Theoretical analysis of decoding failure rate of non–binary QC–MDPC codes

Kirill Vedenev[1][0000−0002−7893−655X] and Yury Kosolapov[1][0000−0002−1491−524X]

Southern Federal University, Rostov-on-Don, Russia
vedenevk@gmail.com
puzzlestorage@gmail.com

Abstract. In this paper, we study the decoding failure rate (DFR) of non-binary QC-MDPC codes using theoretical tools, extending the results of previous binary QC-MDPC code studies. The theoretical estimates of the DFR are particularly significant for cryptographic applications of QC-MDPC codes. Specifically, in the binary case, it is established that exploiting decoding failures makes it possible to recover the secret key of a QC-MDPC cryptosystem. This implies that to attain the desired security level against adversaries in the CCA2 model, the decoding failure rate must be strictly upper-bounded to be negligibly small. In this paper, we observe that this attack can also be extended to the non–binary case as well, which underscores the importance of DFR estimation. Consequently, we study the guaranteed error–correction capability of non–binary QC–MDPC codes under one–step majority logic (OSML) decoder and provide a theoretical analysis of the 1–iteration parallel symbol flipping decoder and its combination with OSML decoder. Utilizing these results, we estimate the potential public-key sizes for QC-MDPC cryptosystems over $\mathbb{F}_4$ for various security levels. We find that there is no advantage in reducing key sizes when compared to the binary case.

Keywords: code–based cryptography · non–binary MDPC codes · symbol flipping · decoding failure rate.

Introduction

With the advent of quantum computers, many traditional public–key cryptosystems based on number–theoretic or elliptic curves primitives are to become vulnerable to attacks using them [14, 42]. So, there is a strong need in developing *post-quantum* cryptographic protocols that will remain secure against adversaries equipped with quantum computers. One of the most prominent and well-established approach to post-quantum cryptography is cryptography based on error-correcting codes.

The first code–based cryptosystem was proposed in 1978 by R. McEliece [31]. The main idea of the McEliece cryptosystem is to mask a generator matrix of a fast–decodable code by permuting its columns and multiplying by a scrambling matrix on the left. The encryption is performed by encoding a message using the public generator matrix and adding an error. So, the security against *message–recovery attacks* is based on NP–hard syndrome decoding problem [13]. The original proposal of R. McEliece was based on binary Goppa codes, so the security against *key–recovery attack* relies on

hardness of the problem of distinguishing permuted Goppa codes. It is worth mentioning that the original McEliece cryptosystem with several improvements is one of three Round 4 competitors in NIST-PQC [1]. Despite many advantages, the main drawback of McEliece cryptosystem is large public–key size. There were many attempts to overcome this by replacing Goppa codes with more efficient ones. The notable examples are Generalized Reed–Solomon codes [35], Reed–Muller codes [43], algebraic geometry codes [29], concatenated codes [40], rank–metric Gabidulin codes [22]. However, most of this modifications were proven unsecure [15, 17, 32, 38, 40, 44]. In addition, several modifications of protocol itself were proposed to avoid key–recovery attacks against McEliece–like cryptosystems based on efficient algebraic codes (e.g. [7, 12, 28, 48]), however most of them were also successfully cryptanalyzed [16, 18, 19, 30, 47, 50].

One of the most efficient approaches to reducing public-key size was proposed by P. Gaborit [23] and is based on using *quasi-cyclic codes (QC-codes)*. A code C of length $n = n'l$ is said to be quasi-cyclic of order n' and index l if its permutation automorphism group $\mathrm{PAut}(C)$ has a cyclic subgroup of order n' that acts freely on coordinates. The quasi-cyclic structure implies the existence of generator and parity-check matrices of C that admit a *block-circulant* representation, i.e.

$$
\begin{pmatrix} \mathrm{rot}(h_{1,1}) & \dots & \mathrm{rot}(h_{1,l}) \\ \vdots & \ddots & \vdots \\ \mathrm{rot}(h_{s,1}) & \dots & \mathrm{rot}(h_{s,l}) \end{pmatrix}, \quad \mathrm{rot}(a_1, a_2, \dots, a_{n'}) = \begin{pmatrix} a_1 & a_2 & \dots & a_{n'} \\ a_n & a_1 & \dots & a_{n'-1} \\ \vdots & \vdots & \ddots & \vdots \\ a_2 & a_3 & \dots & a_1 \end{pmatrix}. \tag{1}
$$

This representation allows storing only the first row of each circulant block $\mathrm{rot}(h_{i,j})$, thereby reducing storage and communication costs. Therefore, the public key sizes of code-based encryption protocols that preserve quasi-cyclic structure can be significantly reduced. Note that many encryption protocols based on *algebraic* QC–codes (e.g. [12,23]) have been successfully attacked [20, 36]. However, protocols based on *random quasi–cyclic moderate density parity–check (QC–MDPC) codes* [33], which have no algebraic structure except being quasi–cyclic, are still considered secure and efficient.

The concept of *moderate-density parity-check (MDPC) codes* extends the idea of *low-density parity-check codes (LDPC codes)* initially introduced by R. Gallager [24]. In Gallager's seminal work [24], it was shown that efficient decoding of binary codes with a parity-check matrix containing a very small constant number of ones in each row is feasible using iterative algorithms such as bit-flipping and belief propagation, provided certain conditions are met (no two rows have two or more ones in the same positions). In 2000, C. Monico et al. [34] considered replacing Goppa codes in the McEliece cryptosystem with LDPC codes and pointed out that these codes can be easily distinguished due to the existence of very low–weight codewords in the dual code. The application of quasi-cyclic LDPC codes in constructing code-based cryptosystems was initially proposed in [11] and further developed in [8, 10]. To mitigate key-recovery attacks based on searching for low-weight dual codewords, it was suggested to replace the permutation matrix in the protocol with a sparse non–singular matrix of a specific form. However, this approach was found to introduce serious vulnerabilities [2, 36]. An alternative method to prevent key–recovery based on the search for low–weight codewords was proposed in [33], where it was suggested to use random QC–MDPC

codes instead of LDPC. The difference between MDPC and LDPC codes lies in the slightly higher weight of the rows in the parity-check matrices, i.e., which is of order $O(\sqrt{n})$ for MDPC codes and $O(1)$ for LDPC.

We denote the finite field of size q as $\mathbb{F}_q$. For a vector $v \in \mathbb{F}_q^n$, the notation $\mathrm{supp}(v) = \{i \in [\![1, n]\!] \mid v_i \neq 0\}$ is used to represent the set of indices corresponding to the positions where v is nonzero. Here, $[\![a, b]\!] = \{a, a + 1, \dots, b\}$ denotes set of all integers between a and b. The Hamming weight of vector v, denoted as $\mathrm{wt}(v)$, is defined as the number of nonzero positions in v. A linear code $C \in \mathbb{F}_q^n$ of length n and dimension k is refereed as $[n, k]_q$–code. A generic description of a QC-MDPC cryptosystem [33] in the Niedderiter form [35] is as follows:

– **Key generation** The secret key is the parity-check matrix H of a random QC-MDPC $[n = ln', (l - 1)n']_q$-code, represented as

$$H = \left(H_1 \mid H_2 \mid \ \dots \ \mid H_{l-1} \mid H_l \right). \tag{2}$$

The matrix H consists of circulant $(n' \times n')$–blocks H_i, where each H_i has a row weight of γ. Note that n' is usually chosen to be a prime number p. The public key is the systematic form of H, i.e.

$$\tilde{H} = H_1^{-1} H = \left(I_{n'} \mid H_1^{-1} H_2 \mid \ \dots \ \mid H_1^{-1} H_l, \right)$$

which can be represented by the first rows of $H_1^{-1} H_i$, where $i \in [\![2, l]\!]$, since the product of circulant matrices is also a circulant matrix.
– **Encryption** The plaintext is an error vector $e \in \mathbb{F}_q^n$ of weight t, and the ciphertext is its syndrome $\tilde{s} = \tilde{H} e^\mathsf{T}$.
– **Decryption** To decrypt, the private syndrome $s = H e^\mathsf{T} = H_1 \tilde{s}^\mathsf{T}$ is computed and used as input for the MDPC decoder (bit-flipping or symbol flipping).

Note that in NIST-PQC, the QC–MDPC approach is represented by *BIKE (bit-flipping key encapsulation)* protocol [3].

