input lattice basis B, the volume of the lattice can be computed by the norm of GSO vectors as follows:

$$\mathrm{Vol}\big(\mathcal{L}(B)\big) = \textstyle\prod_{i=1}^{n} \|b_i^*\|. \quad (7)$$

Other important heuristic in lattice theory is Schnorr's Geometric Series Assumption (GSA) which is defined as follows:

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

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

$$\mathrm{Gamma}(x; k, \theta) = \frac{x^{k-1} e^{-x/\theta}}{\Gamma(k)\,\theta^k}, \quad \text{where } x > 0. \quad (8)$$

Exponential distribution. The Exponential distribution, which is a one-parameter and continuous probability distribution, is defined as follows (for input parameter λ):

$$\mathrm{Expo}(x; \lambda) = \lambda e^{-\lambda x}, \quad \text{where } x > 0 \text{ and } \lambda > 0. \quad (9)$$

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

Note: The notation of 1^β represents a vector with length of β as a bounding function with entries of 1.

B. Enumeration and GNR-Pruning

In this paper, for each lattice block of $\mathcal{L}_{[j,k]} = \mathcal{L}(b_j, b_{j+1}, \dots, b_k)$, the block size $\beta = k - j + 1$ is assumed sufficiently big. Also since these lattice blocks are assumed to be used in BKZ algorithms, in fact, the notation of $\mathcal{L}(b_j, b_{j+1}, \dots, b_k)$ refers to the projected form of $\pi_j(b_j, b_{j+1}, \dots, b_k)$, as a lattice block from index j to k, while its vectors are projected on the vectors of $(b_1, b_2, \dots, b_{j-1})$.

Full-enumeration. For fixed enumeration radius R (by no updating radius), the tree of full-enumeration includes all lattice points in n-dimensional ball of radius R.

Full-enumeration Cost. For fixed enumeration radius R (by no updating radius), the number of total nodes of the full-enumeration tree can be estimated as follows [1]:

$$N \approx \textstyle\sum_{l=1}^{k-j+1=d} H_l, \quad (10)$$

where

$$H_l = \frac{1}{2}\frac{V_l(R)}{\prod_{i=d-l+1}^{d}\|b_i^*\|} = \frac{1}{2}\frac{R^l.V_l(1)}{\prod_{i=d-l+1}^{d}\|b_i^*\|}. \quad (11)$$

The value of H_l represents the Gaussian Heuristic prediction of the number of nodes at the level l (see [1], [13]).

Note: In this paper, "enumeration cost", "total nodes of GNR-enumeration" and "number of enumeration tree nodes" are referred to N as the number of total nodes in the tree [13].

The concepts of cylinder-intersection, bounding function and GNR-pruning formally are defined as follows:

Cylinder-intersection. The l-dimensional cylinder-intersection with radius of $(R_1, \dots, R_l)$ is defined as follows [13]:

$$C_{R_1 \dots R_l} = \{(x_1, \dots, x_l) \in \mathbb{R}^l, \forall\, 1 \le i \le l, \sum_{t=1}^{i} {x_t}^2 \le {R_i}^2\}. \quad (12)$$

Bounding function. The vector of $\mathcal{R} = [\mathcal{R}_1, \mathcal{R}_2, \dots, \mathcal{R}_\beta]$ where $0 \le \mathcal{R}_1 \le \mathcal{R}_2 \le \cdots \le \mathcal{R}_\beta = 1$, when multiplied by initial radius of R, 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 tree [13].

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 $\|\pi_{k+1-i}(x.B_{[j,k]})\| \le R$ for $1 \le i \le k-j+1$ as a bounded ball (in full-enumeration) by $\|\pi_{k+1-i}(x.B_{[j,k]})\| \le \mathcal{R}_i \times R$, where $0 \le \mathcal{R}_1 \le \cdots \le \mathcal{R}_{k-j+1}$=1 as a cylinder-intersection [1].

The pseudo-code of the GNR pruned enumeration is shown in Appendix B from [13]. Based on the definition of GNR-pruning, this paper uses the concepts of final solution vector (usually referred to solution vector) and partial solution candidate 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 $\pi_j(v) = \pi_j(x.B_{[j,k]})$ which satisfies all of the conditions of $\|\pi_{k+1-i}(x.B_{[j,k]})\| \le \mathcal{R}_i \times R$ for $1 \le i \le k-j+1$, is a final solution vector.

Note: In this paper, $\pi_j(v)$ is shown by the notation of v for simplicity.

Fact 1 is an obvious proposition on GNR pruned enumeration.

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

Partial solution candidate. For a lattice block of $B_{[j,k]} = (b_j, b_{j+1}, \dots, b_k)$, the coefficient vector $x \in \mathbb{Z}^\beta$ and the projection level of ℓ for $1 \le \ell \le k-j+1$, the projected vector of $\pi_{k+1-i}(v) = \pi_{k+1-i}(x.B_{[j,k]})$ which satisfies all of the conditions of $\|\pi_{k+1-i}(x.B_{[j,k]})\| \le \mathcal{R}_i \times R$ for $1 \le i \le k-\ell+1$, is a partial solution candidate at the level of ℓ in enumeration tree;

The success probability is one of the main features of bounding function which can be defined as follows [13]:

Success probability of bounding function: For any lattice block of $\mathcal{L}_{[j,k]} = [b_j, b_{j+1}, \dots, b_k]$, 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 $p_{succ}(\mathcal{R})$.

For analysis of the success probability of GNR bounding function, Gama et al. use following heuristic [13]:

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 $\|v\|$"*;

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 [13]:

$$v = [z_1, \cdots, z_d] \begin{bmatrix} b_k^*/\|b_k^*\| \\ \vdots \\ b_j^*/\|b_j^*\| \end{bmatrix} = (v_1, \cdots, v_m). \quad (13)$$

where $b_i^*/\|b_i^*\|$ is i-th vector of the orthonormal basis of $b_1^*/\|b_1^*\|, \dots, b_n^*/\|b_n^*\|$) [13]. Also, the solution vector v can be written by the coefficient vector $w = (z_d/\|b_1^*\|, \dots, z_2/\|b_{d-1}^*\|, z_1/\|b_d^*\|)$ on the GSO block basis as follows [12]:

$$v = (v_1, \cdots, v_m) = (w_1, \cdots, w_d) \begin{bmatrix} b_1^* \\ \vdots \\ b_d^* \end{bmatrix}. \quad (14)$$

The coordinates of the coefficient vector z are reversed (i.e., z_i corresponds to $b_{k-i+1}^*/\|b_{k-i+1}^*\|$), and it is clear that $\|z\| = \|v\|$ [13]. 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 $u \backsim Ball_d$). By using these formulations, success probability of a GNR bounding function $\mathcal{R}$ can be defined as follows [13]:

$$p_{succ}(\mathcal{R}) = \Pr_{u \backsim Ball_d}\left(\forall\, i \in [1,d], \sum_{l=1}^{i} {u_l}^2 \le \frac{{R_i}^2}{{R_d}^2}\right) = \Pr_{u \backsim Ball_d}\left(\forall\, i \in [1,d], \sum_{l=1}^{i} {u_l}^2 \le {\mathcal{R}_i}^2\right). \quad (15)$$

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

C. Cost of GNR-enumeration

The estimation of total nodes in GNR pruned enumeration tree is the same as the full-enumeration (Schnorr-Euchner enumeration), except that instead of using balls of radius R, GNR pruned enumeration employs the cylinder-intersections of radius $(R_1, \dots, R_l) = (R \times \mathcal{R}_1, \dots, R \times \mathcal{R}_l)$ for $1 \leq l \leq \beta$. In reminder of this paper, the enumeration radius is determined by parameter of r_{FAC}, as follows:

$$R = r_{\text{FAC}} \times GH(\mathcal{L}). \tag{16}$$

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

$$H'_l = \frac{1}{2}\frac{V_{R_1,\dots,R_l}}{\prod_{i=k-l+1}^{k}\|b_i^*\|} = \frac{1}{2}\frac{R^l V_{\mathcal{R}_1,\dots,\mathcal{R}_l}}{\prod_{i=k-l+1}^{k}\|b_i^*\|}. \tag{17}$$

The volume of cylinder-intersection of $C_{R_1,\dots,R_l}$ can be defined as follows:

$$V_{R_1,\dots,R_l} = \text{Vol}\left(C_{R_1,\dots,R_l}\right) = V_l(R) \times \Pr_{u \backsim Ball_l}\left(\forall j \in [1,l], \sum_{i=1}^{j} {u_i}^2 \leq {\mathcal{R}_j}^2\right). \tag{18}$$

Therefore, the total number of nodes in the GNR pruned enumeration tree can be estimated as follows:

$$N'\left(\mathcal{L}_{[j,k]}, \mathcal{R}', R\right) \approx \sum_{l=1}^{k-j+1} {H'}_l \approx \sum_{l=1}^{\beta} \Pr_{u \backsim Ball_l}\left(\forall j \in [1,l], \sum_{i=1}^{j} {u_i}^2 \leq {\mathcal{R}_j}^2\right) \times H_l. \tag{19}$$

Note: In this paper, the total number of nodes in full-enumeration tree and number of nodes in level l of full-enumeration tree are shown by N and H_l, while the total number of nodes in GNR pruned enumeration tree and number of nodes in level l of GNR pruned enumeration tree are shown by N' and H'_l.

As shown in Section 3.3 from [13], the most populated level of full-enumeration tree is middle-level. By assuming the populated level in middle-level, paper [13] concludes the approximation of $N'\left(\mathcal{L}_{[1,\beta]}, \mathcal{R}', R\right) \approx {H'}_{l=\beta/2}$ for all GNR-enumeration (not just for full-enumeration). To the best of our knowledge, by using this approximation for well-defined bounding function of Linear pruned, paper [13] concludes that the total cost of enumeration pruned by an optimal bounding function with success probability $\approx \%100$ tends to $\frac{1}{2^{\beta/4}}$ times of total cost of full-enumeration; In other side, by using this approximation for some well-defined bounding functions of Piecewise-Linear and Step bounding function, paper [13] tries to show that for extremely small success probability, the total cost of enumeration pruned by these two bounding functions (as extreme-pruning) tends to $\frac{1}{2^{\beta/2}}$ times of total cost of full-enumeration. Our analysis in [14] proves the assumption of most populated level of $l = \beta/2$ and speedup of $\frac{1}{2^{\beta/4}}$ for an optimal bounding function with success probability $\approx \%100$, but rejects the assumption of most populated level of $l = \beta/2$ and speedup of $\frac{1}{2^{\beta/2}}$ for Piecewise-Linear bounding function for extremely small success probability.

The success probability of the bounding function $\mathcal{R}$ can be estimated by Monte-Carlo simulation (see Algorithm 8 in [1]) which is used in some test results of this paper, but it is not efficient since the number of samples required for this estimation is proportional to $\frac{1}{p_{succ}(\mathcal{R})}$ [13]. The Monte-Carlo estimation of success probability is defined by the number of $\frac{1}{p_{succ}(\mathcal{R})}$ samples of random vector $u \backsim Ball_d$, and counting the success of each sample satisfying the bounding function constraints which is defined as follows [1]:

$$\forall\, i \in [1,d], \sum_{l=1}^{i} z_l^2 \leq {\mathcal{R}_i}^2 {R_d}^2 \equiv$$

$$\forall\, i \in [1,d], \sum_{l=1}^{i} {u_l}^2 \leq \frac{{R_i}^2}{{R_d}^2} = {\mathcal{R}_i}^2 \equiv$$

$$\forall\, i \in [1,d], \sum_{l=1}^{i} \frac{\omega_{d-l+1}}{\sum_{t=1}^{d}\omega_t} \leq {\mathcal{R}_i}^2, \tag{20}$$

where $\omega_i \leftarrow \text{Gamma}(1/2,2)$ and $R = R_d$ is the enumeration radius. Some speedup for Monte-Carlo estimation of $p_{succ}(\mathcal{R})$ can be introduced by replacing the ball with a smaller containing body whose volume is known and also the vector u can be sampled uniformly from it [13]. Moreover, some cases are noted in [13], where the volume $V_{\mathcal{R}_1,\dots,\mathcal{R}_l}$ can be computed exactly. In these cases, when vector u is sampled from $Ball_l$, the distribution of vector $({u_1}^2 + {u_2}^2, {u_3}^2 + {u_4}^2, \dots, {u_{l-1}}^2 + {u_l}^2)$ can be given by a Dirichlet distribution with the parameters of $\frac{l}{2} + 1$ ones, which are simply a uniform distribution over the set of all vectors whose coordinates are non-negative and summed to at most 1 (see page 593 of [15]).

