確率的格子点探索の計算量の下界について *

青野　良範[1]　清藤　武暢[2]　四方 順司[3]

概要：本稿では，Eurocrypt 2010 において提案された Gama-Nguyen-Regev の確率的格子点探索手法に関して，計算量の下界を具体的に計算するための一手法を提案する．この手法は，原論文において導入された以外の仮定を用いず，単純な幾何学的定理から出発しているため効率的な数値計算が可能である．また，数値実験によりこの下界がどの程度正確であるかを検証した．

キーワード：格子，確率的格子点探索，下界，安全性評価

A theoretical cost lower bound of lattice vector enumeration

Yoshinori Aono[1]　Takenobu Seito[2]　Junji Shikata[3]

Abstract: We establish an efficiently computable method to give a cost lower bound for Gama-Nguyen-Regev's extreme pruning technique for lattice vector enumeration published in Eurocrypt 2010. Our lower bound stands on a simple geometric lemma and does not require any heuristic assumptions except for that used in their paper. We also showed the result of our preliminary experiments to show the sharpness of our bound.

Keywords: Lattice, extreme pruning technique, lower bound complexity, security estimation.

1. Introduction

The extreme pruning strategy for lattice vector enumeration introduced by Gama, Nguyen and Regev [10] has a lot of success on lattice reduction algorithms [4], [6], [11] and applications for hardness estimations of lattice based cryptography [2], [7].

On the other side of success, we need to consider its drawbacks in the reproducibility. In particular, to find the optimal complexity of pruned enumeration of a desired probability sharply, it needs to optimize the cost function defined over a sequence of real numbers $0 < R_1 \leq R_2 \leq \cdots \leq R_m = 1$ which are called pruning coefficients. To solve it, the original paper [10] proposed a random perturbation method, which is too slow and is not numerically stable in practice. Since then, modified methods are published: the cross-entropy algorithm in Chen's doctor thesis [5], a modified random perturbation method the progressive BKZ library [4], and the Nelder-Mead method in the fpLLL library [8]. Despite these efforts, the numerical optimization problem is still practically inefficient and we can not know how the sequence $(R_1, \ldots, R_m)$ is far from

[1] 情報通信研究機構，〒 184-8795 東京都小金井市貫井北町 4-2-1. National Institute of Information and Communications Technology, 4-2-1, Nukii-Kitamachi, Koganei, Tokyo, 184-8795, Japan. 第一著者は JSPS 科研費 16H02830 の助成を受けている．

[2] 日本銀行金融研究所情報技術研究センター，〒 103-8660 東京都中央区日本橋本石町 2-1-1. Center for Information Technology Studies, Institute for Monetary and Economic Studies, Bank of Japan, 2-1-1 Nihonbashi-Hongokucho, Chuo-ku, Tokyo, 103-8660, Japan.

[3] 横浜国立大学大学院環境情報研究院，〒 240-8501 横浜市保土ヶ谷区常盤台 79-7. Graduate School of Environment and Information Sciences, Yokohama National University, 79-7 Tokiwadai, Hodogaya-ku, Yokohama, 240-8501, Japan.

* 本稿に示されている意見は，著者たち個人に属し，情報通信研究機構，日本銀行および横浜国立大学の公式見解を示すものではない．The views expressed in this paper are those of authors and do not necessarily reflect the official views of National Institute of Information and Communications Technology, Bank of Japan, or Yokohama National University.

the optimal during the computation.

The same kind of problems can arise if we want to use a simulator that calls a subroutine for simulating pruned enumeration for cryptanalyses. Such simulator based estimation must have errors originated from the numerical instability. It causes troubles in security/hardness estimation of lattice problems/cryptosystems.

Another issues for such types of security estimations is bounding direction. To our best knowledge, some of recent security estimations on lattice cryptosystems are based on mixed use of practical upper/lower bound or average estimation of algorithms [2], [4], [7], [14]. These estimations has been making complicated and has added new (somehow debatable) heuristic assumptions more and more. On the other hand, a few number of investigation has been considered (p. 53 in [16], [6]) nevertheless its importance.

In this paper, we propose a theory and algorithms to overcome the problems, i.e., we provide easily computable and reproducible procedures to compute the cost lower bounds of lattice algorithms.

1.1 Our Contributions

For a given lattice basis $B \in \mathbb{Q}^{n\times m}$, searching radius and success probability, the cost estimation (4) by Gama et al. [10] is the linear combination of $\mathrm{vol}(C_k)$ for $k = 1, \ldots, m$ where C_k is a k-dimensional cylinder intersection parametrized by real numbers $(R_1, \ldots, R_m)$ which are called pruning coefficients. The volumes depend on the combination of R_i, and the best enumeration cost under the fixed parameters is given by the optimal combination; for detail, see the brief overview in Section 2.2 or the original paper [10].

In Section 3, we give a general theory for a simple lower bound of each $\mathrm{vol}(C_k)$ for a fixed success probability and input Gram-Schmidt lengths. The estimation stands on the simple geometrical lemma (Lemma 1) on m-dimensional convex bodies and its projections. The lower bound (9) for single usage of enumeration algorithm is immediate and it can be easily adopted Gama et al.'s extreme pruning strategy that uses $M \geq 2$ randomized reduced bases with probability p/M to achieve total success probability p. We show the complexity is bounded from lower by a linear function on p even if we assume the cost of basis randomization is zero. Thus, there is a limitation on the effect of using many randomized basis in this strategy. Interestingly, the lower bound can be computed without knowing pruning coefficients; thus, it might not be useful to construct the pruning coefficients.

