Compact Lattice Signatures

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: Compact Lattice-based Signatures, Short Signatures, Short Keys, Provable Security.

Abstract: Lattice-based signature schemes have seen many improvements in the past few years with recent attempts (Güneysu et al., 2012; Ducas et al., 2013; Ducas et al., 2014; Lyubashevsky, 2016; Ducas et al., 2017) to bring lattice-based signature schemes at par with the traditional number-theoretic signature schemes. However, the trade-off between the signature size and the key size, time for a signature generation, and the practical and provable security is not necessarily the optimal. We propose a compact lattice-based signature scheme with key-size and signatures of order n, where n is the dimension of the lattice. The proposed signature scheme has faster algorithms for key generation, signing, and verification than the existing schemes. The proposed scheme is simple and is competitive with the other post-quantum signature schemes.

1 INTRODUCTION

The lattice-based public key cryptographic primitives like encryption schemes (Hoffstein et al., 1998; Lyubashevsky et al., 2013) and digital signatures (Ducas et al., 2013; Lyubashevsky, 2016; Ducas et al., 2017; Ducas et al., 2014), is almost close to being used as an alternative to the traditional schemes in respect to size and computation time. Most of the lattice-based cryptographic primitives are constructed based on the average-case hardness of the SIS (or Ring SIS) and LWE (or Ring LWE) problem. It was shown these problems enjoy average-case to worst-case reduction making it a potential alternative of number theoretic schemes.

Though the Ring LWE and Ring SIS provide hardness based on the worst-case to average-case reduction, the relatively large number of ring operations needed for constructing schemes affects the efficiency of these schemes. Generally, in the schemes based on the Ring SIS or the Ring LWE set-up, $k(\geq 2)$ ring elements are used for both the public key and the private key, and we need to perform ring operations on k elements for signature and verification. Making the private key with one ring element instead of k would make the signature schemes faster, reducing the computational cost by almost half. There are many roadblocks to construct signatures with only one ring element as the secret key. We have proposed such a signature scheme in this paper.

1.1 Related Work

The recently proposed lattice-based signature schemes can be broadly distinguished into two categories. The schemes in the first category follow the hash-and-sign paradigm and use the GPV sampling (Gentry et al., 2008) procedure to produce basis that is used as a trapdoor for the key generation of this lattice-based signatures. But, the sampling methods used in these schemes are quite complex and have either expensive runtime or low output quality, making these relatively unsuitable for practical implementations (Micciancio and Peikert, 2012).

The schemes in the second category follow the Fiat-Shamir framework. In (Lyubashevsky, 2009), the first scheme has been proposed based on this framework and its security depends on the hardness of solving the Ring SIS problem. Later in (Lyubashevsky, 2012; Güneysu et al., 2012; Ducas et al., 2013; Lyubashevsky, 2016), signature schemes have been proposed whose security depend on the hardness of SIS problem or Ring SIS problem. More recently, Ducas et al. have proposed another lattice-based signature scheme, Dilithium (Ducas et al., 2017)m which is an improvement of (Güneysu et al., 2012) and relies on the hardness of module SIS (or module LWE) problem (thus lowering the dimension of the lattice used by a factor on the security parameter without affecting the security) which is a generalized version of the Ring SIS (or ring LWE) problem.

Das, D. and Saraswat, V.
Compact Lattice Signatures.
DOI: 10.5220/0006861604900495
In *Proceedings of the 15th International Joint Conference on e-Business and Telecommunications (ICETE 2018) - Volume 2: SECRYPT*, pages 490-495
ISBN: 978-989-758-319-3

1.2 Our Contribution

In this paper, we present a lattice-based signature scheme which is based on the Fiat-Shamir framework (Abdalla et al., 2002; Fiat and Shamir, 1986; Pointcheval and Stern, 2000).

