Computers, Materials & Continua

Tech Science Press

Doi:10.32604/cmc.2025.065296

ARTICLE

An Optimization of Weak Key Attacks Based on the BGF Decoding Algorithm

Bing Liu*, Ting Nie, Yansong Liu and Weibo Hu

Department of Cryptography Science and Technology, Beijing Electronic Science and Technology Institute, Beijing, 100070, China

*Corresponding Author: Bing Liu. Email: bing@besti.edu.com

Received: 09 March 2025; Accepted: 23 May 2025; Published: 30 July 2025

ABSTRACT: Among the four candidate algorithms in the fourth round of NIST standardization, the BIKE (Bit Flipping Key Encapsulation) scheme has a small key size and high efficiency, showing good prospects for application. However, the BIKE scheme based on QC-MDPC (Quasi Cyclic Medium Density Parity Check) codes still faces challenges such as the GJS attack and weak key attacks targeting the decoding failure rate (DFR). This paper analyzes the BGF decoding algorithm of the BIKE scheme, revealing two deep factors that lead to DFR, and proposes a weak key optimization attack method for the BGF decoding algorithm based on these two factors. The proposed method constructs a new weak key set, and experiment results eventually indicate that, considering BIKE's parameter set targeting 128-bit security, the average decryption failure rate is lowerly bounded by $\mathrm{DFR}_{AVG} \geq 2^{-103.83}$. This result not only highlights a significant vulnerability in the BIKE scheme but also provides valuable insights for future improvements in its design. By addressing these weaknesses, the robustness of QC-MDPC code-based cryptographic systems can be enhanced, paving the way for more secure post-quantum cryptographic solutions.

KEYWORDS: BIKE; BGF decoding algorithm; weak key attack; GJS attack

1 Introduction

With the rise of quantum computers, current public key encryption (PKE) schemes face unprecedented threats [1]. Traditional public key encryption methods, such as RSA and ECC, rely on the difficulty of integer factorization and discrete logarithm problems. However, quantum computers, by using the Shor algorithm [2], can solve these problems efficiently in polynomial time, thus invalidating traditional encryption methods. To address this challenge, the National Institute of Standards and Technology (NIST) is advancing a standardization project for Post-Quantum Cryptography (PQC), which aims to evaluate and finalize encryption schemes that can withstand quantum attacks [3]. In this context, a scheme called BIKE (Bit Flipping Key Encapsulation) has attracted a lot of attention and has entered the final stage of the NIST PQC standardization competition.

BIKE is a representative QC-MDPC (Quasi Cyclic Medium Density Parity Check) code-based scheme, which is relatively competitive both in terms of code length and efficiency and communication bandwidth, and its security relies on the difficult problem of proposed cyclic codes [4]. However, the DFR problem exists even though BIKE employs a state-of-the-art Black-Gray-Flip (BGF) decoder [5] to reduce the Decoding Failure Rate (DFR) and to improve the decoding efficiency. The security of BIKE's IND-CCA relies on the average Decoding Failure Rate (DFR). The current analysis only gives an estimate of the DFR and does not give a proven upper bound [6]. Therefore, the BIKE instantiation using the BGF decoder does not formally declare IND-CCA security. Currently, attacks such as GJS [7], weak keys [8], and side channels [9] still exist

against the decoding failure probability of BIKE schemes. For example, the amplification principle of Guo et al. [10] as well as Nilsson et al. [11] introduce a GJS reaction attack based on the QC-MDPC code structure, which utilizes the existence of a DFR in the decoding so that the attacker can fully recover the key. Drucker et al. [12] argued that the existence of a weak key affects the DFR and that quantitative proofs of the IND-CCA security of BIKE are needed. Wang et al. [8] proved that the existence of weak keys and quantization of the effect on the DFR of the decoder pose threats to the IND-CCA security of BIKE.

Although BIKE shows great potentiality in the field of post-quantum cryptography, its challenges in terms of DFR and other security aspects suggest that further research and improvements are necessary. To ensure that the final chosen post-quantum cryptography standard provides sufficient security and efficiency, this paper addresses the BIKE algorithm, analyzes the performance of its BGF decoder to reveal the factors that affect the DFR, and proposes a weak-key attack optimization scheme to evaluate the security of the BIKE scheme by the DFR test of the decoder.

2 Preliminaries

2.1 BIKE

BIKE (Bit Flipping Key Encapsulation) is a post-quantum cryptography-based key encapsulation mechanism that uses quasi-cyclic moderate-density parity-check (QC-MDPC) codes and the Niederreiter cryptosystem framework [13]. It features small key sizes, efficient algorithms, and low complexity [14]. Compared to the traditional McEliece cryptosystem, BIKE has smaller communication bandwidth requirements, making it suitable for bandwidth-constrained network environments. Its design is simple with minimal resource usage, making it suitable for both software implementations (such as servers and PCs) and hardware implementations (such as IoT devices and embedded systems) [15]. It is an efficient, flexible, and quantum-resistant cryptographic solution with broad application prospects in the field of post-quantum cryptography [16]. The algorithm of BIKE KEM is divided into three main steps: key generation, key encapsulation, and key decapsulation. The key encapsulation mechanism (KEM) is described as Algorithm 1.

Algorithm 1: The BIKE key encapsulation mechanism

KeyGen: $() \mapsto (h_0, h_1, \sigma), h$
Output: $(h_0, h_1, \sigma) \in \mathcal{H}_w \times \mathcal{M}, h \in \mathcal{R}$
1: $(h_0, h_1) \xleftarrow{\mathcal{D}} H_\omega$
2: $h \leftarrow h_1 h_0^{-1}$
3: $\sigma \xleftarrow{\$} \mathcal{M}$

Encaps: $h \mapsto K, c$
Input: $h \in \mathcal{R}$
Output: $K \in \mathcal{K}, c \in \mathcal{R} \times \mathcal{M}$
1: $m \xleftarrow{\$} \mathcal{M}$
2: $(e_0, e_1) \leftarrow \mathbf{H}(m)$
3: $c \leftarrow (e_0 + e_1 h, m \oplus \mathbf{L}(e_0, e_1))$
4: $K \leftarrow \mathbf{K}(m, c)$

Decaps: $(h_0, h_1, \sigma), c \mapsto K$
Input: $((h_0, h_1), \sigma) \in \mathcal{H}_w \times \mathcal{M}, c = (c_0, c_1) \in \mathcal{R} \times \mathcal{M}$
Output: $K \in \mathcal{K}$
1: $e' \leftarrow \text{decoder}(c_0 h_0, h_0, h_1)$
2: $m' \leftarrow c_1 \oplus \mathbf{L}(e')$
3: if $e' = \mathbf{H}(m')$ then $K \leftarrow \mathbf{K}(m', c)$ else $K \leftarrow \mathbf{K}(\sigma, c)$

To have λ bits of IND-CCA security, the parameters are r, ω, t, and decoder chosen in the setup such that:

1. $\mathrm{QCCF}_{r,w}$ offers λ bits of security
2. $\mathrm{QCSD}_{r,t}$ offers λ bits of security
3. $|\mathcal{M}| = 2^{\ell} \geq 2^{\lambda}$
4. DFR (decoder) $\leq 2^{-\lambda}$

The NIST proposal identifies several security categories related to the strength of key search attacks on grouped ciphers (in the case of AES). The target security levels for BIKE are 1, 3, and 5, which correspond to the security levels of AES-128, AES-192, and AES-256, respectively. The BIKE parameters corresponding to different security levels are shown in Table 1:

Table 1: BIKE parameters

Security	λ	r^{CPA}	r^{CCA}	ω	t	DFR
Level 1	128	10163	12323	142	134	2^{-128}
Level 3	192	19853	24659	206	199	2^{-192}
Level 5	256	32749	40973	274	264	2^{-256}

2.2 BGF Decoding Algorithm

The BGF decoding algorithm (Black-Gray-Flip) is an improved version of Gallager's bit-flip decoding method [17], addressing the issue of high decoding failure rates in MDPC codes. It employs two predefined thresholds to divide bits into two groups: the black set and the gray set. In the first step, bits with the highest number of unsatisfied parity checks are categorized as black bits and are flipped. Bits with fewer unsatisfied checks are considered gray bits and remain unchanged. In the subsequent step, if the number of unsatisfied checks on a bit exceeds a second threshold, black bits that breach this threshold are flipped back to their original state, and gray bits are flipped. After each operation, the algorithm updates the checksum and the count of unsatisfied checks, continuously adjusting to correct any incorrectly flipped bits. As a result, the BGF decoder focuses on fewer positions and has a higher concentration of errors compared to the traditional bit-flip decoder, improving flip accuracy and significantly reducing the likelihood of decoding failure. The specific steps are shown in Algorithms 2 and 3.

Algorithm 2: The black-gray-flip algorithm

Input: parity-check matrix $H \in F_2^{r \times n}$, $y \in F_2^n$, $\tau \in N$
Output: errors vector $y \in F_2^n$
a): $d \leftarrow ColumnWeight\,(H)$
b): $maskTH \leftarrow (d+1)/2$
c): $for\ each\ iter = 1 \ldots MaxIter\ do$
d): $\quad syndrome \leftarrow Hy^T$
e): $\quad upc \leftarrow ComputeUPC(H, sydrome)$
f): $\quad th \leftarrow threshold(context)$
g): $\quad black \leftarrow gray \leftarrow 0^n$
h): $\quad for\ each\ i = 1 \ldots n\ do$

(Continued)

Algorithm 2 (continued)

i): $if\ upc[i] \geq th\ then$
j): $y[i] = 1 - y[i]$
k): $black[i] \leftarrow 1$
l): $else\ if\ upc[i] \geq th - \tau\ then$
m): $gray[i] \leftarrow 1$
n): $end\ if$
o): $end\ for$
p): $if\ iter = 1\ then$
q): $y \leftarrow MaskedBitFlip(y, H, black, maskTh)$
r): $y \leftarrow MaskedBitFlip(y, H, gray, maskTh)$
s): $end\ if$
t): $end\ for$
u): $return\ y$

Algorithm 3: The masked bit flip function algorithm

a): *MaskedBitFlip (y, H, mask, th)*
b): $syndrome \leftarrow Hy^T$
c): $upc \leftarrow ComputeUPC(H, sydrome)$
d): $for\ each\ i = 1 \ldots n\ do$
e): $if\ mask[i] = 1\ and\ upc[i] \geq th\ then$
f): $y[i] = 1 - y[i]$
g): $end\ if$
h): $end\ for$
i): $return\ y$

The reasonableness of the threshold selection can greatly affect the efficiency of the decoding. The threshold of the BGF decoder is defined as an affine function of the weight of the checker [18], but when the weight of the checker is between the average values, the threshold formula is also close to the affine function. Therefore, the lower limit of the threshold is given as:

$$threshold(|s'|) = \max((d+1)/2, \lfloor a|s'| + b \rfloor) \tag{1}$$

The BIKE 128-bit security parameter is used for the test as $(d, t) = (71, 134)$, the value required to achieve the target DFR for a given number of different iterations. The threshold parameter $a = 0.0069722$, $b = 13.530$. The code length of the BGF decoder, and the DFR relationship are shown in Fig. 1.

To accurately estimate the decoding failure probability DFR of this decoding algorithm, the extrapolation method Markovian model [19] is used, which is based on the fact that the DFR curve is a concave function, and the block value of the large parameter can be deduced by testing the DFR under the smaller code length parameter, given some of the parameters, i.e., $r \mapsto \log DFR_{\mathrm{D}}(r)$. For the decoder, security level λ, if $DFR_{\mathrm{D}}(r) \geq 2^{-\lambda}, r_1 < r_2 < r_3$, the following inequality can be obtained by defining the concavity of the Formula (2).

$$\log \mathrm{DFR}_{\mathcal{D}}(r_3) \leq \log \mathrm{DFR}_{\mathcal{D}}(r_1) + \frac{\log \mathrm{DFR}_{\mathcal{D}}(r_2) - \log \mathrm{DFR}_{\mathcal{D}}(r_1)}{r_2 - r_1} \cdot r_3 - r_1 \tag{2}$$

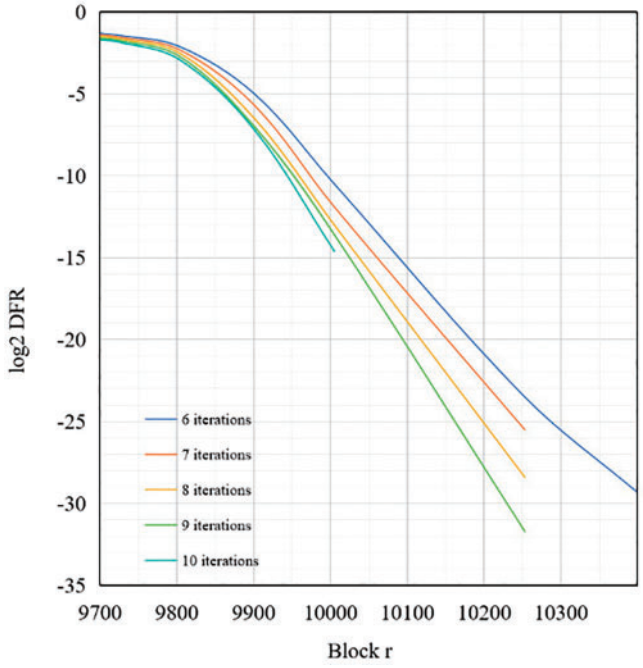

Figure 1: The relationship graph between Block length r and DFR

To accurately assess p_3, introduce the Bernoulli experimental function in statistics, assuming that p_1 and p_2 is the failure probability of two independent Bernoulli tests, in the total number of Bernoulli tests N_i in the sample to test failure situation F_i , $i = 1, 2$ and $p_3 = p_1^{-A} p_2^{1+A}$, the derivation of the formula is:

$$\frac{1}{K}\int_{-\infty}^{tp_3^-}\int_{-\infty}^{0} e^{s\left(\frac{F_2+1}{1+A}\right)} e^{t\left(\frac{F_1+1}{A}+\frac{F_2+1}{1+A}\right)}\left(1-e^{\frac{t}{A}}\right)^{N_1-F_1}\left(1-e^{\frac{s+t}{1+A}}\right)^{N_2-F_2} dtds < \frac{\alpha}{2} \tag{3}$$

$$\frac{1}{K}\int_{\ell p_3^+}^{\infty}\int_{-\infty}^{0} e^{s\left(\frac{F_2+1}{1+A}\right)} e^{t\left(\frac{F_1+1}{A}+\frac{F_2+1}{1+A}\right)}\left(1-e^{\frac{t}{A}}\right)^{N_1-F_1}\left(1-e^{\frac{s+t}{1+A}}\right)^{N_2-F_2} dtds < \frac{\alpha}{2} \tag{4}$$

$$K = A(1+A)\,B(F_1+1, N_1-F_1+1)\,B(F_2+1, N_2-F_2+1)\,. \tag{5}$$

Therefore, two methods, analog extrapolation and formula method, are used to test the decoding DFR lower bounds respectively. To ensure the security of IND-CCA and find the minimum block size, it is necessary to ensure that the selected parameter DFR is less than $2^{-\lambda}$. Where the parameters are chosen as $(r_1, r_2, r_3) = (9803, 9901, 12323)$, $(d, t) = (71, 134)$, and the confidence interval α is 0.01. The average DFR test results for the BGF decoding algorithm under the conditions of $\tau = 3$, IND-CCA security parameters $\lambda = 128$, $iterations = 7$, are shown in Table 2.

To more accurately evaluate the accuracy of these methods and their potential limitations when applied to practical BIKE parameters, this section provides a detailed analysis of their precision and margin of error. As specifically demonstrated in Table 2.

From Table 2, it can be seen that the BGF decoding algorithm can achieve a corresponding r^{CCA}, which achieves better decoding efficiency than other variants of decoding algorithms [20] by exhibiting a lower DFR in constant time. However, the drawbacks are equally obvious, its efficient and accurate decoding sacrifices universality, and its error correction capability is limited in the face of longer code lengths or highly weighted code words, especially at high security levels.

Table 2: BGF decoding analogue data table

$\lambda = 128$	DFR(95%CI)			
	$\log_2 DFR$	$\log_2(DFR^+)$ / $\log_2(DFR^-)$	Margin of error	Recision
Extrapolation method	−146.20	−129.31	11.55%	±14.70%
		−172.32	17.86%	
Formula method	−147.26	−130.31	11.51%	±11.51%
		−164.21	11.51%	

3 DFR Analysis

Decoders for bit-flipping of QC-MDPC codes used by BIKE always have a non-negligible DFR. Although the BGF decoders currently used have a very small DFR, it does not formally declare the security of IND-CCA. This section analyzes the reasons for the existence of DFR and analyzes the special structures that may hinder the decoding to reveal the reasons that affect the decoding.

3.1 Distance Spectrum Analysis

In the GJS reaction attack, Guo [7] found that there is a strong correlation between the DFR and the distance spectrum of the key, and a large amount of information about the distance spectrum of the key can be collected from the decoding failure, and then the key can be recovered from the distance spectrum by the key recovery algorithm. Therefore, the distance spectrum of this attack is the key to whether the key can be recovered or not.

For $h \in F_2[x]/(x^r-1)$, The number of occurrences of a distance can be called multiplicity, and the set of multiplicities is the distance spectrum, which can be expressed as [21]:

$$Spectrum(h) = \{(\delta, \mu(\delta, h)) \mid \delta \in \{0, 1, \ldots, \lfloor r/2 \rfloor\}\} \tag{6}$$

Assuming that the multiplicities of the distance spectra are independent, then for $0 \le m \le d, 1 \le \delta \le \lfloor r/2 \rfloor$ and $h \in F_2[x]/(x^r-1)$ then the multiplicity probability distribution formula is:

$$\pi_m = \Pr[\mu(\delta, \mathrm{h}) = m] = \frac{\mathcal{N}_m}{\binom{r}{d}} \tag{7}$$

$$p_{\ge m} = \Pr\left[\max_{\delta \in \{1,\ldots,\lfloor r/2 \rfloor\}} \mu(\delta, \mathbf{h}) \ge m\right] = 1 - (1-\pi_m)^{\lfloor r/2 \rfloor} \tag{8}$$

$$p_{=m} = \Pr\left[\max_{\delta \in \{1,\ldots,\lfloor r/2 \rfloor\}} \mu(\delta, \mathrm{h}) = m\right] = p_{\ge m} - p_{\ge m+1} \tag{9}$$

$$\mathcal{N}_m = \frac{r}{d-m}\binom{d-1}{d-m-1}\binom{r-d-1}{d-m-1} \tag{10}$$

For a subsequent assessment of the effect of multiplicity on the DFR, the distribution of the parameters at three different security levels is plotted as Fig. 2. It can be observed that the probability of a specific multiplicity in the spectrum of a cyclic block is generally low (non-zero), with only a small percentage higher.

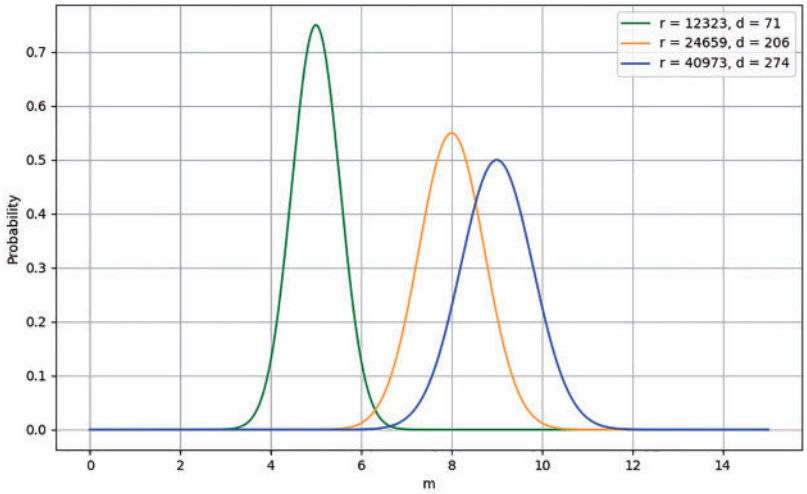

Figure 2: Probability analysis of multiplicity m

Therefore, this explains the strong correlation between the DFR and the distance between non-zero bits in the private key vector (the first row of the cyclic private key matrix) in the GJS attack. That is, if there is a distance between two non-zero bits in the same error pattern, then the probability of decoding failure is much smaller compared to the absence of such a distance.

