IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received September 14, 2020, accepted September 23, 2020, date of publication September 28, 2020,
date of current version October 8, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3026989

Modified pqsigRM: RM Code-Based Signature Scheme

YONGWOO LEE[1], (Graduate Student Member, IEEE), WIJIK LEE[2], YOUNG SIK KIM[3], (Member, IEEE), AND JONG-SEON NO[1], (Fellow, IEEE)

[1]Department of Electrical and Computer Engineering, INMC, Seoul National University, Seoul 08826, South Korea
[2]Samsung Electronics, Hwaseong-si 18448, South Korea
[3]Department of Information and Communication Engineering, Chosun University, Gwangju 61452, South Korea

Corresponding author: Young Sik Kim (iamyskim@chosun.ac.kr)

This work was supported by the Institute for Information and communications Technology Promotion (IITP) grant funded by the Korean Government (MSIP), Research on Lightweight Post-Quantum Cryptosystems for the IoT and Cloud Computing, under Grant R-20160229-002941.

ABSTRACT We present a novel code-based signature scheme called modified pqsigRM. This scheme is based on a modified Reed–Muller (RM) code, which reduces the signing complexity and key size compared with existing code-based signature schemes. In fact, it strengthens pqsigRM submitted to NIST for post-quantum cryptography standardization. The proposed scheme has the advantage of the pqsigRM decoder and uses public codes that are more difficult to distinguish from random codes. We use $(U, U + V)$-codes with the high-dimensional hull to overcome the disadvantages of code-based schemes. The proposed decoder samples from coset elements with small Hamming weight for any given syndrome and efficiently finds such an element. Using a modified RM code, the proposed signature scheme resists various known attacks on RM-code-based cryptography. For 128 bits of classical security, the signature size is 4096 bits, and the public key size is less than 1 MB.

INDEX TERMS Cryptography, digital signatures, error correction codes, post-quantum cryptography (PQC), Reed-Muller (RM) codes.

I. INTRODUCTION

Recently, code-based cryptographic algorithms have been extensively studied in post-quantum cryptography (PQC). Code-based cryptography is based on the syndrome decoding problem and its variants. The syndrome decoding problem is to find a vector $\mathbf{e}$ satisfying $\mathbf{He}^T = \mathbf{s}^T$ and $\mathrm{wt}(\mathbf{e}) \leq w$, where $\mathbf{H}$ is a parity check matrix of a random (n, k) code, $\mathbf{s}$ is a random syndrome vector, w is a small value, and $\mathrm{wt}(\mathbf{e})$ denotes the Hamming weight of a vector $\mathbf{e}$. Berlekamp and McEliece first proved the hardness of the syndrome decoding problem [19] and McEliece proposed a cryptosystem based on Goppa codes [22].

Courtois, Finiasz, and Sendrier proposed the CFS signature scheme [2], which is a code-based signature scheme using a full-domain hash (FDH) approach. In this scheme, $t!$ hashes, and decodings are required on average to sign a message when an (n, k) Goppa code with error correction capability t is used. It is proposed to use high-rate Goppa codes, which have relatively small error correction capability $t = \frac{n-k}{\log n}$, to reduce the signing time. Therefore, it has a large signing complexity and certain drawbacks in terms of parameter scaling. Moreover, it has been shown in [4] that high-rate Goppa codes can be distinguished from random codes. This falsifies the assumption of existential unforgeability under a chosen message attack (EUF-CMA) security proof in [17], which is based on the indistinguishability of Goppa codes. Although Morozov *et al.* claimed to have proved the strong EUF-CMA security of the CFS signature scheme without the indistinguishability of Goppa codes [18], the large key size and expensive signing remain as drawbacks.

There are several variants of the CFS signature scheme, such as signature schemes using LDGM codes [7] and blockwise-triangular secret key [9]. To find a signature with small Hamming weight, the scheme in [7] uses a sparse coset element added to a codeword with small Hamming weight. Even though this is efficient and has a small key size, an attack algorithm was presented in [6]. An attack algorithm for the signature scheme using a blockwise-triangular secret key was also proposed [8].

The associate editor coordinating the review of this manuscript and approving it for publication was Sedat Akleylek.

The Kabatianskii-Krouk-Smeets (KKS) signature scheme [30] and its variants [31], [32] take a different approach than CFS signature scheme. However, owing to the attack proposed in [36], these are considered (at best) to be one-time signature schemes. Moreover, from the attacks in [37], it is known that the parameters in the KKS scheme and its variants should be carefully chosen.

SURF is a variant of CFS signature scheme using $(U, U+V)$-codes [29]. SURF uses $(n, k_U + k_V)$ binary codes defined by $\{(\mathbf{u}|\mathbf{u}+\mathbf{v})|\mathbf{u} \in U, \mathbf{v} \in V\}$, where U and V are $(n/2, k_U)$ and $(n/2, k_V)$ random binary codes, respectively. A variant of the Prange decoder is applied to SURF to find an error vector with a small Hamming weight. The security of SURF is based on the decoding-one-out-of-many (DOOM) problem, in which a solution for the syndrome decoding problem is sought in the presence of several syndromes. Unfortunately, as it has been demonstrated that the hull of any $(U, U+V)$-code is highly probable to be a two-repetition code when U and V are random binary codes [29], the hull of the public key can be used for key attacks on SURF. In the recently proposed signature scheme, Wave [35], the generalized ternary $(U, U+V)$-codes are used instead of binary codes as they efficiently resist the hull attack in [29]. Moreover, finding errors with large Hamming weight for the given syndrome allows small parameters. A tighter security reduction using rejection sampling and preimage samplable functions [34] was proved in [35].

In this paper, a new code-based signature scheme using binary codes with a $(U, U+V)$-code as its subcode is proposed. For two linear codes $\mathcal{C}_1$ and $\mathcal{C}_2$, $\mathcal{C}_2$ is called a subcode of $\mathcal{C}_1$ if all codewords in $\mathcal{C}_2$ are in $\mathcal{C}_1$. The subcode used in the proposed signature scheme is a binary $(U, U+V)$-code, where U and V are obtained by modifying the RM codes. We design V and $U^{\perp}$ to have a sufficient number of common codewords, where $U^{\perp}$ denotes the dual code of U. Using the relationships between U and V, it is shown that the proposed signature scheme resists the attack for $(U, U+V)$-codes in [29]. Further, an efficient and randomized decoding algorithm is proposed. This algorithm makes it possible to reduce the key size and signature length. As the codes in the proposed signature scheme are a modification of RM codes, the decoding algorithm makes use of the recursive structure. The proposed signature scheme is an improvement of pqsigRM [1] submitted to NIST for PQC standardization, and it resolves the weaknesses of early versions of pqsigRM by modifying the public code. Moreover, we ensure the distinguishability of the public code of the proposed signature scheme.

The rest of this paper is organized as follows. In Section II, we discuss FDH code-based signature schemes and RM codes. A new code-based signature scheme, called modified pqsigRM using modified RM code is proposed in Section III. In Section IV, the security of the proposed signature scheme is analyzed, and it is proved that the signature scheme is EUF-CMA secure. The proof is based on two ad-hoc problems and the assumption that these are hard. The two problems are analyzed in Section V. Considering state-of-the-art attacks, we suggest security parameters in Section VI. The paper is concluded in Section VII.

II. PRELIMINARIES

A. BASIC NOTATION

A Vector is denoted in boldface in the form of a column vector. $(\mathbf{x}_0|\mathbf{x}_1)$ denotes the concatenation of two vectors $\mathbf{x}_0$ and $\mathbf{x}_1$. For example, $h(\mathbf{m}|r)$ means the hash function h with input $(\mathbf{m}|r)$, where $(\mathbf{m}|r)$ represents the concatenation of binary representation of vector $\mathbf{m}$ and a random value r. Matrices are denoted by a boldfaced capital letter, for example, $\mathbf{A}$. Matrix multiplication is denoted by $\cdot$ or can be omitted when it is unnecessary. Codes and probability distributions are denoted in calligraphic fonts, for example $\mathcal{C}$, and it can be distinguished by context. $\mathbf{x}^{\sigma}$ denotes that a vector $\mathbf{x}$ is permuted by a permutation σ, for example, $\mathbf{x}^{\sigma} = (x_1, x_3, x_2, x_0)$, where $\mathbf{x} = (x_0, x_1, x_2, x_3)$ and $\sigma = (1, 3, 2, 0)$.

B. CFS SIGNATURE SCHEME

CFS signature scheme is an algorithm that applies the FDH methodology to the Niederreiter cryptosystem. The CFS signature scheme is based on Goppa codes, as McEliece cryptosystem. A summary of CFS signature scheme is given in Algorithm 1.

As described in Algorithm 1, the signing process iterates until a decodable syndrome is obtained. The probability that a given random syndrome can be decoded is $\frac{\sum_{i=0}^{t}\binom{n}{i}}{2^{n-k}} \simeq \frac{1}{t!}$. Hence, the error correction capability $t = \frac{n-k}{\log n}$ should be sufficiently small to reduce the number of iterations. Thus, the high-rate Goppa codes should be used. Regarding the key size, the complexity of the decoding attack on the CFS signature scheme is known to be a small power of the key size, namely, $\approx$ keysize$^{t/2}$. Hence, the key size should be fairly large to meet a certain security level. In summary, the CFS signature scheme is insecure and inefficient owing to the use of Goppa codes.

C. REED–MULLER CODES AND RECURSIVE DECODING

RM codes were introduced by Muller [23] and Reed [24]. and its decoding algorithm, so-called recursive decoding, was proposed in [10]. There are various definitions of RM codes, but we adopt a recursive definition here as recursive decoding is defined by using this structure. An RM code $\mathrm{RM}_{(r,m)}$ is a linear binary $(n = 2^m, k = \sum_{i=0}^{r}\binom{m}{i})$ code, where r and m are integers. $\mathrm{RM}_{(r,m)}$ is defined as $\mathrm{RM}_{(r,m)} := \{(\mathbf{u}|\mathbf{u}+\mathbf{v})|\mathbf{u} \in \mathrm{RM}_{(r,m-1)}, \mathbf{v} \in \mathrm{RM}_{(r-1,m-1)}\}$, where $\mathrm{RM}_{(0,m)} := \{(0, \ldots, 0), (1, \ldots, 1)\}$ with code length 2^m and $\mathrm{RM}_{(m,m)} := \mathbb{F}_2^{2^m}$. This is the well-known Plotkin's construction, and its generator matrix is given by

$$\mathbf{G}_{(r,m)} = \begin{bmatrix} \mathbf{G}_{(r,m-1)} & \mathbf{G}_{(r,m-1)} \\ \mathbf{0} & \mathbf{G}_{(r-1,m-1)} \end{bmatrix},$$

where $\mathbf{G}_{(r,m)}$ is the generator matrix of $\mathrm{RM}_{(r,m)}$.