Let us discuss the proposed scheme in brief here. The details of the scheme is given in Section 3. The secret key g is a polynomial chosen uniformly at random, of degree $n-1$ with binary coefficients. The public keys have two polynomials, h is chosen uniformly at random from the ring $\mathbb{Z}_q[x]/(x^n+1)$ and make $h' = h \star g + 2e \mod q$, where e is a binary polynomial of degree $n-1$ chosen uniformly at random, $\star$ is the ring product called the convolution product. The polynomial e is used to hide the information of g from h and h'. The polynomial h can be shared among all users, but the polynomial h' is individual. To sign a message M, the signer chooses a masking parameter y, whose coefficients are chosen uniformly at random from a set with norm less than q. Then he computes $c \in \mathbb{Z}_q[x]/(x^n+1)$, where $c \leftarrow H((h \star y + M \mod q) \mod 2)$, and computes the potential signature as $z = g \star c + y$ (there is no reduction modulo q in this step). The polynomial z, along with c, will then be output as the signature based on some criteria with the goal to keep the distribution of (z,c) independent of the secret polynomial g which is termed as Rejection sampling. The verification of the scheme is done by checking if the coefficients of z is in the pre-defined bound, and $c = H((h \star z - h' \star c + M \mod q) \mod 2)$

The inclusion of the modulo 2 operation has a potential problem. The security proof from (Lyubashevsky, 2012) is no longer valid. The security proof of (Lyubashevsky, 2012) has the foundation view of many other lattice-based signature schemes like (Güneysu et al., 2012; Ducas et al., 2013; Lyubashevsky, 2016; Ducas et al., 2017). In particular, it is no longer clear how to outline for the security proof as done in (Lyubashevsky, 2012). But we show that this problem can be overcome. We will show that forging a signature is as hard as solving the "super NTRU encrypt problem" (Definition 2).

1.3 Comparison with Other Lattice-based Signature Schemes

In this subsection, we compare the scheme proposed in this paper with other competitive lattice-based schemes which follow the Fiat-Shamir framework.

This work can be seen as an improvement of Guneysu et al.' s (Güneysu et al., 2012). Further, the idea presented in this paper can also be used to improve results of Dilithium (Ducas et al., 2017) which is submitted to the NIST call for post-quantum standards. Let us discuss in brief about the difference between the two schemes. In (Güneysu et al., 2012), the secret key is the pair of polynomials (g,e) drawn from a uniform distribution with a small norm. Both the polynomials play the same importance in the entire scheme. We have shown in this paper that only one polynomial is enough to define a secure signature scheme. In our signature scheme, we have also chosen two polynomials g, e, but the secret is only one polynomial g. The polynomial e is never used in the signing and verification algorithm except for a comparison test in the signing algorithm, which is needed for the correctness of the scheme. The main goal of the polynomial e is to hide information of the secret key g from the public keys. This idea has also helped us to reduce (by a factor two) the number of masking parameters needed to generate for the signing algorithm.

This small improvement has many effects on the signature output over (Güneysu et al., 2012). Guneysu et al. have used compression algorithm in (Güneysu et al., 2012) to decrease the signature size which is still larger than our scheme ($O(n)$ improvement). Further, the scheme is space efficient reducing the secret key size by a factor two. Also, the signature scheme is faster than (Güneysu et al., 2012).

The most efficient lattice-based scheme till date, BLISS (Ducas et al., 2013), uses discrete Gaussian sampling for generating the masking parameter and accurate rejection sampling to create compact signatures. But it has been recently reported (Bruinderink et al., 2016; Pessl, 2016) that these schemes are vulnerable to the side-channel attacks possible due to the usage of discrete Gaussian sampling, resulting in the complete leakage of the secret key. Further, the NTRU problem, on which BLISS relies, has been shown (Kirchner and Fouque, 2017) to be not as hard as previously assumed for large parameters, though the NTRU problem with current parameters is not affected by this attack.

2 PRELIMINARIES

In this section we present the notations used in this paper, hardness assumptions, and concrete instantiation of the proposed scheme.

