# On the Decoding Failure Rate of HQC

Alessandro Annechini and Alessandro Barenghi and Gerardo Pelosi *Member, IEEE*
Department of Electronics, Information and Bioengineering - DEIB
Politecnico di Milano, Milano, Italy
Email: alessandro.annechini@polimi.it, alessandro.barenghi@polimi.it, gerardo.pelosi@polimi.it

***Abstract*—Cryptography based on error correction codes has gained significant interest due to its ability to provide security against both classical and quantum adversaries. In 2025, the U.S. National Institute of Standards and Technology selected the Hamming Quasi-Cyclic (HQC) key encapsulation mechanism for standardization. A key aspect of HQC is the possibility of decryption failures, which reveal information about the private key. To address this issue, the HQC authors developed a probabilistic model for the decoding failure rate (DFR) of the underlying error-correcting code, and adjusted the cryptosystem parameters to thwart attacks based on decryption failures. However, the DFR model relies on the assumption of independence between coordinates of the error vector, which does not hold in HQC. This approximation yields conservative DFR estimates in regimes where failure probabilities can be simulated, and it is hypothesized to remains conservative for cryptographic-grade parameter sets. In this work, we eliminate the independence assumptions and derive a new closed-form DFR model for HQC. We demonstrate that the previous approximation remains conservative in the cryptographic regime and that HQC's current decoding failure rates are lower than the required ones. We describe optimization techniques that enable our probabilistic model to serve as a parameter-tuning tool, and demonstrate how the size of HQC public keys and ciphertexts can be slightly reduced without compromising security.**

## I. Introduction

The research in post-quantum cryptography has met an increased interest in the last decade, due to the advancements in quantum computing technologies. In 2016, the U.S. National Institute of Standards and Technology (NIST) started a contest for the standardization of post-quantum Key Encapsulation Methods (KEMs). Among the proposed cryptosystems, the ones based on hard problems on lattices and error correction codes provided the best performance and security guarantees. In 2023, the lattice-based KEM CRYSTALS-Kyber [1] was selected for standardization, while the code-based KEM Hamming Quasi-cyclic (HQC) [2] has been selected for standardization in 2025 [3]. HQC, which is based on the 2003 template introduced by Alekhnovich in [4], relies on the quasi-cyclic syndrome decoding problem, leveraging the isomorphism between $p \times p$ binary circulant matrices and the polynomial ring $\mathcal{R} = \mathbb{F}_2[x]/\langle x^p - 1\rangle$ to improve performance. At its core, the HQC encryption procedure masks the message by adding a dense binary error vector, while the decryption procedure reduces the density of the error vector and recovers the original message with a public error correction code. The correctness of the cryptosystem hinges on the ability of the decoding procedure to successfully recover an error-affected binary message. The Hamming weight of the error affecting the message is not fixed, and the HQC design trades off a non-null decoding failure (equivalently, decryption failure) probability for a significant reduction in the public key and ciphertext size. Since the error to be decoded depends on the private key, decryption failures leak secret information [5]. To attain security against active attackers, and benefit from the "indistinguishability under chosen ciphertext attack" (IND-CCA2) property, the Decoding Failure Rate (DFR) of the error correction code must be sufficiently low, i.e., at most $2^{-\lambda}$, $\lambda \in \{128, 192, 256\}$, depending on the security level required. To address the issue, the authors of HQC derive a conservative, closed-form probabilistic model for the DFR of the decoder, choosing the parameters of the cryptosystem in order to meet the requirements in terms of DFR [2]. Such model lies on the simplifying assumption that the coordinates of the error vector act as independent and identically distributed Bernoulli random variables: the error vector is thus modeled considering a binary symmetric channel, and the DFR of the concatenated code employed in HQC is derived accordingly. Another implicit assumption present in the currently adopted DFR model for HQC is that symbols of the internal code fail independently from one another. In [6], the authors show that the real and modeled distributions of the error weight have non-negligible statistical distance, while in [7] a less conservative DFR estimate is computed by approximating the error weight distribution with a Gaussian distribution. These works provide evidence that, with a precise model for the error weight and the failure probability of the internal code, the accuracy of the decoding failure rate estimates can be significantly improved.

**Contributions.** In this work, we provide an exact characterization of the error weight distribution, removing all independence assumptions, and we derive a complete closed-form model for the DFR of the concatenated code employed in HQC. We describe a series of optimizations that make the DFR estimates efficiently computable, in turn allowing the use of our closed-form model as a parameter tuning tool. We provide a `C` implementation of our model [8]. We prove that the DFR of HQC meets the required bounds, confirming that the probabilistic model derived in [2] remains conservative in cryptographic-grade regimes. We show that both the public keys and ciphertexts of HQC can be reduced without impacting the security of the cryptosystem, providing a new set of parameters that target both the formally required DFR bound $2^{-\lambda}$, with $\lambda \in \{128, 192, 256\}$, and the bound $2^{-160}$ used in the design of Kyber [1] to provide IND-CCA2.

## II. Preliminaries

Elements in the binary polynomial ring $\mathcal{R} = \mathbb{F}_2[x]/\langle x^p - 1\rangle$ are considered interchangeably as polynomials over the binary field $\mathbb{F}_2$ with degree at most $p-1$ or as vectors in $\mathbb{F}_2^p$. Consequently, addition and multiplication between two polynomials corresponds to component-wise addition and cyclic convolutions of the vectors components, respectively. Given $\mathbf{a}\in\mathcal{R}$, $\mathrm{supp}(\mathbf{a})$ denotes its support (i.e., the set of positions of its non-zero coefficients), while $\mathtt{wt}(\mathbf{a})$ denotes its Hamming weight (i.e., the number of its non-zero components). $\mathcal{R}_\omega$ denotes the group of all polynomials with Hamming weight $\omega$. A Public Key Encryption (PKE) scheme consists of a triplet of polynomial-time algorithms $(\mathsf{KeyGen}, \mathsf{Encrypt}, \mathsf{Decrypt})$, allowing anyone to encrypt messages using the public key, while only the owner of the corresponding private key can decrypt. A PKE scheme that securely encrypts a (relatively short) randomly chosen message rather than a user-provided message is referred to as a KEM.