Due to probabilistic nature of decoding of LDPC and MDPC codes, there is a non–zero probability of decryption failure. In [26] it was shown that decryption failures can be used to recover the secret key in binary case. Hence in order to achieve *indistinguishability against chosen ciphertext attacks*, where an adversary has an access to a decryption oracle (IND–CCA2 *security*), the *decoding failure rate (DFR)* has to be negligibly small, i.e. of order $2^{-\lambda}$, where λ is a security level. In [46], an experimental–based extrapolation framework for estimating DFR has been proposed. In this approach, the DFR curve is assumed to be concave, so estimates for high DFR ($> 10^{-9}$) can be obtained via numerical simulations and then extrapolated to low DFRs providing an upper bound. However, it is known that LDPC and MDPC codes exhibit error floor phenomenon, resulting in violation of concavity assumption (see e.g. [4,6]). Hence DFR estimates obtained by extrapolation could possibly be overly optimistic (see Figure 1). Another approach is to estimate DFR using only theoretical tools. In [45] J. P. Tillich studied guaranteed error–correction performance of binary QC–MDPC codes under one–step majority logic decoder (OSML). In addition, in [45] the DFR of two–iteraion decoder is studied under some reasonable assumptions, i.e. the probability that one
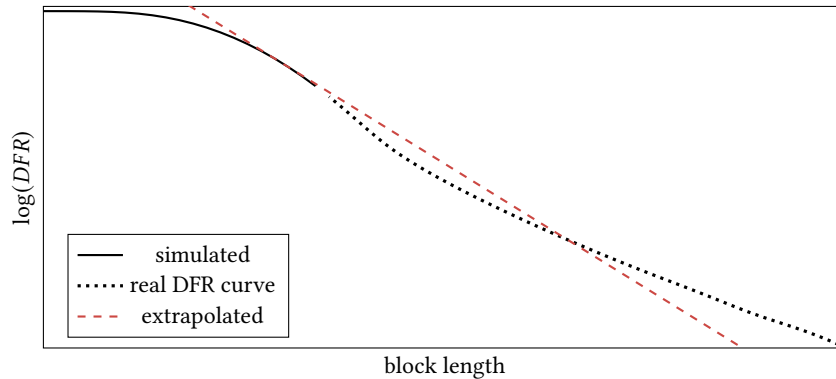
Fig. 1. Approximate illustration of a situation where the use of extrapolation may lead to an incorrect estimation of DFR due to the presence of an error floor.

iteration of parallel bit–flipping decoder reduces error weight enough to be corrected by OSML decoder is computed. In [39], the estimate of the number of errors correctable by OSML decoder was improved. Under the same assumptions as in [45], the worst–case plausibility analysis of one and two iteration randomized serial bit–flipping decoder was performed in [5]. In addition, in [5] a combination of one iteration of randomized serial bit-flipping and OSML was studied, and recommended design parameters for IND–CCA2 secure QC-MDPC cryptosystems were given.

In this paper, we study DFR of *non–binary* QC–MDPC codes using theoretical tools. Namely, we extend the results of [39, 45] to the non–binary case, i.e. we show that error–correcting performance of OSML decoder can also be estimated using similar methods of [39, 45]. In addition, we propose a parallel symbol flipping decoder. Under the same assumptions used in [5, 45], we give theoretical estimates of DFR for the parallel symbol–flipping decoder and its combination with the OSML decoder. We also note that the extension of the randomized serial approach, as considered in [5], in the non-binary case seems to yield unreliable results due to a observed discrepancy between the theoretical estimates and the worst-case simulations. Hence this approach is not included in this paper. In addition, we experimentally demonstrate that slightly modified attack of [26] can also recover secret key in non–binary case. Employing the obtained results, recommended parameters and corresponding key sizes for IND-CCA2–secure QC–MDPC cryptosystems over $\mathbb{F}_q$ for various security levels are computed.

The paper is organized as follows. In Section 1, we present the basic principles of decoding non-binary QC-MDPC codes and study the guaranteed error-correction capability of the one-step majority logic decoder in an assumption-free setting. In Section 2, we provide a plausibility analysis of error counters distribution and flipping probability in the non-binary case. Subsequently, we propose a 1-iteration parallel symbol flipping decoder and theoretically estimate the probability of reducing the error weight to a certain value, allowing for further decoding by the OSML decoder. We also provide experimental validation of the theoretical model. Finally, in Section 3, we

consider the reaction attack against non–binary QC–MDPC cryptosystems and find potential cryptosystem parameters and corresponding public-key sizes.

1 Analysis of guaranteed error–correction capability of non–binary QC-MDPC codes

Recall that a code C with a parity–check matrix $H \in \mathbb{F}_q^{m \times n}$ is said to be a *moderate–density parity–check (MDPC) code* if each row of the $H = (h_{i,j})$ is of weight $O(\sqrt{n})$. In addition, C is said to be (γ, δ)–regular if the weight of each column of H is γ and the weight of each row is δ. Unless otherwise specified, we will focus exclusively on regular MDPC codes.

Let $z = c + e \in \mathbb{F}_q^n$, where $c \in C$ and $\mathrm{wt}(e) \leq t$, be a noisy codeword. By $s = Hz^{\mathsf{T}} = He^{\mathsf{T}}$ we denote the syndrome of e. One can easily note that since i–th position of s is computed as

$$s_i = \langle h_i, e \rangle = \sum_{\omega \in \mathrm{supp}\,(h_i)} h_{i,\omega} e_\omega.$$

Hence, by selecting γ row indices $i_1, i_2, \ldots, i_\gamma$ for which $h_{i_1,j}, \ldots, h_{i_1,j}$ are non-zero, we obtain the following γ equalities:

$$\begin{cases} s_{i_1} h_{i_1,j}^{-1} = e_j + h_{i_1,j}^{-1} \left(\sum_{\omega \in \mathrm{supp}(h_{i_1}) \setminus \{j\}} h_{i_1,\omega} e_\omega \right), \\ \ldots \\ s_{i_\gamma} h_{i_\gamma,j}^{-1} = e_j + h_{i_\gamma,j}^{-1} \left(\sum_{\omega \in \mathrm{supp}(h_{i_\gamma}) \setminus \{j\}} h_{i_\gamma,\omega} e_\omega \right). \end{cases} \tag{3}$$

Since C is an MDPC code, the rows h_i of H are sparse. Considering sparsity of e, it follows that $s_i h_{i,j}^{-1}$ equals e_j with high probability. Hence it is possible to use the values $s_i h_{i,j}^{-1}$ for estimating e.

Let $\mathbb{F}_q = \{\alpha_0 = 0, \alpha_1 = 1, \alpha_2, \ldots, \alpha_{q-1}\}$ be a enumeration of elements of $\mathbb{F}_q$. Let us define

$$\sigma_{j,i} = \left| \left\{ w \mid h_{w,j} \neq 0 \text{ and } s_i h_{w,j}^{-1} = \alpha_i \right\} \right| \tag{4}$$

as the number of rows h_w containing the position j in $\mathrm{supp}(h_w)$ and $s_i h_{w,j}^{-1} = \alpha_i$. The values of $\sigma_{j,i}$ will be referred to as *error counters* in position j. Clearly, $\sigma_{j,i}$ indicates the likelihood that the error value e_j in position j is equal to α_i. In particular, a higher value of $\sigma_{j,0}$ implies that position j is less likely to be corrupted, while higher values of $\sigma_{j,i}$, $i \neq 0$, indicate a greater likelihood that $e_j = \alpha_i \neq 0$.

Therefore, several decoding strategies are possible. For instance, it is possible to choose an information set I of k positions with highest $\sigma_{j,0}$, indicating that these positions less likely to be erroneous, and then use this I for information set decoding *(ordered statistics decoding [21] and statistical decoding [37])*.

Another straightforward decoding algorithm that uses counters is as follows:

1. compute the syndrome s and the counters $\sigma_{j,i}$ for all $j \in [\![1, n]\!]$ and $i \in [\![0, q-1]\!]$;

2. update the position j of the received word z having the highest value of $\sigma_j^* - \sigma_{j,0}$, where

$$\sigma_j^* = \max_{i \in [\![1,q-1]\!]} \sigma_{j,i},\tag{5}$$

to the new value $z_j - \alpha_{i^*}$, where $i^* = \operatorname{argmax}_{i \in [\![1,q-1]\!]} \sigma_{j,i}$;
3. repeat from step 1 until either $s = 0$ or maximum number of iterations is reached.