In Section 4, we give our method to bound the cost to solve the lattice problems. Roughly speaking, our targeting lattice problem is defined for given lattice basis and target point, the goal is to find desired points. In our model (15), we separate the algorithm into two part: the lattice reduction part and the enumeration parts. Thus, the lower bound is achieved by searching the minimum of sum of two costs. For this purpose, we need to know the cost and output of lattice reduction part. It is a difficult task if we simulate them sharply. However, using a new assumption (Assumption 1) deduced from our computer experiments, the situation makes much simpler. Roughly speaking, if we use a typical lattice reduction algorithm, the cost of enumeration with input basis can be bounded lower by the cost with a basis satisfying Schnorr's geometric series assumption (GSA), which claims the graph of $\log \|\mathbf{b}_i^*\|^2$ is a line of slope $r < 1$. Although the GSA does not hold in practice from many observations, it is useful to discuss the lower bound cost. This is the new usage of GSA. Hence, the situation what we need to consider is that the output basis of lattice reduction satisfies GSA. It means that we have only one parameter r to optimize.

Accepting the assumption, we can bound the time for lattice reduction as follows. For our best knowledge, all the lattice reduction algorithms except for the LLL must call a subroutine of lattice vector enumeration to find a short vector; here, the length is parametrized by r. Also, the dimension is bounded lower by the Gaussian heuristic assumption. With these information, we can find the lower bound of searching radius, probability, lattice dimension of enumeration that was called from lattice reduction algorithm.

As a simple application, we give the lower bound cost to solve an approximate shortest vector problem in Section 4.4.

2. Preliminaries

For natural numbers $n \leq m$, $[n, m]$ is the set $\{n, \ldots, m\}$ and we denote $[m] := [1, m]$. Throughout this paper, m and k are usually used for the considered and projected dimension respectively.

The gamma and beta functions are defined by

$$\Gamma(a) = \int_0^\infty t^{a-1} e^{-t} dt$$

and

$$B(\alpha, \beta) = \int_0^1 z^{\alpha-1}(1-z)^{\beta-1} dz.$$

The basic relations $\Gamma(a+1) = a\Gamma(a)$ and $B(a,b) = \Gamma(a)\Gamma(b)/\Gamma(a+b)$ hold.

For $m \in \mathbb{N}$, let $B_m(\mathbf{x}, c)$ be the m-dimensional ball whose center is $\mathbf{x} \in \mathbb{R}^m$ and radius $c > 0$. The center-origin ball is $B_m(c) := B_m(\mathbf{0}, c)$. The volume of m-dimensional ball of radius c is $V_m(c) = \frac{\pi^{m/2} c^m}{\Gamma(\frac{m}{2}+1)}$. In particular, we denote $V_m := V_m(1)$. S^m is the surface of $B_m(1)$.

2.1 Incomplete Beta Functions

For $\alpha, \beta > 0$, the incomplete beta function is

$$I_x(\alpha, \beta) := \frac{\int_0^x z^{\alpha-1}(1-z)^{\beta-1}dz}{B(\alpha,\beta)},$$

and its inverse function is defined by $x = I_y^{-1}(\alpha,\beta) \Leftrightarrow y = I_x(\alpha,\beta)$. Both functions are strictly increasing from $[0,1]$ to $[0,1]$.

A simple bound

$$I_x(a,b) \leq \frac{\int_0^x z^{a-1}dz}{B(a,b)} = \frac{x^a}{a \cdot B(a,b)}$$

holds and thus

$$I_x^{-1}(a,b) \geq (aB(a,b)x)^{1/a}. \tag{1}$$

Fact1 Suppose $(x_1, \ldots, x_m) \leftarrow S^m$. Then, $x_1^2 + \cdots + x_k^2$ follows the beta distribution of parameters $(\alpha, \beta) = \left(\frac{k}{2}, \frac{m-k}{2}\right)$. Thus,

$$\Pr_{(x_1,\ldots,x_m)\leftarrow S^m}\left[x_1^2 + \cdots + x_k^2 \leq C\right] = I_C\left(\frac{k}{2}, \frac{m-k}{2}\right) := \frac{\int_0^C x^{\frac{k}{2}-1}(1-x)^{\frac{m-k}{2}-1}dx}{B(\frac{k}{2}, \frac{m-k}{2})}.$$

In particular, $(x_1, \ldots, x_{m-2})$ follows the uniform distribution in $B_{m-2}(1)$.

Corollary1

$$\Pr_{(x_1,\ldots,x_m)\leftarrow B_m(1)}\left[x_1^2 + \cdots + x_k^2 \leq C\right] = I_C\left(\frac{k}{2}, \frac{m+2-k}{2}\right)$$

.

2.2 Lattice, Enumeration Algorithm, and Cost Estimation

For an independent set of vectors $\mathbf{b}_1, \ldots, \mathbf{b}_n \in \mathbb{Q}^m$, the lattice is defined by the set of the all integer linear combination:

$$L(\mathbf{b}_1, \ldots, \mathbf{b}_n) = \left\{\sum_{i=1}^n a_i \mathbf{b}_i : a_i \in \mathbb{Z}\right\}.$$

For a basis, its Gram-Schmidt basis is defined by recursively $\mathbf{b}_1^* = \mathbf{b}_1$ and $\mathbf{b}_i^* = \mathbf{b}_i - \sum_{j=1}^{i-1} \mu_{i,j}\mathbf{b}_j^*$ where $\mu_{i,j} = \langle \mathbf{b}_i, \mathbf{b}_j^* \rangle / \langle \mathbf{b}_j^*, \mathbf{b}_j^* \rangle$ for $i = 2, \ldots, n$. The new basis $\mathbf{b}_1^*, \ldots, \mathbf{b}_n^*$ that are orthogonal to each other spans the same space to the original basis:

$$\mathrm{span}(L) := \left\{\sum_{i=1}^n w_i \mathbf{b}_i : w_i \in \mathbb{R}\right\} = \left\{\sum_{i=1}^n x_i \mathbf{b}_i^* : x_i \in \mathbb{R}\right\}.$$

Thus, any lattice point $\mathbf{v} = \sum_{i=1}^n a_i \mathbf{b}_i$ can be presented by using Gram-Schmidt basis: $\mathbf{v} = \sum_{i=1}^n x_i \mathbf{b}_i^*$. For this presentation, the j-th projection is $\pi_j(\mathbf{v}) = \sum_{i=j}^n x_i \mathbf{b}_i^*$.