The PKE scheme underlying the HQC-KEM is defined as follows. The $\mathsf{KeyGen}$ procedure samples a dense polynomial $\mathbf{h}\in\mathcal{R}$, and two sparse polynomials $\mathbf{x},\mathbf{y}\in\mathcal{R}_\omega$. The public key is composed by $\mathbf{h}$, and the syndrome $\mathbf{s} = \mathbf{x} + \mathbf{h}\cdot\mathbf{y}$, while the private key is defined as the $\mathbf{x},\mathbf{y}$ pair. The $\mathsf{Encrypt}$ procedure works by encoding a $k$-bit message $\mathtt{msg}$ into a structured code with $n$-bit codewords, and by adding to such codeword a random error, $\mathbf{e}$, combined with the public key to obtain the ciphertext as a noisy codeword. In particular, it yields two vectors $(\mathbf{u},\mathbf{v})$, where $\mathbf{u} = \mathbf{r_1} + \mathbf{h}\cdot\mathbf{r_2}$, and $\mathbf{v} = \textsc{Encode}(\mathtt{msg}) + \textsc{Truncate}(\mathbf{s}\cdot\mathbf{r_2}+\mathbf{e})$, with $\mathbf{r_1},\mathbf{r_2}\in\mathcal{R}_{\omega_r}$, and $\mathbf{e}\in\mathcal{R}_{\omega_e}$ sparse), where $\textsc{Encode}$ provides the codeword of a structured error-correcting code and $\textsc{Truncate}$ denotes truncation by $p-n$ bits. The $\mathsf{Decrypt}$ routine employs the private key to remove the large part of the noise and subsequently decode the corrupted codeword back to the original message. It computes the vector $\mathbf{v}' = \mathbf{v} - \textsc{Truncate}(\mathbf{u}\cdot\mathbf{y})$, which equals the encoded message plus a sparse error: $\mathbf{v}' = \textsc{Encode}(\mathtt{msg}) + \textsc{Truncate}(\mathbf{x}\cdot\mathbf{r_1}+\mathbf{r_2}\cdot\mathbf{y}+\mathbf{e})$. The message $\mathtt{msg}$ is then recovered by decoding the error-affected codeword $\mathbf{v}'$. The error correction code used in HQC is the concatenation of a Reed-Solomon (RS) and a Duplicated Reed-Muller (DRM) code. The external code is a shortened RS code over $\mathbb{F}_{2^8}$, with length $n_1$ and dimension $\frac{\lambda}{8}$, where $\lambda$ is the size of the original message. The internal code is a DRM code over $\mathbb{F}_2$, with length $n_2$ and dimension 8. The length of the concatenated code is $n = n_1 n_2$, and its dimension is $k = \lambda$.

An attacker, without the secret key, faces the hardness of decoding a general quasi-cyclic code, which is believed to be computationally intractable, since it is closely related to the generic syndrome decoding problem, proven $\mathsf{NP\text{-}hard}$ [9]. However, if a private key is used multiple times, then an attacker can exploit the non-zero DFR of HQC decoder. Since the error $\mathbf{e}'=\mathbf{x}\cdot\mathbf{r_1}+\mathbf{r_2}\cdot\mathbf{y}+\mathbf{e}$ depends on the secret polynomials $\mathbf{x}$ and $\mathbf{y}$, decryption failures can reveal them partially, and several key recovery attacks have been devised for systems with non-negligible DFR [5], [10]–[12]. Consequently, a negligible DFR is a critical requirement for HQC.

## III. Decoding Failure Rate Model

In this section, we derive a closed form DFR estimation model for the decoder of HQC in four steps. In Section III-A, we compute the probability distribution of the Hamming weight of the error in input to the HQC decoder, without any independence assumption on the error coordinates. In Section III-B we describe a model of the DRM code that generalizes the analysis in [2]. In Section III-C we derive the DFR of the RS code, combining this result with the distribution of the error weight to obtain a closed-form DFR. In Section III-D, we show how to compute DFR estimates efficiently, and reduce the sizes of public keys and ciphertexts.

### A. Error Hamming weight distribution

The error in input to HQC decoder is composed by the first $n$ bits of the vector $\mathbf{e}' = \mathbf{x}\cdot\mathbf{r_1}+\mathbf{r_2}\cdot\mathbf{y}+\mathbf{e}$, where $\mathtt{wt}(\mathbf{x}) = \mathtt{wt}(\mathbf{y}) = \omega$, $\mathtt{wt}(\mathbf{r_1}) = \mathtt{wt}(\mathbf{r_2}) = \omega_r$, and $\mathtt{wt}(\mathbf{e}) = \omega_e$. We begin by computing the distribution of $\mathtt{wt}(\mathbf{x}\cdot\mathbf{r_1})$ (and, equivalently, of $\mathtt{wt}(\mathbf{r_2}\cdot\mathbf{y})$). Since $\mathbf{x}\cdot\mathbf{r_1}=\sum_{k\in\mathrm{supp}(\mathbf{x})} x^k\cdot\mathbf{r_1}$, we see that $\mathbf{x}\cdot\mathbf{r_1}$ is equivalent to the sum of $\mathtt{wt}(\mathbf{x}) = \omega$ bitwise (distinct) rotations of $\mathbf{r_1}$. Let $\mathrm{supp}(\mathbf{x})$ be $\{\alpha_0,\alpha_1,\ldots,\alpha_{\omega-1}\}$ and $\mathbf{t}_\ell := \sum_{i=0}^{\ell} x^{\alpha_i}\cdot\mathbf{r_1}$. Define $\mathcal{T}_\ell$ as the random variable modeling the Hamming weight of $\mathbf{t}_\ell$. We have that $\mathbf{t_0} = \mathbf{0}$, which implies $\Pr(\mathcal{T}_0 = 0) = 1$, and that $\mathbf{t}_\omega = \mathbf{x}\cdot\mathbf{r_1}$. We now show that the distribution of $\mathcal{T}_\ell$ can be computed starting from the distribution of $\mathcal{T}_{\ell-1}$, meaning that the p.m.f. of the sequence of random variables $\mathcal{T}_1,\mathcal{T}_2,\ldots,\mathcal{T}_\omega$ can be computed starting from the p.m.f. of $\mathcal{T}_0$. We begin with the following proposition.

