Round5: Compact and Fast Post-Quantum Public-Key Encryption

Hayo Baan[1], Sauvik Bhattacharya[1], Scott Fluhrer[2], Oscar Garcia-Morchon[1], Thijs Laarhoven[3], Ronald Rietman[1], Markku-Juhani O. Saarinen[4], Ludo Tolhuizen[1], and Zhenfei Zhang[5]

[1] Royal Philips N.V., Netherlands. Email: sauvik.bhattacharya@philips.com
[2] Cisco, USA.
[3] Eindhoven University of Technology, Netherlands.
[4] PQShield Ltd., United Kingdom.
[5] Algorand, USA.

Abstract. We present the ring-based configuration of the NIST submission Round5, a Ring Learning with Rounding (RLWR)- based IND-CPA secure public-key encryption scheme. It combines elements of the NIST candidates Round2 (use of RLWR as underlying problem, having $1 + x + \ldots + x^n$ with $n + 1$ prime as reduction polynomial, allowing for a large design space) and HILA5 (the constant-time error-correction code XEf). Round5 performs part of encryption, and decryption via multiplication in $\mathbb{Z}_p[x]/(x^{n+1} - 1)$, and uses secret-key polynomials that have a factor $(x - 1)$. This technique reduces the failure probability and makes correlation in the decryption error negligibly low. The latter allows the effective application of error correction through XEf to further reduce the failure rate and shrink parameters, improving both security and performance.
We argue for the security of Round5, both formal and concrete. We further analyze the decryption error, and give analytical as well as experimental results arguing that the decryption failure rate is lower than in Round2, with negligible correlation in errors.
IND-CCA secure parameters constructed using Round5 and offering more than 232 and 256 bits of quantum and classical security respectively, under the conservative core sieving model, require only 2144 B of bandwidth. For comparison, similar, competing proposals require over 30% more bandwidth. Furthermore, the high flexilibity of Round5's design allows choosing finely tuned parameters fitting the needs of diverse applications – ranging from the IoT to high-security levels.

Keywords: Lattice cryptography · Learning With Rounding · Prime cyclotomic ring · Public-key encryption · IND-CPA · Error correction

1 Introduction

Standardization bodies such as NIST [29] and ETSI [16,17] are currently in the process of evaluating and standardizing post-quantum cryptography (PQC),

alternative solutions to RSA and elliptic curve cryptography that are secure against quantum computers. Lattice-based cryptography is a prominent branch of post-quantum cryptography that is based on well-studied problems and offers very good performance characteristics.

Motivation. The choice of the underlying polynomial ring greatly affects the performance of schemes based on ideal lattices, i.e., those based on the Ring Learning with Errors (RLWE) [27] and the Ring Learning with Rounding (RLWR) [6] problems. A common choice [9,3] of the polynomial ring to instantiate an RLWE or RLWR problem is $\mathbb{Z}_q[x]/\Phi_{2n}(x)$ where n is a power of 2. Proposals such as [3,8,23,10] using this ring enjoy lower decryption failure rates due to the sparse nature of the $\Phi_{2n}(x)$ leading to lesser noise propagation. However, requiring that n be a power of 2 restricts the choice of n. Proposals such as [5,34] choose instead the $\mathbb{Z}_q[x]/\Phi_{n+1}(x)$ where $\Phi_{n+1}(x) = x^n + x^{n-1} + \ldots + 1$ for $n+1$ a prime, thus offering a much denser design space. However, due to the worse noise propagation in this polynomial, the decryption failure rate of such schemes suffers.

Error correction has been shown to improve the security and performance of ideal lattice based cryptosystems in [18], and has been practically demonstrated in schemes such as [33,19]. We observe that error correction, when $\mathbb{Z}_q[x]/\Phi_{2n}(x)$ is used, can bring limited reduction in bandwidth requirements if n is limited to powers of two. On the other hand, applying error correction in schemes using $\mathbb{Z}_q[x]/\Phi_{n+1}(x)$ can bring major improvements since, if failure probability is improved, then it is relatively easy to find slightly smaller n values that directly reduce bandwidth requirements. However, as we will see, multiplications in $\mathbb{Z}_q[x]/\Phi_{n+1}(x)$ lead to correlated decryption errors that limit the application of error correction.

Contributions. In this paper, we present the ring version of the Round5 cryptosystem submitted to NIST. Round5 builds upon the rounding-based Round2 [5] scheme, that is constructed based on the prime-order cyclotomic ring, and *XEf*, the constant-time error correction code in HILA5 [33]. Round2 can finely tune its parameter n for each targeted security level, which in combination with rounding and its characteristically small key-sizes leads to efficient performance. However, having a design based on the $\Phi_{n+1}(x)$ polynomial, operational correctness in Round2 suffers from the above mentioned drawbacks.

Our contributions in this work are as follows:

1. We present the RLWR-based Round5 cryptosystem (Sec. 3), that combines the *dense parameter space* offered by the prime-order $\Phi_{n+1}(x)$ cyclotomic polynomial ($n+1$ a prime), with the *low decryption failure rates* typical of the power-of-two $\Phi_{2n}(x)$ polynomial (n a power of two), such as in NewHope [3] and Kyber [8].
   Round5 does this by computing public-keys modulo $\Phi_{n+1}(x)$, such that $n+1$ is a prime (allowing a wide choice for this security parameter), yet computing part of the ciphertext modulo $N_{n+1}(x) = x^{n+1} - 1$ and requiring that

secret-keys are polynomials having a factor $(x-1)$. The latter two ensure that an additional term originating from reductions modulo $\Phi_{n+1}(x)$ in the public-keys vanishes during reduction modulo $N_{n+1}(x)$ in encryption and decryption, leading to a decryption error term that has a noise propagation as low as in the case of the $\Phi_{2n}(x)$ polynomial.

2. We present detailed analytical and experimental results on the decryption error in Round5, especially the occurrence and behavior of correlated errors occurring due to reductions modulo $\Phi_{n+1}(x)$. Our experimental simulations support the claim that the dependence between errors when performing encryption and decryption modulo $N_{n+1}(x)$, although still existent, is negligible; these results are of independent interest and apply also to schemes defined based on the power-of-two $\Phi_{2n}(x)$ polynomial.
3. Based on our above results on independent bit errors when using the $N_{n+1}(x)$ polynomial, we extend the design of Round2 further in Round5 by incorporating the error correction code *XEf*, originally due to [33]. Our choice of this code is motivated by the following.
   Firstly, XEf is designed to easily implement constant-time correction of up to f errors, where f is arbitrary, in practice between 2 and 5, and can be chosen based upon the usage scenario. This flexibility of XEf fits the overall design goals of Round5. In comparison, the only other NIST [29] post-quantum candidate utilizing constant-time error correction is ThreeBears [19], however its Melas code can correct only (up to) 2 errors. Another NIST candidate, LAC [26] uses BCH error correction, for which no obvious constant-time implementation exists [25].
   Secondly, operations in XEf are based on Boolean logic only, and are therefore simple and fast. XEf's performance is therefore at least at par with, if not better, than the constant-time Melas error correction of the ThreeBears [19] submission, which involves multiplication operations in $\mathbb{F}_{2^9}$. However, we note that the performance overhead of error correction is in general, negligible compared to other, more significant overheads in ideal lattice based cryptosystems, such as polynomial ring multiplications.
   Thus, XEf allows Round5 to further drop its decryption failure rate significantly, shrink parameters, and in the process improve security and performance, while remaining flexible enough to optimize its performance when targeting different applications.

2 Background

For each positive integer a, we denote the set $\{0, 1, \ldots, a-1\}$ by $\mathbb{Z}_a$. For a set A, we denote by $a \xleftarrow{\$} A$ that a is drawn uniformly at random from A. For $x \in \mathbb{Q}$, we denote by $\lfloor x \rfloor$ and $\lfloor x \rceil$ rounding downwards to the next smaller integer and rounding to the closest integer (with rounding up in case of a tie) respectively.

Let $n+1$ be prime. The $(n+1)$-th cyclotomic polynomial $\Phi_{n+1}(x)$ then equals $x^n + x^{n-1} + \cdots + x + 1$. We denote the polynomial ring $\mathbb{Z}[x]/\Phi_{n+1}(x)$ by $\mathcal{R}_n$. We denote by $N_{n+1}(x)$ the polynomial $x^{n+1} - 1 = \Phi_{n+1}(x)(x-1)$. For each positive

integer a, we write $\mathcal{R}_{n,a}$ for the set of polynomials of degree less than n with all coefficients in $\mathbb{Z}_a$. We call a polynomial in $\mathcal{R}_n$ *ternary* if all its coefficients are 0, 1 or -1. Throughout this document, regular font letters denote elements from $\mathcal{R}_n$. For each $v \in \mathcal{R}_n$, the Hamming weight of v is defined as its number of non-zero coefficients. We denote with $\mathcal{H}_n(h)$ the set of ternary polynomials of degree less than n, with Hamming weight h.

