A Digital Signature Scheme Based on CVP_∞[*]

Thomas Plantard, Willy Susilo, and Khin Than Win

Centre for Computer and Information Security Research
School of Computer Science and Software Engineering
University of Wollongong
Wollongong NSW 2522, Australia
{thomaspl,wsusilo,win}@uow.edu.au

Abstract. In Crypto 1997, Goldreich, Goldwasser and Halevi (GGH) proposed a lattice analogue of McEliece public key cryptosystem, which security is related to the hardness of approximating the closest vector problem (CVP) in a lattice. Furthermore, they also described how to use the same principle of their encryption scheme to provide a signature scheme. Practically, this cryptosystem uses the euclidean norm, l_2-norm, which has been used in many algorithms based on lattice theory. Nonetheless, many drawbacks have been studied and these could lead to cryptanalysis of the scheme. In this paper, we present a novel method of reducing a vector under the l_∞-norm and propose a digital signature scheme based on it. Our scheme takes advantage of the l_∞-norm to increase the resistance of the GGH scheme and to decrease the signature length. Furthermore, after some other improvements, we obtain a very efficient signature scheme, that trades the security level, speed and space.

1 Introduction

After the seminal work by Ajtai and Dwork [3] and the first lattice-based cryptosystem from Goldreich, Goldwasser and Halevi [21], many cryptosystems based on lattice theory have been proposed. These systems use the Shortest Vector Problem (SVP) or the Closest Vector Problem (CVP) as their underlying hard problem to construct the trapdoor functions. For a recent survey on the SVP-based cryptosystem, we refer the readers to [47].

In Crypto 1997, Goldreich, Goldwasser and Halevi (GGH) proposed a cryptosystem based on the lattice theory [21], which is a lattice analogue of the McEliece cryptosystem [37]. The security of GGH is related to the hardness of approximating the CVP in a lattice. Furthermore, they also noted that using the underlying principle of their encryption scheme, a signature scheme can be constructed. Nonetheless, the resulting signature scheme did not attract much interest in the research community until a relatively efficient signature scheme called the NTRUSign was proposed [28]. The GGH signature system can be described using three algorithms:

<u>Setup:</u> Compute a "good basis" and a "bad basis" of a lattice $\mathcal{L}$. $\mathcal{L}(G) = \mathcal{L}(B)$. Provide B as public and keep G secret.

[*] This work is supported by ARC Discovery Grant DP0663306.

R. Cramer (Ed.): PKC 2008, LNCS 4939, pp. 288–307, 2008.

Sign: Use the good basis to have an efficient approximation of the closest vector of a vector. The initial vector is the *message* and the approximation is the *signature*. GGH uses the first Babai's method [6] to approximate CVP: $s = \lceil mG^{-1} \rfloor G$ where $\lceil x \rfloor$ represent the closest integer of x if x is a real and the vector $[\lceil x_0 \rfloor, \lceil x_1 \rfloor, \ldots, \lceil x_{n-1} \rfloor]$ if x is a vector of $\mathbb{R}^n$.

Verify: Check if the approximation is in the lattice of basis $\mathcal{L}(B)$: $\exists x \overset{?}{\in} \mathbb{Z}^n, s = xB$. The vector-signature should be also a good approximation of the vector-message.

The important points for the security and efficiency of this cryptosystem are defined as follows.

i) It is easy to compute a "bad basis" from a "good basis", but it is difficult to compute a "good basis" from a "bad basis".
ii) It is easy to compute a good approximation of CVP with a "good basis" but difficult to do so with a "bad basis".
iii) It is easy to check the inclusion of a vector in a lattice even with a "bad basis".

In 1999, Nguyen [41] proposed the first attack against the GGH cryptosystem. This attack is based on the utilization by GGH of a non singular matrix with a small norm for a good basis to use Babai's method. Due to this attack, the utilization of GGH requires a lattice with big dimension (> 500), to ensure its security. Nonetheless, the computation of the Babai's approximation becomes very expensive. In 2001, Micciancio [38] proposed some major improvements of the speed and the security of GGH. In this scheme, the public key uses the Hermite Normal Form (HNF) basis for the "bad basis". The HNF basis is better to answer the inclusion question and it also seems to be more difficult to transform to a "good basis" compared to another basis. For the signature scheme, Micciancio used the reduced-vector instead of a closest vector. The reduced vector is in fact the difference between a vector and its closest vector. Using this method, the length of the signature is shorter. In 2002, Gentry and Szydlo [19] found a problem in GGH signature scheme which seems to be not zero-knowledge. Szydlo gave an algorithm [53] to elaborate this problem further. This method uses several vector-signatures given by the Babai's method to attack GGH. However, this method seems to be not very efficient. In 2003, NTRUSign [28] was created based on a very similar method to GGH but with most improvements on the utilization of NTRU basis [29] for the "good basis". Those basis seem to be more resistant against the previously known attacks. Nevertheless, in 2006, Nguyen and Regev [42] proposed a general attack against both GGH signature scheme and NTRUSign. This clever attack used the large CVP approximations naturally given by the signature of messages to design the fundamental parallelepiped of the "good basis".

Our Results

In this paper, we intend to use the l_∞-norm instead of the l_2-norm to construct a digital signature scheme which is similar to GGH signature scheme. By using the l_∞-norm, we aim to increase the security of the resulting cryptosystems, together with its efficiency in terms of signature length and time computation.

Paper Organization
This paper is organized as follows. We start the paper by providing some preliminary work and knowledge on lattice theory for cryptography. Then, we proceed with the eigenvalue theory and other useful definitions used throughout this paper. Then, we present the main part of the work, which is the reduction vector in l_∞-norm and the related theorems, followed by a signature scheme and its further improvements. Finally, we conclude the paper by comparing our scheme with the GGH signature scheme.

2 Lattice Theory for Cryptography

In this section, we will review some basic concepts of the lattice theory, and in particular addressing the NP-hardness of the trapdoor problems used. For a more complex account, we refer the readers to [45].

The lattice theory, also known as the geometry of numbers, has been introduced by Minkowski in 1896 [40]. The complete discussion on the basic of lattice theory can be found from [11,36,15].

Definition 1 (Lattice). *A* lattice $\mathcal{L}$ *is a discrete sub-group of* $\mathbb{R}^n$*, or equivalently the set of all the integral combinations of* $d \leq n$ *linearly independent vectors over* $\mathbb{R}$*.*

$$\mathcal{L} = \mathbb{Z}\, b_1 + \cdots + \mathbb{Z}\, b_d, \quad b_i \in \mathbb{R}^n.$$

$B = (b_1, ..., b_d)$ *is called a basis of* $\mathcal{L}$*,* d*, the dimension of* $\mathcal{L}$*.*

Definition 2 (Full-rank Lattice). *Let* $\mathcal{L} \subset \mathbb{R}^n$ *be a lattice. If its dimension* d *is equal to* n *then the lattice* $\mathcal{L}$ *is called full-rank.*

Definition 3 (Fundamental Parallelepiped). *Let be* $B = (b_1, ..., b_n)$ *a basis of a full-rank lattice* $\mathcal{L} \subset \mathbb{R}^n$ *then the set*

$$\mathcal{H} = \left\{ \sum_{i=1}^{n} x_i b_i, (x_1, \ldots, x_n) \in [0, 1[^n \right\}$$

is called a fundamental parallelepiped.

