NIST Internal Report
NIST IR 8545

Status Report on the Fourth Round of the NIST Post-Quantum Cryptography Standardization Process

Gorjan Alagic
Maxime Bros
Pierre Ciadoux
David Cooper
Quynh Dang
Thinh Dang
John Kelsey
Jacob Lichtinger
Yi-Kai Liu
Carl Miller
Dustin Moody
Rene Peralta
Ray Perlner
Angela Robinson
Hamilton Silberg
Daniel Smith-Tone
Noah Waller

This publication is available free of charge from:
https://doi.org/10.6028/NIST.IR.8545

NATIONAL INSTITUTE OF
STANDARDS AND TECHNOLOGY
U.S. DEPARTMENT OF COMMERCE

NIST Internal Report
NIST IR 8545

Status Report on the Fourth Round of the NIST Post-Quantum Cryptography Standardization Process

Gorjan Alagic
Maxime Bros
Pierre Ciadoux
David Cooper
Quynh Dang
Thinh Dang
John Kelsey
Jacob Lichtinger

Carl Miller
Dustin Moody
Rene Peralta
Ray Perlner
Angela Robinson
Hamilton Silberg
Daniel Smith-Tone
Noah Waller
Computer Security Division
Information Technology Laboratory

Yi-Kai Liu
Applied and Computational Mathematics Division
Information Technology Laboratory

This publication is available free of charge from:
https://doi.org/10.6028/NIST.IR.8545

March 2025

U.S. Department of Commerce
Howard Lutnick, Secretary

National Institute of Standards and Technology
Craig Burkhardt, Acting Under Secretary of Commerce for Standards and Technology and Acting NIST Director

Publication History
Approved by the NIST Editorial Review Board on 2025-03-05

How to cite this NIST Technical Series Publication:
Alagic G, Bros M, Ciadoux P, Cooper D, Dang Q, Dang T, Kelsey J, Lichtinger J, Liu YK, Miller C, Moody D, Peralta R, Perlner R, Robinson A, Silberg H, Smith-Tone D, Waller N (2025) Status Report on the Fourth Round of the NIST Post-Quantum Cryptography Standardization Process. (National Institute of Standards and Technology, Gaithersburg, MD), NIST Internal Report (IR) NIST IR 8545.
https://doi.org/10.6028/NIST.IR.8545

Author ORCID iDs

Gorjan Alagic: 0000-0002-0107-6037
Maxime Bros: 0000-0001-7838-2529
Pierre Ciadoux: 0009-0001-2272-681X
David Cooper: 0009-0001-2410-5830
Quynh Dang: 0009-0005-9801-6805
Thinh Dang: 0000-0001-9705-0925
John Kelsey: 0000-0002-3427-1744
Jacob Lichtinger: 0000-0003-2407-5309
Yi-Kai Liu: 0000-0001-7458-4721
Carl Miller: 0000-0003-1917-1531
Dustin Moody: 0000-0002-4868-6684
Rene Peralta: 0000-0002-2318-7563
Ray Perlner: 0000-0001-8793-2238
Angela Robinson: 0000-0002-1209-0379
Hamilton Silberg: 0009-0004-4178-8954
Daniel Smith-Tone: 0000-0002-7995-8734
Noah Waller: 0000-0002-6979-9725

Contact Information

pqc-comments@nist.gov

Additional Information

Additional information about this publication is available at https://csrc.nist.gov/pubs/ir/8545/final, including related content, potential updates, and document history.

All comments are subject to release under the Freedom of Information Act (FOIA).

Abstract

NIST is selecting public-key cryptographic algorithms through a public, competition-like process to specify additional digital signature, public-key encryption, and key-establishment algorithms to supplement FIPS 186-5, SP 800-56Ar3, and SP 800-56Br2. These algorithms are intended to protect sensitive information well into the foreseeable future, including after the advent of quantum computers. In the fourth round of the Post-Quantum Cryptography Standardization Process, NIST selected four candidate algorithms for key establishment to be studied: BIKE, Classic McEliece, HQC, and SIKE. This report describes the evaluation and selection process of these fourth-round candidates based on public feedback and internal review. The report summarizes each of the candidate algorithms and identifies those selected for standardization. The only key-establishment algorithm that will be standardized is HQC, and NIST will develop a standard based on HQC to augment its key-establishment portfolio.

Keywords

cryptography; key-encapsulation mechanism (KEM); key establishment; post-quantum cryptography; quantum-resistant; quantum-safe.

Reports on Computer Systems Technology

Contents

List of Tables

Supplemental Content

The NIST Post-Quantum Cryptography Standardization Process web page is available at https://csrc.nist.gov/Projects/post-quantum-cryptography/post-quantum-cryptography-standardization.

Acknowledgments

NIST would like to thank all of the candidate submission teams who developed, designed, and analyzed the post-quantum candidate algorithms and prepared detailed submission packages.

NIST is also grateful for the efforts of those in the cryptographic community who provided security, implementation, and performance analyses of the candidate algorithms during the four rounds of the process. NIST would not be able to select post-quantum digital algorithms for standardization without the combined efforts of these individuals and the algorithm submitters.

The authors of this report also appreciate the efforts by other members of NIST's PQC team who reviewed candidate algorithms, analyses, and public comments; performed testing; provided technical and administrative support; and participated in numerous meetings to discuss the selection of the candidate to be standardized. They are Zuzana Bajcsy, Lily Chen, Morris Dworkin, Sara Kerman, and Andrew Regenscheid.

1. Introduction

The National Institute of Standards and Technology (NIST) initiated the Post-Quantum Cryptography (PQC) Standardization Process in December 2016 to select quantum-resistant public-key cryptographic algorithms for standardization in response to the substantial development and advancement of quantum computing. After three rounds of evaluation and analysis, NIST announced the selection of the first algorithms to be standardized [2]. The key encapsulation mechanism (KEM) selected for standardization was CRYSTALS-Kyber (ML-KEM [3]). The digital signatures selected were CRYSTALS-Dilithium (ML-DSA [4]), Falcon (FN-DSA), and SPHINCS$^+$ (SLH-DSA [5]). For a detailed explanation of NIST's choices, as well as a summary of the third round, see NIST IR 8413 [2].

In addition to those initial selections, NIST advanced four KEM candidates to the fourth round for continued evaluation: BIKE [6], Classic McEliece [7], HQC [8], and SIKE [9]. These algorithms were all based on different security assumptions than ML-KEM. NIST indicated that it would select one or two of the algorithms for standardization at the conclusion of the fourth round.

The fourth round began in July 2022 and involved a thorough analysis of the theoretical and empirical evidence used to justify the security of the candidates. During this time, the submitters of SIKE acknowledged its insecurity and recommended against its further use. The submission teams of the unbroken fourth-round candidates were invited to present updates for their candidate algorithms at the Fifth NIST PQC Standardization Conference in Rockville, Maryland, on April 10-12, 2024. The submitters participated in a joint panel to discuss the candidates' merits, and several researchers presented work that was relevant to the PQC standardization process.

Throughout the fourth round, NIST received valuable feedback from the cryptographic community. Based on this feedback and internal reviews of the fourth-round candidates, NIST announced the selection of HQC in March 2025 for standardization.

Table 1 shows a timeline of major events with respect to the NIST PQC Standardization Process to date.

Table 1. Timeline of the NIST Post-Quantum Cryptography Standardization Process

Date	*Event*
April 2015	Workshop on Cybersecurity in a Post-Quantum World, NIST, Gaithersburg, MD
February 2016	PQC Standardization: Announcement and outline of NIST's Call for Submissions presentation given at PQCrypto 2016
April 2016	Release of IR 8105, *Report on Post-Quantum Cryptography* [10]
December 2016	Federal Register Notice – Announcing Request for Nominations for Public-Key Post-Quantum Cryptographic Algorithms [11]
November 30, 2017	Submission Deadline for NIST PQC Standardization Process
December 2017	Announcement of first-round candidates and beginning of first-round public comment period
April 2018	First NIST PQC Standardization Conference, Ft. Lauderdale, FL
January 2019	Announcement of second-round candidates; release of IR 8240, *Status Report on the First Round of the NIST Post-Quantum Cryptography Standardization Process* [12]; and beginning of second-round public comment period
August 2019	Second NIST PQC Standardization Conference, Santa Barbara, CA
April 2020	Call for feedback on the selection of third-round candidates
July 2020	Announcement of third-round finalists and alternate candidates; release of IR 8309, *Status Report on the Second Round of the NIST Post-Quantum Cryptography Standardization Process* [13]; and beginning of third-round public comment period
June 2021	Third NIST PQC Standardization Conference, held virtually
July 2022	Announcement of candidate algorithms to be standardized and alternate candidates advancing to the fourth round; release of IR 8413, *Status Report on the Third Round of the NIST Post-Quantum Cryptography Standardization Process*; and beginning of fourth-round public comment period
October 2022	Fourth round specifications published on NIST's PQC website
November 2022	Fourth NIST PQC Standardization Conference, held virtually
August 2023	Draft versions of FIPS 203 [14], FIPS 204 [4], and FIPS 205 [5] posted for public comment
April 2024	Fifth NIST PQC Standardization Conference, Gaithersburg, MD
August 2024	Final versions of FIPS 203 [14], FIPS 204 [4], and FIPS 205 [5] published
January 2025	Draft for KEM guidance SP 800-227 posted for public comment
March 2025	Announcement of fourth-round candidate algorithm to be standardized and release of IR 8545, *Status Report on the Fourth Round of the NIST Post-Quantum Cryptography Standardization Process*

1.1. Purpose and Organization of This Document

This report summarizes the fourth round of the NIST PQC Standardization Process.

