Noname manuscript No.
(will be inserted by the editor)

Practical MP-LWE-based encryption balancing security-risk vs. efficiency

R. Steinfeld · A. Sakzad · R.K. Zhao

the date of receipt and acceptance should be inserted later

Abstract Middle-Product Learning With Errors (MP-LWE) is a variant of the LWE problem introduced at CRYPTO 2017 by Rosca et al [RSSS17]. *Asymptotically*, the theoretical results of [RSSS17] suggest that MP-LWE gives lattice-based public-key cryptosystems offering a 'security-risk vs. efficiency' trade-off: higher performance than cryptosystems based on unstructured lattices (LWE problem) and lower risk than cryptosystems based on structured lattices (Polynomial/Ring LWE problem). However, although promising in *theory*, [RSSS17] left the *practical* implications of MP-LWE for lattice-based cryptography unclear.

In this paper, we show how to build *practical* public-key cryptosystems with strong security guarantees based on MP-LWE. On the implementation side, we present optimised fast algorithms for computing the *middle-product* operation over polynomial rings $\mathbb{Z}_q[x]$, the dominant computation for MP-LWE-based cryptosystems. On the security side, we show how to obtain a nearly *tight* security proof for MP-LWE from the hardest Polynomial LWE problem over a large family of rings, improving on the loose reduction of [RSSS17]. We also show and analyze an optimised cryptanalysis of MP-LWE that narrows the complexity gap to the above security proof. To evaluate the practicality of MP-LWE, we apply our results to construct, implement and optimise parameters for a practical MP-LWE-based public-key cryptosystem, Titanium, and compare its benchmarks to other lattice-based systems. Our results show that MP-LWE offers a new 'security-risk vs. efficiency' trade-off in lattice-based cryptography in *practice*, not only asymptotically in theory.

Keywords: middle-product learning with errors (MP-LWE), lattice-based cryptography, quantum-resistant cryptography, public-key encryption, KEM, cryptography implementation.

MSC: 68P25.

Ron Steinfeld, Amin Sakzad, and Raymond K. Zhao,
Faculty of Information Technology, Monash University, Australia
E-mail: {ron.steinfeld,amin.sakzad,raymond.zhao}@monash.edu

1 Introduction

1.1 Background

Until quite recently, there have been two quite distinct approaches to lattice-based cryptography. The first 'low security risk' oriented approach is based on unstructured lattices, using the Learning With Errors (LWE) problem [Reg05]. The lack of any special structure in the LWE problem allows it to enjoy strong security guarantees, such as a security reduction from the worst-case hardness of lattice problems over all lattices [Reg05]. Yet the drawback of this approach is the large matrices involves and consequent relatively heavy computational cost of the resulting cryptosystems, such as Frodo [BCD+16] and FrodoKEM [ABD+17].

The second 'high performance' oriented approach is based on structured lattices, specifically using the Polynomial/Ring Learning With Errors (PLWE/RLWE) problems [SSTX09, LPR10, BV11]. The polynomial rings $R_{q,f} = \mathbb{Z}_q[x]/(f(x))$ for a fixed ring polynomial $f(x)$ (e.g. cyclotomic polynomials of the form $f(x) = x^n + 1$ for n a power of 2) underlying this approach permit the use of succinct matrices and fast polynomial arithmetic based on the number theoretic transform (NTT), leading to low computational and storage costs of the resulting cryptosystems, such as New Hope [ADPS16]. However, the high efficiency of this approach comes with a higher security risk. Specifically, the ring polynomial $f(x)$ is fixed at design time and the cryptosystem security relies on security of PLWE^f (i.e., the PLWE problem in the specific ring $R_{q,f}$ defined by f). Unfortunately, the dependence of the hardness of PLWE^f on the choice of f is not very well understood. For example, it is known that certain choices of f lead to either an insecure PLWE^f with small errors relative to the underlying lattice geometry [EHL14, ELOS15, CIV16, Pei16], or to efficient subexponential approximation factor quantum algorithms for the related approximate shortest vector problem in ideals of the polynomial ring $\mathbb{Z}[x]/(f(x))$ (for certain cyclotomic f's) [CDPR16, CDW16]. Thus, there is a security risk in fixing f today in a cryptosystem, as future attacks on PLWE^f for the potentially weak f used in the cryptosystem may be discovered. We remark that the Module Polynomial LWE problem (as used in [BDK+17, DKRV17]) also has a similar potential risk, since the module is defined over a fixed polynomial ring, and attacks on the ring could translate into attacks on the module [AD17]. [1]

Recently, the theoretical foundations for a new third 'intermediate' approach was introduced [RSSS17], seeking to achieve an intermediate point in the 'security-risk vs. efficiency' trade-off curve, sitting in between the unstructured lattice (LWE) first approach and the structured lattice (PLWE) second approach above. To obtain a lower security risk than reliance on $\mathsf{PLWE}^f/\mathsf{RLWE}^f$ (or Module-RLWE) over a single fixed ring $\mathbb{Z}_q[x]/(f(x))$, this approach, initiated by Lyubashevsky [Lyu16] for the design of digital signatures and extended by Rosca et al. [RSSS17] for design of public-key encryption, aims at problems that are provably as hard as PLWE^f for the hardest f in a *large family of polynomials*, to hedge against the weakness of specific polynomial rings, while achieving a better efficiency than schemes based on

[1] We remark that the risk for Module PLWE may be lower than for PLWE since existing 'direct' attacks on Module Polynomial LWE problem require a larger module rank to be solved than for attacks on the PLWE instance. But [AD17] shows at least asymptotically that, at the cost of polynomially-larger error parameter, a poly-time PLWE attack over the ring trnslates into a poly-time Module PLWE attack over the same ring.

unstructured LWE, by working over polynomial rings of the form $\mathbb{Z}_q[x]$ (with no fixed ring modulus f). In particular, [RSSS17] introduced a new variant of LWE over the ring $\mathbb{Z}_q[x]$ called *Middle-Product* LWE (MP-LWE), and gave a polynomial time security reduction from (decision) PLWE^f to (decision) MP-LWE of parameter n, for every monic f of degree n whose constant coefficient is coprime with q. The middle-product operation underlying MP-LWE consists of a multiplication in the ring $\mathbb{Z}_q[x]$ followed by a truncation of coefficients, keeping only the middle coefficients.

While the work of [RSSS17] provides a promising *theoretical* foundation for the third 'intermediate' risk-performance balance approach to lattice-based cryptography based on the MP-LWE problem, the *practical* significance of the results in [RSSS17] is unclear on several fronts, as follows.

Firstly, it is not investigated in [RSSS17] how to devise optimised algorithms and cryptosystem parameters to efficiently compute the middle-product operation underlying MP-LWE over $\mathbb{Z}_q[x]$ (as opposed to the ring multiplication over $\mathbb{Z}_q[x]/(f(x))$ used in optimised algorithms for PLWE^f based cryptosystems like New Hope [ADPS16]). Consequently, the work of [RSSS17] leaves open the important question of what practical cryptosystem performance is achievable with such optimised algorithms and parameters at a given security level, compared to performance achievable by state of the art cryptosystems based on the classical LWE and PLWE approaches.

Secondly, the security reduction in [RSSS17] is not tight. Specifically, in the reduction presented in [RSSS17] from PLWE^f to MP-LWE, the error standard deviation parameter is amplified by the reduction by a large (though polynomial in the dimension) factor linear in the so-called 'Expansion Factor' of f (introduced in [LM06]) and a dimension parameter d. Consequently, the result in [RSSS17] as it stands does not give meaningful concrete security guarantees unless performance is hugely sacrificed (to the point that it may not be better in practice than the lower risk unstructured LWE based approach).

Thirdly, from a practical lattice cryptanalysis perspective, the result in [RSSS17] says that MP-LWE is *at least* as hard as PLWE^f but leaves the possibility that MP-LWE is much harder, since the dimension of the secret vector in the latter is significantly larger than in the former. This highlights the need for cryptanalysis of MP-LWE to study this potential complexity gap.

1.2 Our Contributions

In this paper, we show how the Middle Product LWE (MP-LWE) problem can be applied to obtain *practical* public-key cryptosystems based on the 'intermediate' risk-performance balance approach initiated in [RSSS17], by addressing the above mentioned shortcomings left open in the work of [RSSS17]. Specifically, we present the following contributions: [2]

- *Optimised Middle Product Algorithms:* We present (Sec. 3) practical and optimised fast algorithms for the middle-product operation underlying MP-LWE. Our algorithm is a generalisation of the Number Theoretic Transform (NTT)-based

[2] Compared to [RSSS17], in this paper, we present research contributions of original NIST-Titanium [SSZa]. The implementation is further improved over that in [SSZa] and the contributions listed here are all new compared to [RSSS17].

algorithm of [HQZ04] to allow for flexible choice of dimensions of the argument polynomials (which is needed to optimise MP-LWE-based cryptosystem parameters for efficiency). We also show how to optimise this algorithm further by introducing a 'Partial $\mathsf{MP}-\mathsf{NTT}$' algorithm that exploits the sparse input and output of the NTTs used within the middle-product computation.

- *Tight Security Proof for* MP-LWE*:* We present (Sec. 4.1) a refined concrete analysis of tightness of the security reduction in [RSSS17] from PLWE^f (for some fixed ring polynomial f) to MP-LWE. We define a measure of this tightness as a property of f called the *geometric factor* of f, and we present a natural large polynomial family $\mathcal{F}$ with an optimal geometric factor of 1. Consequently, we obtain *tight* security reduction (in terms of error parameter amplification) to MP-LWE from the PLWE^f hardness assumption with respect to any f in the family $\mathcal{F}$, closing the tightness issue of the reduction in [RSSS17]. Our reduction can be applied to give concrete hardness guarantees for practical parameter selection (see Sec. 5).
- *Improved Cryptanalysis of* MP-LWE*:* Based on a simple observation exploiting the sparse structure of the MP-LWE matrix, we present and analyse an optimised variant of the 'primal' lattice attack on MP-LWE and show that it asymptotically closes the dimension complexity gap between PLWE^f and MP-LWE problems mentioned above, as the number t of MP-LWE samples increases, for a fixed modulus q. In practice, for smaller values of t, our attack leaves a remaining approximation factor gap of $q^{1/t}$ between the two problems.
- *Application to a practical* MP-LWE*-based cryptosystem:* Titanium*:* To show the practical utility of our results, we present (Sec. 5) the optimised parameters and implementation performance benchmarks of an MP-LWE-based public-key cryptosystem called Titanium (which has been submitted to the NIST PQC process [NISa]), whose implementation is based on our fast middle product algorithm and whose parameters choice take into account our tight security reduction to give strong concrete security guarantees with respect to the hardest PLWE^f over f in our family $\mathcal{F}$. In particular, we show how to choose parameter sets for Titanium for a range of desired security levels, such that the main computational cost of our middle product operations reduces to NTT computations in dimension 256. The use of NTT in dimension 256 as a subroutine has proved to yield a high computational efficiency as well as flexibility and reusability of NTT code for different security levels in recent Module-RLWE based cryptosystem implementations [BDK+17]. We show that our Titanium MP-LWE-based implementation can also enjoy similar flexibility and core NTT code reusability properties. Our performance benchmarks for Titanium show a significant speedup compared to state of the art cryptosystems based on the unstructured LWE problem, while posing a lower security risk than structured lattice schemes over a fixed polynomial ring.
- *Worst-case hardness problem underlying* Titanium*:* Titanium is submitted to the NIST PQC process [NISa], which is still under way. The security analysis in the NIST PQC submission [SSZa] of Titanium bases security on the hardness of MP-LWE and hence, via our tight reduction above, on the average-case hardness of PLWE^f over f in our ring family $\mathcal{F}$. As further qualitative evidence for the security of Titanium, we show that MP-LWE (and hence Titanium security) is tightly as hard as PLWE^f over f in another family $\mathcal{F}_0$ (a slight generalization of the family $\mathcal{F}$ above), which in turn enjoys a provable average case hardness based on the *worst-case* module lattice problems over many rings in $\mathcal{F}_0$, as established

by a set of recent results from [RWS17] (whereas such a worst-case hardness guarantee for PLWE over f in $\mathcal{F}$ is not known). However, we only view this worst-case result as qualitative hardness evidence, since in common with other worst-case to average reductions [RWS17, LPR10], this worst-case reduction is not tight and also requires larger error standard deviations than those used in practical cryptosystems like Titanium.

We believe our results constitute a step forward towards extending the practical applicability of the 'intermediate risk-performance' (MP-LWE based) approach to lattice-based cryptography to other cryptographic primitives.

2 Preliminaries

We use the following notations:

- For $k > 0$, and a ring R, we let $R^{<k}[x]$ denote the set of polynomials with coefficients in R of degree $< k$.
- Given a polynomial $a = a_0 + a_1 x + \cdots + a_{k-1}x^{k-1} \in R^{<k}[x]$, we let

$$\mathsf{PolToVec}(a) := \mathbf{a} = (a_0, \ldots, a_{k-1})^T \in R^k,$$

  and $\mathsf{Rev}(\mathbf{a}) = (a_{k-1}, \ldots, a_0)^T \in R^k$. The latter notation is extended to the corresponding polynomial too. For two polynomials a and b (not necessarily with same degree), it is easy to check that:

$$\mathsf{Rev}(a \cdot b) = \mathsf{Rev}(a) \cdot \mathsf{Rev}(b). \tag{1}$$

- Given two polynomials $a = a_0 + a_1 x + \cdots + a_{k-1}x^{k-1} \in R^{<k}[x]$ and $b = b_0 + b_1 x + \cdots + b_{m-1}x^{m-1} \in R^{<m}[x]$, we denote by $a \cdot b \in R^{<k+m-1}[x]$ the ordinary polynomial product of a and b over $R[x]$.
- Let d_a, d_b, d, k be integers such that $d_a + d_b - 1 = d + 2k$. The middle-product $\odot_d : R[x]^{<d_a} \times R[x]^{<d_b} \to R[x]^{<d}$ is the map:

$$(a, b) \mapsto a \odot_d b = \left\lfloor \frac{(a \cdot b) \bmod x^{k+d}}{x^k} \right\rfloor,$$

  in which the notation $\lfloor \cdot / x^k \rfloor$ means that we divide by x^k as a power series in x and drop the terms $c_j x^j$ with $j < 0$. We use the same notation $\odot_d$ for every d_a, d_b such that $d_a + d_b - 1 - d$ is non-negative and even.
- Let f be a polynomial of degree m with coefficients in ring R. We define M_f as the (Hankel) matrix in $R^{m \times m}$ such that for any $1 \leq i, j \leq m$, the coefficient $(\mathsf{M}_f)_{i,j}$ is the constant coefficient of $x^{i+j-2} \bmod f$.

The (reversed) coefficient vector of the middle-product of two polynomials is in fact equal to the product of the Toeplitz matrix associated to one polynomial by the (reversed) coefficient vector of the second polynomial.

Lemma 1 *Let $d, k > 0$. Let $r \in R^{<k+1}[x]$ and $a \in R^{<k+d}[x]$ and $b = r \odot_d a$. Then $\mathsf{Rev}(\mathbf{b}) = \mathsf{Toep}^{d,k+1}(r) \cdot \mathsf{Rev}(\mathbf{a})$. In other words, we have $\mathbf{b} = \mathsf{Rev}\left(\mathsf{Toep}^{d,k+1}(r) \cdot \mathsf{Rev}(\mathbf{a})\right)$.*

The middle-product is an additive homomorphism when either of its inputs is fixed. As a consequence of the associativity of matrix multiplication and Lemma 1, the middle-product satisfies the following associativity property, which is crucial to the correctness of Titanium (see Section 5).

Lemma 2 ([RSSS17]) *Let $d, k, n > 0$. For all $r \in R[x]^{<k+1}$, $a \in R[x]^{<n+1}$, $s \in R[x]^{<n+d+k}$, we have $r \odot_d (a \odot_{d+k} s) = (r \cdot a) \odot_d s$.*

3 Fast Middle Product Algorithm and Optimisations

A previous work [HQZ04] proposed fast variants of both the Karatsuba multiplication based algorithm and the Fast Fourier Transform (FFT) based algorithm to efficiently compute the special case middle product of the form $a \odot_n b$, when $a \in \mathbb{Z}_q^{<n}[x]$, and $b \in \mathbb{Z}_q^{<2n-1}[x]$. In this section, we generalize the FFT-based approach of [HQZ04] to give a middle product algorithm using the NTT for the general case of $a \odot_d b$, $a \in \mathbb{Z}_q^{<d_a}[x]$, and $b \in \mathbb{Z}_q^{<d_b}[x]$. In addition, we further optimise our algorithm for the case where the dimension d_a and d_b share a large common factor, by applying the partial NTT transformation similar to [SB93]. We did not look much at other polynomial multiplication techniques like Karatsuba.[3] We prefer the flexibility of NTT, i.e. the ability to do pre-computations in key generation algorithm (e.g. sampling the secret in the NTT domain, and pre-computation of NTT), and the ability to reduce our various large dimensions to a single core NTT routine in dimension 256.

For a vector $\mathbf{a} \in \mathbb{Z}_q^{\dim}$ with prime q and $0 \leq k \leq \dim -1$, we let

$$\mathsf{NTT}_{\dim}\left(\omega'_{\dim}, \mathbf{a}\right)_k = \sum_{i=0}^{\dim -1} a_i \cdot (\omega'_{\dim})^{i \cdot k},$$

$$\mathsf{NTT}^{-1}_{\dim}\left(\omega'_{\dim}, \mathbf{a}\right)_k = \dim^{-1} \cdot \sum_{i=0}^{\dim -1} a_i \cdot (\omega'_{\dim})^{i \cdot k},$$

where the $\omega'_{\dim}$ is a fixed primitive dim-th root of unity $\omega_{\dim} \in \mathbb{Z}_q$ or its inverse $\omega^{-1}_{\dim}$.