The volume of a fundamental parallelepiped is invariant regardless of the chosen basis. This invariant is called the *determinant* of $\mathcal{L}$ and can be computed as $\det \mathcal{L} = |\det B|$.

Remark 1. There also exists a definition of the determinant for a non full-rank lattice. However, in this paper, we only focus on the basic of lattice theory that is required throughout the paper. Since we only deal with full-rank integer lattice, consequently with a basis $B \in \mathbb{Z}^{n,n}$, therefore we simplify the definition as above.

For a given lattice $\mathcal{L}$, there exists an infinity of basis. However, the Hermite Normal Form basis (Definition 4) is unique [13].

Definition 4 (HNF). *Let $\mathcal{L}$ be a full-rank lattice and H a basis of $\mathcal{L}$. H is a Hermite Normal Form basis of $\mathcal{L}$ if and only if*

$$\forall i, j, \quad H_{i,j} \begin{cases} = 0 & \text{if } i < j \\ \geq 0 & \text{if } i \geq j \\ < H_{j,j} & \text{if } i > j \end{cases}$$

The HNF basis can be computed from a given basis in a polynomial time [32]. For efficient solutions, we refer the readers to [39].

Remark 2. The HNF basis is a “good basis” for solving the problem of inclusion of a vector in a lattice [13]. As it was successfully used by [38], we will also incorporate it in this paper with some further improvements.

Many algorithmic problems of the lattice theory are built upon two other problems which are clearly more difficult, namely the Shortest Vector Problem (SVP) and the Closest Vector Problem (CVP).

Definition 5 (SVP). *Let B be a given basis of a lattice $\mathcal{L}$. The Shortest Vector Problem is to find a vector $u \neq 0$ such that $\forall v \in \mathcal{L}, \|u\| \leq \|v\|$ for a given norm $\|.\|$.*

Definition 6 (CVP). *Let B be a given basis of a lattice $\mathcal{L}$ and w a vector. The Closest Vector Problem is to find a vector u such that $\forall v \in \mathcal{L}, \|w-u\| \leq \|w-v\|$ for a given norm $\|.\|$.*

CVP is NP-hard for all norms l_p (Definition 7) including l_∞-norm [9].

Definition 7 (l_p-norm). *Let w be a vector of $\mathbb{R}^n$. The l_p-norm is the function $\|.\|_p$ such that $\|w\|_p = \left(\sum_{i=0}^{n-1} |w_i|^p\right)^{1/p}$.*

The l_2-norm is also known as the euclidean norm. The l_∞-norm, also known as the infinity norm, is computed as $\|w\|_\infty = \max\{|w_i|, 0 \leq i < n\}$.

The l_2 and l_∞ norms have been studied and used in the lattice theory. The NP-hardness of the two problems for these two norms has been proven. In 1981, Emde Boas proved the NP-hardness of CVP_∞, SVP_∞ and CVP_2 in [9]. Subsequently, in 1998, Ajtai proved the NP-hardness of SVP_2 in [2]. Consequently, there exists only some exponential algorithms to completely solve those problems. We summarize this result in the table 1.

However, some approximation versions of these two problems exist in the literature.

Definition 8 (AppSVP, resp. AppCVP). *Let B be a given basis of a lattice $\mathcal{L}$, w a vector and a real $\gamma \geq 1$. The AppSVP, resp. AppCVP, is to find a vector u such that $\forall v \in \mathcal{L}, \|u\| \leq \gamma\|v\|$, resp. $\|w-u\| \leq \gamma\|w-v\|$ for a given norm $\|.\|$.*

Table 1. Exponential algorithms for SVP and CVP

	Deteministic	Probabilistic
SVP	$d^{\frac{d}{2e}}$ [31,26,24]	$(2+\frac{1}{\epsilon})^d$ [4,8]
CVP	$d^{\frac{d}{2}}$ [31,26,24]	$(2+\frac{1}{\epsilon})^d$ [5,8]

The NP-hardness of these two approximation problems has also been well studied (for more detail, see [10] or more recently [46]). Table 2 summarizes some main results on the NP-hardness of these two approximation problems for the euclidean and the infinity norms for the approximation factor γ in function of the dimension d of the studied lattice.

[22] proved that SVP is not harder than CVP.

Table 2. The approximation factor γ for the NP-hardness of AppSVP and AppCVP with l_2 and l_∞ norms

	Euclidean Norm		Infinity Norm	
Problems	$AppSVP_2$	$AppCVP_2$	$AppSVP_\infty$	$AppCVP_\infty$
NP-hard	$2^{\log^{1-\epsilon} d}$ [25]	$2^{\log^{1-\epsilon} d}$ [17]	$d^{1/\log\log d}$ [16]	$d^{1/\log\log d}$ [16]
not NP-hard [1]	$\sqrt{d/\log d}$ [20]	$\sqrt{d/\log d}$ [20]	$d/\log d$ [20]	$d/\log d$ [20]

Remark 3. Table 2 seems to show that the approximation problems seem to be more difficult for the l_∞-norm compared to the l_2-norm. This impression is supported by a recent paper by Khot [33] which presented a result that proved that SVP will be more and more difficult in l_p if p grows. A more recent paper of Regev and Rosen [48] proved that a lot of classic problems, including SVP and CVP, are easier under the l_2-norm than under every other l_p-norm, including l_∞-norm.

Remark 3 is supported by the fact that most of the polynomial and efficient algorithm to approximate SVP and CVP are for the l_2-norm.

– For SVP, in 1982 Lenstra, Lenstra and Lovasz [35] proposed a powerful polynomial algorithm, known as the LLL algorithm, to efficiently approximate SVP and more generally the length of the basis itself. This algorithm approximate SVP for the l_2-norm within an approximation factor $\gamma = 2^{(d-1)/2}$ in theory but seems to be much more efficient in practice [44]. In addition, a lot of improvements have been proposed on LLL to obtain a better approximation factor and/or a better time complexity. For the recent result on LLL, refer to [43,52]. Combining this approach with the BKZ method [49,50], which can be seen as a generalization of LLL, is a very powerful way to attack a cryptosystem based or linked to SVP_2.

[1] Unless the polynomial hierarchy collapses.

- For CVP, in 1986 Babai [6] proposed two polynomial methods. Those algorithms approximate CVP for the l_2-norm within a factor $\gamma = 1 + 2d(9/2)^{d/2}$ and $\gamma = 2^{d/2}$, respectively. Babai's algorithms use an LLL-reduced basis. Consequently all the variants of LLL, including BKZ utilization [51] proposed by Schnorr, are naturally the improvement of Babai's methods. Moreover, there exists an heuristic way to directly approximate CVP using an approximate algorithm for SVP [41]. See [1] for a general survey of AppCVP.

All the existing algorithms have been created for the euclidean norm. Nevertheless, the l_2-norm algorithm can be used to approximate SVP and CVP for the l_∞-norm using the equivalence of norms, $\forall v \in \mathbb{R}^n, \|v\|_\infty \leq \|v\|_2 \leq n^{1/2}\|v\|_\infty$ [23].

The final approximation for l_∞ will be clearly worst than for l_2 and this method cannot be used to solve exactly the SVP and CVP under l_∞.

Remark 4. In this paper, we aim to construct a lattice-based cryptosystem which is more resistant than the existing ones in the literature using the l_∞-norm. A recent work by Chen and Meng [12] clearly went this way. They proved the NP-hardness of the closest vector problem with preprocessing over l_∞-norm. Regev and Rosen [48] gave the factor of $\log d^{1/2-\epsilon}$ for the NP-hardness of CVP with preprocessing under l_p-norm, $2 \leq p \leq \infty$.