Remark 1. One can easily note that the syndrome weight after step 2 is decreased by $\sigma_j^* - \sigma_{j,0}$. Therefore, the error position and error value in step 2 are chosen to decrease the syndrome weight the most. In this formulation the decoding approach described above was proposed in [9] as a generalization of Gallager's bit–flipping. In the binary case, the Gallagher's decoder is also a greedy algorithm that reduces the syndrome weight the most in each step .

1.1   One–Step Majority Logic Decoding

In this subsection, we study guaranteed decoding performance of regular MDPC codes under the OSML decoder (Algorithm 1) which can be considered as single iteration version of parallel symbol flipping.

Algorithm 1: OSML

Input: syndrome $s = He^\mathsf{T}$
Output: estimated error $\tilde{e}$
$\tilde{e} \leftarrow 0^n, \quad s \leftarrow Hz^\mathsf{T};$
for $j \leftarrow 1$ **to** n **do**
    using (4), (5) compute $\sigma_j = (\sigma_{j,0}, \ldots, \sigma_{j,q-1})$ and σ_j^* ;
    if $\sigma_j^* \geq th_j$ **then**
        $l \leftarrow \operatorname{argmax}_{i \in [\![1,q-1]\!]} \sigma_{j,i};$
        $\tilde{e}_j \leftarrow \tilde{e}_j + \alpha_l;$
end
return $\tilde{e}$

Remark 2. Note that in the decoder description, instead of recovering the corrected codeword $c \in C$ from the noisy codeword $z = c + e$ by iteratively subtracting the estimated error from z, we employ an equivalent formulation where we iteratively find the estimated error $\tilde{e}$ itself.

Let $X \in \mathbb{F}_q^{m \times n}$ be an $(m \times n)$–matrix, and let $I \subset [\![1,m]\!]$ and $J \subset [\![1,n]\!]$ be sets of row and column indices, respectively. We denote the matrix composed of the elements of X with indices $(i,j) \in I \times J$ as $X_{I,J} = (x_{i,j})_{i \in I, j \in J}$. For convenience, we use the notations $X_{:,J}$ and $X_{I,:}$ to represent $X_{[\![1,m]\!],J}$ and $X_{I,[\![1,n]\!]}$, respectively.

Proposition 1. *Let $H = (h_{i,j}) \in \mathbb{F}_q^{m \times n}$ be a parity–check matrix of a MDPC code, and let $e \in \mathbb{F}_q^n$ be an error of weight t. Define $H^{(j)}$ as the matrix consisting of rows from the set*

$$\left\{ h_{i,j}^{-1} \cdot \left(H_{i,\, [\![1,n]\!] \setminus \{j\}} \right) \mid i \in [\![1,m]\!], \ h_{i,j} \neq 0 \right\}.$$

Let

$$a_l = \text{wt}(H_{:,l}^{(j)}), \quad \mu(s) = \sum_{\substack{\omega \in \text{indicies of } s \text{ largest} \\ \text{values of } a_l}} a_\omega,$$

If $e_j = \alpha_i$, then $\sigma_{j,i}$ can be lower bounded as follows

$$\sigma_{j,i} \geq \begin{cases} \gamma - \mu(t), & e_j = \alpha_i = 0, \\ \gamma - \mu(t-1), & e_j = \alpha_i \neq 0. \end{cases}$$

Proof. Using (3), we obtain that $\sigma_{j,i}$ denotes the frequency of occurrence of α_i in the vector

$$v = \begin{pmatrix} s_{i_1} h_{i_1,j}^{-1} \\ \vdots \\ s_{i_\gamma} h_{i_\gamma,j}^{-1} \end{pmatrix} = \begin{pmatrix} e_j \\ \vdots \\ e_j \end{pmatrix} + \underbrace{H^{(j)} e'^{\mathsf{T}}}_{v'}, \quad e' = e_{[\![1,n]\!] \setminus \{j\}}.$$

Hence if $e_j = \alpha_i$ then $\sigma_{j,i} = \gamma - \text{wt}(v')$. Since v' is a linear combination of $\text{wt}(e')$ columns of $H^{(j)}$, its weight can be upper bounded by

$$\text{wt}(v') \leq \mu(\text{wt}(e')) = \begin{cases} \mu(t), & e_j = 0, \\ \mu(t-1), & e_j \neq 0. \end{cases}$$

This concludes the proof of the proposition.

Remark 3. Note that the weight $\text{wt}(H_{:,l}^{(j)})$ of l–th column $H_{:,l}^{(j)}$ of $H^{(j)}$ equals

$$\left| \text{supp}\left(H_{:,l}\right) \cap \text{supp}\left(H_{:,j}\right) \right|.$$

Corollary 1. *Let $\text{wt}(e) \leq t$. If $\mu(t) < th_j \leq \gamma - \mu(t-1)$, then the OSML decoder correctly estimates the j–th position of e.*

Proof. If $e_j = 0$, then $\sigma_{j,0} \geq \gamma - \mu(t)$ and hence $\sigma_j^* \leq \gamma - \sigma_{j,0} \leq \mu(t)$. It follows that setting $th_j > \mu(t)$ in Algorithm 1 will ensure that no non–erroneous position will be corrupted.

If $e_j = \alpha_i \neq 0$, then $\sigma_{j,i} \geq \gamma - \mu(t-1)$. Since $\mu(t) < \gamma - \mu(t-1)$ and $\mu(t) \geq \mu(t-1)$, it follows that $\mu(t-1) < \gamma/2$ and thereby $\sigma_{j,i} \geq \gamma - \mu(t-1) > \gamma/2$. This implies that a clear majority of equalities in (3) vote for α_i and hence $\sigma_j^* = \sigma_{j,i}$ (see (5)). Therefore, setting $th_j < \gamma - \mu(t-1)$ will ensure that error value in a erroneous position will be estimated correctly.

Corollary 2. *The guaranteed error–correction capability of OSML decoder is t if for all $j \in [\![1,n]\!]$ it is possible to choose th_j according to Corollary 1.*

Note that OSML is a very simple yet effective decoder that is capable of correcting low–weight error patterns. However, it is particularly useful as a second decoding iteration because it does not rely on probabilistic assumptions. It can effectively decode errors of a certain weight that remain after previous iterations, even if they have a harder–to–decode structure that would make plausibility analysis based on probabilistic assumptions irrelevant.

2   Plausibility analysis of 1–iteration parallel symbol flipping decoder

In this section, we provide an analysis of the single-iteration parallel symbol flipping algorithm. Namely, following the approach of [45], we estimate the probability of correcting an error using this decoder under several probabilistic assumptions. Furthermore, under the same assumptions, we estimate the probability of decreasing the error weight to a value that allows correction by the OSML decoder. This provides an upper bound on the decoding failure rate for the combination of a single iteration of parallel symbol flipping followed by the OSML decoder.

2.1   Distribution of counters

Below we give necessary results on probabilistic distributions of syndrome values and counters $\sigma_{j,i}$, $j \in [\![1, n]\!]$, $i \in [\![0, q-1]\!]$, required for further analysis of decoding iteration of proposed parallel symbol–flipping decoder. Our analysis will rely on several assumptions that are analogous to those used in [5, 45].