Accordingly, in this particular case, some conditions should be assumed, such as $\mathcal{R}_1 = \mathcal{R}_2$, $\mathcal{R}_3 = \mathcal{R}_4$, …, $\mathcal{R}_{d-1} = \mathcal{R}_d$, where $0 \leq \mathcal{R}_1 \leq \mathcal{R}_3 \leq \cdots \leq \mathcal{R}_{d-1}$ and even number of block sizes $d = \beta = 2\ell$ [13]. In Appendix A of [1], it is shown that for any vector $(t_1, \dots, t_\ell) \in \mathbb{R}_{\geq 0}^{\ell}$, the related polytope is denoted by $\mathcal{P}_\ell(t_1, \dots, t_\ell)$, which is defined as [1]: $\mathcal{P}_\ell(t_1, \dots, t_\ell) = \left\{(x_1, \dots, x_\ell) \in \mathbb{R}^\ell \mid \forall i \in \{1, \dots, \ell\}, x_i \geq 0 \text{ and } \sum_{j=1}^{i} x_j \leq t_i\right\}$. The volume of $\mathcal{P}_\ell(t_1, \dots, t_\ell)$ is computed as follows:

$$\text{Vol}\mathcal{P}_\ell(t_1, \dots, t_\ell) =$$

$$\int_{x_1=0}^{t_1}\int_{x_2=0}^{t_2-x_1}\dots\int_{x_\ell=0}^{t_\ell-\sum_{i=1}^{\ell-1}x_i} dx_\ell \dots dx_2\, dx_1 \xrightarrow{y_i=\sum_{j=1}^{i}x_j}$$

$$Vol\mathcal{P}_\ell(t_1, \dots, t_\ell) =$$

$$\int_{y_1=0}^{t_1}\int_{y_2=y_1}^{t_2}\dots\int_{y_\ell=y_{\ell-1}}^{t_\ell} dy_\ell \dots dy_2\, dy_1. \tag{21}$$

The integral of (21) can be computed numerically as discussed in [1]. For a polytope $\mathcal{P}_\ell(\mathcal{R}_1{}^2, \mathcal{R}_2{}^2, \mathcal{R}_3{}^2, \dots, \mathcal{R}_d{}^2)$, the coefficient vector $u = (u_1, u_2, \dots, u_\beta)$, which corresponding to the block $\mathcal{L}_{[j,k]}$, can be found in GNR-enumeration by the following probability [1]:

$$\Pr_{u\backsim Ball_d}\left(\forall j \in [1,d], \sum_{i=1}^{j} u_i{}^2 \le \mathcal{R}_j{}^2\right) = \frac{\mathrm{Vol}\mathcal{P}_\ell(\mathcal{R}_2{}^2, \mathcal{R}_4{}^2, \dots, \mathcal{R}_d{}^2)}{\mathrm{Vol}\mathcal{P}_\ell(1,1,1,\dots,1)}. \quad (22)$$

In practice, assuming such this case for bounding function $\mathcal{R}$ does not corrupt the generality of discussion, and just introduces some partial approximations. For bounding function $\mathcal{R}$ which does not satisfy these constraints (i.e., $\mathcal{R}_1 = \mathcal{R}_2$, …, $\mathcal{R}_{d-1} = \mathcal{R}_d$ where $0 \le \mathcal{R}_1 \le \mathcal{R}_3 \le \cdots \le \mathcal{R}_{d-1}$ and $\beta = 2\ell$), the probability of $\Pr_{u\backsim Ball_d}$ can be approximated as follows [1]:

$$\Pr_{u\backsim Ball_d}\left(\forall j \in [1,d], \sum_{i=1}^{j} u_i{}^2 \le \mathcal{R}_j{}^2\right) \approx \left\lfloor\frac{d}{2}\right\rfloor! \int_{y_1=0}^{\mathcal{R}_2{}^2} \int_{y_2=y_1}^{\mathcal{R}_4{}^2} \dots \int_{y_{\lfloor\frac{d}{2}\rfloor}=y_{\lfloor\frac{d}{2}\rfloor-1}}^{\mathcal{R}_d{}^2} dy_{\lfloor\frac{d}{2}\rfloor} \dots dy_2\, dy_1. \quad (23)$$

Also, to have a better estimation, a partial modification of this approximation is defined as follows:

$$\Pr_{u\backsim Ball_d}\left(\forall j \in [1,d], \sum_{i=1}^{j} u_i{}^2 \le \mathcal{R}_j{}^2\right) \approx \approx \ell! \times \frac{\mathrm{Vol}\mathcal{P}_\ell(\mathcal{R}_2^2, \dots, \mathcal{R}_{2\ell}^2) + \mathrm{Vol}\mathcal{P}_\ell(\mathcal{R}_1^2, \dots, \mathcal{R}_{2\ell-1}^2)}{2} \approx \ell! \times \frac{\sum_{i=0}^{1} \int_{y_1=0}^{\mathcal{R}_2^2 - i} \dots \int_{y_{\ell/2}=y_{\ell/2-1}}^{\mathcal{R}_{2\ell}^2 - i} dy_{\ell/2} \dots dy_1}{2}. \quad (24)$$

D. Complementary Concepts

The definition of static success probability is the same as the original definition of success probability when enumeration radius R is set to λ_1 as follows [12]:

Static success probability of bounding function: For any lattice block of $\mathcal{L}_{[j,k]} = [b_j, b_{j+1}, \dots, b_k]$, initial enumeration radius $R = \lambda_1$ and bounding function $\mathcal{R}$, the static success probability of $p_{succ}(\mathcal{R})$ 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.

The first version of static success probability is formulated exactly similar to (15) as follows [12]:

$$p_{succ}^{new0}\left(\mathcal{L}_{[1,d]}, \mathcal{R}, R\right) = p_{succ}(\mathcal{R}) = \Pr_{u\backsim Ball_d}\left(\forall j \in [1,d], \sum_{i=1}^{j} u_i{}^2 \le \mathcal{R}_j{}^2\right). \quad (25)$$

Note: Following expressions are equivalent in this paper: “Success probability”, “Static success probability”, “Success probability of GNR pruned enumeration”, “Success probability of bounding function”.

In other side, by using Rogers’ theorem, dynamic success frequency can be defined as follows [12]:

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

The dynamic success frequency is formulated as follows [12]:

$$f_{succ}^{new0}\left(\mathcal{L}_{[1,d]}, \mathcal{R}, R\right) = C_{Roger} \times \frac{{r_{\mathrm{FAC}}}^\beta}{2} \times p_{succ}^{new0}\left(\mathcal{L}_{[1,d]}, \mathcal{R}, R\right). \quad (26)$$

Note: As suggested in [12], this paper sets C_{Roger} to 1.

Note: If this is assumed that there is no updating radius in GNR-enumeration, then the dynamic success frequency of bounding function can be assumed as the expected number of solutions visited in enumeration tree, else this dynamic success frequency is more than the expected number of solutions visited in GNR-enumeration.

As discussed in [12], there are different asymptotical/experimental results which verify the convergence of the expected value of the best vectors of lattices with sufficiently big block sizes to $\mathrm{GH}(\mathcal{L}_{[j,k]})$. Based on experimental tests by Chen and Nguyen [1] to compare the final solution norm of enumeration with value of $\mathrm{GH}(\mathcal{L}_{[j,k]})$, depending on the starting index j of a local block for one round of BKZ, following cases are observed:

- For the first lattice blocks in rounds of BKZ, the final solution norm is significantly lower than $\mathrm{GH}(\mathcal{L}_{[j,k]})$. The behaviour of solution norm in running of BKZ is named “head concavity phenomenon” in BKZ, which is discussed in [2].
- For the last lattice blocks in rounds of BKZ (tail of GSO norms), the GSO norms are significantly larger than $\mathrm{GH}(\mathcal{L}_{[j,k]})$. This behaviour of solution norm is named as “tail convexity” in [12].
- For the middle lattice blocks in rounds of BKZ which includes the most of the enumeration calls, the solution norms are mostly bounded as follows [1]:

$$0.95\ \mathrm{GH}(\mathcal{L}_{[j,k]}) \le \|v\| \le 1.05\ \mathrm{GH}(\mathcal{L}_{[j,k]}). \quad (27)$$

This third behaviour of BKZ, can be named as “random manner of middle lattice blocks”.

To the best of our knowledge, this test in [1] is performed with some block sizes of $\beta \le 70$. There are other experimental/asymptotical results on the expected norm of final solution vector which briefly are counted in Section 2.7 from [12].

In fact, the probability distribution of best solution norm for a lattice basis/block is stated in Chen's thesis [16] as following theorem [2]:

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 $\mathrm{Expo}(1/2)$ as $n \to \infty$.

The random variable of $\lambda_1(\mathcal{L})$ for lattices with rank d can be sampled by following relation [2]:

$$\lambda_1(\mathcal{L}) \leftarrow \left(\frac{\mathrm{X}\,\mathrm{Vol}(\mathcal{L})}{V_d(1)}\right)^{1/d}, \text{ where } \mathrm{X} \leftarrow \mathrm{Expo}\left(\frac{1}{2}\right). \tag{28}$$

Note: Theorem 1 can be considered only for full-enumeration or a GNR-enumeration pruned by a bounding function with success probability $\approx \%100$, not for any GNR pruned enumeration.

There is a brief, but sufficient survey of the norm of full/pruned enumerations in Section 2.7 from [12].

At this point, some necessary concepts from [12] which are needed in our analysis are counted as follows:

- *Cutting point*. The enumeration cut point index is defined as the last GSO norm index Cut where $\|b^*_{\mathrm{Cut}}\|^2 \leq R^2\mathcal{R}^2_{d-\mathrm{Cut}+1}$ and $2 \leq \mathrm{Cut} \leq d$.
- *Last non-zero index of $\mathscr{g}$*. The projected vector $b^*_{\mathscr{g}} \in \{b^*_1, \dots, b^*_d\}$ which is eliminated after inserting the enumeration solution v, has the GSO norm of $\|b^*_{\mathscr{g}}\| \leq \|v\|$; The coefficient $w_{\mathscr{g}}$ is always the last non-zero coefficient in vector of w for lattice block of $\mathcal{L}_{[1,d]}$, as follows (see Theorem 2 in [12]):

$$w_{\mathscr{g}} = y_{\mathscr{g}} = 1. \tag{29}$$

- For a GNR-enumeration with radius $R = r_{\mathrm{FAC}} \times \mathrm{GH}(\mathcal{L}_{[1,d]})$ 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 $\mathscr{g}$ for the solution vectors v returned by this enumeration, can be estimated by our non-exact approximate formula of (27) in [12] or can be estimated by our exact formula of (44) in Lemma 8 from [12];
- The norm of solution vector v returned by a pruned enumeration with radius factor of r_{FAC} and success probability $p_{succ}(\mathcal{R}) = \frac{2}{{r_{\mathrm{FAC}}}^{\mathrm{Cut}}}$ over lattice block $\mathcal{L}_\beta$ can be sampled by (30) (see Lemma 2 from [12]):

$$\|v\| = \sqrt[\mathrm{Cut}]{1 + \mathrm{rand}_{[0\dots1]}\,({r_{\mathrm{FAC}}}^{\mathrm{Cut}} - 1)} \times \mathrm{GH}(\mathcal{L}_{\mathrm{Cut}}),$$
$$\text{where } r_{\mathrm{FAC}} = R/\mathrm{GH}(\mathcal{L}_{\mathrm{Cut}}). \tag{30}$$

- If the norm of shortest vector in lattice block $\mathcal{L}_\beta$ is less than enumeration radius R, then the norm of solution vector v which is returned by a GNR pruned enumeration with radius factor of r_{FAC} and static success probability P over lattice block $\mathcal{L}_\beta$, can be sampled by (31):

$$\|v\| = \begin{cases} \mathrm{X}^{1/\mathrm{Cut}}\,\mathrm{GH}(\mathcal{L}_{\mathrm{Cut}}), \text{ where } \mathrm{X} \leftarrow \mathrm{Expo}\left(\frac{1}{2}\right), & \text{if } \mathrm{P} \approx 1 \\ \sqrt[\mathrm{Cut}]{1 + \mathrm{rand}_{[0\dots1]}\left(\frac{2}{\mathrm{P}} - 1\right)} \times \mathrm{GH}(\mathcal{L}_{\mathrm{Cut}}), & \text{if } \frac{2}{{r_{\mathrm{FAC}}}^{\mathrm{Cut}}} \leq \mathrm{P} < 1 \\ \sqrt[\mathrm{Cut}]{1 + \mathrm{rand}_{[0\dots1]}({r_{\mathrm{FAC}}}^{\mathrm{Cut}} - 1)} \times \mathrm{GH}(\mathcal{L}_{\mathrm{Cut}}), & \text{if } \mathrm{P} < \frac{2}{{r_{\mathrm{FAC}}}^{\mathrm{Cut}}} \;\&\; \mathrm{rand}_{\left[0\dots\frac{2}{{r_{\mathrm{FAC}}}^{\mathrm{Cut}}}\right]} \leq \mathrm{P} \\ \mathrm{Un\_Successfull}, & \text{if } \mathrm{P} < \frac{2}{{r_{\mathrm{FAC}}}^{\mathrm{Cut}}} \;\&\; \mathrm{rand}_{\left[0\dots\frac{2}{{r_{\mathrm{FAC}}}^{\mathrm{Cut}}}\right]} > \mathrm{P} \end{cases}$$
$$\text{where } r_{\mathrm{FAC}} = R/\mathrm{GH}(\mathcal{L}_{\mathrm{Cut}}). \tag{31}$$

Remark 1. For an input lattice block $\mathcal{L}_{[1,d]}$ and enumeration radius R, by using the concept of cutting point "Cut", the formula of (36) in Lemma 2 from [12] and the formula of (37) in Lemma 3 from [12], are revised to formula of (30) and (31) by setting $\mathcal{L}_{\mathrm{Cut}}$ with dimension of Cut and GSO basis of $B^*_{[1,\mathrm{Cut}]} = [\|b^*_1\|, \dots, \|b^*_{\mathrm{Cut}}\|]$ instead of $\mathcal{L}_\beta$ with dimension of β and GSO basis of $B^*_{[1,\beta]} = [\|b^*_1\|, \dots, \|b^*_\beta\|]$.

Our Contributions

The estimations of GNR-enumeration cost (by relation (19)) and the success probability of GNR-bounding function (by relation (15)) are defined in [1] under Heuristic 2. Unfortunately, paper [1] only considers one type of pruning in these estimations which is defined by condition of (20). In fact, the condition of (20) is used to determine the possibility and probability of laying a partial solution candidate in the corresponding cylinder-intersection by bounding function of $\mathcal{R}$. Here, three more types of pruning are introduced which are ignored in former estimations of success probability and enumeration cost. These pruning types include following cases:

- ***Pruning by concept of full-enumeration success probability***. This type of pruning is discussed and analysed in third section (Part A); Also, we propose the concept of optimal enumeration radius to eliminate this type of pruning while the cost of enumeration is held minimized;
- ***Pruning by ignoring the enumeration tree levels of "$l = 1$ to $d - \mathrm{Cut}$"***. This type of pruning is observed if $\mathrm{Cut} < d$; We discuss massively on this concept in [12]; In third section (Part D), we propose a simple technique to eliminate this type of pruning by introducing a mapping technique which can be included in generating GNR bounding function to force $\mathrm{Cut} = d$;
- ***Pruning by finding a final solution in GNR enumeration tree levels of $l = d - \mathrm{Cut} + 2$ to d.*** In fact, this item is not a real pruning, but since it prevents from opening the child nodes of an enumeration tree node which includes some final solutions, this is considered as pruning; Moreover, it is impossible to eliminate this type (of pruning) at all, since this is an intrinsic phase in enumeration

function, unless we force the enumeration function to abort the function after finding the first final solution vector, such as the pseudo-code of Algorithm 2 in [13]; In fact, if dynamic success frequency would be small (e.g., $f_0 \approx O(1)$), then aborting enumeration function after first finding of final solution is reasonable, but for big value of dynamic success frequency f_0, this is expected that enumeration function updates radius after each success in finding solution and then continues to traverse the remain of the enumeration tree (similar to the pseudo-code of Algorithm 9 in [1]).

By introducing these three types of pruning plus the cylinder-intersection pruning by (20), the estimations of success probability and enumeration cost are revised respectively in third section (Part B) and third section (Part C).