Round5 as presented in this paper relies on the same underlying problem as in [5] tailored to the ring case. Like [5], Round5 as submitted to NIST relies on the General Learning with Rounding problem.

Definition 1 (Ring Learning with Rounding (RLWR)). *Let n, p, q be positive integers such that $q \geq p \geq 2$. Let $\mathcal{R}_{n,q}$ be a polynomial ring, and let D_s be a probability distribution on $\mathcal{R}_n$. The search version of the RLWR problem $\texttt{sRLWR}_{n,m,q,p}(D_s)$ is as follows: given m samples of the form $\left\langle \left\lfloor \frac{p}{q} \langle as \rangle_q \right\rceil \right\rangle_p$ with $a \in \mathcal{R}_{n,q}$ and a fixed $s \leftarrow D_s$, recover s.*

The decision version of the RLWR problem $\texttt{dRLWR}_{n,m,q,p}(D_s)$ is to distinguish between the uniform distribution on $\mathcal{R}_{n,q} \times \mathcal{R}_{n,p}$ and the distribution $\left(\boldsymbol{a}_i, b_i = \left\langle \left\lfloor \frac{p}{q} \langle as \rangle_q \right\rceil \right\rangle_p\right)$ with $a \xleftarrow{\$} \mathcal{R}_{n,q}$ and a fixed $s \leftarrow D_s$.

We note that the original decisional RLWR assumption [6] is to distinguish from $\mathcal{R}_{n,q} \times \langle \mathcal{R}_{n,q} \rangle_p$. We simplify it to uniform case since $p|q$ in our setting.

Round5 uses XEf, an f-bit majority logic error correcting block code, to decrease the decryption failure rate. The code is built using the same strategy as codes used by TRUNC8 [32] (2-bit correction) and HILA5 [33] (5-bit correction). The XEf code is described by $2f$ "registers" r_i of size $|r_i| = l_i$ with $i = 0, \ldots, 2f-1$. We view the κ-bits payload block m as a binary polynomial $m_{\kappa-1}x^{\kappa-1} + \cdots + m_1 x + m_0$ of length κ. Registers are defined via cyclic reduction $r_i = m \bmod x^{l_i} - 1$. A transmitted message consists of the payload m concatenated with register set r (a total of $\mu = \kappa + xe$ bits, where $xe = \sum l_i$).

Upon receiving a message $(m' \mid r')$ one computes the register set r'' corresponding to m' and compares it to the received register set r' – that may also have errors. Errors are in coefficients m'_k where there are parity disagreements for multitude of registers r_i. We use a majority rule and flip bit m'_k if

$$\sum_{i=0}^{2f-1} ((r'_i[\langle k \rangle_{l_i}] - r''_i[\langle k \rangle_{l_i}]) \bmod 2) \geq f + 1 \tag{1}$$

where the sum is taken as the number of disagreeing register parity bits at k.

3 Round5

The core of Round5 is *r5_cpa_pke*, an IND-CPA secure public-key encryption scheme based on the Ring Learning with Rounding (RLWR) problem. r5_cpa_pke is constructed as a *noisy El Gamal encryption* scheme similar to the works in [24]

Algorithm 1: r5_cpa_pke_keygen()

$a \xleftarrow{\$} \mathcal{R}_{n,q}$

$s \xleftarrow{\$} \mathcal{H}_n(h)$

$b = \left\langle \left\lfloor \frac{p}{q} \left(\langle as \rangle_{\Phi_{n+1}(x)} + h_1 \right) \right\rfloor \right\rangle_p$

return $(pk = (a, b), sk = s)$

Algorithm 2: r5_cpa_pke_encrypt(pk, m)

$r \xleftarrow{\$} \mathcal{H}_n(h)$

$u = \left\langle \left\lfloor \frac{p}{q} \left(\langle ar \rangle_{\Phi_{n+1}(x)} + h_1 \right) \right\rfloor \right\rangle_p$

$v = \left\langle \left\lfloor \frac{t}{p} \left(\text{Sample}_\mu \langle br \rangle_{\xi(x)} + h_1 \right) \right\rfloor + \frac{t}{2} \texttt{xef\_compute}_{\kappa,f}(m) \right\rangle_t$

return $ct = (u, v)$

Algorithm 3: r5_cpa_pke_decrypt(sk, ct)

$v_p = \frac{p}{t} v$

$y = \left\langle \left\lfloor \frac{2}{p} \left(v_p - \text{Sample}_\mu \langle su \rangle_{\xi(x)} + h_2 \right) \right\rfloor \right\rangle_2$

$\hat{m} = \texttt{xef\_correct}_{\kappa,f}(y)$

return $\hat{m}$

and [4]. Public keys are noisy RLWR samples in $\mathbb{Z}[x]/\Phi_{n+1}(x)$, computed via a lossy rounding down to a smaller modulus.

Round5 and its core r5_cpa_pke builds on Round2 [5], specifically the building block CPA-PKE. r5_cpa_pke is thus described in Algorithms 1, 2 and 3, which it inherits from the ring variant of CPA-PKE, along with the cryptosystem parameters, positive integers n, h, p, q, t, μ, f, τ, and a security parameter κ. The moduli q, p, t are powers of 2, such that $t|p|q$. It is required that $p^2 \geq qt$ (see Sec. 5.1), $\mu \leq n$ and $\mu \geq \kappa$. h is the Hamming weight of secret polynomials. r5_cpa_pke also defines a generic polynomial $\xi(x) \in \{N_{n+1}(x), \Phi_{n+1}(x)\}$, which is used to reduce the result of polynomial multiplication during encryption and decryption. In this paper, we discuss performance (in the form of decryption failure behavior) and security trade-offs and requirements for the cases that $\xi(x) = N_{n+1}(x)$ and $\xi(x) = \Phi_{n+1}(x)$.

Algorithm 1 first samples a public polynomial a with coefficients in $\mathbb{Z}_q$, a secret-key polynomial s and computes the public-key polynomial b by rounding its coefficients (to the closest integer) to a smaller modulus $p < q$. Here, rounding is described in terms of rounding downwards, and addition of a rounding constant $h_1 = q/2p$. In Algorithm 2, the encryptor samples an ephemeral secret encryption randomness r and uses it along with a to compute the first ciphertext component u similar to b. The second ciphertext component v is computed using the public-key b and r to obtain a RLWR sample, which is then used as a one-time pad to encrypt the message (which is additionally encoded using an error correction code). Finally, the decryptor in Algorithm 3 computes $\langle su \rangle_{\xi(x)} \approx \langle br \rangle_{\xi(x)}$ and recovers the message. The rounding constant $h_2 = p/2t + p/4 - q/2p$ is used here to remove bias in the decryption error.

Since not all coefficients of v are needed to encrypt a κ bit message, encryption uses the function $\text{Sample}_\mu : c \in \mathcal{R}_{n,p} \to \mathbb{Z}_p^\mu$, whose output corresponds to the μ lowest order polynomial coefficients of c: $c_0 + c_1 x + \cdots + c_{\mu-1} x^{\mu-1}$. The use of Sample_μ makes encryption and decryption more efficient since only μ coefficients

need to be computed in the ciphertext instead of all n. This also improves the failure probability since the encryptor and decryptor need to agree on fewer symbols. Further, this also requires sending fewer symbols, reducing bandwidth required.

The integer f denotes the error-correction capability of a code $Xef_{\kappa,f} \subset \mathbb{Z}_2^{\mu}$. We have an encoding function $\texttt{xef\_compute}_{\kappa,f} : \{0,1\}^{\kappa} \rightarrow Xef_{\kappa,f}$ and a decoding function $\texttt{xef\_correct}_{\kappa,f} : \mathbb{Z}_2^{\mu} \rightarrow \{0,1\}^{\kappa}$ such that for each $m \in \{0,1\}^{\kappa}$ and each error $e = (e_0, \ldots, e_{\mu-1})$ with at most f bits equal to 1

$$\texttt{xef\_correct}_{\kappa,f}(\texttt{xef\_compute}_{\kappa,f}(m) + e) = m. \tag{2}$$

Secret-keys in Round5 are *sparse*, *ternary* and *balanced*, i.e., they are polynomials of degree at most $(n-1)$, exactly $h/2$ coefficients of which are $+1$, $h/2$ are -1, and the rest zero. Having a fixed weight (sparse) reduces probability of decryption failure and makes computations faster. The latter is also helped by the fact that non-zero components are either $+1$ or -1 (ternary), implying that multiplications can be accomplished using only additions and subtractions. Finally, having an equal number of $+1$'s and -1's (balanced) ensures that the secret-keys have a factor $(x-1)$. Section 4 analyzes how this ensures that decryption errors are not correlated, allowing error correction to be used in Round5. As an additional benefit, the decryption failure rate remains low and at the level of $x^{2^{\kappa}} + 1$ cyclotomic polynomials, despite using reductions modulo $\Phi_{n+1}(x)$ to compute public-keys.