Such a result is obvious, the result in the cyclic condition of the QC code, using the error pattern between the two 1's of the distance and the private key vector of the same distance. Its checksum value decreases due to multiplicative cancellation when identical distances are encountered. The cancellation frequency is inversely proportional to the resultant checksum weight, with higher cancellation counts yielding lower weight values. Therefore, the probability of decoding error when the weight is 0 is greater than the probability of error when the weight is non-zero, and the corresponding DFR will be larger.

3.2 Checksum Analysis

In coding theory, the checker is an important factor affecting decoding. For bit-flip decoding, the number of iterations completes the decoding within 3–5 iterations on average, and further iterations have almost no effect on the decoding probability [22]. Therefore, the first correct flip greatly affects whether the decoding can be successful or not. In the first round of decoding, the number of errors ℓ in each checksum equation determines the number of correctly changed counters. For example, if the number of errors in a checksum equation is even, then all counters will be correctly changed; if it is odd, it means that there is at least one incorrect bit, in which case all counters will be incremented, but only one bit is incorrect. Thus, all but one counter will change correctly and the rest will increase incorrectly.

To more accurately analyze the effect of the number of errors in each checksum equation on the decoding, it is assumed that the number of errors involved in the equations obey the following distribution for any $i \in \{0, \ldots, r-1\}, a \in \{0,1\}, \ell \in \{0, \ldots, \min(\omega, t)\}$, assuming that the number of errors involved in the equations H is chosen uniformly at random from $\mathcal{H}_{d,\omega,r\times n}$, and e uniformly at random from $\mathcal{E}_{n,t}$ [23]:

$$\Pr\left[\rho_i = \ell \mid s_i\right] = \frac{g_{s_i}(\ell)}{\mathcal{N}_{g_{s_i}}} \tag{11}$$

$$g_a(\ell) = \frac{\binom{w}{\ell}\binom{n-w}{t-\ell}}{\binom{n}{t}} \mathbf{1}_{a+2\mathbb{Z}}(\ell) \tag{12}$$

$$\mathcal{N}_{g_1} = \sum_{\ell \text{ is odd}} \frac{\binom{w}{\ell}\binom{n-w}{t-\ell}}{\binom{n}{t}}, \ \mathcal{N}_{g_0} = 1 - \mathcal{N}_{g_1}. \tag{13}$$

where $s_i \in \{0,1\}$, ℓ is the number of errors. When $s_i = 0$, the number of errors involved in the equation is even, the distribution ρ_i is an odd multiple of the number of errors ℓ, and when $s_i = 0$, the distribution is an even multiple of the number of errors.

Testing the parameters for selecting 128-bit security, it can be observed from the Table 3 that the probability decreases with an increase of ℓ and the average probability is generally lower for even multiples of the number of errors than for odd multiples. This is because when the number of errors is even, the number of counters correctly changed is $w - \ell$, and the number of errors changed is ℓ. On the contrary, when the number of errors is odd, the number of counters correctly changed is ℓ, and the number of errors changed is $w - \ell$. Thus, having an even number of counters helps in decoding, while having an odd number of counters ℓ has a negative effect on decoding.

Table 3: Probability analysis of ℓ

ℓ	$\log_2 \Pr[\rho_i = \ell \mid s_i = 0]$	ℓ	$\log_2 \Pr[\rho_i = \ell \mid s_i = 0]$
0	−0.39	1	−0.14
2	−2.13	3	−3.50
4	−6.53	5	−8.68
6	−12.35	7	−15.03
8	−19.15	9	−22.23
10	−26.72	11	−30.14
12	−34.94	13	−38.65
14	−43.71	15	−47.67
16	−52.98	17	−57.16
18	−62.68	19	−67.08
20	−72.80	21	−77.39
22	−83.30	23	−88.07
24	−94.15	25	−99.09
26	−105.33	27	−110.45
28	−116.85	29	−122.11
30	−128.67	31	−134.08

4 Optimization of Weak-Key Attack

4.1 Introduction

This scheme is based on the BGF decoding algorithm of the BIKE scheme. Building upon previous weak key attacks that only considered the multiplicity of the distance spectrum or the influence of error pattern near-codewords individually, it constructs a weak key that results from the combined effect of both the multiplicity of the key distance spectrum and the error pattern near-codewords.

4.2 Our Construction

The overall steps of the scheme are as follows:

1. (Constructing the key) Select the parameters (r, d, t) and multiplicity m, get the key $h \in F_2[x]/(x^r - 1), |h| = d, \mu(\delta, h) = m$, where $\delta \in \{0, 1, \ldots, \lfloor r/2 \rfloor\}$.

2. (Sampling error patterns) Select specific parameters ℓ and collect the patterns of the near-codewords that satisfy the conditions $\ell = \left| h_i^T * e \right|$ according to the key constructed h in the first step (ℓ denotes the number of keys and error vectors at the same position in the first check equation that are both 1).

3. (Measure the probability of decoding failure) Generate the ciphertext according to the BIKE encryption encapsulation, send the ciphertext to the target oracle predicate machine decoder decrypt the ciphertext, test the DFR under the small parameter block, and then use the model of the extrapolation method to test under the target parameter conditions.

4. (Analysis of search density) First calculate the search density of step 1, defined as the probability $\eta(m)$, then calculate the search density of step 2, defined as probability $\eta(\ell)$, and finally calculate the density of the analysis as a whole, that is, the overall probability $\eta_{\mathrm{D}} = \eta(\ell) \cdot \eta(m)$.

5. (Analysis of security) Calculate the product of the two based on the results of Step 3 and Step 4, and perform a comparative analysis to determine whether the results are below the minimum requirements for NIST standardization, and thus whether there is a negative impact on decoding.

4.3 Scheme Analysis

Based on the analysis in Section 3, the multiplicity of the distance spectrum and the overlapping characteristics of the error vectors both impact decoding. The higher the multiplicity, the more complex the key structure, and the higher the decoding failure probability. Conversely, the higher the overlap of the error vectors, the poorer the error correction capability of the decoder [24].

The core advantage of this scheme lies in combining the key multiplicity with the overlapping characteristics of the near-codeword error vectors. These two factors work together on the decoder, significantly increasing the decoding failure probability without adding extra density. Theoretically, this combination can more effectively increase the DFR, thus enhancing the effectiveness of the attack.

4.4 Experimental Methodology

The scheme is based on BIKE's BGF decoding algorithm and simulates the DFR of QC-MDPC codes under IND-CCA security conditions in the $\lambda = 128$ case. Where the parameters of the BIKE scheme are $(r, w, t) = (12323, 142, 134), w = 2d$. The parameter selection for the BGF implementation of the decoding algorithm is the same as in Section 2.

To ensure the accuracy of evaluating the DFR, for the multiplicity $5 \leq m \leq 20$, the test values of r are $r_1 = 9717$ and $r_2 = 9811$, while for the multiplicity $m > 20$, the block parameters of r are $r_1 = 10{,}099$ and $r_2 = 10{,}271$, the simulated decoding samples for each point are 10^6 times, the DFR with its iteration number

of 7 is measured by distributing it under the condition of $\alpha = 0.05$, and the confidence interval value is 95%. Then, the test values of the DFR with the small parameters of r_1 and r_2 are extrapolated to $r = 12,323$.