A. Definition of Optimal Enumeration Radius

By using the definition of Hermite's constant in second section (Part A), in worst case of the full-enumeration, the optimal enumeration radius can be assumed as $R = \sqrt{\gamma_n}\mathrm{vol}(\mathcal{L})^{1/n}$ [13], while in this section, first definition of optimal enumeration radius is introduced in average-case. The enumeration radius R in [1] is defined as follows (by some partial modification):

$$R = \begin{cases} \min\left(\sqrt{\Upsilon}\,\mathrm{GH}(\mathcal{L}_{[j,k]}), \|b_j^*\|\right), & if\ k-j+1 \geq 30 \\ \|b_j^*\|, & \text{otherwise} \end{cases}, \quad (32)$$

where $\sqrt{\Upsilon}$ is the initial radius parameter. For block sizes of $\beta = k - j + 1 \geq 30$, value of r_{FAC} is defined as follows (by using relation (16) and (32)):

$$r_{\mathrm{FAC}} = \frac{\min\left(\sqrt{\Upsilon}\,\mathrm{GH}\left(\mathcal{L}_{[j,k]}\right), \|b_j^*\|\right)}{\mathrm{GH}\left(\mathcal{L}_{[j,k]}\right)}. \quad (33)$$

The main problem in choosing enumeration radius is to find the smallest radius which is not smaller than the shortest vector in the input lattice block. For this end, Chen and Nguyen claim that, the radius parameter of Υ in practice can be selected as $\sqrt{\Upsilon} = \sqrt{1.1} \approx 1.05$ (see [1]), but to the best of our knowledge, this value is estimated only by some experimental tests over BKZ with block size $\beta < 70$ (see Fig. 3 in [1]). By using Theorem 1, the optimal enumeration radius can be defined by the concept of full-enumeration success probability. A full-enumeration with initial radius R intrinsically prunes enumeration by using enumeration radius (i.e., the use of an enumeration radius is concretely a type of pruning). Following lemma formally defines the success probability of full-enumeration:

Lemma 1. For given lattice block $\mathcal{L}_\beta$ with block size of β, the success probability of a full-enumeration with initial radius R can be defined by (34):

$$p_{succ}\left(1^\beta, R, \mathcal{L}_\beta\right) = 1 - e^{-\frac{r_{\mathrm{FAC}}{}^\beta}{2}}. \quad (34)$$

Proof. By using (9), (15), (16) and (28), this success probability of $p_{succ}\left(1^\beta, R, \mathcal{L}_\beta\right)$ for lattice block $\mathcal{L}_\beta$ is estimated as follows (for $\mathrm{X} \leftarrow \mathrm{Expo}(1/2)$):

$p_{succ}\left(1^\beta, R, \mathcal{L}_\beta\right) = \mathrm{prob}\left(\lambda_1\left(\mathcal{L}_\beta\right) < R\right) = \mathrm{prob}\left(\mathrm{X} < r_{\mathrm{FAC}}{}^\beta\right) = 1 - e^{-\frac{r_{\mathrm{FAC}}{}^\beta}{2}}$.

Since the success probability of full-enumeration is not noted in former studies, these studies (former studies) always assumed implicitly to use $p_{succ}\left(1^\beta, R, \mathcal{L}_\beta\right) = 1$. For a typical lattice block $\mathcal{L}_\beta$, the ideal enumeration radius would be $R = \lambda_1\left(\mathcal{L}_\beta\right)$ which defines the radius factor of r_{FAC} by using the tight bound (upper-bound and lower-bound) of $r_{\mathrm{FAC}} = \frac{\lambda_1(\mathcal{L}_\beta)}{\mathrm{GH}(\mathcal{L}_\beta)}$. As mentioned, former estimation of enumeration radius in (27) uses experimental tests to estimate the bound of r_{FAC} in average-case (see Fig. 3 in [1]). Theorem 2 introduces an exact definition of this bound.

Theorem 2. For given number X from random lattice blocks, the effective upper-bound/lower-bound of $r_{\mathrm{FAC}} = \frac{\lambda_1(\mathcal{L}_\beta)}{\mathrm{GH}(\mathcal{L}_\beta)}$ can be estimated in average-case as follows:

$$r_{\mathrm{FAC}_{\min}} \leq r_{\mathrm{FAC}} \leq r_{\mathrm{FAC}_{\mathrm{opt}}}, \quad (35)$$

where

$$r_{\mathrm{FAC}_{\mathrm{opt}}} = \sqrt[\beta]{-2\ln(1-\mathrm{p}_{\mathrm{opt}})}\ \textit{and}$$

$$r_{\mathrm{FAC}_{\min}} = \sqrt[\beta]{-2\ln\left(1-\mathrm{p}_{\min}\right)}\ \textit{and}$$

$\mathrm{p}_{\min} = 1/\mathrm{X}$ *and* $\mathrm{p}_{\mathrm{opt}} = 1 - \varepsilon$.

Proof. The lower-bound and upper-bound for $r_{\mathrm{FAC}} = \frac{\lambda_1(\mathcal{L}_\beta)}{\mathrm{GH}(\mathcal{L}_\beta)}$ are formally defined based on relation (34), as follows:

Minimum hopeful radius parameter ($r_{\mathrm{FAC}_{\min}}$). For given number X of random lattice blocks, the minimum radius parameter leads to success probability of $\mathrm{p}_{\min} = p_{succ}\left(1^\beta, R, \mathcal{L}_\beta\right) = \frac{1}{\mathrm{X}}$ for full-enumeration over these number of X blocks where $R = r_{\mathrm{FAC}_{\min}} \times \mathrm{GH}\left(\mathcal{L}_\beta\right)$ (i.e., only one of the full-enumerations over these X blocks probably returns the best solution).

Optimal radius parameter ($r_{\mathrm{FAC}_{\mathrm{opt}}}$). For given number of X random lattice blocks, the minimum radius parameter leads to success probability of $\mathrm{p}_{\mathrm{opt}} = p_{succ}\left(1^\beta, R, \mathcal{L}_\beta\right) = 1 - \varepsilon$ for full enumeration over these X blocks where $R = r_{\mathrm{FAC}_{\mathrm{opt}}} \times \mathrm{GH}\left(\mathcal{L}_\beta\right)$ (i.e., all of full-enumerations over these X blocks return the best solution).

$$r_{\mathrm{FAC}_{\min}} \leq r_{\mathrm{FAC}} = \frac{\lambda_1(\mathcal{L}_\beta)}{\mathrm{GH}(\mathcal{L}_\beta)} \leq r_{\mathrm{FAC}_{\mathrm{opt}}}.$$

By expanding the definitions of $r_{\mathrm{FAC}_{\mathrm{opt}}}$ and $r_{\mathrm{FAC}_{\min}}$ by relation (34) in Lemma 1:

$\sqrt[\beta]{-2\ln(1-\mathrm{p_{min}})} \le r_{\mathrm{FAC}} = \frac{\lambda_1(\mathcal{L}_\beta)}{\mathrm{GH}(\mathcal{L}_\beta)} \le \sqrt[\beta]{-2\ln(1-\mathrm{p_{opt}})}$.

Note: The optimal radius parameter $r_{\mathrm{FAC_{opt}}}$ corresponds with optimal enumeration radius as $R_{\mathrm{opt}} = r_{\mathrm{FAC_{opt}}} \times \mathrm{GH}(\mathcal{L}_\beta)$.

Remark 2. The random manner of lattice blocks $\mathcal{L}_{[j,k]}$ in BKZ algorithm is observed only for $\mathrm{Hdown} \le j \le \mathrm{Tup}$ where "Hdown" represents the maximum index in head concavity and "Tup" represents the minimum index in tail convexity; So for each round of BKZ algorithm (or BKZ-simulation), the number of X random lattice blocks can be assumed as $\mathrm{X} = \mathrm{Tup} - \mathrm{Hdown} + 1$;

Our estimation results by formula of (35) for block sizes of $50 \le \beta \le 240$ are shown in fourth section (Part A). In actual running of BKZ, simulation of BKZ, also our reasoning and proofs, the value of r_{FAC} is assumed as a variable between 1 to $\sqrt{\mathrm{Y}}$, therefore the success probability of full-enumeration would be mostly $p_{succ}(1^\beta, R, \mathcal{L}_\beta) \ge 39\%$ (see our estimations by formula of (35) for block sizes of $50 \le \beta \le 240$ in Table 1 and Table 2 from fourth section (Part A)). Also, to have better sense about ignoring full-enumeration success probability in former studies, note to following example:

> *Chen and Nguyen use the enumeration radius of $R = GH(\mathcal{L}_{[j,k]})$ in estimation of upper-bound for extreme pruned enumeration cost in Table 5 at [1], while by using our reasoning in this section, all these extreme numerations fail to find best solution with probability $\approx 61\%$, which can be penalized by increasing these estimated costs with factor of at most $\frac{100}{39} \approx 2^{1.36}$. At result, when the value of r_{FAC} is variable between 1 to $\sqrt{Y}$, the effect of full-enumeration success probability can be ignored in asymptotical analysis of cost estimation.*

B. Revised Estimation of Enumeration Success Prob.

This is worthy of mentioning that the GSO partial solution candidates in level l from GNR pruned-enumeration tree are only limited to those enumeration tree nodes which satisfy "*bounding condition*" at level l, which is defined in (20) and the probability of this condition is referred in this paper as $\mathrm{Pr}_{u\backsim Ball_l}$ (also see this condition in line 10 from Algorithm 2 in [13] or line 16 from Algorithm 9 in [1]). Moreover, the final solutions are GSO partial candidates in level $l = d$, and the probability of this condition is referred generally as success probability p_{succ}. In fact, this section tries to revise the probability of this condition as $\mathrm{Pr}_{u\backsim Ball_l}$ (or p_{succ}) to be more exact. By assuming Heuristic 2, this section introduces an exact estimation of success probability in following lemma:

Lemma 2. Under Gaussian Heuristic and Heuristic 2, for an input lattice block of $\mathcal{L}_{[1,d]} = [b_1, b_2, \dots, b_d]$ with shortest vector of v with norm of $\|v\| = \lambda_1(\mathcal{L}_{[1,d]})$, the success probability of finding this solution vector v by GNR-enumeration with enumeration radius $R \ge \|v\|$ can be estimated by (36):

$$p_{succ}^{new1}(\mathcal{L}_{[1,d]}, \mathcal{R}, R) = \sum_{\jmath=2}^{\mathrm{Cut}} \left[\mathrm{Prob}(\mathscr{g} = \jmath) \times p_{succ}(1^\jmath, R, \mathcal{L}_\jmath) \times \mathrm{Pr}_{u\backsim Ball_{\jmath-1}} \left(\forall t \in [d-\jmath+2, d], \sum_{i=d-\jmath+2}^{t} u_i^2 \le \frac{R^2\mathcal{R}_t^2 - \|b_\jmath^*\|^2}{R^2 - \|b_\jmath^*\|^2} \right) \right]. \quad (36)$$

Proof. Under assumption of Heuristic 2, the success probability can be estimated by the idea proposed in relation of (15). Also, since $w_\mathscr{g} = 1$ (by using Theorem 2 in [12]), for $\mathscr{g} = 1$, this is only needed to determine whether the first vector of block $\mathcal{L}_{[1,d]}$ as b_1^* has the norm of $w_\mathscr{g}\|b_\mathscr{g}^*\| = \|b_1^*\| \le R$ or not? The probability of this case as $v = b_1^*$, with respect to all other linear combinations of v by using vectors of $\{b_1^*, b_2^*, \dots, b_{\mathrm{Cut}}^*\}$ is zero, so $\mathscr{g} = 1$ is ignored in (36). By using our definition of "Cutting Point", the probability of visiting GSO partial solution candidates in level l from GNR pruned-enumeration tree for given bounding function $\mathcal{R}$ and lattice block $\mathcal{L}_{[1,d]}$ with cut point of Cut, can be estimated as follows:

$$\mathrm{Pr}_{u\backsim Ball_l}^{new1}(\mathcal{L}_{[1,d]}, \mathcal{R}, R, \mathscr{g} = \mathrm{Cut}) \approx \mathrm{Pr}_{u\backsim Ball_l}\left(\forall t \in [d-\mathrm{Cut}+2, l], \frac{w_{\mathrm{Cut}}^2 \|b_{\mathrm{Cut}}^*\|^2}{R^2} + \sum_{i=d-\mathrm{Cut}+2}^{t} u_i^2 \le \mathcal{R}_t^2 \right), \quad (37)$$

where $d - \mathrm{Cut} + 2 \le l \le d$.