Section 2 enumerates the candidates that were included in the fourth round. It also describes the evaluation criteria and selection process used to ultimately select HQC for standardization.

Section 3 summarizes each of the fourth-round candidates, including a brief description of the algorithm and its characteristics with regard to security, performance, and implementation. This section also presents the rationale for standardizing some candidate algorithms and not others.

Section 4 concludes and describes the next steps in the standardization process.

2. Evaluation Criteria and Selection Process

2.1. Acceptance of the Fourth-Round Candidates

NIST selected four candidate algorithms for the fourth round, all of which were KEMs. Classic McEliece was a third-round finalist, and the other three algorithms were alternates [13]. The set of finalists included the algorithms that NIST considered to be the most promising to fit the majority of use cases and be ready for standardization soon after the third round. The alternate candidates were regarded as potential candidates for future standardization, most likely after another round of evaluation.

The submission teams were allowed to make minor modifications and resubmit their packages, which had to meet the same requirements as the original submissions. The complete updated specifications were posted on NIST's PQC website [15] for public review on October 27, 2022. Most of the changes focused on fixing minor issues that were identified during the third round and clarifying or simplifying the submission specification. One modification of note that occurred during the fourth round is BIKE's decoder. The thresholds for the decoder were altered to reduce the risk of decryption failure. No major redesigns or changes were allowed.

Table 2. Fourth-round KEM candidates organized by category, with the candidate selected for standardization bolded and in blue

Code-Based	Isogeny-Based
BIKE	SIKE
HQC	
Classic McEliece	

2.2. Evaluation Criteria

NIST's Call for Proposals [16] identified three broad aspects of the evaluation criteria that would be used to compare candidate algorithms throughout the NIST PQC Standardization Process: 1) security, 2) cost and performance, and 3) algorithm and implementation characteristics. These criteria are described below, along with a discussion of how they impacted the fourth-round candidate evaluations.

2.2.1. Security

As with the previous phases of the PQC Standardization Process, security was the most important factor that NIST considered when evaluating the fourth-round candidate schemes. In the third round of the PQC Standardization Process, NIST selected one KEM — Kyber — that was then standardized as ML-KEM in FIPS 203 [14]. The security of ML-KEM is based primarily on the presumed hardness of certain computational problems in lattices. As discussed in the third-round report, NIST values having a variety of computational hardness assumptions and aims to reduce the risk that a single cryptanalytic breakthrough will leave no viable standard for key establishment. In pursuit of that goal, NIST selected fourth-round candidates whose security was based on computational assumptions that differ significantly from that of ML-KEM. Specifically, the candidates consisted of the isogeny-based KEM SIKE and the code-based KEMs BIKE, HQC, and Classic McEliece. See Table 2.

NIST's key-establishment standards are currently utilized in a wide variety of applications. The specific properties required for a key-establishment scheme to provide security in a given application can vary. However, in terms of formal security definitions, a single notion suffices for key-establishment schemes that are intended for general use: semantic security with respect to adaptive chosen ciphertext attacks (equivalently, IND-CCA2 security). ML-KEM is believed to satisfy IND-CCA2 security and is expected to serve as a general-purpose scheme in any application that calls for NIST-approved post-quantum key-establishment.

The formal security statuses of the fourth-round KEM candidates vary significantly. SIKE, the sole isogeny-based candidate, was broken and thus does not satisfy IND-CCA2 security [17]. The code-based candidates BIKE, HQC, and Classic McEliece are believed to satisfy IND-CCA2 security. However, NIST's level of confidence in the IND-CCA2 security of these schemes is not equal. Notably, NIST has a higher level of confidence in the IND-CCA2 security of HQC than BIKE (see Sec. 3 for further details).

Submitters to the fourth round were encouraged but not required to provide proofs of IND-CCA2 security (from clearly stated computational assumptions) in relevant models. NIST defined five security categories to compare the security strengths provided by the submissions. Submitters were asked to provide a classification of the security of the parameter sets of their schemes following the definitions provided in [16].

NIST also listed other desirable security properties, such as resistance to side-channel and multi-key attacks and resistance to misuse. Submissions were encouraged to note any additional desirable security properties that they provided. Finally, NIST required submission packages to summarize known cryptanalytic attacks on the scheme and complexity estimates for those attacks.

2.2.2. Cost and Performance

The second-most important criterion when evaluating candidate algorithms was their performance characteristics:

- Sizes of encapsulation keys and ciphertexts
- Computational efficiencies of encapsulations, decapsulations, and key generations (i.e., the speeds of the algorithms)

Tables 3 through 5 show representative benchmarks for key generations, encapsulations, and decapsulations of BIKE, HQC, and Classic McEliece, respectively. Each row is a specific parameter set from the corresponding submission. The “Level” columns indicate the security categories that the submission parameter sets claim to meet. BIKE and HQC each had one parameter set per security category, while Classic McEliece had two. The Classic McEliece f versions have faster key generation, while the non-f versions have simpler key generation.

In these benchmarks, BIKE is 6-10 times slower than HQC in key generation, 5-7 times slower than HQC in decapsulation, and about twice as fast as HQC in encapsulation. Key generation in Classic McEliece is an outlier, being three orders of magnitude more costly than HQC.

Parameter Set	Level	keygen	encaps	decaps
BIKE Level 1	I	637	111	1 428
BIKE Level 3	III	1 892	251	4 313
BIKE Level 5	V	4 535	505	10 382

Table 3. Performance of BIKE in thousands of cycles on x86_64 [1]

Parameter Set	Level	keygen	encaps	decaps
hqc-128	I	105	197	360
hqc-192	III	244	460	746
hqc-256	V	4246	844	1 410

Table 4. Performance of HQC in thousands of cycles on x86_64 [1]

Parameter Set	Level	keygen	encaps	decaps
mceliece348864	I	137 345	49	120
mceliece348864f		114 189	45	120
mceliece460896	III	430 364	91	232
mceliece460896f		313 600	92	231
mceliece6688128	V	674 012	196	273
mceliece6688128f		493 758	176	274
mceliece6960119	V	602 164	167	252
mceliece6960119f		404 166	169	253
mceliece8192128	V	686 110	203	269
mceliece8192128f		453 985	206	269

Table 5. Performance of Classic McEliece in thousands of cycles on x86_64 [1]

Tables 6 through 8 show the sizes of keys and ciphertexts for BIKE, HQC, and Classic McEliece. The encapsulation keys of HQC are about 41-47 % larger than those of BIKE. The ciphertexts of HQC are about three times larger than the ciphertexts of BIKE.

