A Novel Lattice Reduction Algorithm

Dipayan Das[1] and Vishal Saraswat[2]

[1]*Department of Mathematics, National Institute of Technology (NIT), Durgapur, India*

[2]*Department of Computer Science and Engineering, Indian Institute of Technology (IIT), Jammu, India*

Keywords: Lattice Based Crypto, CVP, SVP, Lattice Reduction.

Abstract: The quantum threats have made the traditional number theoretic cryptography weak. Lattice based cryptographic constructions are now considered as an alternative of the number theoretic cryptography which resists the quantum threats. The cryptographic hardness of the lattice based constructions mainly lies on the difficulty of solving two problems, namely, shortest vector problem (SVP) and closest vector problem (CVP). Solving these problems become "somewhat" easier if the lattice basis is almost orthogonal. Given any basis, finding an almost orthogonal basis is termed as lattice basis reduction (or simply lattice reduction). The SVP has been shown to be reducible to the CVP but the other way is still an open problem. In this paper, we work towards proving the equivalence of the CVP and SVP and provide a history of the progress made in this direction. We do a brief review of the existing lattice reduction algorithms and present a new lattice basis reduction algorithm similar to the well-studied Korkine-Zolotareff (KZ) reduction which is used frequently for decoding lattices. The proposed algorithm is very simple — it calls the shortest vector oracle for $n-1$ times and outputs an almost orthogonal lattice basis with running time $O(n^3)$, n being the rank of the lattice.

1 INTRODUCTION

The two main hard problems of interest in the lattices are SVP and CVP. Many cryptographic schemes (Goldreich et al., 1997; Ajtai and Dwork, 1997; Hoffstein et al., 1998) have been developed based on the hardness of either of the two problems or some variants of it.

Both CVP and SVP were shown to be NP hard (van Emde Boas, 1981; Khot, 2005). But, CVP is considered to be the hardest of all the lattice problems. Let us discuss it briefly. In (Goldreich et al., 1999a; Micciancio, 2008), it is shown that an oracle solving CVP can be used to solve SVP and SIVP respectively. In the other direction, solving CVP with an oracle that solves the other hard lattice problem is not known properly. For example, it is still unknown whether SVP oracle can be used to solve CVP. Though in (Kannan, 1987a), it has been shown that if we have an oracle that solves SVP exactly, it can be used to solve CVP with an approximation factor of $O(\sqrt{n})$ using homogenization technique in a higher dimension. Though in (Micciancio and Goldwasser, 2012), it is suggested that approximating CVP within the same factor as of (Kannan, 1987a) can be achieved making $O(n \log n)$ calls to an oracle which solves an approximate solution of SVP, approximation factor less than $\sqrt{2}$.

The best known deterministic algorithm to solve CVP in a general lattice is given in (Micciancio and Voulgaris, 2013) which takes $\tilde{O}(2^{2n})$ operations and $\tilde{O}(2^n)$ space. The algorithm uses a description of the Voronoi cell of the lattice as a pre-processing function before the target vector is revealed. Prior to (Micciancio, 2001), the best deterministic algorithm to solve CVP was due to Kannan (Kannan, 1987b) which takes $n^{O(n)}$ running time. In (Hanrot and Stehlé, 2007), the Kannan method was improved, which solves CVP in running time $n^{0.5n}$.

There are randomized algorithms which perform better than the deterministic ones. For example in (Ajtai et al., 2001), a sieve algorithm was introduced known as "AKS Sieve", which solves SVP in running time $2^{O(n)}$. The AKS method is based on an improved sampling method that generates short vectors from the given lattice. In (Ajtai et al., 2002), the AKS Sieve was reformulated to solve CVP with an approximation factor $1+\varepsilon$ in running time $2^{O(1+\varepsilon^{-1})}$. The running time was improved using a variant of AKS method in (Blömer and Naewe, 2007) and (Arvind and Joglekar, 2008) keeping the approximation factor same. Another randomized technique used to solve the hard problems of the lattice is Sampling

Das, D. and Saraswat, V.
A Novel Lattice Reduction Algorithm.
DOI: 10.5220/0006862104960501
In *Proceedings of the 15th International Joint Conference on e-Business and Telecommunications (ICETE 2018) - Volume 2: SECRYPT*, pages 496-501
ISBN: 978-989-758-319-3