Assumption 1. *Let H be a parity–check of a random QC–MDPC code C in block–circulant form. It is assumed that each row of H is well modeled as a sample from uniform distribution over $\mathbb{F}_q^n$.*

Proposition 2. *Let $x \in \mathbb{F}_q^n$, $y \in \mathbb{F}_q^n$ be uniformly sampled. Let*

$$A_m = \Pr[\langle x, y \rangle \neq 0 \mid |\mathrm{supp}(x) \cap \mathrm{supp}(y)| = m].$$

Then A_m can be found recursively using

$$A_m = \begin{cases} (1 - A_{m-1}) + \frac{q-2}{q-1} A_{m-1}, & m \geq 1 \\ 0, & m = 0. \end{cases}$$

Proof. Without loss of generality, we assume that $\mathrm{supp}(x) \cap \mathrm{supp}(y) = \{1, \ldots, m\}$. It follows that

$$A_m = \Pr\left[\left(\sum_{i=1}^{m-1} x_i y_i = 0\right), \; x_m y_m \neq 0\right] + \Pr\left[\left(\sum_{i=1}^{m-1} x_i y_i \neq 0\right), x_m y_m \neq -\sum_{i=1}^{m-1} x_i y_i\right] =$$

$$= \Pr\left[\left(\sum_{i=1}^{m-1} x_i y_i = 0\right)\right] \cdot \Pr\left[x_m y_m \neq 0 \mid \left(\sum_{i=1}^{m-1} x_i y_i = 0\right)\right] +$$

$$+ \Pr\left[\left(\sum_{i=1}^{m-1} x_i y_i \neq 0\right)\right] \cdot \Pr\left[x_m y_m \neq -\alpha \mid \left(\sum_{i=1}^{m-1} x_i y_i = \alpha \neq 0\right)\right] =$$

$$= (1 - A_{m-1}) \cdot 1 + A_{m-1} \frac{q-2}{q-1}.$$

Theorem 1. *Let $H = (h_{i,j})$ be a parity–check matrix of (γ, δ)–regular QC-MDPC code C of length n. Let $e \in \mathbb{F}_q^n$ be a random error of weight t, and $s = eH^{\mathsf{T}}$ be its syndrome. Then for any row h_i of H, such that $j \in \mathrm{supp}(h_i)$*

$$\Pr[s_i h_{i,j}^{-1} = e_j \mid e_j \neq 0] = \sum_{i=0}^{\min(\delta-1,t-1)} \frac{\binom{\delta-1}{i}\binom{n-\delta}{t-i-1}}{\binom{n-1}{t-1}}(1 - A_i), \quad (6)$$

$$\Pr[s_i h_{i,j}^{-1} = e_j \mid e_j = 0] = \sum_{i=0}^{\min(\delta-1,t)} \frac{\binom{\delta-1}{i}\binom{n-\delta}{t-i}}{\binom{n-1}{t}}(1 - A_i), \quad (7)$$

$$\Pr[s_i h_{i,j}^{-1} = \alpha \neq e_j \mid e_j \neq 0] = (q-1)^{-1}\left(1 - \Pr[s_i h_{i,j}^{-1} = e_j \mid e_j \neq 0]\right), \quad (8)$$

$$\Pr[s_i h_{i,j}^{-1} = \alpha \neq 0 \mid e_j = 0] = (q-1)^{-1}\left(1 - \Pr[s_i h_{i,j}^{-1} = e_j \mid e_j = 0]\right). \quad (9)$$

Proof. Since $j \in \mathrm{supp}(h_i)$, Equation (3) implies that $s_i h_{i,j}^{-1} = e_j + h_{i,j}^{-1}\langle e', h'\rangle$, where

$$e' = e_{\llbracket 1,n \rrbracket \setminus \{j\}}, \quad h' = H_{i,\llbracket 1,n \rrbracket \setminus \{j\}}.$$

One can easily note that

$$\mathrm{wt}(e') = \begin{cases} t, & e_j = 0 \\ t - 1, & e_j \neq 0 \end{cases}, \quad \mathrm{wt}(h') = \delta - 1. \quad (10)$$

Since $s_i h_{i,j}^{-1} = e_j$ if and only if $\langle h', e'\rangle = 0$, it follows that

$$\Pr[s_i h_{i,j}^{-1} = e_j \mid e_j = \alpha] = \Pr[\langle e', h'\rangle = 0].$$

So, using Assumption 1, we obtain

$$\Pr[\langle e', h'\rangle = 0] = \sum_{i=0}^{\min(\mathrm{wt}(e'),\mathrm{wt}(h'))} \Pr[\langle e', h'\rangle = 0, \ |\mathrm{supp}(e') \cap \mathrm{supp}(h')| = i] =$$

$$= \sum_{i=0}^{\min(\mathrm{wt}(e'),\mathrm{wt}(h'))} (1 - A_i) \cdot \Pr[|\mathrm{supp}(e') \cap \mathrm{supp}(h')| = i] =$$

$$= \sum_{i=0}^{\min(\mathrm{wt}(e'),\mathrm{wt}(h'))} \frac{\binom{\mathrm{wt}(h')}{i}\binom{n-1-\mathrm{wt}(h')}{\mathrm{wt}(e')-i}}{\binom{n-1}{\mathrm{wt}(e')}}(1 - A_i).$$

Substituting (10) into this formula, we obtain (6) and (7). In addition, when $\langle e', h'\rangle \neq 0$, the product $\langle e', h'\rangle$ can assume any non–zero element of $\mathbb{F}_q$ with equal probabilities. Consequently, we obtain (8) and (9).

In the parallel symbol flipping decoder (see Algorithm 2), we propose the following flipping criterion based on counter values, using three decoding thresholds: th_0, th_E, and th_D. Namely, the position j of the received noisy codeword $z = c + e$ will be updated to $z_j - \alpha_i$ if the following conditions are satisfied:

1. $\sigma_{j,i} > \sigma_{j,\omega}$ for all $\omega \in [\![0, q-1]\!] \setminus \{i\}$, and thus $\sigma_j^* = \sigma_{j,i}$,
2. $\sigma_j^* \geq th_E$,
3. $\sigma_{j,0} < th_0$,
4. $\sigma_j^* - \sigma_{j,0} \geq th_D$.

Note that conditions 1–4 can be replaced by the single condition

$$\sigma_j = (\sigma_{j,0}, \ldots, \sigma_{j,q-1}) \in \Delta_{th_0, th_E, th_D}(i),$$

where $\Delta_{th_0, th_E, th_D}(i)$ is defined as follows

$$\Delta_{th_0, th_E, th_D}(i) = \Delta(i) = \left\{ (b_0, \ldots, b_{q-1}) \in \mathbb{Z}^q \mid \sum_{\omega=0}^{q-1} b_\omega = \gamma, \ b_i > \max_{\omega \neq i} b_z, \right.$$

$$\left. b_0 \leq th_0, \ b_i \geq th_E, \ b_i - b_0 \geq th_D \right\}.$$

In the following theorem, we will estimate the probability that the flipping criterion accurately determines the positions and values of errors.

Assumption 2. *We assume that the probability* $\Pr[\sigma_j \in \Delta(i)]$ *to flip position j to value $z_j - \alpha_i$ is a function only of error weight, i.e. it does not depend on error structure and the location j.*

Theorem 2. *Let H be a parity–check matrix of (γ, δ)–regular QC-MDPC code C of length n and dimension k. Let $e \in \mathbb{F}_q^n$ be a random error of weight t. Define*

$$p_1 = \Pr[s_i h_{i,j}^{-1} = e_j \mid e_j \neq 0], \quad p_2 = \Pr[s_i h_{i,j}^{-1} = \alpha \neq e_j \mid e_j \neq 0],$$

$$p_3 = \Pr[s_i h_{i,j}^{-1} = e_j \mid e_j = 0], \quad p_4 = \Pr[s_i h_{i,j}^{-1} = \alpha \neq e_j \mid e_j = 0].$$

Then the probability that non–zero error value will be estimated correctly is

$$p_{e \to c}(t) = \Pr[\sigma_j \in \Delta(i) \mid e_j = \alpha_i \neq 0] = \sum_{(b_0, \ldots, b_{q-1}) \in \Delta(i)} \frac{\gamma!}{b_0! \ldots, b_{q-1}!} p_1^{b_i} p_2^{\gamma - b_i}, \quad (11)$$

and the probability of incorrect estimate in non–erroneous position is

$$p_{c \to e}(t) = (q-1) \cdot \Pr[\sigma_j \in \Delta(i) \mid e_j = 0], \quad (12)$$

where

$$\Pr[\sigma_j \in \Delta(i) \mid e_j = 0] = \sum_{(b_0, \ldots, b_{q-1}) \in \Delta(i)} \frac{\gamma!}{b_0! \ldots, b_{q-1}!} p_3^{b_0} p_4^{\gamma - b_0}, \quad i \neq 0.$$

Proof. From Assumption 2 it follows that the probability

$$\Pr\left[\sigma_j = (b_0, \ldots, b_{q-1}) \mid e_j = \alpha_i \neq 0 \right]$$

can be modelled using multinomial distribution with parameters

$$\left(\Pr[s_i h_{i,j}^{-1} = 0 \mid e_j \neq 0], \ldots, \Pr[s_i h_{i,j}^{-1} = \alpha_{q-1} \mid e_j \neq 0]\right) = (\underbrace{p_2, \ldots, p_2}_{i-1}, p_1, \underbrace{p_2, \ldots, p_2}_{q-i}).$$

Hence

$$\Pr\left[\sigma_j = (b_0, \ldots, b_{q-1}) \mid e_j = \alpha_i \neq 0\right] = \frac{\gamma!}{b_0! \ldots, b_{q-1}!} p_1^{b_i} p_2^{\gamma - b_i},$$

which implies (11). By similar reasoning, we can also obtain (12).