Recursive decoding is a soft-decision decoding algorithm that depends on the recursive structure of the RM codes;

Algorithm 1 CFS Signature Scheme [2]

Key generation:

$\mathbf{H}$ is the parity check matrix of an (n, k) Goppa code
The error correction capability t is $\frac{n-k}{\log n}$
$\mathbf{S}$ and $\mathbf{Q}$ are an $(n-k) \times (n-k)$ scrambler matrix and $n \times n$ permutation matrix, respectively
Secret key: $\mathbf{H}$, $\mathbf{S}$, and $\mathbf{Q}$
Public key: $\mathbf{H}' \leftarrow \mathbf{SHQ}$

Signing:

$\mathbf{m}$ is a message to be signed
$i \leftarrow 1$
Do
$i \leftarrow i + 1$
Find syndrome $\mathbf{s} \leftarrow h(h(\mathbf{m})|i)$
Compute $\mathbf{s}' \leftarrow \mathbf{S}^{-1}\mathbf{s}$
Until a decodable syndrome $\mathbf{s}'$ is found
Find an error vector satisfying $\mathbf{He}'^T \leftarrow \mathbf{s}'$
* Compute $\mathbf{e}^T \leftarrow \mathbf{Q}^{-1}\mathbf{e}'^T$, and then the signature is $(\mathbf{m}, \mathbf{e}, i)$

Verification:

Check wt$(\mathbf{e}) \leq t$ and $\mathbf{H}'\mathbf{e}^T = h(h(\mathbf{m})|i)$
If True, then return ACCEPT; else, return REJECT

Algorithm 2 Recursive Decoding of RM Code [10]

function RecursiveDecoding($\mathbf{y}$, r, m)
if $r = 0$ **then**
Perform MD decoding on RM$(0, m)$
else if $r = m$ **then**
Perform MD decoding on RM(r, r)
else
$(\mathbf{y}'|\mathbf{y}'') \leftarrow \mathbf{y}$
$\mathbf{y}^{\mathbf{v}} = \mathbf{y}' \cdot \mathbf{y}''$
$\hat{\mathbf{v}} \leftarrow$ RecursiveDecoding($\mathbf{y}^{\mathbf{v}}$, $r-1$, $m-1$)
$\mathbf{y}^{\mathbf{u}} \leftarrow (\mathbf{y}' + \mathbf{y}'' \cdot \hat{\mathbf{v}})/2$
$\hat{\mathbf{u}} \leftarrow$ RecursiveDecoding($\mathbf{y}^{\mathbf{u}}$, r, $m-1$)
Output $(\hat{\mathbf{u}}|\hat{\mathbf{u}} \cdot \hat{\mathbf{v}})$
end if
end function

it is described in detail in Algorithm 2, where $\mathbf{y}' \cdot \mathbf{y}''$ denotes the component-wise multiplication of the vectors $\mathbf{y}'$ and $\mathbf{y}''$. In recursive decoding, a binary symbol $a \in \{0, 1\}$ is mapped onto $(-1)^a$, and it is assumed that all codewords belong to $\{-1, 1\}^n$.

First, $\mathbf{y}''$ (the second half of the received vector $\mathbf{y}$) is component-wisely multiplied by $\mathbf{y}'$ (the first half of the received vector). Then, a codeword from $\mathrm{RM}_{(r,m-1)}$ (i.e., $\mathbf{u}$) is removed from $\mathbf{y}''$ as it is both in $\mathbf{y}'$ and $\mathbf{y}''$, and then only $\mathbf{v}$ and the error vector remain. This is regarded as a codeword of $\mathrm{RM}_{(r-1,m-1)}$ added to an error vector and is referred to as $\hat{\mathbf{v}}$. Using $\hat{\mathbf{v}}$, we can remove the codeword of $\mathrm{RM}_{(r-1,m-1)}$ from the second half of the received vector. $\mathbf{y}'$ is then added to $\mathbf{y}'' \cdot \hat{\mathbf{v}}$, and the sum is divided by 2. This is regarded as a codeword of $\mathrm{RM}_{(r,m-1)}$ added to the error vector, and then decoding is performed. Recursively, the received vector is further divided into sub-vectors of length $n/4$, $n/8$, etc. Finally, we reach $\mathrm{RM}_{(m,m)}$ or $\mathrm{RM}_{(0,m)}$, then the division terminates and the minimum distance (MD) decoding of $\mathrm{RM}_{(m,m)}$ or $\mathrm{RM}_{(0,m)}$, which is trivial, is performed. The decoding for the entire code is performed by reconstructing these results into $(U, U+V)$ form.

III. MODIFIED REED–MULLER CODES AND PROPOSED SIGNATURE SCHEME

In this section, we propose new codes, their decoder, and a signature scheme that uses these codes and decoders. The proposed code essentially has a $(U, U+V)$-code as its subcode, and recursively, U and V are also $(U, U+V)$-codes. This recursive structure allows the decoding of any given vector in $\mathbb{F}_2^n$. Then, we can find an error vector with small Hamming weight for any given syndrome corresponding to the received vector. Starting from $(U, U+V)$-codes, we replace certain rows and append random rows on the generator matrix of $(U, U+V)$-codes. Thus, these codes are no longer $(U, U+V)$-codes. However, they have a $(U, U+V)$-subcode and can use the decoder for $(U, U+V)$-codes.

A. PARTIAL PERMUTATION OF GENERATOR MATRIX AND MODIFIED REED–MULLER CODES

New codes named modified RM codes are defined in this section. We first present the core of the proposed codes, which is a $(U, U+V)$-code. Subsequently, we describe which rows are replaced or appended to the generator matrix. The rationale for these operations is provided in Section V.

For a code $\mathcal{C}$, we define its hull by the intersection of the code and its dual, in other words, $hull(\mathcal{C}) = \mathcal{C} \cap \mathcal{C}^{\perp}$. The proposed $(U, U+V)$-code is designed to have a high-dimensional hull, where $dim(U^{\perp} \cap V)$, dimenstion of $U^{\perp} \cap V$, is large. In general, for a $(U, U+V)$-code $\mathcal{C}$, a codeword $(\mathbf{u}|\mathbf{u}+\mathbf{v}) \in hull(\mathcal{C})$ satisfies $\mathbf{v} = \mathbf{u}^{\perp}$ and $\mathbf{u}+\mathbf{v} = \mathbf{v}^{\perp}$, where $\mathbf{u} \in U$ and $\mathbf{v} \in V$. Hence, when $U^{\perp} \cap V = \{\mathbf{0}\}$, $hull(\mathcal{C})$ has only $(\mathbf{u}|\mathbf{u})$ codewords, and this may reveal the secret key. To avoid this, the proposed code is designed so that $dim(U^{\perp} \cap V)$ is large.

For convenience, we focus on the generator matrix. First, we construct the generator matrix $\mathbf{G}_{(r,m)}$ of an RM code and then permute its submatrices. An example is shown in Figure 1, where σ_p^1 and σ_p^2 denote two independent partial permutations that randomly permute only p out of $n/4$ columns. As will be explained in Section VI-B, p is related to the decoding performance. To generate σ_p^1 and σ_p^2, p column indices are randomly selected from the index set $\{0, 1, \ldots, n/4-1\}$, and the selected indices are randomly permuted, whereas the others are not. Then, σ_p^1 is used to permute the submatrices corresponding to $\mathbf{G}_{(r,m-2)}$'s in the first $dim(\mathrm{RM}_{(r,m-2)})$ rows, and σ_p^2 is used to permute the submatrix corresponding to $\mathbf{G}_{(r-2,m-2)}$ in the last

$\mathbf{G}_{(r,m-2)}^{\sigma_p^1}$	$\mathbf{G}_{(r,m-2)}^{\sigma_p^1}$	$\mathbf{G}_{(r,m-2)}^{\sigma_p^1}$	$\mathbf{G}_{(r,m-2)}^{\sigma_p^1}$
0	$\mathbf{G}_{(r-1,m-2)}$	0	$\mathbf{G}_{(r-1,m-2)}$
0	0	$\mathbf{G}_{(r-1,m-2)}$	$\mathbf{G}_{(r-1,m-2)}$
0	0	0	$\mathbf{G}_{(r-2,m-2)}^{\sigma_p^2}$

FIGURE 1. Generator matrix of partially permuted RM code with parameter (r, m).

$dim(\mathrm{RM}_{(r-2,m-2)})$ rows, as shown in Figure 1. The codes generated by the generator matrix in Figure 1 are called partially permuted RM codes. It should be noted that, unlike in the case of code-based cryptographic algorithms, we permute submatrices of the generator matrix rather than the entire matrix here. We note that the entire matrix should also be permuted to design a signature scheme. This will be discussed on the key generation in Section III-C.

$dim(U^{\perp} \cap V)$ is large for the following reasons. Let $\mathbf{G}_U$ and $\mathbf{G}_V$ denote the generator matrices of U and V, respectively:

$$\mathbf{G}_U = \begin{bmatrix} \mathbf{G}_{(r,m-2)}^{\sigma_p^1} & \mathbf{G}_{(r,m-2)}^{\sigma_p^1} \\ \mathbf{0} & \mathbf{G}_{(r-1,m-2)} \end{bmatrix},$$

$$\mathbf{G}_V = \begin{bmatrix} \mathbf{G}_{(r-1,m-2)} & \mathbf{G}_{(r-1,m-2)} \\ \mathbf{0} & \mathbf{G}_{(r-2,m-2)}^{\sigma_p^2} \end{bmatrix}.$$

Then, the generator matrix of the dual code of U is

$$\mathbf{G}_U^{\perp} = \begin{bmatrix} \mathbf{G}_{(r,m-2)}^{\perp\sigma_p^1} & \mathbf{0} \\ \mathbf{G}_{(r-1,m-2)}^{\perp} & \mathbf{G}_{(r-1,m-2)}^{\perp} \end{bmatrix}.$$

Thus, $U^{\perp} \cap V$ has a subcode that is the intersection of the codewords generated by $\begin{bmatrix} \mathbf{G}_{(r-1,m-2)} & \mathbf{G}_{(r-1,m-2)} \end{bmatrix}$ and the codewords generated by $\begin{bmatrix} \mathbf{G}_{(r-1,m-2)}^{\perp} & \mathbf{G}_{(r-1,m-2)}^{\perp} \end{bmatrix}$. Its dimension is $\min(dim(\mathrm{RM}_{(r-1,m-2)}), dim(\mathrm{RM}_{(m-r-2,m-2)}))$, as the dual of $\mathrm{RM}_{(r,m)}$ is equal to $\mathrm{RM}_{(m-r-1,m)}$ and $\mathrm{RM}_{(r',m)} \subseteq \mathrm{RM}_{(r,m)}$, where $r' \leq r$.

With the partially permuted RM codes, the received vector and the syndrome have the same parity, causing the signature leak. Thus, the generator matrix in Figure 1 should be further modified. That is, some rows are replaced with repetitions of random codewords and random rows are appended to the generator matrix. Considering $\mathbf{G}_U$, it is also an $(U, U+V)$-code, which can similarly be divided into (permuted) $(U, U+V)$-codes. By repeating this process 2^{m-r} times, the rows of the partially permuted RM code consist of the 2^{m-r} repeated generator matrices of $\mathrm{RM}_{(r,r)}$, which are $2^r \times 2^r$ identity matrices. Then, $\mathrm{RM}_{(r,r)}$ is replaced by a repeated random $(2^r, k_{rep})$ code such that its dual code has at least one non-zero codeword with odd Hamming weight.