As a final note, the NIST submission Lizard [10,11] also uses sparse, ternary secret-keys, and similar to our proposal enjoys the resulting benefits in decryption failure probability and computational efficiency. However, Lizard (specifically, its ring-based instantiation RLizard) uses Φ_{2n} (for n a power of 2) as the reduction polynomial. It thus does not require balanced secret-keys and our technique for reducing error correlations; however, its ring choice limits its parameter choices and design space.

4 Correctness analysis

In this section, the decryption failure behavior of r5_cpa_pke is analyzed. We first present a sufficient condition for correct decryption. We then analyze the probability of this condition not being satisfied and describe how we evaluated this decryption failure probability.

Sufficient condition for correctness. Let $\Delta = (h_1 + h_2)1_{\mu} - i_v + \text{Sample}_{\mu}(\langle\langle br - su \rangle\rangle_{\xi})$, where $\frac{t}{p} i_v(x)$ represents the error introduced in the ciphertext component $v(x)$ due to rounding downwards; each coefficient of $i_v(x)$ is in $\mathbb{Z}_{p/t}$, and 1_a is the polynomial of degree $a-1$ with all coefficients equal to 1. As shown in Appendix A, if the i-th coefficient of the polynomial y in decryption and the i-th coefficient of $\texttt{xef\_compute}_{\kappa,f}(m)$ do not agree, then

$$\left\langle \frac{q}{p} \Delta_i \right\rangle_q \in \left[\frac{q}{4}, q - \frac{q}{4} \right]. \tag{3}$$

Decryption failure probability. The probability of decryption failure in coefficient i before error correction is thus at most the probability that (3) is satisfied. We write $b \equiv \frac{p}{q}(\langle as\rangle_{\Phi_{n+1}} + h_1 1_n) - i_b$ with all coefficients of i_b in $[0, 1)$. We thus have that $\frac{q}{p}b \equiv \langle as\rangle_{\Phi_{n+1}} + j_b \pmod q$ with all coefficients of $j_b = h_1 1_n - i_b$ in $I = (-\frac{q}{2p}, \frac{q}{2p}] \cap \mathbb{Z}$. Similarly, $\frac{q}{p}u \equiv \langle ar\rangle_{\Phi_{n+1}} + j_u \pmod q$ with all components of j_u in I. We thus can write

$$\frac{q}{p}(br - su) \equiv \langle sa\rangle_{\Phi_{n+1}} r - s\langle ar\rangle_{\Phi_{n+1}} + j_b r - s j_u \pmod q. \tag{4}$$

Obviously, if $\xi = \Phi_{n+1}$, then $\langle sa\rangle_{\Phi_{n+1}} r - s\langle ar\rangle_{\Phi_{n+1}} \equiv 0 \pmod \xi$. The same is true if $\xi = N_{n+1}$ and r and s both are multiples of $(x-1)$. This is so as there are $\lambda_s, \lambda_r \in \mathbb{Z}[x]$ such that $\langle as\rangle_{\Phi_{n+1}} r - s\langle ar\rangle_{\Phi_{n+1}} = \lambda_s \Phi_{n+1}(x) r(x) - s\lambda_r \Phi_{n+1}$. As $(x-1)$ divides s and r, both $\Phi_{n+1} r$ and $s\Phi_{n+1}$ are divisible by N_{n+1}. As a result, for $\xi \in \{\Phi_{n+1}, N_{n+1}\}$ we have that

$$\frac{q}{p}\Delta \equiv j_v + \text{Sample}_\mu\left(\langle j_b r - s j_u\rangle_\xi\right) \pmod q. \tag{5}$$

In our analysis below, the coefficients of j_b and j_u are drawn independently and uniformly from I, and the coefficients of j_v are drawn independently and distributed as $\frac{q}{p}y$ with y uniform on $(-\frac{p}{2t}, \frac{p}{2t}] \cap \mathbb{Z}$.

4.1 Computing failure probability when $\xi = \Phi_{n+1}$

We now combine (3) and (5) for the case that $\xi = \Phi_{n+1}$. As $N_{n+1}(x)$ is a multiple of $\Phi_{n+1}(x)$, we have that $\langle f\rangle_{\Phi_{n+1}} = \langle\langle f\rangle_N\rangle_{\Phi_{n+1}}$. Moreover, if $g(x) = \sum_{i=0}^n g_i x^i$, then $\langle g\rangle_{\Phi_{n+1}} = g - g_n \Phi_{n+1}$. In particular, for all polynomials s, e,

$$\text{if } \langle se\rangle_N = \sum_{k=0}^{n} c_k(s,e) x^k, \text{ then } \langle se\rangle_{\Phi_{n+1}} = \sum_{k=0}^{n-1} (c_k(s,e) - c_n(s,e)) x^k, \tag{6}$$

Hence, if the i-th bit is not retrieved correctly, then

$$\langle (j_v(x))_i + c_i(j_b, r) - c_n(j_b, r) - c_i(s, j_u) + c_n(s, j_u)\rangle_q \in \left[\frac{q}{4}, q - \frac{q}{4}\right]. \tag{7}$$

Assuming independence, and taking into account that r and s contain $h/2$ ones and $h/2$ minus ones, $c_k(j_b, r) - c_n(j_b, r) - c_k(s, j_u) + c_n(s, j_u)$ is distributed as the difference of $2h$ independent random variables on I, minus the sum of $2h$ independent random variables on I. The probability that (7) is satisfied thus can be computed explicitly. By the union bound, the probability that at least one of the μ symbols is not retrieved correctly is at most μ times the probability that (7) is satisfied.

4.2 Correlation in decryption errors when $\xi = \Phi_{n+1}$

A basic requirement for using XEf error correction code is that the errors it aims to correct are independent. However, the condition in (7) for a decryption error in position i shows terms $c_n(j_b, r)$ and $c_n(s, j_u)$ that are common to all positions

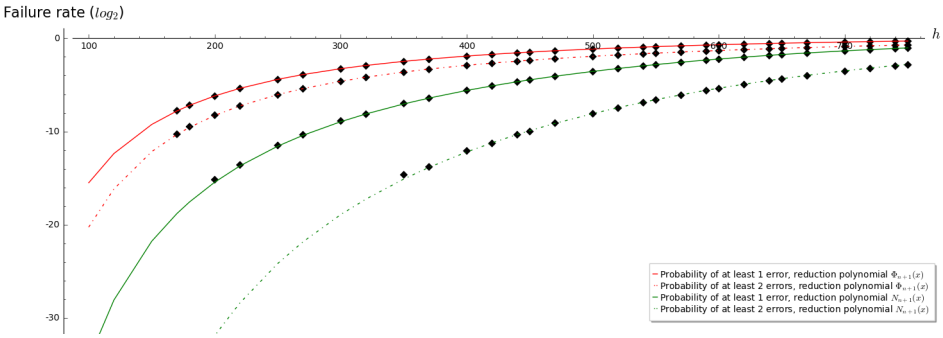

Fig. 1: Probabilities of at least one (continuous lines) and at least two errors (dotted lines) in Round5 ring parameters, plotted against the Hamming weight of secrets (X-axis), for the reduction polynomials $\Phi_{n+1}(x)$ and $N_{n+1}(x)$. Diamonds represent corresponding probabilities computed from actual Round5 simulations for the same parameters. Scripts for analyzing and reproducing these results can be found at www.round5.org.

i. Figure 1 shows the effect of this dependency, by comparing the estimated probabilities of at least one error and that of at least two errors occurring, when the reduction polynomial $\xi = N_{n+1}$ (as in r5_cpa_pke) and when $\xi = \Phi_{n+1}$ (as in Round2 [5]), respectively. It can be seen that due to correlated errors, the probability of at least two errors occurring when the reduction polynomial is $\xi = \Phi_{n+1}$ is much larger than in the case of the $N_{n+1}(x)$ reduction polynomial. As a consequence, the XEf code cannot be directly employed with the reduction polynomial $\xi = \Phi_{n+1}$ as used in Round2.