The relation (37) assumes that last non-zero index for all partial (and final) solutions is $\mathscr{g} = \mathrm{Cut}$. Let's try to estimate the probability of finding the partial solution vectors which are limited to the ones with any possible last non-zero index of $\mathscr{g} = \jmath \le \mathrm{Cut}$. For this end, $\mathrm{Pr}_{u\backsim Ball_l}^{new1}$ in (37) can be modified into (38):

$$\mathrm{Pr}_{u\backsim Ball_l}^{new1}(\mathcal{L}_{[1,d]}, \mathcal{R}, R, \mathscr{g} = \jmath \le \mathrm{Cut}) \approx \mathrm{Pr}_{u\backsim Ball_l}\left(\forall t \in [d-\mathscr{g}+2, l], \frac{w_\mathscr{g}^2 \|b_\mathscr{g}^*\|^2}{R^2} + \sum_{i=d-\mathscr{g}+2}^{t} u_i^2 \le \mathcal{R}_t^2 \right), \quad \text{where } d - \mathscr{g} + 2 \le l \le d. \quad (38)$$

Remark 3. By our definition of cutting point of Cut and last non-zero index of $\mathscr{g} \le \mathrm{Cut}$ (see Section 3.2.1 and Section 3.2.2 from [12]), this is clear that the condition of "$\frac{w_\mathscr{g}^2 \|b_\mathscr{g}^*\|^2}{R^2} = \frac{\|b_\mathscr{g}^*\|^2}{R^2} \le \mathcal{R}_{d-\mathscr{g}+1}^2$" is always expected to be "True", therefore the probability of visiting GSO partial solution candidates in level $l = d - \mathscr{g} + 1$ can be defined as follows:

$$\mathrm{Pr}_{u\backsim Ball_{l=d-\mathscr{g}+1}}^{new1}(\mathcal{L}_{[1,d]}, \mathcal{R}, R, \mathscr{g}) = \mathrm{Pr}_{u\backsim Ball_{l=d-\mathscr{g}+1}}\left(\frac{w_\mathscr{g}^2 \|b_\mathscr{g}^*\|^2}{R^2} \le \mathcal{R}_{d-\mathscr{g}+1}^2 \right) = 1. \quad (39)$$

By using $w_{\mathcal{g}} = 1$ (which is in Theorem 2 from [12]):

$$\Pr^{new1}_{u\backsim Ball_l}\left(\mathcal{L}_{[1,d]}, \mathcal{R}, R, l, \mathcal{g} = \mathcal{j}\right) \approx$$

$$\Pr_{u\backsim Ball_l}\left(\forall t \in [d-\mathcal{g}+2, l], \sum_{i=d-\mathcal{g}+2}^{t} u_i^2 \le \mathcal{R}_t^2 - \frac{\|b_{\mathcal{g}}^*\|^2}{R^2}\right),$$

$$where\ d-\mathcal{g}+2 \le l \le d. \tag{40}$$

At this point, we use the definition of last non-zero index of $\mathcal{g}$ (in Section 3.2.1 from [12]) in sampling of random vector u from $Ball_l$ with radius of unit-length. By only focusing on the GSO partial solution candidates in level l with any possible last non-zero index of $\mathcal{g} = \mathcal{j}$, GNR-enumeration opens the child nodes of these partial solutions (unless, at last level $l = d$ which returns these final solutions, and comes back to previous level of enumeration tree to find the other solutions). The direction of visiting nodes in a GNR-enumeration tree is from the last index of GSO block to the first one. Accordingly, by using Lemma A.1 in [1] and considering this fact that the effective radius of surrounding unit ball of dimension $\mathcal{D} = d$ is reduced into a ball of dimension $\mathcal{D} = l - d + \mathcal{g} - 1$ with radius of $1 - \frac{\|b_{\mathcal{g}}^*\|^2}{R^2}$, the estimation of $\Pr^{new1}_{u\backsim Ball_l}$ in (40) can be revised into $\Pr^{new2}_{u\backsim Ball_l}$ as follows:

$$\Pr^{new2}_{u\backsim Ball_l}\left(\mathcal{L}_{[1,d]}, \mathcal{R}, R, \mathcal{g} = \mathcal{j}\right) \approx$$

$$\frac{Vol\mathcal{P}_\ell\left(\mathcal{R}_{d-\mathcal{g}+2}^2 - \frac{\|b_{\mathcal{g}}^*\|^2}{R^2}, \ldots, \mathcal{R}_l^2 - \frac{\|b_{\mathcal{g}}^*\|^2}{R^2}\right)}{Vol\mathcal{P}_\ell\left(1 - \frac{\|b_{\mathcal{g}}^*\|^2}{R^2}, \ldots, 1 - \frac{\|b_{\mathcal{g}}^*\|^2}{R^2}\right)} \approx$$

$$\frac{Vol\mathcal{P}_\ell\left(\frac{R^2\mathcal{R}_{d-\mathcal{g}+2}{}^2 - \|b_{\mathcal{g}}^*\|^2}{R^2 - \|b_{\mathcal{g}}^*\|^2}, \ldots, \frac{R^2\mathcal{R}_l{}^2 - \|b_{\mathcal{g}}^*\|^2}{R^2 - \|b_{\mathcal{g}}^*\|^2}\right)}{Vol\mathcal{P}_\ell(1,1,\ldots,1)} \approx$$

$$\Pr_{u\backsim Ball_{\mathcal{D}}}\left(\forall t \in [d-\mathcal{g}+2, l], \sum_{i=d-\mathcal{g}+2}^{t} u_i^2 \le \frac{R^2\mathcal{R}_t^2 - \|b_{\mathcal{g}}^*\|^2}{R^2 - \|b_{\mathcal{g}}^*\|^2}\right) \approx \tag{41}$$

$$\left\lceil \frac{\mathcal{D}}{2} \right\rceil ! \times \mathrm{Vol}\mathcal{P}_\ell\left(\mathrm{T}_1, \ldots, \mathrm{T}_{\lceil \mathcal{D}/2 \rceil}\right) \approx$$

$$\left\lceil \frac{\mathcal{D}}{2} \right\rceil ! \times \int_{y_1=0}^{\mathrm{T}_1} \ldots \int_{y_{\lceil \mathcal{D}/2 \rceil}=y_{\lceil \mathcal{D}/2 \rceil-1}}^{\mathrm{T}_{\lceil \mathcal{D}/2 \rceil}} dy_{\lceil \mathcal{D}/2 \rceil} \ldots dy_1, \tag{42}$$

where $\mathrm{T}_i = \frac{R^2\mathcal{R}_{2\lfloor (d-\mathcal{g}+2)/2+i \rfloor}^2 - \|b_{\mathcal{g}}^*\|^2}{R^2 - \|b_{\mathcal{g}}^*\|^2}$ *and*

$$1 \le \mathcal{D} = l - d + \mathcal{g} - 1 \le \mathcal{g} - 1 \text{ and}$$
$$d - \mathcal{g} + 2 \le l \le d.$$

Note: All the notations with formats of $\Pr_{u\backsim Ball_{\dots}}$, $\Pr^{new\dots}_{u\backsim Ball_l}$ and p_{succ} in this paper show the probability value and obviously are upper-bounded by 1.

The pseudo-code of estimator for $\Pr^{new2}_{u\backsim Ball_l}$ in relations of (41) and (42) as "*Our estimator of success probability*" is proposed in Algorithm 1:

Algorithm 1: Estimation of probability of $\Pr^{new2}_{u\backsim Ball_l}$ in relation (41)

Input: Bounding func. $\mathcal{R}$, enum radius R, GSO norms $[\|b_1^*\|, \ldots$ level l, total block size d, last non zero index of $\mathcal{g}$.

1: for$(t = d - \mathcal{g} + 2, \ldots, l)$ $\mathcal{R}''^2_{t-d+\mathcal{g}-1} \leftarrow \min\left(\frac{R^2\mathcal{R}_t^2 - \|b_{\mathcal{g}}^*\|^2}{R^2 - \|b_{\mathcal{g}}^*\|^2}, 1\right)$;/* *see (41)* */

2: $\mathcal{D} = l - d + \mathcal{g} - 1$;

3: for$(k = 0,1)\{$//*begin* for1

4: $\mathrm{C} \leftarrow 1$;//$C \in \mathbb{R}[X]$ *is a polynomial*

5: for$(j = \mathcal{D}, \mathcal{D}-2, \ldots, 2)\{$//*begin* for2
$\mathrm{C} \leftarrow \int_{t=0}^{x} \mathrm{C}(t)\, dt$; $\mathrm{C} \leftarrow \mathrm{C}(\mathcal{R}''^2_{j-k}) - \mathrm{C}(x)$;$\}$// *end* for2

6: $p_k \leftarrow \mathrm{C}(0) \times \left\lceil \frac{\mathcal{D}}{2} \right\rceil !$;/* *see (42)* */$\}$//*end* for1

Output: $(p_1 + p_2)/2$ as the success probability

By applying the probability of last non-zero index as $\mathrm{Prob}(\mathcal{g} = \mathcal{j})$ by using Lemma 8 in [12], and our proposed concept of full-enumeration success probability (see (34) in Lemma 1 at third section (Part A)), our revised estimation of the probability of finding the GSO partial solution candidates in level l, with any possible last non-zero index of $\mathcal{g} = \mathcal{j}$, can be defined as follows:

$$\Pr^{new3}_{u\backsim Ball_l}\left(\mathcal{L}_{[1,d]}, \mathcal{R}, R, \mathcal{g} = \mathcal{j}\right) \approx \mathrm{Prob}(\mathcal{g} = \mathcal{j}) \times p_{succ}\left(1^{\mathcal{j}}, R, \mathcal{L}_{\mathcal{j}}\right) \times \Pr^{new2}_{u\backsim Ball_l}\left(\mathcal{L}_{[1,d]}, \mathcal{R}, R, \mathcal{g} = \mathcal{j}\right) \approx \tag{43}$$

$$\mathrm{Prob}(\mathcal{g} = \mathcal{j}) \times p_{succ}\left(1^{\mathcal{j}}, R, \mathcal{L}_{\mathcal{j}}\right) \times \Pr_{u\backsim Ball_{\mathcal{D}}}\left(\forall t \in [d-\mathcal{j}+2, l], \sum_{i=d-\mathcal{j}+2}^{t} u_i^2 \le \frac{R^2\mathcal{R}_t^2 - \|b_{\mathcal{j}}^*\|^2}{R^2 - \|b_{\mathcal{j}}^*\|^2}\right), \tag{44}$$

where $1 \le \mathcal{D} = l - d + \mathcal{g} - 1 \le \mathcal{g} - 1$ *and*

$$d - \mathcal{g} + 2 \le l \le d.$$

Now, the expected value of the probability of finding the GSO partial solution candidates in level l (by considering whole indices of $2 \le \mathcal{g} \le \mathrm{Cut}$) can be estimated as follows:

$$\mathrm{E}\left[\Pr^{new3}_{u\backsim Ball_l}\left(\mathcal{L}_{[1,d]}, \mathcal{R}, R, 2 \le \mathcal{g} \le \mathrm{Cut}\right)\right] \approx$$

$$\begin{cases} \sum_{\mathcal{j}=d-l+2}^{\mathrm{Cut}} \Pr^{new3}_{u\backsim Ball_l}\left(\mathcal{L}_{[1,d]}, \mathcal{R}, R, \mathcal{g} = \mathcal{j}\right), \text{for } d - \mathrm{Cut} + 2 \le l \le d \\ \Pr^{new3}_{u\backsim Ball_{l_0=d-\mathrm{Cut}+2}}\left(\mathcal{L}_{[1,d]}, \mathcal{R}, R, \mathcal{g} = \mathrm{Cut}\right), \text{for } l = d - \mathrm{Cut} + 1. \end{cases} \tag{45}$$

Note: For level of $l = d - \mathrm{Cut} + 1$ in (45), the probability in this level is equal to the probability of $\Pr^{new3}_{u\backsim Ball_{l=d-\mathrm{Cut}+2}}$ at level of $l = d - \mathrm{Cut} + 2$;

Note: Since $\mathcal{D} = l - d + \mathcal{j} - 1 \ge 1$ in (44), therefore the index of $\mathcal{j}$ in (45) starts from $\mathcal{j} = d - l + 2$, instead of index of $\mathcal{j} = 2$;

Finally by using (45), our revised estimation of success probability of bounding function $\mathcal{R}$, as the expected value of the probability of finding final solutions in level $l = d$

(by considering whole indices of $2 \le \mathcal{g} \le \text{Cut}$) can be estimated as follows:

$p_{succ}^{new1}(\mathcal{L}_{[1,d]}, \mathcal{R}, R) \approx \text{E}\left[\Pr_{u \backsim Ball_{l=d}}^{new3}(\mathcal{L}_{[1,d]}, \mathcal{R}, R, 2 \le \mathcal{g} \le \text{Cut})\right]$.

Finally this lemma (Lemma 2) is proved.

Since our estimation of success probability by relation (36) in Lemma 2, is not easy to work and analyze, Remark 4 introduces a suitable formula which approximates the success probability.

Remark 4. Our estimation of the success probability in Lemma 2 can be approximated by (46):

$$p_{succ}^{\text{Approx1}}(\mathcal{L}_{[1,d]}, \mathcal{R}, R) = \left(\sum_{j=1}^{\text{Cut}} \text{Prob}(\mathcal{g} = j)\right) \times p_{succ}(1^{\text{Cut}}, R, \mathcal{L}_{\text{Cut}}) \times \Pr_{u \backsim Ball_{l=d}}^{new2}(\mathcal{L}_{[1,d]}, \mathcal{R}, R, \mathcal{g} = \text{Cut}). \quad (46)$$

As mentioned in second section (Part D), dynamic success frequency shows the expected number of solutions in enumeration tree (by assumption of no updating radius). By using formula (21) in paper [12], the success probability of p_{succ}^{new1} in (36) can be changed into dynamic success frequency of f_{succ}^{new1} as follows:

$$f_{succ}^{new1}(\mathcal{L}_{[1,d]}, \mathcal{R}, R) \approx C_{Roger} \frac{{r_{\text{FAC}}}^{\text{Cut}}}{2} p_{succ}^{new1}(\mathcal{L}_{[1,d]}, \mathcal{R}, R), \quad \textit{where } r_{\text{FAC}} = \frac{R}{\text{GH}(\mathcal{L}_{[1,\text{Cut}]})}. \quad (47)$$

Note: As suggested in [12], this paper sets C_{Roger} to 1.

Accordingly, by using (47), the sampling method for computing the number of solutions (showing by the notation of K) in a typical GNR-enumeration with dynamic success frequency of $f_{succ}^{new1} = f_0$ can be defined as follows:

$$K = \begin{cases} \lceil f_0 \rceil, & \text{if } \text{rand}_{[0\ldots1]} \le f_0 - \lfloor f_0 \rfloor \\ \lfloor f_0 \rfloor, & \text{otherwise.} \end{cases} \quad (48)$$

C. Revised Estimation of Enumeration Cost

This section proposes following algorithm to estimate the total nodes of GNR-enumeration tree. This algorithm includes the concepts of all four pruning types, along with the process of updating radius. Lemma 3 formally introduces our revised estimation of GNR-enumeration cost by using Algorithm 2. Besides the better estimation of enumeration cost, Algorithm 2 can be used as a sampling method of solution norm too, similar to Lemma 3 in [12].

Note: The array of "Solution" defined in line 9 of Algorithm 2, includes the number of K entries, in the way that each of these entries has two fields: "index" (as the index of that leaf node in enumeration tree) and "norm" (as the norm of final solution in that leaf node). The notation of "Solution$[i]$#index" and "Solution$[i]$#norm" in Algorithm 2 respectively represent the index and norm of final solution in leaf node i.

Algorithm 2: Enumeration cost with updating radius (enum_cost_UpdateR)

Input: GSO norms $B_{[1,d]}^* = [\|b_1^*\|, \ldots, \|b_d^*\|]$ of $\mathcal{L}_{[1,d]}$, Bounding function $\mathcal{R}$, enum radius R, parameter "abort".