3 Matrix Norm, Eigenvalues, Spectral Radius and Condition Number

In this section, we briefly review some definitions of the eigenvalue theory that will be required throughout this paper. Most of the following definitions and properties can been found in [14,55,30]. In the following definitions, let $n \in \mathbb{N}$.

Definition 9 (Matrix Norm). *Let A be a square matrix in $\mathbb{C}^{n,n}$. A matrix norm denoted as $\|A\|$ is said to be* consistent *to a vector norm $\|.\|$, if we have $\|A\| = sup\{\|xA\|, \quad x \in \mathbb{C}^n, \quad \|x\| = 1\}$.*

The matrix norm $\|.\|_p$, consistent to the vector norm defined in Definition 7, can be easily computed for $p = 1, 2, \infty$. For other values of p, see [27] for estimating methods of $\|.\|_p$.

Definition 10 (Polytope Norm). *We denote $\|.\|_P$ as the matrix norm consistent to the vector norm $\|.\|_P$ defined as $\forall v \in \mathbb{C}^n, \quad \|v\|_P = \|vP^{-1}\|_\infty$ where P is a non singular matrix.*

To compute the polytope norm $\|.\|_P$ of a matrix, we have $\forall A \in \mathbb{C}^{n,n}, \quad \|A\|_P = \|PAP^{-1}\|_\infty$.

Definition 11 (Eigenvalue). *Let A be a square matrix in $\mathbb{C}^{n,n}$, a complex number λ is called a eigenvalue of A if there exists a column-vector $h \neq 0$ such that $Ah = \lambda h$. The column-vector h is called an eigenvector of A.*

If h is an eigenvector then for any real number $\alpha \neq 0$, αh is also an eigenvector. A matrix composed by n eigenvectors of n eigenvalues is an eigenmatrix. There is an infinity of eigenmatrix. We specially focus on the eigenmatrix H which minimizes the *condition number* (Definition 12) of the infinity norm.

Definition 12 (Condition Number). *Let $\|.\|$ be a matrix norm and A a non singular matrix. The condition number of A, denoted as $\kappa(A)$, is such that $\kappa(A) = \|A\|\|A^{-1}\|$.*

In this paper, $\kappa(A)$ use the l_∞-norm: $\kappa(A) = \|A\|_\infty \|A^{-1}\|_\infty$.

Definition 13 (Spectral Radius). *Let A be a square matrix in $\mathbb{C}^{n,n}$. We denote $\rho(A)$ as the spectral radius of A defined as the maximum of the absolute value of the eigenvalues of A: $\rho(A) = max\,\{|\lambda|, Ax = x\lambda\}$.*

Theorem 1. *For any matrix norm $\|.\|$, $\forall A \in \mathbb{C}^{n,n}, \quad \rho(A) \leq \|A\|$.*

In fact, the spectral radius can be seen as the lower bound of all the matrix norm of a matrix: $\rho(A) = inf\,\{\|A\|\}$.

The spectral radius has some useful properties as follows.

Theorem 2. *For any matrix norm $\|.\|$ and any square matrix A , $\lim_{k\to\infty} \|A^k\| = \rho(A)^k$.*

Using this property, we can obtain the following property.

Theorem 3. *Let $A \in \mathbb{C}^{n,n}$ be a square matrix, the series $I + A + A^2 + A^3 + \ldots$ converge to $\frac{1}{1-A}$ if and only if $\rho(A) < 1$ where $\rho(A)$ is the spectral radius of A.*

See [55] for the proofs of Theorems 1, 2 and 3.
The last property of the spectral radius that will be used in this paper is provided in the Theorem 4.

Theorem 4. *For any square matrix A and any real number $\epsilon > 0$, there exists a polytope norm $\|.\|_P$ such that $\|A\|_P \leq \rho(A) + \epsilon$.*

The proof of Theorem 4 is given in [30] by providing a way to compute the matrix P. In fact, there exists an infinity of such matrix P connected by a multiplication by a non singular diagonal matrix. If the eigenvalues are distinct, we can use an eigenmatrix for P. Here, we focus on the matrix P that minimizes $\kappa(P)$.

4 Vector Reduction in l_∞-Norm

In this section, we propose a new method of vector reduction using a modification of the Babai's method. This new algorithm uses another definition of a "good basis" to obtain an approximation of CVP_∞. To approximate the closest vector w of a vector v, Babai used the approximation given by the equation

$u = \lceil vG^{-1} \rfloor G$. As explained previously, this approximation has two major problems when it is used in cryptography, namely an expensive computation and a mark of the "good basis" on the approximate vector. To solve these two problems, we propose a new approximation of the vector v. This approximation is inspired by the work of Bajard, Imbert and Plantard [7] which proposed a method to reduce some number representation for modular arithmetic. The method used in this paper can be seen as a generalization of their technique. An important point is the conservation of the efficiency which is main feature in modular arithmetic operations.

Our focus is on the reduced vector, v mod $\mathcal{L}$, and *not* on the closest vector. We note that these two problems are completely equivalent. The reduced vector w is equal to the difference between a vector v and its closest vector u. So to reduce a vector, the Babai method becomes $w = v - \lceil vG^{-1} \rfloor G$. We decompose G into two matrices: $G = D - M$. We will see that the choice of D and M determine if G is a "good basis" or not. We use this decomposition to approximate v.

$$w = v - \lceil v(D-M)^{-1} \rfloor G.$$

We assume that D is non singular, so we are able to compute D^{-1}.

$$\begin{array}{l} w = v - \lceil v((1-MD^{-1})D)^{-1} \rfloor G \\ w = v - \lceil vD^{-1}(1-MD^{-1})^{-1} \rfloor G. \end{array}$$

We modify the Babai's approximation to a new approximation.

$$w' = v - \lceil vD^{-1} \rfloor \lceil (1-MD^{-1})^{-1} \rfloor G.$$

Let's analyze more precisely the second part of this approximation. If we have the spectral radius $\rho(MD^{-1}) < 1$, we can use the Theorem 3 to obtain

$$\lceil (1-MD^{-1})^{-1} \rfloor = \lceil 1 + MD^{-1} + (MD^{-1})^2 + (MD^{-1})^3 + \dots \rfloor$$

Since $\rho(MD^{-1}) < 1$, this series on the right term converges. Here, we make a very quick approximation of $\lceil (1-MD^{-1})^{-1} \rfloor$ to 1. At the end of this analysis, we propose a new approximation w of the closest vector of v.

$$w = v - \lceil vD^{-1} \rfloor (D-M).$$

We will consider this approximation to be *precise enough* if $\rho(MD^{-1}) < 1$. Hence, we propose a new definition of a "good basis" as follows.

Definition 14 (Good Basis). *Let D, M be two square matrices and $\mathcal{L}$ be the lattice which has $D-M$ for the basis. $D-M$ is called a "good basis" of $\mathcal{L}$ if $\rho(MD^{-1}) < 1$.*

Now, we can propose an algorithm to reduce a vector v with a "good basis".