2.2   Analysis of parallel symbol-flipping decoder

In this subsection, we employ results of previous subsection to give an plausibility analysis of the one–step parallel symbol flipping decoder (Algorithm 2) and its combination with OSML decoder (Algorithm 3).

Algorithm 2: 1–iteration parallel symbol flipping decoder

Input: syndrome $s = He^{\mathsf{T}}$
Output: estimated error $\tilde{e}$
$\tilde{e} \leftarrow 0^n \in \mathbb{F}_q^n$;
for $j \leftarrow 1$ **to** n **do**
$\quad$ using (4), (5) compute $\sigma_j = (\sigma_{j,0}, \ldots, \sigma_{j,q-1})$ and σ_j^*;
$\quad$ **if** $\sigma_j \in \Delta(s)$ **then**
$\quad\quad$ $\tilde{e}_j \leftarrow \tilde{e}_j + \alpha_s$
end
return $\tilde{e}$

Note that, after each iteration some error positions can be estimated correctly and some non–erroneous positions can be estimated to be erroneous incorrectly. In the following proposition, we provide an analysis of the probability that 1-iteration version of this decoder transforms a random error e of weight t into some new error e' of weight t'.

Proposition 3. *Let e be a random error of weight t, then after execution Algorithm 2*

1. *the probability to correctly estimate u error positions from e is*

$$P_{correct}(t, u) = \binom{t}{u} (p_{e \to c}(t))^u (1 - p_{e \to c}(t))^{t-u},$$

2. *the probability to corrupt v non–erroneous positions is*

$$P_{corrupt}(t, v) = \binom{n-t}{v} (p_{c \to e}(t))^v (1 - p_{c \to e})^{n-t-v},$$

3. *the probability to transform e into an error e' of weight t' is*

$$\Pr(t \to t') = \sum_{t-u+v=t'} P_{correct}(t, u) P_{corrupt}(t, v).$$

Proof. Assumption 2 implies that the flip decisions are statistically independent and depend solely on the error weight. It follows that $P_{correct}(t, u)$ and $P_{corrupt}(t, v)$ can be modeled as samples from binomial distributions with parameters $p_{e \to c}(t)$ and $p_{c \to e}(t)$ described in Theorem 2, respectively. The last claim trivially follows from the first two.

Corollary 3. *The decoding failure rate of 1-iteration parallel symbol-flipping decoder can be estimated as follows*

$$DFR_1 = 1 - \Pr(t \to 0).$$

Note that the new error e' is not random anymore and, therefore, the same analysis for further iteration is not possible. However, it is possible to decode e' using OSML decoder, which rely on no probabilistic assumptions.

Algorithm 3: PSF+OSML

Input: syndrome $s = He^\mathsf{T}$
Output: estimated error $\tilde{e}$
$\tilde{e} \leftarrow 0^n \in \mathbb{F}_q^n$;
for $j \leftarrow 1$ **to** n **do**
    using (4), (5) compute $\sigma_j = (\sigma_{j,0}, \ldots, \sigma_{j,q-1})$ and σ_j^*;
    if $\sigma_j \in \Delta(s)$ **then**
        $\tilde{e}_j \leftarrow \tilde{e}_j + \alpha_s$
end
$s \leftarrow He^\mathsf{T} - H\tilde{e}^\mathsf{T}$;
$\tilde{e} \leftarrow \tilde{e} + \text{OSML}(s)$;
return $\tilde{e}$

Thus, we obtain the following corollary:

Corollary 4. *Let e be a random error of weight t, let τ be the number of errors which can be corrected with certainty using OSML decoder. Then DFR of this combination is upper bounded by*

$$DFR_2 = 1 - \sum_{t'=0}^{\tau} \Pr(t \to t').$$

In Figures 2, 3, 4, we present the results of numerical simulations and compare them with the obtained theoretical estimates. Each experiment involved generating a random key and decoding a random error. For each error weight, the experiments were conducted until 100 decoding failures were detected or until 10^8 experiments were performed, whichever occurred first.

We observe that the theoretical estimates of DFR_1 and DFR_2 closely match the simulation results, substantiating the accuracy of the obtained theoretical model.

3 Choice of cryptosystem parameters

The choice of parameters of QC–MDPC cryptosystems is determined by the complexity of potential attacks on such cryptosystems. Specifically, the parameters of the
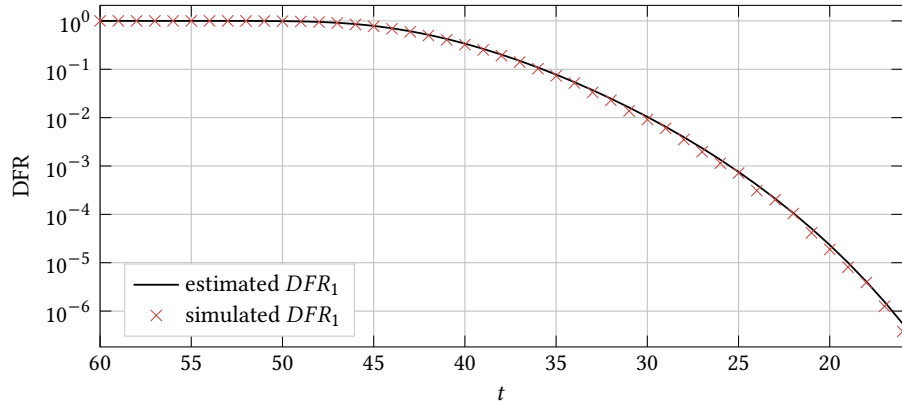
Fig. 2. Simulation results of DFR_1 for random QC-MDPC $[n = 2 \cdot 2339, k = 2339]_4$−codes over $\mathbb{F}_4$ ($l = 2, p = 2339, \gamma = 37$), with decoding thresholds $(th_0, th_E, th_D) = (18, 4, 4)$
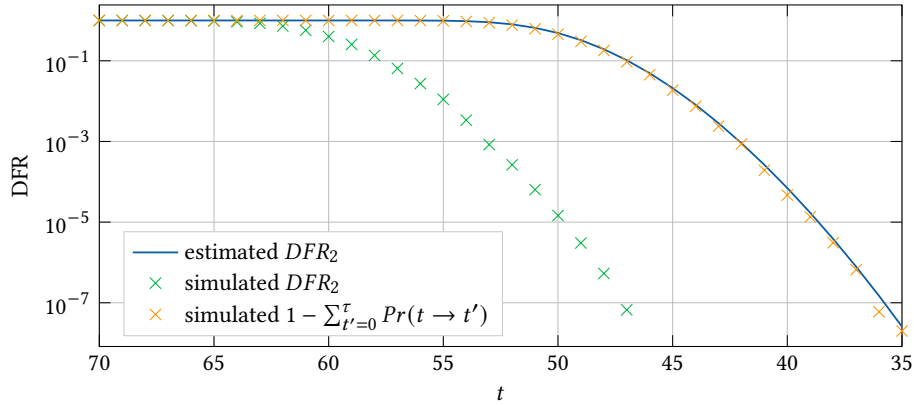

Fig. 3. Simulation results of DFR_2 for random QC-MDPC $[n = 2 \cdot 2339, k = 2339]_4$−codes over $\mathbb{F}_4$ ($l = 2, p = 2339, \gamma = 37, (th_0, th_E, th_D) = (18, 4, 4)$, and $\tau = 4$). For each experiment, we generated a random code and then checked if its OSML bound (see Corollary 2) is $\geq \tau$. If a code had a lower bound, it was rejected. We chose $\tau = 4$ to reject no more than 50% of keys (the actual rejection rate was 3%).

cryptosystem should be chosen in such a way that the best key-recovery attacks and message−recovery attacks require a sufficiently large number of operations.