For any a, (6) can be used to compute $p(i \mid a)$, the probability that bit i is not retrieved correctly, given that $-c_n(j_b, r) + c_n(s, j_u) \equiv a \pmod q$. We assume that having a bit error in position i, given that $c_n(s, j_u) - c_n(j_b, r) \equiv a$, is independent of having a bit error in another position j, given that $c_n(s, j_u) - c_n(j_b, r) \equiv a$. The probability of having exactly k bit errors, given that $c_n(s, j_u) - c_n(j_b, r) \equiv a$, then equals $\binom{\mu}{k}(p(0 \mid a)^k(1 - p(0 \mid a))^{\mu-k}$. By summing these probabilities over a, weighted with the probability that $c_n(s, j_u) - c_n(j_b, r) \equiv a$, the probability of having exactly k bit errors is obtained. In Figure 1, the result of application of this method is also compared with simulations of scaled-down Round5 parameters; Section 4.4 contains details.

4.3 Computing failure probability when $\xi = N_{n+1}$

Combination of (3) and (5) for $\xi = N_{n+1}$ implies that if an error occurs in position i, then

$$\langle (j_v(x))_i + c_i(j_b, r) - c_i(s, j_u) \rangle_q \in \left[\frac{q}{4}, q - \frac{q}{4}\right]. \tag{8}$$

Note that in order that (8) can be used, it is required that s and r both are multiples of $(x - 1)$, as is the case with Round5.

Assuming independence, and assuming that r and s contain $h/2$ ones and $h/2$ minus ones, $c_i(j_b, r) - c_i(s, j_u)$ is distributed as the sum of h independent uniform random variables on I, minus the sum of h independent uniform random variables on I. The probability that (8) is satisfied thus can be computed explicitly.

Now let the error-correcting code be capable of correcting f symbol errors. Assuming that $c_i(s, e)$ and $c_j(s, e)$ are independent whenever $i \neq j$, the probability of not decoding correctly is at most $\sum_{e \geq f+1} \binom{\mu}{e} p_n^e (1 - p_n)^{\mu - e}$.

4.4 Correlation and Error correction: Experimental results

Figure 1 compares the estimated probabilities of at least one error occurring and that of at least two errors occurring, when $\xi = N_{n+1}$ (as in r5_cpa_pke) and when $\xi = \Phi_{n+1}$ (as in Round2 [5]), respectively. These estimates are computed by explicitly convolving probability distributions. Parameters are simulated *without* error correction, and are $n = 800$, $q = 2^{11}$, $p = 2^7$, $t = 2^4$, $\mu = \kappa = 128$, while the Hamming weight varies between 100 and 750 in order to show its effect on both the bit failure rate and error correlation. The influence of the highest-order coefficients $c_n(s, e)$ common to all coefficients in the Φ_{n+1} case is accounted for as explained in Section 4.2. Clearly, the probability of at least two errors is much higher when multiplications are done modulo Φ_{n+1} instead of N_{n+1}, and in the latter case, this probability is significantly lower than the probability of at least one error. Figure 1 also shows corresponding probabilities of at least one and at least two errors, obtained from simulations of *actual, scaled-down* r5_cpa_pke parameters, showing that the actual behavior closely matches estimates.

To conclude, the effect of dependency due to polynomial multiplication modulo Φ_{n+1} as in Round2 is made negligible by the combined use of polynomial multiplication modulo N_{n+1} and balanced secrets in Round5, allowing the use of forward error correction, resulting in better security and performance.

5 Security analysis

In Section 5.1, we show that if $\xi = \Phi_{n+1}$, then r5_cpa_pke is IND-CPA secure. Section 5.2 details how Round2's use of the function $Sample_\mu$ prevents known distinguishing attacks such as the "*Evaluate at 1*" attack [20]. Next, Section 5.3 extends the IND-CPA security proof in Section 5.1 to a RLWE-variant of r5_cpa_pke, which gives strong confidence in Round5's design. Finally, in Section 5.4 it is discussed why this proof does not directly translate to an RLWR-based design and a simple design change in Round5 that would make it apply, but which is not introduced since it does not bring major benefits from a concrete security viewpoint.

5.1 IND-CPA security of r5_cpa_pke when $\xi = \Phi_{n+1}$

When the reduction polynomial $\xi(x)$ in Round5 equals $\Phi_{n+1}(x)$, then r5_cpa_pke is an IND-CPA secure public-key encryption scheme, under the assumption that

the decision Ring Learning with Rounding (RLWR) problem with sparse-ternary secrets ($\mathtt{dRLWR}_{\mathrm{spt}}$) is hard for the polynomial ring $\mathbb{Z}[x]/\Phi_{n+1}(x)$. [6, Theorem 3.2] proves that the RLWR problem for any distribution on the secrets is hard assuming that the RLWE problem is hard for the same distribution, for a super-polynomial modulus q. This gives confidence in the asymptotic hardness of our scheme's underlying problem.

The below theorem (informal) gives a tight, classical reduction against classical or quantum adversaries in the standard model:

Theorem 1 *For every adversary $\mathcal{A}$ against r5_cpa_pke, there exist distinguishers $\mathcal{B}$ and $\mathcal{C}$ such that, for $z = \max(p, tq/p)$,*

$$Adv^{IND\text{-}CPA}_{r5\_cpa\_pke(\xi=\Phi_{n+1})}(\mathcal{A}) \leq Adv^{\mathtt{dRLWR}_{spt}}_{n,1,q,p}(\mathcal{B}) \; + Adv^{\mathtt{dRLWR}_{spt}}_{n,2,q,z}(\mathcal{C}). \tag{9}$$

The proof of the above theorem follows a similar approach as [13] to equalize the noise ratios q/p and p/t in (the coefficients of) the two ciphertext components u and v, allowing them to be expressed as two RLWR samples with a common secret and noise distribution (with noise ratio q/z). This technique however does not apply if the reduction polynomial ξ in Round5 is N_{n+1}, as is required for the secure usage of (XEf) error correction in Round5 (see Section 4.3).

5.2 Distinguishing attack at $x = 1$ for $\xi = N_{n+1}$

When $\xi = N_{n+1}$ and $\mu = n + 1$, a distinguisher can be built from the evaluation of the ciphertext component $v(x)$ in Algorithm 2 in $x = 1$. This is based on the fact that $(x - 1)$ divides both $r(x)$ and $N_{n+1}(x)$. The attack does not apply if $\mu \leq n$ as in Round5, as the sum of the coefficients of $v(x)$ hidden by Sample_μ is uniformly distributed. Further details can be found in Appendix B.

5.3 IND-CPA security of r5_cpa_pke with $\xi = N_{n+1}$ and independent noise

A variant of r5_cpa_pke where the noise is independently sampled from a given distribution instead of being generated via rounding, is an IND-CPA secure public-key encryption scheme, if the decision Ring LWE problem for $\mathbb{Z}[x]/\Phi_{n+1}(x)$ is hard; this results gives confidence in Round5's RLWR-based design.

Theorem 2 *For every adversary $\mathcal{A}$ against a variant r5_cpa_pke$'$ of r5_cpa_pke where the noise is independently sampled, there exist distinguishers $\mathcal{C}$ and $\mathcal{E}$ such that*

$$Adv^{IND\text{-}CPA}_{r5\_cpa\_pke'(\xi=N_{n+1})}(\mathcal{A}) \leq Adv^{\mathtt{RLWE}(\mathbb{Z}_q[x]/\Phi_{n+1}(x))}_{m=1}(\mathcal{C}) + Adv^{\mathtt{RLWE}(\mathbb{Z}_q[x]/\Phi_{n+1}(x))}_{m=2}(\mathcal{E}). \tag{10}$$

where m denotes the number of RLWE samples available to each distinguisher.

A more detailed version of the above theorem and its proof can be found in Appendix C.

Algorithm 4: round_to_root(a, q, p)

$b \leftarrow \left\lfloor \frac{p}{q} a \right\rfloor$
for $i \leftarrow 0$ **to** $n-1$ **do**
  $e_i \leftarrow \left(\texttt{idx} = i \in \mathbb{Z},\ \texttt{val} = \frac{p}{q} a - \left\lfloor \frac{p}{q} a \right\rfloor \in \mathbb{Q} \right)$
Sort e in descending order of $e.\texttt{val}$.
$k \leftarrow p \left\lceil \frac{b(1)}{p} \right\rfloor - b(1)$
for $i \leftarrow 0$ **to** $k-1$ **do**
  $b_{e_i.\texttt{idx}} \leftarrow b_{e_i.\texttt{idx}} + 1$
return b

5.4 IND-CPA security of r5_cpa_pke with $\xi = N_{n+1}$ and Rounding noise

The proof of IND-CPA security for a RLWE variant of r5_cpa_pke in Sec. 5.3 requires both the secrets and also the noise polynomials to be multiples of $(x-1)$ (this is used in an essential step of the proof, see Appendix C). This last requirement is the reason why this proof does not apply to Round5 with $\xi(x) = x^{n+1} - 1$ using RLWR defined as *component-wise rounding*. This deterministic component-wise rounding does not allow enforcing that the noisy "rounding" polynomials are multiples of $(x-1)$.