Algorithm 1. Vector Reduction

```
Input  : A vector $v \in \mathbb{Z}^n$.
Data   : A non-singular diagonal matrix $D \in \mathbb{Z}^{n,n}$ and a square matrix
         $M \in \mathbb{Z}^{n,n}$. A lattice $\mathcal{L}$ of basis $D - M$.
Output : A vector $w \in \mathbb{Z}^n$ such that $w \equiv v \pmod{\mathcal{L}}$ and $\|w\|_D < 1$.
begin
    $w \leftarrow v$;
    repeat
        $q \leftarrow \lceil wD^{-1} \rfloor$;
        $w \leftarrow w - q(D - M)$;
    until $\|w\|_D < 1$;
end
```

Algorithm 1 has a loop and hence, it repeats its approximation several times. This is different from the Babai's algorithm which does not have any loop. In our case, the loop is required to replace the approximation of $\lceil (1 - MD^{-1})^{-1} \rfloor$ by 1. The loop corresponds to the different power of MD^{-1} that we have omitted.

Remark 5. The Algorithm 1 returns a vector with $\|w\|_D = \|wD^{-1}\|_\infty < 1$, which is the reason why we consider it like an approximation of CVP_∞. However, it is only true when $D = \beta Id$ that we have a classic definition of l_∞ reduction. The important point is that the coefficients $|w_i| < D_{i,i}$ do not depend on any average or any direct influence from the other coefficients of w. This property comes from the polytope norm which includes the l_∞-norm. That is the intrinsic difference between the l_∞-norm, a polytope norm, and the l_2-norm, a ellipsoidal norm.

It is trivial to prove that Algorithm 1 is exact.

a) $w = w - q(D - M)$ with $q \in \mathbb{Z}^n$. The loop does not change the congruence of w mod $\mathcal{L}$. So at the end, $w \equiv v$ mod $\mathcal{L}$ holds.
b) If Algorithm 1 ends then $\|w\|_D < 1$.

However, condition for Algorithm 1 termination has to be defined. There exists a very similar problem of successive approximation convergence in the literature. To compute a vector x with $xA = y$ for some problematic matrix A, a complete theory has been developed with some equivalent decomposition, $A = D - M$ where A is called M-matrix. Some equivalent result for convergence, $\rho(MD^{-1}) < 1$ has been found. See [54,34] for more detail on this theory.

However, even if this theory is very similar, it does not solve the question of Algorithm 1 termination. Therefore, we propose Theorem 5 which is inspired by such a theory to answer this question.

Theorem 5. *Let $n \in \mathbb{N}$, $D, M \in \mathbb{Z}^{n,n}$ be two square matrices with D non singular and diagonal. The successive approximation w_i of a vector w given by $w_0 = w$ and $w_i = w_{i-1} - \lceil w_{i-1}D^{-1} \rfloor (D - M)$ for $i > 0$.*

i) For any l_p-norm with $\|MD^{-1}\|_p < 1$, we have $\lim_{i\to\infty}\|w_i\|_D \leq \frac{\|1-MD^{-1}\|_p}{1-\|MD^{-1}\|_p}\frac{n^{1/p}}{2}$.

ii) For any polytope norm with $\|MD^{-1}\|_P < 1$, we have $\lim_{i\to\infty}\|w_i\|_D \leq \frac{\|1-MD^{-1}\|_P}{1-\|MD^{-1}\|_P}\frac{\kappa(P)}{2}$.

iii) For any non singular eigenmatrix P of MD^{-1}, we have $\lim_{i\to\infty}\|w_i\|_D \leq \frac{\rho(1-MD^{-1})}{1-\rho(MD^{-1})}\frac{\kappa(P)}{2}$.

Proof. First, we decompose the successive approximation

$$\begin{array}{l} w_i = w_{i-1} - \lceil w_{i-1}D^{-1}\rfloor (D-M) \\ w_i = w_{i-1} - (w_{i-1}D^{-1} + \epsilon_i)(D-M) \text{ where } \epsilon_i \in [-1/2, 1/2]^n \\ w_i = w_{i-1} - w_{i-1} + w_{i-1}D^{-1}M - \epsilon_i(D-M) \\ w_i = w_{i-1}D^{-1}M - \epsilon_i(D-M) \end{array}$$

We want to evaluate $w_iD^{-1} = w_{i-1}D^{-1}MD^{-1} - \epsilon_i(1-MD^{-1})$. Now, for any norm $\|.\|$, we have

$$\begin{array}{l} \|w_iD^{-1}\| = \|w_{i-1}D^{-1}MD^{-1} - \epsilon_i(1-MD^{-1})\| \\ \|w_iD^{-1}\| \leq \|w_{i-1}D^{-1}\|\|MD^{-1}\| + \|\epsilon_i\|\|(1-MD^{-1})\| \end{array}$$

Let be Δ the max of $\|\epsilon_i\|$, we obtain $\|w_iD^{-1}\| = \|w_{i-1}D^{-1}\|\|MD^{-1}\| + \Delta\|(1-MD^{-1})\|$. So, if $\|MD^{-1}\| < 1$ this sequence converge to $\lim_{i\leftarrow\infty}\|w_iD^{-1}\| \leq \Delta\|(1-MD^{-1})\|\sum_{i=0}^{\infty}\|MD^{-1}\|$. Because we have $\|MD^{-1}\| < 1$, we obtain

$$\lim_{i\leftarrow\infty}\|w_iD^{-1}\| \leq \Delta\frac{\|(1-MD^{-1})\|}{1-\|MD^{-1}\|}.$$

To finish this proof, we have to adapt this result to different norm.

i) If $\|.\|$ is a l_p-norm, we obtain $\lim_{i\leftarrow\infty}\|w_iD^{-1}\|_p \leq \Delta\frac{\|(1-MD^{-1})\|_p}{1-\|MD^{-1}\|_p}$.
We can evaluate $\Delta = \frac{n^{1/p}}{2}$.

$$\lim_{i\leftarrow\infty}\|w_iD^{-1}\|_p \leq \frac{\|(1-MD^{-1})\|_p}{1-\|MD^{-1}\|_p}\frac{n^{1/p}}{2}$$

We know also that for any vector v, $\|v\|_\infty \leq \|v\|_p$.

$$\lim_{i\leftarrow\infty}\|w_iD^{-1}\|_\infty \leq \frac{\|(1-MD^{-1})\|_p}{1-\|MD^{-1}\|_p}\frac{n^{1/p}}{2}$$

With the definition of the $\|.\|_D$ norm, we obtain

$$\lim_{i\leftarrow\infty}\|w_i\|_D \leq \frac{\|(1-MD^{-1})\|_p}{1-\|MD^{-1}\|_p}\frac{n^{1/p}}{2}$$

ii) If $\|.\|$ is a polytope norm $\|.\|_P$, we obtain $\lim_{i\leftarrow\infty}\|w_iD^{-1}\|_P \leq \varDelta\frac{\|(1-MD^{-1})\|_P}{1-\|MD^{-1}\|_P}$. We can evaluate $\varDelta = \frac{1}{2}\|P^{-1}\|_\infty$.

$$\lim_{i\leftarrow\infty}\|w_iD^{-1}\|_P \leq \frac{\|(1-MD^{-1})\|_P}{1-\|MD^{-1}\|_P}\frac{\|P^{-1}\|_\infty}{2}$$

By definition, we have $\|wD^{-1}\|_P = \|wD^{-1}P^{-1}\|_\infty$. To evaluate $\|w\|_D$, we have $\|w\|_D = \|wD^{-1}\|_\infty = \|wD^{-1}P^{-1}P\|_\infty \leq \|wD^{-1}P^{-1}\|_\infty\|P\|_\infty$. Now, we can evaluate the limit of $\|w_i\|_D$.