For a lattice L, denote $\lambda_1(L)$ the smallest nonzero norm of points in L, i.e., the length of shortest vector. The problem for searching $\mathbf{v} \in L$ so that $\|\mathbf{v}\| = \lambda_1(L)$ is called the shortest vector problem. The approximate Hermite shortest vector problem (HSVP_α) [9] is the problem of finding vector $\mathbf{v}$ shorter than $\alpha \cdot \det(L)^{1/n}$.

Consider a continuous set $S \subset \mathrm{span}(L)$ and denote its volume by $\mathrm{vol}(S)$. The Gaussian heuristic assumption claims that the number of lattice points in S is approximately given by $\mathrm{vol}(S)/\mathrm{vol}(L)$. In particular, we can see $\lambda_1(L)$ is close to $\ell = V_n^{-1/n}\det(L)^{1/n}$ so that $V_n(\ell) = \det(L)$. We denote this length $GH(L)$ and call the Gaussian heuristic length of L.

Root-Hermite factor and geometric series assumption: From the experimental observations by Gama, Nguyen and Stehlé [9], [15] for lattice reduction algorithms that works on any lattice dimension n, there exists a constant δ_0 so that the output of lattice reduction algorithm over random lattices satisfies $\|\mathbf{b}_1\| \approx \delta_0^n \det(L)^{1/n}$. This δ_0 is called *the root Hermite factor* of the algorithm. We call the basis is δ_0-reduced if $\|\mathbf{b}_1\| \leq \delta_0^n \det(L)^{1/n}$ holds, thus, it is a solution of $\mathrm{HSVP}_{\delta_0^n}$ problem.

Depending varieties of algorithms, the shapes of Gram-Schmidt lengths can be changed if they all achieve the same root Hermite factor δ_0. However, they are typically concave curves close to a line. Schnorr's geometric series assumption (GSA) [18] claims that $||\mathbf{b}_i^*||^2$ is approximated by $||\mathbf{b}_1||^2 r^{i-1}$ by a constant $r < 1$. Hence, each Gram-Schmidt lengths of a δ_0-reduced basis can be approximated by

$$||\mathbf{b}_i^*|| = r^{\frac{2i-1-n}{4}} \det(L)^{1/n} \text{ where } r = \delta_0^{\frac{-4n}{n-1}}. \tag{2}$$

Figure 1 shows the graph of $\log ||\mathbf{b}_i^*||$ between an output of a BKZ variant by Aono et al. [4] and a line from GSA of the same $\|\mathbf{b}_1\|$. We call the sequence $(||\mathbf{b}_1^*||, \ldots, ||\mathbf{b}_n^*||) = (r^{(1-n)/4}\det(L)^{1/n}, \ldots, r^{(n-1)/4}\det(L)^{1/n})$ the δ_0-*GSA basis*, which is an abnormal notation because it is not lattice basis.

This assumption was used to estimate the practical

hardness of lattice cryptography. However, for highly reduced lattice basis, such as BKZ-100, the last $||\mathbf{b}_i^*||$ does not form a line in general. Such phenomenon is justified by the Gaussian heuristic. Hence, it is not reasonable to estimate the *expected* complexity by using GSA. On the other hand, we will demonstrate it can be used for a lower bound in Section 4.

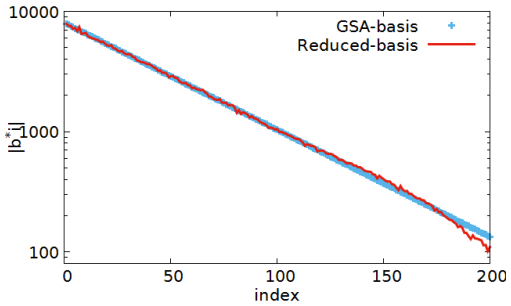

Fig. 1 Comparison of $||\mathbf{b}_i^*||$ between real reduced basis and equivalent δ_0-GSA basis

Pruned enumeration and its complexity: Let us fix a lattice basis B of rank m and its Gram-Schmidt lengths $||\mathbf{b}_1^*||, \ldots, ||\mathbf{b}_m^*||$, bounding radius c. Suppose we have a sequence of pruning coefficients $0 < R_1 \leq R_2 \leq \cdots \leq R_m = 1$. Define the set

$$C_k = \left\{ (x_1, \ldots, x_k) \in \mathbb{R}^k : \sum_{i=1}^{\ell} x_i^2 < R_\ell^2 \text{ for } \forall \ell \in [k] \right\}. \tag{3}$$

Then, the cost for pruned lattice vector enumeration [10] is given as follows under the Gaussian heuristic assumption.

$$\begin{aligned} N &= \frac{1}{2} \sum_{k=1}^{m} \frac{c^k \mathrm{vol} \left\{ \mathbf{x} \in \mathbb{R}^k : \sum_{i=1}^{\ell} x_i^2 < R_\ell^2 \text{ for } \forall\ \ell \in [k] \right\}}{\prod_{i=m-k+1}^{m} \|\mathbf{b}_i^*\|} \\ &= \frac{1}{2} \sum_{k=1}^{m} \frac{c^k \mathrm{vol}(C_k)}{\prod_{i=m-k+1}^{m} \|\mathbf{b}_i^*\|} \end{aligned} \tag{4}$$

Note that the factor $1/2$ is from the symmetry in the shortest vector computation, it is vanished if we consider the closest vector problem and its variants.