Round5's design can be adapted to use a slightly different type of rounding informally named as "rounding to the root lattice" [14,15,28] – that allows the IND-CPA proof to work. This alternate rounding technique is described in Algorithm 4, that takes as input an $a \in \mathbb{Z}_q[x]$, integer moduli q, p where $p < q$ and returns a $b \in \mathbb{Z}_p[x]$ satisfying $b(1) \equiv 0 \pmod{p}$.

Rounded noise introduced in b using Algorithm 4 is a polynomial whose coefficients sum to zero, so that a direct translation of the IND-CPA proof in Sec. 5.3 to the RLWR case is possible. However, this modification – going from component-wise rounding to rounding to the root lattice – would introduce additional complexity with no clear concrete security benefits. First, Sample_μ gets rid of $n+1-\mu$ coefficients so that knowing k is irrelevant. Second, concrete security attacks use the norm of the noise that hardly changes here. Because of these two reasons, we argue that the current Round5 design (and the rounding used in it) is sound and secure, and further modifications are not required.

6 Parameters, Performance and Comparison

Round5 has a large design space, adding to the parameters available in Round2, namely n, h, q, p, t, also f. If $f > 0$, then $\xi(x) = N(x)$. By searching over the design space, we obtain parameters that minimize bandwidth requirements given a minimum targeted security level and failure probability. The failure probability analysis is done as in Section 4. Concrete security is analyzed in the standard manner [5], the primal [4], dual [1], hybrid [22], and sparse secret attacks [1,5] are considered, under both sieving [7] and enumeration [2] cost models. Details

are not included due to space limits. A script to verify computations is available at www.round5.org.

Table 1: Parameters: "C" denotes security level against classical adversaries, while "Q" denotes that against quantum ones. Bandwidth is in bytes.

Name Set	Parameters (n, h, q, p, t, f)	Failure rate	Sieving (C/Q)	Enumeration (C/Q)	Bandwidth (pk/ct)
R5ND_1KEM_5c	$490, 162, 2^{10}, 2^7, 2^3, 5$	2^{-88}	128/122	170/135	445 + 549
R5ND_1KEM_0c	$618, 104, 2^{11}, 2^8, 2^4, 0$	2^{-65}	128/122	160/133	634 + 682
R5ND_1KEM_4longkey	$490, 162, 2^{10}, 2^7, 2^3, 4$	2^{-71}	128/122	170/135	453 + 563
R5ND_1PKE_5c	$508, 136, 2^{10}, 2^7, 2^4, 5$	2^{-142}	128/122	166/134	461 + 636
R5ND_5PKE_5c	$940, 414, 2^{12}, 2^8, 2^3, 2$	2^{-144}	256/232	390/307	972 + 1172
R5ND_0KEM_2iot	$372, 178, 2^{11}, 2^7, 2^3, 2$	2^{-41}	96/90	129/96	342 + 394
NewHope1024-CCA-KEM [31]	N/A	2^{-216}	257/233	-	1824 + 2208
Kyber1024 [8]	N/A	2^{-169}	241/218	-	1440 + 1504
FireSaber-KEM [23]	N/A	2^{-165}	270/245	-	1312 + 1472

Table 1 includes a number of exemplary Round5 parameter sets. Also shown are a number of similar proposals for comparison. R5ND_1KEM_5c and R5ND_1KEM_0c both target NIST security category 1 as IND-CPA secure KEMs. However, the second requires around 33% more bandwidth since it does not use error correction ($f = 0$). This demonstrates the benefit of error correction.

R5ND_1KEM_4longkey also targets NIST security category 1 as an IND-CPA secure KEM. However, it uses the flexibility of Sample_μ to encapsulate a longer key (192 bits instead of 128) so that the (quantum) hardness of attacking the shared secret is as much as (quantum) attacking the underlying lattice problem.

R5ND_1KEM_5c and R5ND_1PKE_5c differ in the target failure probability. The latter is constructed by applying the Fujisaki-Okamoto transform [21] on r5_cpa_pke in a standard manner and combining with a secure (one-time) data encapsulation scheme (e.g., AES256); its failure rate is much lower to achieve the IND-CCA security required of public-key encryption (PKE). Comparing the above two parameter sets shows that a more relaxed failure probability target leads to bandwidth savings of more than 100 B.

R5ND_5PKE_5c targets NIST security category 5 as an IND-CCA secure PKE. It requires 2144 B of bandwidth. Among existing proposals targeting the same security category, NewHope1024-CCA-KEM [31] requires 88% more bandwidth, FireSaber [12] requires 30% more, and Kyber1024 requires 37% more. Round5's compact keys fit easily in protocols with a limited (1500 B) MTU.

Finally, parameter set R5ND_0KEM_2iot shows that Round5's design flexibility makes it easy to obtain parameters that offer a reasonable security level, but require relatively little bandwidth enabling security in more resource constrained applications such as IoT.

7 Conclusions and Future work

In this work, we introduced *Round5*, a lattice-based cryptosystem consisting of a public-key encryption scheme that uses rounding both to introduce noise (for security) and at the same time reduce the key-size, improving performance. Public-keys are computed via ring multiplications in $\mathbb{Z}[x]/\Phi_{n+1}(x)$, thus offering a wide variety of choices for the security parameter n, in turn allowing to finely tune the parameters and performance of Round5. A novel contribution of this work is to compute part of the ciphertext, on the other hand, via ring multiplications in $\mathbb{Z}[x]/N_{n+1}(x)$; this, in combination with the fact that Round5 secret-keys are polynomials with a factor $(x-1)$, allows to have low decryption failure rates similar to schemes constructed using the $x^{2^k}+1$ cyclotomic polynomial, while still allowing to have the above mentioned benefit of the $\mathbb{Z}[x]/\Phi_{n+1}(x)$ polynomial ring.

Further, this leads to very low dependencies between coefficients and independent bit failures, so that error correction can be used to further improve failure rates, performance (since parameters can be shrunk) and security (since more noise can be added). For the latter, r5_cpa_pke uses the XEf f-bit error correcting code originally introduced in the HILA5 scheme [33]. The main advantage of XEf codes is that they avoid table look-ups and conditions altogether and are therefore resistant to timing attacks.

An interesting open question is to investigate a variant of Round5 where component-wise rounding is replaced by the alternate rounding technique described in Algorithm 4 and investigate implications on the resulting scheme's concrete security and decryption failure behavior.

Acknowledgements

We thank Mike Hamburg for helpful discussions on combining features from the prime-order cyclotomic and power-of-two cyclotomic polynomial rings in a lattice based cryptosystem. We thank Léo Ducas for helpful discussions on rounding to the root lattice, and techniques required for proving IND-CPA security for a rounding-based scheme using N_{n+1} as reduction polynomial. Finally, we wish to thank our anonymous reviewers for their helpful comments that led to improving the content and readability of the paper.

References

1. Martin R. Albrecht. On dual lattice attacks against small-secret LWE and parameter choices in HElib and SEAL. Cryptology ePrint Archive, Report 2017/047, 2017.
2. Martin R. Albrecht, Rachel Player, and Sam Scott. On the concrete hardness of learning with errors. Cryptology ePrint Archive, Report 2015/046, 2015.
3. Erdem Alkim, Léo Ducas, Thomas Pöppelmann, and Peter Schwabe. Post-quantum key exchange - a new hope. Cryptology ePrint Archive, Report 2015/1092, 2015.