The function $\mathsf{Zpad}\left(\dim, \mathbf{a}\right)$ pads a vector $\mathbf{a} = (a_0, \ldots, a_{k-1}) \in \mathbb{Z}_q^k$ to the dimension dim with zeros:

$$\mathsf{Zpad}\left(\dim, \mathbf{a}\right) = (a_0, \ldots, a_{k-1}, 0, \ldots, 0) \in \mathbb{Z}_q^{\dim}.$$

The following Lemma shows how the well-known NTT-based algorithm can be employed to compute a polynomial multiplication.

Lemma 3 *For polynomial $a \in \mathbb{Z}_q^{<d_a}[x]$, $b \in \mathbb{Z}_q^{<d_b}[x]$, and integer $\dim \geq d_a + d_b - 1$, let $\mathbf{a} = \mathsf{Zpad}\left(\dim, \mathsf{PolToVec}\left(a\right)\right) \in \mathbb{Z}_q^{\dim}$, and $\mathbf{b} = \mathsf{Zpad}\left(\dim, \mathsf{PolToVec}\left(b\right)\right) \in \mathbb{Z}_q^{\dim}$, the (zero-padded) coefficient vector of the polynomial product $a \cdot b \in \mathbb{Z}_q^{<d_a+d_b-1}[x]$ can be computed as:*

$$\mathsf{NTT}^{-1}_{\dim}\left(\omega^{-1}_{\dim}, \mathsf{NTT}_{\dim}\left(\omega_{\dim}, \mathbf{a}\right) \circ \mathsf{NTT}_{\dim}\left(\omega_{\dim}, \mathbf{b}\right)\right),$$

where $\circ$ is the point-wise multiplication of two vectors.

[3] For an attempt to employ Karatsuba for MP computation, the interested reader is refered to https://github.com/kzoacn/PolyMultiply.

Algorithm 1 : Generalised MP-NTT

Input: $a \in \mathbb{Z}_q^{<d_a}[x]$, and $b \in \mathbb{Z}_q^{<d_b}[x]$.
Output: $\mathbf{c}' = \mathsf{PolToVec}(a \odot_d b) \in \mathbb{Z}_q^d$.
1: **function** MP-NTT(a, b)
2: Let dim be a NTT dimension such that dim $\geq d_b$.
3: Let $\omega_{\dim}$ denote the dim-th root of unity.
4: Compute $\mathbf{a}' = \mathsf{NTT}_{\dim}\left(\omega_{\dim}, \mathsf{Zpad}\left(\dim, \mathsf{PolToVec}\left(\mathsf{Rev}\left(a\right)\right)\right)\right) \in \mathbb{Z}_q^{\dim}$.
5: Compute $\mathbf{b}' = \mathsf{NTT}_{\dim}\left(\omega_{\dim}^{-1}, \mathsf{Zpad}\left(\dim, \mathsf{PolToVec}\left(b\right)\right)\right) \in \mathbb{Z}_q^{\dim}$.
6: Compute $\mathbf{c} = \mathsf{NTT}_{\dim}^{-1}\left(\omega_{\dim}, \mathbf{a}' \circ \mathbf{b}'\right) \in \mathbb{Z}_q^{\dim}$.
7: Let $\mathbf{c}' = (c_0, \ldots, c_{d-1}) \in \mathbb{Z}_q^d$.
8: **end function**

3.1 Generalised Middle Product Algorithm

To compute $a \odot_d b$, for $a \in \mathbb{Z}_q^{<n}[x]$, and $b \in \mathbb{Z}_q^{<n+d-1}[x]$, a naive approach computes and keeps the middle d coefficients of $a \cdot b \in \mathbb{Z}_q^{<2n+d-2}[x]$, by computing three $\mathsf{NTT}_{\dim}$ such that dim $\geq 2n + d - 2$, according to Lemma 3. A faster algorithm [HQZ04] works for the special case when $d = n$ (i.e., computing $a \odot_n b$, for $a \in \mathbb{Z}_q^{<n}[x]$, and $b \in \mathbb{Z}_q^{<2n-1}[x]$), based on the $2n$-dimensional Fast Fourier Transform (FFT). In this case, this algorithm reduces the lower bound of the NTT dimension from $3n - 2$ to $2n$, which saves about $1/3$ of the NTT computation time.

However, the constraint that the number of the coefficients in the MP result is equal to the dimension of a for $a \odot b$, is too restrictive for both the parameter choice and the efficient implementation of MP-based cryptosystems. Here, we observe a generalisation of an algorithm given in [HQZ04] by removing this limitation. For some arbitrary integers d_a, d_b, d, and k, such that $d_a + d_b - 1 = d + 2k$, the generalised MP-NTT in Algorithm 1 computes $a \odot_d b$ using the NTT, for $a \in \mathbb{Z}_q^{<d_a}[x]$, and $b \in \mathbb{Z}_q^{<d_b}[x]$.

Theorem 1 (Adapted from [HQZ04], Th. 11) *Let $M_{p,q,n} : R^{<p}[x] \times R^{<q}[x] \to R^{<n}[x]$, and $\Pi_{n,p,q} : R^{<n}[x] \times R^{<p}[x] \to R^{<q}[x]$, be the bilinear forms defined by:*

$$M_{p,q,n}(y, z) = \left(\sum_{j+k=i+p-1} y_j z_k \right)_{0 \leq i < n},$$

$$\Pi_{n,p,q}(x, y) = \left(\sum_{i+j=k} x_i y_j \right)_{0 \leq k < q}.$$

Then, for any $(X, Y, Z) \in R^{<n}[x] \times R^{<p}[x] \times R^{<q}[x]$, we have:

$$(X | M_{p,q,n}(Y, Z)) = (\Pi_{n,p,q}(X, \mathsf{Rev}(Y)) | Z),$$

where $|$ denotes the canonical inner product of two vectors of the same length. For $q = n + p - 1$, $\Pi_{n,p,q}$ is $X \cdot Y$, and $M_{p,q,n}$ is $Y \odot_n Z$.

We now show that Algorithm 1 computes the middle product.

Lemma 4 *Let $a \in \mathbb{Z}_q^{<d_a}[x]$, $b \in \mathbb{Z}_q^{<d_b}[x]$, and $\mathbf{c} =$ MP-NTT(a, b). Then*

$$\mathbf{c} = \mathsf{PolToVec}(a \odot_d b) \in \mathbb{Z}_q^d.$$

Proof For integers d_a, d_b, and d, such that $d_b = d_a + d - 1$, $c \in \mathbb{Z}_q^{<d}[x]$, $a \in \mathbb{Z}_q^{<d_a}[x]$, and $b \in \mathbb{Z}_q^{<d_b}[x]$, we write $\mathit{\Pi}_{d,d_a,d_b}(c, \mathsf{Rev}(a))$ as:

$$\mathit{\Pi}_{d,d_a,d_b}(c, \mathsf{Rev}(a)) = \mathrm{dim}^{-1} \cdot \sum_{m=0}^{\mathrm{dim}-1} \left(\omega_{\mathrm{dim}}^{m \cdot i} | c\right) \left(\omega_{\mathrm{dim}}^{m \cdot j} | \mathsf{Rev}(a)\right) \omega_{\mathrm{dim}}^{-m \cdot k} \bmod q,$$

where $0 \leq i, j, k < \mathrm{dim}$, and the NTT dimension $\mathrm{dim} \geq d_b$. Then, by Theorem 1, we have:

$$\begin{aligned}(c | M_{d_a,d_b,d}(a,b)) &= (\mathit{\Pi}_{d,d_a,d_b}(c, \mathsf{Rev}(a)) | b) \\ &= \mathrm{dim}^{-1} \cdot \sum_{m=0}^{\mathrm{dim}-1} \left(\omega_{\mathrm{dim}}^{m \cdot i} | c\right) \left(\omega_{\mathrm{dim}}^{m \cdot j} | \mathsf{Rev}(a)\right) \left(\omega_{\mathrm{dim}}^{-m \cdot k} | b\right) \bmod q.\end{aligned}$$

Let $\mathbf{a}' = \mathsf{PolToVec}(\mathsf{Rev}(a))$, $\mathbf{b} = \mathsf{PolToVec}(b)$, and $\mathbf{c} = \mathsf{PolToVec}(a \odot_d b)$. We have:

$$\begin{aligned}\mathbf{c} = M_{d_a,d_b,d}(a,b) &= \mathrm{dim}^{-1} \cdot \sum_{m=0}^{\mathrm{dim}-1} \left(\omega_{\mathrm{dim}}^{m \cdot j} | \mathsf{Rev}(a)\right) \left(\omega_{\mathrm{dim}}^{-m \cdot k} | b\right) \omega_{\mathrm{dim}}^{m \cdot i} \bmod q \\ &= \mathrm{dim}^{-1} \cdot \sum_{m=0}^{\mathrm{dim}-1} \left(\omega_{\mathrm{dim}}^{m \cdot i}\right) \left(\mathsf{NTT}_{\mathrm{dim}}(\omega_{\mathrm{dim}}, \mathbf{a}')_m\right) \left(\mathsf{NTT}_{\mathrm{dim}}\left(\omega_{\mathrm{dim}}^{-1}, \mathbf{b}\right)_m\right) \\ &= \mathsf{NTT}_{\mathrm{dim}}^{-1}\left(\omega_{\mathrm{dim}}, \mathsf{NTT}_{\mathrm{dim}}(\omega_{\mathrm{dim}}, \mathbf{a}') \circ \mathsf{NTT}_{\mathrm{dim}}\left(\omega_{\mathrm{dim}}^{-1}, \mathbf{b}\right)\right) = \mathsf{MP\text{-}NTT}(a,b),\end{aligned}$$

for $0 \leq i, j, k < \mathrm{dim}$, and $\mathsf{NTT}_{\mathrm{dim}}(\omega_{\mathrm{dim}}, \mathbf{a})_m$ is the m-th coordinate of the NTT result. □

3.2 Partial MP-NTT

Let $\mathrm{dim} \geq d_b > d_a$ in computation of $\mathsf{MP\text{-}NTT}(a,b)$, with $a \in \mathbb{Z}_q^{<d_a}[x]$, and $b \in \mathbb{Z}_q^{<d_b}[x]$. The Algorithm 1 needs to pad vectors $\mathsf{PolToVec}(a)$ and $\mathsf{PolToVec}(b)$ to dimension dim, before passing both vectors to the $\mathsf{NTT}_{\mathrm{dim}}$. The naive NTT implementation is relatively inefficient for such a vector with many zero coordinates, due to the unnecessary additions, subtractions, and multiplications with zero. However, for the NTT dimension $\mathrm{dim} = \mathrm{dim}_1 \cdot \mathrm{dim}_2$, and some integer $m \leq \mathrm{dim}_1$, if the coordinates of the input vector $\mathbf{a}$ satisfy $a_i = 0$ for all $i \geq m \cdot \mathrm{dim}_2$, we can compute the NTT transformation by only using the first $m \cdot \mathrm{dim}_2$ coordinates (partial NTT transformation), which avoids the unnecessary computations with zeros.

The partial NTT is similar to the FFT decomposition in [SB93]. Let us denote $\omega_{\mathrm{dim}_1} = \omega_{\mathrm{dim}}^{\mathrm{dim}_2}$, and $\omega_{\mathrm{dim}_2} = \omega_{\mathrm{dim}}^{\mathrm{dim}_1}$. For some integer $m \leq \mathrm{dim}_1$, the partial $\mathsf{NTT}_{\mathrm{dim}}(\omega_{\mathrm{dim}}, \mathbf{a})$ transformation that only uses the first $m \cdot \mathrm{dim}_2$ coordinates, can be written as:

$$\mathsf{NTT}_{\mathrm{dim}}(\omega_{\mathrm{dim}}, \mathbf{a})_{j_1 + \mathrm{dim}_1 \cdot j_2} = \sum_{i_2=0}^{\mathrm{dim}_2 - 1} \left[\omega_{\mathrm{dim}}^{i_2 \cdot j_1} \cdot \left(\sum_{i_1=0}^{m-1} a_{\mathrm{dim}_2 \cdot i_1 + i_2} \cdot \omega_{\mathrm{dim}_1}^{i_1 \cdot j_1}\right)\right] \cdot \omega_{\mathrm{dim}_2}^{i_2 \cdot j_2},$$

for $0 \leq j_1 < \mathrm{dim}_1$, and $0 \leq j_2 < \mathrm{dim}_2$. Similarly, the $\mathsf{MP\text{-}NTT}$ only requires the first d coordinates output of NTT^{-1}, and the naive NTT^{-1} implementation wastes

Algorithm 2 : Partial NTT Algorithm: $\mathsf{NTT\text{-}P}_{\dim}$

Input: m, $\omega'_{\dim}$, and $\mathbf{a}$, satisfying $\dim = \dim_1 \cdot \dim_2$, and $a_i = 0$ for all $i \geq m \cdot \dim_2$.
Output: Partial $\mathsf{Perm}_{\dim}\left(\mathsf{NTT}_{\dim}\left(\omega'_{\dim}, \mathbf{a}\right)\right) \in \mathbb{Z}_q^{\dim}$ transformation.

function $\mathsf{NTT\text{-}P}_{\dim}(m, \omega'_{\dim}, \mathbf{a})$
  for $n_2 = 0, \ldots, \dim_2 - 1$ **do**
    Let $\mathbf{a}'_1$ be the computation result of classical $\mathsf{NTT}_{m \to \dim_1}(\mathbf{a}, n_2)$.
    Set $\left(a_{n_2}, a_{\dim_2 + n_2}, \ldots, a_{(\dim_1 - 1)\cdot \dim_2 + n_2}\right) = \mathbf{a}'_1$.
  end for
  for $n_1 = 0, \ldots, \dim_1 - 1$ **do**
    for $n_2 = 0, \ldots, \dim_2 - 1$ **do**
      Set $a_{\dim_2 \cdot n_1 + n_2} = a_{\dim_2 \cdot n_1 + n_2} \cdot \omega'^{n_1 n_2}_{\dim}$.
    end for
  end for
  for $n_1 = 0, \ldots, \dim_1 - 1$ **do**
    Let $\mathbf{a}_2$ be the decimation $\left(a_{n_1 \cdot \dim_2}, a_{n_1 \cdot \dim_2 + 1} \ldots, a_{(n_1+1)\cdot \dim_2 - 1}\right)$ of $\mathbf{a}$.
    Let $\mathbf{a}'_2$ be the computation result of radix-2 $\mathsf{NTT}_{\dim_2}\left(\omega'_{\dim_2}, \mathbf{a}_2\right)$.
    Set $\left(a_{n_1 \cdot \dim_2}, a_{n_1 \cdot \dim_2 + 1} \ldots, a_{(n_1+1)\cdot \dim_2 - 1}\right) = \mathbf{a}'_2$.
  end for
end function

lots of CPU time in computing the unnecessary coordinates. If we only need the first $m \cdot \dim_2$ coordinates output for some integer $m \leq \dim_1$, we can also rewrite the NTT^{-1} decomposition similar to [SB93], for $0 \leq j_1 < m$, and $0 \leq j_2 < \dim_2$:

$$\mathsf{NTT}^{-1}_{\dim}(\omega_{\dim}, \mathbf{a})_{j_2 + \dim_2 \cdot j_1} =$$

$$\dim^{-1} \cdot \left\{ \sum_{i_1=0}^{\dim_1 - 1} \left[\omega_{\dim}^{i_1 \cdot j_2} \cdot \left(\sum_{i_2=0}^{\dim_2 - 1} a_{\dim_1 \cdot i_2 + i_1} \cdot \omega_{\dim_2}^{i_2 \cdot j_2} \right) \right] \cdot \omega_{\dim_1}^{i_1 \cdot j_1} \right\}.$$

These two NTT decompositions can be viewed as the combination of two sub-dimension NTTs on dimension $\dim_1$ and $\dim_2$, respectively. We denote these two sub-dimension NTTs as $\mathsf{NTT}_{\dim_1}$ and $\mathsf{NTT}_{\dim_2}$. For an efficient implementation of an MP-based cryptosystem, we suggest all the NTT dimensions to share a large power-of-two common factor $\dim_2$, and the remaining factor $\dim_1$ in each NTT dimension to be some small arbitrary integer (around or less than 10). Therefore, we achieve an efficient implementation, by combining an efficient radix-2 $\mathsf{NTT}_{\dim_2}$ algorithm for the large shared sub-dimension $\dim_2$, which has the time complexity $O(\dim_2 \cdot \log \dim_2)$ operations in $\mathbb{Z}_q$; and classical matrix multiplication $\mathsf{NTT}_{\dim_1}$ algorithms with complexity $O\left((\dim_1)^2\right)$ operations in $\mathbb{Z}_q$, for those small sub-dimensions $\dim_1$. A good choice of $\dim_2$ is 256, which has proved high efficiency in recent $\mathsf{Module\text{-}RLWE}$ implementations [BDK+17], and therefore the MP-based cryptosystem implementation can also achieve similar flexibility and reusability of code, compared to the $\mathsf{Module\text{-}RLWE}$ based schemes.

On input $\mathbf{a} \in \mathbb{Z}_q^\ell$, Algorithm Perm lets $\mathbf{a}'_i = \mathbf{a}_{(i \bmod \dim_2)\cdot \dim_1 + \lfloor i/\dim_2 \rfloor}$ for $i < \ell$, and outputs $\mathbf{a}' \in \mathbb{Z}_q^\ell$. On the other hand, on input $\mathbf{a} \in \mathbb{Z}_q^\ell$, Algorithm $\mathsf{InvPerm}$ lets $\mathbf{a}'_i = \mathbf{a}_{(i \bmod \dim_1)\cdot \dim_2 + \lfloor i/\dim_1 \rfloor}$ for $i < \ell$, and outputs $\mathbf{a}' \in \mathbb{Z}_q^\ell$. We give the pseudocode of our partial NTT implementations in Algorithm 2 and 3, respectively. Note that both the output of $\mathsf{NTT\text{-}P}_{\dim}$ and the input of $\mathsf{NTT\text{-}P}^{-1}_{\dim}$ are permuted by $\mathsf{Perm}_{\dim}$, and $\mathsf{NTT\text{-}P}^{-1}_{\dim}$ is the step-by-step reversal of $\mathsf{NTT\text{-}P}_{\dim}$.