Gaussian distribution with proper parameter. Agarwal et al. have given a randomized algorithm that solves an exact version of SVP using discrete Gaussian sampling in $2^{n+O(n)}$ time and space (Aggarwal et al., 2015a). In (Aggarwal et al., 2015b), the similar type of sampling technique is used to solve exact CVP. Though the work is a bit complicated and uses almost all previously known methods like basis reduction, Voronoi description, and sieving along with the sampling of shifted discrete Gaussian distribution with the proper parameter to solve CVP exactly. A nice survey of solving CVP of the known methods are given in (Agrell et al., 2002).

1.1 Basis Reduction and Related Work

A lattice has an infinite number of basis of rank greater than 1. If $B, \tilde{B}$ are the two basis that generates the same lattice L, then it can be shown that $\tilde{B} = UB$ for some uni-modular matrix U, that is integer matrix with absolute determinant value 1.

Given any lattice, finding an orthogonal basis (if exists) is hard. Even, deciding whether there exist an orthogonal basis is not known for general lattices. But in "lattices with symmetry", which is a special type of lattice, Gentry-Szydlo algorithm given in (Gentry and Szydlo, 2002) can determine this problem with high probability. It gives an efficient method to achieve an orthogonal basis (if there is) for an ideal lattice.

Basis reduction algorithms have many applications. It is important in areas like communications (Agrell et al., 2002; Wubben et al., 2011), combinatorial optimization (Eisenbrand, 2010), cryptography (Hanrot et al., 2011a), number theory (Goldreich et al., 1999b), etc. A nice survey in this regard is given in (Wubben et al., 2011).

Hermite (1850) has given the idea of basis reduction but unfortunately, he doesn't provide any such algorithm for basis reduction. According to Hermite a basis $B = \{b_1, b_2, \ldots, b_n\}$ is reduced if $\|b_i\| \leq \|b'_i\|$ for all basis B' of the same lattice and $\|b_j\| = \|b'_j\|$, $j = 1, 2, \ldots, i-1$ (Hermite, 1850). Later in 1905, Minkowski (Minkowski, 1905) has defined criteria of basis reduction known as Minkowski reduction which is very similar to that of Hermite. He made a slight change of the second criteria of Hermite. Instead of $\|b_j\| = \|b'_j\|$, he has directly taken $b_j = b'_j$. Two types of basis reduction that have gained fame and popularity is LLL reduction and KZ reduction. Let us define these reduced basis. Before that, some special basis is considered in which the KZ reduction and LLL reduction is well understood.

It can be shown that every lattice has a basis which can be represented as an upper triangular matrix (Agrell et al., 2002). We consider such a basis. The basis B is called KZ reduced if b_1 is a shortest vector of the lattice and the modulus of non-zero non-diagonal elements of each column is less than or equal to the modulus of half of diagonal elements of the corresponding column. That is $|b_{ki}| \leq \frac{1}{2}|b_{ii}|$ for each $i = 1, 2, \ldots, n$ and $k = 1, 2, \ldots, n$. Geometrically the last criteria says that the angle made by any two basis vectors is at least 60 degree. That is any two basis vectors are highly orthogonal. Let us illustrate this in dimension 2. Let B be the KZ reduced basis in dimension and rank 2. Let

$$B = \begin{pmatrix} b_1 \\ b_2 \end{pmatrix} = \begin{pmatrix} b_{11} & 0 \\ b_{21} & b_{22} \end{pmatrix}$$

and $\mid b_{21} \mid \leq \frac{1}{2} \mid b_{11} \mid$. This implies that $\|b_2\|\cos\theta \leq \frac{1}{2}\|b_1\|$, where θ is the angle between b_1 and b_2, that is,

$$\cos\theta \leq \frac{1}{2}\|b_1\|/\|b_2\| \leq \frac{1}{2}$$

The last inequality is due to the fact that B is KZ reduced and so b_1 is the shortest vector in the lattice. Similarly, for any dimension and rank, we can redefine the second criteria of KZ reduced basis as the criteria that any two bases element is highly orthogonal.