In [10], they assume the probability model for the shortest vector problem. Under the reasonable assumption, the probability to find a vector $\mathbf{v}$ by using searching radius $c = \|\mathbf{v}\|$ is given by

$$p := \Pr_{(x_1,\ldots,x_m) \leftarrow S^m \cdot \|\mathbf{v}\|} \left[\sum_{i=1}^{\ell} x_i^2 < \|\mathbf{v}\|^2 \cdot R_\ell^2 \text{ for } \forall\ \ell \in [m] \right]. \tag{5}$$

Hence, the best enumeration algorithm of success probability p_0 is given by best combination of $(R_1, \ldots, R_m)$ that minimizes (4) subject to that (5) is larger than p_0. However, it is not easy task to find optimal pruning coefficients. We will give a lower bound of enumeration cost without computing exact coefficients.

3. Bounding Cost for Lattice Vector Enumeration

3.1 Geometric Lemma and General Theory

In order to bound the cost, we need to bound each volume factor $\mathrm{vol}(C_k)$ in (4). The following geometric lemma have a crucial role.

Lemma1 Let C_k be a finite k-dimensional object, i.e., the k-dimensional volume $\mathrm{vol}(C_k) < \infty$. Let τ_k be the radius so that $V_k(\tau_k) = \mathrm{vol}(C_k)$. Fix a radial basis function $r(\mathbf{x}) = \phi(\|\mathbf{x}\|)$ where $\phi(\|\mathbf{x}\|)$ is a positive decreasing function on the radius: $\phi(x) \geq \phi(y) \geq 0$ for any $0 \leq x \leq y$. Then we have

$$\int_{C_k} r(\mathbf{x}) d\mathbf{x} \leq \int_{B_k(\tau_k)} r(\mathbf{x}) d\mathbf{x}. \tag{6}$$

Proof. By $V_k(\tau_k) = \mathrm{vol}(C_k)$, $V := \mathrm{vol}(C_k \setminus B_k(\tau_k)) = \mathrm{vol}(B_k(\tau_k) \setminus C_k)$ holds. Since $\phi(\|\mathbf{x}\|)$ is decreasing, we have the inequalities

$$\int_{C_k \setminus B_k(\tau_k)} r(\mathbf{x}) d\mathbf{x} \leq V \cdot \phi(\tau_k) \leq \int_{B_k(\tau_k) \setminus C_k} r(\mathbf{x}) d\mathbf{x}$$

Hence,

$$\begin{aligned} \int_{C_k} &= \int_{C_k \cap B_k(\tau_k)} + \int_{C_k \setminus B_k(\tau_k)} \\ &\leq \int_{C_k \cap B_k(\tau_k)} + \int_{B_k(\tau_k) \setminus C_k} = \int_{B_k(\tau_k)}. \end{aligned}$$

□

If the LHS of (6) is known value and the RHS is an easily invertible function $F(\tau_k)$ with respect to the radius, we have $\tau_k \geq F^{-1}(LHS)$ since F is always a strictly increasing function. Thus, it derives the lower bound.

$$\mathrm{vol}(C_k) = V_k(\tau_k) \geq V_k(F^{-1}(LHS)) \tag{7}$$

3.2 Application to Short Vector Search

We start our argument at the single usage of Gama et al.'s pruned enumeration [10]. Fixing the pruning coefficients $R_1, \ldots, R_m$, the intermediate searching areas C_k are fixed by (3). The probability (5) is bounded upper as

$$
\begin{aligned}
p = (5) \quad &\le \Pr_{\mathbf{x} \leftarrow S^m \cdot \|\mathbf{v}\|} \left[\sum_{i=1}^{\ell} x_i^2 < \|\mathbf{v}\|^2 \cdot R_\ell^2 \text{ for } \forall\ \ell \in [m-2] \right] \\
&\text{(by relaxed condition)} \\
&= \Pr_{\mathbf{x} \leftarrow B_{m-2}(1)} \left[\sum_{i=1}^{\ell} x_i^2 < R_\ell^2 \text{ for } \forall\ \ell \in [m-2] \right] \\
&= \frac{\mathrm{vol}(C_{m-2})}{V_{m-2}(1)}.
\end{aligned} \tag{8}
$$

For any $k \le m-2$,

$$
\begin{aligned}
\mathrm{vol}(C_{m-2}) &= \int_{C_k} \mathrm{vol}\{\mathbf{z} \in C_{m-2} : (z_1, \ldots, z_k) = \mathbf{x})\} d\mathbf{x} \\
&\le \int_{C_k} \mathrm{vol}\{\mathbf{z} \in B_{m-2}(1) : (z_1, \ldots, z_k) = \mathbf{x})\} d\mathbf{x}
\end{aligned}
$$

where the volumes are the $(m-2-k)$-dimensional volume defined on the coordinates $(z_{k+1}, \ldots, z_{m-2})$. The latter integrating function

$$
\begin{aligned}
r(\mathbf{x}) &= \mathrm{vol}\{\mathbf{z} \in B_{m-2}(1) : (z_1, \ldots, z_k) = \mathbf{x})\} \\
&= \begin{cases} B_{m-2-k}(\sqrt{1 - \|\mathbf{x}\|^2}) & (\text{if } \|\mathbf{x}\| \le 1) \\ 0 & \text{otherwise} \end{cases}
\end{aligned}
$$

satisfies the requirement of Lemma 1. Thus, we have

$$
\mathrm{vol}(C_{m-2}) \le \int_{B_k(\tau_k)} r(\mathbf{x}) d\mathbf{x} = V_{m-2}(1) \cdot I_{\tau_k^2}\left(\frac{k}{2}, \frac{m-k}{2}\right),
$$

and have the lower bound of the radius

$$
\tau_k \ge \sqrt{I^{-1}_{\mathrm{vol}(C_{m-2})/V_{m-2}(1)}\left(\frac{k}{2}, \frac{m-k}{2}\right)} \ge \sqrt{I^{-1}_{p}\left(\frac{k}{2}, \frac{m-k}{2}\right)}.
$$