Algorithm 3 : Partial NTT inverse algorithm: $\mathsf{NTT\text{-}P}^{-1}_{\mathrm{dim}}$

Input: m, ω'_{dim}, and $\mathbf{a}$. Assume $\mathrm{dim} = \mathrm{dim}_1 \cdot \mathrm{dim}_2$, and $\mathbf{a}$ is permutated by $\mathsf{Perm}_{\mathrm{dim}}$.
Output: the first $m \cdot \mathrm{dim}_2$ coordinates of the $\mathsf{NTT}^{-1}_{\mathrm{dim}}\left(\omega'_{\mathrm{dim}}, \mathsf{InvPerm}_{\mathrm{dim}}(\mathbf{a})\right)$.

function $\mathsf{NTT\text{-}P}^{-1}_{\mathrm{dim}}(m, \omega'_{\mathrm{dim}}, \mathbf{a})$
  for $n_1 = 0, \ldots, \mathrm{dim}_1 - 1$ **do**
    Let $\mathbf{a}_2$ be the decimation $\left(a_{n_1 \cdot \mathrm{dim}_2}, a_{n_1 \cdot \mathrm{dim}_2 + 1} \cdots, a_{(n_1+1) \cdot \mathrm{dim}_2 - 1}\right)$ of $\mathbf{a}$.
    Let $\mathbf{a}'_2$ be the computation result of radix-2 $\mathsf{NTT}_{\mathrm{dim}_2}\left(\omega'_{\mathrm{dim}_2}, \mathbf{a}_2\right)$.
    Set $\left(a_{n_1 \cdot \mathrm{dim}_2}, a_{n_1 \cdot \mathrm{dim}_2 + 1} \cdots, a_{(n_1+1) \cdot \mathrm{dim}_2 - 1}\right) = \mathbf{a}'_2$.
  end for
  for $n_1 = 0, \ldots, \mathrm{dim}_1 - 1$ **do**
    for $n_2 = 0, \ldots, \mathrm{dim}_2 - 1$ **do**
      Set $a_{\mathrm{dim}_2 \cdot n_1 + n_2} = a_{\mathrm{dim}_2 \cdot n_1 + n_2} \cdot \omega'^{n_1 \cdot n_2}_{\mathrm{dim}}$.
    end for
  end for
  for $n_2 = 0, \ldots, \mathrm{dim}_2 - 1$ **do**
    Let $\mathbf{a}'_1$ be the computation result of classical $\mathsf{NTT}_{\mathrm{dim}_1 \to m}(\mathbf{a}, n_2)$.
    Set $\left(a_{n_2}, a_{\mathrm{dim}_2 + n_2}, \ldots, a_{(m-1) \cdot \mathrm{dim}_2 + n_2}\right) = \mathbf{a}'_1$.
  end for
  for $i = 0, \ldots, m \cdot \mathrm{dim}_2 - 1$ **do**
    Set $a_i = a_i \cdot \mathrm{dim}^{-1}$.
  end for
end function

Algorithm 4 : Partial MP-NTT-P

Input: $a \in \mathbb{Z}_q^{<d_a}[x]$, and $b \in \mathbb{Z}_q^{<d_b}[x]$.
Output: $\mathbf{c}' = \mathsf{PolToVec}(a \odot_d b) \in \mathbb{Z}_q^d$.

function MP-NTT-P(a, b)
  Let $\mathrm{dim} = \mathrm{dim}_1 \cdot \mathrm{dim}_2$ be the minimal multiple of dim_2 satisfying $\mathrm{dim} \geq d_b$.
  Let ω_{dim} denote the dim-th root of unity.
  Let $m_a \leq \mathrm{dim}_1$ be the minimal integer such that $d_a \leq m_a \cdot \mathrm{dim}_2$.
  Compute $\mathbf{a}' = \mathsf{NTT\text{-}P}_{\mathrm{dim}}(m_a, \omega_{\mathrm{dim}}, \mathsf{Zpad}(m_a \cdot \mathrm{dim}_2, \mathsf{PolToVec}(\mathsf{Rev}(a)))) \in \mathbb{Z}_q^{\mathrm{dim}}$.
  Let $m_b \leq \mathrm{dim}_1$ be the minimal integer such that $d_b \leq m_b \cdot \mathrm{dim}_2$.
  Compute $\mathbf{b}' = \mathsf{NTT\text{-}P}_{\mathrm{dim}}\left(m_b, \omega^{-1}_{\mathrm{dim}}, \mathsf{Zpad}(m_b \cdot \mathrm{dim}_2, \mathsf{PolToVec}(b))\right) \in \mathbb{Z}_q^{\mathrm{dim}}$.
  Let $m_c \leq \mathrm{dim}_1$ be the minimal integer satisfying $d \leq m_c \cdot \mathrm{dim}_2$.
  Compute $\mathbf{c} = \mathsf{NTT\text{-}P}^{-1}_{\mathrm{dim}}(m_c, \omega_{\mathrm{dim}}, \mathbf{a}' \circ \mathbf{b}') \in \mathbb{Z}_q^{m_c \cdot \mathrm{dim}_2}$.
  Let $\mathbf{c}' = (c_0, \ldots, c_{d-1}) \in \mathbb{Z}_q^d$.
end function

Given NTT dimension $\mathrm{dim} = \mathrm{dim}_1 \cdot \mathrm{dim}_2$, $a \in \mathbb{Z}_q^{<d_a}[x]$, and $b \in \mathbb{Z}_q^{<d_b}[x]$, such that $d_a \leq m_a \cdot \mathrm{dim}_2$ and $d_b \leq m_b \cdot \mathrm{dim}_2$, for some integer $m_a, m_b \leq \mathrm{dim}_1$, we present the MP-NTT-P algorithm merged with the partial NTT transformations in Algorithm 4.

Lemma 5 *Given* NTT *dimension* $\mathrm{dim} = \mathrm{dim}_1 \cdot \mathrm{dim}_2$, $a \in \mathbb{Z}_q^{<d_a}[x]$, *and* $b \in \mathbb{Z}_q^{<d_b}[x]$, *such that* $d_a \leq m_a \cdot \mathrm{dim}_2$ *and* $d_b \leq m_b \cdot \mathrm{dim}_2$, *for some integer* $m_a, m_b \leq \mathrm{dim}_1$, *we have*

$$\mathsf{MP\text{-}NTT\text{-}P}(a, b) = \mathsf{MP\text{-}NTT}(a, b).$$

Let T_{dim_2} denote the running time of the radix-2 NTT subroutine for dimension dim_2, which is the dominate part of NTT-P's running time. Therefore, the MP-NTT-P

algorithm has time complexity ($\mathbb{Z}_q$ operations) of about:

$$3 \cdot (\dim_1 \cdot T_{\dim_2}) + O(\dim).$$

Note that in the above estimate, we neglected quadratic run-time dependence on classical NTT for dimension $\dim_1$, since we assume $\dim_1 = O(1)$ is small in our case.

Compared to the implementation submitted in [SSZa], our new implementation [4] merges the modulo q reductions among the intermediate levels of the NTT (i.e. we do not perform reductions on each level) to improve the efficiency. In addition, we acknowledge the issues mentioned in [Sei18], that the compiler may not generate constant time executable code for the modulo arithmetic (i.e. the "%" operator) in C programming language. We examined our previous implementation by a code review, and replaced all the modulo arithmetic by either the Montgomery reduction or the Barrett reduction implemented in constant time [Har14].

4 Tighter Security Analysis of MP-LWE

In this Section, we present a tighter security analysis of the MP-LWE problem than previously known, to improve the applicability of the security analysis to practical schemes. In particular, we tighten the analysis from both the security proof and cryptanalysis directions to close the gap between them and establish a 'near equivalence' between MP-LWE and PLWE^f (for f in a large family) as follows:

(1) For the security proof direction, we show how to remove the (polynomial in dimension) amplification factor in error standard deviation of the security reduction in [RSSS17] from the PLWE^f problem (for some f) to the MP-LWE problem, by optimising the reduction and specializing it to a more restricted, but still huge, class of f's. This shows that MP-LWE is *concretely* at least as hard as PLWE^f for a large class of f's and the same error distribution, not only asymptotically up to polynomial approximation factors as in the reduction of [RSSS17].

(2) For the cryptanalysis direction, we analyse a simple optimisation of generic LWE lattice attacks, which takes advantage of the sparse block Toeplitz structure of the matrix underlying MP-LWE, and show that this closes the gap in optimum lattice dimension for attacking MP-LWE versus the dimension for attacking the underlying PLWE^f problem, leaving only a gap factor $q^{1/t}$, in the uSVP approximation factor needed for solving the two problems, assuming t MP-LWE samples in the MP-LWE instance. This gap factor tends to 1 as t grows, so for sufficiently large t, t-sample MP-LWE is *concretely* no harder than PLWE^f for a large class of f's (for small t, the concrete hardness of MP-LWE may be higher than PLWE^f).

4.1 Tighter MP-LWE security proof from hardest PLWE^f in a family

We recall the definition of the PLWE^f problem (originally defined as the search variant in [SSTX09] and as the decision variant in [BV11]) and the MP-LWE problem defined in [RSSS17] (for convenience, our definition uses discrete error distributions rather than continuous distributions used in [RSSS17]).

[4] Available at https://github.com/raykzhao/Titanium.

Definition 1 (**$\mathsf{PLWE}^f_{q,t,\chi}$ Problem**) Let $q \geq 2$, $m, t > 0$, f a polynomial of degree m, χ a distribution over $\mathbb{Z}[x]/f$. The (decision t-sample) Polynomial LWE Problem $\mathsf{PLWE}^f_{q,t,\chi}$ consists in distinguishing between t samples of the form $(a_i, b_i = a_i \cdot s + e_i)$ (where $s \hookleftarrow (U(\mathbb{Z}_q[x]/f))$, $a_i \hookleftarrow U(\mathbb{Z}_q[x]/f)$ and $e_i \hookleftarrow \chi$ for $i = 1, \ldots, t$) and t independent samples from $U(\mathbb{Z}_q[x]/f \times \mathbb{Z}_q[x]/f)$, with non-negligible advantage.

Definition 2 (**$\mathsf{MP\text{-}LWE}_{q,n,d,t,\chi}$ Problem**) Let $n, d > 0$, $q \geq 2$, and a distribution χ over $\mathbb{Z}^{<d}[x]$. The (decision t-sample) Middle-Product LWE Problem $\mathsf{MP\text{-}LWE}_{q,n,d,t,\chi}$ consists in distinguishing between t samples of the form $(a_i, b_i = a_i \odot_d s + e_i)$ (where $s \hookleftarrow U(\mathbb{Z}_q^{<n+d+1}[x])$, $a_i \hookleftarrow U(\mathbb{Z}_q^{<n}[x])$, and $e_i \hookleftarrow \chi$ for $i = 1, \ldots, t$) and t independent samples from $U(\mathbb{Z}_q^{<n}[x] \times \mathbb{Z}_q^{<d}[x])$, with non-negligible advantage.

Geometric factor of a polynomial. A close look at the reduction of [RSSS17] from $\mathsf{MP\text{-}LWE}$ to PLWE^f for a ring polynomial f, shows that the reduction amplifies the standard deviation of the error distribution by a factor that depends on a certain matrix $\mathsf{M}^{d'}_f$ related to f (this matrix also distorts the shape of the error distribution in the reduction). Consequently, the tightness of the reduction is dictated by this matrix. Accordingly, we call this matrix the *geometric matrix* of f, and we also define as a measure of error standard deviation amplification factor the *geometric factor* of f. These are precisely defined as follows.

Definition 3 (**Geometric Matrix/Factor**) Given a monic polynomial f of degree n, and an integer $d' \leq n$, its d'-geometric matrix $\mathsf{M}^{d'}_f$ is defined as the top d' rows of the Hankel matrix M_f having anti-diagonal element $\mathsf{ADiag}_j(\mathsf{M}_f)$ as the constant coefficient of the polynomial $x^{j-1} \bmod f$, for $j = 1, \ldots, 2m-1$. The geometric factor $G^{d'}(f)$ of f is defined as $G^{d'}(f) = \|\mathsf{M}^{d'}_f\|$. For a family $\mathcal{F}$ of polynomials of degree $\geq d'$, we define its geometric factor $G(\mathcal{F})$ as the maximum of $G^{d'}(f)$ over all f in $\mathcal{F}$.

We can now state the core result of [RSSS17] as follows.

Theorem 2 (Adapted from [RSSS17], Le. 3.7) *Let $n, d', t > 0$, $q \geq 2$, and let f denote a polynomial $f \in \mathbb{Z}[x]$ that is monic, has constant coefficient coprime with q, and has degree m in $[d', n]$. Let $\chi_{e,\mathsf{P}}$ denote a PLWE error distribution over $\mathbb{Z}[x]/f$ (i.e., over $\mathbb{Z}^m$ in the coefficient representation of $\mathbb{Z}[x]/f$), and let $\chi_{e,\mathsf{MP}}$ denote a $\mathsf{MP\text{-}LWE}$ error distribution over $\mathbb{Z}^{<d'}[x]$ (i.e., over $\mathbb{Z}^d$ in the coefficient representation of $\mathbb{Z}^{<d}[x]$) defined in the coefficient representation by*

$$\chi_{e,\mathsf{MP}} \stackrel{\text{def}}{=} \mathsf{J} \cdot \mathsf{M}^{d'}_f \cdot \chi_{e,\mathsf{P}}, \tag{2}$$

where J is the matrix for the coefficient reversal function Rev (with 1's on the anti-diagonal and 0's elsewhere).

Then any attack against $\mathsf{MP\text{-}LWE}_{q,n,d',t,\chi_{e,\mathsf{MP}}}$ with run-time $T_{\mathsf{MP\text{-}LWE}}$ and advantage $\varepsilon_{\mathsf{MP\text{-}LWE}}$, implies an attack against the $\mathsf{PLWE}^f_{q,t,\chi_{e,\mathsf{P}}}$ problem with run-time

$$T_{\mathsf{PLWE}} \approx T_{\mathsf{MP\text{-}LWE}} \tag{3}$$

and distinguishing advantage

$$\varepsilon_{\mathsf{PLWE}} \geq \varepsilon_{\mathsf{MP\text{-}LWE}}. \tag{4}$$

For general f, the geometric matrix $\mathsf{M}_f^{d'}$ of f is closely related to the structure of f, and this causes the distribution $\chi_{e,\mathsf{MP}}$ to also depend on the structure of f. However, the overall goal of [RSSS17] was to reduce from PLWE^f for a large family of f's to MP-LWE with a *single* $\chi_{e,\mathsf{MP}}$ distribution that is *independent of* f. In the full security reduction of [RSSS17] (see Theorem 3.6 there), this is achieved by setting $\chi_{e,\mathsf{MP}}$ as a spherical Gaussian distribution and adding a 'noise unskewing' step in the reduction to 'unskew' the covariance matrix back to a diagonal matrix. Then the main result (Theorem 3.6) in [RSSS17] is that PLWE^f with a spherical Gaussian error distribution with standard deviation $\alpha \cdot q$ reduces to MP-LWE with a spherical Gaussian error distribution with standard deviation $\alpha' \cdot q$ amplified by the d'-geometric factor of f, i.e., $\alpha' = G^{d'}(f) \cdot \alpha = \|\mathsf{M}_f^{d'}\| \cdot \alpha$.

In [RSSS17], it is observed (Lemma 3.7 of [RSSS17]) that the geometric factor can be upper bounded by the expansion factor $\mathsf{EF}(f)$ of f (defined in [LM06]) as follows: $\|\mathsf{M}_f^{d'}\| \leq \|\mathsf{M}_f\| \leq \deg(f) \cdot \mathsf{EF}(f)$. Then using known bounds on $\mathsf{EF}(f)$ from [LM06], this leads to bounds on the geometric factor of f. However, this leads to large upper bounds on the geometric factor, and therefore a loose reduction (e.g. even for 'nice' polynomials like $f = x^m + 1$, we have $\mathsf{EF}(f) = 2$ but this only gives $G^{d'}(f) \leq 2d$, which is a large bound on the geometric factor which is linear in the dimension parameter d).

Instead, here we obtain a much tighter upper bound on the geometric factor by directly bounding it from its definition, taking into account the restriction to the first d' rows of M_f to improve our geometric factor bound to the optimal value of 1 for a suitable large family of polynomials. Moreover, we show that for this family, the geometric matrix of f is a projection matrix, so that even the error distribution *shape* is preserved exactly, making the overall reduction more general in terms of error distribution than the overall reduction of [RSSS17], which was restricted to Gaussian error distributions. This allows the reduction to not only be quite tight (to allow a meaningful setting of our parameters based on the hardness of PLWE) but also to be applicable to efficiently sampleable error distributions; in particular, it can be applied with the centered binomial difference distribution [ADPS16] (see our Titanium cryptosystem in the following Section).

To obtain our tight reduction, we apply Theorem 2 to the following ring polynomial family $\mathcal{F}$:

Definition 4 (Ring polynomial family $\mathcal{F}$) For integers $n \geq m' \geq d'$, we denote by $\mathcal{F}(n, m', d')$ the set of ring polynomials f of the form

$$f(x) = x^m + \sum_{i \leq \ell(m)} f_i \cdot x^i \tag{5}$$

with

$$m' \leq m \leq n, \tag{6}$$

and

$$\ell(m) = \min(m/2 + 1, m + 1 - d'), \tag{7}$$

and

$$f_0 \in \{-1, 1\}. \tag{8}$$

We choose $\mathcal{F}$ as a hardness basis for our security proof of MP-LWE because:

– If $m' \leq (1-\varepsilon')n$ for a constant $\varepsilon' > 0$, $\mathcal{F}$ contains an exponentially large (in n) number of polynomials (rings).
– It potentially (for a suitable choice of d', m' and n) contains known ring modulus polynomials previously used in lattice-based cryptography, such cyclotomic polynomials $x^m + 1$ (for m a power of 2), used in encryption schemes since [SSTX09] and later in New Hope [ADPS16], as well as non-cyclotomic polynomials such as $x^m - x - 1$ (for m prime) used in NTRU Prime [BCLvV16].
– MP-LWE enjoys a tight reduction from the hardest ring in the family, due to the family's optimal d'-geometric factor of 1, by the results summarized below, and the reduction preserves the shape of the distribution if the latter is balanced and has independent coordinates.