We now append random independent rows to the generator matrix. One row to be appended is a random codeword of the dual code. This should be independent of the existing rows; i.e., it should not belong to the hull of the code. Furthermore, it should be verified that the hull has codewords with Hamming weight that is not a multiple of four as a result of appending this row. The others are k_{app} random independent vectors including at least one vector of odd Hamming weight. These k_{app} vectors are independent of the partially permuted RM codes and independent of each other.

After all these modifications, the resulting code is called a modified RM code. An example of its generator matrix is given in Figure 2.

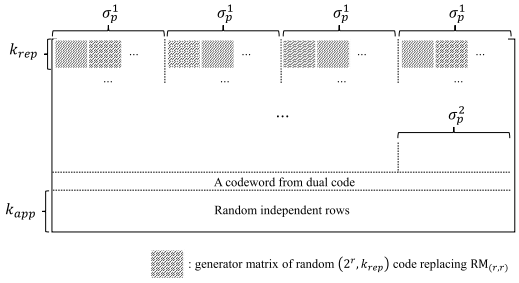


FIGURE 2. Generator matrix of modified RM code.

B. DECODING OF MODIFIED REED–MULLER CODES

Unlike the Niederreiter cryptosystem and CFS signature scheme, it is required to find an error vector whose Hamming weight is greater than the error correction capability. Hence, there may exist several solutions $\mathbf{e}$ satisfying $\mathbf{He}^T = \mathbf{s}^T$ and $\mathrm{wt}(\mathbf{e}) \leq w$ for a given syndrome $\mathbf{s}$. Such decoding can be achieved by the modified Prange decoder using the $(U, U+V)$ structure, as in the signature schemes in [29], [35]. However, in this section, a new decoder is proposed that uses the recursive structure of the subcode of modified RM codes and it achieves better performance than the modified Prange decoder. In other words, it finds error vectors whose Hamming weights are less than the result in [29]. This results in the smaller parameters, considering attacks as in [28].

In addition to the decoding performance, a major difference between the proposed decoder and the modified Prange decoder is their input. The input of the modified Prange decoder used in [35] and [29] is a syndrome vector. In contrast, the input of the proposed decoder is an n dimensional vector $\mathbf{r}$ satisfying $\mathbf{Hr}^T = \mathbf{s}$, which is called received vector in coding theory, and the decoder outputs codewords close to the received vector. An error vector with a small Hamming weight is obtained by subtracting the output from the received vector. Even if two different received vectors in the same coset are given, the proposed decoder can return different outputs. Besides, as the input of the decoder is a random received vector, decoding can be performed even if random rows are appended to the generator matrix.

As stated in the previous section, random rows (one from the dual code and the others being k_{app} independent random vectors) are appended to the generator matrix of the partially

permuted RM codes. Let $\mathcal{C}_{app}$ be the code spanned by the added $k_{app}+1$ rows. The number of codewords increases by $2^{k_{app}+1}$ times when rows are appended by adding codewords of $\mathcal{C}_{app}$ to each $(U, U+V)$-codeword. Choosing a codeword of $\mathcal{C}_{app}$ (including $\mathbf{0}$), subtracting it from the received vector $\mathbf{r}$, decoding it, and adding the subtracted codewords back is the decoding process when rows are appended. Thus, the code is decodable even if arbitrary random codes are appended to its generator matrix.

Hence, it suffices to explain the decoding algorithm for the $(U, U+V)$-subcode of a modified RM code. This decoding basically follows the recursive decoding of RM codes [10]. The difference is the partial permutation and the replacement of $\mathrm{RM}_{(r,r)}$. Considering the decoding proposed in [10], we have $\mathbf{c} = (\mathbf{u}|\mathbf{u}+\mathbf{v})$ for all $\mathbf{c} \in \mathrm{RM}_{(r,m)}$, where $\mathbf{u} \in \mathrm{RM}_{(r,m-1)}$ and $\mathbf{v} \in \mathrm{RM}_{(r-1,m-1)}$. $\mathrm{RM}_{(r,m-1)}$ and $\mathrm{RM}_{(r-1,m-1)}$ are also $(U, U+V)$-codes, except for $r=0$ or $r=m$. Here, if the code corresponding to $\mathbf{u}$ or $\mathbf{v}$ is replaced with a code other than the RM code and the decoding of the replaced code can be performed appropriately, the entire code $\mathbf{c}$ can also be decoded [15].

When the subcode of the RM code is replaced with its permutation, the entire code can also be decoded by slightly modifying the recursive decoding. Moreover, no decoding failure occurs because the recursion eventually reaches $\mathrm{RM}_{(0,m')}$, $\mathrm{RM}_{(r',r')}$, or the $(2^r, k_{rep})$ code to replace $\mathrm{RM}_{(r,r)}$ and there exists polynomial-time MD decoder for these codes. Even the $(2^r, k_{rep})$ random code is MD decodable in constant time because it is a small code. To handle partial permutations, when the code is decodable, it uses the fact that the permutation is always decodable if the permutation is known. Depermutation and decoding followed by permutation is the decoding process for permuted codes.

In general, the output distribution of decoding is crucial for security. Thus, we also propose a randomized decoding method, the output of which is almost uniformly distributed. Using the algorithm described above, a random decoder can easily be designed. Algorithm 3 summarizes the randomized decoding. It is easy to find a received vector (regardless of its Hamming weight) for any given syndrome; a coset element corresponding to the syndrome is randomly selected. This is given to the decoder as an input. Finally, the decoder finds a different error vector with a small Hamming weight for different inputs.

Algorithm 3 Decoding for Modified RM Code

```
function Decode(s; H)
    r ← Prange(H, s)
    while True do
        r ← r+ random codeword
        c ← ModDec(r, r, m)
        if wt(r + c) ≤ w then
            Output r + c
        end if
    end while
end function

function ModDec(y, r, m)
    y ← y^{σ^{-1}}
    if r = 0 then
        Output MD decoding on RM(0, m)
    else if r = m then
        Output MD decoding on RM(r, r)
         or replaced (2^r, k_rep) code
    else
        (y'|y'') ← y
        y^v = y' · y''
        v̂ ← ModDec(y^v, r − 1, m − 1)
        y^u ← (y' + y'' · v̂)/2
        û ← ModDec(y^u, r, m − 1)
        y ← (û|û · v̂)
    end if
    Output y^σ
end function
```

*σ is σ_p^1 or σ_p^2 for permuted block and identity, otherwise.

C. PROPOSED SIGNATURE SCHEME

Herein, the proposed modified pqsigRM signature scheme using the codes in the previous section is presented. Its decoding algorithm is presented in Section III-B.

1) KEY GENERATION

Let $\mathbf{G}$ be the generator matrix of a modified (n, k) RM code, and $\mathbf{H}$ be the parity check matrix. Let $\mathbf{S}$ be an $(n-k) \times (n-k)$ random non-singular matrix and $\mathbf{Q}$ be an $n \times n$ random permutation matrix. Then, the public key is $\mathbf{H}' = \mathbf{SHQ}$, and the secret keys are $\mathbf{H}$, $\mathbf{S}$, and $\mathbf{Q}$.

2) SIGNING

To sign a given message $\mathbf{m}$, we randomly select a coin $\mathbf{i}$ from $\{0,1\}^{\lambda_0}$. A binary vector $\mathbf{s} = h(h(\mathbf{m}|\mathbf{H}')|\mathbf{i})$ is calculated, where $h : \{0,1\}^* \to \{0,1\}^{n-k}$ is a cryptographic hash function. Our goal is to find the error vector $\mathbf{e}$ satisfying $\mathbf{H}'\mathbf{e}^T = \mathbf{SHQe}^T = \mathbf{s}$. Let $\mathbf{s}' = \mathbf{S}^{-1}\mathbf{m}$.

Performing the decoding as in Algorithm 3, we find an error vector $\mathbf{e}'$ satisfying $\mathbf{He}'^T = \mathbf{s}'$. If $\mathrm{wt}(\mathbf{e}') \le w$, we compute $\mathbf{e}^T = \mathbf{Q}^{-1}\mathbf{e}'^T$, and the signature is then given as $(\mathbf{m}, \mathbf{e}, \mathbf{i})$.

3) VERIFICATION

If $\mathrm{wt}(\mathbf{e}) \le w$ and $\mathbf{H}'\mathbf{e}^T = h(h(\mathbf{m}|\mathbf{H}')|\mathbf{i})$, we return ACCEPT; otherwise, we return REJECT.

The key generation, signing, and verification processes are summarized in Algorithm 4. For simplicity, let $\mathbf{H}$ represent all the secrets such as partial permutations σ_p^1 and σ_p^2, appended rows, and replaced codes. It should be noted that in the signing process, we choose a random coset element and perform ModDec(·). As ModDec(·) returns different outputs for different inputs even in the same coset, we can achieve randomized decoding. The output distribution of this randomized decoding output is analyzed in

Section V. We add a salt λ_0 to obtain a tight security proof.

Algorithm 4 Modified pqsigRM Signature Scheme

Key Generation:

Using σ_p^1 and σ_p^2, generate a partially permuted generator matrix $\mathbf{G}$

Generate $\mathbf{H}$ from $\mathbf{G}$

Generate $\mathbf{S}$ and $\mathbf{Q}$

Compute $\mathbf{H}' \leftarrow \mathbf{SHQ}$

Secret key: $\mathbf{H}$, $\mathbf{S}$, $\mathbf{Q}$

Public key: $\mathbf{H}'$

Signing:

$\mathbf{m}$ is a message to be signed

$\mathbf{i} \hookleftarrow \{0,1\}^{\lambda_0}$

Find syndrome $\mathbf{s} \leftarrow h(h(\mathbf{m}|\mathbf{H}')|\mathbf{i})$

$\mathbf{s}'^T \leftarrow \mathbf{S}^{-1}\mathbf{s}^T$

Perform decoding $\mathbf{e}' \leftarrow \text{Decode}(\mathbf{s}; \mathbf{H})$

* Compute $\mathbf{e}^T \leftarrow \mathbf{Q}^{-1}\mathbf{e}'^T$, and then the signature is $(\mathbf{m}, \mathbf{e}, \mathbf{i})$

Verification:

Check $\text{wt}(\mathbf{e}) \le w \wedge \mathbf{H}'\mathbf{e}^T = h(h(\mathbf{m}|\mathbf{H}')|\mathbf{i})$

If True, then return ACCEPT; else, return REJECT

IV. SECURITY ANALYSIS OF MODIFIED pqsigRM

In this section, the security of the proposed modified pqsigRM will be analyzed. We will consider the best-known algorithms for solving DOOM. Thereafter, we will discuss the resistance of the proposed signature scheme against key substitution attacks. Finally, it will be proved that the modified pqsigRM is EUF-CMA secure.