The most effective message−recovery attacks are a family of information set decoding (ISD) algorithms, designed for decoding random codes. This family includes the Prange algorithm, the Lee-Brickell algorithm, Stern algorithm, BJMM, ball−collision, etc. An overview of ISD−algorithms can be found in [49]. The average complexity of these algorithms can be directly estimated using a formula that depends on parameters such as the field size q, code length n, code dimension k, and the weight of the error
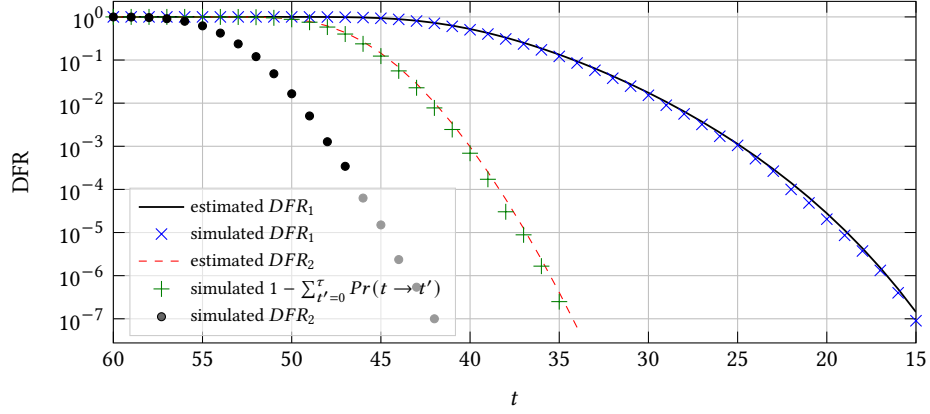
Fig. 4. Simulation results of DFR_1 and DFR_2 for random QC–MDPC $[n = 2 \cdot 1583, k = 1583]_8$–codes over $\mathbb{F}_8$ ($l = 2, p = 1583, \gamma = 37, (th_0, th_E, th_D) = (18, 4, 4)$, and $\tau = 4$)

w that needs to be found. For non–binary code direct complexity estimates for the Lee-Brickell and Stern algorithms can be found in [49], for BJMM in [25], and for ball-collision in [27].

It should be noted that for quasi-cyclic codes of order n', it has been shown [41] that the complexity of ISD attacks can be reduced by a factor of $\sqrt{n'}$ compared to codes without any structure. One of the features of QC-MDPC cryptosystems is that for key-recovery attacks, which involve finding low-weight dual codewords, the best attacks are also based on ISD. This is because the same algorithms can easily be adapted to search for codewords of a given weight instead of finding an error of a given weight. For quasi-cyclic codes, in this case, it is also possible to reduce the complexity by a factor of n' compared to random codes.

Furthermore, we must consider the decoding failure rate since in [26], Q. Guo et al. proposed a *reaction attack* that allows the recovery of secret keys in cryptosystems based on *binary QC-MDPC codes* by exploiting decoding failures. The original description assumes that $l = 2$, i.e., $n = 2n'$, but it can be easily generalized to other cases. This attack is based on the observation that certain error patterns are more easily decodable than other ones. Namely, let $\mathcal{E}_r$ be the set of error patterns of the following form:

$$\mathcal{E}_r = \{(e, \mathbf{0}) \in \mathbb{F}_2^{2p} \mid e \in \mathbb{F}_2^p, \exists \text{ distinct } s_1, s_2, \ldots, s_t, \text{ s.t. } e_{s_i} = 1 \text{ and}$$
$$s_{2i} = (s_{2i-1} + r) \mod n' \text{ for } i \in [\![1, t/2]\!]\}$$

Let $\mathbf{h_1} \in \mathbb{F}_q^{n'}$ denote the first row of H_1 (see (2)). Let $\psi(r)$ denote the number of pairs of non-zero positions of $\mathbf{h_1}$ placed at distance d. The distance between i and j is computed as $\min\{(i - j) \mod n', (j - i) \mod n'\}$. The set of values $\psi(i), i \in [\![1, \lfloor n'/2 \rfloor]\!]$, is called the *distance spectrum* of $\mathbf{h_1} \in \mathcal{R}_n$. In [26], it was shown that there is a correlation between the decoding failure rate on errors from $\mathcal{E}_r$ and the value of $\psi(r)$. Specifically, the larger $\psi(r)$ is, the lower the DFR for errors from $\mathcal{E}_r$.

Therefore, computing the DFR on errors from $\mathcal{E}_r$ for different r allows for the recovery of the distance spectrum of $\mathbf{h}_1$ and subsequently $\mathbf{h}_1$ itself. Consequently, it becomes possible to reconstruct the secret key of binary QC-MDPC cryptosystems by exploiting the decoding failures. Below, we demonstrate how this attack can be applied to the non-binary case as well.

In our experiments, we observed a correlation between the DFR for errors from $\tilde{\mathcal{E}}r$ and the values of $\psi(r)$, where the set $\tilde{\mathcal{E}}_r$ is defined as follows:

$$\tilde{\mathcal{E}}_r = \{(e, \mathbf{0}) \in \mathbb{F}_q^{2p} \mid e \in \mathbb{F}_2^p, \ \exists \text{ distinct } s_1, s_2, \ldots, s_t, \text{ s.t. } e_{s_i} \neq 0 \text{ and}$$
$$s_{2i} = (s_{2i-1} + r) \ \mathrm{mod} \ n' \text{ for } i \in [\![1, t/2]\!]\}$$

For instance, we conducted simulations to decode errors of weight $t = 84$ from $\tilde{\mathcal{E}}_r$ using Algorithm 4 for random QC-MDPC codes over $\mathbb{F}_4$ with parameters $n' = 2339$, $l = 2$, and $\gamma = 37$, which ensure a minimal cost of ISD-based key-recovery and message-recovery attacks of 2^{80} bit operations [9]. The results obtained from these simulations are presented in Table 1. As shown in the table, a strong dependency between the distance spectrum and the DFR for errors of this specific form can still be observed.

Table 1. Dependency between simulated DFR for random errors $e \in \tilde{\mathcal{E}}_r$ and the values $\psi(r)$. The results are averaged over 100 random QC–MDPC $[4678, 2339]$–codes.

$\psi(r)$	0	1	2	3	4
simulated DFR	0.0203	0.0134	0.0085	0.0059	0.0039

Algorithm 4: Sorted Parallel Symbol Flipping

Input: syndrome $s = He^\mathsf{T}$
Output: estimated error $\tilde{e}$
$\tilde{e} \leftarrow 0^n$;
for $it \leftarrow 1$ **to** 5 **do**
    using (4), (5) compute $\sigma_j = (\sigma_{j,0}, \ldots, \sigma_{j,q-1})$ and σ_j^* for all $j \in [\![1, n]\!]$,
    $i \in [\![0, q-1]\!]$;
    $th \leftarrow 20\text{th\_largest}(\sigma_j^* - \sigma_{j,0})$;
    for $j \leftarrow 1$ **to** n **do**
        if $\sigma_j - \sigma_{j,0} \geq \max(th, 1)$ **then**
            $i^* \leftarrow \mathrm{argmax}_{i \in [\![1, q-1]\!]} \sigma_{i,j}$;
            $\tilde{e}_j \leftarrow \tilde{e}_j + \alpha_{i*}$;
    end
    $s \leftarrow He^\mathsf{T} - H\tilde{e}^\mathsf{T}$;
    if $s = 0$ **then return** $\tilde{e}$;
end
return *fail*;

Thus, it is possible to reconstruct the support of the secret vector $\mathbf{h}_1$ (up to a cyclic shift) using the following steps:

1. for each $r \in [\![1, \lfloor n'/2 \rfloor]\!]$ numerically estimate DFR for random errors from $\tilde{\mathcal{E}}_r$, and then use the obtained results to recover the distance spectrum ψ of $\mathbf{h}_1$;
2. recover $\mathrm{supp}(h_1)$ using the procedure described in [26] for finding positions of ones in $\mathbf{h}_1$ for the binary case

Once $\mathrm{supp}(\mathbf{h}_1)$ is recovered, it is possible to recover the whole secret key $(\mathbf{h}_1, \mathbf{h}_2)$ in the non-binary case as follows. Let I be an information set such that

$$|I \cap \mathrm{supp}(\mathbf{h}_1 \mid \mathbf{h}_2)| = 1,$$

then the matrix $\tilde{H}_{:,I}^{-1}\tilde{H} = H_{:,I}^{-1}H$ contains the row $(\mathbf{h}_1, \mathbf{h}_2)$ or its quasi–circular shift. When $\mathrm{supp}(\mathbf{h}_1)$ is known, I can be constructed of one element from $\mathrm{supp}(\mathbf{h}_1)$, $n' - \gamma$ elements from $[\![1, n']\!] \setminus \mathrm{supp}(\mathbf{h}_1)$, and randomly guessed $\gamma - 1$ elements from $[\![n'+1, 2n']\!]$. Therefore, the probability of finding a suitable I can be estimated as follows:

$$\binom{n' - \gamma}{\gamma - 1} \cdot \binom{n'}{\gamma - 1}^{-1}.$$

So, the method described above in our experiments allowed reconstruction of secret key with significantly lower complexity than claimed security level of 2^{80} bit operations.