Like KZ reduced basis, there exist another famous basis reduction criteria known as LLL reduction. The only difference LLL reduction criteria have that instead of b_1 to be the shortest lattice vector, it has the criteria that $\|b_1\| \leq \frac{2}{\sqrt{3}}\|b_2\|$. The second condition of LLL reduction criteria is same as that of KZ.

It is easy to see that any basis which is KZ reduced is also LLL reduced. The reason for the superiority of the LLL reduction criteria is that there exists an efficient tool to achieve a LLL reduced basis (Lenstra et al., 1982). Later, there have been algorithms for reduction based on both KZ and LLL (Schnorr, 1987). It is known as the BKZ reduction algorithm.

The current best lattice basis reductions (theoretically and practically) can be classified by a pair algorithms. Firstly, the rational BKZ algorithm (Schnorr and Euchner, 1994; Schnorr, 1987), in its updated BKZ 2.0 form (Chen and Nguyen, 2011) doing some modifications like repetitive preprocessing and rapid aborting strategies (Gama et al., 2010; Hanrot et al., 2011b). Secondly, the Slide reduction algorithm proposed (Gama and Nguyen, 2008b), a novel generalization of LLL (Lenstra et al., 1982; Nguyen, 2009) which almost approximates SVP within small factors.

The two algorithms call the SVP oracle for dimension of lower order sublattices which are characterized by a bound k (termed as the "block size") on the lattice dimensions. The algorithm of Slide reduction has qualities like it makes only a polynomial number of calls to the shortest vector oracle, all the shortest vector calls are applied on the sublattices projected in the dimension k, and it obtains the current best worst-case upper bound on the length of the outputted shortest vector: $\gamma_k^{(n-1)/2(k-1)} \det(L)^{1/n}$, the constant $\gamma_k = \Theta(k)$ is known as the Hermite constant. Lamentably, it has been stated in (Gama and Nguyen, 2008b; Gama and Nguyen, 2008a) that the experimental results of the algorithm which follows Slide reduction perform better than BKZ, which outputs shorter vectors for some standard block size.

On the other hand, the BKZ algorithm has its own flaws too. There is no guarantee for the termination of the algorithm even after a polynomial number of shortest vector oracle call, and its experimental time complexity has appeared in (Gama and Nguyen, 2008a) and is reported to increase super-polynomially in the dimension of the lattice with fixed small block size.

Micciancio et al. have given new methods that can be changed to define better reduction methods in (Micciancio and Walter, 2016). They have analyzed their theoretical performance with block size comparatively high.

The LLL algorithm can be processed very quickly under a few conditions and has some characteristics which are studied recently in (Chang et al., 2013; Wen et al., 2016; Wen and Chang, 2017b). In some applications of communication theory, one needs to get solutions of a sequence of CVP's, where the target vectors are different on the same lattice. Here, instead of adapting the LLL algorithm, one usually adapts the KZ reduction to do the pre-processing step, as it becomes more efficient in practice.

There are many variations of KZ basis reduction available today. They are given in (Agrell et al., 2002; Schnorr, 1987; Wen and Chang, 2017a).

1.2 Our Contribution

We have proposed an algorithm for generating a new basis for a given lattice, which is very orthogonal. It is kind of similar to KZ reduction (Korkine and Zolotareff, 1873), but not exactly the same. One holding the SVP oracle first finds out the shortest vector (up to sign) of the lattice corresponding to the given basis and then transforms the original basis into a new basis where the first row is the shortest vector. In practice, this can be implemented by calling the LLL algorithm. But, while the LLL reduction algorithm executes each iteration in such a way that the final output is a basis in which the basis vectors are at most 30 degrees from being orthogonal, our proposed reduction algorithm tries to maximize the orthogonality between the resulting basis vectors and simultaneously ensuring that the basis vectors are shortest possible by calling the SVP oracle iteratively.