To test out the effect of key multiplicity and error pattern near-codewords on BIKE (i.e., BGF decoder), increase the weight of the constructed weak key multiplicity m from 5 to 30, and increase the constructed error pattern parameter ℓ from 5 to 25. The simulation results of the DFR tests are as follows [25].

According to the analysis in Table 4, it can be observed that, without considering the error model parameters ℓ, the DFR increases as the multiplicity of secret keys m increases, indicating that the failure probability of the decoding increases. At the same time, when the key has greater multiplicity, the DFR increases with the number of error parameters ℓ, showing that the error rate and the multiplicity of the secret key become more significant, and the probability of decoding failure increases.

Table 4: DFR at $r = 12323$

$\log_2$DFR		ℓ					
		0	**5**	**10**	**15**	**20**	**25**
m	5	**−96.83**	−87.26	−75.19	−55.61	−45.44	−23.96
	10	**−93.45**	−82.19	−65.27	−50.95	−42.93	−20.31
	15	**−80.36**	−65.32	−46.31	−43.14	−36.67	−11.65
	20	**−72.62**	−61.37	−43.21	−36.23	−22.34	−8.32
	25	**−60.19**	−45.31	−25.33	−23.29	−13.31	−6.66
	30	**−19.63**	−26.99	−15.61	−12.36	−9.49	−4.23

When considering multiple key multiplicities and error parameters, the increase in DFR is significant, and the impact of each factor is more prominent. This indicates that the key multiplicity and the near-codeword error vectors have a strong correlation with DFR, which has a major synergistic effect. By comparing the original data with $\ell = 0$, it can be seen that each row of DFR increases, explaining how the two factors interact to affect the decoding.

While DFR testing provides foundational insights, solely relying on this metric without assessing the attack method's efficacy through holistic cryptanalysis is inadequate. For rigorous security evaluation, we must examine the cryptographic systematically, as developed in subsequent analysis.

4.5 Scheme Security Analysis

In order to gain insight into the impact of weak keys on the IND-CCA security of the BIKE mechanism, introduce the concept of $P_{\mathcal{D}} = \eta_{\mathcal{D}} \cdot \mathrm{DFR}_{\mathcal{D},\mathcal{H}}$. For weak keys, a set of weak keys is considered weak if there exists a set of weak keys such that the average DFR of the keys in it is higher than the average DFR of the overall set of keys and the density of the keys $\mathcal{W}$ is sufficiently high that it significantly affects the average DFR of the overall set of keys for the security parameter λ, i.e.,

$$\frac{|\mathcal{W}|}{|\mathcal{H}|}\mathrm{DFR}_{\mathcal{D},\mathcal{H}} > 2^{-\lambda} \tag{14}$$

Therefore, the following inequality must be satisfied, otherwise the IND-CCA security of BIKE is considered to be potentially problematic as it is affected by weak keys.

$$P_{\mathcal{D}} = \frac{|\mathcal{W}|}{|\mathcal{H}|}\mathrm{DFR}_{\mathcal{D},\mathcal{H}} = \eta_{\mathcal{D}} \cdot \mathrm{DFR}_{\mathcal{D},\mathcal{H}} < 2^{-\lambda} \tag{15}$$

Under the IND-CCA language security condition, the attack collects the key of $\mu(\delta, h) = m$, $0 \le m < d$. The multiplicity m of the distance spectrum of (r, d), and the upper bound of the weak key set of the decoder $\mathcal{W}$ can be expressed as:

$$|\mathcal{W}(m)| \le 2\left\lfloor \frac{r}{2} \right\rfloor \frac{r}{d-m}\binom{d-1}{d-m-1}\binom{r-d-1}{d-m-1} \tag{16}$$

In particularly, when $m = d-1$ then the upper bound for the weak key set is $|\mathcal{W}(m)| = 2r\lfloor r/2 \rfloor$, however, for when $m < d-1$, then the upper bound is detailed in the distance spectral analysis 2.2. Thus, the overall upper $\eta(m)$ bound on the density is:

$$\eta(m) = \frac{|\mathcal{W}(m)|}{|\mathcal{H}(m)|} \le \frac{2 \cdot r\left\lfloor \frac{r}{2} \right\rfloor \binom{d-1}{d-m-1}\binom{r-d-1}{d-m-1}}{(d-m)\binom{r}{d}} \tag{17}$$

For the near-codewords in $\mathcal{N}$, assume that there exist weights ω such that $\forall u \in \mathcal{N}$, then $\forall v \in \mathcal{A}_{t,\ell}(\mathcal{N})$, $|v-u| = w + t - 2\ell$, giving an upper bound on the density $\eta(\mathcal{N})$ as [26]:

$$\eta(\ell) = \mathcal{D}_{\mathcal{N},\mathrm{t}}(\ell) \le |\mathcal{N}| \frac{\binom{d}{\ell}\binom{r-d}{t-\ell}}{\binom{r}{t}} \tag{18}$$

The search complexity of this scheme is the product of the densities of the two, $\eta_{\mathcal{D}} = \eta(m) \cdot \eta(\ell)$. Therefore, the search density is upper bounded:

$$\eta_{\mathcal{D}} = \eta(m) \cdot \eta(\ell) \le |\mathcal{N}| \frac{2 \cdot r\left\lfloor \frac{r}{2} \right\rfloor \binom{d-1}{d-m-1}\binom{r-d-1}{d-m-1}\binom{d}{\ell}\binom{r-d}{t-\ell}}{(d-m)\binom{r}{d}\binom{r}{t}} \tag{19}$$

To facilitate the analysis, the overall complexity of the scheme can be calculated according to the above formula, and the data are shown in Table 5.

Without considering the error pattern parameter ℓ, the density gradually decreases as the multiplicity m increases. This is because the structure of high-multiplicity keys is more complex, making it more difficult to find keys with the same multiplicity. Similarly, without considering the multiplicity m, the density gradually decreases as the parameter ℓ increases. This indicates that the higher the overlap between the error vector and the key, the more difficult it is to find such an error vector.

Table 5: Density analysis

$\log_2\eta_{\mathcal{D}}$		ℓ					
		0	**5**	**10**	**15**	**20**	**25**
m	5	**−10.22**	−16.87	−28.64	−50.87	−76.41	−122.97
	10	**−48.16**	−51.74	−62.16	−73.74	−99.28	−145.84
	15	**−86.69**	−90.37	−96.14	−101.37	−126.91	−173.47
	20	**−125.86**	−131.66	−141.43	−152.66	−158.20	−204.76
	25	**−165.72**	−170.15	−175.92	−181.15	−192.69	−239.25
	30	**−206.55**	−215.64	−237.41	−250.64	−262.18	−276.74

When both the key multiplicity m and the error pattern parameter ℓ are considered simultaneously, the density is smaller than when only a single factor is considered. This is because meeting the conditions for both factors is more stringent. For example, when $m = 30$ and $\ell = 25$, there are fewer keys that meet the requirements, and their density in the Table 5 is necessarily the smallest. However, the higher the search complexity, the more difficult it is for an attacker to find keys and error patterns that meet specific conditions.