1: $\text{Cut} = \text{GET}_{\text{CUT}}(B_{[1,d]}^*, \mathcal{R}, R)$;
2: $\text{gh} = \text{GH}(\mathcal{L}_{[1,\text{Cut}]})$; $r_{\text{FAC}} = \frac{R}{\text{GH}(\mathcal{L}_{[1,\text{Cut}]})}$;
3: $f_0 = f_{succ}^{new1}(\mathcal{L}_{[1,d]}, \mathcal{R}, R)$;//by formula (47)
4: $K = \begin{cases} \lceil f_0 \rceil, & \text{if } \text{rand}_{[0\ldots1]} \le f_0 - \lfloor f_0 \rfloor \\ \lfloor f_0 \rfloor, & \text{otherwise} \end{cases}$; // by formula (48)
5: $N_{\text{new1}} = 0$;
6: for$(l = d - \text{Cut} + 1, \ldots, d)$\{/*begin for1*/
7: $H_l^{\text{new}} = \text{E}\left[\Pr_{u \backsim Ball_l}^{new3}(\mathcal{L}_{[1,d]}, \mathcal{R}, R, 2 \le \mathcal{g} \le \text{Cut})\right] \times H_l$;
/*by using (45) where H_l is defined in (11).*/
8: $N_{\text{new1}} \mathrel{+}= H_l^{\text{new}}$; \}/*end for1*/
9: Solution := array$[1 \ldots K]$ of Struct \{index, norm\};
10: for$(t = 1, \ldots, K)$\{//begin for2
11: loop$\{j \leftarrow \text{randINT}_{[1\ldots N_{\text{new1}}]};\}$
12: until$(\forall\, 1 \le i < t$: Solution$[i]$#index $\ne j)$; //uniform random selection without substitution
13: Solution$[t]$#index $\leftarrow j$;
14: Solution$[t]$#norm $\leftarrow$ $\sqrt[\text{Cut}]{1 + \text{rand}_{[0\ldots1]}({r_{\text{FAC}}}^{\text{Cut}} - 1)} \times \text{gh}$; /*see (30) by Remark 1*/\}//end for2
15: Sort(array = "Solution", key = "index"); /* Sorting of array of "Solution" based on "key = index" in an increase order */
16: $R_{\text{new}} \leftarrow R$; $\text{last}_{\text{idx}} = 0$; $N_{\text{new2}} \leftarrow 0$;
17: for$(t = 1, \ldots, K)$\{//begin for3
18: if(Solution$[t]$#norm $< \text{R}_{\text{new}}$)\{//begin if2
19: for$(l = d - \text{Cut} + 1, \ldots, d)$\{//begin for4
20: $N_{\text{new2}} \mathrel{+}= \frac{\text{Solution}[t]\text{\#index} - \text{last}_{\text{idx}}}{N_{\text{new1}}} H_l^{\text{new}} \left(\frac{R_{\text{new}}}{R}\right)^l$; \}//end for4
21: $\text{R}_{\text{new}} \leftarrow$ Solution$[t]$#norm; $\text{Last}_{\text{idx}} = t$;
22: if(abort = true) return $[N_{\text{new2}}, R_{\text{new}}]$;
23: \}\}//end if2//end for3
24: for$(l = d - \text{Cut} + 1, \ldots, d)$\{//begin for5
25: $N_{\text{new2}} \mathrel{+}= \frac{N_{\text{new1}} - \text{last}_{\text{idx}}}{N_{\text{new1}}} H_l^{\text{new}} \left(\frac{R_{\text{new}}}{R}\right)^l$; /*for last update of radius up to end*/\}//end for5

Output: $[N_{\text{new2}}, R_{\text{new}}]$./* N_{new2} is returned as the total enumeration cost and R_{new} is returned as the sampled norm of enumeration solution*/

Note: For better speedup in running-time of Algorithm 2, "Insertion Sort" can be used instead of line 12, so that

the repeated indices can be checked and eliminated easily by "Insertion Sort".

Lemma 3. For an input lattice block $\mathcal{L}_{[1,d]}$, bounding function $\mathcal{R}$ and enumeration radius R, under Gaussian Heuristic and Heuristic 2, by assuming that each node at the same level of GNR-enumeration tree includes same number of child nodes, Algorithm 2 samples the norm of final solution, also it estimates the total nodes of GNR enumeration after being pruned by four proposed types of pruning and updating the enumeration radius after each success of finding solution.

Proof. To prove this lemma, this is needed to show that the concept of four types of pruning (which are proposed at the beginning of third section) and also updating radius are considered collectively by Algorithm 2, in estimation of the total nodes of GNR-enumeration (also sampling the norm of final solution of GNR-enumeration);

The function of $\mathrm{GET}_{\mathrm{CUT}}$ in line 1 from Algorithm 2 returns the cut point of bounding function $\mathcal{R}$ with enumeration radius R, for an input lattice block (i.e., the same operations as lines 11 to 16 from Algorithm 3 in [12]). The line 2 form Algorithm 2 works based on Remark 1. Dynamic success frequency and number of final solutions in GNR-enumeration tree with no updating radius, are respectively defined in lines of 3 and 4 in Algorithm 2.

Lines of 5 to 8 estimate the total nodes of GNR-enumeration tree after four types of pruning (which are proposed at the beginning of third section) as follows:

$$N_{new1}\left(\mathcal{L}_{[1,d]},\mathcal{R},R\right)=\sum_{l=d-\mathrm{Cut}+1}^{d}\mathrm{E}\left[\mathrm{Pr}_{u\sim Ball_l}^{new3}\left(\mathcal{L}_{[1,d]},\mathcal{R},R,2\le \mathcal{G}\le \mathrm{Cut}\right)\right]\times H_l, \tag{49}$$

where H_l *is defined in (11).*

To complete this proof, updating radius should be considered in revising our estimation of N_{new1} to be more exact. For this end, we should describe the assumption that each node at the same level of GNR-enumeration tree includes same number of child nodes. This is clear that GNR-enumeration is a pre-order tree search. The root of this tree which corresponds with b_{Cut}^{*}, against the ordinary trees, has two nodes with coefficient of $w_{\mathrm{Cut}}=0$ or $w_{\mathrm{Cut}}=1$. This assumption is illustrated with a simple example as follows:

Lets assume that for an input lattice block of $\mathcal{L}_{[1,6]}=\{b_1^{*},b_2^{*},b_3^{*},b_4^{*},b_5^{*},b_6^{*}\}$ *with block size of* $d=6$*, this enumeration tree has the depth of 5 (i.e.,* $Cut=5$*). Also assume the following number of nodes at each level after four types of pruning (which is computed in line 7 from Algorithm 2):* $H_{l=2}^{new}=2$, $H_{l=3}^{new}=4$, $H_{l=4}^{new}=8$, $H_{l=5}^{new}=5$, $H_{l=6}^{new}=3$*; The total number of nodes in pre-order search (with no update of radius) is* $N_{new1}=\sum_{l=d-Cut+1}^{d}H_l^{new}=22$*. Now by the assumption that each node at the same level of GNR-enumeration tree includes same number of child nodes, each nodes in root (corresponding with* $l=2$ *and GSO vector of* b_5^{*}*) has* $\frac{H_{l=3}^{new}}{H_{l=2}^{new}}=2$ *child nodes, each nodes in level* $l=3$ *has* $\frac{H_{l=4}^{new}}{H_{l=3}^{new}}=2$ *child nodes, each nodes in level* $l=4$ *has* $\frac{H_{l=5}^{new}}{H_{l=4}^{new}}=0.625$ *child nodes, each nodes in level* $l=5$ *has* $\frac{H_{l=5}^{new}}{H_{l=6}^{new}}=0.6$ *child nodes, and the nodes in level* $l=6$ *are leaf nodes;*

By this example, we introduce an outline of our main assumption in Lemma 3. In fact, we use this assumption to determine the approximate number of nodes at each level which should be visited between two specific nodes in pre-order search of GNR-enumeration tree. For this case, again the previous example can be used, and it is asked to determine the approximate number of nodes at each level which are visited after 5th node up to 14th node in pre-order search. There are 9 nodes which should be visited after node of $i=5$ to reach the node of $j=14$, so by using our main assumption, the number of nodes at each level l, which should be visited between these two specific nodes of i and j, can be estimated by $\frac{j-i}{N_{new1}}H_l^{\mathrm{new}}$ (e.g., for level $l=4$, this number of node is $\approx 3.\overline{27}$);

Three states are considered for output of this algorithm:

- If parameter of "abort" would be "true", then Algorithm 2 (at line 22) returns the total nodes of enumeration as "$\mathrm{Solution}[1]\#\mathrm{index}$", and samples the solution norm as "$\mathrm{Solution}[1]\#\mathrm{norm}$":
$$N_{\mathrm{new2}}=\sum_{l=d-\mathrm{Cut}+1}^{d}\frac{\mathrm{Solution}[t=1]\#\mathrm{index}-\mathrm{last}_{\mathrm{idx}}}{N_{\mathrm{new1}}}H_l^{\mathrm{new}}\left(\frac{R_{\mathrm{new}}}{R}\right)^{l}=\mathrm{Solution}[t=1]\#\mathrm{index},$$
where $R_{\mathrm{new}}=R$ *and* $\mathrm{last}_{\mathrm{idx}}=0$.
- Also, if expected number of final solutions would be $K=0$, then lines of 17 to 23 are not performed, and lines of 24 and 25 are only performed and finally this algorithm returns the total nodes of enumeration as "N_{new1}" and samples the solution norm as "R";
- After sampling the solution indices in pre-order search of GNR-enumeration in lines of 10 to 15 from Algorithm 2, by using our proposed idea, the number of enumeration nodes, after finding a solution and before finding next solution (lines 19 to 20 from Algorithm 2) or finishing the search of enumeration tree (lines 24 to 25 from Algorithm 2), are estimated. Also, by using the factor of $\left(\frac{R_{\mathrm{new}}}{R}\right)^{l}$ in lines 20 and 25 from Algorithm 2, we apply the process of updating radius in estimation of total nodes of enumeration (in the way that it is discussed for our example in this proof). Moreover, this is clear that the final solution norm which is returned by GNR-enumeration is equal

to the last update of enumeration radius (as last setting of R_{new} in Algorithm 2).

D. Revised Generation of Bounding Function

To find better solution vector, it is reasonable to run enumeration function over bigger block sizes (i.e., cutting point of $\text{Cut} < d$ is not pleasant). Also, by forcing $\text{Cut} = d$, it is easier to generate of a bounding function with an intended success probability by relation (36) in Lemma 2. Moreover, by forcing $\text{Cut} = d$, some other functions, relations, propositions and formulations in BKZ-simulation can be simplified too. Following lemma formally defines our technique to force $\text{Cut} = d$:

Lemma 4. The bounding function $\mathcal{R}$ with dimension d with our revised success probability defined by (36) can be generated as follows:

$$\mathcal{R}_{i+1}^2 \leftarrow \left(1 - \frac{\|b_d^*\|^2}{R^2}\right)\mathcal{R}_i'^2 + \frac{\|b_d^*\|^2}{R^2}, \tag{50}$$

where $1 \le i \le d-1$ *and* $\mathcal{R}_1^2 \leftarrow \frac{\|b_d^*\|^2}{R^2}$ *and* $p_{succ}^{new1}(\mathcal{L}_{[1,d]}, \mathcal{R}, R) \approx p_{succ}(\mathcal{R}') \times p_{succ}(1^d, R, \mathcal{L}_d)$ *and bounding function* $\mathcal{R}'$ *with dimension* $d-1$ *and* $p_{succ}(\mathcal{R}')$ *defined by* (15).

Proof. Assume that the bounding function of $\mathcal{R}'$ with dimension $d-1$ in this lemma is defined as follows:

$$\mathcal{R}_i'^2 = \frac{R^2\mathcal{R}_{i+1}^2 - \|b_d^*\|^2}{R^2 - \|b_d^*\|^2}, \qquad 1 \le i \le d-1.$$

The corresponding success probability of $\mathcal{R}'$ is defined by (15), as follows:

$$p_{succ}(\mathcal{R}') = \Pr_{u \backsim Ball_{d-1}}\left(\forall\, i \in [1, d-1], \textstyle\sum_{l=1}^{i} u_l{}^2 \le \mathcal{R}_i'^2\right) =$$

$$\Pr_{u \backsim Ball_{d-1}}\left(\forall\, i \in [1, d-1], \textstyle\sum_{l=1}^{i} u_l{}^2 \le \frac{R^2\mathcal{R}_{i+1}^2 - \|b_d^*\|^2}{R^2 - \|b_d^*\|^2}\right) =$$

$$\Pr_{u \backsim Ball_{d-1}}\left(\forall\, t \in [2, d], \textstyle\sum_{l=1}^{t} u_l{}^2 \le \frac{R^2\mathcal{R}_t^2 - \|b_d^*\|^2}{R^2 - \|b_d^*\|^2}\right) \Rightarrow$$

By using (41):

$$p_{succ}(\mathcal{R}') = \Pr_{u \backsim Ball_{l=d}}^{new2}(\mathcal{L}_{[1,d]}, \mathcal{R}, R, \mathcal{G} = \text{Cut}) =$$

$$\Pr_{u \backsim Ball_{\mathcal{D}=l-d+\text{Cut}-1}}\Big(\forall t \in [d - \text{Cut} + 2, l], \textstyle\sum_{i=d-\text{Cut}+2}^{t} u_i^2 \le \frac{R^2\mathcal{R}_t^2 - \|b_{\text{Cut}}^*\|^2}{R^2 - \|b_{\text{Cut}}^*\|^2}\Big).$$

Because of $\mathcal{R}_1^2 = \frac{\|b_d^*\|^2}{R^2}$ (see relation (50)), the cutting point is $\text{Cut} = d$, and summation of $\sum_{\mathcal{J}=1}^{\text{Cut}=d} \text{Prob}(\mathcal{G} = \mathcal{J})$ equals to 1. Also, since enumeration radius is not changed, and Gaussian Heuristic of $\mathcal{L}_{[1,d]}$ is close to Gaussian Heuristic of $\mathcal{L}_{[1,d-1]}$ (by relation (5)) as $\text{GH}(\mathcal{L}_{[1,d-1]}) \approx \text{GH}(\mathcal{L}_{[1,d]})$, so the enumeration radius factor r_{FAC} is not changed nearly for $\mathcal{L}_{[1,d-1]}$ and $\mathcal{L}_{[1,d]}$. Accordingly, by using (34) and Remark 4:

$$p_{succ}(\mathcal{R}') \times p_{succ}(1^d, R, \mathcal{L}_d) = p_{succ}^{\text{Approx1}}(\mathcal{L}_{[1,d]}, \mathcal{R}, R) \approx p_{succ}^{new1}(\mathcal{L}_{[1,d]}, \mathcal{R}, R).$$

This proof is completed.