We call the SVP oracle and size reduce the basis formed by the remaining basis vectors (other than the shortest one) and continue iteratively as follows. We first reduce the $n-1$ basis vectors by a suitable integer combination of the shortest vector (size reduce the $n-1$ basis vectors similar to what is done in LLL or KZ), such that the resulting $n-1$ basis vectors are short with respect to the shortest vector. Details of the size reduction algorithms can be found in (Laarhoven et al., 2012). Then the SVP oracle is called on the "reduced" $n-1$ basis vectors. We iterate the reduction in this fashion until we obtain last shortest vector in one dimensional sublattice.

Let us describe briefly about the difference between the KZ reduction and the algorithm proposed by us. For KZ reduction, in the second step right after we use the SVP oracle to find out the shortest vector, we need to project the remaining $n-1$ basis vectors to the subspace that is orthogonal to the shortest vector, and then call the SVP oracle on the projected $n-1$ vectors. The output of the SVP oracle will help to decide the second KZ reduced basis vector, and the KZ reduction moves on in this style.

2 PRELIMINARIES

We use $\mathbb{R}$, $\mathbb{Z}$ to define the sets of Reals and Integers respectively. Let $\mathbb{R}^m$ be the Euclidean vector space of dimension m, and $\|\cdot\|$ is the Euclidean norm ℓ_2. Let $B = \{b_1, b_2, \ldots, b_k\}$, $1 \le k < m$, be vectors of $\mathbb{R}^m$ which are linearly independent. We define a lattice $L \subset \mathbb{R}^m$ generated by an arbitrary basis B as

$$L(B) \text{ or } L\{B\} = \left\{ \sum_{i=1}^{k} c_i b_i \;\middle|\; c_i \in \mathbb{Z} \right\}.$$

We define the dimension of the lattice to be m and the rank to be k. For the sake of simplicity, we are considering the integral lattice that is lattice which is generated by integer basis. A lattice L is a discrete additive subgroup of $\mathbb{R}^m$. We can represent a basis of a lattice of rank k and dimension m as a $k \times m$ matrix B where every row b_i is the basis vector of the lattice. If $m = k$ we call it full rank lattice.

Definition 1 (Shortest vector problem (SVP))**.** Given a lattice $L = L(B)$, SVP is to find a non-zero vector $x \in L$ such that $\|x\| \le \|v\|$ for all $v \in L \setminus \{0\}$.

Definition 2 (Closest vector problem (CVP))**.** Given a lattice $L = L(B)$ and a target vector $t \in \mathbb{R}^m$ (not necessarily in the lattice L), CVP is to find a vector $x \in L$ such that $\|x - t\| \le \|v - t\|$ for all $v \in L$.

3 SOME IMPORTANT RESULTS

Here we provide some important results that are related to our basis reduction algorithm.

Proposition 1. *Any pair of bases of a lattice L are connected by a uni-modular matrix.*

Proposition 2. *Let $B = \{b_1, b_2, \dots, b_n\}$ be a basis of the lattice L and let v be a shortest vector in L. Then v can be written uniquely as $v = \sum_{i=1}^{n} \alpha_i b_i$ where* $\gcd(\alpha_1, \alpha_2, \dots, \alpha_n) = 1$.

Proof. Let $d = \gcd(\alpha_1, \alpha_2, \dots, \alpha_n)$. Then, for $i = 1, \dots, n$, α_i/d is an integer so that $v' = \sum_{i=1}^{n} (\alpha_i/d) b_i$ is another vector in the lattice L and $\|v'\| = \|v\|/d$. Since v is a shortest vector in L, $d = 1$. □

Proposition 3 ((Newman, 1972; Magliveras et al., 2008))**.** *Let $\alpha_1, \alpha_2, \dots, \alpha_n \in \mathbb{Z}$ be such that* $\gcd(\alpha_1, \alpha_2, \dots, \alpha_n) = d_n$ *. Then there exists an integer matrix $\mathcal{U}$ having the initial row as $(\alpha_1, \alpha_2, \dots, \alpha_n)$ and determinant value d_n.*

Proof. Let $\alpha_1, \alpha_2, \dots, \alpha_n \in \mathbb{Z}$ be such that all α_i's are not zero. By doing permutation (if required), we can have $\alpha_1 \neq 0$. Let us define $d_1 = \alpha_1, d_i = \gcd(\alpha_1, \alpha_2, \dots, \alpha_i)(2 \leq i \leq n), d = d_n$. All d_i's are well defined (because we have considered $\alpha_1 \neq 0$) and $d_i = \gcd(d_{i-1}, \alpha_i)$ $(2 \leq i \leq \ n)$. By Euclidean algorithm, we can find t_i, s_i efficiently such that $d_i = t_{i-1} d_{i-1} + s_{i-1} \alpha_i$ where $|s_{i-1}| \leq d_{i-1}$.