It follows that, when choosing the parameters of QC–MDPC cryptosystem that can be converted into IND–CCA2 secure KEM in non–binary case the design criteria are the complexity of ISD–based key–recovery, and message–recovery attacks and small enough decoding failure rate making reaction attacks infeasible. Table 2 provides potential parameters of QC-MDPC cryptosystems over $\mathbb{F}_4$, with $l = 2$ and $n' = p$ being a prime such that the polynomial $x^p - 1$ has a low number of irreducible factors. These parameters are given for three different security levels: $\lambda \in \{128, 192, 256\}$, which correspond to the complexity of breaking AES with the corresponding key sizes. All the proposed instances are designed to have $DFR_2 \leq 2^{-\lambda}$ (see Corollary 4). Note that the resulting public key sizes (pk_{size}) are slightly larger than in the binary

Table 2. Cryptosystem parameters

q	λ	p	γ	t	(th_0, th_E, th_D)	τ	pk_{size} (bits)
4	128 (2^{143} bit operations)	16 651	71	132	$(\gamma, 5, 5)$	9	33, 302
4	192 (2^{207} bit operations)	30 971	103	197	$(\gamma, 6, 6)$	12	61, 942
4	256 (2^{272} bit operations)	47 903	137	263	$(\gamma, 6, 6)$	16	95, 806

case $(28, 277, 52, 667, 83, 579$ respectively [5]). Moreover, increasing the field size to $q = 8$ with security level $\lambda = 128$ yields an estimated public key size of $36, 321$ bits ($p = 12, 107, \gamma = 69, t = 130$). Thus, for a fixed security level, public key size grows with increasing field size. Indeed, to maintain the same or smaller pk_{size} when increasing q, one must consider shorter MDPC codes. However, due to the complexity of ISD-based key–recovery and message–recovery attacks, γ and t are nearly the same across various ranges of q, implying higher-density codes. So, the increased field size does not appear to compensate for the negative impact of increased code density.

Conclusion

In this paper, we have studied the guaranteed error-correction capability of the one-step majority logic (OSML) decoder and provided a plausibility analysis of the 1-iteration parallel symbol flipping decoder for non-binary QC-MDPC codes. Through this analysis, we were able to estimate the decoding failure rate (DFR) of the combined use of these decoders, where parallel symbol flipping is employed to reduce the error weight to a level at which the OSML decoder can successfully correct any remaining errors. Consequently, we have obtained worst-case estimates of the DFR, considering some minimalistic and reasonable assumptions. The accuracy and validity of our theoretical model have been verified through numerical simulations.

Furthermore, we have demonstrated the importance of considering key-recovery reaction attacks when designing non-binary QC-MDPC cryptosystems. This implies that such cryptosystems need to be constructed with an extremely low DFR in order to achieve IND-CCA2 security with long-term keys. Finally, we have provided possible parameters for different NIST security levels of non-binary QC-MDPC cryptosystems, along with their theoretically estimated DFR.

It should be noted that the resulting key sizes are slightly larger than those in the binary case. Therefore, it appears that using non-binary QC-MDPC codes does not offer any benefits in terms of reducing the public-key sizes of IND-CCA2-secure cryptosystems considering the reaction attack. However, there is a possibility that replacing the quasi-cyclic structure with a more general (non-abelian) quasi-group structure, specifically replacing circulant matrices with matrices of multiplication operators in group algebras, could potentially hinder the reaction attack.

Additionally, by abandoning the requirement of key re-usage, it becomes possible to consider more sophisticated decoders for cryptosystems resistant against chosen plaintext attacks (CPA–secure). The study of such decoders can only be carried out through experimental methods and may provide benefits in terms of reducing key sizes, as previously explored in [9].

It is worth mentioning that the obtained in this paper theoretical models could potentially be useful for providing conservative estimates of the DFR of non-binary codes in telecommunications applications.

References

1. Alagic, G., Apon, D., Cooper, D., Dang, Q., Dang, T., Kelsey, J., Lichtinger, J., Miller, C., Moody, D., Peralta, R., et al.: Status report on the third round of the nist post-quantum cryptography standardization process. US Department of Commerce, NIST (2022). https://doi.org/10.6028/NIST.IR.8413

2. Apon, D., Perlner, R., Robinson, A., Santini, P.: Cryptanalysis of ledacrypt. In: Micciancio, D., Ristenpart, T. (eds.) Advances in Cryptology – CRYPTO 2020. pp. 389–418. Springer International Publishing, Cham (2020)

3. Aragon, N., Barreto, P.S., Bettaieb, S., Bidoux, L., Blazy, O., Deneuville, J.C., Gaborit, P., Gueron, S., Guneysu, T., Melchor, C.A., et al.: Bike: bit flipping key encapsulation, bikesuite.org

4. Arpin, S., Billingsley, T.R., Hast, D.R., Lau, J.B., Perlner, R., Robinson, A.: A Study of Error Floor Behavior in QC-MDPC Codes (2022). https://doi.org/10.1007/978-3-031-17234-2_5

5. Baldi, M., Barenghi, A., Chiaraluce, F., Pelosi, G., Santini, P.: Analysis of in-place randomized bit-flipping decoders for the design of LDPC and MDPC code-based cryptosystems. In: Obaidat, M.S., Ben-Othman, J. (eds.) E-Business and Telecommunications. pp. 151–174. Springer International Publishing, Cham (2021). https://doi.org/10.1007/978-3-030-90428-9_7

6. Baldi, M., Barenghi, A., Chiaraluce, F., Pelosi, G., Santini, P.: Performance bounds for QC-MDPC codes decoders. In: Wachter-Zeh, A., Bartz, H., Liva, G. (eds.) Code-Based Cryptography. pp. 95–122. Springer International Publishing, Cham (2022)

7. Baldi, M., Bianchi, M., Chiaraluce, F., Rosenthal, J., Schipani, D.: Enhanced public key security for the McEliece cryptosystem. Journal of Cryptology **29**, 1–27 (1 2016). https://doi.org/10.1007/s00145-014-9187-8

8. Baldi, M., Bodrato, M., Chiaraluce, F.: A new analysis of the McEliece cryptosystem based on QC-LDPC codes. In: Security and Cryptography for Networks: 6th International Conference, SCN 2008, Amalfi, Italy, September 10-12, 2008. Proceedings 6. pp. 246–262. Springer (2008)

9. Baldi, M., Cancellieri, G., Chiaraluce, F., Persichetti, E., Santini, P.: Using non-binary LDPC and MDPC codes in the McEliece cryptosystem. In: 2019 AEIT International Annual Conference (AEIT). pp. 1–6. IEEE (2019)

10. Baldi, M., Chiaraluce, F.: Cryptanalysis of a new instance of McEliece cryptosystem based on QC-LDPC codes. In: 2007 IEEE International Symposium on Information Theory. pp. 2591–2595. IEEE (2007)

11. Baldi, M., Chiaraluce, F., Garello, R., Mininni, F.: Quasi-cyclic low-density parity-check codes in the McEliece cryptosystem. In: 2007 IEEE International Conference on Communications. pp. 951–956. IEEE (2007)

12. Berger, T.P., Cayrel, P.L., Gaborit, P., Otmani, A.: Reducing key length of the McEliece cryptosystem (2009). https://doi.org/10.1007/978-3-642-02384-2_6