Parameter Set	Level	Encapsulation Key	Decapsulation Key	Ciphertext
BIKE Level 1	I	1 541	281	1 573
BIKE Level 3	III	3 083	419	3 115
BIKE Level 5	V	5 122	580	5 154

Table 6. BIKE keys and ciphertext sizes in bytes

Parameter Set	Level	Encapsulation Key	Decapsulation Key	Ciphertext
hqc-128	I	2 249	40	4 497
hqc-192	III	4 522	40	9 042
hqc-256	V	7 245	40	14 485

Table 7. HQC keys and ciphertext sizes in bytes

Parameter Set	Level	Encapsulation Key	Decapsulation Key	Ciphertext
mceliece348864 mceliece348864f	I	261 120	6 492	96
mceliece460896 mceliece460896f	III	524 160	13 608	156
mceliece6688128 mceliece6688128f	V	1 044 992	13 932	208
mceliece6960119 mceliece6960119f	V	1 047 319	13 948	194
mceliece8192128 mceliece8192128f	V	1 357 824	14 120	208

Table 8. Classic McEliece keys and ciphertext sizes in bytes

There are a few studies comparing the performances of the KEMs in various protocols [18–21]. The study on the performance of post-quantum XML encryption and SAML SSO [21] contains data that compare BIKE and Classic McEliece in those protocols. For hybrid XML encryption, Classic McEliece slightly outperforms BIKE in decryption time and total time but results in much larger data sizes. When used for SAML SSO, BIKE generally outperforms Classic McEliece in time and produces much smaller bandwidths. Experiments on the performance of post-quantum KEMs in TLS 1.3 and QUIC [18–20] produce data that compare BIKE and HQC. Generally, when network conditions (e.g., transmission rates and packet loss) are ignored or sufficiently good, HQC results in faster handshakes. In contrast, when network conditions are sufficiently bad, BIKE outperforms HQC. Packet delay seems to affect both HQC and BIKE equally.

These results align with a prior expectation about the performances of BIKE and HQC based on their differences in speeds and sizes. When the size differences between HQC and BIKE do not affect the protocol execution time, the protocol runs faster with HQC. When the differences affect the protocol execution time noticeably, BIKE is more attractive than HQC. For TLS, BIKE would likely be more attractive than HQC over the web. The cited studies do not provide data for Classic McEliece, which is likely not a desirable choice for TLS 1.3 and QUIC due to its generally large encapsulation keys.

2.2.3. Algorithm and Implementation Characteristics

The Call for Proposals [22] also requested various desirable algorithm and implementation characteristics for consideration, particularly flexibility, simplicity, and ease of adoption.

An important characteristic of candidates is their potential performance impact on existing widely used protocols (e.g., TLS, IPSec, and SSH) and certificates. The third round included real-world experiments to identify potential performance problems with the algorithms. These experiments continued into the fourth round with a greater focus on HQC and BIKE (see Sec. 2.2.2).

NIST believes it is important to select cryptographic standards that will be capable of protecting sensitive government information as well as being widely adopted for use in industry. In selecting a cryptographic algorithm for standardization, an evaluation factor is whether a patent might hinder the adoption of a cryptographic standard. All submission teams were required to submit statements regarding knowledge of patents involving their algorithms and implementations, which are available on the NIST PQC fourth round submissions website [23]. The submitters of HQC indicated two patents that could potentially be relevant to an implementation of HQC. However, the patent owner committed and agreed to grant to any interested party on a worldwide basis a non-exclusive license for the purpose of implementing the standard without compensation and under reasonable terms and conditions that are demonstrably free of any unfair discrimination.[1]

2.3. Selection of the Candidates for Standardization

In relative order of importance, NIST considered the security, cost and performance, and algorithm and implementation characteristics of the candidates in selecting what to standardize. Early in the fourth round, published cryptanalytic results demonstrated that SIKE was insecure [17, 24, 25], resulting in its removal from consideration [9].

[1] See the Statement by Patent Owner included with the HQC submission at https://csrc.nist.gov/csrc/media/Projects/post-quantum-cryptography/documents/round-4/final-ip-statements/HQC-Statements-Round4.pdf

In IR 8413 [2], NIST requested feedback on specific use cases for which Classic McEliece would be a good solution. Responses noted that Classic McEliece may provide better performance than BIKE or HQC for applications in which a public key can be transferred once and then used for several encapsulations (e.g., file encryption and virtual private networks [VPNs]) due to its small ciphertext size and fast encapsulation and decapsulation. There was also some interest in Classic McEliece based on the perception that it is a conservative choice. However, the interest expressed in Classic McEliece was limited, and having more standards to implement adds complexity to protocols and PQC migration.

Classic McEliece is currently under consideration for standardization by the International Organization for Standardization (ISO). Concurrent standardization of Classic McEliece by NIST and ISO risks the creation of incompatible standards. After the ISO standardization process has been completed, NIST may consider developing a standard for Classic McEliece based on the ISO standard. However, Classic McEliece is no longer under consideration for standardization as part of the current NIST PQC Standardization Process.

At the end of the third round, NIST indicated its intent to standardize at most one of BIKE or HQC for use as a general-purpose KEM [2]. As specified in the Call for Proposals [22], submitted KEMs were evaluated based on how well they appear to provide IND-CCA2 security, particularly for KEMs intended for general use. While NIST has confidence in the indistinguishability under chosen-plaintext attack (IND-CPA) security of BIKE and HQC, both schemes require a sufficiently low decryption failure rate (DFR) in order to be IND-CCA2-secure. There is evidence that HQC has a sufficiently low DFR and recent work indicates that with minor modifications, BIKE achieves the same [26]. However, NIST does not consider the DFR analysis for BIKE to be as mature as that for HQC. Additionally, HQC is not believed to require additional modifications to achieve the desired security properties. Given the critical need for strong IND-CCA2 security in a general-purpose KEM, HQC was selected for standardization.

In summary, NIST has only selected HQC for standardization. The algorithms that were not selected are not under consideration for standardization by NIST as part of the current NIST PQC Standardization Process.

3. Summary of the Fourth-Round Candidates

This section describes each of the fourth-round candidates, including their advantages and disadvantages and why a scheme was selected for standardization or not.

Section 3 of IR 8413[2] introduces some computational and security concepts and history that might be referenced throughout the subsequent subsections. The provided information reduced redundancy, as some of the candidates' security analyses have properties in common. The information was not intended to be an exhaustive security or literature review.

3.1. HQC

HQC (Hamming Quasi-Cyclic) is a KEM based on quasi-cyclic codes, where no trapdoor is hidden in the code [27]. It was designed to leverage the structural advantages of quasi-cyclic codes while maintaining a more direct security reduction to the problem of decoding a random linear code. Unlike the other code-based candidates, the only coding-theory hardness assumptions required by HQC's security proof are parameterizations of the decisional Quasi-Cyclic Syndrome Decoding (QCSD) assumption. BIKE additionally assumes the hardness of Quasi-Cyclic Codeword Finding (QCCF), and Classic McEliece requires assumptions concerning binary Goppa codes [27, 28].

Design. HQC is similar in structure to Learning with Errors (LWE)-based cryptosystems, like Regev [29], LPR (Lyubashevsky, Peikert, Regev) [30], and ML-KEM [14]. The IND-CPA-secure public-key encryption (PKE) can be described as follows.

Let $\mathscr{R} = \mathbb{F}_2[x]/(x^n - 1)$ for n prime such that $x^n - 1$ has only two irreducible factors modulo 2. The secret key is a randomly sampled pair $(x, y) \in \mathscr{R}^2$, and the public key is the pair $(h, s = x + h \cdot y)$, where h is randomly sampled from $\mathscr{R}$. Because the secret key is generated independently of the underlying quasi-cyclic code, there is no hidden structure in the HQC public parity-check matrix. This enables the security reduction to be independent of the decoding algorithm used for decryption [27]. In addition to h, the public key includes a public generator matrix $G \in \mathbb{F}_2^{k \times n}$ for a concatenated Reed-Muller Reed-Solomon (RMRS) code. The structure of this code is assumed to be visible to all parties.

To encrypt a message $m \in \mathbb{F}_2^k$, the sender randomly samples three polynomials $e, r_1, r_2 \in \mathscr{R}$ of appropriate low weights and responds with the ciphertext

$$c = (u, v) := (r_1 + h \cdot r_2, mG + s \cdot r_2 + e). \tag{1}$$

To decrypt, the receiver uses the decoding algorithm for an RMRS code to decode $(v - u \cdot y)$.

Security. The IND-CPA security of HQC relies on the difficulty of the QCSD with parity problem. Applying the Fujisaki Okamoto (FO) transform [31] to the CPA-secure PKE achieves an IND-CCA2 KEM.

The decoder used in HQC has a well-defined minimum distance d and, consequently, a determinable error-correction capability $\delta = \lfloor \frac{d-1}{2} \rfloor$. The probability that an HQC ciphertext includes error e such that $|e| > \delta$ is captured in a closed-form analysis and used to produce a heuristic[2] upper bound on the DFR. A sufficiently low DFR is required for the IND-CCA2 security proof of the relevant $FO^{\not\perp}$ transform [31] to be valid and to prevent key-recovery attacks in a key-pair-reuse setting [32].

As with the other code-based schemes, the best known attacks are based on information set decoding.