We can construct a matrix $\mathcal{U}$ such that

$$\mathcal{U} := (\mathcal{U}_{i,j}) := \begin{pmatrix} \alpha_1 & \alpha_2 & \alpha_3 & \dots & \alpha_n \\ -s_1 & t_1 & 0 & \dots & 0 \\ -\frac{\alpha_1 s_2}{d_2} & -\frac{\alpha_2 s_2}{d_2} & t_2 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ -\frac{\alpha_1 s_{n-1}}{d_{n-1}} & -\frac{\alpha_2 s_{n-1}}{d_{n-1}} & -\frac{\alpha_3 s_{n-1}}{d_{n-1}} & \dots & t_{n-1} \end{pmatrix}$$

Thus, $\mathcal{U}$ is an integral matrix and using simple induction it can be shown that $\det(\mathcal{U}) = d_n = d$. □

Proposition 4. *Let $L = L(B)$ where $B = \{b_1, b_2, \dots, b_n\}$ is a basis of L. Let $v = \sum_{i=1}^{n} \alpha_i b_i$ be a shortest vector in $L(B)$. Then v and $n-1$ vectors in L can be extended to form a basis of L.*

Proof. We can create $n-1$ vectors as $\sum_{j=1}^{n} \mathcal{U}_{i,j} b_j$, $2 \leq i \leq n$, where

$$\begin{aligned} \mathcal{U}_{i,j} &= -\frac{\alpha_j s_{i-1}}{d_{i-1}} && 1 \leq j \leq i-1, 2 \leq i \leq n, \\ \mathcal{U}_{i,i} &= t_{i-1} && 2 \leq i \leq n, \\ \mathcal{U}_{i,j} &= 0 && i+1 \leq j \leq n, 2 \leq i \leq n \end{aligned}$$

The result now follows from the Propositions 2 and 3. □

4 BASIS REDUCTION ALGORITHM

Here we propose the basis reduction algorithm (Figure 1) and provide the main result of this paper in the Section 4.1. We discuss the time complexity of the algorithm in Subsection 4.2.

Algorithm for basis reduction

```
Input ← $L(B)$
for($i = 1; i < n; i++$)
    Run the SVP oracle $O$ on $L(B)$
    to get the shortest lattice vector $\sigma_{i-1}$
    in rank $n - i + 1$
    $L(B) \leftarrow L(B/\sigma_{i-1})$
    $L(B) \leftarrow$ size reduce $L(B)$
Output ← $\tilde{B} = \{\sigma_0, \sigma_1, \dots, \sigma_{n-1}\}$
```

Figure 1: Our Proposed lattice basis reduction.

4.1 Analysis of the Proposed Algorithm

Theorem 3. *Let $L = L\{B\}$ be a given lattice. Suppose we have an oracle O that solves* SVP *in any rank of a lattice, then we have an efficient algorithm to reduce it to a basis $\tilde{B}$ that is the highly orthogonal one using $n-1$ oracle calls such that $L = L\{B\} = L\{\tilde{B}\}$.*

Proof. Let B_0 be a given lattice basis which generates $L = L\{B_0\}$. We run the oracle O on $L\{B_0\}$ and get the shortest vector as σ_0. By Proposition 4, we can generate a new basis $B'_0 = \{\sigma_0, b_1, b_2, \dots, b_{n-1}\}$ such that $L = L\{B\} = L\{B'_0\}$.