The following proposition and its corollary show that $\mathcal{F}$ has an optimal geometric factor.

Proposition 1 *Let $f(x) = x^m + \sum_{i \leq \ell} f_i \cdot x^i$, with $\ell \leq m/2 + 1$. Then,*

$$\mathsf{ADiag}_1(\mathsf{M}_f) = 1 \tag{9}$$

and

$$\mathsf{ADiag}_j(\mathsf{M}_f) = 0 \text{ if } 2 \leq j \leq m \text{ or } m+2 \leq j \leq 2m-\ell, \tag{10}$$

and

$$\mathsf{ADiag}_j(\mathsf{M}_f) = -f_0 \text{ if } j = m+1 \tag{11}$$

and

$$\mathsf{ADiag}_j(\mathsf{M}_f) = f_{2m+1-j} \cdot f_0 \text{ if } 2m-\ell+1 \leq j \leq 2m-1. \tag{12}$$

Proof From the definition of M_f, we have $\mathsf{ADiag}_j(\mathsf{M}_f) = (x^{j-1} \bmod f(x)) \bmod x$ for $1 \leq j \leq 2n-1$. From this (9) and (10) for $j \leq m$ follow immediately. For $j = m+1$, we have $x^m \bmod f(x) = -\sum_{i \leq \ell} f_i \cdot x^i$ from the definition of f, which gives (11). For $m+2 \leq j \leq 2m-\ell$, we have $x^{j-1} \bmod f(x) = x^{j-m-1} \cdot (-\sum_{i \leq \ell} f_i \cdot x^i) \bmod f(x) = 0$ since $j-m-1+\ell \leq m-1$, giving (10). Finally, for $2m-\ell+1 \leq j \leq 2m-1$, we have $x^{j-1} \bmod f(x) = x^{j-m-1} \cdot (-\sum_{i \leq \ell} f_i \cdot x^i) \bmod f(x) = -\sum_{i \leq \ell} f_i \cdot x^{i+j-m-1} \bmod f(x)$. Using $(x^{i+j-m-1} \bmod f(x)) \bmod x = -f_0$ if $i + j - m - 1 = m$ (i.e., $i = 2m+1-j$) and $(x^{i+j-m-1} \bmod f(x)) \bmod x = 0$ if $m+1 \leq i+j-m-1 \leq 2m-\ell$, we get (12). □

For f as in Proposition 1, the first d' rows of M_f contain elements from anti diagonals $1, \ldots, m+d'-1$. Therefore, if the condition $2m-\ell+1 > m+d'-1$ holds (or equivalently, $\ell \leq m+1-d'$), the condition (12) is never satisfied in the first d' rows of M_f, so the non-zero columns of $\mathsf{M}_f^{d'}$ are orthogonal and (using $|f_0| = 1$) have unit norm (with one '1' coordinate and the rest 0). We therefore obtain the following corollary, which is our main result in this Section.

Corollary 1 *For integers n and $d' \leq m' \leq n$, the family $\mathcal{F}(n, m', d')$ in Def. 4 has geometric factor $G(\mathcal{F}) = 1$.*

Corollary 2 *For integers n and $d' \leq m' \leq n$, let $t > 0$, $q \geq 2$, and f denote a polynomial $f \in \mathcal{F}(n, m', d')$ of degree m in $[m', n]$. Let $\chi_{e,\mathsf{P}}$ denote a PLWE error distribution over $\mathbb{Z}[x]/f$ (i.e., over $\mathbb{Z}^m$ in the coefficient representation of $\mathbb{Z}[x]/f$) that has independent identically distributed coordinates, i.e., $\chi_{e,\mathsf{P}} = \chi_c^m$ for some*

distribution χ_c over $\mathbb{Z}$ which is balanced (i.e., $\chi_c(x) = \chi_c(-x)$ for all $x \in \mathbb{Z}$). Let $\chi_{e,\mathsf{MP}} = \chi_c^{d'}$.

Then any attack against $\mathsf{MP\text{-}LWE}_{q,n,d',t,\chi_{e,\mathsf{MP}}}$ with run-time $T_{\mathsf{MP\text{-}LWE}}$ and advantage $\varepsilon_{\mathsf{MP\text{-}LWE}}$, implies an attack against the $\mathsf{PLWE}^f_{q,t,\chi_{e,\mathsf{P}}}$ problem with run-time

$$T_{\mathsf{PLWE}} \approx T_{\mathsf{MP\text{-}LWE}} \tag{13}$$

and distinguishing advantage

$$\varepsilon_{\mathsf{PLWE}} \geq \varepsilon_{\mathsf{MP\text{-}LWE}}. \tag{14}$$

In particular, this reduction holds for $\chi_c = \mathsf{BinDiff}(\eta)$ (i.e., $\chi_{e,\mathsf{MP}} = \mathsf{BinDiff}(\eta)^{d+k}$), the error distribution specified for Titanium-CPA.

Proposition 1 together with Corollary 2 show that for f in family $\mathcal{F}(n, m', d')$, the reduction from PLWE to MP-LWE is tight in terms of error variance, and moreover preserves the shape of the distribution, under mild conditions. They also show that the error distribution in MP-LWE can be exactly the same as in PLWE. This allows us to compute practical concrete parameters based on the hardness of PLWE, compared to using the impractical parameters one would get with the loose reduction in [RSSS17]. Variables m' and n are the min/max degree of ring polynomial in family whose hardest PLWE complexity is relied upon. As m' decreases, the family grows (reducing the risk) but hardness of PLWE in the rings of lowest dimension m' decreases. We chose m' in Table 1 minimal while guaranteeing PLWE security in dimension m' exceeds the desired level.

4.2 A variant tight reduction from a ring-family with worst-case hardness

In this subsection, we prove in Corollary 3 a slight variant of the tight reduction in Corollary 2. The variant reduction maintains the tightness of Corollary 2, but applies to a more general family of ring polynomials $\mathcal{F}_0$ in which the condition (8) on the constant coefficient f_0 of f is replaced with the much weaker condition that that coefficient is coprime to q (see Definition 5). This more general class $\mathcal{F}_0$ has the advantage that it contains many rings over which the average-case hardness of certain PLWE problems is known [RWS17] to be related to the hardness of worst-case lattice problems. Namely, such a worst-case to average-case reduction could be derived by combining Theorem 2.13 in Sec. 3 of [RWS17] with the results of Sec. 4 of [RWS17], and the results of [LS15] and [AD17] (see Fig. 1 of [RWS17]; we remark that the condition on the size of f_0 is needed by Theorem 4.7 of [RWS17]). Therefore, Corollary 3 below shows that the hardness of MP-LWE can be related to the hardness of worst-case module lattice problems over many rings (via PLWE over $\mathcal{F}_0$ rather than PLWE over $\mathcal{F}$). Although the error variance needed for those worst-case reductions to hold is much larger than that used in the MP-LWE instances underlying Titanium, we view these worst-case to average-case reductions as additional qualitative evidence for the hardness of PLWE over $\mathcal{F}_0$. In particular, it shows that the extra restriction (7) we introduce on the PLWE family $\mathcal{F}_0$ (versus the family used in [RSSS17] and the worst-case reduction for MP-LWE in [RWS17]) to achieve a tight reduction from MP-LWE to PLWE still leaves many rings in the family for which PLWE has a hardness reduction from worst-case problems.

Definition 5 (Ring polynomial family $\mathcal{F}_0$) For integers $d' \leq m' \leq n$, we denote by $\mathcal{F}_0(n, m', d')$ the set of ring polynomials f satisfying conditions (5), (6) and (7) and the condition that the constant coefficient f_0 of f is coprime to q (but f_0 does not need to satisfy (8)).

The variant reduction in Corollary 3 below reduces from a PLWE instance over a ring of dimension $\geq d'$ to only $d'-1$ MP-LWE samples (one sample less than the d' MP-LWE samples provided by Corollary 3). To apply Corollary 3 to the hardness of Titanium (which relies on the $(d+k)$-sample MP-LWE), we therefore set $d' = d+k+1$ (whereas we could set $d' = d+k$ when applying Corollary 2, i.e. this just increases the minimal PLWE dimension d' for the hardness basis of Titanium by 1).

Corollary 3 *For integers n and $d' \leq m' \leq n$, let $t > 0$, $q \geq 2$, and f denote a polynomial $f \in \mathcal{F}_0(n, m', d')$ of degree m in $[m', n]$. Let $\chi_{e,\mathsf{P}}$ denote a* PLWE *error distribution over $\mathbb{Z}[x]/f$ (i.e., over $\mathbb{Z}^m$ in the coefficient representation of $\mathbb{Z}[x]/f$) that has independent identically distributed coordinates, i.e., $\chi_{e,\mathsf{P}} = \chi_c^m$ for some distribution χ_c over $\mathbb{Z}$ which is balanced (i.e., $\chi_c(x) = \chi_c(-x)$ for all $x \in \mathbb{Z}$). Let $\chi_{e,\mathsf{MP}} = \chi_c^{d'-1}$.*

Then any attack against $\mathsf{MP\text{-}LWE}_{q,n,d'-1,t,\chi_{e,\mathsf{MP}}}$ with run-time $T_{\mathsf{MP\text{-}LWE}}$ and advantage $\varepsilon_{\mathsf{MP\text{-}LWE}}$, implies an attack against the $\mathsf{PLWE}^f_{q,t,\chi_{e,\mathsf{P}}}$ problem with run-time $T_{\mathsf{PLWE}} \approx T_{\mathsf{MP\text{-}LWE}}$ and and distinguishing advantage $\varepsilon_{\mathsf{PLWE}} \geq \varepsilon_{\mathsf{MP\text{-}LWE}}$.

Proof Given a $\mathsf{PLWE}^f_{q,t,\chi_{e,\mathsf{P}}}$ instance, we first apply Theorem 2 to reduce it to a $\mathsf{MP\text{-}LWE}_{q,n,d',t,\chi_{e,\mathsf{MP}}}$ instance. By Lemma 1, this instance can be written in the form

$$\mathsf{J}^{d'} \cdot \mathbf{b}_i = \mathsf{Toep}^{d',n}(a_i) \cdot \mathsf{J}^{n+d'} \cdot \mathbf{s} + \mathsf{J}^{d'} \cdot \mathsf{M}_f^{d'} \cdot \mathbf{e}_i, \tag{15}$$

where $\mathbf{e}_i$ is sampled from the PLWE error distribution $\chi_{e,\mathsf{P}} = \chi_c^m$. By Proposition 2 and the condition (7), the matrix $\mathsf{J}^{d'} \cdot \mathsf{M}_f^{d'}$ has as its top $(d'-1)$ rows the matrix

$$\left[0^{(d'-1)\times(m-(d'-1))} \middle| -f_0 \cdot \mathsf{I}^{d'-1}\right],$$

where $\mathsf{I}^{d'-1}$ denotes the $(d'-1)$-dimensional identity matrix. Therefore, dropping the last row of (15) gives

$$\mathsf{J}^{d'-1} \cdot \mathbf{b}'_i = \mathsf{Toep}^{d'-1,n}(a_i) \cdot \mathsf{J}^{n+d'-1} \cdot \mathbf{s}' - f_0 \cdot \mathbf{e}'_i, \tag{16}$$

where $\mathbf{b}'_i$, $\mathbf{s}'$, and $\mathbf{e}'_i$ consist of the last $(d'-1)$ coordinates of $\mathbf{b}_i$, $\mathbf{s}$, and $\mathbf{e}_i$, respectively. Multiplying (16) by $-f_0^{-1}$ gives the desired $\mathsf{MP\text{-}LWE}_{q,n,d'-1,t,\chi_{e,\mathsf{MP}}}$ instance

$$\mathsf{J}^{d'-1} \cdot \mathbf{b}''_i = \mathsf{Toep}^{d'-1,n}(a_i) \cdot \mathsf{J}^{n+d'-1} \cdot \mathbf{s}'' + \mathbf{e}'_i, \tag{17}$$

where $\mathbf{b}''_i = -f_0^{-1} \cdot \mathbf{b}'_i$ and $\mathbf{s}'' = -f_0^{-1} \cdot \mathbf{s}'$. It is easy to see that the reduction also maps uniformly random $\mathbf{b}_i$'s to uniformly random $\mathbf{b}''_i$'s, as claimed. □

Table 1 Concrete polynomial family ($\mathcal{F}$) parameters for Titanium cryptosystem.

Parameter	Toy64	Lite96	Std128	Med160	Hi192	Super256
$m_{min} = m'$	654	770	896	1230	1486	1998
$m_{max} = n$	684	800	1024	1280	1536	2048
$\ell(m') = m' - d'$	142	35	128	462	462	718
lo bnd on $\log_3(\lvert\mathcal{F}\rvert)$	172	65	256	512	512	768
power-of-two inclusion	×	×	✓	×	×	✓

4.2.1 Example of Concrete Ring Polynomial families:

Table 1 shows our parameter choices for the family $\mathcal{F}(n, m', d')$ used as the PLWE^f hardness basis for our Titanium cryptosystem parameter sets by applying Corollary 2 (see following Section for more details on Titanium). Note that degree of polynomials in $\mathcal{F}$ range from $m_{min} = m'$ up to $m_{max} = n$, and the largest non-leading monomial degree of polynomials in $\mathcal{F}$ is $\ell(m') = m_{min} - d'$. We also show a lower bound on the number of polynomials in the family, just counting those with ± 1 non-zero coefficients, showing the huge size of our concrete families. We remark that a recent result [RWS17] gives worst-case to average case hardness results for a family similar to our family $\mathcal{F}$, assuming the error distribution coefficient standard deviation is sufficiently large (however this value is larger than we use in our practical parameter choices in Titanium, so the results of [RWS17] do not apply to our parameter settings).

4.3 Tighter cryptanalysis of MP-LWE

4.3.1 Lattice attacks on MP-LWE*:*

Let $n' = n + d' - 1$. We recall that an $\mathsf{MP\text{-}LWE}_{q,n,d',t,\chi_{e,\mathsf{MP}}}$ instance is of the form $(a_i, b_i = a_i \odot_{d+k} s + e_i)_{i \le t} \in (\mathbb{Z}_q^{<n}[x] \times \mathbb{Z}_q^{d+k}[x])^t$ with the secret key $s \in \mathbb{Z}_q^{<n'}[x]$. The search attack on MP-LWE consists in recovering s from $(a_i, b_i)_{i \le t}$. Viewing MP-LWE as a special case of an LWE instance in dimension $n' = n + d + k - 1$, any of the known algorithms for the search (rather than decision) variant of LWE could be used. In particular, we could use a search variant of the 'dual lattice' attack applied to MP-LWE. One such variant that seems to give the lowest complexity is based on Kannan's embedding method [Kan87] to convert the LWE instance to a 'unique SVP' instance, as analysed by Albrecht et al. in [AFG13] and Alkim et al. in [ADPS16]. We describe the improved [ADPS16] variant of this attack below (we call it the 'primal embedding attack'), and then we explain how to optimise it to take advantage of the special Toeplitz structure of the MP-LWE matrix.

The generic primal 'embedding attack' Let $t' = t \cdot d'$. The 'embedding attack' [Kan87, AFG13] consists in rewriting the $\mathsf{MP\text{-}LWE}_{q,n,d',t,\chi_{e,\mathsf{MP}}}$ instance $(a_i, b_i = a_i \odot_{d'} s + e_i)_{i \le t} \in (\mathbb{Z}_q^{<n}[x] \times \mathbb{Z}_q^{d'}[x])^t$ as an LWE instance $(\mathsf{A}, \mathbf{b}) \in \mathbb{Z}_q^{t' \times n'} \times \mathbb{Z}_q^{t'}$ over $\mathbb{Z}_q$, where $\mathbf{b} = \mathsf{A} \cdot \mathbf{s} + \mathbf{e}$. We use a subset of $m^* \le t' = t \cdot d'$ samples (rows) of the instance to give the sublattice LWE instance $(\mathsf{A}^*, \mathbf{b}^* = \mathsf{A}^* \cdot \mathbf{s} + \mathbf{e}^*) \in \mathbb{Z}_q^{m^* \cdot n'} \times \mathbb{Z}_q^{m^*}$, where the sublattice dimension m^* is chosen by the attacker to optimise the attack

(see below). The attack constructs a (column) basis matrix $\bar{\mathsf{A}}^* \in \mathbb{Z}^{m^* \times m^*}$ for the m^*-dimensional LWE (primal) lattice $L_q(\mathsf{A}^*) = \{\mathsf{A}^* \cdot \mathbf{u} + q \cdot \mathbb{Z}^{m^*} : \mathbf{u} \in \mathbb{Z}_q^{n'}\}$ and builds a basis $\mathsf{B} \in \mathbb{Z}^{(m^*+1)\times(m^*+1)}$ for an *embedding* lattice $L(\mathsf{B})$ of the form

$$\mathsf{B} = \begin{pmatrix} \bar{\mathsf{A}}^* & \mathbf{b}^* \\ \mathbf{0}^T & 1 \end{pmatrix},$$

The (m^*+1)-dimensional embedding lattice $L(\mathsf{B})$ generated by the columns of B therefore contains an embedding of the LWE primal lattice $L_q(\mathsf{A}^*)$ and the target LWE vector $\mathbf{b}^*$. In particular, the embedding lattice $L(\mathsf{B})$ contains the short vector $\mathbf{v} = (\mathbf{e}, 1)^T$ of norm $\|\mathbf{v}\| \approx \|\mathbf{e}^*\|$. By running a lattice basis reduction algorithm on B, the attack aims at recovering $\mathbf{v}$ as one of the vectors in the reduced basis (which immediately reveals the error $\mathbf{e}^*$ and then the secret $\mathbf{s}$). There are several analyses/variants of the success condition (and complexity) of this attack in the literature. Here, we evaluate it using the heuristic analysis approach of [ADPS16], based on the block-structure of the BKZ lattice reduction algorithm and the Geometric Series Assumption for the output basis, which tends to give lower (and probably more realistic) complexity estimates than more rigorous approaches [AFG13].