**Proposition 1.** *Given $k\in\mathrm{supp}(\mathbf{t_{\ell-1}})$, the probability that $k\in\mathrm{supp}(x^{\alpha_\ell}\cdot\mathbf{r_1})$ is:*

$$p_1 := \frac{\sum_{i=1,\ \mathrm{odd}}^{\min(\ell-1,\omega_r)}\binom{\omega_r}{i}\binom{p-\omega_r}{\ell-1-i}\left(\frac{\omega_r-i}{p-\ell+1}\right)}{\sum_{i=1,\ \mathrm{odd}}^{\min(\ell-1,\omega_r)}\binom{\omega_r}{i}\binom{p-\omega_r}{\ell-1-i}}$$

*Proof.* Consider all the $p$ bitwise rotations of $\mathbf{r_1}$ (i.e., $x^i\cdot\mathbf{r_1}$, $0\le i\le p-1$): since $\mathtt{wt}(\mathbf{r_1}) = \omega_r$ the term $x^k$ appears in exactly $\omega_r$ rotations of $\mathbf{r_1}$. Let $\beta = \{i \mid k\in\mathrm{supp}(x^i\cdot\mathbf{r_1})\}$ be the set of indexes, i.e., rotation amounts, corresponding to such rotations. The probability that $k\in\mathrm{supp}(\mathbf{t_{\ell-1}})$ is equal to the probability that $|\beta\cap\{\alpha_0,\ldots,\alpha_{\ell-1}\}|$ is odd. Since the support of $\mathbf{x}$ (therefore the set $\{\alpha_0,\ldots,\alpha_{\ell-1}\}$) and the support of $\mathbf{r_1}$ are independent, such probability is $\sum_{i=1,\ \mathrm{odd}}^{\min(\ell-1,\omega_r)}\frac{\binom{\omega_r}{i}\binom{p-\omega_r}{\ell-1-i}}{\binom{p}{\ell-1}}$, by a counting argument. Fixing to $y$ the number of rotations of $\mathbf{r_1}$ added to $\mathbf{t_{\ell-1}}$ that include the term $x^k$ (i.e., $|\beta\cap\{\alpha_0,\ldots,\alpha_{\ell-1}\}| = y$), the probability that $k\in\mathrm{supp}(x^{\alpha_\ell}\mathbf{r_1})$ is $\frac{\omega_r-y}{p-(\ell-1)}$. The probability that the term $x^k$ appears in $x^{\alpha_\ell}\mathbf{r_1}$, given that $k\in\mathrm{supp}(\mathbf{t_{\ell-1}})$, is thus $p_1$. □

**Proposition 2.** *Given $k\notin\mathrm{supp}(\mathbf{t_{\ell-1}})$, the probability that $k\in\mathrm{supp}(x^{\alpha_\ell}\cdot\mathbf{r_1})$ is:*

$$p_0 := \frac{\sum_{i=1,\ \mathrm{even}}^{\min(\ell-1,\omega_r)}\binom{\omega_r}{i}\binom{p-\omega_r}{\ell-1-i}\left(\frac{\omega_r-i}{p-\ell+1}\right)}{\sum_{i=1,\ \mathrm{even}}^{\min(\ell-1,\omega_r)}\binom{\omega_r}{i}\binom{p-\omega_r}{\ell-1-i}}$$

*Proof.* The proof is the same as Proposition 1, with the only difference being that the number of rotations of $\mathbf{r_1}$ composing $\mathbf{t_{\ell-1}}$ that include the term $x^k$ is even. □

Given that $\mathbf{t_\ell} = \mathbf{t_{\ell-1}} + x^{\alpha_\ell} \cdot \mathbf{r_1}$, and that the $+$ operation in $\mathcal{R}$ corresponds to a bitwise-xor between the coefficients of the two operands, we have that $\text{supp}(\mathbf{t_\ell}) = \text{supp}(\mathbf{t_{\ell-1}}) \Delta \text{supp}(x^{\alpha_\ell} \cdot \mathbf{r_1})$ (where $\Delta$ denotes symmetric difference between the sets), and therefore:

$$\texttt{wt}(\mathbf{t_\ell}) = \texttt{wt}(\mathbf{t_{\ell-1}}) + \omega_r - 2|\text{supp}(\mathbf{t_{\ell-1}}) \cap \text{supp}(x^{\alpha_\ell} \cdot \mathbf{r_1})| \quad (1)$$

We now prove the following proposition.

**Proposition 3.** *Let $\rho(u, y)$ be the p.m.f. of $|\text{supp}(\mathbf{t_{\ell-1}}) \cap \text{supp}(x^{\alpha_\ell} \cdot \mathbf{r_1})|$ conditioned over $\texttt{wt}(\mathbf{t_{\ell-1}})$, i.e., $\rho(u, y) := \Pr(|\text{supp}(\mathbf{t_{\ell-1}}) \cap \text{supp}(x^{\alpha_\ell} \cdot \mathbf{r_1})| = u \mid \texttt{wt}(\mathbf{t_{\ell-1}}) = y)$. Then:*

$$\rho(u, y) = \frac{\binom{y}{u}\binom{p-y}{\omega_r-u}\left(\frac{p_1\cdot(1-p_0)}{p_0\cdot(1-p_1)}\right)^u}{\sum_{j=\max\{0,\omega_r-(p-y)\}}^{\min\{\omega_r,y\}}\binom{y}{j}\binom{p-y}{\omega_r-j}\left(\frac{p_1\cdot(1-p_0)}{p_0\cdot(1-p_1)}\right)^j}$$

*Proof.* $\Pr(|\text{supp}(\mathbf{t_{\ell-1}}) \cap \text{supp}(x^{\alpha_\ell} \cdot \mathbf{r_1})| = u \mid \texttt{wt}(\mathbf{t_{\ell-1}}) = y)$ is equal to the probability that $u$ elements of the support of $x^{\alpha_\ell} \cdot \mathbf{r_1}$ also appear in the support $\text{supp}(\mathbf{t_{\ell-1}})$ (of size $y$). Proposition 1 defines the probability $p_1$ that an element of $\text{supp}(\mathbf{t_{\ell-1}})$ is also in $\text{supp}(x^{\alpha_\ell} \cdot \mathbf{r_1})$, while Proposition 2 defines the probability $p_0$ that an element not in $\text{supp}(\mathbf{t_{\ell-1}})$ is in $\text{supp}(x^{\alpha_\ell} \cdot \mathbf{r_1})$. Note that, if $p_0 = p_1$ were to hold, then the two supports would be uncorrelated, and their intersection would follow a hypergeometric distribution. Since $p_0 \neq p_1$, the quantity $|\text{supp}(\mathbf{t_{\ell-1}}) \cap \text{supp}(x^{\alpha_\ell} \cdot \mathbf{r_1})|$ follows Fisher's non-central hypergeometric distribution [13], since for all possible values $0 \leq k \leq p-1$, the probability of it being "selected" (i.e., appearing in $\text{supp}(x^{\alpha_\ell} \cdot \mathbf{r_1})$) is not uniform, but depends on whether $k$ belongs to $\text{supp}(\mathbf{t_{\ell-1}})$ or not. □

With this result, we can derive the p.m.f. of $\mathcal{T}_\ell$ by knowing the p.m.f. of $\mathcal{T}_{\ell-1}$.