Let $L\{B_1\} = L\{B'_0 \setminus \{\sigma_0\}\}$ be a new lattice generated by eliminating all the integer linear combinations of σ_0. The rank of the lattice $L\{B_1\}$ is $n-1$ with basis $B_1 = \{b_1, b_2, \dots, b_{n-1}\}$. We size-reduce the elements of B_1 which can be done efficiently such that B_1 has short vectors with respect to σ_0. We now run the oracle O on $L\{B_1\}$ to get the shortest vector σ_1 in rank $n-1$. By Proposition 4, we can create a new basis $B'_1 = \{\sigma_1, b'_2, \dots, b'_{n-1}\}$ such that $L = L\{B_1\} = L\{B'_1\}$.

Let $L\{B_2\} = L\{B'_1 \setminus \{\sigma_1\}\}$ be a new lattice generated by eliminating all the integer linear combinations of σ_1. The rank of the lattice $L\{B_2\}$ is $n-2$ with basis $B_2 = \{b'_2, b'_3, \dots, b'_{n-1}\}$. We size-reduce the elements of B_2 such that B_2 has short vectors with respect to σ_1. We now run the oracle O on $L\{B_2\}$ to get the shortest vector σ_2 in rank $n-2$.

Doing accordingly, let $L\{B_{n-1}\} = L\{B'_{n-2} \setminus \{\sigma_{n-2}\}\}$, where σ_{n-2} is the oracle output of shortest vector in rank 2, B'_{n-2} is the basis of rank 2 including the shortest vector.

After reduction, we now run the oracle O on $L\{B_{n-1}\}$ to get σ_{n-1}, the shortest vector in rank 1.

Let $\tilde{B} = \{\sigma_0, \sigma_1, \ldots, \sigma_{n-1}\}$. We can see that $\tilde{B}$ and B generate the same lattice that is $L\{B\} = L\{\tilde{B}\}$. Since σ_i's are the shortest vector of rank $n-i$, so we can conclude that $\tilde{B}$ is the "good" basis, that is vectors are sufficiently orthogonal to one another. □

4.2 Running Time Analysis

Let $B = \{b_1, b_2, \ldots, b_n\}$ is a basis of the lattice L and let $v = \sum_{i=1}^{n} \alpha_i b_i$ be a shortest vector in L. Let α_0 and α_1 be the two maximum absolute values of all $|\alpha_i|$. Then, the worst-case time complexity of computing basis with the shortest vector is $O(n^2 \log \alpha_0 \log \alpha_1)$.

For one oracle call, constructing the new basis, the number of bit operations needed is $O(n^2 \log \alpha_0 \log \alpha_1)$. So, for $n-1$ oracle calls, worst-case running time is $O(n^3 \log \alpha_0 \log \alpha_1)$ bit operations which is the required worst-case time complexity for our new basis construction as in Theorem 3.

5 DISCUSSIONS AND OPEN PROBLEMS

We have described a polynomial time algorithm for lattice basis reduction by calling the shortest vector oracle using simple geometry and linear algebra. In general, the best known algorithms take an exponential time to find the shortest vector but in some restricted lattices like root lattices, SVP can be found in polynomial time. So in lattices with special structures, our algorithm can be a practical use which is yet to be analyzed.

In Section 1.1, we have stated that in general lattices deciding whether a lattice has an orthogonal basis is hard. Nothing much is known in this regard. Can we construct some algorithm that will decide whether the lattice has an orthogonal basis using an SVP oracle? This question is yet to be answered. Can we make some tweak in our algorithm so that it can answer the above question? Then it will imply that solving the shortest vector problem is as hard as deciding whether a lattice has an orthogonal basis.

ACKNOWLEDGEMENTS

We thank the anonymous reviewers for the constructive and helpful comments. Part of the work was carried out while visiting the R.C.Bose Centre for Cryptology and Security, Indian Statistical Institute, Kolkata. We are thankful to Kajla Basu for her support.

REFERENCES

Aggarwal, D., Dadush, D., Regev, O., and Stephens-Davidowitz, N. (2015a). Solving the shortest vector problem in 2 n time using discrete gaussian sampling. In *Proceedings of the forty-seventh annual ACM symposium on Theory of computing*, pages 733–742. ACM.

Aggarwal, D., Dadush, D., and Stephens-Davidowitz, N. (2015b). Solving the closest vector problem in 2ˆ n time–the discrete gaussian strikes again! In *Foundations of Computer Science (FOCS), 2015 IEEE 56th Annual Symposium on*, pages 563–582. IEEE.