By using Lemma 4, Remark 5 generates the extreme/non-extreme bounding function with given dynamic success frequency f_0 which is estimated by (47).

Remark 5. The bounding function $\mathcal{R}$ with dimension d and given dynamic success frequency f_0 estimated by (47), can be generated by proposed three steps in Fig. 1:

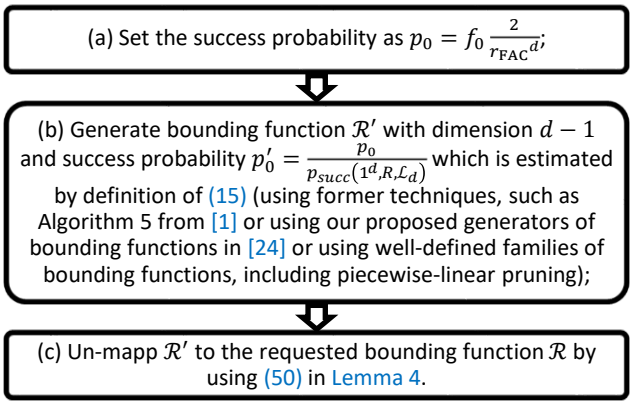

Fig. 1: Steps of Generating bounding function $\mathcal{R}$ with dimension d and success frequency f_0 estimated by (47).

Results and Discussion

In this section, our test results show the simulation/experimental outcomes of our contributions in this paper. The tests in this paper use the random instances of SVP lattice challenges in the sense of Goldstein and Mayer [17], [18]. Also, two libraries of fplll library [19] and NTL library [20] are used for these tests. All the implementations and simulations are compiled with MSVC x64 bit C++. These tests use the following hardware platform: ASUS motherboard series Z97-K, Intel® Core™ i7-4790K processor with the base frequency of 4 GHz, 16 GB RAM; also, the running times are provided only for a single real-core.

A. Our Estimations for Parameter of $\sqrt{\Upsilon}$

As mentioned, former studies usually use the initial enumeration radius parameter of $\sqrt{\Upsilon} = 1.05$, but Theorem 2 defines the optimal initial radius parameter $\sqrt{\Upsilon}$ (as optimal initial radius factor r_{FAC}) and a bound for the norm of shortest vector of lattice blocks in average-case. Our definition in Theorem 2 can be used dynamically to compute optimal enumeration radius in actual running of BKZ-algorithm (or BKZ-simulation). By using relation (34) in Lemma 1, the success prbability of full-enumeration for block sizes of $50 \le \beta \le 240$ in different values of initial radius parameter $\sqrt{\Upsilon}$ (as initial radius factor of r_{FAC}) is shown in Table 1 and Table 2.

Note: The success probability of full-enumeration in former studies is set to 1.

Table 1: Success probability of full-enumeration for $50 \leq \beta \leq 90$ in different values of r_{FAC}

radFac	0.91	0.92	0.93	0.94	0.95	0.96	0.97	0.98	0.99	1	1.01	1.02	1.03	1.04	1.05
β=50	0.00	0.01	0.01	0.02	0.04	0.06	0.10	0.17	0.26	0.39	0.56	0.74	0.89	0.97	1
β=60	0.00	0.00	0.01	0.01	0.02	0.04	0.08	0.14	0.24	0.39	0.60	0.81	0.95	0.99	1
β=70	0.00	0.00	0.00	0.01	0.01	0.03	0.06	0.11	0.22	0.39	0.63	0.86	0.98	1	1
β=80	0.00	0.00	0.00	0.00	0.01	0.02	0.04	0.09	0.20	0.39	0.67	0.91	1	1	1
β=90	0.00	0.00	0.00	0.00	0.00	0.01	0.03	0.08	0.18	0.39	0.71	0.95	1	1	1

Table 2: Success probability of full-enumeration for $100 \leq \beta \leq 240$ in different values of r_{FAC}

radFac	0.95	0.96	0.97	0.98	0.99	1	1.002	1.004	1.006	1.008	1.01	1.012	1.014	1.016	1.018	1.02	1.03
β=100	0.00	0.01	0.02	0.06	0.17	0.39	0.46	0.53	0.60	0.67	0.74	0.81	0.87	0.91	0.95	0.97	1
β=120	0.00	0.00	0.01	0.04	0.14	0.39	0.47	0.55	0.64	0.73	0.81	0.88	0.93	0.97	0.99	1	1
β=140	0.00	0.00	0.01	0.03	0.12	0.39	0.48	0.58	0.69	0.78	0.87	0.93	0.97	0.99	1	1	1
β=160	0.00	0.00	0.00	0.02	0.10	0.39	0.50	0.61	0.73	0.83	0.91	0.97	0.99	1	1	1	1
β=180	0.00	0.00	0.00	0.01	0.08	0.39	0.51	0.64	0.77	0.88	0.95	0.99	1	1	1	1	1
β=200	0.00	0.00	0.00	0.01	0.06	0.39	0.53	0.67	0.81	0.91	0.97	1	1	1	1	1	1
β=220	0.00	0.00	0.00	0.01	0.05	0.39	0.54	0.70	0.85	0.94	0.99	1	1	1	1	1	1
β=240	0.00	0.00	0.00	0.00	0.04	0.39	0.55	0.73	0.88	0.97	1	1	1	1	1	1	1

Also for block sizes of $50 \leq \beta \leq 240$, our proposed bound of radius factor $\sqrt{\Upsilon}$, which is defined by formula of (35), is shown in Table 3.

Table 3: Our proposed lower-bound/upper-bound for initial radius factor $\sqrt{\Upsilon}$ for $50 \leq \beta \leq 240$ with assumption of 100 middle random lattice blocks

Block Size	$\mathbf{radFac_{min}}$	$\mathbf{radFac_{opt}}$
$\boldsymbol{\beta = 50}$	0.925	1.045
$\boldsymbol{\beta = 60}$	0.937	1.038
$\boldsymbol{\beta = 70}$	0.946	1.032
$\boldsymbol{\beta = 80}$	0.952	1.028
$\boldsymbol{\beta = 90}$	0.958	1.025
$\boldsymbol{\beta = 100}$	0.962	1.022
$\boldsymbol{\beta = 120}$	0.968	1.019
$\boldsymbol{\beta = 140}$	0.972	1.016
$\boldsymbol{\beta = 160}$	0.976	1.014
$\boldsymbol{\beta = 180}$	0.979	1.012
$\boldsymbol{\beta = 200}$	0.981	1.011
$\boldsymbol{\beta = 220}$	0.982	1.01
$\boldsymbol{\beta = 240}$	0.984	1.009

By assuming the number of 100 middle random lattice blocks in BKZ running, Table 3 introduces the optimal initial radius parameter of $r_{\mathrm{FAC_{opt}}}$ for this number of random lattice blocks in middle of BKZ with full-enumeration success probability of $\mathrm{p_{opt}} = p_{succ}\left(1^{\beta}, R, \mathcal{L}_{\beta}\right) = 0.99$. Also, Table 3 introduces the minimum radius parameter $r_{\mathrm{FAC_{min}}}$ for this number of random lattice blocks in middle of BKZ with full-enumeration success probability of $\mathrm{p_{min}} = p_{succ}\left(1^{\beta}, R, \mathcal{L}_{\beta}\right) = 0.01$ (see our discussions in third section (Part A)). By these estimations, the values of $r_{\mathrm{FAC_{opt}}}$ in Table 3 can be used instead of initial radius parameter of $\sqrt{\Upsilon} = 1.05$.

B. Test Results for Our Revision of Success Probability

By definition of cutting point in [12], this is found that if this point (cutting point) would be less than the block size, then in some specific cases (e.g., the demanded success probability is extremely small or the input block is much reduced), former studies (such as [1], [13]) may generate bounding functions with some success probabilities which unintentionally becomes less than intended one! To show the importance of this case, a test is introduced to show that GNR-enumeration with extremely small success probability generated by [1], [13] over strong-reduced bases (nearly HKZ-reduced bases)

can be actually smaller than estimated one by relation (15). This test uses following studies for comparison:

- The estimation of success probability by Chen-Nguyen [1] in (25),
- The estimation of dynamic success frequency by Aono et al. [1], [3] in (26),
- Monte-Carlo estimation of success probability [13] by condition of (20).

This test uses following bounding functions: full-enumeration, some bounding functions with no known families (for success probabilities of 0.25, 0.5, 0.6, 0.7, 0.8, 0.9, 0.95 which are estimated by Monte-Carlo), linear-pruning (with success probability of 0.01) and five piecewise-linear bounding function with parameters of "a = 0.4", "a = 0.3", "a = 0.2", "a = 0.1", "a = 0.05". The entries of these bounding function plotted on Fig. 2; The success probability of these bounding functions is defined by (15) which uses Monte-Carlo estimation.

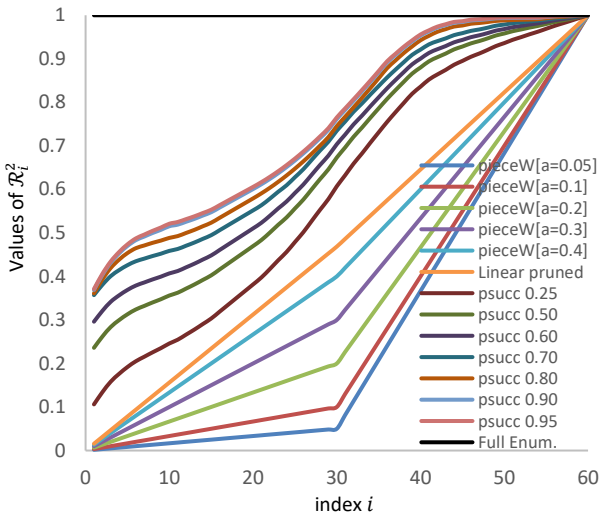

Fig. 2: Some bounding functions with different success probabilities.

The quality of randomization, LLL-reduction, and nearly HKZ-reduction for 20 random lattice bases in the sense of Goldstein and Mayer [17], [18] in dimension of $n = 60$ is illustrated in Fig. 3; In this test, for randomization of lattice blocks, the re-randomization strategy of fplll library [19] is used, which works by permuting basis vectors and the triangular transformation matrix with coefficients of $\{-1,0,1\}$, also for LLL reduction the parameter of $\delta = 0.99$ is set, finally for nearly HKZ reduction, this paper uses $\mathrm{BKZ}_{\beta=60}$ from NTL library [20].

The quality of these three types of reduction are shown by GSO norms of $\|b_i^*\|^2$ which are plotted in log2 form in Fig. 3.

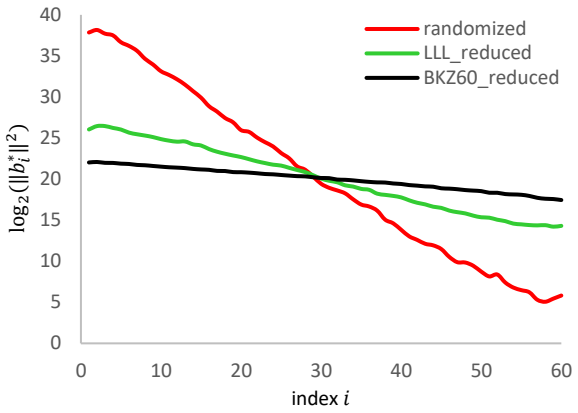

Fig. 3: Quality of randomized/LLL-reduced/nearly HKZ-reduced basis with dimension 60.

For determining the cutting point in Fig. 4, the entries of bounding function (i.e., $\mathcal{R}_i^2$ for $1 \le i \le 60$) are scaled by multiplying with squared value of enumeration radius (i.e., in the form of $R^2\mathcal{R}_i^2$). The initial radius parameter in this test is set to $\Upsilon = 1.13$, so the squared value of enumeration radius would be $R^2 = 1.13 \times \mathrm{GH}^2(\mathcal{L}_{[1,60]})$. This is worthy of noting that the indices of entries in bounding function $\mathcal{R}_i^2$ correspond with the inverse of indices in squared GSO norm of $\|b_i^*\|^2$ (i.e., in Fig. 4, the value of $R^2\mathcal{R}_{\beta-i+1}$ corresponds with $\|b_i^*\|^2$). The values of squared GSO norm of $\|b_i^*\|^2$ and values of $R^2\mathcal{R}_{\beta-i+1}$ in Fig. 4 are plotted in form of $\log_2$ (the parameter of "GH^2" in Fig. 4 represents the squared value of the shortest vector norm estimation in (5) by Gaussian Heuristic).

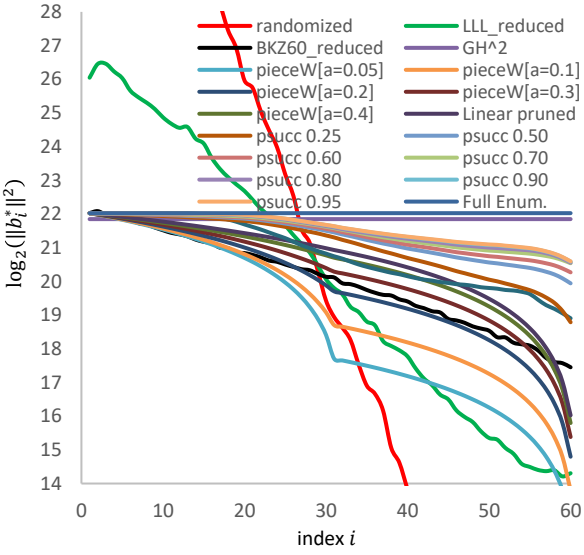

Fig. 4: Different qualities of basis and different bounding functions with scaled entries of $R^2 \times \mathcal{R}_{\beta-i+1}$ in dimension 60 to find the cutting point.