As the public key of the proposed signature scheme is a modification of an RM code, one may consider key recovery attacks on RM codes, such as Minder and Shokrollahi [13] and Chizhov and Borodin [12] attacks, as well as square code attacks [11]. However, owing to the partial permutation as well as the appending and replacement of codewords in the generator matrix, these attacks cannot be adopted here. Table 1 shows the comparison between the proposed modified pqsigRM and the original pqsigRM.

TABLE 1. Comparison of the proposed modified pqsigRM and the original pqsigRM.

	modified pqsigRM	original pqsigRM [1]
key generation method	partial permutation, row appending and replacement	column puncturing and insertion
randomized decoding	yes	no
attack	none	finding puncturing with hull

A. DECODING ONE OUT OF MANY

Information set decoding is a brute-force attack method that finds an error vector $\mathbf{e}$ such that $\mathbf{He}^T = \mathbf{s}$ and $\text{wt}(\mathbf{e}) \le w$, where Stern improved the attack complexity in [14]. It has been extensively studied, and Dumer's algorithm [38] as well as more involved variants in [39], [40] have been proposed.

In the variants of the CFS signature scheme, there are several hash queries. Therefore, to launch a forgery attack, it suffices to find an error vector with small Hamming weight for any of the syndromes. Hence, the decoding problem DOOM given below is adequate for tight security proof. The usual FDH proof for existential forgery using syndrome decoding would require a work factor $\ge q_{\mathcal{H}} \cdot 2^{\lambda}$, where $q_{\mathcal{H}} \le 2^{\lambda}$ is the number of hash queries. However, with DOOM, the work factor is required to be $\ge 2^{\lambda}$. Although the work factor of DOOM is greater than that of syndrome decoding, it provides tighter bounds for security.

Problem 1 (DOOM):

Instance: *A parity check matrix* $\mathbf{H} \in \mathbb{F}_2^{(n-k)\times n}$ *of an* (n, k) *linear code, syndromes* $\mathbf{s}_1, \mathbf{s}_2, \cdots, \mathbf{s}_q \in \mathbb{F}_2^{n-k}$, *and an integer* w.

Output: $(\mathbf{e}, i) \in \mathbb{F}_2^n \times [1, q]$ *such that* $\text{wt}(\mathbf{e}) \le w$ *and* $\mathbf{He}^T = \mathbf{s}_i^T$.

We consider the case in which the adversary has q instances and $M = \max(1, \binom{n}{w}/2^{n-k})$ solutions for each instance. Of course, in our case, w is not small, and thus M is $\binom{n}{w}/2^{n-k}$. In [28], the work factor of solving DOOM is given as

$$WF_q^M = \min_{p,l}\left(\frac{C_q(p,l)}{\mathcal{P}_{qM}(p,l)}\right),$$

where

$$C_q(p,l) = \max\left(\sqrt{q\binom{k+l}{p}}, \frac{q\binom{k+l}{p}}{2^l}\right), q \le \binom{k+l}{p}$$

is the complexity of solving the DOOM problem using Dumer's algorithm and

$$\mathcal{P}_{qM}(p,l) = 1 - \left(1 - \frac{\binom{n-k-l}{w-p}\binom{k+l}{p}}{\binom{n}{w}}\right)^{qM}$$

is the success probability. This work factor is the reference for choosing the parameters of the signature scheme. Although advanced algorithms for information set decoding can be adapted to DOOM to reduce complexity, this has not yet been conducted. The proposed signature scheme is designed to use codes with a high-dimensional hull. Hence, the attacker can exploit this. However, to our knowledge, there is no algorithm for information set decoding or DOOM that considers this.

B. SECURITY AGAINST KEY SUBSTITUTION ATTACKS

In a key substitution attack, the adversary attempts to find a valid key that is different from the correct key and can be used for signature verification. If the adversary knows the secret key and the public key corresponding to a message–signature pair, we have a weak-key substitution attack, whereas

if the adversary knows only the public key, we have a strong-key substitution attack. Both polynomial-time weak- and strong-key substitution attacks on the CFS signature scheme were proposed in [21]. A modification of the CFS scheme that resists such attacks was also proposed in [21]. In this modification, the syndrome $\mathbf{s}$ is generated by hashing the message, counter, and public key, rather than hashing only the message and counter. It has been demonstrated that this modified CFS signature scheme is secure against key substitution attacks [18]. In the modified pqsigRM, the syndrome is given as $\mathbf{s} = h(h(\mathbf{m}|\mathbf{H}')|\mathbf{i})$, and thus it is also secure against key substitution attacks.

C. EUF-CMA SECURITY

Here, we prove the EUF-CMA security of the modified pqsigRM. The methods presented below are adapted from the EUF-CMA security proof of SURF and Wave [29], [35]. It should be noted that although a key attack for SURF is presented in [29], its proof technique is valid and generally applicable. The proof is essentially the same except for the code used for the key and the decoding algorithm for signing.

1) BASIC TECHNIQUES FOR EUF-CMA SECURITY PROOF

EUF-CMA is a widely used attack model against signature schemes. In the security reduction task, EUF-CMA is viewed as a game played between an adversary and a challenger. The public key PK, hash oracle $\mathcal{H}$, and signing oracle Σ are given to a $(t, q_{\mathcal{H}}, q_{\Sigma}, \epsilon)$-adversary $\mathcal{A}$, where $\mathcal{A}$ can query at most $q_{\mathcal{H}}$ hash values and q_{Σ} signatures for inputs of its own choice. Within a maximum computation time t, $\mathcal{A}$ attempts to find a valid message–signature pair $(\mathbf{m}^*, \sigma^*)$. $\mathcal{A}$ wins the game if Verifying$(\mathbf{m}^*, \sigma^*, PK) = 1$ and σ^* has not been provided by Σ; otherwise, the challenger wins the game. The winning probability of the $(t, q_{\mathcal{H}}, q_{\Sigma}, \epsilon)$-adversary is at least ϵ.

Definition 1 (EUF-CMA Security): Let $\mathcal{S}$ be a signature scheme. We define the EUF-CMA success probability against $\mathcal{S}$ as

$$Succ_{\mathcal{S}}^{EUF-CMA}(t, q_{\mathcal{H}}, q_{\Sigma}) := \max(\epsilon | \exists (t, q_{\mathcal{H}}, q_{\Sigma}, \epsilon)\text{-}adversary).$$

The signature scheme $\mathcal{S}$ is called $(t, q_{\mathcal{H}}, q_{\Sigma})$-secure in EUF-CMA if the above success probability is a negligible function of the security parameter λ.

We use the statistical and computational distance as basic metrics.

Definition 2 (Statistical Distance): The statistical distance between two discrete probability distributions $\mathcal{D}^0$ and $\mathcal{D}^1$ over the same space $\mathcal{E}$ is defined as

$$\rho(\mathcal{D}^0, \mathcal{D}^1) := \frac{1}{2} \sum_{x \in \mathcal{E}} |\mathcal{D}^0(x) - \mathcal{D}^1(x)|.$$

Proposition 1 [29]: Let $(\mathcal{D}_1^0, \ldots, \mathcal{D}_n^0)$ and $(\mathcal{D}_1^1, \ldots, \mathcal{D}_n^1)$ be two n-tuples of discrete probability distributions over the same space. For all $n \geq 0$, we have

$$\rho(\mathcal{D}_1^0 \otimes \cdots \otimes \mathcal{D}_n^0, \mathcal{D}_1^1 \otimes \cdots \otimes \mathcal{D}_n^1) \leq \sum_{i=1}^{n} \rho(\mathcal{D}_i^0, \mathcal{D}_i^1).$$

Definition 3 (Computational Distance and Indistinguishability): The computational distance between two distributions $\mathcal{D}^0$ and $\mathcal{D}^1$ in time t is

$$\rho_c(\mathcal{D}^0, \mathcal{D}^1) := \frac{1}{2} \max_{|\mathcal{A}| \leq t} \left(Adv^{\mathcal{D}^0, \mathcal{D}^1}(\mathcal{A}) \right),$$

where $|\mathcal{A}|$ denotes the running time of $\mathcal{A}$, and $Adv^{\mathcal{D}^0, \mathcal{D}^1}$ is the advantage of distinguisher $\mathcal{A}$, which returns $b \in \{0, 1\}$ against $\mathcal{D}^0$ and $\mathcal{D}^1$:

$$Adv^{\mathcal{D}^0, \mathcal{D}^1} := \mathbb{P}_{\xi \sim \mathcal{D}^0}(\mathcal{A}(\xi) \text{ outputs } 1) - \mathbb{P}_{\xi \sim \mathcal{D}^1}(\mathcal{A}(\xi) \text{ outputs } 1).$$

The EUF-CMA security of the modified pqsigRM is reduced to the *modified RM code distinguishing problem* and *DOOM with high-dimensional hull*, which are defined as follows.

Problem 2 (Modified RM Code Distinguishing Problem):

Instance: *A code $\mathcal{C}$ with high-dimensional hull.*

Output: *A bit $b \in \{0, 1\}$, where $b = 1$ if $\mathcal{C}$ is a permutation of the modified RM code; otherwise, $b = 0$.*

Problem 3 (DOOM With High-Dimensional Hull):

Instance: *A parity check matrix $\mathbf{H}' \in \mathbb{F}_2^{(n-k) \times n}$ of an (n, k) code with high-dimensional hull, syndromes $\mathbf{s}_1, \mathbf{s}_2, \cdots, \mathbf{s}_q \in \mathbb{F}_n^{(n-k)}$, and an integer w.*

Output: *$(\mathbf{e}, i) \in \mathbb{F}_2^n \times [1, q]$ such that $\mathrm{wt}(\mathbf{e}) \leq w$ and $\mathbf{H}\mathbf{e}^T = \mathbf{s}_i^T$.*

Definition 4 (One-Wayness of DOOM With High-Dimensional Hull): We define the success of an algorithm $\mathcal{A}$ against DOOM with high-dimensional hull and parameters n, k, q, w as

$$Succ^{n,k,q,w}(\mathcal{A}) = \mathbb{P}(\mathcal{A}(\mathbf{H}, \mathbf{s}_1, \ldots, \mathbf{s}_q) \text{ is a solution of Problem 3}),$$

where $\mathbf{H}$ is chosen uniformly from the parity check matrix of (n, k) codes with a high-dimensional hull, $\mathbf{s}_i$ is chosen uniformly in $\mathbb{F}_2^{n-k}$, and the probability is taken over these choices and the internal coin of algorithm $\mathcal{A}$. The computational success of breaking DOOM with a high-dimensional hull in time t is defined by

$$Succ_{DOOMHull}^{n,k,q,w}(t) = \max_{|\mathcal{A}| \leq t} \left(Succ^{n,k,q,w}(\mathcal{A}) \right).$$

We assume here that the probability is negligible (as a function of λ) for the parameters given in Table 2.

We will discuss these problems in greater detail in Section V. It is worth noting that there are sufficiently many codes with high-dimensional hull for the parameters given in Tables 2 and 4 [25].