Performance. The quasi-cyclic structure of HQC enables small public-key and ciphertext sizes, although they are noticeably larger than the structured lattice KEMs. HQC ciphertexts and public keys are roughly 2.9 and 1.5 times the size of BIKE ciphertexts and public keys, respectively (see Tables 6 and 7).

Although the bandwidth of HQC exceeds that of BIKE, HQC's key generation and decapsulation are significantly faster than those of BIKE (see Tables 3 and 4). As a result, the performances of HQC and BIKE in applications are difficult to compare. Experiments on TLS 1.3 handshake performance under varying network conditions have revealed that HQC outperforms BIKE under ideal network conditions [33]. However, in the case of nonzero packet loss rates, BIKE outperforms HQC. In addition to the benchmarks included in the HQC submission for a hardware implementation, there have been several hardware implementation results published in the literature [34–36].

Significant events since Round 3. To address security and performance, HQC added a salt to mitigate multi-ciphertext attacks and switched to using implicit rejection for their FO transform. Additionally, several changes to the implementation were made to avoid timing attacks.

Overall assessment. NIST determined that HQC would provide a good complement to ML-KEM, since it is based on a different underlying security problem and still retains reasonable performance characteristics for general applications. The only other fourth-round candidate that could potentially serve this purpose was BIKE, which relies on similar code-based assumptions to HQC. Compared to BIKE, HQC has larger public key and ciphertext sizes but cheaper key generation and decryption. NIST was unable to make a definitive assessment as to which performance profile is better but found it likely that either performance profile would be acceptable for most general applications.

The decisive factor in favor of HQC relative to BIKE is HQC's stable DFR analysis. A sufficiently low DFR is required to achieve IND-CCA2 security, and there have been persistent uncertainties regarding BIKE's DFR. While DFR estimation techniques for BIKE have recently

[2] HQC's DFR analysis makes the simplifying assumption that the coordinates of $\mathbf{e}' = \mathbf{x} \cdot \mathbf{r}_2 - \mathbf{r}_1 \cdot \mathbf{y} + \mathbf{e}$ are independent variables. The HQC submission document [8] gives theoretical and experimental justifications for this assumption.

improved, previous inaccurate DFR estimates have resulted in BIKE being attacked as late as the fourth round, and BIKE would likely require post-selection tweaks to achieve IND-CCA2 security. In contrast, DFR estimates for all HQC parameter sets have remained stable throughout the NIST PQC Standardization Process. The IND-CCA2 security of HQC has not been successfully attacked since May 2020 when HQC discarded parameter sets targeting a higher DFR than $2^{-\lambda}$ for λ bits of security. NIST is confident that HQC as submitted provides a low enough DFR to achieve IND-CCA2 security.

3.2. BIKE

BIKE (Bit-Flipping Key Encapsulation) is a KEM based on binary linear quasi-cyclic moderate density parity check (QC-MDPC) codes [37]. The BIKE cryptosystem was initially designed for ephemeral key use but now claims to also support static key use.

Design. The binary linear QC-MDPC code $C(n,k)$ used in BIKE is constructed as follows. The secret key is a parity check matrix $H_{r\times 2r}$ for a quasi-cyclic moderate density parity check code composed of two circulant blocks, where r is prime and chosen so that x^{r-1} has only two irreducible factors modulo 2. Each row of H has Hamming weight $w \approx \sqrt{n}$, where $w \equiv 2 \mod 4$. All matrix operations in BIKE can be viewed as polynomial operations due to the isomorphism between the ring of $v \times v$ circulant matrices and the polynomial ring $\mathbb{F}_2[x]/(x^v+1)$ for any $v \in \mathbb{N}$. The secret key may then be thought of as a 1×2 module (h_0, h_1). The public key $H_{\text{pub}} = (1, h_0^{-1}h_1)$ is the secret key in systematic form, which is computed by multiplying H by h_0^{-1}.

The underlying BIKE PKE follows Niederreiter-style encryption. At a high level, a message is encoded as an error vector e of weight t, and the corresponding ciphertext is computed as $H_{\text{pub}}e^T$. Decryption is accomplished by multiplying the ciphertext by h_0 to produce the syndrome He^T and then using the recommended Black-Grey-Flip bit-flipping decoder [38] to recover e.

Security. The proof of IND-CPA security of the underlying PKE in the random oracle model (ROM) depends on the difficulty of solving the decisional QCSD and QCCF problems. The FO transform, as described in [31], is applied to the CPA-secure PKE to achieve a claimed IND-CCA2 KEM. The PKE must be δ-correct[3] for $\delta \leq 2^{-\lambda}$ to apply this transformation.

Iterative bit-flipping decoders for QC-MDPC codes are difficult to analyze in closed form, and the anticipated DFR is too low to compute directly. Moreover, the DFR of MDPC and LDPC codes under iterative decoding follows two regimes: a *waterfall* region in which decoding failures decrease rapidly followed by an *error floor* region in which decoding failures decrease at a much slower pace as the signal-to-noise ratio increases. Understanding the

[3] A KEM is δ-correct if the decapsulation fails (i.e., disagrees with encapsulation) with probability at most δ on average over all keys and messages. Similarly, a decoder will be δ-correct if its failure rate is at most δ on average when the input is drawn uniformly.

DFR of BIKE has remained an open problem during the fourth round. Analyzing the BIKE DFR has involved studying the impacts of *weak keys* and *near codewords* on decoding performance.

The first classification of weak keys for QC-MDPC codes was given in [39] and generalized in [40, 41]. Since these classes of weak keys have small cardinality, they were determined to have minimal impact on the overall BIKE DFR. A new class of weak keys was discovered [42] based on the *gathering property*. These weak keys were shown to cause an average DFR of a least 2^{-117} for BIKE level 1 parameters, defeating the IND-CCA2 security of BIKE.

The BIKE team studied the weak keys with the gathering property and found that the decoding failures were largely caused by incorrect flips happening early in the decoding process. Namely, bits not in error were incorrectly flipped during the first iterations of the decoder. To mitigate the effect of the gathering keys, the BIKE team introduced a new decoder known as BIKE-flip that sets a high bit-flipping threshold at the first decoding iteration and then gradually lowers the threshold throughout decoding [6]. Results indicated that the BIKE-flip decoder significantly reduced the impact of gathering keys, although this analysis was limited to classes of weak keys with a high enough DFR to be directly measured. Subsequently, a model introduced by [26] was able to predict variations in DFR based on the structure of a key. This would allow a modification of the BIKE key-generation algorithm in which keys that are not expected to have a typical value for the DFR are rejected.

Near codewords are error vectors of low weight (u) that map to syndromes of low weight (v) and are well-studied in the LDPC literature as impediments to the iterative decoding process. Moreover, these vectors are known to significantly contribute to *error floor* behavior. A particular class of *near codewords*, where $u = v = \frac{w}{2}$, was defined in [41] and shown to exist for BIKE. The impact of these *near codewords* on the decoding performance for BIKE was initially analyzed in [41] and further studied in [26]. In [26], a Markov model that tracked proximity to near codewords was used to predict the error floors for QC-MDPC codes under a generic iterative decoder. Results indicated that the error floor behavior in the BIKE DFR curves was dominated by convergence to these near codewords during failed decoding instances [26]. Furthermore, the model predicted that increasing BIKE security level 1 block lengths from $r = 12323$ to $r = 13477$ would result in a conservative DFR estimate of $2^{-129.5}$ for typical keys.

Performance. The sizes of BIKE's public keys and ciphertexts were roughly 70% and 30% of HQC's, respectively. However, BIKE's decapsulation and key-generation algorithms were roughly 5-6 times slower than HQC's, respectively. The performance of BIKE and HQC in applications was difficult to compare. Experiments on TLS 1.3 handshake performance under varying network conditions have revealed that HQC outperforms BIKE under ideal network conditions [33]. However, BIKE outperforms HQC when non-zero packet loss rates are introduced.

Significant events since Round 3. The BIKE specification was updated at the beginning of the fourth round and included a change from the previous approach of sampling fixed-weight vectors to a data-oblivious technique. This modification had no noticeable performance impacts but eliminated certain side-channel attacks. To offer more resistance against multi-target key attacks, BIKE's FO transform to attain IND-CCA2 security now includes a hash of part of the public key. As noted in the *Security* section above, a new decoder (BIKE-Flips) was used, which has better resilience to decryption failure for weak keys.

Overall assessment. NIST found that BIKE is a KEM that would complement ML-KEM well with respect to having a different underlying security problem and balanced performance characteristics. BIKE also offers smaller keys and ciphertexts than HQC. NIST reviewed several DFR analyses of BIKE, including recent results indicating that an approximate 9% increase in block size leads to a sufficiently low DFR for security level 1 parameters. Despite these promising results, NIST found the security analysis of HQC to be more mature and stable than that of BIKE. As such, NIST has not selected BIKE for standardization.