2.1 Notations

For a distribution $\mathcal{D}$, we use the notation $x \xleftarrow{\$} \mathcal{D}$ to mean that x is chosen according to the distribution $\mathcal{D}$. If S is a set, then $x \xleftarrow{\$} S$ means that x is chosen uni-

formly at random from S. Throughout the paper, we will consider n is a power of 2 and q is a prime integer of the form $q = 1 \mod 2n$. Any element in $\mathbb{Z}_q$ is represented by integers within the interval $[-\frac{q-1}{2}, \frac{q-1}{2}]$. Whenever dealing with elements that are in $\mathbb{Z}_q$, we will assume that all operations in which they are involved with a reduction modulo q. We will denote $R = \mathbb{Z}[x]/(x^n+1)$ to be the ring of polynomials in $\mathbb{Z}[x]$ modulo x^n+1. By R_q we represent the elements of R whose coefficients are in $\mathbb{Z}_q$. In particular, $R_2 \simeq \mathbb{Z}_2^n$ denotes the polynomials of degree less than n and binary coefficients (that is, in $\{0,1\}$) and $R_3 \simeq \mathbb{Z}_3^n$ denotes the polynomials of degree less than n and coefficients in $\{0,1,-1\}$. Let $D_k \subset R$ such that it consists of all polynomials of degree less than n and coefficients are bounded by $k(< q)$. All logarithms used in this paper are base 2.

2.2 Hardness Assumptions

The Ring SIS problem may be put as: For a subset S of R_q, and two elements s_1 and s_2 chosen uniformly from S, an adversary $\mathcal{A}$ is given an ordered pair of polynomials $(a,t) \in R_q \times R_q$, where a is chosen uniformly from R_q and $t = a \star s_1 + s_2$, and is asked to find s_1' and s_2' from S such that $a \star s_1' + s_2' = t$.

If S is such that the norms of s_1 and s_2 are sufficiently large, that is, if $\|s_i\|_\infty > \sqrt{q}$, then there are possibly many solutions (s_1', s_2') such that $t = a \star s_1' + s_2'$ and then the problem of finding any (s_1, s_2) can be connected to worst-case lattice problems in ideal lattices (Lyubashevsky and Micciancio, 2006; Peikert and Rosen, 2006).

On the other hand, when S is such that the norm of s_1 and s_2 are less than $\sqrt{q}$, then with high probability there is a unique solution of s_1, s_2. This is an instance of Ring LWE. It has been shown (Lyubashevsky et al., 2013) that if we choose s_i from discrete Gaussian distribution instead of uniform distribution then solving the search problem (recovering s_i from (a,t)) is as hard as the worst-case lattice problems (like approximate Shortest Independent Vectors Problem) on x^n+1 cyclic lattices (lattices that correspond to some ideals in the ring $\mathbb{Z}[x]/(x^n+1)$) using quantum algorithm. Further, deciding (a,t) from truly random elements of $R_q \times R_q$ is as hard as the search problem (Lyubashevsky et al., 2013) using a classical reduction. Unfortunately, in the later reduction, the modulus q must be a prime integer. The reduction from the search version is more general, and it takes place for any modulus q. Recently, in (Peikert et al., 2017), Peikert and Regev have shown that the decisional problem is hard even when q is not a prime integer. It is shown in (Peikert et al., 2017) that there is a polynomial time quantum reduction from worst-case lattice problems to decisional Ring LWE for any modulus.

Till date, there is no known algorithm that takes advantage of the fact that the distribution of s_i is uniform (instead of Gaussian) and consists of elements from R_2. In this paper, we define our signature scheme based on the presumed hardness of the search Ring LWE problem with particularly "aggressive" parameters.

We recall below the NTRU encrypt problem remains hard even after 20 years of cryptanalytic efforts.