Agrell, E., Eriksson, T., Vardy, A., and Zeger, K. (2002). Closest point search in lattices. *IEEE transactions on information theory*, 48(8):2201–2214.

Ajtai, M. and Dwork, C. (1997). A public-key cryptosystem with worst-case/average-case equivalence. In *Proceedings of the twenty-ninth annual ACM symposium on Theory of computing*, pages 284–293. ACM.

Ajtai, M., Kumar, R., and Sivakumar, D. (2001). A sieve algorithm for the shortest lattice vector problem. In *Proceedings of the thirty-third annual ACM symposium on Theory of computing*, pages 601–610. ACM.

Ajtai, M., Kumar, R., and Sivakumar, D. (2002). Sampling short lattice vectors and the closest lattice vector problem. In *Computational Complexity, 2002. Proceedings. 17th IEEE Annual Conference on*, pages 53–57. IEEE.

Arvind, V. and Joglekar, P. S. (2008). Some sieving algorithms for lattice problems. In *LIPIcs-Leibniz International Proceedings in Informatics*, volume 2. Schloss Dagstuhl-Leibniz-Zentrum für Informatik.

Blömer, J. and Naewe, S. (2007). Sampling methods for shortest vectors, closest vectors and successive minima. In *International Colloquium on Automata, Languages, and Programming*, pages 65–77. Springer.

Chang, X., Wen, J., and Xie, X. (2013). Effects of the LLL reduction on the success probability of the Babai point and on the complexity of sphere decoding. *IEEE Transactions on Information Theory*, 59(8):4915–4926.

Chen, Y. and Nguyen, P. Q. (2011). BKZ 2.0: Better lattice security estimates. In *International Conference on the Theory and Application of Cryptology and Information Security*, pages 1–20. Springer.

Conway, J. and Sloane, N. (1982). Fast quantizing and decoding and algorithms for lattice quantizers and codes. *IEEE Transactions on Information Theory*, 28(2):227–232.

Eisenbrand, F. (2010). Integer programming and algorithmic geometry of numbers. *50 Years of Integer Programming 1958-2008*, pages 505–559.

Gama, N. and Nguyen, P. (2008a). Predicting lattice reduction. *Advances in Cryptology–EUROCRYPT 2008*, pages 31–51.

Gama, N. and Nguyen, P. Q. (2008b). Finding short lattice vectors within Mordell's inequality. In *Proceedings of the fortieth annual ACM symposium on Theory of computing*, pages 207–216. ACM.

Gama, N., Nguyen, P. Q., and Regev, O. (2010). Lattice enumeration using extreme pruning. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 257–278. Springer.

Gentry, C. and Szydlo, M. (2002). Cryptanalysis of the revised NTRU signature scheme. In *International Conference on the Theory and Applications of Cryptographic Techniques*, pages 299–320. Springer.

Goldreich, O., Goldwasser, S., and Halevi, S. (1997). Public-key cryptosystems from lattice reduction problems. In *Advances in Cryptology-CRYPTO'97: 17th Annual International Cryptology Conference, Santa Barbara, California, USA, August 1997. Proceedings*, page 112. Springer.

Goldreich, O., Micciancio, D., Safra, S., and Seifert, J.-P. (1999a). Approximating shortest lattice vectors is not harder than approximating closest lattice vectors. *Information Processing Letters*, 71(2):55–61.

Goldreich, O., Ron, D., and Sudan, M. (1999b). Chinese remaindering with errors. In *Proceedings of the thirty-first annual ACM symposium on Theory of computing*, pages 225–234. ACM.

Hanrot, G., Pujol, X., and Stehlé, D. (2011a). Algorithms for the shortest and closest lattice vector problems. In *International Conference on Coding and Cryptology*, pages 159–190. Springer.

Hanrot, G., Pujol, X., and Stehlé, D. (2011b). Analyzing blockwise lattice algorithms using dynamical systems. In *CRYPTO*, volume 6841, pages 447–464. Springer.