**Proposition 4.** *The p.m.f. of $\mathcal{T}_\ell$ given the p.m.f. of $\mathcal{T}_{\ell-1}$ is:*

$$\Pr(\mathcal{T}_\ell = z) = \sum_{\substack{y \geq \max\{0, z-\omega_r\} \\ y \leq \min\{p, z+\omega_r\} \\ y+\omega_r \equiv_2 z}} \Pr(\mathcal{T}_{\ell-1} = y) \cdot \rho\left(\frac{y+\omega_r-z}{2}, y\right)$$

*Proof.* Assuming $\texttt{wt}(\mathbf{t_{\ell-1}}) = y$ and $\texttt{wt}(\mathbf{t_\ell}) = z$, from Equation 1 we obtain $|\text{supp}(\mathbf{t_{\ell-1}}) \cap \text{supp}(x^{\alpha_\ell} \cdot \mathbf{r_1})| = \frac{y+\omega_r-z}{2}$. The probability of such a value for $|\text{supp}(\mathbf{t_{\ell-1}}) \cap \text{supp}(x^{\alpha_\ell} \cdot \mathbf{r_1})|$ is $\rho\left(\frac{y+\omega_r-z}{2}, y\right)$ Using the law of total probability over all values of $y$, we obtain the stated formula. □

Since the distribution of $\mathcal{T}_0$ is known, we can compute the p.m.f. of the random variables $\mathcal{T}_1, \mathcal{T}_2, \ldots, \mathcal{T}_\omega$. The last element of this sequence, $\mathcal{T}_\omega$, corresponds to the r.v. modeling the Hamming weight of $\mathbf{x} \cdot \mathbf{r_1}$ (and $\mathbf{r_2} \cdot \mathbf{y}$). We now compute the probability distribution of the Hamming weight of $\mathbf{x} \cdot \mathbf{r_1} + \mathbf{r_2} \cdot \mathbf{y}$.

**Proposition 5.** *Let $\widetilde{\mathcal{T}}$ be the r.v. modeling $\texttt{wt}(\mathbf{x} \cdot \mathbf{r_1} + \mathbf{r_2} \cdot \mathbf{y})$. Then $\Pr(\widetilde{\mathcal{T}} = z) =$*

$$\sum_{0 \leq y_1, y_2 \leq p} \Pr(\mathcal{T}_\omega = y_1) \Pr(\mathcal{T}_\omega = y_2) \frac{\binom{y_1}{\left(\frac{y_1+y_2-z}{2}\right)}\binom{p-y_1}{y_2-\left(\frac{y_1+y_2-z}{2}\right)}}{\binom{p}{y_2}}$$

*Proof.* Given $\texttt{wt}(\mathbf{x} \cdot \mathbf{r_1}) = y_1$ and $\texttt{wt}(\mathbf{r_2} \cdot \mathbf{y}) = y_2$, the size of their intersection follows a hypergeometric distribution. The event $\texttt{wt}(\mathbf{x} \cdot \mathbf{r_1}) = z$ implies that their intersection is equal to $\frac{y_1+y_2-z}{2}$. Using the law of total probability over all possible values of $y_1$ and $y_2$, we obtain the stated formula. □

We conclude the first step of our analysis of the HQC decoder by computing the probability distribution of the Hamming weight of $\mathbf{e'} = \mathbf{x} \cdot \mathbf{r_1} + \mathbf{r_2} \cdot \mathbf{y} + \mathbf{e}$, i.e., the error polynomial to be corrected by the decoder. To this end, we define $\mathcal{E}$ to be the r.v. modeling $\texttt{wt}(\mathbf{e'})$.

**Theorem 1.**

$$\Pr(\mathcal{E} = \epsilon) = \sum_{0 \leq z \leq p} \Pr(\widetilde{\mathcal{T}} = z) \frac{\binom{y}{\left(\frac{y+\omega_e-\epsilon}{2}\right)}\binom{p-z}{\omega_e-\left(\frac{z+\omega_e-\epsilon}{2}\right)}}{\binom{p}{\omega_e}}$$

*Proof.* In Proposition 5, we derive the p.m.f. of the Hamming weight of the sum of two polynomials, $\mathbf{x} \cdot \mathbf{r_1}$ and $\mathbf{r_2} \cdot \mathbf{y}$. We can repeat the same process to derive the p.m.f. of the weight of the sum involving $\mathbf{x} \cdot \mathbf{r_1} + \mathbf{r_2} \cdot \mathbf{y}$ and $\mathbf{e}$. The proof is equivalent to the one of Proposition 5, with the only difference being that the weight of one of the two addends is fixed ($\texttt{wt}(\mathbf{e}) = \omega_e$). □

### *B. Duplicated Reed Muller Model*

Having computed the distribution of $\texttt{wt}(\mathbf{e'})$, the amount of errors to be corrected by the decoder, we provide a generalization of the Duplicated Reed Muller (DRM) DFR model reported in [2]. Our approach is the following: we fix the number of errors to $\texttt{wt}(\mathbf{e'}) = \epsilon$, and we consider the $n_1$ symbols (encoded with the DRM code) of the external Reed Solomon code. Each symbol, of length $n_2$, will include some of the $\epsilon$ errors present in the whole vector $\mathbf{e'}$. Instead of assuming each symbol (decoded with the DRM code) to fail independently from the others, we consider the fact that decoding failures for some DRM codewords imply that the amount of errors left in the other codewords is lower than average. Conversely, decoding successes for some DRM codewords imply that the other instances are more likely to fail. We quantify this line of reasoning by modeling the decoding action of each DRM codeword sequentially (i.e., considering the process as $n_1$ decoding attempts on each symbol), and defining two random variables: *i)* $\mathcal{F}_\ell^\epsilon$, $0 \leq \ell \leq n_1$, modeling the number of failures after $\ell$ steps when $\texttt{wt}(\mathbf{e'}) = \epsilon$ ($0 \leq \mathcal{F}_\ell^t \leq \ell$); *ii)* $\mathcal{W}_\ell^\epsilon$, modeling the number of *errors* corrupting the first $\ell$ symbols ($0 \leq \mathcal{W}_\ell^\epsilon \leq \min\{\epsilon, \ell n_2\}$). At the beginning of the decoding process ($\ell = 0$), we have that $\Pr(\mathcal{F}_0^\epsilon = 0) = 1$ and $\Pr(\mathcal{W}_0^\epsilon = 0) = 1$. At the end of the decoding process, we have that $\mathcal{F}_{n_1}^\epsilon$ models the number of erroneous symbols that must be corrected by the RS code.