4. Erdem Alkim, Léo Ducas, Thomas Pöppelmann, and Peter Schwabe. NewHope without reconciliation. Cryptology ePrint Archive, Report 2016/1157, 2016.
5. Hayo Baan, Sauvik Bhattacharya, Oscar Garcia-Morchon, Ronald Rietman, Ludo Tolhuizen, Jose-Luis Torre-Arce, and Zhenfei Zhang. Round2: KEM and PKE based on GLWR. Cryptology ePrint Archive, Report 2017/1183, 2017.
6. Abhishek Banerjee, Chris Peikert, and Alon Rosen. Pseudorandom functions and lattices. Cryptology ePrint Archive, Report 2011/401, 2011.
7. Anja Becker, Léo Ducas, Nicolas Gama, and Thijs Laarhoven. New directions in nearest neighbor searching with applications to lattice sieving. Cryptology ePrint Archive, Report 2015/1128, 2015.
8. Joppe Bos, Léo Ducas, Eike Kiltz, Tancrède Lepoint, Vadim Lyubashevsky, John M. Schanck, Peter Schwabe, and Damien Stehlé. CRYSTALS – Kyber: a CCA-secure module-lattice-based KEM. Cryptology ePrint Archive, Report 2017/634, 2017.
9. Joppe W. Bos, Craig Costello, Michael Naehrig, and Douglas Stebila. Post-quantum key exchange for the TLS protocol from the ring learning with errors problem. In *2015 IEEE Symposium on Security and Privacy, SP 2015, San Jose, CA, USA, May 17-21, 2015*, pages 553–570, 2015.
10. Jung Hee Cheon, Duhyeong Kim, Joohee Lee, and Yongsoo Song. Lizard: Cut off the Tail! Practical Post-Quantum Public-Key Encryption from LWE and LWR. Cryptology ePrint Archive, Report 2016/1126, 2016.
11. Jung Hee Cheon, Sangjoon Park, Joohee Lee, Duhyeong Kim, Yongsoo Song, Seungwan Hong, Dongwoo Kim, Jinsu Kim, Seong-Min Hong, Aaram Yun, Jeongsu Kim, Haeryong Park, Eunyoung Choi, Kimoon Kim, Jun-Sub Kim, and Jieun Lee. Lizard. Technical report, National Institute of Standards and Technology, 2017. Available at https://csrc.nist.gov/projects/post-quantum-cryptography/round-1-submissions.
12. Jan-Pieter D'Anvers, Angshuman Karmakar, Sujoy Sinha Roy, and Frederik Vercauteren. SABER. Technical report, National Institute of Standards and Technology, 2017. Available at https://csrc.nist.gov/projects/post-quantum-cryptography/round-1-submissions.
13. Jan-Pieter D'Anvers, Angshuman Karmakar, Sujoy Sinha Roy, and Frederik Vercauteren. Saber: Module-LWR based key exchange, CPA-secure encryption and CCA-secure KEM. Cryptology ePrint Archive, Report 2018/230, 2018.
14. Leo Ducas. Public discussion, https://csrc.nist.gov/CSRC/media/Projects/Post-Quantum-Cryptography/documents/round-1/official-comments/Round5-official-comment.pdf, August 2018. Messages on the NIST PQC mailing list.
15. Léo Ducas and Wessel P.J. van Woerden. The closest vector problem in tensored root lattices of type A and in their duals. Cryptology ePrint Archive, Report 2016/910, 2016. https://eprint.iacr.org/2016/910.
16. ETSI. "ETSI launches Quantum Safe Cryptography specification group", March 2015.
17. ETSI. Terms of reference for ETSI TC cyber working group for quantum-safe cryptography (ETSI TC cyber WG-QSC), 2017. Accessed: 15-02-2017.
18. Tim Fritzmann, Thomas Pöppelmann, and Johanna Sepulveda. Analysis of error-correcting codes for lattice-based key exchange. Cryptology ePrint Archive, Report 2018/150, 2018.
19. Mike Hamburg. Three Bears. Technical report, National Institute of Standards and Technology, 2017. Available at https://csrc.nist.gov/projects/post-quantum-cryptography/round-1-submissions.

20. Jeffrey Hoffstein, Jill Pipher, and Joseph H. Silverman. NTRU: A ring-based public key cryptosystem. In Joe P. Buhler, editor, *Algorithmic Number Theory, Third International Symposium, ANTS-III, Portland, Oregon, USA, June 21-25, 1998, Proceedings*, pages 267–288. Springer, 1998.
21. Dennis Hofheinz, Kathrin Hövelmanns, and Eike Kiltz. A modular analysis of the Fujisaki-Okamoto transformation. Cryptology ePrint Archive, Report 2017/604, 2017.
22. Nick Howgrave-Graham. A Hybrid Lattice-Reduction and Meet-in-the-Middle Attack Against NTRU. In Alfred Menezes, editor, *Advances in Cryptology - CRYPTO 2007: 27th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 19-23, 2007. Proceedings*, pages 150–169. Springer, 2007.
23. J.P.d'Anvers, A. Karmakar, S. Sinha Roy, and F. Vercauteren. Saber: Module-LWR Based Key Exchange, CPA-Secure Encryption and CCA-Secure KEM. In *Progress in Cryptology: AfricaCrypt 2018*, pages 282–305, 2018.
24. Richard Lindner and Chris Peikert. Better key sizes (and attacks) for lwe-based encryption. Cryptology ePrint Archive, Report 2010/613, 2010.
25. Xianhui Lu. Public discussion, https://csrc.nist.gov/CSRC/media/Projects/Post-Quantum-Cryptography/documents/round-1/official-comments/LAC-official-comment.pdf, December 2018. Messages on the NIST PQC mailing list.
26. Xianhui Lu, Yamin Liu, Dingding Jia, Haiyang Xue, Jingnan He, and Zhenfei Zhang. LAC. Technical report, National Institute of Standards and Technology, 2017. Available at https://csrc.nist.gov/projects/post-quantum-cryptography/round-1-submissions.
27. Vadim Lyubashevsky, Chris Peikert, and Oded Regev. On ideal lattices and learning with errors over rings. Cryptology ePrint Archive, Report 2012/230, 2012.
28. Robby G. McKilliam, I. Vaughan L. Clarkson, and Barry G. Quinn. An Algorithm to Compute the Nearest Point in the Lattice A_{n}^*. *CoRR*, abs/0801.1364, 2008.
29. NIST. Submission requirements and evaluation criteria for the post-quantum cryptography standardization process. POST-QUANTUM CRYPTO STANDARDIZATION. Call For Proposals Announcement, 2016.
30. Chris Peikert, Oded Regev, and Noah Stephens-Davidowitz. Pseudorandomness of Ring-LWE for Any Ring and Modulus, 2017.
31. Thomas Pöppelmann, Erdem Alkim, Roberto Avanzi, Joppe Bos, Leo Ducas, Antonio de la Piedra, Peter Schwabe, and Douglas Stebila. NewHope. Technical report, National Institute of Standards and Technology, 2017. Available at https://csrc.nist.gov/projects/post-quantum-cryptography/round-1-submissions.
32. Markku-Juhani O. Saarinen. Ring-LWE ciphertext compression and error correction: Tools for lightweight post-quantum cryptography. In *Proceedings of the 3rd ACM International Workshop on IoT Privacy, Trust, and Security*, IoTPTS '17, pages 15–22. ACM, April 2017.
33. Markku-Juhani O. Saarinen. HILA5: On reliability, reconciliation, and error correction for Ring-LWE encryption. In Carlisle Adams and Jan Camenisch, editors, *SAC 2017*, volume 10719 of *Lecture Notes in Computer Science*, pages 192–212. Springer, 2018.
34. Nigel P. Smart, Martin R. Albrecht, Yehuda Lindell, Emmanuela Orsini, Valery Osheter, Kenny Paterson, and Guy Peer. LIMA. Technical report, National Institute of Standards and Technology, 2017. Available at https://csrc.nist.gov/projects/post-quantum-cryptography/round-1-submissions.

A Probability of decryption failures in Round5

In decryption, the polynomial $y = \langle \lfloor \frac{2}{p}\zeta \rceil \rangle_2$ is computed, where $\zeta = \langle \frac{p}{t}v - \text{Sample}_\mu(\langle su \rangle_\xi) + h_2 1_\mu \rangle_p$, where 1_μ is the polynomial of degree $\mu - 1$ with all coefficients equal to 1. First, a sufficient condition is derived so that y and $\eta = \texttt{xef\_compute}_{\kappa,f}(m)$ agree in a given coefficient. We have that

$$v \equiv \left\langle \frac{t}{p}\text{Sample}_\mu(\langle br \rangle_\xi + h_1 1_n) - \frac{t}{p}i_v \right\rangle_p + \frac{t}{2}\eta \pmod t,$$

where $\frac{t}{p}i_v$ is the error introduced by the rounding downwards, with each component of $\boldsymbol{i}_v$ in $\mathbb{Z}_{p/t}$. As a result,

$$\zeta \equiv \frac{p}{2}\eta + \Delta \pmod p \text{ with } \boldsymbol{\Delta} = (h_1 + h_2)1_\mu - i_v + \text{Sample}_\mu(\langle br - su + h_4 \boldsymbol{j} \rangle_\xi). \quad (11)$$

As $y = \lfloor \frac{2}{p}\boldsymbol{\zeta} - \frac{1}{2} \rceil$, it holds that $y \equiv \eta + \lfloor \frac{2}{p}\Delta - \frac{1}{2}1_n \rceil \equiv \eta + \lfloor \frac{2}{p}\{\Delta - \frac{p}{4}1_n\}_p \rceil \pmod 2$. Here $\{w\}_p$ denotes the integer in $(-p/2, p/2]$ that is equivalent to w modulo p. As a consequence, $y_i = \eta_i$ whenever $|\{\boldsymbol{\Delta}_i - \frac{p}{4}\}_p| < \frac{p}{4}$. We infer that $y_i = \eta_i$ whenever

$$|\left\{ \frac{q}{p}\boldsymbol{\Delta}_i - \frac{q}{4} \right\}_q| < \frac{q}{4} \quad (12)$$