To comprehensively analyze the variation of the upper bound of density under the influence of dual factors, a visualization of the density η_D as a function of m and ℓ was plotted based on formula 20. From the Fig. 3, it can be observed that high-density regions (such as the yellow areas) correspond to lower search complexity, indicating that attackers can more easily find weak keys or error patterns. Conversely, low-density regions (such as the deep purple areas) correspond to higher search complexity, meaning that attackers find it difficult to locate keys and error patterns that meet the specified conditions. Therefore, searching for keys and error vectors that satisfy both conditions simultaneously is extremely challenging, with a very low numerical density, as seen in the low-density deep purple regions, where the magnitude can be as low as $2^{-276.74}$. Compared to the total number of weak keys, this is negligible. However, more attention should be paid to the yellow regions at the front, where the density is extremely high. The average DFR in these regions may not meet security standards, and this part is highly likely to enhance the effectiveness of attacks. For an in-depth analysis to evaluate the IND-CCA security of the BIKE scheme in the presence of weak keys, after evaluating the overall density of the weak keys, it is also necessary to compute the value of $P_{\mathcal{D}}$ (the product of the density and the DFR), which must satisfy the following inequality [27].

$$P_{\mathcal{D}} = \eta_{\mathcal{D}} \cdot \mathrm{DFR}_{\mathcal{D},\mathcal{H}} > 2^{-\lambda} \tag{20}$$

To explore the specific negative impact of the key on the security of the scheme, it is necessary to quantify the test results. Table 6 shows the quantized values of $P_{\mathcal{D}}$ the product of $\eta_{\mathcal{D}}$ and the DFR in Tables 4 and 5, which calculates the overall security of the scheme with the changes of the parameters variables and the values denoted by $\log_2 P_{\mathcal{D}}$.

In Table 6, the value of $P_{\mathcal{D}}$ reflects the effectiveness of the attacker's. The lower the $P_{\mathcal{D}}$ value, the worse the attacker's effectiveness. For example, $m = 30$ and $\ell = 25$, $P_{\mathcal{D}} = 2^{-280.97}$, which means that the attacker can hardly find a key and error pattern that meet the conditions in this case. However, the higher the $P_{\mathcal{D}}$ value, the better the attack effectiveness. For instance, when $m = 5$ and $\ell = 10$, the $P_{\mathcal{D}}$ value can reach up to $2^{-103.83}$, which is higher than $2^{-107.05}$, when $m = 5$ and $\ell = 0$, improving the attack effectiveness by three percentage points.

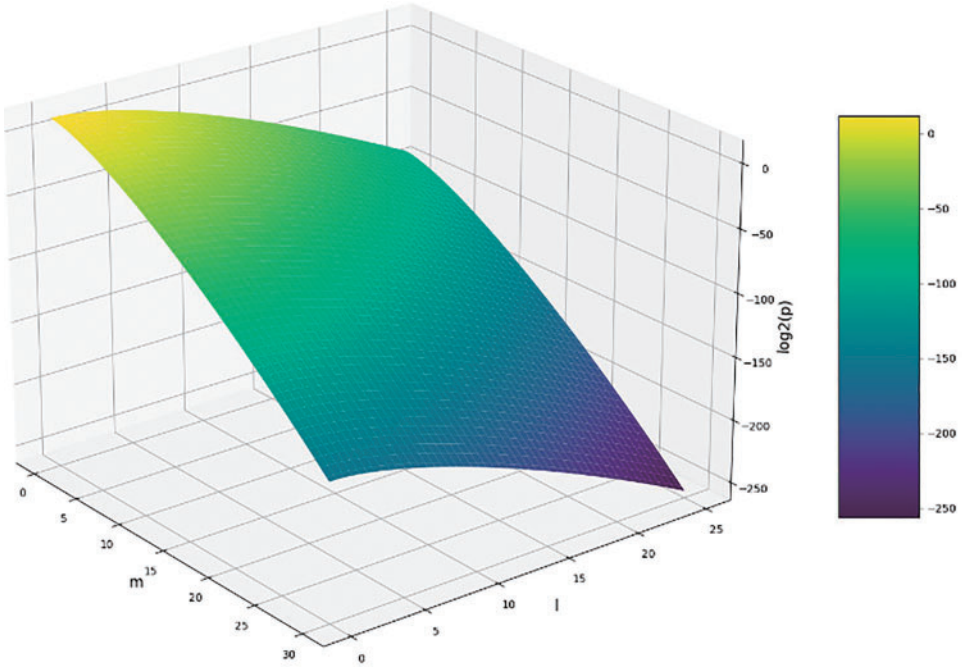

Figure 3: Bivariate density analysis plot

Table 6: Overall security

$\log_2 P_{\mathcal{D}}$		ℓ					
		0	**5**	**10**	**15**	**20**	**25**
m	5	**−107.05**	−104.13	−103.83	−106.48	−121.85	−146.93
	10	**−141.61**	−133.93	−127.43	−124.69	−142.21	−166.15
	15	**−167.06**	−155.69	−142.45	−144.51	−163.58	−185.12
	20	**−198.48**	−193.03	−184.64	−188.89	−180.54	−213.08
	25	**−225.91**	−215.46	−201.25	−204.44	−206.00	−245.91
	30	**−226.18**	−242.63	−253.02	−217.00	−271.67	−280.97

To more intuitively demonstrate the improvement in attack effectiveness, Fig. 4 illustrates the enhancement within the region $5 \leq m \leq 25$, $0 < \ell \leq 10$ (hereinafter referred to as the effective region). In the effective region, for each m value, the three bar graphs from bottom to top represent ℓ values of 0, 5, and 10, respectively. By comparing the original data marked on the white background, it can be observed that the attack effectiveness is improved by an average of 5 to 6 percentage points, especially when $m = 15$ and $\ell = 15$, where the maximum improvement can reach 14.7 percentage points.

Additionally, when $m = 10$, by selecting an appropriate ℓ parameter ($\ell = 10$), the $P_{\mathcal{D}}$ value can be increased to $2^{-127.43}$, which fails to meet the minimum NIST security standard of 128-bit. This indicates that attackers can effectively enhance their attack effectiveness by the ℓ parameter within the effective region. The bar graphs in Fig. 4 show the improvement in attack effectiveness under different combinations of m and ℓ, providing a more comprehensive understanding of the impact of key multiplicity and error vector parameters on the security of the BIKE scheme. This optimized attack scheme significantly enhances the

effectiveness of the attack. By selecting appropriate parameters m and ℓ, the attack scheme achieves notable improvements within the effective region, with an average enhancement of 5 to 6 percentage points and a maximum improvement of up to 14.7 percentage points. The optimized scheme brings keys that originally met security standards close to or slightly below the NIST security threshold (for example, when $m = 10$ and $\ell = 10$, $P_{\mathcal{D}} = 2^{-127.43}$). The best attack effectiveness can reach $2^{-103.83}$, significantly increasing the efficiency of the attack and posing a potential threat to the existing BIKE algorithm.