Definition 1 (NTRU Encrypt Problem)**.** Given an NTRU public key $H \in R_p$ and the encryption of the message m, $E = r \star H + m \mod p$, where r is a small random polynomial and p is an appropriately chosen integer, find the message m. [1]

The hardness part of the NTRU encrypt problem is that the polynomial E is pseudo-random in R_p. We define an extension of the NTRU encrypt problem below.

Definition 2 (Super NTRU Encrypt Problem)**.** Given $(a, t(= a \star s + 2e)) \in R_q \times R_q$, where $a \xleftarrow{\$} R_q$, $s \xleftarrow{\$} R_2$ and $e \xleftarrow{\$} R_2$, find the secret s.

The super NTRU encrypt problem can be seen as the encryption of the message $2e$, and the pseudo-random public key has been replaced with the truly uniform random one, and thus, harder than the NTRU encrypt problem. In fact, the super NTRU encrypt problem is equivalent to the binLWE problem and which has been used in previously proposed schemes (Brakerski et al., 2013; Fan and Vercauteren, 2012). It is still unknown how much the super NTRU encrypt problem is easier than the standard LWE problem. The secret and error in the super NTRU encrypt problem are sufficiently small compared to the standard LWE problem. It is known that solving LWE problems with smaller secrets (secrets following the error distribution) are not easier than regular LWE problems with arbitrary secrets (Applebaum et al., 2009). Further, there is a reduction from regular LWE to super NTRU encrypt problem which has an expansion factor of $\log q$ in the dimension (Brakerski et al., 2013). Nevertheless, there is no attack as such in our set of parameters.

[1] In the original NTRU scheme, the ring was $\mathbb{Z}_p[x]/(x^n-1)$, but lately, researchers have also used $\mathbb{Z}_p[x]/(x^n+1)$ when n is a power of 2. Indeed, the latter choice seems at least as secure

3 PROPOSED SIGNATURE SCHEME

In this section, we present our proposed signature scheme and its concrete instantiation.

3.1 Signature Scheme

Key Generation: We generate a polynomial g which is chosen uniformly at random from R_2. That is, g is a polynomial in R, whose coefficients are from the set $\{0,1\}$. Then we generate another polynomial h uniformly at random from R_q. We then set $h' = h \star g + 2e \mod q$, where e is a polynomial with coefficients chosen uniformly at random from the set $\{0,1\}$. The signing key is the polynomial g and the verification key is the pair of polynomials (h,h').

Signature: To sign a message $M \in R_2 \simeq \{0,1\}^n$, the signer first generates a y randomly from D_k and computes $c = H((h \star y + M \mod q) \mod 2)$. The signer checks if $\|h \star y - 2e \star c + M\|_\infty \leq q/2$ and if not, it chooses another y and repeats. Then the signer computes $z = g \star c + y$ and checks if $\|z\|_\infty \leq k - 32$ and if not, it chooses another y and repeats. This is the rejection sampling step in our signature scheme. Finally, the signer outputs (z,c). The effect of k plays a vital role in our signature scheme. If the value of k is too small then the probability that $z \in D_{k-32}$ is too small.

Verification: To verify the correct signature, the verifier checks whether z is small, that is, $\|z\|_\infty \leq k - 32$ and if $c = H((h \star z - h' \star c + M \mod q) \mod 2)$. Only if both the conditions are satisfied, the verifier confirms it to be the valid signature.

3.2 Concrete Instantiations of the Proposed Scheme

We now give some concrete instantiations of our proposed signature scheme. The security of the scheme depends on the hardness of solving the super NTRU encrypt problem. We have set our parameters based on the parameters suggested in (Gama and Nguyen, 2008), (Chen and Nguyen, 2011) and (Micciancio and Regev, 2009). Forging signature in our scheme using lattice reduction mechanism needs the root Hermite factor at least $1.0004, 1.0002, 1.0001$ for $n = 512, 1024, 2048$ respectively which is out of scope by the known lattice reduction techniques.