13. Berlekamp, E., McEliece, R., van Tilborg, H.: On the inherent intractability of certain coding problems (corresp.). IEEE Transactions on Information Theory **24**, 384–386 (5 1978). https://doi.org/10.1109/TIT.1978.1055873

14. Bernstein, D.J., Lange, T.: Post-quantum cryptography. Nature **549**, 188–194 (9 2017). https://doi.org/10.1038/nature23461

15. Borodin, M.A., Chizhov, I.V.: Effective attack on the McEliece cryptosystem based on Reed-Muller codes. Discrete Mathematics and Applications **24**(5), 273–280 (2014)

16. Couvreur, A., Lequesne, M., Tillich, J.P.: Recovering short secret keys of RLCE in polynomial time. In: Ding, J., Steinwandt, R. (eds.) Post-Quantum Cryptography. pp. 133–152. Springer International Publishing, Cham (2019)

17. Couvreur, A., Márquez-Corbella, I., Pellikaan, R.: Cryptanalysis of public-key cryptosystems that use subcodes of algebraic geometry codes. In: Coding theory and applications. pp. 133–140. Springer (2015)

18. Couvreur, A., Otmani, A., Tillich, J.P., Gauthier-Umaña, V.: A polynomial-time attack on the BBCRS scheme. In: Katz, J. (ed.) Public-Key Cryptography – PKC 2015. pp. 175–193. Springer Berlin Heidelberg, Berlin, Heidelberg (2015)

19. Deundyak, V.M., Kosolapov, Y.V., Maystrenko, I.A.: On the decipherment of Sidelnikov-type cryptosystems. In: Baldi, M., Persichetti, E., Santini, P. (eds.) Code-Based Cryptography. pp. 20–40. Springer International Publishing, Cham (2020)

20. Faugère, J.C., Otmani, A., Perret, L., Tillich, J.P.: Algebraic cryptanalysis of McEliece variants with compact keys (2010). https://doi.org/10.1007/978-3-642-13190-5_14

21. Fossorier, M.P., Lin, S.: Soft-decision decoding of linear block codes based on ordered statistics. IEEE Transactions on Information Theory **41**(5), 1379–1396 (1995)

22. Gabidulin, E.M., Paramonov, A., Tretjakov, O.: Ideals over a non-commutative ring and their application in cryptology. In: Workshop on the Theory and Application of of Cryptographic Techniques. pp. 482–489. Springer (1991)

23. Gaborit, P.: Shorter keys for code based cryptography. In: Proceedings of the 2005 International Workshop on Coding and Cryptography (WCC 2005). pp. 81–91 (2005)
24. Gallager, R.: Low-density parity-check codes. IRE Transactions on information theory **8**(1), 21–28 (1962)
25. Gueye, C.T., Klamti, J.B., Hirose, S.: Generalization of bjmm-isd using may-ozerov nearest neighbor algorithm over an arbitrary finite field. In: Codes, Cryptology and Information Security: Second International Conference, C2SI 2017, Rabat, Morocco, April 10–12, 2017, Proceedings-In Honor of Claude Carlet. pp. 96–109. Springer (2017)
26. Guo, Q., Johansson, T., Wagner, P.S.: A key recovery reaction attack on QC-MDPC. IEEE Transactions on Information Theory **65**, 1845–1861 (3 2019). https://doi.org/10.1109/TIT.2018.2877458
27. Interlando, C., Khathuria, K., Rohrer, N., Rosenthal, J., Weger, V.: Generalization of the ball-collision algorithm. arXiv preprint arXiv:1812.10955 (2018)
28. Ivanov, F., Krouk, E., Zyablov, V.: New code-based cryptosystem based on binary image of generalized Reed-Solomon code. In: 2021 XVII International Symposium" Problems of Redundancy in Information and Control Systems"(REDUNDANCY). pp. 66–69. IEEE (2021)
29. Janwa, H., Moreno, O.: McEliece public key cryptosystems using algebraic-geometric codes. Designs, Codes and Cryptography **8**(3), 293–307 (1996)
30. Kosolapov, Y., Lelyuk, A.: Cryptanalysis of the BBCRS system on Reed-Muller binary codes. Bulletin of the South Ural State University. Series "Mathematical Modelling, Programming and Computer Software" **14**, 18–32 (2021). https://doi.org/10.14529/mmp210302
31. McEliece, R.J.: Public-key cryptosystem based on algebraic coding theory. PL Deep Space Network Progress Report **42**, 114–116 (1978)
32. Minder, L., Shokrollahi, A.: Cryptanalysis of the sidelnikov cryptosystem. In: Annual International Conference on the Theory and Applications of Cryptographic Techniques. pp. 347–360. Springer (2007)
33. Misoczki, R., Tillich, J.P., Sendrier, N., Barreto, P.S.L.M.: MDPC-McEliece: New McEliece variants from moderate density parity-check codes. pp. 2069–2073. IEEE (7 2013). https://doi.org/10.1109/ISIT.2013.6620590
34. Monico, C., Rosenthal, J., Shokrollahi, A.: Using low density parity check codes in the McEliece cryptosystem. p. 215. IEEE (2000). https://doi.org/10.1109/ISIT.2000.866513
35. Niederreiter, H.: Knapsack-type cryptosystems and algebraic coding theory. Problems of Control and Information Theory **15**, 159–166 (1986)
36. Otmani, A., Tillich, J.P., Dallot, L.: Cryptanalysis of two McEliece cryptosystems based on quasi-cyclic codes. Mathematics in Computer Science **3**, 129–140 (4 2010). https://doi.org/10.1007/s11786-009-0015-8
37. Overbeck, R.: Statistical decoding revisited. In: Information Security and Privacy: 11th Australasian Conference, ACISP 2006, Melbourne, Australia, July 3-5, 2006. Proceedings 11. pp. 283–294. Springer (2006)
38. Overbeck, R.: Structural attacks for public key cryptosystems based on Gabidulin codes. Journal of cryptology **21**(2), 280–301 (2008)
39. Santini, P., Battaglioni, M., Baldi, M., Chiaraluce, F.: Analysis of the Error Correction Capability of LDPC and MDPC Codes Under Parallel Bit-Flipping Decoding and Application to Cryptography. IEEE Transactions on Communications **68**, 4648–4660 (8 2020). https://doi.org/10.1109/TCOMM.2020.2987898
40. Sendrier, N.: On the structure of randomly permuted concatenated code. Ph.D. thesis, INRIA (1995)
41. Sendrier, N.: Decoding one out of many. In: Post-Quantum Cryptography: 4th International Workshop, PQCrypto 2011, Taipei, Taiwan, November 29–December 2, 2011. Proceedings 4. pp. 51–67. Springer (2011)

42. Shor, P.W.: Algorithms for quantum computation: discrete logarithms and factoring. In: Proceedings 35th annual symposium on foundations of computer science. pp. 124–134. Ieee (1994)
43. Sidelnikov, V.M.: A public-key cryptosystem based on binary Reed-Muller codes. Discrete Mathematics and Applications **4**(3), 191–208 (1994)
44. Sidelnikov, V.M., Shestakov, S.O.: On insecurity of cryptosystems based on generalized Reed-Solomon codes. Discrete Mathematics and Applications **2** (1992)
45. Tillich, J.P.: The decoding failure probability of MDPC codes. pp. 941–945. IEEE (6 2018). https://doi.org/10.1109/ISIT.2018.8437843
46. Vasseur, V.: Post-quantum cryptography: a study of the decoding of QC-MDPC codes. Ph.D. thesis, Université de Paris (2021)
47. Vedenev, K., Kosolapov, Y.: Cryptanalysis of Ivanov–Krouk–Zyablov Cryptosystem. In: Deneuville, J.C. (ed.) Code-Based Cryptography. pp. 137–153. Springer Nature Switzerland, Cham (2023)
48. Wang, Y.: Quantum resistant random linear code based public key encryption scheme RLCE. In: 2016 IEEE International Symposium on Information Theory (ISIT). pp. 2519–2523. IEEE (2016)
49. Weger, V.: Information Set Decoding in the Lee Metric and the Local to Global Principle for Densities. Ph.D. thesis, PhD thesis, University of Zurich (2020)
50. Wieschebrink, C.: Cryptanalysis of the niederreiter public key scheme based on GRS subcodes. In: Sendrier, N. (ed.) Post-Quantum Cryptography. pp. 61–72. Springer Berlin Heidelberg, Berlin, Heidelberg (2010)