In this section, we describe how to compute the joint distribution of $(\mathcal{F}_\ell^\epsilon, \mathcal{W}_\ell^\epsilon)$, starting from the distribution of $(\mathcal{F}_{\ell-1}^\epsilon, \mathcal{W}_{\ell-1}^\epsilon)$, by providing a generalization of the formulas for the DRM decoding failure probability derived in [2]. We start by reporting the following fact, proven in [2, Proposition 6.1.4].

**Lemma 1.** *Consider a DRM code with length $n_2$, dimension 8 and minimum distance $d_2$. The number of weight-$t$ error patterns ($\frac{d_2}{2} \leq t \leq n_2$) that lead to a decoding failure can be upper bounded by $\mathfrak{U}_t =$*

$$\frac{1}{2}255\binom{d_2}{\binom{d_2}{2}}\binom{d_2}{\left(t-\frac{d_2}{2}\right)} + 255\sum_{j=\frac{d_2}{2}+1}^{d_2}\binom{d_2}{j}\binom{d_2}{t-j} +$$

$$+\frac{1}{2}\binom{255}{2}\sum_{j=0}^{\frac{d_2}{2}}\binom{\binom{d_2}{2}}{j}^3\binom{\binom{d_2}{2}}{t-d_2+j}$$

We extend the definition of $\mathfrak{U}_t$ by considering $\mathfrak{U}_t = 0$ for $0 \leq t \leq \frac{d_2}{2} - 1$. Since the error pattern is independent from the codewords of the DRM code, the probability $\mathfrak{p}_t$ that a weight-$t$ error pattern causes a decoding failure can be upper bounded by $\min\left\{1, \mathfrak{U}_t/\binom{n_2}{t}\right\}$. When $\mathfrak{p}_t$ is sufficiently high, a tighter upper bound for $\mathfrak{p}_t$ can be easily computed by means of numerical simulation and conservative statistical tests: in particular, for our estimates we performed $10^6$ decoding trials for each value of $t$, and defined $\mathfrak{p}_t$ as the minimum between $\mathfrak{U}_t/\binom{n_2}{t}$ and the 90% upper confidence bound obtained from the simulations. We can use $\mathfrak{p}_t$ to compute the probability distribution of the joint r.v. $(\mathcal{F}_\ell^\epsilon, \mathcal{W}_\ell^\epsilon)$, conditioned over $(\mathcal{F}_{\ell-1}^\epsilon, \mathcal{W}_{\ell-1}^\epsilon)$.

**Proposition 6.** *Consider a DRM code with length $n_2$, dimension 8 and minimum distance $d_2$, and assume $(\mathcal{F}_{\ell-1}^\epsilon, \mathcal{W}_{\ell-1}^\epsilon) = (f, w)$. Then:*

$$\Pr\left((\mathcal{F}_\ell^\epsilon, \mathcal{W}_\ell^\epsilon) = (f, w+t) | (\mathcal{F}_{\ell-1}^\epsilon, \mathcal{W}_{\ell-1}^\epsilon) = (f, w)\right) = \frac{\binom{n_2}{t}\binom{p-\ell n_2}{\epsilon-w-t}}{\binom{p-(\ell-1)n_2}{\epsilon-w}} \cdot (1-\mathfrak{p}_t),$$

$$\Pr\left((\mathcal{F}_\ell^\epsilon, \mathcal{W}_\ell^\epsilon) = (f+1, w+t) | (\mathcal{F}_{\ell-1}^\epsilon, \mathcal{W}_{\ell-1}^\epsilon) = (f, w)\right) = \frac{\binom{n_2}{t}\binom{p-\ell n_2}{\epsilon-w-t}}{\binom{p-(\ell-1)n_2}{\epsilon-w}} \cdot \mathfrak{p}_t$$

*Proof.* While decoding the $\ell$-th symbol, the fact that $(\mathcal{F}_{\ell-1}^\epsilon, \mathcal{W}_{\ell-1}^\epsilon) = (f, w)$ implies that a total of $w$ errors out of $\epsilon$ lie in the first $(\ell-1)$ symbols (i.e., in the first $(\ell-1)n_2$ bits of $\mathbf{e}'$), meaning that a total of $\epsilon - w$ errors lie in the last $p-(\ell-1)n_2$ bits of $\mathbf{e}'$, where also the $\ell$-th symbol is located. Given $\mathcal{W}_{\ell-1}^\epsilon = w$, the number of errors affecting the $\ell$-th symbol (having length $n_2$) follows a hypergeometric distribution, therefore $\Pr(\mathcal{W}_\ell^\epsilon = w+t | \mathcal{W}_{\ell-1}^\epsilon = w) = \frac{\binom{n_2}{t}\binom{p-\ell n_2}{\epsilon-w-t}}{\binom{p-(\ell-1)n_2}{\epsilon-w}}$. Given $\mathcal{W}_{\ell-1}^\epsilon = w$ and $\mathcal{W}_\ell^\epsilon = w+t$, the probability that the $\ell$-th decoding action fails (i.e., $\mathcal{F}_\ell^\epsilon = \mathcal{F}_{\ell-1}^\epsilon + 1$) is $\mathfrak{p}_t$, and the probability that the $\ell$-th decoding action succeeds (i.e., $\mathcal{F}_\ell^\epsilon = \mathcal{F}_{\ell-1}^\epsilon$) is $1-\mathfrak{p}_t$. □

We can apply the law of total probability to the result of Proposition 6, obtaining the following.