The primal attack analysis of [ADPS16] assumes that the BKZ reduction algorithm is applied to B and applies the Geometric Series Assumption (GSA) [Sch03] to model the BKZ output basis Gram-Schmidt norms $\|\mathbf{b}_i^*\|$ $(i = 0, \ldots, m^*)$ as a geometric series; namely, since the lattice determinant is with high probability $\det(L(\mathsf{B})) = \det(L_q(\mathsf{A}^*) = q^{m^*-n'}$, we have:

$$\|\mathbf{b}_i^*\| = \delta(b)^{m^*-2i} \cdot q^{\frac{m^*-n'}{m^*+1}}, i = 0, \ldots, m^*. \tag{18}$$

The heuristic in [ADPS16] is that if the attack fails to recover the short vector $\mathbf{v}$ from the BKZ reduced basis, the projection $\pi_S(\mathbf{v})$ of $\mathbf{v}$ onto the vector space S spanned by the last b BKZ GSO vectors $\mathbf{b}_{m^*+1-d}^*, \ldots, \mathbf{b}_{m^*+1}^*$ should behave as a random b-dimensional projection, with expected norm $\|\pi_S(\mathbf{v})\| \approx \sqrt{b} \cdot \alpha \cdot q$. On the other hand, BKZ reduction ensures that $\|\mathbf{b}_{m^*+1-d}^*\|$ is the norm of the shortest non-zero vector in the projection of $L(\mathsf{B})$ onto S. Therefore, under these heuristics, if the condition $\|\mathbf{b}_{m^*+1-d}^*\| > \sqrt{b} \cdot \alpha \cdot q$ holds, a failure of the BKZ reduced basis to contain $\mathbf{v}$ implies a contradiction, so we expect that the reduced basis will contain $\mathbf{v}$ when the latter condition holds. Based on the GSA (18), this gives the heuristic attack success condition (with high probability) if

$$f_2(m^*, b) \stackrel{\text{def}}{=} \delta(b)^{2b-1-(m^*+1)} \cdot q^{1-\frac{n'}{m^*+1}} > \sqrt{b} \cdot \alpha \cdot q. \tag{19}$$

For each fixed b, $f_2(m^*, b)$ can be minimized with respect to m^*, with the optimum choice of $m^* = m^*_{\mathsf{opt}}$ being

$$m^*_{\mathsf{opt}}(b)+1 \approx \sqrt{\frac{n' \log q}{\log \delta(b)}} \text{ and } f_2(m^*_{\mathsf{opt}}(b), b) \approx \delta(b)^{2b-1-\sqrt{\frac{n' \log q}{\log \delta(b)}}} \cdot q^{1-\sqrt{\frac{n' \log \delta(b)}{\log q}}}. \tag{20}$$

The expected classical (resp. quantum) log time complexity of the primal embedding attack, according to analysis approach 2 is,

$$\lambda_{C,\mathsf{emb},2} = \log_2(T_{\mathsf{CBKZ}}(b)) \text{ and } \lambda_{Q,\mathsf{emb},2} = \log_2(T_{\mathsf{QBKZ}}(b)) \tag{21}$$

where the classical and quantum BKZ run-time is estimated [LMvdP15] by

$$T_{\mathsf{CBKZ}}(b) = 2^{0.292 \cdot b + o(b)} \text{ and } T_{\mathsf{QBKZ}}(b) = 2^{0.265 \cdot b + o(b)}, \tag{22}$$

respectively, on quantum and classical computing model. Following the conservative 'core-hardness' methodology used in [ADPS16, BCD+16], we estimate the BKZ running-time by only counting the time of a single SVP oracle call.

Optimised primal 'embedding attack' against MP-LWE Our PLWE^f-based complexity lower bounds on MP-LWE in the previous Section reduce from PLWE with a secret polynomial of dimension $\leq n$, but the above 'generic' embedding attack against MP-LWE works on the MP-LWE secret in a larger dimension $n' = n + d' - 1$. Thus there is an apparent complexity gap of $d' - 1$ in the secret vector dimension between those lower and upper bounds. We now explain a simple optimisation of the generic 'embedding attack' against MP-LWE that takes advantage of the 'block Toeplitz' structure of the MP-LWE matrix to give a lower complexity attack on MP-LWE, closing some of this apparent complexity gap.

Our optimised attack is based on the following simple observation about MP-LWE, which allows us to reduce the dimension of the MP-LWE secret when selecting the optimum dimension for the primal attack.

Proposition 2 *For integers q, n, d', t and an error distribution χ_c over $\mathbb{Z}$, let $\chi_c^{d'}$ be the distribution over $\mathbb{Z}^m$ consisting of d' independent samples from χ_c. Then for any d^* with $1 \leq d^* \leq d'$, there is a polynomial time reduction from $\mathsf{MP\text{-}LWE}_{q,n,d',t,\chi_c^{d'}}$ to $\mathsf{MP\text{-}LWE}_{q,n,d^*,t,\chi_c^{d^*}}$.*

Proof Given an instance $(a_i, b_i = a_i \odot_{d'} s + e_i)_{i \leq t} \in (\mathbb{Z}_q^{<n}[x] \times \mathbb{Z}_q^{d'}[x])^t$ of $\mathsf{MP\text{-}LWE}_{q,n,d',t,\chi_{e,\mathsf{MP}}}$ and $d^* \leq n$, the reduction maps it to (a_i, b'_i), where $b'_i = \lfloor \frac{b_i}{x^{d'-d^*}} \rfloor$ consists of the top d^* coefficients of b. Indeed, by Lemma 1, we have $\mathsf{Rev}(\mathbf{b}_i) = \mathsf{Toep}^{d',n}(a_i) \cdot \mathsf{Rev}(\mathbf{s}) + \mathbf{e}_i$, and since the top d^* rows of $\mathsf{Toep}^{d',n}(a_i)$ consists of the smaller Toeplitz matrix $\mathsf{Toep}^{d^*,n}(a_i)$ concatanated with a $d^* \times d' - d^*$ matrix of zeros on the right, we have $\mathsf{Rev}(\mathbf{b}'_i) = \mathsf{Toep}^{d^*,n}(a_i) \cdot \mathsf{Rev}(\mathbf{s}') + \mathbf{e}'_i$, i.e., $b'_i = a_i \odot_{d^*} s' + e'_i$ for $i = 1, \ldots, t$, where $s' = \lfloor \frac{s}{x^{d'-d^*}} \rfloor$ consists of the top $n + d^* - 1$ coefficients of s and $e'_i = \lfloor \frac{s}{x^{d'-d^*}} \rfloor$ consists of the top $n + d^* - 1$ coefficients of e_i (note that the dimension $n + d^* - 1$ of the secret s' is smaller than the dimension $n + d' - 1$ of s). The reduction also maps uniform b_i's to uniform b'_i's. □

To obtain our improved embedding attack on MP-LWE, we apply Proposition 2 with $d^* = m^*/t$ before applying the original embedding attack, where m^* is an optimised dimension for the embedding attack. This gives us LWE instance with respect to a lower dimensional secret $\mathbf{s}' \in \mathbb{Z}_q^{n+m^*/t-1}$ consisting of the top $n+m^*/t-1$ coefficients of the original $n' = n+d'-1$ dimensional MP-LWE secret s. Therefore, our sublattice LWE instance in the optimised attack has the form $(\mathsf{A}^*, \mathbf{b}^* = \mathsf{A}^* \cdot \mathbf{s}^* + \mathbf{e}^*) \in \mathbb{Z}_q^{m^* \times (n+m^*/t-1)} \times \mathbb{Z}_q^{m^*}$, where we also remove the last $d' - m^*/t$ columns of A to form A^*.

The analysis of this attack proceeds identically to the analysis of the generic attack above, replacing generic condition (19) with the MP-LWE-optimised attack success condition

$$f_2(m^*, b) \stackrel{\text{def}}{=} q^{-1/t} \cdot \delta(b)^{2b-1-(m^*+1)} \cdot q^{1-\frac{n-1/t}{m^*+1}} > \sqrt{b} \cdot \alpha \cdot q. \tag{23}$$

Minimising the left-hand side of (19) with respect to the lattice dimension m^* gives the optimum values

$$\begin{cases} m^*_{\mathsf{opt}}(b) + 1 \approx \sqrt{\frac{(n-(1+1/t))\log q}{\log \delta(b)}} \text{ and} \\ f_2(m^*_{\mathsf{opt}}(b), b) \approx q^{-1/t} \cdot \delta(b)^{2b-1-\sqrt{\frac{(n-1/t)\log q}{\log \delta(b)}}} \cdot q^{1-\sqrt{\frac{(n-1/t)\log \delta(b)}{\log q}}} \end{cases} \tag{24}$$

Notice that the optimum sublattice dimension $m^*_{\mathsf{opt}}(b)$ in our optimised MP-LWE attack is the optimum dimension for a primal attack on LWE with a secret of dimension $n-(1+1/t) \approx n$ (versus the unoptimised MP-LWE attack above that corresponds to LWE with secret dimension $n' = n + d' - 1$). Thus this attack closes the 'LWE secret dimension gap' of $d' - 1$ mentioned above. However, the remaining overhead for this attack over the standard embedding attack on LWE in dimension n is the extra factor $q^{1/t}$ in the Hermite Factor function f_2, which is a constant between 3 and 4 for our parameter settings (we refer to Table 7 in Section 5 for concrete security estimates quantifying this gap). We leave it as an open problem to find improved optimised attacks on MP-LWE that also close this remaining gap.

We remark that the 'optimised embedding attack' on MP-LWE as described above only recovers the first $n + m^*/t$ coefficients of the n'-dimensional MP-LWE secret s. But assuming $n > d'$ (this assumption holds for our Titanium parameters in the following Section), this constitutes more than half of the coefficients of s. The remaining coefficients of s can then be recovered by either repeating the attack using the last $n + m^*/t$ coefficients of s (which doubles the run-time), or (at even lower complexity) by solving the remaining $(d' - m^*/t)$-dimensional LWE instance in the last coefficients of s.

5 Titanium: A practical application to MP-LWE

In this Section, we specify our Titanium-CPA and Titanium-CCA algorithms in formats suitable for correctness and security analysis. The design for the IND-CPA secure version of Titanium, to be called here Titanium-CPA, is based on the MP-LWE-based public-key cryptosystem described in Section 4 of [RSSS17].

5.1 Overview of [RSSS17] cryptosystem

The public key consists of t MP-LWE samples of the form $\mathsf{pk} = (a_i, b_i = a_i \odot_{d+k} s + e_i)_{1 \le i \le t}$, where $a_i \in \mathbb{Z}_q^{<n}[x]$ are uniformly random polynomials, $s \in \mathbb{Z}_q^{<n+k+d-1}[x]$ is a uniformly random secret key polynomial, and $e_i \in \mathbb{Z}_q^{<d+k}[x]$ are error polynomials with 'small' coefficients sampled from an appropriate error distribution χ_e, which is a rounded continuous Gaussian distribution in [RSSS17]. The secret key is $\mathsf{sk} = s$. To encrypt a message $m \in \{0,1\}^{<d}[x]$, the encryption algorithm uses an analogue of Regev's encryption scheme [Reg05], computing

$$c_1 = \sum_{1 \le i \le t} r_i \cdot a_i \text{ and } c_2 = \sum_{1 \le i \le t} r_i \odot_d b_i + m \cdot \lfloor q/2 \rfloor,$$

using random polynomials r_i with 'small' coefficients sampled from an appropriate error distribution χ_r, which is uniform on binary coefficients in [RSSS17]. The

decryption algorithm decrypts a ciphertext (c_1, c_2) by exploiting the associativity property of middle-product (Lemma 2); $r \odot_d (a \odot_{d+k} s) = (r \cdot a) \odot_d s$, which implies that the decryption algorithm can compute

$$c_1 - c_2 \odot_d s = \sum_{1 \le i \le t} r_i \odot_d e_i + m \cdot \lfloor q/2 \rfloor \approx m \cdot \lfloor q/2 \rfloor,$$

since $\sum_{1 \le i \le t} r_i \odot_d e_i$ is 'small' compared to $q/2$ with overwhelming probability for appropriate choice of parameters. Then m can be recovered (except with negligible error probability) in decryption by rounding $c_1 - c_2 \odot_d s$ to a multiple of $\lfloor q/2 \rfloor$.

5.2 Titanium parameters

The scheme is an adaptation of Regev's cryptosystem from [Reg09] but adapted to give a security reduction from the MP-LWE problem introduced in [RSSS17]. Our scheme relies on the following parameters and probability distributions:

- **Main Parameter notions**:
  - n - dimension of public key polynomials a_i,
  - k - degree of encryption randomness polynomials r_i,
  - d - dimension of message polynomial $\mathsf{enc}(\mu)$,
  - t - number of public key polynomials a_i,
  - q - ciphertext modulus,
  - p - plaintext modulus,
  - cmp - number of chopped ciphertext least significant bits, and
  - d_1, d_2, d_3 - NTT dimensions.
- **Distribution (χ_e and χ_r) Sampling Parameters**:
  - η - number of trials parameter of BinDiff error distribution χ_e,
  - b_1 - log (base 2) of first χ_r interval half size $B_1/2$,
  - b_2 - log (base 2) second χ_r interval half size $B_2/2$,
  - N_{dec1} - number of coefficients in χ_r in B_1 interval, and
  - N_{dec} - total number of coefficients.

5.3 Titanium-CPA Algorithms

We give the Titanium-CPA algorithms here. Let $\chi_e = (\mathsf{BinDiff}(\eta)^{d+k})^t$, where $\mathsf{BinDiff}(\eta)$ is the 'binomial difference' distribution over $\mathbb{Z}$ with parameter η, i.e., the distribution of the random variable $X - Y$ when random variables X, Y are independently sampled from the binomial distribution with number of trials parameter η, a positive integer, and success probability in each trial parameter $1/2$. Let also $\chi_r = \mathsf{ZeInt U}(B_1)^{N_{\mathsf{dec1}}} \times \mathsf{ZeInt U}(B_2)^{N_{\mathsf{dec}} - N_{\mathsf{dec1}}}$ be the distribution of $(r_1, \ldots, r_t)$, where the first N_{dec1} coefficients (in the concatenated vector of $N_{\mathsf{dec}} = t \cdot (k+1)$ polynomial coefficients) are independently sampled (pseudorandomly) from the zero-excluded interval uniform distribution $\mathsf{ZeInt U}(B_1)$ with even parameter $B_1 = 2^{b_1+1}$, and the remaining $N_{\mathsf{dec}} - N_{\mathsf{dec1}}$ coefficients of $(r_1, \ldots, r_t)$ are independently sampled (pseudorandomly) from the zero-excluded interval uniform distribution $\mathsf{ZeInt U}(B_2)$ with even parameter $B_2 = 2^{b_2+1}$. Then to encrypt a message $m \in \mathbb{Z}_p^{<d}[x]$, we have Algorithm 6.

Algorithm 5 : Titanium-CPA.KeyGen

Input: 1^λ.
Output: pk and sk.
function KeyGen(1^λ)
  Let $s \hookleftarrow U(\mathbb{Z}_q^{<n+d+k-1}[x])$.
  Let $(\bar{a_1}, \ldots, \bar{a_t}) \hookleftarrow U(\mathbb{Z}_q^{<n}[x])^t$.
  Let $(e_1, \ldots, e_t) \hookleftarrow \chi_e \in (\mathbb{Z}_q^{<d+k}[x])^t$.
  for $i \leq t$ **do**
    Let $b_i = \mathsf{Rev}(\bar{a_i}) \odot_{d+k} s + e_i \in \mathbb{Z}_q^{<d+k}[x]$.
  end for
  Let $\mathsf{pk} = ((\bar{a_1}, \ldots, \bar{a_t}), (b_1, \ldots, b_t))$ and $\mathsf{sk} = s$.
end function

Algorithm 6 : Titanium-CPA.Encrypt

Input: pk and m.
Output: $\mathsf{ct} = (c_1', c_2')$.
function Encrypt(pk, m)
  Let $(r_1, \ldots, r_t) \hookleftarrow \chi_r \in (\mathbb{Z}_q^{<k+1}[x])^t$.
  Let $c_1' = \sum_{i=1}^t r_i \cdot \bar{a_i}$
  Let $c_2' = \sum_{i=1}^t \mathsf{Rev}(r_i) \odot_d b_i + \lfloor q/p \rfloor \cdot m \in \mathbb{Z}_q^{<d}[x]$.
end function

The rounding algorithm Round divides each coefficient of its argument polynomial by $\lfloor q/p \rfloor$ and rounds the result to the nearest integer mod q, rounding up if the division is an odd integer multiple of $1/2$ (we assume that q is odd and the argument polynomial coefficients are reduced mod q into the interval $\{-\lfloor q/2 \rfloor, \ldots, \lfloor q/2 \rfloor\}$).