TABLE 2. Parameters for each security level.

λ (security)	80	128	256
(r, m)	(5,11)	(6,12)	(6,13)
n	2048	4096	8192
k	1025	2511	4097
w	325	495	1370
k_{rep}	30	62	62
k_{app}	2	2	2
p (recommended)	$\geqslant$130	$\geqslant$386	$\geqslant$562
Signature length (bits)	2048	4096	8192
Public key size (MB)	0.249	0.773	3.99
$\log_2$ WF	80	128	256

2) PROOF OF EUF-CMA SECURITY

Let $\mathcal{S}_{pqsigRM}$ denote the proposed modified pqsigRM. The following definitions as well as the theorem and its proof are adopted from those in [29], [35].

Definition 5 (Challenger Procedures in the EUF-CMA Game): The challenger procedures in the EUF-CMA game corresponding to $\mathcal{S}_{pqsigRM}$ are defined as follows:

proc Init(λ)	proc Hash($\mathbf{m}$, $\mathbf{i}$)
$(PK, SK) \leftarrow \mathrm{Gen}(1^\lambda)$ $\mathbf{H}' \leftarrow PK$ $(\mathbf{H}, \mathbf{S}, \mathbf{Q}) \leftarrow SK$ return $\mathbf{H}'$	return $h(\mathbf{m}, \mathbf{i})$
proc Sign($\mathbf{m}$)	**proc Finalize($\mathbf{m}, \mathbf{e}, \mathbf{i}$)**
$\mathbf{i} \hookleftarrow \{0,1\}^{\lambda_0}$ $\mathbf{s} \leftarrow \mathrm{Hash}(\mathbf{m}, \mathbf{i})$ $\mathbf{e} \leftarrow \mathrm{Decode}(\mathbf{S}^{-1}\mathbf{s}^T; \mathbf{H})$ return $(\mathbf{eQ}, \mathbf{i})$	$\mathbf{s} \leftarrow \mathrm{Hash}(\mathbf{m}, \mathbf{i})$ return $\mathbf{H}'\mathbf{e}^T = \mathbf{S}^T \wedge wt(\mathbf{e}) = w$

We note that the procedures in Definition 5 simplify Algorithm 4. We can now modify the security reduction in [29], [35] and prove the EUF-CMA security of the modified pqsigRM as follows.

Theorem 1 (Security Reduction): Let $Succ^{EUF-CMA}_{\mathcal{S}_{pqsigRM}}(t, q_\mathcal{H}, q_\Sigma)$ be the success probability of the EUF-CMA game corresponding to $\mathcal{S}_{pqsigRM}$ for time t when the number of queries to the hash oracle (resp. signing oracle) is $q_\mathcal{H}$ (resp. q_Σ). Then, in the random oracle model, we have for all t

$$\begin{aligned} &Succ^{EUF-CMA}_{\mathcal{S}_{pqsigRM}}(t, q_\mathcal{H}, q_\Sigma) \\ &\quad \leq 2Succ^{n,k,q,w}_{DOOMHull}(t_c) + q_\mathcal{H}\mathbb{E}_{\mathbf{H}'}\left(\rho(\mathcal{D}_w^{\mathbf{H}'}, \mathcal{U}_s)\right) \\ &\qquad + q_\Sigma\rho(\mathcal{D}_w, \mathcal{U}_w) + \rho_c(\mathcal{D}_{pub}, \mathcal{D}_{rand})(t_c) + \frac{1}{2^\lambda}, \end{aligned}$$

where $t_c = t + O(q_\mathcal{H} \cdot n^2)$, $\mathcal{D}_w^{\mathbf{H}'}$ is the distribution of the syndromes $\mathbf{H}'\mathbf{e}^T$ when $\mathbf{e}$ is drawn uniformly from the binary vectors of weight w, $\mathcal{U}_s$ is the uniform distribution over $\mathbb{F}_2^{n-k}$, $\mathcal{D}_w$ is the distribution of the decoding result of Algorithm 3, $\mathcal{U}_w$ is the uniform distribution over the binary vectors of weight w, $\mathcal{D}_{rand}$ is the uniform distribution over the random codes with high-dimensional hull, and $\mathcal{D}_{pub}$ is the uniform distribution over the public keys of modified pqsigRM.

Proof: Let $\mathcal{A}$ be a $(t, q_\mathcal{H}, q_\Sigma, \epsilon)$-adversary against $\mathcal{S}_{pqsigRM}$, and let $(\mathbf{H}_0, \mathbf{s}_1, \ldots, \mathbf{s}_{q_\mathcal{H}})$ be a random instance of DOOM with high-dimensional hull for the parameters $n, k, q_\mathcal{H}$, and w. We stress that $\mathbf{s}_1, \ldots, \mathbf{s}_{q_\mathcal{H}}$ are random independent vectors of $\mathbb{F}_2^{n-k}$. Let $\mathbb{P}(S_i)$ denote the probability that $\mathcal{A}$ wins Game i.

Game 0 is the EUF-CMA game for $\mathcal{S}_{pqsigRM}$.

Game 1 is the same as Game 0 except for the following failure event F: There is a collision in a signature query. From the difference lemma in [41], we have

$$\mathbb{P}(S_1) \leq \mathbb{P}(S_0) + \mathbb{P}(F). \tag{1}$$

The following lemma is from [35].

Lemma 2: For $\lambda_0 = \lambda + 2\log_2(q_\mathcal{H})$, we have $\mathbb{P}(F) \leq \frac{1}{\lambda}$.

Game 2 is obtained from Game 1 by changing `Hash` and `Sign` as follows, where S_w denotes the set of vectors with Hamming weight w in $\mathbb{F}_2^n$:

proc Hash($\mathbf{m}$, $\mathbf{i}$)	proc Sign($\mathbf{m}$)
if $\mathbf{i} \in L_\mathbf{m}$ $\mathbf{e}_{\mathbf{m},\mathbf{i}} \hookleftarrow S_w$ return $\mathbf{H}'\mathbf{e}_{\mathbf{m},\mathbf{i}}^T$ else $j \leftarrow j + 1$ return $\mathbf{s}_j$	$\mathbf{i} \leftarrow L_\mathbf{m}$.next() $\mathbf{s} \leftarrow \mathrm{Hash}(\mathbf{m}, \mathbf{i})$ $\mathbf{e} \leftarrow \mathrm{Decode}(\mathbf{S}^{-1}\mathbf{s}^T; \mathbf{H})$ return $(\mathbf{eQ}, \mathbf{i})$

Index j is initialized to 0 in the `Init` procedure. We introduce the list $L_\mathbf{m}$, which contains $q_\mathcal{H}$ random elements of $\mathbb{F}_2^{\lambda_0}$ for each message $\mathbf{m}$. The list is sufficiently large so that all queries are satisfied. The `Hash` procedure returns $\mathbf{H}'\mathbf{e}_{\mathbf{m},\mathbf{r}}^T$ if and only if $\mathbf{i} \in L_\mathbf{m}$; otherwise, it returns $\mathbf{s}_j$. The `Sign` process is unchanged unless $\mathbf{i} \in L_\mathbf{m}$.

The statistical distance between the syndromes generated by matrix $\mathbf{H}'$ and the uniform distribution over $\mathbb{F}_2^{n-k}$ is $\rho(\mathcal{D}_w^{\mathbf{H}'}, \mathcal{U}_s)$. This is the difference between `Hash` in Game 1 and Game 2 when $\mathbf{i} \in L_\mathbf{m}$. There are at most $q_\mathcal{H}$ such instances. Thus, by Proposition 1, it follows that

$$\mathbb{P}(S_2) \leq \mathbb{P}(S_1) + q_\mathcal{H}\mathbb{E}_{\mathbf{H}'}\left(\rho(\mathcal{D}_w^{\mathbf{H}'}, \mathcal{U}_s)\right). \tag{2}$$

Game 3 is obtained from Game 2 by replacing Decode with $\mathbf{e}_{\mathbf{m},\mathbf{i}}$ in `Sign` procedure as follows:

Game 3

proc Sign($\mathbf{m}$)
$\mathbf{i} \leftarrow L_\mathbf{m}$.next()
$\mathbf{s} \leftarrow \mathrm{Hash}(\mathbf{m}, \mathbf{i})$
$\mathbf{e} \leftarrow \mathbf{e}_{\mathbf{m},\mathbf{i}}$
return $(\mathbf{e}, \mathbf{i})$

Game 5

proc Finalize($\mathbf{m}, \mathbf{e}, \mathbf{i}$)
$\mathbf{s} \leftarrow \mathrm{Hash}(\mathbf{m}, \mathbf{i})$
$b \leftarrow \mathbf{H}'\mathbf{e}^T = \mathbf{S}^T \wedge wt(\mathbf{e}) = w$
return $b \wedge (\mathbf{i} \notin L_\mathbf{m})$

$\mathbf{e}$ is drawn according to the proposed decoding algorithm Decode in Game 2, whereas it is now drawn according to the uniform distribution $\mathcal{U}_w$. By Proposition 1, we have

$$\mathbb{P}(S_3) \leq \mathbb{P}(S_2) + q_\Sigma\rho(\mathcal{D}_w, \mathcal{U}_w). \tag{3}$$

Game 4 is the game in which $\mathbf{H}'$ is replaced with $\mathbf{H}_0$. This implies that the adversary is forced to construct a solution for DOOM with high-dimensional hull. Here, if a difference between Game 3 and Game 4 is detected, then this yields a distinguisher between $\mathcal{D}_{pub}$ and $\mathcal{D}_{rand}$. According to [29], the cost to call `Hash` does not exceed $O(n^2)$, and thus the running

time of the challenger is $t_c = t + O(q_{\mathcal{H}} \cdot n^2)$. Therefore, we have

$$\mathbb{P}(S_4) \leq \mathbb{P}(S_3) + \rho_c(\mathcal{D}_{pub}, \mathcal{D}_{rand})(t_c). \tag{4}$$

Game 5 is modified in `Finalize`. The success of Game 5 implies $\mathbf{i} \notin L_{\mathbf{m}}$ and the success of Game 4. A valid forgery $\mathbf{m}^*$ has never been queried by `Sign`, and the adversary has never accessed $L_{\mathbf{m}^*}$. As there are q_Σ signing queries, we have

$$\mathbb{P}(S_5) = (1 - 2^{\lambda_0})^{q_\Sigma} \mathbb{P}(S_4).$$

Moreover, $(1 - 2^{\lambda_0})^{q_\Sigma} \geq \frac{1}{2}$ because we assumed $\lambda_0 = \lambda + 2\log_2(q_\Sigma)$. Thus, this can be simplified to

$$\mathbb{P}(S_5) \geq \frac{1}{2}\mathbb{P}(S_4). \tag{5}$$

$\mathbb{P}(S_5)$ is the probability that $\mathcal{A}$ returns a solution for DOOM with high-dimensional hull, which yields

$$\mathbb{P}(S_4) \leq 2Succ^{n,k,q,w}_{DOOMHull}(t_c). \tag{6}$$

Combining (1)–(6) concludes the proof. □

V. INDISTINGUISHABILITY OF CODE AND SIGNATURE IN THE PROPOSED SCHEME

It is challenging to prove the hardness of distinguishing a public code of a code-based cryptographic algorithm from a random code. As it is difficult to prove the hardness of distinguishing the public code from a random code, several cryptographic algorithms are designed by assuming it. In this section, we will consider possible attack algorithms and consider the difficulty of distinguishing the public code and signatures. Moreover, the difficulty of distinguishing signatures from random errors is also analyzed.