Therefore, we obtain our lower bound for the enumeration of probability p and radius c:

$$
\frac{1}{2} \sum_{k=1}^{m} \frac{c^k V_k(1) \left[I_p^{-1}\left(\frac{k}{2}, \frac{m-k}{2}\right) \right]^{\frac{k}{2}}}{\prod_{i=m-k+1}^{m} \|\mathbf{b}_i^*\|}. \tag{9}
$$

Model to find multiple points: By a similar argument, the lower bound for another model can be derived. By Gaussian heuristic, the number of lattice points within the searching area is about $N = c^m \mathrm{vol}(C_m)/\det(L)$. Thus, we can also bound the complexity of the enumeration to find multiple points shorter than c by setting the condition $\mathrm{vol}(C_m) \ge N \det(L)/c^m$. By the same argument with above, we have

$$
\tau_k \ge \sqrt{I^{-1}_{\mathrm{vol}(C_m)/V_n(1)}\left(\frac{k}{2}, \frac{m-k}{2}\right)}
$$

and thus, our lower bound for the enumeration cost to find N vectors shorter than c is given as follows:

$$
\frac{1}{2} \sum_{k=1}^{m} \frac{c^k V_k(1) \left[I^{-1}_{\frac{N \det(L)}{V_n(c)}}\left(\frac{k}{2}, \frac{m+2-k}{2}\right) \right]^{k/2}}{\prod_{i=n-k+1}^{n} \|\mathbf{b}_i^*\|}. \tag{10}
$$

This is valid for the parameters satisfying $N \det(L) \le V_m(c)$.

3.3 Multiple Usage of Lattice Bases

Using our lower bound for single usage of enumeration algorithm, we can bound the cost of Gama-Nguyen-Regev's extreme pruning that uses multiple random bases. For a reasonably high success probability p, they run M trials of probabilistic enumeration with a low probability p/M with using randomized bases. Thus, the total cost is the sum of $(M-1)$ randomizations and M lattice reductions *, and M enumerations.

We estimate the lower bound cost by

$$
\begin{aligned}
TotalCost = \quad &(M-1) \cdot Cost(LatticeReduction) \\
&+ M \cdot Cost(Enumeration) \\
&> M \cdot Cost(Enumeration).
\end{aligned} \tag{11}
$$

The cost of M enumerations can be bounded lower for each situation.

Probability model: For the lower bound for Gama-Nguyen-Regev's probabilistic model, we can show the lower bound by using (1):

$$
\begin{aligned}
TotalCost &> \frac{M}{2} \sum_{k=1}^{m} \frac{V_k(c) \left[I^{-1}_{p/M}\left(\frac{k}{2}, \frac{m-k}{2}\right) \right]^{\frac{k}{2}}}{\prod_{i=m-k+1}^{m} \|\mathbf{b}_i^*\|} \\
&> \frac{p}{4} \sum_{k=1}^{m} \frac{V_k(c) \cdot k \cdot B\left(\frac{k}{2}, \frac{m-k}{2}\right)}{\prod_{i=m-k+1}^{m} \|\mathbf{b}_i^*\|}.
\end{aligned} \tag{12}
$$

Short vector search: For the target number N of lattice points that we want to find, if we use M randomized bases, at least N/M target number is necessary for each basis. It should be larger than N/M since duplication of found vectors. For these parameters, (11) is bounded lower by using (10), and by the inequality (1), we have

$$
\begin{aligned}
&M \cdot Cost(Enumeration) \\
&> \frac{M}{2} \sum_{k=1}^{m} \frac{c^k V_k(1) \left[I^{-1}_{\frac{N \det(L)}{M V_m(c)}}\left(\frac{k}{2}, \frac{m+2-k}{2}\right) \right]^{k/2}}{\prod_{i=m-k+1}^{m} \|\mathbf{b}_i^*\|} \\
&> \frac{N \det(L)}{4 V_m(c)} \sum_{k=1}^{m} \frac{V_k(c) \cdot k \cdot B\left(\frac{k}{2}, \frac{m+2-k}{2}\right)}{\prod_{i=m-k+1}^{m} \|\mathbf{b}_i^*\|} \\
&= \frac{N}{4} \sum_{k=1}^{m} \left[\prod_{i=1}^{m-k} \frac{\|\mathbf{b}_i^*\|}{c} \right] \cdot k \cdot \frac{V_k(1)}{V_m(1)} \cdot B\left(\frac{k}{2}, \frac{m+2-k}{2}\right) \\
&= \frac{N}{2} \sum_{k=1}^{m} \left[\prod_{i=1}^{m-k} \frac{\|\mathbf{b}_i^*\|}{c\sqrt{\pi}} \right] \cdot \Gamma\left(\frac{m+2-k}{2}\right) \\
&= \frac{N}{2} \sum_{k=1}^{m} \left[\prod_{i=1}^{k} \frac{\|\mathbf{b}_i^*\|}{c\sqrt{\pi}} \right] \cdot \Gamma\left(\frac{k}{2}\right).
\end{aligned} \tag{13}
$$

The last equation holds by swapping index $m-k$ by k.

* This lattice reduction level may be weaker than the original basis.

Remark that we do not need to consider the number of randomized bases in these cases. These inequalities mean that the affects of extreme pruning are limited by linear functions of probability or number of target points.