$$\begin{array}{l}\lim_{i\leftarrow\infty}\|w_i\|_D \leq \frac{\|(1-MD^{-1})\|_P}{1-\|MD^{-1}\|_P}\frac{\|P^{-1}\|_\infty}{2}\|P\|_\infty\\ \lim_{i\leftarrow\infty}\|w_i\|_D \leq \frac{\|(1-MD^{-1})\|_P}{1-\|MD^{-1}\|_P}\frac{\kappa(P)}{2}\end{array}$$

iii) If $\|.\|$ is a polytope norm $\|.\|_P$ where P is an non singular eigenmatrix of MD^{-1}, we obtain the same result with $\|MD^{-1}\|_P = \rho(MD^{-1})$. We have also $\|1-MD^{-1}\|_P = \rho(1-MD^{-1})$ because an eigenmatrix of A is also a eigenmatrix of any polynomial composition of A.

$$\lim_{i\leftarrow\infty}\|w_i\|_D \leq \frac{\rho(1-MD^{-1})}{1-\rho(MD^{-1})}\frac{\kappa(P)}{2}$$ □

We note that this proof is very similar and inspired by some proofs found in [34] to solve close problem of successive approximation convergence.

Remark 6. Theorem 5 clearly provides some conditions to terminate Algorithm 1. These three conditions are complementary.

i) The l_p-norm can be used to have a fast approximation. See [27] for some methods to compute l_p norm for a matrix if p is not simple $p = 1, 2, \infty$.
ii) The polytope norm provides a way to be closer to $\|MD^{-1}\|_P \sim \rho(MD^{-1})$ which is the lower bound. But its computation can be long to minimize $\kappa(P)$.
iii) The non singular eigenmatrix are the best evaluation but it requires us to have distinct eigenvalues, which we do not always have.

In fact, after several practical tests and theoretical analysis, we are able to make a conjecture.

Conjecture 1. Let $n \in \mathbb{N}$, $D, M \in \mathbb{Z}^{n,n}$ be two square matrices with D non singular and diagonal. The successive approximation w_i of a vector w given by $w_0 = w$ and $w_i = w_{i-1} - \lfloor w_{i-1}D^{-1}\rfloor (D-M)$ for $i > 0$ converge if $\rho(MD^{-1}) < \frac{1}{2}$.

This conjecture will be used for the practical implementation of Algorithm 1. For the rest of this paper, sometimes we refer to $\rho(MD^{-1})$ only with ρ, when the context is clear.

5 Signature Scheme

In this section, we describe our new signature scheme, which comprises of the three algorithms: *Setup*, *Sign* and *Verify*.

Setup
a) Choose an integer n.
b) Compute a randomly integer matrix $M \in \{-1, 0, 1\}^{n,n}$.
c) Compute $D = \lfloor 2\rho(M) + 1 \rfloor Id$.
d) Compute the Hermite Normal Form H of the basis $D - M$.
e) The public key is (D, H), and the secret key is M.

Sign To sign a message $m \in \{0, 1\}^*$, one does the following.
a) Compute the vector $v = h(m) \in \mathbb{Z}^n$ where h is a hash function such that

$$\begin{aligned} h : \quad m \quad &\to v \\ : \{0,1\}^* &\to \{x \in \mathbb{Z}^n, \quad \|x\|_{D^2} < 1\} \end{aligned}$$

b) Using Algorithm 1, compute w, which is a reduced vector of v.
c) The signature on m is w.

Remark 7. The three choices of $M \in \{-1, 0, 1\}^{n,n}$, $\rho < 1/2$ and $\|x\|_{D^2} < 1$ are arbitrary and they can be changed. However these choices seem to be practically reasonable.

Verify To verify a message-signature pair, (m, w), one does the following.
a) Check if $\|w\|_D < 1$.
b) Compute the vector $h(m) \in \mathbb{Z}^n$.
c) Check if the vector $h(m) - w$ is in the lattice of basis H.

6 Improvements

In this section, we present some improvements to our scheme to make it practical. These improvements provide some choices to the main algorithm, in order to optimize it during the implementation of the algorithm.

6.1 Signature

The main part of the signing algorithm is in the reduction part as defined in (Algorithm 1). The fact that D is a diagonal matrix will simplify a lot of computations of wD^{-1}. This computation corresponds to the computation of the quotient of $\frac{w_i}{D_{i,i}}$. In fact the reduction algorithm needs the rest of this division as well. Based on this observation, we can rewrite Algorithm 1 as shown in Algorithm 2.

Remark 8. Algorithm 2 could be completely optimized by the utilization of $D = \beta Id$ with β be a power of two. This choice transforms the division corresponding to the two first lines of the loop to a shift operation. Hence, the reduction of a vector can be summarized to shift and addition operations, assuming that the matrix has low coefficients.

Algorithm 2. Sign

```
Input  : A vector $v \in \mathbb{Z}^n$
Data   : Two square matrices $D, M$
Output : A vector $w \in \mathbb{Z}^n$
begin
    $w \leftarrow v$;
    $i \leftarrow 0$;
    repeat
        $k \leftarrow 0$;
        $q \leftarrow \left\lfloor \frac{w_i}{D_{i,i}} \right\rceil$;
        $w_i \leftarrow w_i - qD_{i,i}$;
        for $j = 0$ to $n - 1$ do
            $w_{i+j \bmod n} \leftarrow w_{i+j \bmod n} + q \times M_{i,j}$;
            if $|w_{i+j \bmod n}| < D_{i+j \bmod n, i+j \bmod n}$ then $k = k + 1$;
        end
        $i \leftarrow i + 1 \bmod n$;
    until $k = n$;
end
```

6.2 Verification

The main part of the verification algorithm is the time to verify the inclusion of w in the lattice $\mathcal{L}$. As we described in Remark 2, the utilization of the HNF accelerates this computation and it was successfully used in [38]. If we choose to keep only some special lattices, then we can also do some further improvements.

Definition 15. *Let be H the HNF basis of a full-rank lattice $\mathcal{L}$, we will called H optimal if* $\forall i > 1 \quad H_{i,i} = 1$.

With an optimal HNF basis H, a vector w is in the lattice of basis H if and only if $\sum_{i=1}^{n-1} w_i \times H_{i,0} \equiv w_0 \pmod{H_{0,0}}$.

With this setting, we can propose a very simple algorithm to verify the signature as follows.

Algorithm 3. Verify

```
Data   : Two square matrices $D, H$
Input  : Two vectors $v, w \in \mathbb{Z}^n$
Output : A boolean
begin
    for $i = 0$ to $n - 1$ do if $|w_i| \geq D_{i,i}$ then return False;
    $s \leftarrow 0$;
    for $i = 1$ to $n - 1$ do $s \leftarrow s + (v_i - w_i) \times H_{i,0}$;
    if $s = v_0 - w_0 \bmod H_{0,0}$ then return True else return False
end
```

Remark 9. Optimal HNF simplifies the verification method and also minimizes the size of the public key. We note that in this case, we only need to send the first column of the matrix H. Consequently, we will use the optimal HNF for a "bad basis".