3.3. Classic McEliece

Design. Classic McEliece is a code-based KEM that uses binary Goppa codes in the Niederreiter variant of the McEliece cryptosystem combined with standard techniques to achieve IND-CCA2 security. Due to the use of Goppa codes, the KEM has perfect correctness.[4] It is a merger of the second-round submissions Classic McEliece and NTS-KEM. The original McEliece cryptosystem was published in [43] and was also based on binary Goppa codes.

Security. The Classic McEliece submission cites [44] and other results as giving a tight proof of the submitted KEM's IND-CCA2 security in the quantum random oracle model based on the assumption that the 1978 McEliece scheme provides one-way under chosen-plaintext attacks (OW-CPA) security. Confidence in the security of the 1978 scheme was mostly established based on the scheme's long history of surviving cryptanalysis with only minor changes in the complexity of the best-known attack. Alternatively, the security of the scheme could be established under the assumptions that row-reduced parity check matrices for the binary Goppa codes used by Classic McEliece are indistinguishable from row-reduced parity check matrices for random linear codes of the same dimensions and that the syndrome decoding problem is hard for random linear codes with those dimensions. The state of the art in cryptanalysis does not contradict these assumptions, although binary Goppa codes with very different dimensions from those used by the Classic McEliece submission have been shown to be distinguishable from random codes [45]. More recent work [46] has proposed a distinguisher that claims to asymptotically break the indistinguishabil-

[4] A perfectly correct KEM or PKE is one for which every ciphertext generated using the encapsulation/encryption function may be correctly decrypted using the decapsulation/decryption function. In contrast, some KEMs and PKEs have a very small decryption failure rate.

ity of Goppa codes with parameters that are similar to those used by Classic McEliece but that target a much higher security level.

A number of approaches to the cryptanalysis of Classic McEliece have been studied. The most effective known attacks and those used to set the parameters of Classic McEliece are information set decoding attacks, which are similar to the best-known attacks against BIKE and HQC. Unlike the other two schemes, information set decoding is only applicable to message recovery, not key recovery. These attacks ignore the structure of the binary code and seek to recover the error vector based on its low Hamming weight. These techniques originated with Prange's algorithm in 1962 [47] and have undergone a series of improvements [48–56]. However, the net effect of these improvements has been fairly modest, and most of the change in concrete security is due to improvements that were discovered more than 30 years ago. Quantum versions of information set decoding (ISD) algorithms have also been studied [57]. These results represent a generic Grover-based speedup of classical ISD algorithms and indicate that ISD algorithms can be sped up nearly as much as brute force search problems. In a multi-ciphertext setting, a further improvement [58] can reduce the cost of decoding a single ciphertext by a factor equal to approximately the square root of the number of ciphertexts.

Key-recovery attacks have also been studied, which attempt to find the private key by algebraic techniques or brute-force searches. Algebraic techniques have been used to break variants of McEliece based on other algebraic codes [59–63] or Goppa codes with additional structure imposed [64], but they appear to be significantly more costly than ISD for attacking the parameters submitted for Classic McEliece. Nonetheless, algebraic attacks that target the structure of Goppa codes and achieve either key recovery or a distinguisher from a random linear code have remained an active area of research [46, 65–69].

Performance. Classic McEliece has a very large public-key size and fairly slow key generation, which will likely make it undesirable in many common settings. However, its profile could have some advantages in settings where a public key is reused many times and does not need to be retransmitted for each new communication [70]. In particular, Classic McEliece has the smallest ciphertext sizes of any of the NIST PQC candidates.

Significant events since Round 3. At the beginning of the fourth round, the submission team introduced a modification to the FO transform to incorporate implicit rejection without plaintext confirmation. This tweak aimed to reduce the potential for patent concerns and simplify the specification and software code.

During the fourth round, there has been significant progress in cryptanalysis techniques that are applicable to key recovery and the related problem of distinguishing a Goppa code from a random linear code [46, 66–69]. While these techniques are still far from concretely affecting the security of the submitted parameter sets of Classic McEliece, they somewhat weaken the argument that the long-term security of Classic McEliece is guaranteed by its long history of cryptanalysis.

Additionally, during the third round, Classic McEliece was proposed to be added to the ISO/International Electrotechnical Commission (IEC) standard ISO/IEC 18033-2. This concurrent standardization effort remains active and ongoing.

Overall assessment. NIST remains confident in the security of Classic McEliece,[5] although recent progress in cryptanalysis somewhat undermines the case for treating it as an especially conservative choice. Its large public-key size makes Classic McEliece an unattractive choice for most common applications, but it offers an excellent performance profile for applications that are sensitive to ciphertext size, where public keys are rarely transmitted.

NIST does not find the case for standardizing Classic McEliece compelling, due to skepticism that it will see widespread use. In the event that Classic McEliece does become widely used through other standards, and that NIST remains confident in its security while also determining that there is sufficient need, NIST may develop a NIST standard based on the widely used version.

3.4. SIKE

Cryptographic schemes that are based on the hardness of the discrete logarithm problem on elliptic curves are known to be quantum-insecure because of Shor's algorithm. However, elliptic curves can be used in a different way to construct PKE and KEM protocols. An *isogeny* from one elliptic curve to another elliptic curve (over the same field) is a rational map that is also a group homomorphism. Given two isogenous curves E and E', efficiently constructing an isogeny from E to E' is generally unknown. The assumed hardness of finding an isogeny between two elliptic curves combined with the Diffie–Hellman model for key exchange enables the construction of a family of isogeny-based KEMs.

SIKE is a KEM based on isogenies of supersingular elliptic curves that follows and improves upon the construction known as Supersingular Isogeny Diffie–Hellman (SIDH) [72]. In SIKE, one party prepares a secret isogeny ϕ from a publicly known elliptic curve E_0 to a new curve E and computes the images of the generators of a known torsion subgroup (under ϕ) as the public key. This public key is then used to carry out a Diffie–Hellman procedure. The security of SIKE depends crucially on the assumption that it is infeasible for an adversary to compute the secret isogeny ϕ from public information.

However, in mid-2022, researchers showed that the secret isogeny ϕ can be efficiently recovered from the public key [17]. Attacks on SIKE were further improved and generalized by other researchers [25, 73], and the authors of SIKE have acknowledged the break [9]. Attempts to patch the vulnerabilities were ineffective or had weaknesses in some instances [74]. While these attacks were devastating for SIKE, they do not apply to many

[5] Independent estimates [56, 71] of the cost of information set decoding algorithms have long suggested that Classic McEliece's parameter sets (i.e., mceliece460896 and mceliece460896f) that claim Category 3 security fall short of their security target. However, NIST remains confident that these parameter sets at least meet the criteria for Category 2 security.

other isogeny-based cryptographic schemes. The attacks relied on the information provided by the image of a torsion subgroup in the SIKE public key, while other isogeny-based schemes do not utilize these auxiliary torsion points.

SIKE is an insecure KEM, and it has been eliminated from the NIST PQC project.

4. Conclusion

This report summarizes the evaluation criteria for selecting the fourth-round candidate algorithms, their basic designs, and their advantages and disadvantages. NIST greatly appreciates the participation in the NIST PQC Standardization Process. The announcement of the standardization of HQC marks the end of the fourth round, and also marks an end to the standardization process which began with the NIST Call for Proposals in 2016 [22]. We note that not all NIST PQC standardization is concluded, as NIST is also currently evaluating additional digital signatures [75].

NIST will create a draft standard based on HQC and post it for public comment. After the comments are adjudicated, NIST will publish a final version in approximately two years. The standardization of HQC will be the second PQC KEM after ML-KEM. NIST recently published draft SP 800-227, *Recommendations for Key-Encapsulation Mechanisms* [76], which describes the basic definitions, properties, and applications of KEMs. It also provides recommendations for implementing and using KEMs in a secure manner.

NIST plans to host another NIST PQC Standardization Conference in September 2025, with more details to be provided.

References

[1] Open quantum safe (OQS) algorithm performance visualizations. Available at https://openquantumsafe.org/benchmarking.