Equivalently, as $\frac{q}{p}\boldsymbol{\Delta}_i$ has integer components, if $y_i \neq \eta_i$, then

$$\left\langle \frac{q}{p}\boldsymbol{\Delta}_i \right\rangle_q \in \left[\frac{q}{4}, q - \frac{q}{4} \right] \quad (13)$$

In order to analyze this probability, we work out $\frac{q}{p}\boldsymbol{\Delta} - \frac{q}{4}\boldsymbol{j}$, using (11). We write $j_v = \frac{q}{p}((h_1 + h_2)1_\mu - i_v - \frac{p}{4}1_\mu)$. The definitions of h_1 and h_2 imply that $j_v = \frac{q}{p}(\frac{p}{2t}1_\mu - i_v)$. Each coefficient of $\boldsymbol{i}_v$ is in $\mathbb{Z}_{p/t}$. The value of h_2 thus ensures that the absolute value of each coefficient of $\frac{p}{2t} - i_v$ is at most $\frac{p}{2t}$.
We now analyze $\frac{q}{p}\langle br - su \rangle_\xi$. Similarly to the expression for $\boldsymbol{v}$, we write

$$b = \left\langle \frac{p}{q}\left(\langle as \rangle_{\Phi_{n+1}} + h_1 1_n \right) - \frac{p}{q}i_b \right\rangle_p \text{ and } u = \left\langle \frac{p}{q}(\langle ar \rangle_{\Phi_{n+1}} + h_1 1_n) - \frac{p}{q}i_u \right\rangle_p,$$

with all components of i_b and i_u in $\mathbb{Z}_{q/p}$. We thus have

$$\frac{q}{p}(br - su) \equiv \langle sa \rangle_{\Phi_{n+1}} \boldsymbol{r} - s\langle a\boldsymbol{r} \rangle_{\Phi_{n+1}} + j_b r - s j_u \pmod q \quad (14)$$

$$\text{where } j_b = h_1 1_n - i_b \text{ and } j_u = h_1 1_n - i_u. \quad (15)$$

As $h_1 = \frac{q}{2p}$, all entries of j_b and of j_u are from the set $I := (-\frac{q}{2p}, \frac{q}{2p}] \cap \mathbb{Z}$. Obviously, if $\xi(x) = \Phi_{n+1}(x)$, then $\langle sa \rangle_{\Phi_{n+1}} r - s\langle ar \rangle_{\Phi_{n+1}} \equiv 0 \pmod \xi$. The same is true if $\xi = N_{n+1}$ and r and s both are multiple of $(x - 1)$. Indeed, there are $\lambda_s, \lambda_r \in \mathbb{Z}[x]$ such that $\langle sa \rangle_{\Phi_{n+1}} = sa + \lambda_r \Phi_{n+1}$ and $\langle ar \rangle_{\Phi_{n+1}} = ar - \lambda_s \Phi_{n+1}$. As a consequence, $\langle as \rangle_{\Phi_{n+1}} r - s\langle ar \rangle_{\Phi_{n+1}} = \lambda_s \Phi_{n+1} r - s\lambda_r \Phi_{n+1}$. As $(x - 1)$

divides s and r, both $\Phi_{n+1}r$ and $s\Phi_{n+1}$ are divisible by N_{n+1}. As a result, for $\xi \in \{\Phi_{n+1}, N_{n+1}\}$

$$\frac{q}{p}\Delta \equiv j_v + \text{Sample}_\mu\left(\langle j_b r - s j_u\rangle_\xi\right) \pmod q. \tag{16}$$

The probability of a decryption failure in position i before error correction is at most the probability that (13) is satisfied.

In our analysis of (13) combined with (16), the coefficients of j_b and j_u are drawn independently and uniformly from $I = (-\frac{q}{2p}, \frac{q}{2p}] \cap \mathbb{Z}$, and the coefficients of j_v are drawn independently and distributed as $\frac{q}{p}y$ with y uniform on $(-\frac{p}{2t}, \frac{p}{2t}] \cap \mathbb{Z}$.

B Distinguishing attack at $x = 1$ for $\xi = N_{n+1}$

The "Evaluate at $x = 1$" distinguishing attack [20] applies against schemes using the ring $\mathbb{Z}[x]/N_{n+1}(x)$. We argue that this attack cannot be applied in Round5 if $\mu \leq n$.

Consider a pair of polynomials $(b(x), v(x))$ with $b(x)$ uniformly distributed on $\mathbb{Z}_p[x]/(x^{n+1}-1)$ and $v(x) = \langle \text{Sample}_\mu(\lfloor \frac{t}{p}(\langle b(x)r(x)\rangle_{N(x)} + h_1)\rfloor) + \frac{t}{2}m(x)\rangle_t$ with $r(x)$ drawn independently and uniformly from the ternary polynomials of degree at most $n-1$ satisfying $r(1) = 0$, and $m(x)$ drawn according to some distribution on $\mathbb{Z}_2[x]/(x^\mu - 1)$. We then have that $v(x) \equiv \lfloor \text{Sample}_\mu(\frac{t}{p}(\langle b(x)r(x)\rangle_{N(x)} + h_1)\rfloor) + \frac{t}{2}m(x) \pmod t$, and so $w(x) = \frac{p}{t}v(x)$ satisfies

$$w(x) \equiv \text{Sample}_\mu(\langle b(x)r(x)\rangle_{N(x)}) + \frac{p}{t}\cdot h_1 \sum_{i=0}^{\mu-1} x^i - \frac{p}{t}\epsilon(x) + \frac{p}{2}m(x) \pmod p.$$

where $\epsilon(x)$ is the result of rounding downwards, so all components of $\frac{p}{t}\epsilon(x)$ are in $[0, \frac{p}{t}) \cap \mathbb{Z}$. As $(x-1)$ divides both $r(x)$ and $N(x)$, it follows that $x-1$ divides $\langle b(x)r(x)\rangle_{N(x)}$, and so if $\mu = n+1$, then

$$w(1) \equiv \frac{p}{t}\cdot h_1 \cdot (n+1) - \frac{p}{t}\sum_{i=0}^{n}\epsilon_i + \frac{p}{2}m(1) \pmod p.$$

For large n, the value of $\frac{p}{t}\sum_{i=0}^{n}\epsilon_i$ is close to its average, i.e., close to $n\frac{p}{2t}$. As a result, has maxima at values $\frac{p}{t}h_1(n+1) - n\frac{p}{2t} + \frac{p}{2}k$ for $0 \leq k \leq 1$. So $w(1)$ can serve as a distinguisher between the above distribution and the uniform one.

Now assume that $\mu < n+1$. We take $\mu = n$, which is the case giving most information to the attacker. Writing $f(x) = \langle b(x)r(x)\rangle_{N(x)} = \sum_{i=0}^{n} f_i x^i$, it holds that

$$w(1) \equiv \sum_{i=0}^{n-1} f_i + \frac{p}{t}\cdot h_1 \cdot n - \frac{p}{t}\epsilon(1) + \frac{p}{2}m(1) \pmod p.$$

As shown above, $f(1) = 0$, and so $\sum_{i=0}^{n-1} f_i = -f_n$. Hence, under the assumption that f_n is distributed uniformly modulo p, also $w(1)$ is distributed uniformly modulo p. The latter assumption is supported by [30].

C Proof of IND-CPA security of r5_cpa_pke RLWE variant

We present the proof of IND-CPA security for an RLWE variant of r5_cpa_pke. The following notation will be used. We write $\phi(x) = 1 + x + \ldots + x^n$, and $N(x) = x^{n+1} - 1$, where $n+1$ is prime. Moreover, $R_\phi = \mathbb{Z}_q[x]/\phi(x)$, and

$$R_0 = \{f(x) = \sum_{i=0}^{n} f_i x^i \in \mathbb{Z}_q[x] \mid \sum_{i=0}^{n} f_i \equiv 0 \pmod{q}\} \tag{17}$$

As $N(x) = (x-1)\phi(x)$, it holds that $\langle (x-1)f(x)\rangle_{N(x)} = (x-1)\langle f(x)\rangle_{\phi(x)}$ for any $f \in \mathbb{Z}[x]$. As a result, $f(x) \mapsto (x-1)f(x)$ is a bijection from R_ϕ to R_0.

In the proof, the following lemma will be used.

Lemma 1 *Let q and $n+1$ be relatively prime, and let $(n+1)^{-1}$ be the multiplicative inverse of $n+1$ in $\mathbb{Z}_q$. The mapping $\mathcal{F}$ defined as*

$$\mathcal{F} : (\sum_{i=0}^{n-1} f_i x^i) \mapsto \sum_{i=0}^{n-1} f_i x^i - (n+1)^{-1} \cdot \left(\sum_{i=0}^{n-1} f_i\right) \cdot \phi(x)$$

is a bijection from R_ϕ to R_0.