7 Comparison with GGH Signature Scheme

The advantage that our system has compared to the GGH signature scheme is the use of the l_∞-norm, which will make the scheme more resistant and difficult to attack. Furthermore, a shorter signature length and an efficient computation to compute with Algorithm 1 can be achieved with the help of fast arithmetic operations. The details of these advantages are provided in this section.

7.1 Resistance

An approximation of CVP_∞ also provides an approximation of CVP_2 by the equivalence of norm. Theoretically, the complexity of our cryptosystem cannot be less than the initial GGH signature scheme and Micciancio's improvements. However, parameter choices are essential to achieve a practical high resistance scheme.

The best basic way to attack our scheme is by finding M using D on $\mathcal{L}(H)$: $D \equiv M \pmod{\mathcal{L}(H)}$. In other words, $\forall i, \quad (0, \ldots, 0, D_{i,i}, 0, \ldots, 0) \equiv (M_{i,1}, \ldots, M_{i,n}) \pmod{\mathcal{L}(H)}$. The attacker has to find some very good approximations (most of the time the exact result) of the CVP for the l_∞-norm. This attack seems to be the easiest way compared to solving CVP_∞ for a given vector-message. If the attacker can solve CVP_∞ for every vector of D, he can use Algorithm 1 to create a false signature. Therefore, we consider an attack to be successful if the attacker can find a matrix M' such that $D \equiv M' \pmod{\mathcal{L}(H)}$ with $\rho(M'D^{-1}) < 1$ and not only if $M' = M$.

As remarked in Remark 3 the l_∞-norm seems to be more resistant. A powerful advantage of its system clearly comes from the intrinsic difference between the l_2 and the l_∞ norms. Effectively, the utilization of approximation algorithms for the l_2-norm to solve approximation problem for l_∞-norm will be worst. Moreover, some special matrices M could be used to take advantage of the intrinsic difference between those two norms to make those algorithms completely inefficient: the row vector M_i of M are such that $\|M_i\|_\infty < D_{i,i}$. If we take $\|M_i\|_2 > D_{i,i}$ or at least $\|M_i\|_2 \sim D_{i,i}$, it will raise some problems to use l_2 algorithm.

A brute force attack to find a row vector of M, where $M_{i,j} \in \{-1, 0, 1\}$, is $O(3^n)$. This brute force attack is faster than solving exactly a CVP using Kannan's method [31], which has the complexity of $n^{O(n)}$. Note that these two possible attacks are in the exponential order. When n is chosen to be large, then these techniques cannot be employed. Therefore, in order to attack it, only an approximation of CVP that can be computed, rather than solving it. Although the approximation of CVP is polynomial, the attack is a heuristic attack and therefore there is no assurance that the result is precise enough.

A theoretical timing attack is also possible as the time of the signature depends on the message-vector. However, such an attack seems very unlikely: to obtain information on the form of message vector if its reduction took 4 or 5 loops instead of

6 seems very hard. There exists a simple way to completely prevent this hypothetical attack. A simple improvement of Algorithm 2 is the utilization of a random initialization of i: $i \leftarrow rand(0 \dots n-1)$ instead of a classic $i \leftarrow 0$. Besides the fact that there is no real reason to begin with 0, this improvement will provide two advantages. Firstly, temporary approximation vectors are not the same between two reductions of the same vector: that will change the number of loops to reduce to the same vector. This property gives an advantage against side-channel attacks, like timing attack. The most important advantage is that this method grows the length of the set of vectors of $\{v, \|v\|_D\}$ that can be returned. This property provides a strong resistance against the attack described in [42].

Another remark is on the fact that D is public. However, GGH basis where taken as $\sqrt{n}Id - M$ with $M_{i,j} \in [-4, 4]$. So D can be easily guessed as well for GGH and attacks on GGH do not use this fact.

To finalize the comments on the security, we need to comment our scheme against the most successful attack against GGH signature scheme and NTRUSign. In 2006, Nguyen and Regev [42] proposed a clever way to design the fundamental parallelepiped using some signature-message which represent a CVP approximation. We also note that this attack will be ineffective against our system. All the signature-message are in $\{xD, x \in]-1, 1[^n\}$. Finding the design of this volume is not particularly useful since D is already given as a public parameter. In Figure 1, we present an example of some signature-message on $\mathbb{R}^2$ after reduction with Babai's method or with our method. Even if the dimension 2 is far away of cryptographic dimension, we can still see the mark used by [42]. In fact, we see that the vectors of the basis can be designed after enough Babai reductions, but that we can only design D after reduction by Algorithm 2.

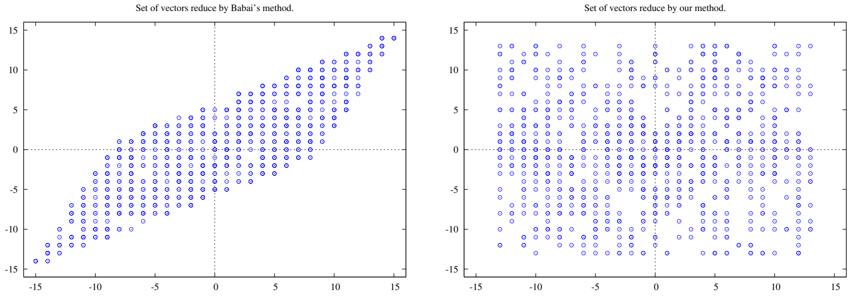

Fig. 1. Signature-message on $\mathbb{R}^2$ for Babai's reduction and our reduction

7.2 Speed

For an optimized version of the signature scheme (Algorithm 2), Algorithm 1 uses only shift and addition operations. However, we need to know the average number of loops to reduce a signature vector. Even if the proof of Theorem 5 gives us a bound on the worst case, the average case seems to be difficult to evaluate. In Figure 2, we present an average number of iterations from Algorithm 2. On every dimension $n \in [50, 350]$, we have compute a 100 random couples D, M following the methods used in the Setup algorithm: $M \in \{-1, 0, 1\}^{n,n}$ and $D = \lfloor 2\rho(M) + 1 \rfloor Id$. With each of this basis $D - M$, we have reduced 100 random message vector chosen in $[0, \lfloor 2\rho(M) + 1 \rfloor^2 [^n$. Figure 2 shows the average of the number of loops required to reduce a message vector to a signature vector.

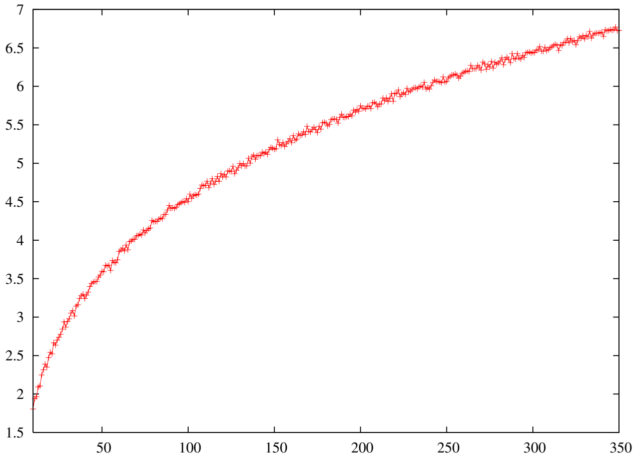

Fig. 2. Average number of loops used to reduce a message vector to a signature vector

From Figure 2, one can conclude that on average, the number of loops required for signing is between 5 and 7 to achieve a good security level, which is approximately began from 200. Furthermore, Figure 2 also shows that the average number of loops are logarithmic on n, $O(\log n)$. We note that our reduction is applicable only for some special lattices. Nevertheless, the resulting efficiency obtained from these lattices are very interesting to develop efficient and fast digital signature schemes. As explained earlier, a loop can be minimized to only shift and addition operations. It provides us with a very competitive way to reduce a vector when the first Babai's reduction uses two matrix multiplications. The first matrix multiplication in Babai's reduction is the most expensive operation, since it requires a high precision on a floating point matrix multiplication. In contrast to Babai's method, our method can be used in a huge dimension that will provide higher level of security without any time constraint.