Recent progress in BKW-style algorithms for solving LWE has given rise to variants (Albrecht et al., 2015; Albrecht, 2017) of the dual-lattice attack against LWE in the presence of an unusually short secret. These variants scale the exponent of the dual-lattice attack by a factor of $2L/(2L+1)$ and halve the dimension n of the lattice under consideration at a multiplicative cost of $2h$ operations, when $\log q = \Theta(L \log n)$, h is the constant hamming weight of the secret and L is the maximum depth of supported circuits. Applying these techniques to parameter sets $n = 1024$ and $\log q \approx 47$ suggested for a promised 80 bits of security by the homomorphic encryption libraries SEAL v2.0 and HElib, yields revised security estimates of 68 bits and 62 bits respectively.

These attacks call for a revision of the suggested parameters for many of the cryptographic schemes based the hardness of LWE. We suggest three sets of parameters, the first two to keep consistency with the suggested parameters of the existing schemes and the third one to maintain security against the latest attacks. It may be noted that even with larger parameters, our scheme is still practical due to the small size of the keys and the resulting efficiency.

The secret key consists of a polynomial with binary coefficients of degree less than n. So, the size of the secret key is n bits. The public key of the signature scheme is two polynomials (h,h') of degree less than n and coefficients in $\mathbb{Z}_q$. If trusted randomness is available, then everyone can share the same h which considerably lowers the public key size because the public key h can be included in the signing and verification algorithms. So, the public key size depends on the polynomial h' and the size is approximately $n \log q$ bits. The signature size will depend mainly on a polynomial of degree less than n and coefficients in the range $[-k+32, k-32]$. So, the approximate size of the signature output is $n \log(2(k-32)+1)$ bits.

4 SECURITY ANALYSIS OF THE PROPOSED SIGNATURE SCHEME

In this section, we analyze the security of the proposed signature scheme.

Theorem 3. *Any polynomial adversary knowing the verification key (h,h'), cannot recover the private signing key g based on the hardness of the super NTRU encrypt problem.*

Proof. By definition, $h' = h \star g + 2e \mod q$, where g and e are polynomials chosen uniformly at random from R_2 and h is a polynomial chosen uniformly at random from R_q. Hence, recovering the signing key

Table 1: Parameter definition of the proposed scheme.

Parameters	Definitions	Sample Instantiations		
n	An integer, power of 2	512	1024	2048
q	A prime number	$\approx 2^{15}$	$\approx 2^{16}$	$\approx 2^{17}$
k		2^{13}	2^{14}	2^{15}
R	The ring $\mathbb{Z}[x]/(x^n+1)$			
R_q	The ring $\mathbb{Z}_q[x]/(x^n+1)$, centre-lifted coefficients			
H	A random oracle that maps elements of R_2 to $\{f \in R_3 : \|f\|_1 \leq 32\}$			
μ	Expected number of times the signing algorithm need to perform to get a valid signature	7	7	7
$\|sig\|$	The signature size in bits $\approx n\log(2(k-32)+1)$	7,165	15,357	32,765
$\|sec\|$	The signing key size in bits $\approx n$	512	1024	2048
$\|pub\|$	The verification key size in bits $\approx n\log(q)$	7,680	16,384	34,816

g given the verification key (h,h') is exactly the super NTRU encrypt problem as stated in Definition 2. Thus, the result follows directly from the assumed hardness of the super NTRU encrypt problem. □

Theorem 4. *The proposed scheme is strongly unforgeable based on the hardness of solving the super NTRU encrypt problem.*

Proof. Basically, we will show that signature forgery is equivalent to key recovery for our scheme. That is, if there exists an adversary $\mathcal{F}$ who can forge a signature of our scheme in time t, then there exists an algorithm which can recover the secret key of the scheme in time $t + \mathsf{poly}(n)$ just from the verification key.

Given (h,h'), we challenge the forger to generate a forgery. We answer its hash queries and signature queries as follows.