**Theorem 2.** $\Pr\left((\mathcal{F}_\ell^\epsilon, \mathcal{W}_\ell^\epsilon) = (f, w)\right) =$

$$= \sum_{t=0}^{n_2}\Pr\left((\mathcal{F}_{\ell-1}^\epsilon, \mathcal{W}_{\ell-1}^\epsilon) = (f, w-t)\right) \cdot \frac{\binom{n_2}{t}\binom{p-\ell n_2}{\epsilon-w}}{\binom{p-(\ell-1)n_2}{\epsilon-w+t}} \cdot (1-\mathfrak{p}_t) +$$
$$+ \sum_{t=0}^{n_2}\Pr\left((\mathcal{F}_{\ell-1}^\epsilon, \mathcal{W}_{\ell-1}^\epsilon) = (f-1, w-t)\right) \cdot \frac{\binom{n_2}{t}\binom{p-\ell n_2}{\epsilon-w}}{\binom{p-(\ell-1)n_2}{\epsilon-w+t}} \cdot \mathfrak{p}_t$$

### C. Reed Solomon Model

We now employ the analysis in the previous section to derive the DFR of the external RS code and, thus, of the decoder of HQC. Since the distribution of $(\mathcal{F}_0^\epsilon, \mathcal{W}_0^\epsilon)$ is known, with Theorem 2 we can compute the probability distribution of $(\mathcal{F}_1^\epsilon, \mathcal{W}_1^\epsilon)$, $(\mathcal{F}_2^\epsilon, \mathcal{W}_2^\epsilon)$, …, up to $(\mathcal{F}_{n_1}^\epsilon, \mathcal{W}_{n_1}^\epsilon)$. The r.v. $\mathcal{F}_{n_1}^\epsilon$ models the number of instances where the DRM decoder failed at recovering the correct symbol. Since the external RS decoder corrects up to half its minimum distance $d_1$, we have that the probability of failure, given the Hamming weight of the error $\mathtt{wt}(\mathbf{e}') = \epsilon$, is $\Pr(\mathcal{F}_{n_1}^\epsilon \geq \frac{d_1-1}{2})$. Combining this result with the distribution of the Hamming weight of the error $\mathtt{wt}(\mathbf{e}')$, provided by Theorem 1, we get the following closed-form model for the decoding failure rate DFR of HQC.

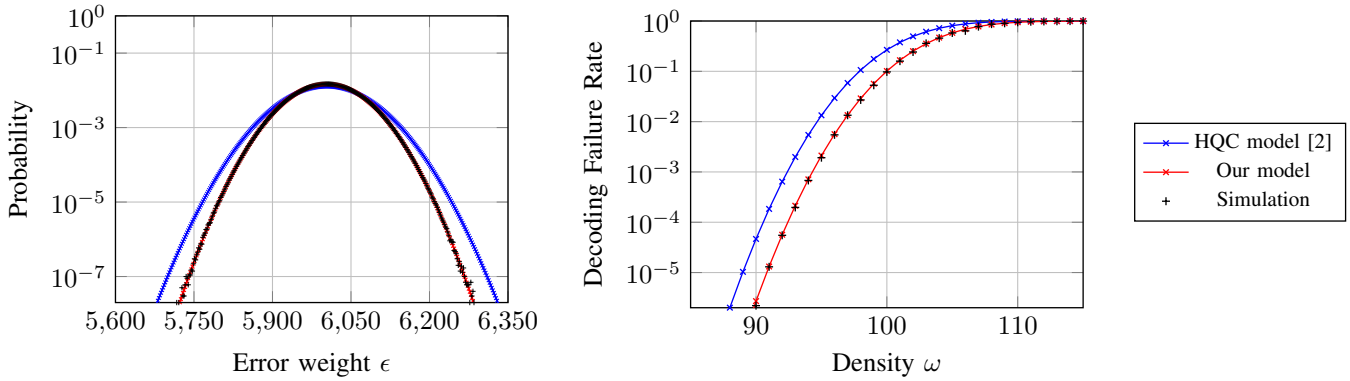
**Theorem 3.** *The DFR of the concatenated code employing a Duplicated Reed Muller code with length $n_2$, dimension 8, and distance $d_2$ as internal code, and a Reed Solomon code with length $n_1$ and distance $d_1$ as external code is:*

$$\mathtt{DFR} = \sum_{\epsilon=0}^{p}\left[\Pr(\mathcal{E}=\epsilon)\sum_{w=0}^{\epsilon}\sum_{f=\frac{d_1-1}{2}+1}^{n_1}\Pr\left((\mathcal{F}_\ell^\epsilon, \mathcal{W}_\ell^\epsilon) = (f, w)\right)\right]$$

### D. Optimizations and Parameter Tuning

In this section, we describe a set of techniques that can be employed to speed up the computation of the DFR estimates without altering the correctness of the results, and we show how our DFR model can be employed as a parameter tuning tool for computing optimal HQC parameters (no changes to the cryptosystem implementation are required).

Since the p.m.f. $\Pr(\mathcal{E}=\epsilon)$ can be efficiently computed by parallelizing the calculations across all values of $\epsilon$, our goal is to reduce the number of values of $\epsilon$ for which $f(\epsilon) = \Pr(\mathcal{F}_{n_1}^\epsilon \geq \frac{d_1-1}{2})$ must be computed. We start by pointing out that, given two close values $\epsilon_1$ and $\epsilon_2$, the two failure probabilities $f(\epsilon_1) = \Pr(\mathcal{F}_{n_1}^{\epsilon_1} \geq \frac{d_1-1}{2})$ and $f(\epsilon_2) = \Pr(\mathcal{F}_{n_1}^{\epsilon_2} \geq \frac{d_1-1}{2})$ do not differ significantly. We exploit this fact by computing $f(\epsilon)$ only in a subset of points $\mathcal{V} \subset \{0, 1, \ldots, p\}$, and then defining $f_\mathcal{V}(\epsilon)$ by interpolating the samples $(\mathcal{V}, f(\mathcal{V}))$ (e.g., with a simple linear interpolation between the points). What we can compute is an approximation of DFR, $\widetilde{\mathtt{DFR}}(\mathcal{V})$, defined as $\widetilde{\mathtt{DFR}}(\mathcal{V}) = \sum_{\epsilon=0}^{p}\Pr(\mathcal{E}=\epsilon)\cdot f_\mathcal{V}(\epsilon)$. As the number of points in $\mathcal{V}$ increases, $f_\mathcal{V}(\epsilon)$ converges to $f(\epsilon)$, and therefore $\widetilde{\mathtt{DFR}}(\mathcal{V})$ converges to DFR. We employ an iterative strategy, computing $f(\epsilon)$ over a sequence of point sets $\mathcal{V}_0 \subset \mathcal{V}_1 \subset \ldots \subset \mathcal{V}_i$, until $\widetilde{\mathtt{DFR}}(\mathcal{V}_i)$ has converged (i.e., its value did not change in a numerically relevant fashion across multiple iterations). While doing so, we skip the values of $\epsilon$ for which $\Pr(\mathcal{E}=\epsilon) < \widetilde{\mathtt{DFR}}(\mathcal{V}) \cdot 10^{-16}$, as their contribution to