7.3 Space

In this section, we provide some evaluation on the signature space. l_∞-norm is naturally the norm used to evaluate the space complexity of a signature. The fact that Algorithm 1 deals directly with this norm makes an important difference with Babai's method.

Figure 3 shows result of test on the l_∞-norm of reduce vector. We present three curves corresponding to three parameters. For every dimension n ($n \in [50, 350]$), we compute on 100 random matrices chosen in $M \in \{-1, 0, 1\}^{n,n}$,

i) the average spectral radius of M,
ii) the average $\|D\|_\infty$ that we can pick to have $\rho(MD^{-1}) < \frac{1}{2}$. This result correspond also on the max l_∞-norm of any vector reduced by our method,
iii) the average max l_∞-norm of any vector reduced by Babai's method with the same basis $D - M$.

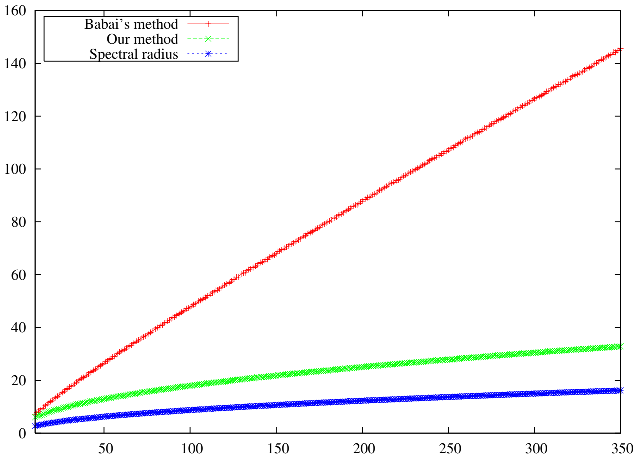

Fig. 3. Average l_∞-norm of signature-vector using different reduction method

Figure 3 has been obtained from the same data set used to generate Figure 2.

The important point of this result is that we can observe that the l_∞ norm of a reduced vector with this type of basis is in $O(n)$ after Babai's reduction and in $O(\sqrt{n})$ after our reduction. This difference clearly comes from the difference between l_2 and l_∞-norm.

To obtain a theoretical limit of this result, we use the result of German [18] which evaluates the limit when the dimension n grow of the spectral radius of a random matrix $A \in \mathbb{C}^{n,n}$ as $\rho(A) = \omega\sqrt{n}$ with $\omega^2 = \frac{1}{n^2}\sum_{i,j=0}^{i,j<n} A_{i,j}^2$.

This limit provides a good approximation of the spectral radius of a random matrix. Using this limit, we obtain the following result for a random matrix M taken in $\{-1, 0, 1\}^{n,n}$ an average approximation about $\rho(M) \sim \sqrt{\frac{2n}{3}}$. Finally, if we want $\rho(MD^{-1}) < \frac{1}{2}$ we need $\|D\|_\infty \sim 2\sqrt{\frac{2n}{3}} \sim 1.63\sqrt{n} = O(\sqrt{n})$. The theoretical approximation of $\rho(M)$ and $\|D\|_\infty$ obtained using German's theorem is very close to our own practical test given in Figure 3.

8 Conclusion and Open Problems

In this paper, we presented a new method of vector reduction under the l_∞-norm. Then, we constructed a signature scheme based on this norm. The resulting scheme seems very interesting, in terms of security, length and speed. We conclude this paper by providing two open research problems. Firstly, how to prove Conjecture 1 of $\rho < \frac{1}{2}$ and secondly, how to derive a formula to compute the average number of iterations in Algorithm 1 which is logarithmic in n as the test has demonstrated.

References

1. Agrell, E., Eriksson, T., Vardy, A., Zeger, K.: Closest point search in lattices. IEEE Transactions on Information Theory 48(8), 2201–2214 (2002)
2. Ajtai, M.: The shortest vector problem in l_2 is NP-hard for randomized reductions. In: 13th Annual ACM Symp. on the Theory of Computing, pp. 10–19 (1998)
3. Ajtai, M., Dwork, C.: A public-key cryptosystem with worst-case/average-case equivalence. In: 29th Annual ACM Symp. on the Th. of Comp., pp. 284–293 (1997)
4. Ajtai, M., Kumar, R., Sivakumar, D.: A sieve algorithm for the shortest lattice vector problem. In: 33rd Annual ACM Symp. on Th. of Comp., pp. 601–610 (2001)
5. Ajtai, M., Kumar, R., Sivakumar, D.: Sampling short lattice vectors and the closest lattice vector problem. In: IEEE CCC, pp. 53–57 (2002)
6. Babai, L.: On Lovász' lattice reduction and the nearest lattice point problem. Combinatorica 6(1), 1–13 (1986)
7. Bajard, J.-C., Imbert, L., Plantard, T.: Modular number systems: Beyond the Mersenne family. In: Handschuh, H., Hasan, M.A. (eds.) SAC 2004. LNCS, vol. 3357, pp. 159–169. Springer, Heidelberg (2004)
8. Blömer, J., Naewe, S.: Sampling Methods for Shortest Vectors, Closest Vectors and Successive Minima. In: Arge, L., Cachin, C., Jurdziński, T., Tarlecki, A. (eds.) ICALP 2007. LNCS, vol. 4596, pp. 65–77. Springer, Heidelberg (2007)
9. Boas, P.V.E.: Another NP-complete problem and the complexity of computing short vectors in lattices. TR 81-04, Math. Dept., Univ. of Amsterdam (1981)
10. Cai, J.-Y.: Some recent progress on the complexity of lattice problems. In: 14th Annual IEEE Conference on Computational Complexity, pp. 158–178 (1999)
11. Cassels, J.W.S.: An Introduction to The Geometry of Numbers. Springer, Heidelberg (1959)
12. Chen, W., Meng, J.: The hardness of the closest vector problem with preprocessing over ℓ_∞ norm. IEEE Trans on Inf Theory 52(10), 4603–4606 (2006)
13. Cohen, H.: A course in computational algebraic number theory. Graduate Texts in Mathematics, vol. 138. Springer, Heidelberg (1993)
14. Collatz, L.: Functional Analysis and Numerical Mathematics. Academic Press Inc., U.S. (1966)
15. Conway, J.H., Sloane, N.J.A.: Sphere Packings, Lattices and Groups. Springer, Heidelberg (1988)
16. Dinur, I.: Approximating SVP_∞ to within almost-polynomial factors is NP-Hard. In: Bongiovanni, G., Petreschi, R., Gambosi, G. (eds.) CIAC 2000. LNCS, vol. 1767, pp. 263–276. Springer, Heidelberg (2000)
17. Dinur, I., Kindler, G., Safra, S.: Approximating CVP to within almost polynomial factor is NP-hard. In: FOCS 1998, pp. 99–111 (1998)