We maintain a list L, which is initially empty, with all the answered hash queries. To answer a hash query on input $x \in R_2$, we look up the list L to check if x has already been queried. If yes, we answer as in the list. Otherwise, we randomly choose a c at uniform from the range of H, add the pair (x,c) to the list L and respond to the forger with $c := H(x)$.

To answer a signature query on a message M, we randomly choose a vector y and check the list L to see if

$$x := (h \star y + M \mod q) \mod 2$$

has already been queried. If yes, we choose another y and repeat. Then we choose vectors (z,c) such that $z = g \star c + y$, $\|z\|_\infty \leq k-32$, $\|c\|_1 \leq 32$.

The signature z can be generated in the above way because its distribution does not rely on the secret key g, and a simulator can sign messages by programming the random oracle.

Then we add the pair (x,c) to the list L and respond to the forger with (c,z) as the signature on the message M.

Finally, we use the rewinding technique as in forking lemma to use the forger to get two distinct signatures (c,z) and (c',z') on the same message M. Then, by the definition of the signature in our scheme,

$$z = g \star c + y \quad \text{and} \quad z' = g \star c' + y$$

so that

$$z - z' = g \star (c - c').$$

From the last equation, $c = c'$ implies $z = z'$, and $z = z'$ implies $c = c'$[2]. Now, recovering g from $z - z' = g \star (c - c')$ can be done in $\mathsf{poly}(n)$ time, and thus, solving the super NTRU encrypt problem. □

5 DISCUSSIONS AND FUTURE WORK

One future direction would be to make our signature scheme based on the hardness of the decisional version of the super NTRU encrypt problem. The decisional version of the super NTRU encrypt problem can be defined as following: distinguish the case when (a,t) is uniform in $R_q \times R_q$ and when $t = a \star s + 2e \mod q$. The signature scheme in (Güneysu et al., 2012) is on the hardness of the decisional super NTRU encrypt problem which has been termed as the Decisional Compact Knapsack problem in the paper.

Further, given the recent advances in the attacks on the LWE problem, modification of our scheme to

[2]The statement "$z = z'$ implies $c = c'$" is only true if g is invertible, which is true with high probability

provide desirable security with smaller n. We believe that a calibration of the parameters such as q, κ and λ should be able to avoid these attacks while maintaining smaller sizes.

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

Abdalla, M., An, J. H., Bellare, M., and Namprempre, C. (2002). From identification to signatures via the Fiat-Shamir transform: Minimizing assumptions for security and forward-security. In *EUROCRYPT*, LNCS, pages 418–433. Springer.

Albrecht, M. R. (2017). On dual lattice attacks against small-secret LWE and parameter choices in HElib and SEAL. In *EUROCRYPT*, volume 10211 of *LNCS*, pages 103–129. Springer.

Albrecht, M. R., Player, R., and Scott, S. (2015). On the concrete hardness of learning with errors. *Journal of Mathematical Cryptology*, 9(3):169–203.

Applebaum, B., Cash, D., Peikert, C., and Sahai, A. (2009). Fast cryptographic primitives and circular-secure encryption based on hard learning problems. In *CRYPTO*, LNCS, pages 595–618. Springer.

Brakerski, Z., Langlois, A., Peikert, C., Regev, O., and Stehlé, D. (2013). Classical hardness of learning with errors. In *STOC*, pages 575–584. ACM.

Bruinderink, L. G., Hülsing, A., Lange, T., and Yarom, Y. (2016). Flush, gauss, and reload–a cache attack on the bliss lattice-based signature scheme. In *CHES*, LNCS, pages 323–345. Springer.

Chen, Y. and Nguyen, P. Q. (2011). BKZ 2.0: Better lattice security estimates. In *ASIACRYPT*, LNCS, pages 1–20. Springer.

Ducas, L., Durmus, A., Lepoint, T., and Lyubashevsky, V. (2013). Lattice signatures and bimodal gaussians. In *CRYPTO*, volume 8042 of *LNCS*, pages 40–56. Springer.