Hanrot, G. and Stehlé, D. (2007). Improved analysis of kannans shortest lattice vector algorithm. In *Annual International Cryptology Conference*, pages 170–186. Springer.

Hermite, C. (1850). Extraits de lettres de M. Ch. Hermite à M. Jacobi sur différents objects de la théorie des nombres. *Journal für die reine und angewandte Mathematik*, 40:261–277.

Hoffstein, J., Pipher, J., and Silverman, J. H. (1998). NTRU: A ring-based public key cryptosystem. In *International Algorithmic Number Theory Symposium*, pages 267–288. Springer.

Kannan, R. (1987a). Algorithmic geometry of numbers. *Annual review of computer science*, 2(1):231–267.

Kannan, R. (1987b). Minkowski's convex body theorem and integer programming. *Mathematics of operations research*, 12(3):415–440.

Khot, S. (2005). Hardness of approximating the shortest vector problem in lattices. *Journal of the ACM (JACM)*, 52(5):789–808.

Korkine, A. and Zolotareff, G. (1873). Sur les formes quadratiques. *Mathematische Annalen*, 6(3):366–389.

Laarhoven, T., van de Pol, J., and de Weger, B. (2012). Solving hard lattice problems and the security of lattice-based cryptosystems. *IACR Cryptology EPrint Archive*, 2012:533.

Lenstra, A. K., Lenstra, H. W., and Lovász, L. (1982). Factoring polynomials with rational coefficients. *Mathematische Annalen*, 261(4):515–534.

Magliveras, S. S., van Trung, T., and Wei, W. (2008). Primitive sets in a lattice. *Australasian Journal of Combinatorics*, 40:173.

Merkle, R. and Hellman, M. (1978). Hiding information and signatures in trapdoor knapsacks. *IEEE transactions on Information Theory*, 24(5):525–530.

Micciancio, D. (2001). The hardness of the closest vector problem with preprocessing. *IEEE Transactions on Information Theory*, 47(3):1212–1215.

Micciancio, D. (2008). Efficient reductions among lattice problems. In *Proceedings of the nineteenth annual ACM-SIAM symposium on Discrete algorithms*, pages 84–93. Society for Industrial and Applied Mathematics.

Micciancio, D. and Goldwasser, S. (2012). *Complexity of lattice problems: a cryptographic perspective*, volume 671. Springer Science & Business Media.

Micciancio, D. and Voulgaris, P. (2013). A deterministic single exponential time algorithm for most lattice problems based on voronoi cell computations. *SIAM Journal on Computing*, 42(3):1364–1391.

Micciancio, D. and Walter, M. (2016). Practical, predictable lattice basis reduction. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 820–849. Springer.

Minkowski, H. (1905). Diskontinuitätsbereich für arithmetische äquivalenz. *Journal für die reine und angewandte Mathematik*, 129:220–274.

Newman, M. (1972). *Integral matrices*, volume 45. Academic Press.

Nguyen, P. Q. (2009). Hermite's constant and lattice algorithms. In *The LLL Algorithm*, pages 19–69. Springer.

Schnorr, C.-P. (1987). A hierarchy of polynomial time lattice basis reduction algorithms. *Theoretical computer science*, 53(2):201–224.

Schnorr, C.-P. and Euchner, M. (1994). Lattice basis reduction: Improved practical algorithms and solving subset sum problems. *Mathematical programming*, 66(1-3):181–199.

van Emde Boas, P. (1981). *Another NP-complete partition problem and the complexity of computing short vectors in a lattice*. Universiteit van Amsterdam. Mathematisch Instituut.

Wen, J. and Chang, X. (2017a). A KZ reduction algorithm. *arXiv preprint arXiv:1702.08152*.

Wen, J. and Chang, X. (2017b). Success probability of the Babai estimators for box-constrained integer linear models. *IEEE Transactions on Information Theory*, 63(1):631–648.

Wen, J., Tong, C., and Bai, S. (2016). Effects of some lattice reductions on the success probability of the zero-forcing decoder. *IEEE Communications Letters*, 20(10):2031–2034.

Wubben, D., Seethaler, D., Jaldén, J., and Matz, G. (2011). Lattice reduction. *IEEE Signal Processing Magazine*, 28(3):70–91.