Fig. 4 shows that the BKZ_{60}-reduced bases are nearly cut with linear pruning at index of $\mathrm{Cut}=58$, cut with piecewise-linear with parameter $\mathrm{a}=0.4$ at index of $\mathrm{Cut}=57$, cut with piecewise-linear by parameter of $\mathrm{a}=0.3$ at index of $\mathrm{Cut}=54$, cut with piecewise-linear by parameter of $\mathrm{a}=0.2$ at index of $\mathrm{Cut}=26$, cut with piecewise-linear by parameter of $\mathrm{a}=0.1$ at index of $\mathrm{Cut}=20$, and cut with piecewise-linear by parameter of $\mathrm{a}=0.05$ at index of $\mathrm{Cut}=17$; Other bounding functions in Fig. 4 have cutting point of $\mathrm{Cut}=60$ for GSO norms of three types of reduced basis. In Table 4, our test results show the comparison of our revised estimation of success probability in (36) and our revised estimation of dynamic success frequency in (47) with some former estimations including: success probability by Chen-Nguyen technique [1] (see relation (25)), dynamic success frequency by Aono et al. [3] (see relation (26)), and static success probability by Monte-Carlo estimator with condition of (20) [13]. By using the initial radius parameter of $\Upsilon=1.13$, the count of solutions in full-enumeration tree would be estimated as $\approx \frac{r_{\mathrm{FAC}}^{\beta}}{2} \approx \frac{(\sqrt{\Upsilon})^{\beta}}{2} \approx \frac{(\sqrt{1.13})^{60}}{2} \approx 19.6$;

Table 4: Comparison of our revised estimation of success probability and dynamic success frequency with former estimations in [1], [3], [13] over nearly HKZ-reduced bases in dimension 60

	Cut Point	p_{succ} by Monte Carlo [13]	p_{succ} by Chen-Nguyen [1]	f_{succ} by Aono et al. [3]	p_{succ} by our estimator of (36)	f_{succ} by our estimator of (47)
PieceWise[a=0.05]	17	-	$2^{-44.6}$	$2^{-40.3}$	0	0
PieceWise[a=0.1]	20	-	$2^{-30.5}$	$2^{-26.2}$	0	0
PieceWise[a=0.2]	26	$2^{-19.8}$	$2^{-17.4}$	$2^{-13.1}$	0	0
PieceWise[a=0.3]	54	$2^{-12.9}$	$2^{-10.7}$	0.012	$2^{-19.8}$	2^{-16}
PieceWise[a=0.4]	57	0.0024	0.01	0.195	$2^{-11.8}$	0.005
Linear-pruning	58	0.01	0.036	0.7	0.003	0.046
BF[p_{succ}=0.25]	60	0.25	0.48	9.4	0.4	7.8
BF[p_{succ}=0.5]	60	0.5	0.82	16	0.77	15.1
BF[p_{succ}=0.6]	60	0.6	0.9	17.6	0.87	17
BF[p_{succ}=0.7]	60	0.7	0.95	18.6	0.93	18.3
BF[p_{succ}=0.8]	60	0.8	0.98	19.1	0.96	18.9
BF[p_{succ}=0.9]	60	0.9	0.993	19.4	0.98	19.2
BF[p_{succ}=0.95]	60	0.95	0.995	19.5	0.99	19.3
Full-Enum.	60	1	1	19.6	1	19.6

However, using $r_{\mathrm{FAC}} \leq 1$ is not a common practice, but in final rounds of BKZ-reduction with high block sizes, this may be observed! Therefore by using the concepts of full-enumeration success probability in third section (Part A), if the enumeration radius factor in this test reaches to $r_{\mathrm{FAC}} \approx 0.98$, this is expected that all the numerical results in column of "p_{succ} by our estimator of (36)" and "f_{succ} by our estimator of (47)" are decreased by factor of 0.01 (see Table 1 and Table 2 in fourth section (Part A)). As shown in Table 4, when the input basis (or lattice block) is strongly reduced (near to HKZ-reduced) and the success probability of bounding function is extremely small (near to extreme pruning), the actual value of success probability and dynamic success frequency can be decreased asymptotically. This test focuses on moderate block sizes ($\beta=60$), while for bigger block sizes, this problem is relaxed automatically! As the block sizes are increased, even for extreme-pruned bounding functions over HKZ-reduced lattice blocks, the cutting point stays around the size of β, so this special case (which makes the

value of success probability dropped asymptotically) cannot be observed for high block size! However for some special setting, this can be seen even for high block sizes yet; For example, Fig. 5 shows the average shape of 7 HKZ-reduced bases with dimension 200 and different piecewise-linear bounding functions. As shown in Fig. 5, the cutting points of all the bounding functions nearly are equal to 200, but piecewise-linear bounding function by parameter of "a=0.01", with extreme pruning and estimated success probability $\approx 2^{-246}$ by relation (25), has the cutting point of $\mathrm{Cut} = 84$, and consequently the estimated success probability and dynamic success frequency of it would be zero by our formulas in (36) and (47)!

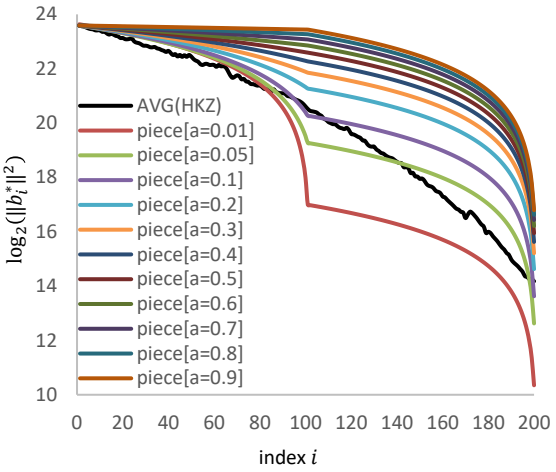

Fig. 5: Different shapes of quality of basis and different bounding functions with scaled entries of $R^2\mathcal{R}_{\beta-i+1}$ in dimension 200 to find the cutting point.

C. Test Results for Our Revision of Enumeration Cost

In this section, the exactness of our revised estimation of GNR-enumeration cost by Algorithm 2 is compared with Chen-Nguyen estimation of GNR-enumeration cost in Algorithm 8 of Appendix A from [1] (see relation (23) and (19)). For actual implementation of GNR-pruned enumeration function, the pseudo-code in Appendix B from [13] is used, but after each success of the enumeration function, the enumeration is not aborted, rather, the best solution and enumeration radius are updated (similar to the pseudo-code in Appendix B from [1]). The cost of experimental running of GNR-enumeration is determined by using a counter in actual enumeration function which counts the number of enumeration tree nodes. This test uses piecewise-linear bounding function with different success probabilities. The bounding functions which are used in this test, use mapping technique (see Lemma 4 and Remark 5 in third section (Part D)). The initial enumeration radius factor in this test is set to $\Upsilon = r_{\mathrm{FAC}} = 1.2$; Table 5 shows our numerical results for this comparison. In Table 5, parameter of "f_{succ}" represents the Dynamic Success Frequency and parameter of "p_{succ}" represents the success probability. This test uses some random lattice basis of dimension 70 in the sense of Goldstein and Mayer [17], [18].

Note: Although floating point arithmetic is known to cause some stability problems during LLL reduction, based on the experiences in [13], such problems during enumeration (even up to the dimension of 110) are not seen.

Table 5: Comparison of our revised estimation of GNR-enumeration cost (in Algorithm 2) with the cost estimation proposed by [1] and the cost computed in experimental running of enumeration

Enum. Cost by compared cases / Success Frequency & Success Probability	Enumeration cost by experimental running	Enumeration cost by our estimator in Algorithm 2	Enumeration cost by Chen-Nguyen in [1]
f_{succ}=0.01 & $p_{succ}=2^{-24}$	$2^{9.5}$	$2^{8.4}$	$2^{12.2}$
f_{succ} =0.1 & $p_{succ}=2^{-21}$	$2^{10.2}$	$2^{11.35}$	$2^{14.21}$
f_{succ}=1 & $p_{succ}=2^{-17}$	2^{11}	$2^{13.5}$	$2^{16.82}$
$f_{succ}=10^1$ & $p_{succ}=2^{-14}$	$2^{12.4}$	$2^{15.7}$	$2^{20.04}$
$f_{succ}=10^2$ & $p_{succ}=2^{-11}$	$2^{14.6}$	$2^{18.68}$	$2^{23.9}$
$f_{succ}=10^3$ & $p_{succ}=2^{-7}$	2^{18}	$2^{21.28}$	$2^{28.52}$
$f_{succ}=10^4$ & p_{succ}=0.06	$2^{23.3}$	$2^{23.78}$	$2^{34.63}$
$f_{succ}=10^5$ & p_{succ}=0.57	$2^{36.47}$	$2^{34.45}$	$2^{47.81}$
f_{succ}=130964 & p_{succ}=0.75	$2^{39.67}$	$2^{37.12}$	$2^{50.2}$

As shown in Table 5, this is clear that, the cost results by our proposed estimator of GNR-enumeration cost in Algorithm 2 are closer to the cost determined in experimental running of enumeration, than the closeness of enumeration cost by Chen-Nguyen estimator (in Algorithm 8 from Appendix A in [1]) to this experimental running cost.

Conclusions

BKZ algorithm has a determinative role in security analysis of lattice-based cryptographic primitives, therefore the total cost of BKZ and quality of output basis 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. Designing a BKZ-simulation with GNR-pruned enumeration needs to some necessary building-blocks which includes definition of 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 some exact definition of optimal enumeration radius, generation of bounding function, estimation of success probability and GNR enumeration cost. Our contributions and results in this paper are described as follows:

- ***Formal definition of optimal enumeration radius.*** By definition of full-enumeration success probability in this paper, the optimal value for radius parameter $\sqrt{\Upsilon}$ (as initial radius factor r_{FAC}) and corresponding bound for solution norm of full-enumeration are defined exactly in Theorem 2 in average case (see our estimations in fourth section (Part A)). This definition can be used dynamically to compute optimal enumeration radius in BKZ simulation and even actual running of BKZ algorithm. In other sides, former studies on BKZ-simulation don't use optimal version of the radius parameter of r_{FAC}. Paper [1] uses as an invariant factor just based on some limited experimental observations (see Figure 3 in [1]), paper [3] uses non-exact assumption of GSA to determine r_{FAC} dynamically for each block size (see relation (9) in [3]), and paper [2] uses no new idea to make the exactness of radius factor better (to the best of our knowledge).

Test Results. Against the success probability of full-enumeration in former studies which is set to 1, our exact estimation of success probability in full-enumeration, for some practical range of block sizes of $50 \leq \beta \leq 240$, is shown in this paper for different values of r_{FAC} based on our proposed theorem (Theorem 2). Also for block sizes of $50 \leq \beta \leq 240$, our better bound of radius factors of $\sqrt{\Upsilon}$ defined by Theorem 2, is introduced in this paper.

- ***Revised estimation of success probability for GNR bounding function.*** The former studies [1]-[3] use the efficient idea by [1] to estimate the success probability of GNR-enumerations (see formulas (15), (23) and (24)); In fact, the estimation by [1] only considers the pruning type by condition of (20) for cylinder-intersection of bounding function; This paper proposes to consider three more types of pruning in estimation of success probability (see our discussions at the beginning of third section); All of these four types of pruning are applied collectively in our estimation of success probability in relation (36); Our results in fourth section (Part B) shows non-negligible gap of our exact estimation of (36) from former estimations in some cases.

Test Results. Our revised estimation of success probability (for GNR bounding function) in our test results on (nearly) HKZ-reduced bases in dimension 60, shows non-negligible gap from former estimations by some main former studies of [1], [3], [13]. Also to have better sense, this paper shows the shape of bounding functions with different success probabilities and the shapes of quality of randomized/LLL-reduced/nearly-HKZ-reduced bases with dimension 60 (also 200) and the corresponding cutting points.

- ***Revised cost estimation of GNR-enumeration.*** The former studies [1]-[3] use the efficient idea by [1] to estimate the cost of GNR-enumerations (see formulas (19), (23) and (24)); Similar to success probability, the cost estimations in [1] only consider the pruning type by condition of (20); Our paper considers all of four proposed types of pruning in estimation of GNR-enumeration cost along with the process of updating enumeration radius in Algorithm 2; Our results in fourth section (Part C) shows the exactness of the estimation by Algorithm 2 against the former studies.

Test Results. Our results show that the cost results by our proposed estimator of GNR-enumeration cost in Algorithm 2 are closer to the cost determined in experimental running of enumeration, than the difference of enumeration cost results by Chen-Nguyen estimator in Algorithm 8 from Appendix A in [1] against the experimental cost results.

- ***A novel technique in generation of bounding function.*** By using Lemma 4, this is possible to generate a bounding function including cutting point of $\mathrm{Cut} = d$ (see Remark 5); In former studies [1]-[3], if the simulation tries to generate bounding functions with much small success probability, this is possible that the success probability of this bounding functions unintentionally becomes much less than intended one

or even zero (because of ignoring the cutting points which are less than the block size, i.e., Cut $< d$; see our results and discussions in fourth section (Part B)).

This is worthy of noting that if we use another SVP-solver instead of GNR-enumeration (e.g., sieving algorithm in [22], enumeration by integrating sparse orthogonalized integer representations in [23], etc.), none of our contributions can be used in BKZ algorithm or BKZ-simulation!

Future Works. Three of our proposed components in this paper (include optimal enumeration radius, generation of bounding function and estimation of success probability) can be used in actual running of BKZ-algorithm, such as our technique of "*BKZ with Progressive Success Probabilities*"[21], [25] which massively generates bounding functions with different success probabilities, and consequently introduce new lattice-reduction security estimates to fix the problem of non-exactness in bit-security estimations of current cryptography schemes (e.g., [26]-[28]), so this is worthy of re-estimating their bit-securities by our revised components. Also Algorithm 2 in this paper samples the norm of final solution which can be used in our revised method for sampling coefficient vector of GNR-enumeration solution in [29]. Moreover, the authors suggest the formal verification and proof of Algorithm 2 corresponding with each claims in Lemma 3, by some theorem provers such as Isabelle/HOL (see our similar works in [30]). At the end, nearly all components and concepts introduced in this paper can be used in design of new BKZ-simulation with better exactness, and consequently it is expected to use this new BKZ-simulation in introducing new lattice-reduction security estimates (as bit-security level by reduction).

Author Contributions

Gholam Reza Moghissi suggested the innovations and wrote the manuscript with the guidance of Dr. Ali Payandeh.

Acknowledgment

The authors gratefully thank the anonymous reviewers and the editor of JECEI for their useful comments and suggestions.

Conflict of Interest

The authors declare no potential conflict of interest regarding the publication of this work. In addition, the ethical issues including plagiarism, informed consent, misconduct, data fabrication and, or falsification, double publication and, or submission, and redundancy have been completely witnessed by the authors.