Ducas, L., Lepoint, T., Lyubashevsky, V., Schwabe, P., Seiler, G., and Stehlé, D. (2017). CRYSTALS - dilithium: Digital signatures from module lattices. *IACR Cryptology ePrint Archive*, 2017:633.

Ducas, L., Lyubashevsky, V., and Prest, T. (2014). Efficient identity-based encryption over NTRU lattices. In *ASIACRYPT*, volume 8874 of *LNCS*, pages 22–41. Springer.

Fan, J. and Vercauteren, F. (2012). Somewhat practical fully homomorphic encryption. *IACR Cryptology ePrint Archive*, 2012:144.

Fiat, A. and Shamir, A. (1986). How to prove yourself: Practical solutions to identification and signature problems. In *CRYPTO*, volume 263 of *LNCS*, pages 186–194. Springer.

Gama, N. and Nguyen, P. Q. (2008). Predicting lattice reduction. In *EUROCRYPT*, LNCS, pages 31–51. Springer.

Gentry, C., Peikert, C., and Vaikuntanathan, V. (2008). Trapdoors for hard lattices and new cryptographic constructions. In *STOC*, pages 197–206. ACM.

Güneysu, T., Lyubashevsky, V., and Pöppelmann, T. (2012). Practical lattice-based cryptography: A signature scheme for embedded systems. In *CHES*, volume 7428 of *LNCS*, pages 530–547. Springer.

Hoffstein, J., Pipher, J., and Silverman, J. H. (1998). NTRU: A ring-based public key cryptosystem. In *ANTS*, volume 1423 of *LNCS*, pages 267–288. Springer.

Kirchner, P. and Fouque, P. (2017). Revisiting lattice attacks on overstretched NTRU parameters. In *EUROCRYPT*, volume 10210 of *LNCS*, pages 3–26. Springer.

Lyubashevsky, V. (2009). Fiat-Shamir with aborts: Applications to lattice and factoring-based signatures. In *ASIACRYPT*, volume 5912 of *LNCS*, pages 598–616. Springer.

Lyubashevsky, V. (2012). Lattice signatures without trapdoors. In *EUROCRYPT*, volume 7237 of *LNCS*, pages 738–755. Springer.

Lyubashevsky, V. (2016). Digital signatures based on the hardness of ideal lattice problems in all rings. In *ASIACRYPT*, LNCS, pages 196–214. Springer.

Lyubashevsky, V. and Micciancio, D. (2006). Generalized compact knapsacks are collision resistant. In *ICALP*, volume 4052 of *LNCS*, pages 144–155. Springer.

Lyubashevsky, V., Peikert, C., and Regev, O. (2013). On ideal lattices and learning with errors over rings. *Journal of ACM*, 60(6):43:1–43:35.

Micciancio, D. and Peikert, C. (2012). Trapdoors for lattices: Simpler, tighter, faster, smaller. In *EUROCRYPT*, volume 7237 of *LNCS*, pages 700–718. Springer.

Micciancio, D. and Regev, O. (2009). Lattice-based cryptography. In *Post-Quantum Cryptography*, pages 147–191. Springer.

Peikert, C., Regev, O., and Stephens-Davidowitz, N. (2017). Pseudorandomness of ring-lwe for any ring and modulus. In *STOC*, pages 461–473. ACM.

Peikert, C. and Rosen, A. (2006). Efficient collision-resistant hashing from worst-case assumptions on cyclic lattices. In *TCC*, volume 3876 of *LNCS*, pages 145–166. Springer.

Pessl, P. (2016). Analyzing the shuffling side-channel countermeasure for lattice-based signatures. In *INDOCRYPT*, LNCS, pages 153–170. Springer.

Pointcheval, D. and Stern, J. (2000). Security arguments for digital signatures and blind signatures. *Journal of Cryptology*, 13(3):361–396.