[2] Alagic G, Apon D, Cooper DA, Dang QH, Dang T, Kelsey JM, Lichtinger J, Liu YK, Miller CA, Moody D, Peralta R, Perlner RA, Robinson A, Smith-Tone D (2022) Status Report on the Third Round of the NIST Post-Quantum Cryptography Standardization Process. (National Institute of Standards and Technology, Gaithersburg, MD), NIST Interagency or Internal Report (IR) NIST IR 8413-upd1, Includes updates as of September 26, 2022. https://doi.org/10.6028/NIST.IR.8413-upd1

[3] National Institute of Standards and Technology (2015) SHA-3 Standard: Permutation-Based Hash and Extendable-Output Functions. (Department of Commerce, Washington, DC), Federal Information Processing Standards Publication (FIPS) NIST FIPS 202. https://doi.org/10.6028/NIST.FIPS.202

[4] National Institute of Standards and Technology (2024) Module-Lattic-Based Digital Signature Standard. (Department of Commerce, Washington, DC), Federal Information Processing Standards Publication (FIPS) NIST FIPS 204. https://doi.org/10.6028/NIST.FIPS.204

[5] National Institute of Standards and Technology (2024) Stateless Hash-Based Digital Signature Standard. (Department of Commerce, Washington, DC), Federal Information Processing Standards Publication (FIPS) NIST FIPS 205. https://doi.org/10.6028/NIST.FIPS.205

[6] Aragon N, Barreto PSLM, Bettaieb S, Bidoux L, Blazy O, Deneuville JC, Gaborit P, Ghosh S, Gueron S, Güneysu T, Melchor CA, Misoczk R, Persichetti E, Richter-Brockmann J, Sendrier N, Tillich JP, Vasseur V, Zémor G (2022) BIKE: Bit Flipping Key Encapsulation, 4th Round submission to the NIST's post-quantum cryptography standardization process. https://csrc.nist.gov/projects/post-quantum-cryptography/round-4-submissions.

[7] Bernstein DJ, Chou T, Cid C, Gilcher J, Lange T, Maram V, von Maurich I, Misoczki R, Niederhagen R, Persichetti E, Peters C, Sendrier N, Szefer J, Tjhai CJ, Tomlinson M, Wang W (2022) Classic McEliece algorithm specifications and supporting documentation, 4th Round submission to the NIST's post-quantum cryptography standardization process. https://csrc.nist.gov/projects/post-quantum-cryptography/round-4-submissions.

[8] Melchor CA, Aragon N, Bettaieb S, Bidoux L, Blazy O, Deneuville JC, Gaborit P, Persichetti E, Zémor G, Bos J, Dion A, Lacan J, Robert JM, Veron P (2022) HQC algorithm specifications and supporting documentation, 4th Round submission to the NIST's post-quantum cryptography standardization process. https://csrc.nist.gov/projects/post-quantum-cryptography/round-4-submissions.

[9] Jao D, Azarderakhsh R, Campagna M, Costello C, Feo LD, Hess B, Jalali A, Koziel B, LaMacchia B, Longa P, Naehrig M, Renes J, Soukharev V, Urbanik D, Pereira G, Karabina K, Hutchinson A (2022) Supersingular Isogeny Key Encapsulation, 4th Round

submission to the NIST's post-quantum cryptography standardization process. https://csrc.nist.gov/projects/post-quantum-cryptography/round-4-submissions.

[10] Chen L, Jordan S, Liu YK, Moody D, Peralta R, Perlner R, Smith-Tone D (2016) Report on post-quantum cryptography (National Institute of Standards and Technology, Gaithersburg, MD), NIST Interagency or Internal Report (IR) 8105. https://doi.org/10.6028/NIST.IR.8105

[11] National Institute of Standards and Technology (2016) Announcing request for nominations for public-key post-quantum cryptographic algorithms. *Federal Register* 81(244):92787–92788. https://federalregister.gov/a/2016-30615.

[12] Alagic G, Alperin-Sheriff J, Apon D, Cooper D, Dang Q, Liu YK, Miller C, Moody D, Peralta R, Perlner R, Robinson A, Smith-Tone D (2019) Status report on the first round of the NIST post-quantum cryptography standardization process (National Institute of Standards and Technology, Gaithersburg, MD), NIST Interagency or Internal Report (IR) 8240. https://doi.org/10.6028/NIST.IR.8240

[13] Alagic G, Alperin-Sheriff J, Apon D, Cooper D, Dang Q, Kelsey J, Liu YK, Miller C, Moody D, Peralta R, Perlner R, Robinson A, Smith-Tone D (2020) Status report on the second round of the NIST post-quantum cryptography standardization process (National Institute of Standards and Technology, Gaithersburg, MD), NIST Interagency or Internal Report (IR) 8309. https://doi.org/10.6028/NIST.IR.8309

[14] National Institute of Standards and Technology (2024) Module-Lattice-Based Key-Encapsulation Mechanism Standard. (Department of Commerce, Washington, DC), Federal Information Processing Standards Publication (FIPS) NIST FIPS 203. https://doi.org/10.6028/NIST.FIPS.203

[15] National Institute of Standards and Technology (2016) NIST post-quantum cryptography standardization. Available at https://csrc.nist.gov/projects/post-quantum-cryptography/post-quantum-cryptography-standardization.

[16] National Institute of Standards and Technology (2022) Call for Additional Digital Signature Schemes for the Post-Quantum Cryptography Standardization Process. Available at https://csrc.nist.gov/csrc/media/Projects/pqc-dig-sig/documents/call-for-proposals-dig-sig-sept-2022.pdf.

[17] Castryck W, Decru T (2023) An Efficient Key Recovery Attack on SIDH. *Advances in Cryptology – EUROCRYPT 2023*, eds Hazay C, Stam M (Springer Nature Switzerland, Cham), pp 423–447. https://doi.org/https://doi.org/10.1007/978-3-031-30589-4_15

[18] Kempf M, Gauder N, Jaeger B, Zirngibl J, Carle G (2024) A Quantum of QUIC: Dissecting Cryptography with Post-Quantum Insights. *IFIP Networking*, pp 195–203. https://doi.org/10.23919/IFIPNetworking62109.2024.10619916

[19] Sosnowski M, Wiedner F, Hauser E, Steger L, Schoinianakis D, Gallenmüller S, Carle G (2023) The performance of post-quantum TLS 1.3. *Companion of the 19th International Conference on emerging Networking EXperiments and Technologies*, pp 19–27. https://doi.org/https://doi.org/10.1145/3624354.3630585

[20] Henrich J, Heinemann A, Wiesmaier A, Schmitt N (2023) Performance impact of PQC KEMs on TLS 1.3 under varying network characteristics. *ISC 2023: 26th International Conference on Information Security*, eds Athanasopoulos E, Mennink B (Springer, Cham, Switzerland, Groningen, The Netherlands), *Lecture Notes in Computer Science*, Vol. 14411, pp 267–287. https://doi.org/10.1007/978-3-031-49187-0_14

[21] Müller J, Oupický J (2024) Post-quantum XML and SAML single sign-on. *Proceedings on Privacy Enhancing Technologies* 2024(4):525–543. https://doi.org/10.56553/popets-2024-0128

[22] National Institute of Standards and Technology (2016) Submission requirements and evaluation criteria for the post-quantum cryptography standardization process. Available at https://csrc.nist.gov/CSRC/media/Projects/Post-Quantum-Cryptography/documents/call-for-proposals-final-dec-2016.pdf.

[23] National Institute of Standards and Technology (2022) NIST post-quantum cryptography standardization round 4 submissions. Available at https://csrc.nist.gov/Projects/post-quantum-cryptography/round-4-submissions.

[24] Maino L, Martindale C (2022) An attack on SIDH with arbitrary starting curve, *Cryptology ePrint Archive preprint*. Available at https://ia.cr/2022/1026.

[25] Robert D (2023) Breaking SIDH in Polynomial Time. *Advances in Cryptology – EUROCRYPT 2023*, eds Hazay C, Stam M (Springer Nature Switzerland, Cham), pp 472–503. https://doi.org/https://doi.org/10.1007/978-3-031-30589-4_17

[26] Arpin S, Lau JB, Perlner R, Robinson A, Tillich JP, Vasseur V (2025) Error floor prediction with Markov models for QC-MDPC codes, Cryptology ePrint Archive, Paper 2025/153. Available at https://eprint.iacr.org/2025/153.

[27] Aguilar-Melchor C, Blazy O, Deneuville JC, Gaborit P, Zémor G (2018) Efficient encryption from random quasi-cyclic codes. *IEEE Transactions on Information Theory* 64(5):3927–3943. https://doi.org/https://doi.org/10.1109/TIT.2018.2804444

[28] Aragon N, Gaborit P, Z'emor G (2020) HQC-RMRS, an instantiation of the HQC encryption framework with a more efficient auxiliary error-correcting code. *ArXiv* abs/2005.10741.