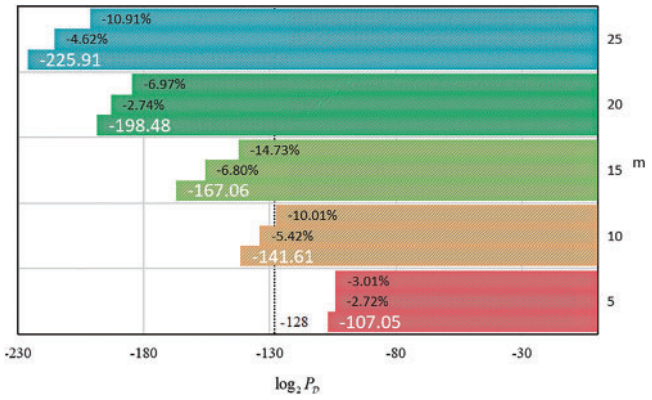

Figure 4: Effectiveness of scheme upgrading

4.6 Comparative Analysis

Wang et al. [8] re-evaluated the DFR of the QC-MDPC code-based scheme by introducing a new concept called the "gathering property". The aggregation property is defined as follows: For $(y_0, y_1) \in \mathrm{R}^2$, $\mathrm{R} = F_2[X]/(X^r - 1)$, if there exists a consecutive sequence of m positions in y_0 containing at least $\omega_H(y_0) - \epsilon$ instances of the element 1, then it is said to satisfy the gathering property. Research has shown a strong correlation between the gathering property and the DFR of QC-MDPC codes. Experimental results indicate that when both the key and the error satisfy the gathering property, the DFR is significantly higher than the average level. Based on the gathering property, the following important theoretical results regarding the DFR of the QC-MDPC scheme were derived.

$$\mathrm{DFR}_{avg} \geq 2^{-116.61} \tag{21}$$

Reference [26] proposed a weak-key model construction method based on the multiplicity analysis of a single-variable distance spectrum (detailed analysis is provided in Section 3.1). The weak-key model structure is defined as $h = (h_0, h_1)$, where each h_i is generated via a mapping function ϕ_δ. The specific expression is:

$$h_i = \phi_\delta\left(x^l\left[\left(1 + x + x^2 + \ldots + x^{f-1}\right) + h_i'\right]\right),\ i \in \{0,1\} \tag{22}$$

In this model, $\delta \in \{1, 2, 3, \ldots, [r/2]\}$ represents the distance between non-zero bits in the weak key, while the parameter $l \in \{0, 1, 2, \ldots, r-1\}$ determines the starting position of the non-zero pattern. The mapping function ϕ_δ operates by replacing the variable x with x^δ, ensuring that the distance between any two consecutive non-zero bits in the polynomial $1 + x + x^2 + \ldots + x^{f-1}$ remains constant at δ. The study further analyzed the impact of this weak-key model on the security of the BIKE scheme. For the BIKE scheme with parameters $(r, \omega, t) = (12323, 142, 134)$ and $\lambda = 128$, the authors experimentally evaluated the performance of weak keys under IND-CCA. The relevant data is presented in Table 7.

Table 7: Average Decoding Failure Rate (DFR) Table for the Distance Spectrum Attack Scheme

f	$\log_2 DFR$	$\log_2 \eta_{\mathcal{D}}$	$DFR \cdot \eta_{\mathcal{D}}$
5	−96.83	−10.22	$2^{-107.05}$
10	−93.45	−48.16	$2^{-141.61}$
15	−80.36	−86.69	$2^{-167.06}$
20	−72.62	−125.86	$2^{-198.48}$
25	−60.19	−165.72	$2^{-225.91}$
30	−41.63	−194.55	$2^{-236.18}$

Compared to the aggregation degree scheme proposed by Wang et al. and the distance spectrum scheme in Reference [8], our proposed scheme demonstrates superior attack performance. In Wang et al.'s scheme, the lower bound of the average DFR is $\mathrm{DFR}_{avg} \geq 2^{-116.61}$, while in Reference [23], the lower bound is $2^{-107.05}$. By optimizing the attack strategy, our scheme successfully increases the average DFR to $\mathrm{DFR}_{\mathrm{avg}} \geq 2^{-103.83}$. This improvement means attackers can trigger decoding failures with higher probability, thereby more easily extracting key-related information. A comparative analysis of the schemes is presented in Table 8.

Table 8: Comparative analysis

	$\log_2 DFR$	$\log_2 \eta_{\mathcal{D}}$	$\log_2 P_{\mathcal{D}}$
Optimized scheme	$2^{-75.19}$	$2^{-28.64}$	$2^{-103.83}$
Reference [8]	$2^{-87.28}$	$2^{-29.33}$	$2^{-116.61}$
Reference [23]	$2^{-96.83}$	$2^{-10.22}$	$2^{-107.05}$

The key innovation of our scheme lies in its ability to increase the decoding failure probability without introducing additional search density. This characteristic allows attackers to significantly enhance the success rate of key recovery while maintaining low computational resource consumption. Specifically, by refining the attack model and algorithms, our scheme enables attackers to more efficiently leverage decoding failure information for key recovery under the same computational complexity.

4.7 Weak-Key Detection

The general methods for dealing with weak keys primarily include three dimensions: static analysis, dynamic testing, and algorithm-specific detection. In terms of static analysis, the evaluation mainly relies on the following technical approaches: key space size analysis, repetition and predictable pattern recognition, known weak key dictionary comparison, and information entropy calculation. Dynamic testing employs practical attack validation techniques, including brute-force attempts, differential analysis, and linear analysis, to empirically assess the key's resistance to attacks.

For algorithm-specific detection, the sample autocorrelation function test stands out due to its exceptional capability in detecting key periodicity. The principle of this method involves calculating the autocorrelation coefficient between the initial sequence and its left-shifted sequence by k positions, thereby quantifying the degree of internal correlation within the sequence. In practical implementation, by analyzing the distribution of autocorrelation coefficients at different lag orders, a significant peak at a specific lag order indicates the presence of correlation at that period. This method not only effectively identifies internal

sequence variation characteristics but also accurately detects periodic properties of the sequence, making it particularly suitable for weak key periodicity detection based on distance spectrum analysis.

In summary, by integrating the technical approaches of static analysis, dynamic testing, and algorithm-specific detection, a comprehensive weak key detection system can be constructed. Among these, the autocorrelation function test, as a critical method in algorithm-specific detection, provides effective technical support for the periodic analysis of weak keys.

5 Conclusion

This study delves into the BGF decoding algorithm of the BIKE scheme and addressing its potential weak key issue, proposes an innovative optimized attack strategy. Through experiments, the specific impact of weak keys on the BGF decoder's DFR is assessed. The analysis indicates that these weak keys pose a potential threat to the IND-CCA security of the BIKE scheme. Therefore, before claiming the IND-CCA security of the BIKE scheme, the security issues caused by weak keys must be addressed. In the future, this research will be extended to algorithms with higher security levels to further validate its universality and effectiveness. This study not only provides new insights into the security of the BIKE scheme but also offers a reference for the future security evaluation of post-quantum cryptography.