5.3.1 Differences between Titanium-CPA *and the cryptosystem in [RSSS17]:*

We now summarize the main differences between Titanium-CPA and the scheme from [RSSS17]:

- *Optimised r_i distribution χ_r:* In [RSSS17], the r_i's are chosen with uniformly random binary coefficients. In Titanium-CPA, we allow the r_i coefficients to be bigger and tune their variance to optimise the resulting key and ciphertext length, as well as the algorithm run-times, for a given security and decryption error probability level. In particular, although increasing the variance of the r_i's implies a corresponding increase in the decryption noise term $\sum_{1 \leq i \leq t} r_i \odot_d e_i$ (which tends to increase the decryption error probability and a corresponding increase in q to compensate), on the other hand the larger entropy of higher variance r_i's reduces the number t of required MP-LWE samples in the public-key to satisfy the LHL entropy condition for the security proof, and a reduced t has a significant improvement on both computation and key length, even if q is increased up to some point. It turns out that optimal values for the variance of the r_i to minimise the public-key length are typically significantly larger than 1.
- *Optimised e_i distribution χ_e:* In [RSSS17], the e_i error (noise) distribution χ_e is chosen as an integer-rounded continuous Gaussian distribution, but sampling from this distribution tends to be computationally expensive. Instead, Titanium-CPA uses a 'binomial difference' distribution BinDiff as also used in New Hope [ADPS16]

Algorithm 7 : Titanium-CPA.Decrypt

Input: sk and ct.
Output: m'.
1: **function** Decrypt(sk, ct)
2: Let $c' = c'_2 - \mathsf{Rev}(c'_1) \odot_d s \in \mathbb{Z}_q^{\leq d}[x]$.
3: Let $m' = \mathsf{Round}(\lfloor q/p \rfloor, c') \in \mathbb{Z}_p^{\leq d}[x]$.
4: **end function**

and Kyber [BDK+17]. This distribution is efficiently sampleable, and approximates a Gaussian distribution. Importantly, our optimisation of the security reduction of [RSSS17] from PLWE^f for f in our ring polynomial family $\mathcal{F}$ to MP-LWE preserves the BinDiff distribution exactly (in shape and variance), so we are still able to provably lower bound the security of Titanium-CPA based on the assumed security of PLWE^f with the BinDiff distribution. As the distribution variance we use 2, which also matches previous choices [BDK+17].

– *Ciphertext length compression*: To reduce the length of our scheme's ciphertext, we also apply a ciphertext compression optimisation technique (used also in previous lattice-based schemes) by chopping off cmp least-significant bits of the coefficients of c_2. This reduces ciphertext length at the cost of a larger decryption error probability. However, as the compression error term is added to the already existing decryption error term, a certain amount of compression can be achieved almost 'for free', i.e., with little effect on the overall decryption error term and hence decryption error probability. Note that we always choose cmp in a manner that while we still meet the probability of error goals, the number of remaining bits in c_2 be a multiple of 8 (for packing/unpacking purposes to one or two bytes).

5.4 Titanium-CCA Algorithms

Our KEM Titanium-CCA applies a variant of the Fujisaki-Okamoto (FO) transform [FO99] from [HHK17] to our IND-CPA encryption scheme Titanium-CPA to turn the latter into an IND-CCA KEM. These algorithms are given in Appendix B.

5.5 Correctness and Security of Titanium

5.5.1 Correctness:

A concrete correctness analysis of Titanium is given in Appendix A, in which we prove and analyse the correctness of our Titanium algorithms using Hoeffding bounds (in contrast to bounds derived from central limit theorem (CLT)). We also explicitly derive the probability of decryption failure p_e.

5.5.2 IND-CPA of Titanium-CPA *from* MP-LWE *hardness:*

We base IND-CPA security of Titanium-CPA on the Middle-Product LWE problem [RSSS17] (MP-LWE). We show that, under appropriate choice of parameters, the

IND-CPA security of Titanium-CPA is as hard as the MP-LWE problem. The proof is based on adapting the Leftover hash Lemma (LHL) based argument from [RSSS17], with relatively mild changes and generalisations to account for the relatively mild differences between Titanium-CPA and the encryption scheme presented in [RSSS17].

Theorem 3 (IND-CPA of Titanium-CPA from MP-LWE, adapted from [RSSS17]) *Assume that*

$$q \text{ is prime }, \tag{25}$$

and the following Leftover Hash Lemma (LHL) condition holds:

$$t \geq \frac{2 \cdot (\log(\Delta_{\mathsf{LHL}}^{-1}) - 1) + (n + d + k) \cdot \log q}{(k+1) \cdot b_{\mathsf{LHL}}}, \tag{26}$$

where

$$b_{\mathsf{LHL}} \stackrel{\text{def}}{=} \rho \cdot (b_1 + 1) + (1 - \rho) \cdot (b_2 + 1), \tag{27}$$

and

$$\rho \stackrel{\text{def}}{=} \frac{N_{\mathsf{dec1}}}{N_{\mathsf{dec}}}, \text{ with } N_{\mathsf{dec}} \stackrel{\text{def}}{=} (k+1) \cdot t. \tag{28}$$

Then any IND-CPA *attack against* Titanium-CPA *with run-time T and advantage ε, implies an attack against the* $\mathsf{MP\text{-}LWE}_{q,n,d+k,D_{\alpha q}}$ *problem with run-time*

$$T_{\mathsf{MP\text{-}LWE}} \approx T \tag{29}$$

and distinguishing advantage

$$\varepsilon_{\mathsf{MP\text{-}LWE}} \geq \varepsilon/2 - \Delta_{\mathsf{LHL}}. \tag{30}$$

The proof is given in Appendix C.

5.6 Parameter sets

5.6.1 Parameter selection procedure:

We summarize the main aspects of our parameter selection procedure based on the goals set in Chapter 6 of document in NIST-Titanium [SSZb] and security evaluation as:

- Fix $p = 2$ and $d = 256$ to support 256 bit plaintexts.
- Fix n, and $k + 1 < n$ at multiples of 256 (or slightly smaller), integer t and η satisfying algebraic attack constraint [ADPS16, BCD+16].
- Determine NTT dimensions $\mathsf{d}_1, \mathsf{d}_2, \mathsf{d}_3$ as a multiple of 256 (see Section 3):
  - Let $\beta_1 = \lceil (d+k)/256 \rceil$ and $\mathsf{d}_1 = \beta_1 \cdot 256$.
  - Let $\beta_2 = \lceil (n+k)/256 \rceil$ and $\mathsf{d}_2 = \beta_2 \cdot 256$.
  - Let $\beta_3 = \lceil (n+d+k-1)/256 \rceil$ and $\mathsf{d}_3 = \beta_3 \cdot 256$.
- Pick the smallest q and a b_{LHL} satisfying (for $\mathsf{cmp} = 0$):
  - NTT constraint: $q = 1 \pmod{l \cdot 256}$, where $l = \mathrm{lcm}(\beta_1, \beta_2, \beta_3)$.
  - Leftover hash Lemma (LHL) constraint: $t \geq t_{\mathsf{LO}} + 0.01$, where t_{LO} is the LHL-based lower bound on t in right-hand side (RHS) of (26), where we set Δ_{LHL} according to our goal (see (6.70) in Document in NIST-Titanium [SSZb]).

  - p_e constraint: Upper bound on decryption error probability p_e bounded using RHS of inequality (40) in Appendix A is less than the RHS of our goal (see (6.71) in Document in NIST-Titanium [SSZb]) with a 5% safety margin.
- Let $b_1 = \lfloor b_{\mathsf{LHL}} \rfloor - 1$ and $b_2 = b_1 + 1$ and compute $N_{\mathsf{dec1}} \in \mathbb{Z}$ such that Eq. (27) is satisfied.
- Let $\lambda_{\mathsf{PLWE},C}$ and $\lambda_{\mathsf{PLWE},Q}$ quantum and classical attack log complexities against $\mathsf{PLWE}^{(f)}$ for $f \in \mathcal{F}(n, m', d')$ of minimum degree m' and maximum degree n evaluated in (6.36) in Document in NIST-Titanium [SSZb].
- Choose ciphertext compression parameter $\mathsf{cmp} > 0$ subject to p_e constraint above.
- If $\lambda_{\mathsf{PLWE},C}$ and $\lambda_{\mathsf{PLWE},Q}$ satisfy our security goal (see (6.69) in Document in NIST-Titanium [SSZb]), return parameter set. Else, restart with new n, k, t values.

5.6.2 Recommended parameter sets:

We specify total of 6 different parameters sets Toy64, Lite96, Std128, Med160, Hi192, Super256, intended to correspond to the brute force key search security level of a symmetric key cipher with key bit lengths $64, 96, 128, 160, 192, 256$, respectively. This means that any attack that breaks the security of our scheme must require computational resources comparable to or greater than those required for key search on a block cipher with a $64, 96, 128, 160, 192, 256$-bit key, respectively.

The classical attack gate complexity level goal for the six parameter sets / symmetric-key search security levels, is denoted by λ_C with $\lambda_C \in \{79, 111, 145, 175, 207, 272\}$ corresponding to $\approx 2^{15}$ gates cost for each symmetric-key cipher evaluation. Similarly, the quantum attack gate complexity level goal for the six parameter sets / symmetric-key search security levels, is denoted by λ_Q with $\lambda_Q \in \{106, 140, 170, 202, 233, 298\} - \log_2(\mathsf{MD})$, intended to estimate the circuit gate complexity of quantum key search attacks under the assumption that the quantum attack circuit depth is restricted to MD (denoted by MAXDEPTH in [NISa]).

In the following Tables, we give recommended core and error distribution and randomness, and NTT parameters of the following 6 parameter sets: Toy64, Lite96, Std128, Med160, Hi192, Super256. In Table 2, we specify the core parameters of our Titanium-CPA and Titanium-CCA schemes corresponding to each parameter set. For different MD, we have different goals and minimum achieved security levels. The error distribution χ_e sampling parameter η for both Titanium-CPA and Titanium-CCA is set to 4. We also let $p = 2$ everywhere. In Table 3, we present the NTT and fast middle-product NTT dimensions for each parameter sets of Titanium-CPA and Titanium-CCA. In Table 4, we present the relevant randomness sampling parameters for each parameter sets of Titanium-CPA and Titanium-CCA. In Tables 5-6, we show the goal and achieved p_e of Titanium-CPA and Titanium-CCA schemes, respectively. Note that the p_e goal for Titanium-CPA schemes are set to 2^{-30} and p_e's for Titanium-CCA are with respect to $\mathsf{MD} = 40$.

5.6.3 Best known attacks on Titanium-CPA/Titanium-CCA *- Complexity estimates:*

We summarize in Table 7 the computed complexities of best known attacks, $\lambda_{\mathsf{bstatk}}$, on PLWE instances corresponding to our scheme parameter sets. The table also includes, for comparison, the PLWE complexity goals for achieving our target scheme

Table 2 Determined Titanium-CPA and Titanium-CCA core parameters.

CPA Params	Toy64	Lite96	Std128	Med160	Hi192	Super256
n	684	800	1024	1280	1536	2048
k	255	479	511	511	767	1023
d	256	256	256	256	256	256
t	10	8	9	9	7	7
q	240641	84481	86017	301057	737281	1198081
cmp	10	9	9	11	12	13
CCA Params	Toy64	Lite96	Std128	Med160	Hi192	Super256
n	684	800	1024	1280	1536	2048
k	255	479	511	511	767	1023
d	256	256	256	256	256	256
t	10	9	10	10	8	8
q	471041	115201	118273	430081	783361	1198081
cmp	11	9	9	11	12	13

Table 3 The NTT parameters for Titanium-CPA and Titanium-CCA.

CPA Params	Toy64	Lite96	Std128	Med160	Hi192	Super256
d_1	512	768	768	768	1024	1280
d_2	1024	1280	1536	1792	2304	3072
d_3	1280	1536	1792	2048	2560	3328
CCA Params	Toy64	Lite96	Std128	Med160	Hi192	Super256
d_1	512	768	768	768	1024	1280
d_2	1024	1280	1536	1792	2304	3072
d_3	1280	1536	1792	2048	2560	3328

Table 4 The randomness χ_r sampling params for Titanium-CPA and Titanium-CCA.

CPA Params	Toy64	Lite96	Std128	Med160	Hi192	Super256
b_1	7	5	5	7	8	8
b_2	8	6	6	8	9	9
N_{dec}	2560	3840	4608	4608	5376	7168
N_{dec1}	1488	1496	2568	3816	3384	3848
CCA Params	Toy64	Lite96	Std128	Med160	Hi192	Super256
b_1	7	5	4	6	7	7
b_2	8	6	5	7	8	8
N_{dec}	2560	4320	5120	5120	6144	8192
N_{dec1}	328	4168	208	2248	4704	5904

Table 5 The target and achieved probability of error p_e of Titanium-CPA scheme.

CPA Param.	Toy64		Lite96		Std128		Med160		Hi192		Super256	
	Goal	Ach.	Goal	Ach.	Goal	Ach.	Goal	Ach.	Goal	Ach.	Goal	Ach.
$\log_2(p_e^{-1})$	30	30	30	31	30	33	30	41	30	37	30	72

Table 6 The target and achieved probability of error p_e of Titanium-CCA scheme.

CCA Param.	Toy64		Lite96		Std128		Med160		Hi192		Super256	
	Goal	Ach.	Goal	Ach.	Goal	Ach.	Goal	Ach.	Goal	Ach.	Goal	Ach.
$\log_2(p_e^{-1})$	79	85	111	158	143	161	175	199	206	218	271	354

Table 7 Best known attack complexity (MP-LWE Core-SVP or Brute force/Grover) on Titanium-CPA/Titanium-CCA.

Par. Set	**Classical**				**Quantum**			
	$\lambda_{\mathsf{bstatk}}$	$\lambda_{\mathsf{PLWE},m_{\max}}$	$\lambda_{\mathsf{PLWE},m_{\min}}$	λ_C	$\lambda_{\mathsf{bstatk}}$	$\lambda_{\mathsf{PLWE},m_{\max}}$	$\lambda_{\mathsf{PLWE},m_{\min}}$	λ_Q
CCA, Toy64	125	90	85	79	113	83	78	66
CPA, Toy64	134	97	91	79	121	89	84	66
CCA, Lite96	181	129	123	111	164	118	113	98
CPA, Lite96	194	133	127	111	176	122	116	98
CCA, Std128	236	176	149	143	214	161	136	130
CPA, Std128	251	182	171	143	228	166	156	130
CCA, Med160	272	205	195	175	245	187	178	162
CPA, Med160	272	211	201	175	245	194	184	162
CCA, Hi192	272	243	233	207	245	222	214	193
CPA, Hi192	272	244	235	207	245	224	215	193
CCA, Super256	272	333	323	272	245	305	296	258
CPA, Super256	272	333	327	272	245	305	299	258

security levels based on our parameter selection approach. In this Table, the classical columns give the corresponding claimed PLWE complexities $\lambda_{\mathsf{PLWE},m_{\min}}$ and $\lambda_{\mathsf{PLWE},m_{\max}}$ for PLWE with dimensions $m_{\min}$ and $m_{\max}$ corresponding to the minimum and maximum degrees of polynomials in our family $\mathcal{F}$. The claimed quantum PLWE complexities $\lambda_{\mathsf{PLWE},m_{\min}}$ and $\lambda_{Q,\mathsf{PLWE},m_{\max}}$ for PLWE with dimensions $m_{\min}$ and $m_{\max}$ computed based on our dual attack 'core SVP hardness' methodology (which we recall, assumes conservatively, an unlimited quantum circuit depth).

5.6.4 Comparison of Titanium *with other lattice-based schemes:*

Table 8 shows a brief comparison of Titanium-CPA and Titanium-CCA with other lattice-based schemes showing how ours offer an 'intermediate' point in terms of security guarantees versus efficiency.

In particular, we point out the following:

– *Efficiency Aspects:* For our Std128 parameter set, our Titanium-CPA ciphertexts are significantly smaller in size (3.2 times factor) compared to the LWE-based IND-CPA scheme Frodo [BCD+16] at a higher security level. In handshake protocols, the quantity $|\mathsf{pk}| + |\mathsf{ct}|$ is the main communication size, for which we could save ≈ 4.3 Kilo Bytes (KB) compared to Frodo [BCD+16]. Our key generation, encryption, and decryption time are also faster by factors of 1.8, 2.7, and 1.5 compared to Frodo [BCD+16], respectively. Note that we could even save more in these efficiency aspects once AVX2 optimisation techniques are employed. The p_e for Frodo is set to be 2^{-30}, while ours is 2^{-33}.

We also compare the efficiency aspects of Titanium-CCA (and its AVX2 optimised version) to that of Kyber [BDK+17] and FrodoKEM [ABD+17] (and their AVX2 version, respectively). With a smaller quantum security claim, our ciphertexts, secret key, and public key are 3, 6.9, and 15 times larger than the corresponding quantities in Kyber. Our key generation, encapsulation, and decapsulation times are slower by factors of 6.5, 4.3, and 4.4 compared to Kyber [BDK+17], respectively. However in case of FrodoKEM, where we take only the plain cSHAKE128 implementation from [ABD+17] (for a more fair comparison with Titanium, that uses SHAKE256), we see that with a higher quantum security claim, our ciphertexts, secret key, and public key are .5, 1.2, and 2.7 times smaller than the corresponding quantities in FrodoKEM. Our key generation, encapsulation, and decapsulation times are faster by factors of 4.5, 6.2, and 5.4 compared to FrodoKEM [ABD+17], respectively. An implementation with AES instructions and an AVX2 optimised version can also be found in [SSZb] and compared accordingly. Note that the claimed quantum security for Titanium is based on complexity of lattice attacks against the PLWE problem, which is a *lower bound* on the security of Titanium. However, the best known lattice attacks against Titanium (breaking MP-LWE problem) have a much higher complexity, while the claimed quantum security for Frodo, FrodoKEM, and Kyber correspond to actual attacks on the schemes (breaking the PLWE or LWE problem). Thus, our claims for Titanium may actually be very pessimistic in comparison.