A. MODIFICATIONS OF PUBLIC CODE

For successful decoding of any received vector, a $(U, U+V)$-code should be used in the modified RM codes. To resist the attack on $(U, U+V)$-codes proposed in [29], we design a code with high-dimensional hull. Generally, the expected dimension of the hull of a random code is $O(1)$, which is smaller than d with probability $\geq 1 - O(d)$ [25]. This is a difference between random and public codes. However, there is currently no algorithm for solving the syndrome decoding problem by taking advantage of the hull. We consider that a high-dimensional hull is not a significant drawback unless the hull has a certain structure that may reveal the secret. Moreover, in [25], it is demonstrated that there are a large number of codes with the high-dimensional hull. Hence, we can expect the one-wayness of DOOM with the high-dimensional hull as in Definition 4.

Cryptanalysis using hulls is widely used in code-based cryptography. However, this is valid if the hull has a specific structure that allows information leakage about the secret key. Therefore, using only the fact that the dimension of the hull is large, it is difficult to distinguish whether the code is public or random code with the high-dimensional hull.

The EUF-CMA security proof requires the indistinguishability between public and random codes, i.e., $\rho_c(\mathcal{D}_{pub}, \mathcal{D}_{rand})(t_c)$ should be negligible. We will discuss the design methodology and how these modifications can ensure indistinguishability.

Considering the key recovery attack in [29], a $(U, U+V)$-code used in code-based crypto-algorithms should have a high-dimensional hull for security. Even though the public code of the proposed signature scheme is not a $(U, U+V)$-code, it should contain a $(U, U+V)$ subcode for efficient decoding.

The attack on SURF in [29] uses the fact that for any $(U, U+V)$-code, the hull of the public code is highly probable to have a $(\mathbf{u}|\mathbf{u})$ structure when $U^\perp \cap V = \{\mathbf{0}\}$, $dim(U) \geq dim(V)$. This $(\mathbf{u}|\mathbf{u})$ reveals information about the secret permutation Q and enables the attacker to locate the U and $U+V$ codes. To avoid this, we should maintain the high dimension of $U^\perp \cap V$, implying that the public code should have a high-dimensional hull. Hence, we define DOOM with high-dimensional hull and assume that the public code of pqsigRM is indistinguishable from a random code with a hull of the same dimension as that of the public code, rather than any random linear code.

Moreover, k_{app} random rows are appended to the generator matrix, and 2^r rows of the generator matrix, that is, the repeated $\mathrm{RM}_{(r,r)}$, are replaced by k_{rep} random rows; furthermore, a codeword from the dual code is appended to the generator matrix. These modifications are equivalent to increasing the dimension of the code itself, the hull, and the dual of the code, respectively, by appending random codewords. Moreover, by adding random codewords, the code is no longer a $(U, U+V)$-code, and thus distinguishing attacks are more difficult to perform.

We now explain the rationale for the aforementioned modifications, which are applied in addition to partial permutation.

1) k_{app} RANDOM ROWS ARE APPENDED TO THE GENERATOR MATRIX

The Hamming weights of a random code are distributed. However, the partially permuted RM code has only codewords with even Hamming weight. This is because the Hamming weights of codewords of $\mathrm{RM}_{(r,m)}$ are even numbers, and partial permutations do not affect parity.

By appending a random row with odd Hamming weight to the generator matrix, the Hamming weights of the public code become distributed binomially. The problem is that if only one row with odd Hamming weight is appended, it can easily be extracted. This can be resolved by appending more than one codeword. Hence, we append k_{app} random rows such that at least one has odd Hamming weight. By the nature of the decoding process, it is still possible to decode the resulting code.

2) APPENDING A RANDOM CODEWORD OF THE DUAL CODE TO THE GENERATOR MATRIX

The Hamming weights of the codewords in the hull of the partially permuted RM code are only multiples of four. However,

the Hamming weight of the codewords in the hull of a random code may be an arbitrary even number, not only a multiple of four. As in the previous modification, a random codeword is appended to the hull. Thereby, we force the codewords of the hull of the public code to have arbitrary even Hamming weights. As a randomly appended row to the generator matrix is unlikely to be appended to its hull, appending a codeword to the hull is more complicated. The following is the process for appending a random codeword to the hull.

Let $hull(\mathcal{C})$ be the hull of a code $\mathcal{C}$. We define $\mathcal{C}'$ and $\mathcal{C}''$ by $\mathcal{C} = hull(\mathcal{C}) + \mathcal{C}'$ and $\mathcal{C}^{\perp} = hull(\mathcal{C}) + \mathcal{C}''$, where $hull(\mathcal{C})$, $\mathcal{C}'$, and $\mathcal{C}''$ are linearly independent. We can then generate a code with a hull with dimension $dim(hull(\mathcal{C})) + 1$ by the following procedure:

i) Find a codeword $\mathbf{c}_{dual} \in \mathcal{C}''$ such that $\mathbf{c}_{dual} \cdot \mathbf{c}_{dual} = 0$. This is easy because a codeword with even Hamming weight satisfies it.
ii) Let $\mathcal{C}_{inc} = \mathcal{C} + \{\mathbf{c}_{dual}\} = (hull(\mathcal{C}) + \{\mathbf{c}_{dual}\}) + \mathcal{C}'$.
iii) As $\mathbf{c}_{dual} \cdot (hull(\mathcal{C}) + \{\mathbf{c}_{dual}\}) = \{0\}$ and $\mathbf{c}_{dual} \cdot \mathcal{C}' = \{0\}$, we have $\mathbf{c}_{dual} \in \mathcal{C}_{inc}^{\perp}$, where for a vector x and a set of vectors A, $x \cdot A$ is the set of all inner products of x and elements of A.
iv) It can be seen that $\mathcal{C}_{inc} \cap \mathcal{C}_{inc}^{\perp} = (hull(\mathcal{C}) + \{\mathbf{c}_{dual}\})$. Hence, $\mathcal{C}_{inc}$ is a code that has a hull of which dimension is $dim(hull(\mathcal{C})) + 1$.

If the Hamming weights of the codewords of the hull are only multiples of 4, then another c_{dual} is selected, and the above process is repeated.

3) REPEATED $\mathrm{RM}_{(r,r)}$ IS REPLACED WITH RANDOM $(2^r, k_{rep})$ CODES

We note that by replacing repeated $\mathrm{RM}_{(r,r)}$ by random $(2^r, k_{rep})$ codes, the dimension of the code is reduced by $2^r - k_{rep}$; this is equivalent to appending $2^r - k_{rep}$ rows to the parity check matrix. The codewords of the dual code of the partially permuted RM code have only codewords of even Hamming weight owing to a subcode of the partially permuted RM code. This can be resolved by replacing this subcode with another random code such that its MD decoder exists. The partially permuted RM code includes $(\mathrm{RM}_{(r,r)}|\ldots|\mathrm{RM}_{(r,r)})$, and the dual code of this has only codewords of even Hamming weight by the proposition below. It is easy to verify that the dual code of the partially permuted RM code is a subset of the dual code of $(\mathrm{RM}_{(r,r)}|\ldots|\mathrm{RM}_{(r,r)})$. That is, $(\mathrm{RM}_{(r,r)}|\ldots|\mathrm{RM}_{(r,r)})$ causes the dual code of the partially permuted RM code to have only codewords of even Hamming weight.

Proposition 2: Let $\mathcal{C}$ be a code such that its dual code has only codewords of even Hamming weight. Then, the dual of the concatenated code, $\{(\mathbf{c}|\mathbf{c})|\mathbf{c} \in \mathcal{C}\}$, has only codewords of even Hamming weight.

Proof: Let $\mathbf{h} \in (\mathcal{C}|\mathcal{C})^{\perp}$, where $\mathcal{C}$ is an (n, k) code and $\mathcal{C}|\mathcal{C}$ is a concatenated code given as $\{(\mathbf{c}|\mathbf{c})|\mathbf{c} \in \mathcal{C}\}$. We define vectors $\mathbf{h}_1$ and $\mathbf{h}_2$ of length n so that $\mathbf{h} = (\mathbf{h}_1|\mathbf{h}_2)$. Clearly, if $\mathbf{h}_1 \in \mathcal{C}^{\perp}$, then $\mathbf{h}_2 \in \mathcal{C}^{\perp}$. If $\mathbf{h}_1 \notin \mathcal{C}^{\perp}$, we have $\mathbf{h}_1 \cdot \mathbf{c} + \mathbf{h}_2 \cdot \mathbf{c} = 0$, i.e., $\mathbf{h}_1 \cdot \mathbf{c} = \mathbf{h}_2 \cdot \mathbf{c}$. This implies that $\mathbf{h}_1 = \mathbf{h}_2$. Hence, wt($\mathbf{h}$) is even. □

By replacing the repeated $\mathrm{RM}_{(r,r)}$ with a random code such that its dual code has codewords of odd Hamming weight, we can force the dual of the public code to have codewords with odd Hamming weight.

Clearly, the dual code of $\mathrm{RM}_{(r,r)}$ is $\{\mathbf{0}\}$. We replace $\mathrm{RM}_{(r,r)}$ with a random $(2^r, k_{rep})$ code. We note that the dual code of this $(2^r, k_{rep})$ code must have codewords with odd Hamming weight. The generator matrix is modified in this manner, rather than by appending rows to the parity check matrix, to ensure that the entire code is decodable.

B. PUBLIC CODE INDISTINGUISHABILITY

In the EUF-CMA security proof, $\rho_c(\mathcal{D}_{pub}, \mathcal{D}_{rand})$ is required to be negligible, that is, the modified RM code distinguishing problem should be hard. As it is challenging to find the computational distance between public and random codes, in this section, we study the randomness of the public code and consider possible attacks.

1) PUBLIC CODE IS NOT A $(U, U+V)$-CODE

After random rows have been appended to the generator matrix of a $(U, U+V)$-code, the resulting code is unlikely to be a $(U, U+V)$-code. Considering the following proposition, it can be seen that with probability $O(2^{k_U - n/2})$, a $(U, U+V)$-code remains a $(U, U+V)$-code after a row has been appended to its generator matrix.

Proposition 3: Let $\mathcal{C}$ be a $(U, U+V)$-code. Then, for all codewords $(\mathbf{c}'|\mathbf{c}'') \in \mathcal{C}$, $(\mathbf{0}|\mathbf{c}' - \mathbf{c}'') \in \mathcal{C}$.

It is expected that attacking the modified RM code is difficult because the appended codewords change the algebraic structure of the code (i.e., the $(U, U+V)$ structure), there is considerable randomness, and there is currently no recovery algorithm.