3.4 Computer Experiments

Systematic Upper bound: To show the sharpness of our lower bound, we give a method to compute upper bound cost. In contrast to the lower bound situation, a possible upper bound can be computed by setting feasible bounding coefficients. Thus, using a finite set of bounding coefficients whose probability is larger than p, an upper bound is given by the minimum cost among the coefficients. For this purpose, we define the pruning coefficients in dimension m parametrized by $\alpha \in \mathbb{R}$ and $j \in [m]$ by

$$R_i(\alpha, j) = \min((i/j)^\alpha, 1). \tag{14}$$

For given parameters $(||\mathbf{b}_1^*||, \ldots, ||\mathbf{b}_n^*||, c, p)$, and for each integer j, we can compute α so that the lower bound probability is p by the binary search. Then, compute the minimum of upper bound cost among all j, we can obtain an upper bound of enumeration cost of probability p. Here, we remark that for given pruning coefficients, there is polynomial time algorithms [10] to compute good lower and upper bounds for the cost and probability.

Comparison: Figure 2 shows the comparison among the systematic upper bound defined in this section, expected enumeration cost (4), lower cost bound for single usage of pruned enumeration (9), and the lower cost bound for extreme pruning technique (13). Here we used LLL reduced bases of random 100 and 160 dimensional lattices, and the radius is $c = GH(L)$. In 160 dimension, for $p < 10^{-4}$, we can see the gap between ENUMCost and GNR Lower is less than 10^6, and gap between Systematic Upper and GNR Lower is less than 10^{10}.

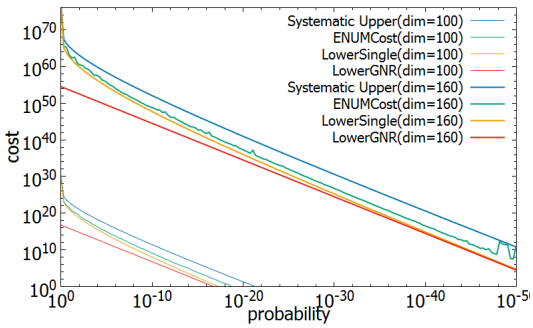

Fig. 2 Simulation of pruned enumcost and upper/lower bound for LLL bases of 100 and 160 dimensions

4. Bounding Cost of Lattice Problems

The cost of lattice problem is typically given by the following model. For a given lattice basis and target point, separate a considered algorithm into two parts: lattice reduction and lattice point search, the cost of attacker is defined by

$$Cost(Problem) = \min \frac{Cost(LatticeReduction) + Cost(PointSearch)}{\text{Success probability}}. \tag{15}$$

Here, the minimum is taken over all typical lattice reduction algorithms and the pruned lattice enumeration algorithm. Parameters in each step are optimized via suitable preliminary simulations.

Since we now have the lower bound for lattice point search, what we need to discuss is the lower bound for the lattice reduction part and the output Gram-Schmidt lengths of it. We divide the class of lattice reduction algorithm by the root Hermite factor δ_0. Then, for a fixed total success probability, the minimizing problem in (15) is

$$\min_{\delta_0} \min_{LR(\delta_0)} [CostLR(\delta_0) + CostEnum(LR(\delta_0))]. \tag{16}$$

Note that this is still a very abstract representation. $LR(\delta_0)$ is the set of all lattice reduction algorithm that achieve the root Hermite factor δ_0, and $CostEnum(LR(\delta_0))$ is the enumeration cost for the basis outputted by such algorithm. In this section, we give reasonable lower bound for the above two costs.

4.1 Enumeration Cost over a Reduced Basis

Fix the root Hermite factor δ_0. The cost we want to bound in this section is $CostEnum(LR(\delta_0))$, that is, the cost (4) for a given radius and success probability, and the Gram-Schmidt lengths of output of a lattice reduction algorithm; $\|\mathbf{b}_1\| = \delta_0 \det(L)^{1/m}$ is known but other projected lengths are unknown.

From our experiments, we observed the cost (4) for δ_0-GSA basis is typically lower than the cost for original basis in many situations. Figure 3 shows the cost comparison among the simulated cost for original basis and equivalent GSA basis, and the lower bound cost (9) with parameters $p = 10^{-3}$ and $c = \mathrm{GH}(L)$. From the observation, we claim the following assumption.

Assumption1 For reasonable success probability p and searching radius c, the cost (4) of an output basis of a typical lattice reduction algorithm, is larger

than the cost of δ_0-GSA basis where δ_0 is computed by $(\|\mathbf{b}_1\|/\det(L)^{1/m})^{1/m}$.

Note that we may be able to consider an artificial counterexample to break this assumption. For example, for an LLL-reduced basis, apply strong BKZ algorithm for its projected sublattice $\pi_2(\mathbf{b}_2), \ldots, \pi_2(\mathbf{b}_m)$.

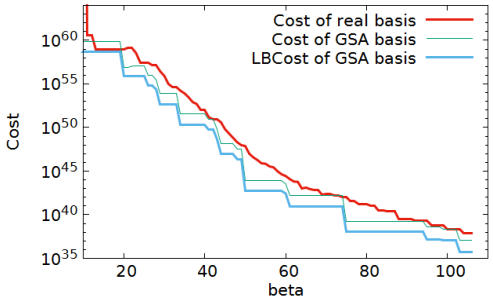

Fig. 3 Comparison of various simulated costs between real reduced basis and equivalent GSA basis. Parameters are $p = 10^{-3}$ and $c = \mathrm{GH}(L)$.

4.2 Bounding cost for lattice reduction

We give our lower bound for finding δ_0-reduced basis by a lattice reduction algorithm. For readability, we use $CostLR(m, \delta_0)$ to denote the cost for lattice reduction cost by giving lattice dimension m explicitly. This is the matter what we want to bound in this section.

Clearly, $CostLR(m, \delta_0) > CostLR(n, \delta_0)$ holds for $m > n$. However, there exists the lower bound on the dimension from the Gaussian heuristic, i.e., the dimension must satisfy $\delta_0^m > V_m^{-1/m}$. If m and δ_0 do not satisfy it, the cost bound is not valid since it is hard to exist a vector shorter than $\delta_0^m \det(L)^{1/m}$.