Proof. It is easy to see that $\mathcal{F}$ maps R_ϕ to R_0. To show that $\mathcal{F}$ is a bijection, let $g(x) = \sum_{i=0}^{n} g_i x^i \in R_0$, and let $f(x) = \sum_{i=0}^{n} \langle g_i - g_n \rangle_q x^i$. Clearly, $f \in \mathbb{Z}_q[x]$ has degree at most $n-1$, and by direct computation, $\mathcal{F}(f(x)) = g(x)$.

In the description below, $\mathcal{S}$ denotes a set of secrets such that

$$\mathcal{S} \subset \{f(x) = \sum_{i=0}^{n-1} f_i x^i \in \mathbb{Z}_q[x] \mid \sum_{i=0}^{n-1} f_i \equiv 0 \pmod{q}\}, \tag{18}$$

Moreover, $\mathcal{M}$ denotes a message space, and *ECC_Enc* and *ECC_Dec* are error correcting encoding and decoding algorithms such that

$$\{ECC\_Enc(m) \mid m \in \mathcal{M}\} \subset \{f(x) = \sum_{i=0}^{n} f_i x^i \in \mathbb{Z}_2[x] \mid \sum_{i=0}^{n} f_i \equiv 0 \pmod{2}\}. \tag{19}$$

Moreover, χ denotes a probability distribution on R_ϕ.

For understanding Algorithm 7, note that as $(x-1)|s(x)$, we have that $su' \equiv sa'r + se_1 \pmod{N}$, and, as $(x-1)|r(x)$, that $rb' \equiv ra's + re_0 \pmod{N}$. As a consequence,

$$\zeta \equiv v - su' \equiv \frac{q}{2} ECC\_Enc(m) + (x-1)e_2 + re_0 - se_1 \pmod{N}, \text{ whence}$$

$$\lfloor \frac{2}{q}\zeta \rceil \equiv ECC\_Enc(m) + \lfloor \frac{2}{q}((x-1)e_2 + re_0 - se_1) \rceil \pmod{N}.$$

We are now in a position to prove the following result.

Algorithm 5: CPA-PKE.Keygen()

$a' \xleftarrow{\$} R_\phi, s \xleftarrow{\$} \mathcal{S}, e_0 \leftarrow \chi$
$b' = \langle a's + e_0 \rangle_\phi$
$pk = (a', b')$
$sk = s$
return (pk, sk)

Algorithm 6: CPA-PKE.Enc($pk = (a', b'), m \in \mathcal{M}$)

$r \xleftarrow{\$} \mathcal{S}, e_1, e_2 \xleftarrow{\$} \chi$
$u' = \langle a'r + e_1 \rangle_\phi$
$v = \langle \frac{q}{2} ECC\_Enc(m) + b'r + (x-1)e_2 \rangle_N$
$c = (u', v)$
return c

Algorithm 7: CPA-PKE.Dec(sk, c)

$\zeta = \langle v - su' \rangle_N$
$\hat{m} = ECC\_Dec(\langle \lfloor \frac{2\zeta}{q} \rceil \rangle_2)$
return $\hat{m}$

Theorem 3 *For every IND-CPA adversary $\mathcal{A}$ with advantage A, there exist algorithms C and E such that*

$$A \leq Adv_1(C) + Adv_3(E). \tag{20}$$

Here Adv_1 refers to the advantage of distinguishing between the uniform distribution on $(\mathbb{Z}_q[x]/\phi(x))^2$ and the R-LWE distribution

$$(a', b' = \langle a's + e_0 \rangle_\phi) \text{ with } a' \xleftarrow{\$} R_\phi, s \xleftarrow{\$} \mathcal{S}, e_0 \leftarrow \chi \tag{21}$$

Similarly, Adv_3 refers to the advantage of distinguishing between the uniform distribution on $(\mathbb{Z}_q[x]/\phi(x))^4$ and the distribution of two R-LWE samples with a common secret, given by

$$(a', b'', u', v') \text{ with } a', b'' \xleftarrow{\$} \mathbb{Z}_q[x]/\phi(x), u = \langle a'r + e_1 \rangle_\phi, \tag{22}$$

$$v = \langle b"r + e_2 \rangle_\phi \text{ with } r \xleftarrow{\$} \mathcal{S}, e_1, e_2 \leftarrow \chi \tag{23}$$

Proof. We prove the theorem using a sequence of IND-CPA games. We denote by S_i the event that the output of game i equals 1.
Game G_0 is the original IND-CPA game. In Game G_1, the public key (a', b') is replaced by a pair (a', b') uniformly drawn from R_ϕ^2. It can be shown that there exists an algorithm $\mathcal{C}$ for distinguishing between the uniform distribution on R_ϕ^2 and the R-LWE distribution of pairs (a', b') with $a' \xleftarrow{\$} R_\phi$, $b' = \langle as' + e_0 \rangle_\phi$ with $s \xleftarrow{\$} \mathcal{S}$ and $e_0 \leftarrow \chi$ such that

$$\mathrm{Adv}_1(\mathcal{C}) = |\Pr(S_0) - \Pr(S_1)|.$$

In Game G_2, the values $u' = \langle a'r + e_1 \rangle_\phi$ and $\hat{v} = \langle b'r + (x-1)e_2 \rangle_N$ used in the generation of v are simultaneously substituted with uniform random variables from R_ϕ and R_0, respectively. it can be shown that there exists an adversary $\mathcal{D}$ with the same running time as that of $\mathcal{A}$ such that

$$\mathrm{Adv}_2(\mathcal{D}) = |\Pr(S_1) - \Pr(S_2)|.$$

Here Adv_2 refers to the advantage of distinguishing between the uniform distribution on $R_\phi^3 \times R_0$ and the distribution

$$(a', b', u', v) = (a', b', \langle a'r + e_1 \rangle_\phi, \langle b'r + (x-1)e_2 \rangle_N) \text{ with } a', b' \xleftarrow{\$} R_\phi, r \xleftarrow{\$} \mathcal{S}, e_1, e_2 \xleftarrow{\$} \chi. \tag{24}$$

Because of (19), the value of the ciphertext v in Game G_2 is independent of bit b, and therefore $\Pr(S_2) = 1/2$. As a final step, we define $\Psi : R_\phi^3 \times R_0 \to R_\phi^4$ as

$$\Psi(a'(x), b'(x), u'(x), v(x)) = (a'(x), b''(x), u'(x), v'(x)) \text{ with} \tag{25}$$

$$b''(x) = \frac{\mathcal{F}(b'(x))}{x-1}, v'(x) = \frac{v(x)}{x-1} \tag{26}$$

As $\mathcal{F}$ is a bijection from R_ϕ to R_0 (see Lemma 1) and $f(x) \mapsto \frac{f(x)}{x-1}$ is a bijection from R_0 to R_ϕ, it follows that Ψ is a bijection. Writing $b(x) = \mathcal{F}(b'(x))$, we infer that

$$b(x)r(x) = b'(x)r(x) - (n+1)^{-1}b'(1)\phi(x)r(x) \equiv b'(x)r(x) \pmod{N(x)},$$

where the latter equivalence holds as $r(x)$ is a multiple of $(x-1)$, and so

$$v(x) = \langle b'(x)r(x) + (x-1)e_2(x) \rangle_N = \langle b(x)r(x) + (x-1)e_2(x) \rangle_N.$$

As $r(x)$ is a multiple of $x-1$, it follows that $v(x) \in R_0$ and that

$$v'(x) = \frac{v(x)}{x-1} \equiv \langle b''(x)r(x) + e_2(x) \rangle_\phi \text{ where } b''(x) = \frac{b(x)}{x-1}.$$

As a result, the advantage of $\mathcal{E} = \Psi \circ \mathcal{D}$ in distinguishing between the uniform distribution on R_Φ^4 and the distribution

$$(a', b'', u', v') \text{ with } a, b'' \xleftarrow{\$} R_\phi, u'(x) = \langle a'r + e_1 \rangle_\phi \text{ and } v' = \langle b''r + e_2 \rangle_\phi$$

is equal to $\mathrm{Adv}_2(D)$. Note that (a, u') and (b'', v') are two R-LWE samples with common secret $r(x) \in \mathcal{S}$, with a', b'' chosen uniformly in $\mathcal{R}_\phi$ and independent noise polynomials $e_1(x)$ and $e_2(x)$.
As $Pr(S_2) = \frac{1}{2}$, we conclude that

$$Adv(\mathcal{A}) = |\Pr(S_0) - \Pr(S_2)| \leq \sum_{i=0}^{1} |\Pr(S_i) - \Pr(S_{i+1})| = \mathrm{Adv}_1(\mathcal{C}) + \mathrm{Adv}_2(\mathcal{E}).$$