– *Security Guarantees:* We have qualitatively achieved/provided better security proof guarantees than other structured (RLWE-based) schemes; Titanium-CPA security is provably (and tightly) as hard as the hardest instance of PLWE in a family of polynomial rings of size at least 3^{256}, hedging against weakness of a few special (e.g. cyclotomic) rings, whereas Kyber [BDK+17] relies on Module-RLWE over a single specific power-of-2 cyclotomic ring in dimension 256.

A well known way of partially protecting encryption schemes against future improvements in attack complexity is to simply take a large security 'safety margin'. Namely, by scaling up the security parameters of the scheme to increase the bit-security of a scheme to s times the desired value λ against *currently* best known attacks, we protect against future attack bit-security improvements by up to a factor s. As an additional comparison with our scheme, in the last two rows of Table 8, we give the parameters for two such 'scaled up' versions of Kyber [BDK+17], Kyber6912 and Kyber2302, whose security parameter (the underlying module lattice rank m) was scaled up to give approximately the same $|\mathsf{pk}| + |\mathsf{ct}|$ size and $|\mathsf{ct}|$ size, respectively as Titanium-CCA.Std128 (we also change the other parameters to approximately preserve the decryption error probability; the Kyber6144 parameter set is $(n, m, ks, ke, q, rqk, rqc, rq2) = (256, 24, 3, 3, 15361, 2^{12}, 2^{12}, 2^3)$ and the Kyber2304 parameter set is $(n, m, ks, ke, q, rqk, rqc, rq2) = (256, 9, 3, 3, 15361, 2^{12}, 2^{12}, 2^3)$). The 'quantum security' columns shows the security achieved by these variants against best known primal attacks, computed using a modified version of the python script supplied by the Kyber authors [DLL+]. We conclude that Kyber6912 and Kyber2304 achieves the 128-bit security goal as long as the bit complexity of attacks on Module-LWE of rank $k = 27$, (respectively $k = 9$) over the specific cyclotomic ring $\mathbb{Z}[x]/(x^{256}+1)$ improve in future by a factor less than ≈ 12.1 (respectively ≈ 4.1). In contrast, with approximately the same communication costs, Titanium-CCA.Std128 achieves the 128-bit security goal even if future attacks for module LWE over cy-

Table 8 Comparison of Titanium-CPA and Titanium-CCA with Frodo and Kyber. Benchmark CPUs used: Intel Xeon E5 2.6GHz (Frodo), Intel i7-6700 (FrodoKEM), Intel i7-7700K (Titanium) and Intel i7-4770K (Kyber).

Scheme	Quantum Security	Security Guarantees		Efficiency Aspects	
		Problem	Family Size	Size (Bytes)	Cycles
Frodo [ABD+17]	130	LWE	n/a	$\|pk\| = 11296$ $\|sk\| = 11280$ $\|ct\| = 11288$	KeyGen : 2938000 Encrypt : 3484000 Decrypt : 338000
Titanium-CPA.Std128	155	MP-LWE	$\geq 3^{256}$	$\|pk\| = 14720$ $\|sk\| = 32$ $\|ct\| = 3520$	KeyGen : 1619550 Encrypt : 1262047 Decrypt : 217612
Titanium-CPA.Std128 (AVX2 optimised)	155	MP-LWE	$\geq 3^{256}$	$\|pk\| = 14720$ $\|sk\| = 32$ $\|ct\| = 3520$	KeyGen : 828542 Encrypt : 742541 Decrypt : 116311
Kyber [BDK+17]	161	Module-LWE	n/a	$\|pk\| = 1088$ $\|sk\| = 2368$ $\|ct\| = 1184$	KeyGen : 276720 Encaps. : 332800 Decaps. : 376104
Titanium-CCA.Std128	134	MP-LWE	$\geq 3^{256}$	$\|pk\| = 16352$ $\|sk\| = 16384$ $\|ct\| = 3552$	KeyGen : 1806119 Encaps. : 1446751 Decaps. : 1671578
FrodoKEM-640-cSHAKE [ABD+17]	103	LWE	1	$\|pk\| = 9616$ $\|sk\| = 19872$ $\|ct\| = 9736$	KeyGen : 8297000 Encaps. : 9082000 Decaps. : 9077000
Kyber [BDK+17] (AVX2 optimised)	161	Module-LWE	1	$\|pk\| = 1088$ $\|sk\| = 2368$ $\|ct\| = 1184$	KeyGen : 77892 Encaps. : 119652 Decaps. : 125736
Titanium-CCA.Std128 (AVX2 optimised)	134	MP-LWE	$\geq 3^{256}$	$\|pk\| = 16352$ $\|sk\| = 16384$ $\|ct\| = 3552$	KeyGen : 934051 Encaps. : 865352 Decaps. : 986905
FrodoKEM-640-cSHAKE (AVX2 optimised)	103	LWE	n/a	$\|pk\| = 9616$ $\|sk\| = 19872$ $\|ct\| = 9736$	KeyGen : 4212000 Encaps. : 4671000 Decaps. : 4672000
Kyber6144 [BDK+17]	1557	Module-LWE	n/a	$\|pk\| = 9248$ $\|sk\| = 20064$ $\|ct\| = 9312$	n/a n/a n/a
Kyber2304 [BDK+17]	521	Module-LWE	n/a	$\|pk\| = 3488$ $\|sk\| = 7584$ $\|ct\| = 3552$	n/a n/a n/a

clotomic rings improve by an *arbitrary factor*, as long as the complexity of PLWE over *some* non-cyclotomic ring in our family remains approximately unchanged. Our approach therefore offers security guarantees against 'security collapse' due to breakthrough advances in cryptanalysis of specific rings. Such guarantees cannot be practically achieved using simple scaling up of schemes based on fixed rings.

References

[ABD+17] E. Alkim, J. W. Bos, L. Ducas, P. Longa, I. Mironov, M. Naehrig, V. Nikolaenko, C. Peikert, A. Raghunathan, D. Stebila, K. Easterbrook, and B. LaMac-

chia. FrodoKEM learning with errors key encapsulation. https://frodokem.org/files/FrodoKEM-specification-20171130.pdf, 2017.

[AD17] M. R. Albrecht and A. Deo. Large modulus ring-lwe >= module-lwe. *IACR Cryptology ePrint Archive*, 2017:612, 2017.

[ADPS16] E. Alkim, L. Ducas, T. Pöppelmann, and P. Schwabe. Post-quantum key exchange - A new hope. In *25th USENIX Security Symposium, USENIX Security 16, Austin, TX, USA, August 10-12, 2016.*, pages 327–343, 2016.

[AFG13] M. R. Albrecht, R. Fitzpatrick, and F. Göpfert. On the efficacy of solving LWE by reduction to unique-svp. In *Information Security and Cryptology - ICISC 2013 - 16th International Conference, Seoul, Korea, November 27-29, 2013, Revised Selected Papers*, pages 293–310, 2013.

[BCD+16] J. W. Bos, C. Costello, L. Ducas, I. Mironov, M. Naehrig, V. Nikolaenko, A. Raghunathan, and D. Stebila. Frodo: Take off the ring! practical, quantum-secure key exchange from LWE. In *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security, Vienna, Austria, October 24-28, 2016*, pages 1006–1018, 2016.

[BCLvV16] D. J. Bernstein, C. Chuengsatiansup, T. Lange, and C. van Vredendaal. NTRU Prime. Cryptology ePrint Archive, 2016. http://eprint.iacr.org/2016/461.

[BDK+17] J. W. Bos, L. Ducas, E. Kiltz, T. Lepoint, V. Lyubashevsky, J. M. Schanck, P. Schwabe, and D. Stehlé. CRYSTALS - kyber: a cca-secure module-lattice-based KEM. *IACR Cryptology ePrint Archive*, 2017:634, 2017.

[BLM13] S. Boucheron, G. Lugosi, and P. Massart. *Concentration Inequalities: A Nonasymptotic Theory of Independence*. Oxford University Press, 2013.

[BV11] Z. Brakerski and V. Vaikuntanathan. Efficient fully homomorphic encryption from (standard) LWE. In *Proc. of FOCS*, pages 97–106. IEEE Computer Society Press, 2011.

[CDPR16] R. Cramer, L. Ducas, C. Peikert, and O. Regev. Recovering short generators of principal ideals in cyclotomic rings. In *Proc. of EUROCRYPT*. Springer, 2016.

[CDW16] R. Cramer, L. Ducas, and B. Wesolowski. Short Stickelberger class relations and application to Ideal-SVP. *Cryptology ePrint Archive*, 2016. https://eprint.iacr.org/2016/885.

[CIV16] W. Castryck, I. Iliashenko, and F. Vercauteren. Provably weak instances of Ring-LWE revisited. In *Proc. of EUROCRYPT*, pages 147–167. Springer, 2016.

[DKRV17] J-P. D'Anvers, A. Karmakar, S.S. Roy, and F. Vercauteren. SABER: Mod-LWR based KEM. https://csrc.nist.gov/CSRC/media/Projects/Post-Quantum-Cryptography/documents/round-1/submissions/SABER.zip, 2017.

[DLL+] L. Ducas, T. Lepoint, V. Lyubashevsky, P. Schwabe, G. Seiler, and D. Stehlé. Kyber github page.

[DORS08] Y. Dodis, R. Ostrovsky, L. Reyzin, and A. D. Smith. Fuzzy extractors: How to generate strong keys from biometrics and other noisy data. *SIAM J. Comput.*, 38(1):97–139, 2008.

[EHL14] K. Eisenträger, S. Hallgren, and K. Lauter. Weak instances of PLWE. In *Proc. of SAC*. Springer, 2014.

[ELOS15] Y. Elias, K. E. Lauter, E. Ozman, and K. E. Stange. Provably weak instances of Ring-LWE. In *Proc. of CRYPTO*. Springer, 2015.

[FO99] E. Fujisaki and T. Okamoto. Secure integration of asymmetric and symmetric encryption schemes. In *Advances in Cryptology - CRYPTO '99, 19th Annual International Cryptology Conference, Santa Barbara, California, USA, August 15-19, 1999, Proceedings*, pages 537–554, 1999.

[Har14] D. Harvey. Faster arithmetic for number-theoretic transforms. *Journal of Symbolic Computation*, 60:113–119, 2014.

[HHK17] D. Hofheinz, K. Hövelmanns, and E. Kiltz. A modular analysis of the fujisaki-okamoto transformation. Cryptology ePrint Archive, Report 2017/604, 2017. http://eprint.iacr.org/2017/604.

[HQZ04] G. Hanrot, M. Quercia, and P. Zimmermann. The middle product algorithm I. *Appl. Algebra Engrg. Comm. Comput.*, 14(6):415–438, 2004.

[Kan87] R. Kannan. Minkowski's convex body theorem and integer programming. *Math. Oper. Res.*, 12(3):415–440, 1987.

[LM06] V. Lyubashevsky and D. Micciancio. Generalized compact knapsacks are collision resistant. In *Proc. of ICALP*, pages 144–155. Springer, 2006.

[LMvdP15] T. Laarhoven, M. Mosca, and J. van de Pol. Finding shortest lattice vectors faster using quantum search. *Designs, Codes and Cryptography*, 77(2):375–400, 2015.

[LPR10] V. Lyubashevsky, C. Peikert, and O. Regev. On ideal lattices and learning with errors over rings. In *Proc. of EUROCRYPT*, LNCS, pages 1–23. Springer, 2010.

[LS15] A. Langlois and D. Stehlé. Worst-case to average-case reductions for module lattices. *Des. Codes Cryptography*, 75(3):565–599, 2015.

[Lyu16] V. Lyubashevsky. Digital signatures based on the hardness of ideal lattice problems in all rings. In *Proc. of ASIACRYPT*, pages 196–214. Springer, 2016.

[NISa] NIST. NIST post-quantum competition. `http://csrc.nist.gov/groups/ST/post-quantum-crypto/documents/call-for-proposals-final-dec-2016.pdf`. Accessed: 2017-06-13.

[NISb] NIST. SHA-3 standard: Permutation-based hash and extendable-output functions. `http://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.202.pdf`. Accessed: 2017-09-29.

[Pei16] C. Peikert. How not to instantiate Ring-LWE. In *Proc. of SCN*, volume 9841 of *LNCS*, pages 411–430. Springer, 2016.

[Reg05] O. Regev. On lattices, learning with errors, random linear codes, and cryptography. In *Proc. of STOC*, pages 84–93, 2005.

[Reg09] O. Regev. On lattices, learning with errors, random linear codes, and cryptography. *J. ACM*, 56(6), 2009.

[RSSS17] M. Roşca, A. Sakzad, D. Stehlé, and R. Steinfeld. *Middle-Product Learning with Errors*, pages 283–297. Springer International Publishing, 2017.

[RWS17] M. Roşca, A. Wallet, and D. Stehlé. On the ring-lwe and polynomial-lwe problems. Private Communication, 2017.

[SB93] H. V. Sorensen and C. S. Burrus. Efficient computation of the DFT with only a subset of input or output points. *IEEE Transactions on Signal Processing*, 41(3):1184–1200, 1993.

[Sch03] C. P. Schnorr. *Lattice Reduction by Random Sampling and Birthday Methods*, pages 145–156. Springer Berlin Heidelberg, 2003.

[Sei18] Gregor Seiler. Faster AVX2 optimized NTT multiplication for Ring-LWE lattice cryptography. `https://eprint.iacr.org/2018/039.pdf`, 2018.

[SSTX09] D. Stehlé, R. Steinfeld, K. Tanaka, and K. Xagawa. Efficient public key encryption based on ideal lattices. In *Proc. of ASIACRYPT*, pages 617–635. Springer, 2009.

[SSZa] R. Steinfeld, A. Sakzad, and R. K. Zhao. Titanium: Post-quantum public-key encryption and kem algorithms. NIST PQC Standardisation Process submission, available at. Accessed: 2018-05-01.

[SSZb] R. Steinfeld, A. Sakzad, and R. K. Zhao. Titanium: Post-quantum public-key encryption and kem algorithms. `http://users.monash.edu.au/~rste/Titanium.html`. Accessed: 2018-05-01.

A Concrete correctness conditions of Titanium-CPA and computation of p_e

This Section contains the proof of correctness for our Titanium-CPA algorithm and explains our method of computing a numerical provable upper bound on the error probability of decryption, that is also used in our IND-CCA security proof. We first define the concept of δ-correct Titanium-CPA.

Definition 6 Our Titanium-CPA scheme is called δ-correct if for any functions f, we have

$$\Pr\left[\mathsf{Decrypt}(\mathsf{sk},\mathsf{ct}) \neq m \ : \begin{cases} (\mathsf{pk},\mathsf{sk}) \leftarrow \mathsf{KeyGen}; \\ m = f(\mathsf{pk},\mathsf{sk}); \\ \mathsf{ct} \leftarrow \mathsf{Encrypt}(\mathsf{pk},m) \end{cases}\right] \leq \delta. \tag{31}$$

We remark that the above definition of decryption error probability over the choice of both public key and encryption randomness (for any, even key-dependent, messages), matches the definition of δ-correctness in [HHK17], which allows us to apply the security analysis of [HHK17] to the Fujisaki-Okamoto transform applied to Titanium-CPA, which yields our Titanium-CCA scheme.

From now on, we let p_e denotes the LHS of (31). We now analyse the correctness of Titanium-CPA. Let us first expand the main operation in decryption of Titanium-CPA:

$$\begin{aligned}
c' &= c'_2 - \mathsf{Rev}(c'_1) \odot_d s \\
&= \sum_{i=1}^{t} \mathsf{Rev}(r_i) \odot_d b_i + \lfloor q/p \rfloor \cdot m - \mathsf{Rev}\left(\sum_{i=1}^{t} r_i \cdot \bar{a_i}\right) \odot_d s \\
&= \sum_{i=1}^{t} \mathsf{Rev}(r_i) \odot_d (\mathsf{Rev}(a_i) \odot_{d+k} s + e_i) \\
&+ \lfloor q/p \rfloor \cdot m - \sum_{i=1}^{t} \mathsf{Rev}(r_i) \cdot \mathsf{Rev}(a_i) \odot_d s \qquad (32) \\
&= \sum_{i=1}^{t} \mathsf{Rev}(r_i) \cdot \mathsf{Rev}(a_i) \odot_{d+k} s + \sum_{i=1}^{t} \mathsf{Rev}(r_i) \cdot e_i \\
&+ \lfloor q/p \rfloor \cdot m - \sum_{i=1}^{t} \mathsf{Rev}(r_i) \cdot \mathsf{Rev}(a_i) \odot_d s \qquad (33) \\
&= \lfloor q/p \rfloor \cdot m + \sum_{i=1}^{t} \mathsf{Rev}(r_i) \odot_d e_i \in \mathbb{Z}_q^d[x],
\end{aligned}$$

where (32) and (33) are obtained using (1) and Lemma 2, respectively. Therefore, in Decryption algorithm of Titanium-CPA we have

$$\begin{aligned}
m' &= \mathsf{Round}\left(\lfloor q/p \rfloor, c'\right) \\
&= \mathsf{Round}\left(\lfloor q/p \rfloor, \lfloor q/p \rfloor \cdot m + \sum_{i=1}^{t} \mathsf{Rev}(r_i) \odot_d e_i\right) \\
&= m,
\end{aligned}$$

if $\sum_{i=1}^{t} \mathsf{Rev}(r_i) \odot_d e_i$ computed over $\mathbb{Z}_q^d[x]$ (i.e., with reduction mod q) has coefficients smaller than $\lfloor q/p \rfloor/2$, i.e., if

$$\left\| \sum_{i=1}^{t} \mathsf{Rev}(r_i) \odot_d e_i \right\|_\infty < \lfloor q/p \rfloor/2, \qquad (34)$$

with the computations performed over $\mathbb{Z}^d[x]$. We upper bound the probability p_e that (34) does not hold, over the choice of the encryption randomness $(r_1, \ldots, r_t)$ from the distribution χ_r and the choice of key generation errors $(e_1, \ldots, e_t)$ from the distribution χ_e.