We fix m by the smallest integer satisfying this inequality, thus, $\delta_0^n < V_n^{-1/n}$ for $n < m$. Except for the LLL algorithm, all the known lattice reduction algorithms for finding a vector shorter than c must have at least one calling of a subroutine of lattice vector enumeration working over a first sublattice $B_n = (\mathbf{b}_1, \ldots, \mathbf{b}_n)$ with the radius $c = \delta_0^m \det(L)^{1/m}$ and target volume $\mathrm{vol}(C_n) \geq \det(B_n)/c^n$. We denote this cost by $CostENUM(n, \tilde{\ell_n})$ where $\tilde{\ell_n} := (\|\mathbf{b}_1\|, \ldots, \|\mathbf{b}_n\|)$. The lower bound cost is given by (13). Thus, writing $CostLR(m, \tilde{\ell_n})$ as the minimum cost to find an m-dimensional lattice basis B_m such that $\|\mathbf{b}_i^*\| = \ell_i$ for all $i \in [n]$, we have the relation

$$CostLR(m, \delta_0) = \min_{n, \tilde{\ell_n}} \left[CostLR(m, \tilde{\ell_n}) + CostENUM(n, \tilde{\ell_n}) \right]. \quad (17)$$

Here, $\tilde{\ell_n}$ is taken over all possible combination satisfying $\ell_1 \geq \delta_0^m \det(L)^{1/m}$.

Lemma2 In the cost model (17), the subdimension n must be m.

Proof. Suppose $n < m$ and the enumeration subroutine runs over the sublattice $B_n = (\mathbf{b}_1, \ldots, \mathbf{b}_n)$. By the Gaussian heuristic, the found vector is longer than $V_n^{-1/n} \det(B_n)^{1/n} \geq \delta_0^n \det(B_n)^{1/n}$. We show this is larger than $\delta_0^m \det(B_m)^{1/m} > V_m^{-1/m} \det(B_m)^{1/m}$, i.e., there is no vector in the searching range which makes a contradiction.

Let $D = (\mathbf{d}_m, \ldots, \mathbf{d}_1)$ and $(\mathbf{d}_m^\times, \ldots, \mathbf{d}_1^\times)$ be the dual basis of B_m and its Gram-Schmidt basis. The projective sublattice D_i is the lattice spanned by the projections of $\mathbf{d}_i, \ldots, \mathbf{d}_1$ onto $\mathbf{d}_m, \ldots, \mathbf{d}_{i+1}$. By the Gaussian heuristic on the projective sublattice of D, we have

$$\|\mathbf{d}_i^\times\| \geq GH(D_i) = V_i^{-1/i} \prod_{j=1}^{i} \|\mathbf{d}_i^\times\|^{1/i} = \delta_0^i \prod_{j=1}^{i} \|\mathbf{d}_i^\times\|^{1/i}.$$

Using the well known relation $\|\mathbf{b}_i^*\| = 1/\|\mathbf{d}_i^\times\|$, we have $\prod_{j=1}^{i} \|\mathbf{b}_j^*\|^{1/i} > \delta_0^i \|\mathbf{b}_i^*\|$ which derives

$$\prod_{j=1}^{i} \|\mathbf{b}_j^*\|^{1/i} > \delta_0^{\frac{i+1}{i}} \prod_{j=1}^{i+1} \|\mathbf{b}_j^*\|^{1/(i+1)}.$$

Thus,

$$\delta_0^n \det(B_n)^{1/n} > \delta_0^{n + \frac{n+1}{n} + \cdots + \frac{m}{m-1}} \det(B_m)^{1/m} > \delta_0^m \det(B_m)^{1/m}.$$

Therefore, the sublattice B_n does not have a vector shorter than $\delta_0^m \det(B_m)^{1/m}$ if $n < m$. □

Neglecting cost for lattice reduction in the cost (17), we have $CostLR(m, \delta_0) > CostENUM(m, \tilde{\ell_m})$ where $\tilde{\ell_m}$ is from a reduced basis so that $\ell_1 \geq c$. Using Assumption 1, it is bounded lower by the δ_0-GSA basis and also bounded by (13). In conclusion, our lower bound for lattice reduction to find a short vector is

$$CostLR(m, \delta_0) > \frac{1}{2} \sum_{k=1}^{m} r^{\frac{k(k-1)}{4}} \cdot \pi^{-k/2} \cdot \Gamma\left(\frac{k}{2}\right) \quad (18)$$

for the smallest integer m such that $\delta_0^m > V_m^{-1/m}$.

4.3 Comparison with Previous Models

In many existing works, they have given models of the relation between computing time and achieved root Hermite factor δ_0. To compare our lower bound with them, we give a short survey.

Lindner-Peikert [12] estimated $\log_2(t_{BKZ}[sec]) = \frac{1.8}{\log_2(\delta)} - 110$ from their experiments using NTL-BKZ for q-ary lattices derived from random LWE instances. They claimed it as a practical lower bound line from their curve fitting.

Albrecht et al. [2] estimated $\log_2(t_{BKZ}[sec]) = \frac{0.009}{\log_2(\delta)^2} - 27$ that is an extrapolation of the points from BKZ 2.0 simulating in [13] whose origin is proposed in [6]. The time is expectation from the simulator.

Albrecht et al. [1], [3] proposed $\log_2(t_{BKZ}[sec]) = \Theta\left(\frac{\log(1/\log\delta)}{\log\delta}\right)$ under the assumption that we have a β-dimensional SVP oracle that works in time $2^{\Theta(\beta)}$.

These estimations and our lower bound is summarized are Figure 4.