2) DISTINGUISHING USING HULL

When a random row is appended to the generator matrix, it is unlikely to be included in the hull. To achieve this, the appended row should be a codeword of the dual code, and its square should be zero. Hence, we append a codeword from the dual code to the generator matrix.

The appended row can be omitted when the attacker collects several independent codewords with Hamming weight 4 from the hull. However, for any random code with a high-dimensional hull, the same process can be applied, and finally, there only remain codewords of which the Hamming weight is a multiple of 4. Hence, this is not a valid distinguishing attack.

The hull of a random $(U, U+V)$-code is $\{\mathbf{0}\}$ when $k_U < k_V$ and is highly probable to have codewords of $(\mathbf{u}|\mathbf{u})$ form when $k_U \geq k_V$. However, the hull of an RM code is also an RM code, and in our case, the partial permutation randomizes its hull and retains its large dimension. As shown in Section VI, the hull is neither a subcode of the RM code nor

a $(U, U+V)$-code. Moreover, most of the hull depends on the secret partial permutations σ_p^1 and σ_p^2.

C. SIGNATURE LEAKS

In the EUF-CMA security proof, it is required that $\rho(\mathcal{D}_w, \mathcal{U}_w)$ is a negligible function of the security parameter λ. If this is true, then the signature does not leak information. In several signature schemes, such as Durandal, SURF, and Wave, this is achieved and proved. In SURF and Wave, the rejection sampling method is applied to render $\mathcal{D}_w$ indistinguishable.

To apply rejection sampling, the distribution of the decoding output should be known. In SURF and Wave, a simple and efficient decoding algorithm is used, and thus it is easy to find the distribution of the decoding output. However, in our case, the decoding output exhibits a high degree of randomness, and the structure of the decoder is complex. Therefore, it is difficult to analyze the distribution of the decoding output. Instead, we conduct a proof-of-concept implementation of the modified pqsigRM using SageMath. Then, we perform statistical randomness tests under NIST SP 800-22 [42] on the decoding output, and we compare the results with random errors in $\mathbb{F}_2^n$ with Hamming weight w. No significant difference is observed. However, it should be noted that the success of a statistical randomness test does not imply indistinguishability. Thus, the indistinguishability of the signature should be rigorously studied as future work.

VI. PARAMETER SELECTION

A. PARAMETER SETS

The constraint here is that n is a power of two. We can numerically find the feasible ranges of w once n and k are determined. If the security level λ is achieved in this range, we accept the value; otherwise, we increase n. Considering DOOM, a smaller value of w implies higher security. If w is so small that a large number of decoding iterations are required, we could reduce the partial permutation parameter p. p is at most $n/4$, and the characteristics of the codes are retained by lowering p to a certain degree. The method for obtaining the minimum values is described in the following subsection. The discussed state-of-the-art algorithm for DOOM is used as a basis for the parameters proposed in Table 2. We set $k_{app} = 2$ (the minimum value) and $k_{rep} = 2^r - 2$ (the maximum value).

Regarding the key size, the public key is a parity check matrix given in the systematic form and requires $(n-k)n$ bits. The secret key does not include a scrambler matrix $\mathbf{S}$ because it can be obtained from $\mathbf{H}'$, $\mathbf{Q}$, and $\mathbf{H}$. Moreover $\mathbf{H}$ can be represented by σ_p^1, σ_p^2, replacing code, and appending rows.

The comparison of parameter sets is given in Table 3. The key size of the proposed modified pqsigRM is small compared to other algorithms. We note that it is for reference only, and the actual parameter size is given variously along with trade-off with signing complexity, etc. The security level in parallel-CFS is based on the generalized birthday algorithm [5], and the distinguisher for high-rate Goppa code [4] is not considered. For detailed information, see [3] and [35].

TABLE 3. Comparison of parameter sets of several code-based signature schemes for given security.

λ		proposed	Wave [35]	Parallel-CFS [3]
80	pk size	0.249	1.214	20.0
	sgn. len.	2048	8234	294
128	pk size	0.773	3.108	2.7×10^5
	sgn. len.	4096	13174	474
256	pk size	3.99	12.432	9.4×10^{15}
	sgn. len.	8192	26347	1242

B. STATISTICAL ANALYSIS FOR DETERMINING NUMBER OF PARTIAL PERMUTATIONS

If w is excessively small, there is a low probability of finding an error vector with Hamming weight less than equal to w. We present two solutions. One is iterating until an appropriate error vector is obtained, and the other is improving the decoder. The number p of columns permuted in the partial permutation varies from 0 to $n/4$. From numerical analysis, it is demonstrated that small values of p result in low Hamming weight of the decoding output. However, it should be noted that when $p = 0$, the $(U, U+V)$ part of the modified RM codes becomes identical to the RM code except that $\mathrm{RM}_{(r,r)}$ is replaced. Hence, we propose the lower bound of p that does not affect the randomness of the hull.

Regarding the modified RM code, its hull overlaps with (but is not a subset of) the original RM code. If the hull is a subset of the original RM code, and its dimension is large, the codeword of minimum Hamming weight of the original RM code may be included in the hull. Then, attacks such as the Minder–Shokrollahi attack may be applied using codewords with minimum Hamming weight. Therefore, to prevent attacks, the hull of the public code should not be a subset of the original RM code, and $hull(\mathcal{C}_{pub}) \setminus (\mathrm{RM}_{(r,m)}$ permuted by Q) should occupy a large portion of the hull, where $\mathcal{C}_{pub}$ denotes the public code, and $\setminus$ denotes the relative complement.

As the permutation Q is not important for determining the parameter p, we ignore it in this subsection, and the term permutation refers to the partial permutations σ_p^1 and σ_p^2. When $p = n/4$, which implies that σ_p^1 and σ_p^2 are full permutations, the average dimension of the hull and the dimension of $hull(\mathcal{C}_{pub}) \setminus \mathrm{RM}_{(r,m)}$ are given in Table 4. The values may slightly change according to the permutation.

If p is small, the Hamming weight of the errors decreases. Hence, the signing time can be reduced by using partial permutation with p rather than full permutation. The aim is to find a smaller value for p maintaining the dimension of $hull(\mathcal{C}_{pub}) \setminus \mathrm{RM}_{(r,m)}$ as large as that by the full permutation. It can be seen that the average of the dimension of $hull(\mathcal{C}_{pub}) \setminus \mathrm{RM}_{(r,m)}$ tends to increase as p increases, and it is saturated when p is above a certain value, as in Figure 3. Specifically, the dimension of $hull(\mathcal{C}_{pub}) \setminus \mathrm{RM}_{(r,m)}$ is saturated when p is approximately equal to the average dimension of $hull(\mathcal{C}_{pub}) \setminus \mathrm{RM}_{(r,m)}$ with full permutation. Hence, we determine p as 130, 386, and 562 in Table 2.

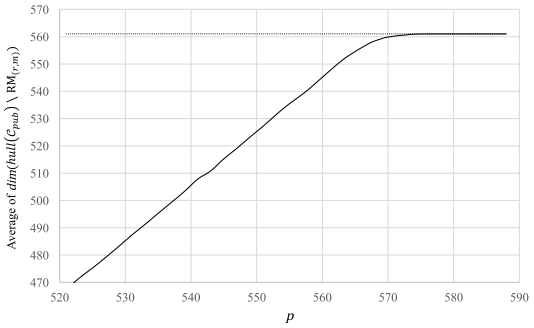


FIGURE 3. Dimension of $hull(\mathcal{C}_{pub}) \setminus \mathrm{RM}_{(6,12)}$ for 128-bit security parameters.

TABLE 4. Average dimension of $hull(\mathcal{C}_{pub})$ and $hull(\mathcal{C}_{pub}) \setminus \mathrm{RM}_{(r,m)}$ with $p = n/4$.

(r, m)	(5,11)	(6,12)	(6,13)
n	2048	4096	8192
k	1025	2511	4097
$dim(hull(\mathcal{C}_{pub}))$	766	1236	2974
$dim(hull(\mathcal{C}_{pub}) \setminus \mathrm{RM}_{(r,m)})$	130	386	562

VII. CONCLUSION

We introduced a new signature scheme, called modified pqsigRM, based on modified RM codes with partial permutation as well as row appending and replacement in the generator matrix. For any given syndrome, an error vector with a small Hamming weight can be obtained. Moreover, the decoding method achieves indistinguishability to some degree because it is collision-resistant. The proposed signature scheme resists all known attacks against cryptosystems based on the original RM codes. The partially permuted RM code improves the signature success condition in previous signature schemes such as CFS and can improve signing time and key size.

We further modified the RM code using row appending/replacement. The resulting code is expected to be indistinguishable from random codes with the same hull dimension; moreover, the decoding of the partially permuted RM code is maintained. Assuming indistinguishability and the hardness of DOOM with a high-dimensional hull, we proved the EUF-CMA security of the proposed signature scheme. The challenge of rigorously verifying these two assumptions will be addressed in the future.

REFERENCES

[1] W. Lee, Y. S. Kim, Y. W. Lee, and J. S. No, "Post quantum signature scheme based on modified Reed–Muller code pqsigRM," in *First Round Submission to the NIST Postquantum Cryptography Call*, Nov. 2017. [Online]. Available: https://csrc.nist.gov/projects/post-quantum-cryptography/round-1-submissions

[2] N. T. Courtois, M. Finiasz, and N. Sendrier, "How to achieve a McEliece-based digital signature scheme," in *Proc. Asiacrypt*, Gold Coast, Australia, Dec. 2001, pp. 157–174.

[3] M. Finiasz, "Parallel-CFS," in *Selected Areas in Cryptography*. Waterloo, ON, Canada: Springer, 2010, pp. 159–170.

[4] J.-C. Faugere, V. Gauthier-Umana, A. Otmani, L. Perret, and J.-P. Tillich, "A distinguisher for high-rate McEliece cryptosystems," *IEEE Trans. Inf. Theory*, vol. 59, no. 10, pp. 6830–6844, Oct. 2013.

[5] D. Wagner, "A generalized birthday problem," in *Proc. Crypto*, Santa Barbara, CA, USA, Aug. 2002, pp. 288–304.

[6] A. Phesso and J.-P. Tillich, "An efficient attack on a code-based signature scheme," in *Proc. PQCrypto*, Fukoka, Japan, 2016, pp. 86–103.

[7] M. Baldi, M. Bianchi, F. Chiaraluce, J. Rosenthal, and D. Schipani, "Using LDGM Codes and sparse syndromes to achieve digital signatures," in *Proc. PQCrypto*, Limoges, France, vol. 7932, 2013, pp. 1–15.

[8] D. Moody and R. Perlner, "Vulnerabilities of 'McEliece in the world of Escher,'" in *Proc. PQCrypto*, Fukuoka, Japan, 2016, pp. 104–117.

[9] D. Gligoroski, S. Samardjiska, H. Jacobsen, and S. Bezzateev, "McEliece in the world of Escher," IACR Cryptol. ePrint Arch., Tech. Rep. 2014/360, 2014.