[29] Regev O (2005) On lattices, learning with errors, random linear codes, and cryptography. *Proceedings of the Thirty-Seventh Annual ACM Symposium on Theory of Computing* STOC '05 (Association for Computing Machinery, New York, NY, USA), p 84–93. https://doi.org/10.1145/1060590.1060603

[30] Lyubashevsky V, Peikert C, Regev O (2010) On ideal lattices and learning with errors over rings. *Advances in Cryptology – EUROCRYPT 2010*, ed Gilbert H (Springer Berlin Heidelberg, Berlin, Heidelberg), pp 1–23. https://doi.org/https://doi.org/10.1007/978-3-642-13190-5_1

[31] Hofheinz D, Hövelmanns K, Kiltz E (2017) A modular analysis of the Fujisaki-Okamoto transformation. *Theory of Cryptography*, eds Kalai Y, Reyzin L (Springer International Publishing, Cham), pp 341–371. https://doi.org/https://doi.org/10.1007/978-3-319-70500-2_12

[32] Guo Q, Johansson T (2020) A new decryption failure attack against HQC. *Advances in Cryptology – ASIACRYPT 2020*, eds Moriai S, Wang H (Springer International Publishing, Cham), pp 353–382. https://doi.org/https://doi.org/10.1007/978-3-030-64837-4_12

[33] Henrich J, Heinemann A, Wiesmaier A, Schmitt N (2023) Performance Impact of PQC KEMs on TLS 1.3 Under Varying Network Characteristics. *Information Security*, eds Athanasopoulos E, Mennink B (Springer Nature Switzerland, Cham), pp 267–287. https://doi.org/https://doi.org/10.1007/978-3-031-49187-0_14

[34] Deshpande S, Xu C, Nawan M, Nawaz K, Szefer J (2023) Fast and efficient hardware implementation of HQC. *Selected Areas in Cryptography - SAC 2023 - 30th International Conference, Fredericton, Canada, August 14-18, 2023, Revised Selected Papers*, eds Carlet C, Mandal K, Rijmen V (Springer), *Lecture Notes in Computer Science*, Vol. 14201, pp 297–321. https://doi.org/10.1007/978-3-031-53368-6_15

[35] Li C, Song S, Tian J, Wang Z, Koç ÇK (2023) An efficient hardware design for fast implementation of HQC. *36th IEEE International System-on-Chip Conference, SOCC 2023, Santa Clara, CA, USA, September 5-8, 2023*, eds Becker J, Marshall A, Harbaum T, Ganguly A, Siddiqui F, McLaughlin K (IEEE), pp 1–6. https://doi.org/10.1109/SOCC58585.2023.10257054

[36] Antognazza F, Barenghi A, Pelosi G, Susella R (2024) A high efficiency hardware design for the post-quantum KEM HQC. *IEEE International Symposium on Hardware Oriented Security and Trust, HOST 2024, Tysons Corner, VA, USA, May 6-9, 2024* (IEEE), pp 431–441. https://doi.org/10.1109/HOST55342.2024.10545409

[37] Misoczki R, Tillich JP, Sendrier N, Barreto PSLM (2013) MDPC-McEliece: New McEliece variants from moderate density parity-check codes. *2013 IEEE International Symposium on Information Theory*, pp 2069–2073. https://doi.org/https://doi.org/10.1109/ISIT.2013.6620590

[38] Drucker N, Gueron S, Kostic D (2020) QC-MDPC decoders with several shades of gray. *Post-Quantum Cryptography*, eds Ding J, Tillich JP (Springer International Publishing, Cham), pp 35–50. https://doi.org/https://doi.org/10.1007/978-3-030-44223-1_3

[39] Drucker N, Gueron S, Kostic D (2020) On constant-time QC-MDPC decoders with negligible failure rate. *Code-Based Cryptography*, eds Baldi M, Persichetti E, Santini P (Springer International Publishing, Cham), pp 50–79. https://doi.org/https://doi.org/10.1007/978-3-030-54074-6_4

[40] Aydin N, Yildiz B, Uludag S (2020) A class of weak keys for the QC-MDPC cryptosystem. *Algebraic and Combinatorial Coding Theory 2020*, pp 1–4. https://doi.org/10.1109/ACCT51235.2020.9383383

[41] Vasseur V (2021) *Post-quantum cryptography: a study of the decoding of QC-MDPC codes*. Ph.D. thesis. Université de Paris, Paris, France.

[42] Wang T, Wang A, Wang X (2023) Exploring decryption failures of BIKE: New class of weak keys and key recovery attacks. *Advances in Cryptology – CRYPTO 2023*, eds Handschuh H, Lysyanskaya A (Springer Nature Switzerland, Cham), pp 70–100. https://doi.org/https://doi.org/10.1007/978-3-031-38548-3_3

[43] McEliece RJ (1978) A Public-Key Cryptosystem Based On Algebraic Coding Theory. *Deep Space Network Progress Report* 44:114–116.

[44] Bindel N, Hamburg M, Hövelmanns K, Hülsing A, Persichetti E (2019) Tighter proofs of CCA security in the quantum random oracle model. *Theory of Cryptography*, eds Hofheinz D, Rosen A (Springer International Publishing, Cham), pp 61–90. https://doi.org/https://doi.org/10.1007/978-3-030-36033-7_3

[45] Faugère JC, Gauthier-Umanã V, Otmani A, Perret L, Tillich JP (2011) A distinguisher for high rate McEliece cryptosystems. *2011 IEEE Information Theory Workshop*, pp 282–286. https://doi.org/https://doi.org/10.1109/ITW.2011.6089437

[46] Randriambololona H (2024) The syzygy distinguisher. *IACR Cryptol ePrint Arch* :1193Available at https://eprint.iacr.org/2024/1193.

[47] Prange E (1962) The use of information sets in decoding cyclic codes. *IRE Transactions on Information Theory* 8(5):5–9. https://doi.org/https://doi.org/10.1109/TIT.1962.1057777

[48] Stern J (1989) A method for finding codewords of small weight. *Coding Theory and Applications*, eds Cohen G, Wolfmann J (Springer Berlin Heidelberg, Berlin, Heidelberg), pp 106–113. https://doi.org/https://doi.org/10.1007/BFb0019850

[49] Dumer I (1991) On minimum distance decoding of linear codes. *Proc. 5th Joint Soviet-Swedish Int. Workshop Inform. Theory*, pp 50–52.

[50] May A, Meurer A, Thomae E (2011) Decoding random linear codes in $\tilde{O}(2^{0.054n})$. *Advances in Cryptology – ASIACRYPT 2011*, eds Lee DH, Wang X (Springer Berlin Heidelberg, Berlin, Heidelberg), pp 107–124. https://doi.org/https://doi.org/10.1007/978-3-642-25385-0_6

[51] Bernstein DJ, Lange T, Peters C (2011) Smaller decoding exponents: Ball-collision decoding. *Advances in Cryptology – CRYPTO 2011*, ed Rogaway P (Springer Berlin Heidelberg, Berlin, Heidelberg), pp 743–760. https://doi.org/https://doi.org/10.1007/978-3-642-22792-9_42

[52] Becker A, Joux A, May A, Meurer A (2012) Decoding random binary linear codes in $2^{n/20}$: How $1+1=0$ improves information set decoding. *Advances in Cryptology – EUROCRYPT 2012*, eds Pointcheval D, Johansson T (Springer Berlin Heidelberg, Berlin, Heidelberg), pp 520–536. https://doi.org/https://doi.org/10.1007/978-3-642-29011-4_31

[53] May A, Ozerov I (2015) On computing nearest neighbors with applications to decoding of binary linear codes. *Advances in Cryptology – EUROCRYPT 2015*, eds Oswald E, Fischlin M (Springer Berlin Heidelberg, Berlin, Heidelberg), pp 203–228. https://doi.org/https://doi.org/10.1007/978-3-662-46800-5_9

[54] Canto Torres R, Sendrier N (2016) Analysis of information set decoding for a sub-linear error weight. *Post-Quantum Cryptography*, ed Takagi T (Springer International Publishing, Cham), pp 144–161. https://doi.org/https://doi.org/10.1007/978-3-319-29360-8_10

[55] Both L, May A (2017) Optimizing BJMM with nearest neighbors: full decoding in $2^{2n/21}$ and McEliece security. *The Tenth International Workshop on Coding and Cryptogra-*

phy, pp –. Available at https://www.cits.ruhr-uni-bochum.de/imperia/md/content/may/paper/bjmm+.pdf.