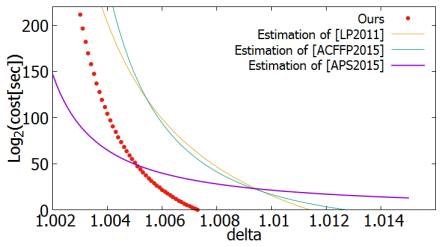

Fig. 4 Comparison among several models to achieve the root Hermite factor δ_0. Our estimation is (18) divided by 2^{25} to convert number of nodes to seconds; [LP2011] is [12]; [ACFFP2015] is [2]; [APS2015] is [3] with the constant $c = 0.05$.

4.4 Estimating the Approximate-SVP

We give a simple example to demonstrate our theory. To solve an α-approximate SVP, it needs to find a vector shorter than about $\alpha GH(L) = (\alpha^{1/m} V_m^{-1/m^2})^m \det(L)^{1/m}$, the root Hermite factor must be smaller than $\alpha^{1/m} V_m^{-1/m^2}$. Figure 5 shows the corresponding lower cost bound (18) for $\alpha = 1, 1.05, 1.5$ and dimensions. Remark that $\alpha = 1.05$ corresponds a rough estimation for TU Darmstadt SVP Challenge [17].

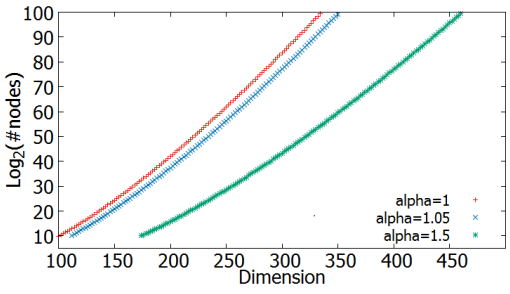

Fig. 5 Lower bound for solving SVP Challenge and comparison to current records

References

[1] M. R. Albrecht. On dual lattice attacks against small-secret LWE and parameter choices in helib and SEAL. In *Advances in Cryptology - EUROCRYPT 2017 - 36th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Paris, France, April 30 - May 4, 2017, Proceedings, Part II*, pages 103–129, 2017.

[2] M. R. Albrecht, C. Cid, J. Faugère, R. Fitzpatrick, and L. Perret. On the complexity of the BKW algorithm on LWE. *Des. Codes Cryptography*, 74(2):325–354, 2015.

[3] M. R. Albrecht, R. Player, and S. Scott. On the concrete hardness of learning with errors. Cryptology ePrint Archive, Report 2015/046, 2015.

[4] Y. Aono, Y. Wang, T. Hayashi, and T. Takagi. Improved progressive BKZ algorithms and their precise cost estimation by sharp simulator. *IACR Cryptology ePrint Archive*, 2016:146, 2016.

[5] Y. Chen. *Réduction de réseau et sécurité concrète du chiffrement complètement homomorphe*. PhD thesis, 2013.

[6] Y. Chen and P. Q. Nguyen. BKZ 2.0: Better lattice security estimates. In D. H. Lee and X. Wang, editors, *ASIACRYPT 2011*, volume 7073 of *Lecture Notes in Computer Science*, pages 1–20. Springer, 2011.

[7] I. Chillotti, N. Gama, M. Georgieva, and M. Izabachène. Faster fully homomorphic encryption: Bootstrapping in less than 0.1 seconds. In *Advances in Cryptology - ASIACRYPT 2016 - 22nd International Conference on the Theory and Application of Cryptology and Information Security, Hanoi, Vietnam, December 4-8, 2016, Proceedings, Part I*, pages 3–33, 2016.

[8] T. F. development team. fplll, a lattice reduction library. Available at https://github.com/fplll/fplll, 2016.

[9] N. Gama and P. Q. Nguyen. *Predicting Lattice Reduction*, pages 31–51. Springer Berlin Heidelberg, Berlin, Heidelberg, 2008.

[10] N. Gama, P. Q. Nguyen, and O. Regev. Lattice enumeration using extreme pruning. In H. Gilbert, editor, *EUROCRYPT 2010*, volume 6110 of *Lecture Notes in Computer Science*, pages 257–278. Springer, 2010.

[11] P.-C. Kuo, M. Schneider, . Dagdelen, J. Reichelt, J. Buchmann, C.-M. Cheng, and B.-Y. Yang. Extreme enumeration on gpu and in clouds: How many dollars you need to break svp challenges. In *Proceedings of the 13th International Conference on Cryptographic Hardware and Embedded Systems*, CHES'11, pages 176–191, Berlin, Heidelberg, 2011. Springer-Verlag.

[12] R. Lindner and C. Peikert. Better key sizes (and attacks) for LWE-based encryption. In A. Kiayias, editor, *CT-RSA*, volume 6558 of *Lecture Notes in Computer Science*, pages 319–339. Springer, 2011.

[13] M. Liu and P. Q. Nguyen. Solving BDD by enumeration: An update. In E. Dawson, editor, *CT-RSA*, volume 7779 of *Lecture Notes in Computer Science*, pages 293–309. Springer, 2013.

[14] D. Micciancio and M. Walter. Practical, predictable lattice basis reduction. In *Proceedings, Part I, of the 35th Annual International Conference on Advances in Cryptology — EUROCRYPT 2016 - Volume 9665*, pages 820–849, New York, NY, USA, 2016. Springer-Verlag New York, Inc.

[15] P. Q. Nguyen and D. Stehlé. *LLL on the Average*, pages 238–256. Springer Berlin Heidelberg, Berlin, Heidelberg, 2006.

[16] P. Q. Nguyen and B. Valle. *The LLL Algorithm: Survey and Applications*. Springer Publishing Company, Incorporated, 1st edition, 2009.

[17] M. Schneider and N. Gama. SVP challenge. Available at `http://www.latticechallenge.org/svp-challenge/`.

[18] C.-P. Schnorr. Lattice reduction by random sampling and birthday methods. In H. Alt and M. Habib, editors, *STACS 2003*, volume 2607 of *Lecture Notes in Computer Science*, pages 145–156. Springer, 2003.