Acknowledgement: Not applicable.

Funding Statement: This research was funded by Beijing Institute of Electronic Science and Technology Postgraduate Excellence Demonstration Course Project (20230002Z0452).

Author Contributions: The authors confirm contribution to the paper as follows: Conceptualization, Bing Liu; methodology, Bing Liu; software, Ting Nie; validation, Yansong Liu; formal analysis, Yansong Liu; investigation, Weibo Hu; resources, Ting Nie; data curation, Yansong Liu; writing—original draft preparation, Ting Nie; writing—review and editing, Bing Liu; visualization, Ting Nie; supervision, Bing Liu; project administration, Ting Nie; funding acquisition, Bing Liu. All authors reviewed the results and approved the final version of the manuscript.

Availability of Data and Materials: The data that support the findings of this study are available from the corresponding author, Bing Liu, upon reasonable request.

Ethics Approval: Not applicable.

Conflicts of Interest: The authors declare no conflicts of interest to report regarding the present study.

References

1. Li S, Chen Y, Chen L, Liao J, Kuang C, Li K, et al. Post-quantum security: opportunities and challenges. Sensors. 2023;23(21):8744. doi:10.3390/s23218744.
2. Shor PW. Algorithms for quantum computation: discrete logarithms and factoring. In: Proceedings 35th Annual Symposium on Foundations of Computer Science. Santa Fe, NM, USA: IEEE; 1994. p. 124–34.
3. Alagic G, Alagic G, Apon D, Cooper D, Dang Q, Dang T, et al. Status report on the third round of the NIST post-quantum cryptography standardization process. Gaithersburg, MD, USA: National Institute of Standards and Technology; 2022.
4. Sendrier N. Code-based cryptography: state of the art and perspectives. IEEE Secur Priv. 2017;15(4):44–50.
5. Drucker N, Gueron S, Kostic D. QC-MDPC decoders with several shades of gray. In: International Conference on Post-Quantum Cryptography. Cham: Springer International Publishing; 2020. p. 35–50.
6. Aragon N, Barreto P, Bettaieb S, Bidoux L, Blazy O, Deneuville JC, et al. BIKE: bit flipping key encapsulation; 2022 [cited 2025 Apr 15]. Available from: https://bikesuite.org/files/v4.2/BIKE_Spec.2021.09.29.1.pdf.

7. Guo Q, Johansson T, Stankovski P. 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; 2016 Dec 4–8; Hanoi, Vietnam. Berlin/Heidelberg, Germany: Springer; 2016. p. 789–815.
8. Wang T, Wang A, Wang X. Exploring decryption failures of BIKE: new class of weak keys and key recovery attacks. In: Annual International Cryptology Conference. Cham: Springer Nature Switzerland; 2023. p. 70–100.
9. Eaton E, Lequesne M, Parent A, Sendrier N. QC-MDPC: a timing attack and a CCA2 KEM. In: International Conference on Post-Quantum Cryptography. Cham: Springer International Publishing; 2018. p. 47–76.
10. Guo Q, Johansson T, Wagner PS. A key recovery reaction attack on QC-MDPC. IEEE Trans Inf Theory. 2018;65(3):1845–61. doi:10.1109/tit.2018.2877458.
11. Nilsson A, Johansson T, Wagner PS. Error amplification in code-based cryptography. Cryptology ePrint Archive; 2018. doi:10.13154/tches.v2019.i1.238-258.
12. Drucker N, Gueron S, Kostic D. On constant-time QC-MDPC decoding with negligible failure rate. Cryptology ePrint Archive; 2019 [cited 2025 Apr 15]. Available from: https://eprint.iacr.org/2017/604.
13. Niederreiter H. Knapsack-type cryptosystems and algebraic coding theory. Prob Control Inform Theory. 1986;15(2):157–66.
14. Fujisaki E, Okamoto T. Secure integration of asymmetric and symmetric encryption schemes. In: Annual International Cryptology Conference. Berlin/Heidelberg, Germany: Springer Berlin Heidelberg; 1999. p. 537–54.
15. Dent AW. A designer's guide to KEMs. In: IMA International Conference on Cryptography and Coding. Berlin/Heidelberg, Germany: Springer Berlin Heidelberg; 2003. p. 133–51.
16. Hofheinz D, Hövelmanns K, Kiltz E. A modular analysis of the Fujisaki-Okamoto transformation. In: Theory of Cryptography Conference. Cham: Springer International Publishing; 2017. p. 341–71.
17. Gallager R. Low-density parity-check codes. IRE Trans Inf Theory. 1962;8(1):21–8. doi:10.1109/tit.1962.1057683.
18. Vasseur V. QC-MDPC codes DFR and the IND-CCA security of bike; 2022 [cited 2025 Apr 15]. Available from: https://ia.cr/2021/1458.
19. Sendrier N, Vasseur V. On the decoding failure rate of QC-MDPC bit-flipping decoders. In: Ding J, Steinwandt R, editors. Post-quantum cryptography. Cham: Springer; 2019. Vol. 11505. p. 404–16.
20. Sendrier N, Vasseur V. About low DFR for QC-MDPC decoding. In: International Conference on Post-Quantum Cryptography. Cham: Springer International Publishing; 2020. p. 20–34.
21. Vasseur V. Post-quantum cryptography: a study of the decoding of QC-MDPC codes. France: Université de Paris; 2021.
22. Chou T, Maezawa Y, Miyaji A. A closer look at the Guo-Johansson–Stankovski attack against QC-MDPC codes. In: International Conference on Information Security and Cryptology. Cham: Springer International Publishing; 2018. p. 341–53.
23. Sendrier N, Vasseur V. On the existence of weak keys for QC-MDPC decoding; 2020 [cited 2025 Apr 15]. Available from: https://ia.cr/2020/1232.
24. Nosouhi MR, Shah SW, Pan L, Zolotavkin Y, Nanda A, Gauravaram P, et al. Weak-key analysis for bike post-quantum key encapsulation mechanism. IEEE Trans Inf Forensics Secur. 2023;18(3):2160–74. doi:10.1109/tifs.2023.3264153.
25. Sendrier N. Secure sampling of constant-weight words–application to bike. Cryptology ePrint Archive; 2021 [cited 2025 Apr 15]. Available from: https://ia.cr/2021/1631.
26. Arpin S, Billingsley TR, Hast DR, Lau JB, Perlner R, Robinson A. A study of error floor behavior in QC-MDPC codes. In: International Conference on Post-Quantum Cryptography. Cham: Springer International Publishing; 2022. p. 89–103.
27. Bardet M, Dragoi V, Luque JG, Otmani A. Weak keys for the quasi-cyclic MDPC public key encryption scheme. In: Progress in Cryptology–AFRICACRYPT 2016: 8th International Conference on Cryptology in Africa; 2016 Apr 13–15; Fes, Morocco: Springer International Publishing; 2016. p. 346–67.