(a) Error weight distribution

(b) Decoding Failure Rate

Fig. 1. Numerical simulations on HQC-1 ($p$=17669, $n_1$=46, $n_2$=384, $\omega$=66, $\omega_r$=$\omega_e$ = 75), compared to the model presented in HQC's specification [2] and our model. (a) Probability distribution of $\mathtt{wt}(\mathbf{e}')$, from $10^8$ generated error polynomials; (b) DFR with $10^6$ decryptions per data point.

the overall DFR estimate can be safely neglected. With these techniques, the time required to estimate the DFR of HQC parameters is reduced from days to a few minutes, without impacting the accuracy of results, allowing for an exhaustive search for optimal parameters. For the parameter sets provided in the next section, we validated the accuracy of the optimized model with an implementation of the full estimation model.

Our goal is to find the optimal parameters for HQC not requiring changes in the implementation of the cryptosystem, therefore we do not modify the structure of the concatenated code, nor the parameters of the internal RM code, allowing a drop-in parameter change in the current HQC structure. Our approach for optimizing the parameters of HQC consists in shortening the external RS code $n_1$ by 2 symbols at a time, reducing then $p$ until either $p < n_1 n_2$ or the DFR requirement is not satisfied. With this strategy, we iteratively reduce the length of both the QC and the RS code, shortening public key and ciphertext. We note that, from a security perspective, changes in $p$ and $n_1$ do not significantly affect the computational complexity of message and key recovery attacks, as long as $p$ is selected to be a prime such that 2 is a generator of $\mathbb{F}_p^*$. Moreover, while in HQC the difference between $p$ and $n = n_1 n_2$ is minimized, $p$ can be chosen to be slightly larger than $n_1 n_2$ with no impact on security.

## IV. Numerical Results

We now present the results of numerical validations of the correctness of our DFR model, and confirm that current HQC parameters are secure by estimating their DFR. We also propose new parameter sets for HQC, providing a slight reduction in public key and ciphertext size, and propose two sets following the DFR criterion adopted in ML-KEM [1], providing a more significant public key/ciphertext size reduction. Figure 1a shows the results of numerical simulations targeting the distribution of the Hamming weight of $\mathbf{e}'$, the error vector in input to the decoder. These results show that our model provides a better fit on the empirical distribution with respect to the binary symmetric channel model [2]. Figure 1b depicts

TABLE I
HQC original and proposed parameters, public key and ciphertext sizes in bytes, DFR estimates

| From | Instance | $p$ | $n_1$ | $\lvert\mathtt{pk}\rvert$ | $\lvert\mathtt{ctx}\rvert$ | Target DFR | Est. DFR |
|---|---|---|---|---|---|---|---|
| [2] | HQC-1 | 17669 | 46 | 2241 | 4433 | $2^{-128}$ | $2^{-147.85}$ |
| [2] | HQC-3 | 35851 | 56 | 4514 | 8978 | $2^{-192}$ | $2^{-207.64}$ |
| [2] | HQC-5 | 57637 | 90 | 7237 | 14421 | $2^{-256}$ | $2^{-285.53}$ |
| Our | HQC-1 | 17387 | 44 | 2206 | 4302 | $2^{-128}$ | $2^{-128.43}$ |
| Our | HQC-3 | 35797 | 54 | 4507 | 8811 | $2^{-192}$ | $2^{-192.80}$ |
| Our | HQC-5 | 56891 | 88 | 7144 | 14168 | $2^{-256}$ | $2^{-256.97}$ |
| Our | HQC-3 | 34589 | 54 | 4356 | 8660 | $2^{-160}$ | $2^{-165.49}$ |
| Our | HQC-5 | 53453 | 82 | 6714 | 13258 | $2^{-160}$ | $2^{-160.24}$ |

the comparison between our model, the model from [2], and numerical simulations, on the DFR of an instance of HQC-1, where the density $\omega$ of the private vectors is increased until decoding failures can be simulated. The figure shows how the current DFR estimate is conservative, while our model precisely predicts the DFR of the concatenated code.

In the security analysis of HQC [2], the DFR estimates assuming the independence of the error vector coordinates was said to be believed to yield conservative estimates also for very low failure rates. We employ our precise DFR estimation technique to check such statement on current HQC parameters. To do so, in Table I we report our DFR estimate for HQC-1, HQC-3 and HQC-5, which require a failure probability at most $2^{-128}, 2^{-192}, 2^{-256}$, respectively. With our closed-form model, we can confirm that all parameter sets of HQC are secure, with a conservative margins as large as $2^{29}$.

We now investigate whether the lengths of the QC code ($p$) and of the RS code ($n_1$) can be reduced or not, while maintaining the required DFR. Using the procedure described in Section III-D, we answer this question in the affirmative, proposing a new set of parameters for HQC-1, HQC-3 and HQC-5, shown in Table I. Compared to the current HQC parameters, our proposal allows for a reduction in public key and ciphertext sizes. The parameter reduction yields a 0.2

TABLE II
ESTIMATED COMPUTATIONAL COMPLEXITY OF MESSAGE RECOVERY AND KEY RECOVERY ATTACKS AGAINST HQC, WITH LOGARITHMIC MEMORY ACCESS COST

| From | Instance | $p$ | $\omega$ | $\omega_r = \omega_e$ | Complexity [14] |
|---|---|---|---|---|---|
| [2] | HQC-1 | 17669 | 66 | 75 | 129.3342 |
| [2] | HQC-3 | 35851 | 100 | 114 | 197.4658 |
| [2] | HQC-5 | 57637 | 131 | 149 | 259.6087 |
| Our | HQC-1 | 17387 | 66 | 75 | 129.2824 |
| Our | HQC-3 | 35797 | 100 | 114 | 197.4611 |
| Our | HQC-5 | 56891 | 131 | 149 | 259.5679 |
| Our | HQC-3 | 34589 | 100 | 114 | 197.3524 |
| Our | HQC-5 | 53453 | 131 | 149 | 259.3743 |

bits loss of security margin according to the public attack complexity estimators available at [14], as shown in Table II, leaving our parameters compliant with NIST security levels.