[10] I. Dumer, "Recursive decoding and its performance for low-rate Reed–Müller codes," *IEEE Trans. Inf. Theory*, vol. 50, no. 5, pp. 811–823, May 2004.

[11] A. Otmani and H. T. Kalachi, "Square code attack on a modified Sidelnikov cryptosystem," in *Proc. C2SI*, 2015, pp. 173–183.

[12] I. V. Chizhov and M. A. Borodin, "The failure of McEliece PKC based on Reed–Müller codes," IACR Cryptol. ePrint Arch., Tech. Rep. 2013/287, 2013.

[13] L. Minder and A. Shokrollahi, "Cryptanalysis of the Sidelnikov cryptosystem," in *Proc. Eurocrypt*, in Lecture Notes in Computer Science, vol. 4515. Barcelona, Spain: Springer, 2007, pp. 347–360.

[14] J. Stern, "A method for finding codewords of small weight," *Coding Theory Appl.*, vol. 388, pp. 106–133, Nov. 1989.

[15] F. Hemmati, "Closest coset decoding of $u|u+v$ codes," *IEEE J. Sel. Areas Commun.*, vol. 7, pp. 982–988, Aug. 1989.

[16] A. Chailloux and T. Debris-Alazard, "A tight security reduction in the quantum random oracle model for code-based signature schemes," 2017, *arXiv:1709.06870*. [Online]. Available: http://arxiv.org/abs/1709.06870

[17] L. Dallot, "Towards a concrete security proof of Courtois, Finiasz, and Sendrier signature scheme," in *Proc. WEWoRC*, vol. 4945. Bochum, Germany: Springer, 2007, pp. 65–77.

[18] K. Morozov, P. S. Roy, R. Steinwandt, and R. Xu, "On the security of the courtois-finiasz-sendrier signature," *Open Math.*, vol. 16, no. 1, pp. 161–167, Mar. 2018.

[19] E. Berlekamp, R. McEliece, and H. van Tilborg, "On the inherent intractability of certain coding problems (Corresp.)," *IEEE Trans. Inf. Theory*, vol. IT-24, no. 3, pp. 384–386, May 1978.

[20] M. Finiasz, "Words of minimal weight and weight distribution of binary goppa codes," in *Proc. IEEE Int. Symp. Inf. Theory*, Jun. 2003, p. 70.

[21] B. Dou, C.-H. Chen, and H. Zhang, "Key substitution attacks on the CFS signature," *IEICE Trans. Fundam. Electron., Commun. Comput. Sci.*, vol. E95-A, no. 1, pp. 414–416, 2012.

[22] R. J. McEliece, "A public-key cryptosystem based on algebraic coding theory," *DSN Prog. Rep.*, vol. 4244, pp. 114–116, Apr. 1978.

[23] D. E. Muller, "Application of Boolean algebra to switching circuit design and to error detection," *Trans. I.R.E. Prof. Group Electron. Comput.*, vol. EC-3, no. 3, pp. 6–12, Sep. 1954.

[24] I. Reed, "A class of multiple-error-correcting codes and the decoding scheme," *Trans. IRE Prof. Group Inf. Theory*, vol. 4, no. 4, pp. 38–49, Sep. 1954.

[25] N. Sendrier, "On the dimension of the hull," *SIAM J. Discrete Math.*, vol. 10, no. 2, pp. 282–293, May 1997.

[26] A. A. Yavuz, A. Mudgerikar, A. Singla, I. Papapanagiotou, and E. Bertino, "Real-time digital signatures for time-critical networks," *IEEE Trans. Inf. Forensics Security*, vol. 12, no. 11, pp. 2627–2639, Nov. 2017.

[27] C. Chen, T. Eisenbarth, I. von Maurich, and R. Steinwandt, "Horizontal and vertical side channel analysis of a McEliece cryptosystem," *IEEE Trans. Inf. Forensics Security*, vol. 11, no. 6, pp. 1093–1105, Jun. 2016.

[28] N. Sendrier, "Decoding one out of many," in *Proc. Int. Workshop Post-Quantum Cryptogr.*, Berlin, Germany: Springer, 2011.

[29] T. Debris-Alazard, N. Sendrier, and J.-P. Tillich, "The problem with the SURF scheme," 2017, *arXiv:1706.08065*. [Online]. Available: http://arxiv.org/abs/1706.08065

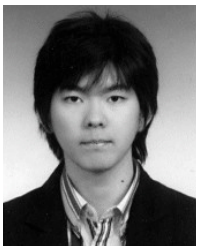
[30] G. Kabatianskii, E. Krouk, and B. Smeets, "A digital signature scheme based on random error-correcting codes," in *Proc. IMA Int. Conf. Cryptogr. Coding*. Cirencester, U.K.: Springer, 1997, pp. 161–167.

[31] G. Kabatiansky, E. Krouk, and S. Semenov, *Error Correcting Coding and Security for Data Networks: Analysis of the Superchannel Concept*. Hoboken, NJ, USA: Wiley, 2005.

[32] P. S. L. M. Barreto, R. Misoczki, and M. A. Simplicio, Jr., "One-time signature scheme from syndrome decoding over generic error-correcting codes," *J. Syst. Softw.*, vol. 84, no. 2, pp. 198–204, Feb. 2011.

[33] N. Aragon, O. Blazy, P. Gaborit, A. Hauteville, and G. Zémor, "Durandal: A rank metric based signature scheme," in *Proc. Annu. Int. Conf. Theory Appl. Cryptograph. Techn.* Darmstadt, Germany: Springer, pp. 728–758, 2019.
[34] C. Gentry, C. Peikert, and V. Vaikuntanathan, "Trapdoors for hard lattices and new cryptographic constructions," in *Proc. 14th Annu. ACM Symp. Theory Comput. STOC*, 2008, pp. 197–206.
[35] T. Debris-Alazard, N. Sendrier, and J.-P. Tillich, "Wave: A new family of trapdoor one-way preimage sampleable functions based on codes," in *Proc. Int. Conf. Theory Appl. Cryptol. Inf. Secur.* Kobe, Japan: Springer, 2019, pp. 21–51.
[36] P.-L. Cayrel, A. Otmani, and D. Vergnaud, "On kabatianskii-krouk-smeets signatures," in *Proc. Int. Workshop Arithmetic Finite Fields*. Madrid, Spain: Springer, 2007, pp. 237–251.
[37] A. Otmani and J.-P. Tillich, "An efficient attack on all concrete KKS proposals," in *Proc. Int. Workshop Post-Quantum Cryptography*. Taipei, Taiwan: Springer, 2011, pp. 98–116.
[38] I. Dumer, "On minimum distance decoding of linear codes," in *Proc. 5th Joint Soviet-Swedish Int. Workshop Inform. Theory*, 1991, pp. 50–52.
[39] A. Becker, A. Joux, A. May, and A. Meurer, "Decoding random binary linear codes in $2^n/20$: How $1 + 1 = 0$ improves information set decoding," in *Proc. Annu. Int. Conf. Theory Appl. Cryptograph. Techn.* Cambridge, U.K.: Springer, 2012, pp. 520–536.
[40] A. May and I. Ozerov, "On computing nearest neighbors with applications to decoding of binary linear codes," in *Proc. Eurocrypt*. Sofia, Bulgaria: Springer, 2015, pp. 203–228.
[41] V. Shoup, "Sequences of games: A tool for taming complexity in security proofs," *IACR Cryptol. ePrint Arch.*, vol. 2004, p. 332, 2004.
[42] I. Lawrence *et al.*, "SP 800-22 Rev. 1a. A statistical test suite for random and pseudorandom number generators for cryptographic applications," Nat. Inst. Stand. Technol., Gaithersburg, MD, USA, Tech. Rep. SP 800-22 Rev. 1a, 2010. Accessed: Sep. 14, 2020.

YONGWOO LEE (Graduate Student Member, IEEE) received the B.S. degree in electrical engineering and computer science from the Gwangju Institute of Science and Technology, Gwangju, South Korea, in 2015, and the M.S. degree in electrical and computer engineering from Seoul National University, in 2017, where he is currently pursuing the Ph.D. degree. He is also a submitter for a candidate algorithm (pqsigRM) in the first round for the NIST Post Quantum Cryptography Standardization. His current research interests include homomorphic encryption and code-based cryptography.



WIJIK LEE received the B.S. and Ph.D. degrees in electrical and computer engineering from Seoul National University. He has been with Samsung Electronics, Hwaseong, South Korea, since 2018. He is also a submitter for a candidate algorithm (pqsigRM) in the first round for the NIST Post Quantum Cryptography Standardization. His research interests include post-quantum cryptography and homomorphic encryption.

YOUNG SIK KIM (Member, IEEE) received the B.S., M.S., and Ph.D. degrees in electrical engineering and computer science from Seoul National University, in 2001, 2003, and 2007, respectively. He joined the Semiconductor Division, Samsung Electronics, where he performed research and development of security hardware IPs for various embedded systems, including modular exponentiation hardware accelerator (called Tornado 2MX2) for RSA and elliptic curve cryptography in smart card products and mobile application processors of Samsung Electronics, until 2010. He is currently a Professor with Chosun University, Gwangju, South Korea. He is also a submitter for two candidate algorithms (McNie and pqsigRM) in the first round for the NIST Post Quantum Cryptography Standardization. His research interests include post-quantum cryptography, the IoT security, physical layer security, data hiding, channel coding, and signal design. He is selected as one of 2025's 100 Best Technology Leaders (for Crypto-Systems) by the National Academy of Engineering of Korea.

JONG-SEON NO (Fellow, IEEE) received the B.S. and M.S.E.E. degrees in electronics engineering from Seoul National University, Seoul, South Korea, in 1981 and 1984, respectively, and the Ph.D. degree in electrical engineering from the University of Southern California, Los Angeles, CA, USA, in 1988. He was a Senior MTS with Hughes Network Systems, from 1988 to 1990. He was an Associate Professor with the Department of Electronic Engineering, Konkuk University, Seoul, from 1990 to 1999. He joined the faculty of the Department of Electrical and Computer Engineering, Seoul National University, in 1999, where he is currently a Professor. His research interests include error-correcting codes, cryptography, sequences, LDPC codes, interference alignment, and wireless communication systems. He became an IEEE Fellow through the IEEE Information Theory Society in 2012. He became a member of the National Academy of Engineering of Korea (NAEK), in 2015, where he is currently the Division Chair of Electrical, Electronic, and Information Engineering. He was a recipient of the IEEE Information Theory Society Chapter of the Year Award in 2007. From 1996 to 2008, he has served as the Founding Chair of the Seoul Chapter of the IEEE Information Theory Society. He was the General Chair of Sequence and Their Applications 2004 (SETA 2004), Seoul. He also served as the General Co-Chair of the International Symposium on Information Theory and Its Applications 2006 (ISITA 2006) and the International Symposium on Information Theory 2009 (ISIT 2009), Seoul. He has served as the Co-Editor-in-Chief for the IEEE JOURNAL OF COMMUNICATIONS AND NETWORKS from 2012 to 2013.