[56] Guo Q, Johansson T, Nguyen V (2024) A new sieving-style information-set decoding algorithm. *IEEE Trans Inf Theory* 70(11):8303–8319. https://doi.org/10.1109/TIT.2024.3457150

[57] Kirshanova E (2018) Improved quantum information set decoding. *Post-Quantum Cryptography*, eds Lange T, Steinwandt R (Springer International Publishing, Cham), pp 507–527. https://doi.org/https://doi.org/10.1007/978-3-319-79063-3_24

[58] Sendrier N (2011) Decoding one out of many. *Post-Quantum Cryptography*, ed Yang BY (Springer Berlin Heidelberg, Berlin, Heidelberg), pp 51–67. https://doi.org/https://doi.org/10.1007/978-3-642-25405-5_4

[59] Sidelnikov VM, Shestakov SO (1992) On insecurity of cryptosystems based on generalized Reed-Solomon codes. *Discrete Mathematics and Applications* 2(4):439–444. https://doi.org/doi:10.1515/dma.1992.2.4.439

[60] Wieschebrink C (2010) Cryptanalysis of the Niederreiter public key scheme based on GRS subcodes. *Post-Quantum Cryptography*, ed Sendrier N (Springer Berlin Heidelberg, Berlin, Heidelberg), pp 61–72. https://doi.org/https://doi.org/10.1007/978-3-642-12929-2_5

[61] Couvreur A, Gaborit P, Gauthier-Umaña V, Otmani A, Tillich JP (2014) Distinguisher-based attacks on public-key cryptosystems using Reed—Solomon codes. *Designs, Codes and Cryptography* 73(2):641–666. https://doi.org/10.1007/s10623-014-9967-z

[62] Minder L, Shokrollahi A (2007) Cryptanalysis of the Sidelnikov cryptosystem. *Advances in Cryptology - EUROCRYPT 2007*, ed Naor M (Springer Berlin Heidelberg, Berlin, Heidelberg), pp 347–360. https://doi.org/https://doi.org/10.1007/978-3-540-72540-4_20

[63] Borodin MA, Chizhov IV (2014) Effective attack on the McEliece cryptosystem based on Reed-Muller codes. *Discrete Mathematics and Applications* 24(5):273–280. https://doi.org/10.1515/dma-2014-0024

[64] Faugère JC, Otmani A, Perret L, de Portzamparc F, Tillich JP (2015) Structural cryptanalysis of McEliece schemes with compact keys. *Designs, Codes and Cryptography* 79:87 – 112. https://doi.org/https://doi.org/10.1007/s10623-015-0036-z

[65] Couvreur A, Otmani A, Tillich J (2014) Polynomial time attack on wild McEliece over quadratic extensions. *Advances in Cryptology - EUROCRYPT 2014 - 33rd Annual International Conference on the Theory and Applications of Cryptographic Techniques, Copenhagen, Denmark, May 11-15, 2014. Proceedings*, eds Nguyen PQ, Oswald E (Springer), *Lecture Notes in Computer Science*, Vol. 8441, pp 17–39. https://doi.org/10.1007/978-3-642-55220-5_2

[66] Mora R, Tillich J (2023) On the dimension and structure of the square of the dual of a Goppa code. *Des Codes Cryptogr* 91(4):1351–1372. https://doi.org/10.1007/S10623-022-01153-W

[67] Bardet M, Mora R, Tillich J (2024) Polynomial time key-recovery attack on high rate random alternant codes. *IEEE Trans Inf Theory* 70(6):4492–4511. https://doi.org/10.1109/TIT.2023.3334592

[68] Couvreur A, Mora R, Tillich J (2023) A new approach based on quadratic forms to attack the mceliece cryptosystem. *Advances in Cryptology - ASIACRYPT 2023 - 29th International Conference on the Theory and Application of Cryptology and Information Security, Guangzhou, China, December 4-8, 2023, Proceedings, Part IV*, eds Guo J, Steinfeld R (Springer), *Lecture Notes in Computer Science*, Vol. 14441, pp 3–38. https://doi.org/10.1007/978-981-99-8730-6_1

[69] Mora R (2024) On the matrix code of quadratic relationships for a Goppa code. *Advances in Mathematics of Communications* 19. https://doi.org/10.3934/amc.2024026

[70] Hülsing A, Ning KC, Schwabe P, Weber FJ, Zimmermann PR (2021) Post-quantum wireguard. *2021 IEEE Symposium on Security and Privacy (SP)*, pp 304–321. https://doi.org/https://doi.org/10.1109/SP40001.2021.00030

[71] Esser A, Bellini E (2022) Syndrome decoding estimator. *Public-Key Cryptography – PKC 2022*, eds Hanaoka G, Shikata J, Watanabe Y (Springer International Publishing, Cham), pp 112–141. https://doi.org/https://doi.org/10.1007/978-3-030-97121-2_5

[72] Jao D, De Feo L (2011) Towards quantum-resistant cryptosystems from supersingular elliptic curve isogenies. *Post-Quantum Cryptography*, ed Yang BY (Springer Berlin Heidelberg, Berlin, Heidelberg), pp 19–34. https://doi.org/https://doi.org/10.1007/978-3-642-25405-5_2

[73] Maino L, Martindale C, Panny L, Pope G, Wesolowski B (2023) A Direct Key Recovery Attack on SIDH. *Advances in Cryptology – EUROCRYPT 2023*, eds Hazay C, Stam M (Springer Nature Switzerland, Cham), pp 448–471. https://doi.org/https://doi.org/10.1007/978-3-031-30589-4_16

[74] Castryck W, Vercauteren F (2023) A Polynomial Time Attack on Instances of M-SIDH and FESTA. *Advances in Cryptology – ASIACRYPT 2023*, eds Guo J, Steinfeld R (Springer Nature Singapore, Singapore), pp 127–156. https://doi.org/https://doi.org/10.1007/978-981-99-8739-9_5

[75] Alagic G, Bros M, Ciadoux P, Cooper D, Dang Q, Dang T, Kelsey J, Lichtinger J, Liu YK, Miller C, Moody D, Peralta R, Perlner R, Robinson A, Silberg H, Smith-Tone D, Waller N (2024) Status Report on the First Round of the Additional Digital Signature Schemes for the NIST Post-Quantum Cryptography Standardization Process. (National Institute of Standards and Technology, Gaithersburg, MD), NIST Interagency or Internal Report (IR) NIST IR 8528. https://doi.org/10.6028/NIST.IR.8528

[76] Alagic G, Barker EB, Chen L, Moody D, Robinson A, Silberg H, Waller N (2025) Recommendation for key-encapsulation mechanisms (U.S. Department of Commerce, Washington, D.C.), Special Publication 800-227 (Initial Public Draft). https://doi.org/10.6028/NIST.SP.800-227.ipd

Appendix A. List of Symbols, Abbreviations, and Acronyms

BIKE	Bit-Flipping Key Encapsulation
CCA	Chosen Ciphertext Attack
CPA	Chosen Plaintext Attack
DFR	Decryption Failure Rate
FIPS	Federal Information Processing Standards
FO	Fujisaki Okamoto
IEC	International Electrotechnical Commission
IND-CCA2	Indistinguishability under Adaptive Chosen-Ciphertext Attack
IND-CPA	Indistinguishability under Chosen-Plaintext Attack
IPsec	Internet Protocol Security
ISD	Information Set Decoding
ISO	International Organization for Standardization
HQC	Hamming Quasi-Cyclic
KEM	Key-Encapsulation Mechanism
LWE	Learning With Errors
ML-KEM	Module Lattice-Based Key-Encapsulation Mechanism (based on Kyber)
NIST	National Institute of Standards and Technology
NIST IR	NIST Interagency or Internal Report
OW-CPA	One-Way under Chosen Plaintext Attack
PKE	Public-Key Encryption
PQC	Post-Quantum Cryptography
QC-MDPC	Quasi-Cyclic Moderate Density Parity Check
QCCF	Quasi-Cyclic Codeword Finding
QCSD	Quasi-Cyclic Syndrome Decoding
QUIC	Quick UDP Internet Connections
RMRS	Reed-Muller Reed-Solomon
ROM	Random Oracle Model
SAML SSO	Security Assertion Markup Language Single Sign-On

SIKE	Supersingular Isogeny Key Encapsulation
SIDH	Supersingular Isogeny Diffie-Hellman
SP	Special Publication
SSH	Secure Shell
TLS	Transport Layer Security
VPN	Virtual Private Network
XML	Extensible Markup Language