Finally, in Table I we also show parameters with DFR $2^{-160}$ for HQC-3 and HQC-5, following the point made by Kyber [1] in its standardization document as ML-KEM [15]: attacks based on a DFR lower than $2^{-160}$ are considered infeasible, regardless of the security level. This approach leads to significantly larger savings; in particular, for HQC-5, the size of the public key can be reduced by $523$ bytes, while the ciphertext size can be reduced by $1163$ bytes.

## REFERENCES


[1] R. Avanzi, J. Bos, L. Ducas, E. Kiltz, T. Lepoint, V. Lyubashevsky, J. M. Schanck, P. Schwabe, G. Seiler, and D. Stehlé, "CRYSTALS-Kyber: Algorithm Specifications and Supporting Documentation (version 3.02)," August 2021. [Online]. Available: https://pq-crystals.org/kyber/data/kyber-specification-round3-20210804.pdf

[2] C. A. Melchor, N. Aragon, S. Bettaieb, L. Bidoux, O. Blazy, J. Bos, J.-C. Deneuville, A. Dion, P. Gaborit, J. Lacan, E. Persichetti, J.-M. Robert, P. Véron, and G. Zémor, "Hamming Quasi-Cyclic (HQC) - Fourth round version," https://pqc-hqc.org/doc/hqc-specification_2024-02-23.pdf, 2025.

[3] G. Alagic, M. Bros, P. Ciadoux, D. Cooper, Q. Dang, T. Dang, J. Kelsey, J. Lichtinger, Y.-K. Liu, C. Miller, D. Moody, R. Peralta, R. Perlner, A. Robinson, H. Silberg, D. Smith-Tone, and N. Waller, "Status Report on the Fourth Round of the NIST Post-Quantum Cryptography Standardization Process," https://doi.org/10.6028/NIST.IR.8545, March 2025.

[4] M. Alekhnovich, "More on Average Case vs Approximation Complexity," in *44th Symposium on Foundations of Computer Science (FOCS 2003), 11-14 October 2003, Cambridge, MA, USA, Proceedings*. IEEE Computer Society, 2003, pp. 298–307. [Online]. Available: https://doi.org/10.1109/SFCS.2003.1238204

[5] Q. Guo and T. Johansson, "A new decryption failure attack against HQC," in *Advances in Cryptology - ASIACRYPT 2020 - 26th International Conference on the Theory and Application of Cryptology and Information Security, Daejeon, South Korea, December 7-11, 2020, Proceedings, Part I*, ser. Lecture Notes in Computer Science, S. Moriai and H. Wang, Eds., vol. 12491. Springer, 2020, pp. 353–382. [Online]. Available: https://doi.org/10.1007/978-3-030-64837-4_12

[6] M. Bombar, N. Resch, and E. Wiedijk, "On the independence assumption in quasi-cyclic code-based cryptography," in *IEEE International Symposium on Information Theory, ISIT 2025, Ann Arbor, MI, USA, June 22-27, 2025*. IEEE, 2025, pp. 1–6. [Online]. Available: https://doi.org/10.1109/ISIT63088.2025.11195347

[7] A. Mesnard, J.-P. Tillich, and V. Vasseur, "The syndrome weight distribution in quasi-cyclic codes, applications to BIKE and HQC," Cryptology ePrint Archive, Paper 2025/2218, 2025. [Online]. Available: https://eprint.iacr.org/2025/2218

[8] A. Annechini, A. Barenghi, and G. Pelosi, "C implementation of the DFR model," https://barenghi.faculty.polimi.it/lib/exe/fetch.php?media=hqc_dfr_implementation.zip, 2026.

[9] E. R. Berlekamp, R. J. McEliece, and H. C. A. van Tilborg, "On the inherent intractability of certain coding problems (Corresp.)," *IEEE Trans. Inf. Theory*, vol. 24, no. 3, pp. 384–386, 1978. [Online]. Available: https://doi.org/10.1109/TIT.1978.1055873

[10] Q. Guo, T. Johansson, and P. Stankovski, "A Key Recovery Attack on MDPC with CCA Security Using Decoding Errors," in *Advances in Cryptology - ASIACRYPT 2016 - 22nd International Conference on the Theory and Application of Cryptology and Information Security, Hanoi, Vietnam, December 4-8, 2016, Proceedings, Part I*, ser. Lecture Notes in Computer Science, J. H. Cheon and T. Takagi, Eds., vol. 10031, 2016, pp. 789–815. [Online]. Available: https://doi.org/10.1007/978-3-662-53887-6_29

[11] Q. Guo, T. Johansson, and P. S. Wagner, "A Key Recovery Reaction Attack on QC-MDPC," *IEEE Trans. Inf. Theory*, vol. 65, no. 3, pp. 1845–1861, 2019. [Online]. Available: https://doi.org/10.1109/TIT.2018.2877458

[12] T. Wang, A. Wang, and X. Wang, "Exploring Decryption Failures of BIKE: New Class of Weak Keys and Key Recovery Attacks," in *Advances in Cryptology - CRYPTO 2023 - 43rd Annual International Cryptology Conference, CRYPTO 2023, Santa Barbara, CA, USA, August 20-24, 2023, Proceedings, Part III*, ser. Lecture Notes in Computer Science, H. Handschuh and A. Lysyanskaya, Eds., vol. 14083. Springer, 2023, pp. 70–100. [Online]. Available: https://doi.org/10.1007/978-3-031-38548-3_3

[13] W. L. Harkness, "Properties of the Extended Hypergeometric Distribution," *The Annals of Mathematical Statistics*, vol. 36, no. 3, pp. 938 – 945, 1965. [Online]. Available: https://doi.org/10.1214/aoms/1177700066

[14] A. Esser, J. A. Verbel, F. Zweydinger, and E. Bellini, "Sok: Cryptographicestimators - a software library for cryptographic hardness estimation," in *Proceedings of the 19th ACM Asia Conference on Computer and Communications Security, ASIA CCS 2024, Singapore, July 1-5, 2024*, J. Zhou, T. Q. S. Quek, D. Gao, and A. A. Cárdenas, Eds. ACM, 2024. [Online]. Available: https://doi.org/10.1145/3634737.3645007

[15] Kevin M. Stine, Director Information Technology Laboratory, "FIPS 2023 Federal Information Processing Standards Publication - Module-Lattice-Based Key-Encapsulation Mechanism Standard (ML-KEM)," https://doi.org/10.6028/NIST.FIPS.203, August 2024, U.S. Department of Commerce, Information Technology Laboratory, National Institute of Standards and Technology, Gaithersburg, MD 20899-8900, USA.