We recall that χ_r has the form:

$$\chi_r = \mathsf{ZeInfU}(B_1)^{N_{\mathsf{dec1}}} \times \mathsf{ZeInfU}(B_2)^{N_{\mathsf{dec}} - N_{\mathsf{dec1}}},$$

i.e., the first N_{dec1} integer coefficients of the concatenated coefficient vectors of the r_i's are sampled from $\mathsf{ZeInfU}(B_1)$ and the remaining $N_{\mathsf{dec}} - N_{\mathsf{dec1}}$ coefficients sampled from $\mathsf{ZeInfU}(B_2)$. Also, χ_e samples each integer coefficient of $(e_1, \ldots, e_t)$ from the $\mathsf{BinDiff}(\eta)$ distribution. For $i = 1, 2$, let us define the distributions χ_i over $\mathbb{Z}$ as the distribution of the product (over $\mathbb{Z}$) of a sample from $\mathsf{ZeInfU}(B_i)$ and an independent sample from $\mathsf{BinDiff}(\eta)$. Let us define $\bar{r_i}$ as $\mathsf{Rev}(r_i)$. Then we observe that for each $1 \leq i \leq t$, each coefficient of $\bar{r_i} \odot_d e_i$ is an inner product between a row of $\mathsf{Toep}^{d,k}(\bar{r_i})$ and the coefficient vector $\mathbf{e}_i$ of e_i. Therefore, by the independence of the r_i and e_i coefficients, the distribution of each coefficient of $\sum_{i=1}^{t} \bar{r_i} \odot_d e_i$ is the distribution of a sum $\sum_{i=1}^{N_{\mathsf{dec}}} x_i$ of independent random variables x_i, where x_i is sampled from the distribution χ_i with

$$\chi_i := \begin{cases} \chi_1 & 1 \leq i \leq N_{\mathsf{dec1}}, \\ \chi_2 & N_{\mathsf{dec1}} < i \leq N_{\mathsf{dec}}. \end{cases} \qquad (35)$$

The probability of error $\bar{p_e}$ for any fixed coordinate of the message can therefore be upper bounded as follows:

$$\bar{p_e} = \Pr\left[\sum_{i=1}^{N} x_i \geq \lfloor q/p \rfloor/2\right],$$

with x_i distributed as in (35). Since the x_i's are independent with $\mathbb{E}[x_i] = 0$ for $1 \leq i \leq N_{\mathsf{dec}}$, we have

$$\begin{aligned}
\bar{p_e} &= \Pr\left[\sum_{i=1}^{N_{\mathsf{dec}}} x_i \geq \lfloor q/p \rfloor/2\right] && (36)\\
&= \Pr\left[\exp\left(s \cdot \sum_{i=1}^{N_{\mathsf{dec}}} x_i\right) \geq \exp\left(s \cdot \lfloor q/p \rfloor/2\right)\right] && (37)\\
&\leq \frac{\mathbb{E}\left[\exp\left(s \cdot \sum_{i=1}^{N_{\mathsf{dec}}} x_i\right)\right]}{\exp\left(s \cdot \lfloor q/p \rfloor/2\right)} && (38)\\
&= \frac{\mathbb{E}\left[\prod_{i=1}^{N_{\mathsf{dec}}} \exp\left(s \cdot x_i\right)\right]}{\exp\left(s \cdot \lfloor q/p \rfloor/2\right)}\\
&= \frac{\prod_{i=1}^{N_{\mathsf{dec}}} \mathbb{E}\left[\exp\left(s \cdot x_i\right)\right]}{\exp\left(s \cdot \lfloor q/p \rfloor/2\right)}, && (39)
\end{aligned}$$

where (37) is true because the mapping $x \mapsto \exp(s \cdot x)$ is monotonically increasing, (38) is obtained using Markov inequality [BLM13], and (39) is valid due to the fact that x_i's are independent of each other. Let us further define

$$M_{\chi_j}(s) := \mathbb{E}_{x \hookleftarrow \chi_j}[\exp(s \cdot x)],$$

for $j \in \{1, 2\}$. Therefore, (39) can be re-written as:

$$\bar{p_e} \leq \frac{\prod_{i=1}^{N_{\mathsf{dec}}} \mathbb{E}\left[\exp\left(s \cdot x_i\right)\right]}{\exp\left(s \cdot \lfloor q/p \rfloor/2\right)} = \frac{M_{\chi_1}^{N_{\mathsf{dec1}}}(s) M_{\chi_2}^{N_{\mathsf{dec}} - N_{\mathsf{dec1}}}(s)}{\exp\left(s \cdot \lfloor q/p \rfloor/2\right)}. \quad (40)$$

In order to minimize $\bar{p_e}$, one needs to find s that minimizes (40). Letting

$$f(s) := \frac{M_{\chi_1}^{N_{\mathsf{dec1}}}(s) M_{\chi_2}^{N_{\mathsf{dec}} - N_{\mathsf{dec1}}}(s)}{\exp\left(s \cdot \lfloor q/p \rfloor/2\right)},$$

one can differentiate f to find the critical point s^*, such that $f(s^*) = 0$ minimizing the right hand side of (40). The well-known bi-section method is now used to numerically evaluate s^* and hence $\bar{p_e}^{\mathsf{Hoeffding}}$ such that $\bar{p_e} \leq \bar{p_e}^{\mathsf{Hoeffding}}$. The above analysis and a union bound over the d coordinates of $\sum_{i=1}^{t} \bar{r_i} \odot_d e_i$ ensures that our Titanium-CPA is $p_e^{\mathsf{Hoeffding}} \leq d \cdot \bar{p_e}^{\mathsf{Hoeffding}}$-correct.

Instead of the above Hoeffding approach, one could use CLT heuristic analysis to upper bound (36). In particular, by the independence of the x_i's, we can approximate the distribution of $\sum_{i=1}^{N_{\mathsf{dec}}} x_i$ by a Gaussian distribution with mean μ and standard deviation σ that we can explicitly compute and then use standard Gaussian tail bounds to bound p_e. To be more precise, a straightforward computation using the independence of the x_i, and that the standard deviation of χ_e is $\sqrt{2\eta/4} = \sqrt{\eta/2}$ shows that the standard deviation of $\sum_{i=1}^{N_{\mathsf{dec}}} x_i$ is given by

$$\sigma = \sqrt{(B_{\mathsf{eff}}^2/12 + B_{\mathsf{eff}}/4 + 1/6) \cdot (\eta/2) \cdot N_{\mathsf{dec}}}, \quad (41)$$

where

$$B_{\mathsf{eff}} = \sqrt{\rho B_1^2 + (1-\rho) B_2^2},$$

and

$$\rho = N_{\mathsf{dec1}}/N_{\mathsf{dec}}.$$

Table 9 The values of $z_{\mathsf{Hoeffding}}$ in (44) and z_{clt} defined in (43) for Titanium-CPA.

Parameter	Toy64	Lite96	Std128	Med160	Hi192	Super256
$z_{\mathsf{Hoeffding}}$	7.39	7.45	7.63	8.32	8.06	10.61
z_{clt}	7.55	7.64	7.83	8.58	8.26	10.93

Table 10 The values of $z_{\mathsf{Hoeffding}}$ in (44) and z_{clt} defined in (43) for Titanium-CCA.

Parameter	Toy64	Lite96	Std128	Med160	Hi192	Super256
$z_{\mathsf{Hoeffding}}$	11.42	15.25	15.36	17.00	17.74	22.45
z_{clt}	11.67	15.61	15.69	17.43	18.23	23.26

Using a standard Gaussian tail bound along with union bound over the d coordinates as above, one gets

$$p_e^{\mathsf{clt}} \leq (2d) \cdot \exp(-z_{\mathsf{clt}}^2/2), \tag{42}$$

where

$$z_{\mathsf{clt}} = \lfloor q/p \rfloor/(2\sigma). \tag{43}$$

Furthermore, using union bound one can calculate $z_{\mathsf{Hoeffding}}$ such that the calculated $p_e^{\mathsf{Hoeffding}}$ satisfies the following inequality

$$p_e^{\mathsf{Hoeffding}} \leq (2d) \cdot \exp(-z_{\mathsf{Hoeffding}}^2/2). \tag{44}$$

In Tables 9-10, we compare our derived $z_{\mathsf{Hoeffding}}$ in (44) with that of z_{clt} in (43) for our different parameter sets. The results suggest that our provable Hoeffding bounds on the decryption error probability are close optimal, as they are not much higher than the bounds obtained from the CLT heuristic.

A.1 Concrete correctness condition of Titanium-CCA

We similarly define the following correctness for Titanium-CCA

Definition 7 Our Titanium-CCA scheme is called δ-correct if

$$\Pr[\mathsf{Decrypt}(\mathsf{sk}, \mathsf{ct}) \neq k | (\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{KeyGen}; (k, \mathsf{ct}) \leftarrow \mathsf{Encrypt}(\mathsf{pk})] \leq \delta.$$

As we follow the KEM construction given in [HHK17], the following result is outstanding.

Lemma 6 *If* Titanium-CPA *is δ-correct and* G *and* H *are random oracles, then our* Titanium-CCA *is δ-correct.*

B Titanium-CCA Algorithms

We use hash functions for our Titanium-CCA. Cryptographic Hash functions G and H are modelled as a 'random oracle' in the security analysis, and are instantiated using the SHAKE256 mode in [NISb].

C Leftover Hash Lemma and Proof of Theorem 3

We use the following variant of the Leftover hash Lemma (LHL)[DORS08].

Algorithm 8 : Titanium-CCA.KeyGen

Input: 1^λ.
Output: pk and sk.
function $\mathsf{KeyGen}(1^\lambda)$
  Let $(\mathsf{sk.cpa}, \mathsf{pk.cpa}) = \mathsf{Titanium\text{-}CPA.KeyGen}(1^\lambda)$.
  Let $\mathsf{rdec} \hookleftarrow U(\mathsf{byte}^{32})$.
  Let $\mathsf{sk} = (\mathsf{sk.cpa}, \mathsf{rdec}, \mathsf{pk.cpa})$ and $\mathsf{pk} = \mathsf{pk.cpa}$.
end function

Algorithm 9 : Titanium-CCA.Encrypt

Input: pk.
Output: ct and ss.
function $\mathsf{Encrypt}(\mathsf{pk})$
  Sample $\mathsf{m} \hookleftarrow U(\mathsf{byte}^{32})$.
  Let $(\mathsf{seedenc.cpa}, \mathsf{dcca}) = \mathsf{G}(\mathsf{m}) \in \mathsf{byte}^{32} \times \mathsf{byte}^{32}$.
  Let $\mathsf{ct.cpa} = \mathsf{Titanium\text{-}CPA.Encrypt}(\mathsf{pk}, \mathsf{m})$.
  Let $\mathsf{ct} = (\mathsf{ct.cpa}, \mathsf{dcca})$
  Let $\mathsf{ss} = \mathsf{H}(\mathsf{m}, \mathsf{ct}) \in \mathsf{byte}^{32}$.
end function

Algorithm 10 : Titanium-CCA.Decrypt

Input: sk and ct.
Output: ss.
function $\mathsf{Decrypt}(\mathsf{sk}, \mathsf{ct})$
  Let $\mathsf{m}' = \mathsf{Titanium\text{-}CPA.Decrypt}(\mathsf{sk.cpa}, \mathsf{ct.cpa})$.
  Let $(\mathsf{seedenc.cpa}', \mathsf{dcca}') = \mathsf{G}(\mathsf{m}') \in \mathsf{byte}^{32} \times \mathsf{byte}^{32}$.
  Let $\mathsf{ct.cpa}' = \mathsf{Titanium\text{-}CPA.Encrypt}(\mathsf{pk}, \mathsf{m}')$.
  if $(\mathsf{ct.cpa}', \mathsf{dcca}') = (\mathsf{ct.cpa}, \mathsf{dcca})$ **then**
    Let $\mathsf{ss} = \mathsf{H}(\mathsf{m}', \mathsf{ct})$.
  else
    Let $\mathsf{ss} = \mathsf{H}(\mathsf{rdec}, \mathsf{ct})$.
  end if
end function

Lemma 7 *Let X, Y, Z denote finite sets. Let $\mathcal{H}$ be a universal family of hash functions $h : X \to Y$. Let $f : X \to Z$ be arbitrary. Then for any random variable T taking values in X, we have:*

$$\Delta\left((h, h(T), f(T)), (h, U(Y), f(T))\right) \;\leq\; \frac{1}{2} \cdot \sqrt{\gamma(T) \cdot |Y| \cdot |Z|},$$

where $\gamma(T) = \max_{t \in X} \Pr[T = t]$.

We will apply the LHL to the following universal hash family that arises in our construction.

Lemma 8 (Adapted from [RSSS17]) *Let $q, k, d \geq 2$, q prime, and $\mathsf{Supp}_r \subseteq \mathbb{Z}_q^{<k+1}[x]$. For $(b_i)_i \in (\mathbb{Z}_q^{<d+k}[x])^t$, we let $h_{(b_i)_i}$ denote the map that sends $(r_i)_{i \leq t} \in (\mathsf{Supp}_r)^t$ to $\sum_{i \leq t} r_i \odot_d b_i \in \mathbb{Z}_q^{<d}[x]$. Then the hash function family $(h_{(b_i)_i})_{(b_i)_i}$ is universal.*

Proof Our aim is to show that for $r_1, \ldots, r_t$ not all 0 in Supp_r, we have

$$\Pr_{(b_i)_i, (b_i')_i}\Big[\sum_{i \leq t} r_i \odot_d b_i = \sum_{i \leq t} r_i \odot_d b_i'\Big] = q^{-d}.$$

W.l.o.g. we may assume that $r_1 \neq 0$. By linearity, it suffices to prove that for all $y \in \mathbb{Z}_q^{<d}[x]$,

$$\Pr_{b_1}\Big[r_1 \odot_d b_1 = y\Big] = q^{-d}.$$

Let j be minimal such that the coefficient in x^j of r_1 is non-zero and hence co-prime to q. Then the equation $r_1 \odot_d b_1 = y$ restricted to entries $j+1$ to $j+d$ is a triangular linear system in the coefficients of b_1 with diagonal coefficients invertible mod q. The map $b_1 \mapsto r_1 \odot_d b_1$ restricted to these coefficients of b_1 is hence a bijection. This gives the equality above.

C.1 Proof of Theorem 3

Proof We summarize the modifications of the argument in [RSSS17] and the concrete reduction cost. The proof consists in three games (let p_i be the attacker A's success probability in Game_i).

- Game_0 : The original IND-CPA game.
- Game_1 : Instead of generating $\mathsf{pk} = (\bar{a}_i, b_i)_{i \le t}$ with $b_i = a_i \odot_{d+k} s + e_i \in \mathbb{Z}_q^{<d+k}[x]$ using Titanium-CPA.KeyGen, where we define $a_i = \mathsf{Rev}(\bar{a}_i)$ for $i = 1, \ldots, t$, the challenger sets $b_i \hookleftarrow U(\mathbb{Z}_q^{<d+k}[x])$ independently of a_i.
  We can construct a distinguishing attacker against $\mathsf{MP\text{-}LWE}_{q,n,d+k,D_{\alpha q}}$ given t samples, that has run-time $T_{\mathsf{MP\text{-}LWE}} = T + O(t \cdot (n+d+k) \cdot \log q)$ and distinguishing advantage $\varepsilon_{\mathsf{MP\text{-}LWE}} = |p_1 - p_0|$. Given t MP-LWE samples $(a'_i, b'_i)_{i \le t}$, the MP-LWE attacker computes $\bar{a}_i = \mathsf{Rev}(a'_i)$ and $b_i = b'_i$ for $i = 1, \ldots, t$, and sets $\mathsf{pk} = (\bar{a}_i, b_i)_{i \le t}$ as the public key. If (a'_i, b'_i) have the MP distribution (resp. uniform distribution), then $(\bar{a}_i, b_i)_{i \le t}$ have the correct public key distribution as in Game_0 (resp. Game_1), using the fact that Rev is an injective mapping on $\mathbb{Z}_q^{<n}[x]$.
- Game_2 : Instead of generating the second challenge ciphertext component c_2 as $c'_2 = \sum_{i=1}^{t} \mathsf{Rev}(r_i) \odot_d b_i + \lfloor q/p \rfloor \cdot m \in \mathbb{Z}_q^{<d}[x]$, the challenger sets $c_2 \hookleftarrow U(\mathbb{Z}_q^{<d}[x])$, but leaves $c_1 = \sum_{i \le t} r_i \cdot a_i$ as before. By the Leftover Hash Lemma 7 with $\gamma(T) = B_1^{N_{\mathsf{dec1}}} \cdot B_2^{N_{\mathsf{dec}} - N_{\mathsf{dec1}}}$ the (exponential of) the inverse min-entropy of the input $(\mathsf{Rev}(r_1), \ldots, \mathsf{Rev}(r_t))$ to the universal hash family in Lemma 8, $|Y| = q^d$ the hash output space size, and $|Z| = q^{n+k}$ the size of the leakage space due to c_1, the statistical distance between the distributions of the challenge ciphertext in Game_2 and Game_1 is at most Δ_{LHL} if the condition

$$\frac{1}{2} \cdot \sqrt{B_1^{-N_{\mathsf{dec1}}} \cdot B_2^{-(N_{\mathsf{dec}} - N_{\mathsf{dec1}})} q^{n+d+k}} \le \Delta_{\mathsf{LHL}} \tag{45}$$

  holds, which is equivalent to (26), using the definitions $N_{\mathsf{dec}} \stackrel{\text{def}}{=} (k+1) \cdot t$, $B_1 = 2^{b_1+1}$ and $B_2 = 2^{b_2+1}$.

In the last game, the attacker's view is independent of the encrypted challenge message, so $p_2 = 1/2$. It follows that $|p_0 - p_2| = |p_0 - 1/2| = \varepsilon/2 \le |p_1 - p_0| + |p_2 - p_1| \le \varepsilon_{\mathsf{MP\text{-}LWE}} + \Delta_{\mathsf{LHL}}$, which gives (30).