18. Geman, S.: The spectral radius of large random matrices. The Annals of Probability 14(4), 1318–1328 (1986)
19. Gentry, C., Szydlo, M.: Cryptanalysis of the revised NTRU signature scheme. In: Knudsen, L.R. (ed.) EUROCRYPT 2002. LNCS, vol. 2332, pp. 299–320. Springer, Heidelberg (2002)
20. Goldreich, O., Goldwasser, S.: On the limits of non-approximability of lattice problems. In: the 30th Annual ACM Symp on Th of Computing, pp. 1–9 (1998)
21. Goldreich, O., Goldwasser, S., Halevi, S.: Public-key cryptosystems from lattice reductions problems. In: Kaliski Jr., B.S. (ed.) CRYPTO 1997. LNCS, vol. 1294, pp. 112–131. Springer, Heidelberg (1997)
22. Goldreich, O., Micciancio, D., Safra, S., Seifert, J.-P.: Approximating shortest lattice vectors is not harder than approximating closest lattice vectors. Information Processing Letters 71(2), 55–61 (1999)
23. Golub, G.H., Loan, C.F.V.: Matrix Computations. The Johns Hopkins University Press (1983)
24. Hanrot, G., Stehle, D.: Improved analysis of Kannan's shortest lattice vector algorithm. In: Menezes, A. (ed.) CRYPTO 2007. LNCS, vol. 4622, Springer, Heidelberg (2007)
25. Haviv, I., Regev, O.: Tensor-based hardness of the shortest vector problem to within almost polynomial factors. In: Thirty-nineth annual ACM symposium on Theory of computing, pp. 469–477 (2007)
26. Helfrich, B.: Algorithms to construct Minkowski reduced an Hermite reduced lattice bases. Theoretical Computer Science 41, 125–139 (1985)
27. Higham, N.J.: Estimating the matrix p-norm. Numerische Mathematik 62, 539–556 (1992)
28. Hoffstein, J., Howgrave-Graham, N., Pipher, J., Silverman, J.H., Whyte, W.: NTRUSign: Digital signatures using the NTRU lattice. In: Joye, M. (ed.) CT-RSA 2003. LNCS, vol. 2612, pp. 122–140. Springer, Heidelberg (2003)
29. Hoffstein, J., Pipher, J., Silverman, J.H.: NTRU: A ring-based public key cryptosystem. In: Buhler, J.P. (ed.) ANTS 1998. LNCS, vol. 1423, pp. 267–288. Springer, Heidelberg (1998)
30. Householder, A.S.: The theory of matrices in numerical analysis. Blaisdell Pub. Co., New York (1964)
31. Kannan, R.: Improved algorithms for integer programming and related lattice problems. In: Proceedings of the Fifteenth Annual ACM Symposium on Theory of Computing, Boston, Massachusetts, April 1983, pp. 193–206 (1983)
32. Kannan, R., Bachem, A.: Polynomial algorithms for computing the Smith and Hermite normal forms of an integer matrix. SIAM J. of Comp 8(4), 499–507 (1979)
33. Khot, S.: Hardness of approximating the shortest vector problem in high l_p norms. In: The 44th Annual IEEE Symposium on FOCS, pp. 290–297 (2003)
34. Krasnosel'Skii, M.A., Vainikkov, G.M., Zabreiko, P.P., Rutitskii, Y.B., Stetsenko, V.Y.: Approximate solution of operator equations (1972)
35. Lenstra, A.K., Lenstra, H.W., Lovász, L.: Factoring polynomials with rational coefficients. In: Mathematische Annalen, Springer-Verlag, vol. 261, pp. 513–534. Springer, Heidelberg (1982)
36. Lovász, L.: An Algorithmic Theory of Numbers, Graphs and Convexity. In: CBMS-NSF Regional Conference Series in Applied Mathematics, vol. 50, SIAM Publications, Philadelphia (1986)
37. McEliece, R.J.: A public-key cryptosystem based on algebraic coding theory. Deep Space Network Progress Report 44, 114–116 (1978)

38. Micciancio, D.: Improving lattice based cryptosystems using the Hermite normal form. In: Silverman, J.H. (ed.) CaLC 2001. LNCS, vol. 2146, pp. 126–145. Springer, Heidelberg (2001)
39. Micciancio, D., Warinschi, B.: A linear space algorithm for computing the Hermite normal form. In: Intl. Symp. on Symb. Alg. Comp., pp. 231–236 (2001)
40. Minkowski, H.: Geometrie der Zahlen. B.G. Teubner, Leipzig (1896)
41. Nguyen, P.Q.: Cryptanalysis of the Goldreich-Goldwasser-Halevi cryptosystem from crypto 1997. In: Wiener, M.J. (ed.) CRYPTO 1999. LNCS, vol. 1666, pp. 288–304. Springer, Heidelberg (1999)
42. Nguyen, P.Q., Regev, O.: Learning a parallelepiped: Cryptanalysis of GGH and NTRU signatures. In: Vaudenay, S. (ed.) EUROCRYPT 2006. LNCS, vol. 4004, pp. 271–288. Springer, Heidelberg (2006)
43. Nguyen, P.Q., Stehlé, D.: Floating-point LLL revisited. In: Cramer, R.J.F. (ed.) EUROCRYPT 2005. LNCS, vol. 3494, pp. 215–233. Springer, Heidelberg (2005)
44. Nguyen, P.Q., Stehlé, D.: LLL on the average. In: Hess, F., Pauli, S., Pohst, M. (eds.) ANTS 2006. LNCS, vol. 4076, pp. 238–256. Springer, Heidelberg (2006)
45. Nguyen, P.Q., Stern, J.: The two faces of lattices in cryptology. In: Silverman, J.H. (ed.) CaLC 2001. LNCS, vol. 2146, pp. 146–180. Springer, Heidelberg (2001)
46. Peikert, C.: Limits on the hardness of lattice problems in ℓ_p norms. In: Twenty-Second Annual IEEE Conference on Computational Complexity, pp. 333–346 (2007)
47. Regev, O.: Lattice-based cryptography. In: Dwork, C. (ed.) CRYPTO 2006. LNCS, vol. 4117, pp. 131–141. Springer, Heidelberg (2006)
48. Regev, O., Rosen, R.: Lattice problems and norm embeddings. In: Thirty-eighth annual ACM symposium on Theory of computing, pp. 447–456 (2006)
49. Schnorr, C.-P.: A hierarchy of polynomial time lattice basis reduction algorithms. Theoretical Computer Science 53(2–3), 201–224 (1987)
50. Schnorr, C.-P.: A more efficient algorithm for lattice basis reduction. Journal of Algorithms 9(1), 47–62 (1988)
51. Schnorr, C.-P.: Block Korkin-Zolotarev bases and successive minima (1996)
52. Schnorr, C.-P.: Fast LLL-type lattice reduction. Information and Computation 204(1), 1–25 (2006)
53. Szydlo, M.: Hypercubic lattice reduction and analysis of GGH and NTRU signatures. In: Biham, E. (ed.) EUROCRYPT 2003. LNCS, vol. 2656, pp. 433–448. Springer, Heidelberg (2003)
54. Varga, R.S.: Matrix Iterative Analysis. Prentice-Hall, Englewood Cliffs (1962)
55. Wilkinson, J.H.: The algebraic eigenvalue problem. Oxford University Press, Inc., New York (1965)