Abbreviations

$\mathcal{L}(b_1 \dots b_n)$	A lattice by basis vectors of $b_1 \dots b_n$
$\mathcal{L}(B)$	A lattice by basis matrix of B
$\mathcal{L}_{[j,k]}$	A lattice by GSO-projected basis vectors of $\pi_j(b_j) \dots \pi_j(b_k)$
$\det B$	Determinant of basis matrix of B
$\lambda_1(\mathcal{L})$	First Successive-Minima of lattice $\mathcal{L}$
$\|v\|$	Euclidean norm of lattice vector v
SVP	Shortest Vector Problem
LLL	Lenstra-Lenstra-LovÃasz algorithm
BKZ	Block Korkin-Zolotarev algorithm
β	Input parameter of Lattice block size
d	Lattice block size in running BKZ which is varied from β to 2
BKZ_{60}	BKZ algorithm with block size $\beta = 60$
HKZ	Hermite-Korkine-Zolotarev algorithm
GSA	Geometric Series Assumption
Enum	Enumeration
$\mathrm{Vol}(\mathcal{L}(B))$	Volume of lattice $\mathcal{L}(B)$
$Ball_n(R)$	n-dimensional sphere with radius R
$\mathrm{Vol}(Ball_n(R))$	Volume of $Ball_n(R)$
$V_n(R)$	Volume of $Ball_n(R)$
$\Gamma(x)$	Gamma function with parameter x
$\mathrm{GH}(\mathcal{L})$	The estimation of value of $\lambda_1(\mathcal{L})$ by Gaussian Heuristic of lattice $\mathcal{L}$
GSO	Gram-Schmidt Orthogonal
$B^*_{[1,d]}$	GSO basis of lattice $\mathcal{L}_{[1,d]}$ as $[b_1^*, b_2^*, \dots, b_d^*]$
GSO norms $B^*_{[1,d]}$	The norms of $\|b_1^*\|, \dots, \|b_d^*\|$
$\pi_i(b_i)$	i-th vector of GSO basis of B^*
b_i^*	Another notation for $\pi_i(b_i)$
$\mu_{i,j}$	GSO coefficient of i, j as $\mu_{i,j} = \frac{b_i b_j^*}{\|b_j^*\|^2}$
$\mathrm{Gamma}(x; k, \theta)$	Gamma distribution function
$\mathrm{Expo}(x; \lambda)$	Exponential distribution function

1^x	A vector with length of x whose all entries are 1
GNR	Gamma-Nguyen-Regev (pruning)
$\pi_j(b_j, \dots, b_k)$	The projected form of the lattice block of $[b_j, \dots, b_k]$ whose vectors are projected on the vectors of $(b_1, \dots, b_{j-1})$
$\mathcal{L}(b_j, \dots, b_k)$	Another notation for $\pi_j(b_j, \dots, b_k)$
N	Number of total nodes of full-enumeration tree
N'	Total number of nodes in GNR pruned enumeration tree
H_l	Gaussian Heuristic prediction of number of nodes at level l in full-enumeration tree
H'_l	Gaussian Heuristic prediction of number of nodes at the level l in GNR pruned enumeration tree
R	Radius of *n*-dimensional ball in enumeration tree
$C_{R_1 \dots R_l}$	l-dimensional cylinder-intersection with radii of $[R_1, \dots, R_l]$
$\mathcal{R}$	Vector of $\mathcal{R} = [\mathcal{R}_1, \mathcal{R}_2, \dots, \mathcal{R}_\beta]$ as the bounding function
$p_{succ}(\mathcal{R})$	Success probability of bounding function $\mathcal{R}$
$\boldsymbol{\gamma_n}$	Hermite's constant
$\mathbf{Pr}_{u \backsim Ball_d}$	Probability of visiting GSO partial solution candidates in level l from GNR enumeration tree by assuming vector u is chosen uniformly distributed from d-dimensional ball of the radius 1
$\mathbf{Pr}^{new1}_{u \backsim Ball_l}$	Our revised version of $\Pr_{u \backsim Ball_d}$ including cut point of Cut
$\mathbf{Pr}^{new2}_{u \backsim Ball_l}$	A minor revision of $\Pr^{new1}_{u \backsim Ball_l}$
$\mathbf{Pr}^{new3}_{u \backsim Ball_l}$	Our revised version of $\Pr^{new2}_{u \backsim Ball_l}$ with any last non-zero index of $\mathcal{g} = j$
$\boldsymbol{p}^{\mathbf{Approx1}}_{\boldsymbol{succ}}(\mathcal{L}_{[1,d]}, \mathcal{R},$	Our approximation of success probability $p^{new1}_{succ}(\mathcal{L}_{[1,d]}, \mathcal{R}, R)$
$\boldsymbol{r}_{\mathbf{FAC}}$	Radius parameter, defined as $\frac{R}{GH(\mathcal{L})}$
$\mathbf{Vol}(C_{R_1,\dots,R_l})$	Volume of cylinder-intersection of $C_{R_1,\dots,R_l}$
$\boldsymbol{V}_{R_1,\dots,R_l}$	Another notation for $\mathrm{Vol}(C_{R_1,\dots,R_l})$
$\boldsymbol{\mathcal{P}_\ell(t_1, \dots, t_\ell)}$	A polytope with radii of $t_1, \dots, t_\ell$
$\mathbf{Vol}\boldsymbol{\mathcal{P}_\ell(t_1, \dots, t_\ell)}$	Volume of polytope $\mathcal{P}_\ell(t_1, \dots, t_\ell)$
$\boldsymbol{p}^{\boldsymbol{new0}}_{\boldsymbol{succ}}(\mathcal{L}_{[1,d]}, \mathcal{R}, R)$	Original version of success probability (the equivalent notation for $p_{succ}(\mathcal{R})$)
$\boldsymbol{p}^{\boldsymbol{new1}}_{\boldsymbol{succ}}(\mathcal{L}_{[1,d]}, \mathcal{R}, R)$	Our version of success probability
$\boldsymbol{f}_{\boldsymbol{succ}}(\mathcal{L}_{[1,d]}, \mathcal{R}, R)$	Original dynamic success frequency
$\boldsymbol{f}^{\boldsymbol{new0}}_{\boldsymbol{succ}}(\mathcal{L}_{[1,d]}, \mathcal{R}, R)$	The equivalent notation for f_{succ}
$\boldsymbol{f}^{\boldsymbol{new1}}_{\boldsymbol{succ}}(\mathcal{L}_{[1,d]}, \mathcal{R}, R)$	Our revised version of dynamic success frequency
$\boldsymbol{C_{Roger}}$	An abstract parameter (not a real parameter) in f^{new}_{succ}, and is set to 1
$\sqrt{\boldsymbol{\Upsilon}}$	Initial radius parameter
$\mathbf{p_{min}}$	Success probability of full-enum. corresponding with $r_{\mathrm{FAC_{min}}}$
$\mathbf{p_{opt}}$	Success probability of full-enum. corresponding with $r_{\mathrm{FAC_{opt}}}$
$\boldsymbol{r}_{\mathbf{FAC_{min}}}$	Minimum hopeful radius parameter
$\boldsymbol{r}_{\mathbf{FAC_{opt}}}$	Optimal radius parameter
$\boldsymbol{R}_{\mathbf{opt}}$	Optimal enumeration radius
Hdown	Maximum index in head concavity
Tup	Minimum index in tail convexity
$\mathbf{randINT}_{[x \dots y]}$	Return a uniformly random integer number between x to y
$\mathbf{rand}_{[x \dots y]}$	Return a uniformly random real number between x to y
$\boldsymbol{\mathcal{g}}$	Last non-zero coefficient index in coefficient vector w (see [12])
$\boldsymbol{w}$	A Coefficient Vector defined for GNR enumeration solution [12]
Cut	GNR enum cut point index (see [12])
$\mathbf{Prob}(\boldsymbol{\mathcal{g}} = \boldsymbol{j})$	Probability distribution of $\mathcal{g}$ for solution vectors of v (see [12])
$\mathbf{E}[\mathrm{X}]$	Expected value of X

K	Sampled number of solutions in GNR-enumeration function
$N_{new1}(\mathcal{L}_{[1,d]}, \mathcal{R}, R)$	Total nodes of GNR-enumeration tree after four types of pruning
H_l^{new}	Gaussian Heuristic prediction of nodes count at level l of GNR enumeration tree (line 7 in Algorithm 2).
$N_{new2}(\mathcal{L}_{[1,d]}, \mathcal{R}, R)$	Total nodes of GNR-enumeration tree after four types of pruning and aborting after finding first solution
$\mathbf{a}$	The parameter of piecewise-linear bounding function

References

[1] Y. Chen, P. Q. Nguyen, "BKZ 2.0: Better lattice security estimates," in Proc. International Conference on the Theory and Application of Cryptology and Information Security: 1-20, Berlin Heidelberg, 2011.

[2] S. Bai, D. Stehlé, W. Wen, "Measuring, Simulating and Exploiting the Head Concavity Phenomenon in BKZ," in Proc. Advances in Cryptology – ASIACRYPT 2018: 369-404, 2018.

[3] Y. Aono, Y. Wang, T. Hayashi, T. Takagi, "Improved progressive BKZ algorithms and their precise cost estimation by sharp simulator," in Proc. Annual International Conference on the Theory and Applications of Cryptographic Techniques: 789-819, Berlin, Heidelberg, 2016.

[4] J. Hoffstein, J. Pipher, J. M. Schanck, J. H. Silverman, W. Whyte, Zhenfei Zhang, "Choosing parameters for NTRUEncrypt," Cryptology ePrint Archive, Report 2015/708, 2015.

[5] M. R. Albrecht, B. R. Curtis, A. Deo, A. Davidson, R. Player, E. W. Postlethwaite, F. Virdia, T. Wunderer, "Estimate all the {LWE, NTRU} schemes!," in Proc. International Conference on Security and Cryptography for Networks, 2018.

[6] M. R. Albrecht, et al., "Estimate all the {LWE, NTRU} schemes!," [Online]. Available at: https://estimate-all-the-lwe-ntru-schemes.github.io/docs/.

[7] "Post-Quantum Cryptography Standardization Project", [Online]. Available at: https://csrc.nist.gov/Projects/post-quantum-cryptography.

[8] J. Sharafi, H. Daghigh, "A Ring-LWE-based digital signature inspired by Lindner–Peikert scheme," J. Math. Cryptology, 16(1): 205-214, 2022.

[9] N. Samardzic, A. Feldmann, A. Krastev *et al.*, "CraterLake: a hardware accelerator for efficient unbounded computation on encrypted data," in Proc. ISCA: 173-187, 2022.

[10] K. Cong, D. Cozzo, V. Maram, N. P. Smart, "Gladius: LWR based efficient hybrid public key encryption with distributed decryption," in Proc. International Conference on the Theory and Application of Cryptology and Information Security: 125-155, Cham, 2021.

[11] T. Espitau, A. Joux, N. Kharchenko, "On a dual/hybrid approach to small secret LWE," in Proc. International Conference on Cryptology in India: 440-462, Cham, 2020.

[12] G. Moghissi, A. Payandeh, "Better sampling method of enumeration solution for BKZ-Simulation," ISC Int. J. Inf. Secur., 13(2): 177-208, 2021.

[13] N. Gama, P. Q. Nguyen, O. Regev, "Lattice enumeration using extreme pruning," in Proc. EUROCRYPT '10, volume 6110 of LNCS. Springer, 2010.

[14] G. R. Moghissi, A. Payandeh, "Rejecting claimed speedup of $2^{\beta/2}$ in extreme pruning and revising BKZ 2.0 for better speedup," J. Comput. Secur., 8(1): 65-91, 2021.

[15] L. Devroye, "Sample-based non-uniform random variate generation," in Proc. the 18th conference on Winter simulation: 260-265, 1986.

[16] Y. Chen, "Reduction de reseau et securite concrete du chiffrement completement homomorphe," PhD thesis, Paris 7, 2013.

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

[18] D. Goldstein, A. Mayer, "On the equidistribution of Hecke points," Forum Math., 15(2): 165-190, Berlin, 2003.

[19] GitHub hosting service, "fplll library project," [Online]. Available at: https://github.com/fplll/.

[20] V. Shoup, "NTL: a library for doing number theory". [Online]. Available at: http://www.shoup.net/ntl/.

[21] G. R. Moghissi, A. Payandeh, "Using progressive success probabilities for sound-pruned enumerations in BKZ algorithm," Int. J. Compu. Network Inf. Secur, 10(9): 10-24, 2018.

[22] L. Ducas, "Shortest vector from lattice sieving: A few dimensions for free,"in Proc. EUROCRYPT: 125-145, 2018.

[23] Z. Zheng, X. Wang, Y. Yu, "Orthogonalized lattice enumeration for solving SVP," Sci. China Inf. Sci. 61: 032115, 2018.

[24] G. R. Moghissi, A. Payandeh, "Optimal bounding function for GNR-enumeration," Int. J. Math. Sci. Comput. (IJMSC), 8(1): 1-17, 2022.

[25] G. R. Moghissi, A. Payandeh, "Design of optimal progressive BKZ with increasing success-probabilities and increasing block-sizes," J. Comput. Secur., 9(2): 65-93, 2022.

[26] D. J. Bernstein *et al.*, NTRU Prime. Technical report, National Institute of Standards and Technology, 2020.

[27] J. Bos *et al.*, "CRYSTALS - kyber: A CCA-secure module-lattice-based KEM," in Proc. 2018 IEEE European Symposium on Security and Privacy (EuroS&P): 353-367, London, UK, 2018.

[28] J. P. D'Anvers *et al.,* SABER. Technical report, National Institute of Standards and Technology (2020).

[29] G. R. Moghissi, A. Payandeh, "Revised method for sampling coefficient vector of GNR-enumeration solution," Int. J. Math. Sci. Comput. (IJMSC), 8(3): 1-20, 2022.

[30] G. R. Moghissi, A. Payandeh, "Formal verification of NTRUEncrypt scheme," Int. J. Comput. Network Inf. Secur., 8(4): 44, 2016.

Biographies

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.

- Email: fumoghissi@chmail.ir
- ORCID: 0000-0001-9189-6786
- Web of Science Researcher ID: NA
- Scopus Author ID: NA
- Homepage: NA

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 the Malek-e-Ashtar University of Technology, Iran. He has published many papers in international journals and conferences. His research interests include information theory, coding theory, cryptography, security protocols, secure communications, and satellite communications.

- Email: payandeh@mut.ac.ir
- ORCID: 9246-9953-0002-0000
- Web of Science Researcher ID: NA
- Scopus Author ID: NA
- Homepage: NA

How to cite this paper:
G. R. Moghissi, A. Payandeh, "Revised estimations for cost and success probability of GNR-enumeration," J. Electr. Comput. Eng. Innovations, 11(2): 459-480, 2023.

DOI: 10.22061/jecei.2023.9228.588

URL: https://jecei.sru.ac.ir/article_1880.html