First PQC Standardization Conference 2018, NIST

pqsigRM:
Modified RM Code-Based Signature Scheme

April 13, 2018

Wijik Lee, Yongwoo Lee, **Jong-Seon No**[*1], Young-Sik Kim[2]
[1]Department of ECE, INMC, Seoul National University, Seoul, Korea
[2]Chosun University, Gwangju, Korea

Coding and Cryptography Lab.
Dept. of ECE, Seoul National University

Outline

I. Introduction
II. pqsigRM
III. Known Issues and Solutions

Outline

Code-Based Signature Scheme

- CFS signature scheme is one of the well-known post-quantum signature scheme.
- RM code-based CFS signature scheme is proven to be insecure due to Minder-Shokrollahi's attack and later the Chizhov-Borodin's attack and square code attack.
- We propose the modification methods for the CFS signature scheme based on the modified RM codes.

CFS Signature Scheme

- CFS signature scheme (Courtois, Finiasz, Sendrier, 2001)
  - Using Goppa code.
- Message is hashed to a syndrome and a signature is treated as an error.
  - $h(m)$: Hashed massage
  - Find signature z such that $H'z = h(h(m)|i)$, where H' is a parity check matrix and i is a counter.
- Disadvantage
  - The probability of finding decodable syndrome is $\frac{1}{t!}$, which is too low.
  - The private and public key sizes are large.
- Other signature schemes have been broken, such as KKS, KKS variants, and CFS based on LDGM codes.

RM Code-Based CFS Signature Scheme

- Decoding of RM code can perform closest coset decoding.
  - RM code-based CFS signature scheme takes **less signing time** than Goppa code-based CFS signature scheme.
- Attacks on RM code-based cryptosystems/signature schemes.
  - Minder-Shokrollahi's attack
  - Chizhov-Borodin's attack
  - Square code attack
- Our proposed pqsigRM is the modified version of the RM code-based CFS signature scheme to prevent these attacks.

Outline

I. Introduction

II. pqsigRM

III. Known Issues and Solutions

Public Key of pqsigRM

- Delete the rows of index set L_D in the systematic form of parity check matrix $H = [P^T \,|I]$.
- Replace the p rows of the parity part P^T by the binary random vectors.
- Then, the modified matrix H_m is given as

$$H_m = \begin{array}{c} n-k-p \\ p \end{array} \left[\begin{array}{c:c:c} \overbrace{P'^T}^{k} & \overbrace{I_{n-k-p}}^{n-k-p} & \overbrace{0}^{p} \\ \hdashline \multicolumn{2}{c:}{R} & I_p \end{array} \right]$$

Figure: Modified parity check matrix of the proposed signature scheme.

- $H' = SH_mQ$ is the public key of pqsigRM, where S is a $(n-k) \times (n-k)$ scrambling matrix and Q is a permutation matrix.

Outline

I. Introduction

II. pqsigRM

III. Known Issues and Solutions

Known Issues

- Attacks revealing puncturing/insertion have been proposed by the pqc-forum.
  - The signature has higher probability for element 1 in the punctured/inserted positions of signature .
  - The near-minimum codewords have higher probability for element 1 in the punctured/inserted positions of codewords.
  - The hull of public code has all zero in the punctured/inserted positions of codewords.
- We have prevented these attacks by the following modification.

The Generator Matrix of pqsigRM Public Code

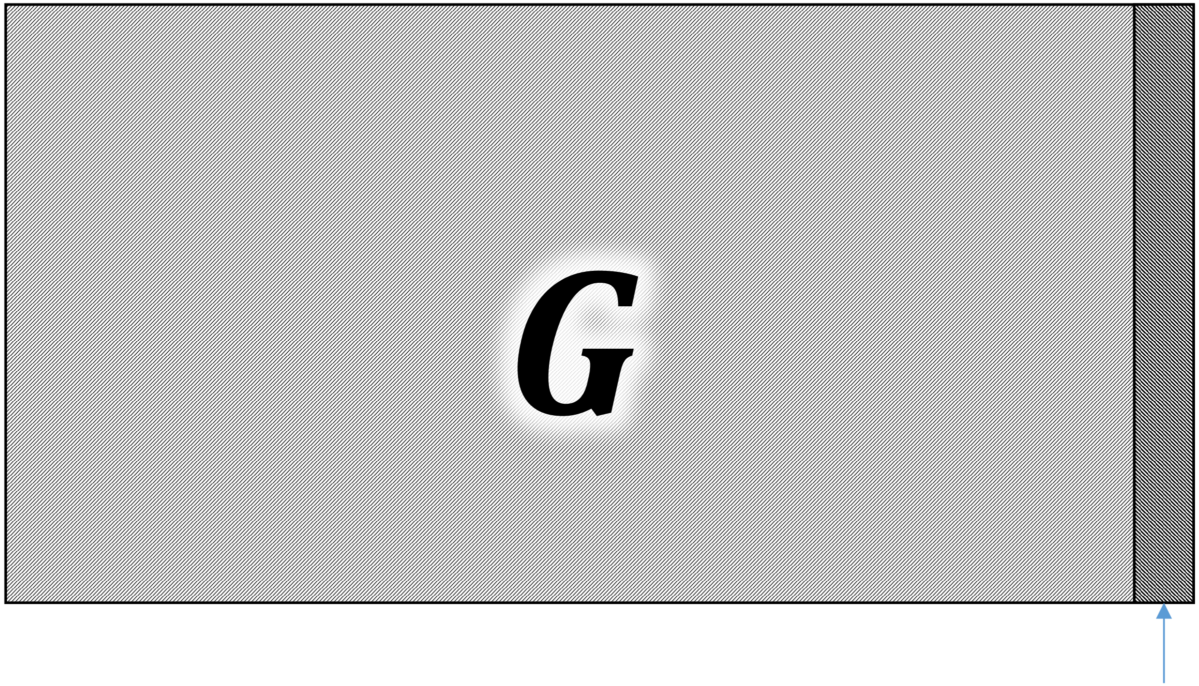

Figure: The generator matrix of pqsigRM public code.

Modification of Generator Matrix of RM(5,11)

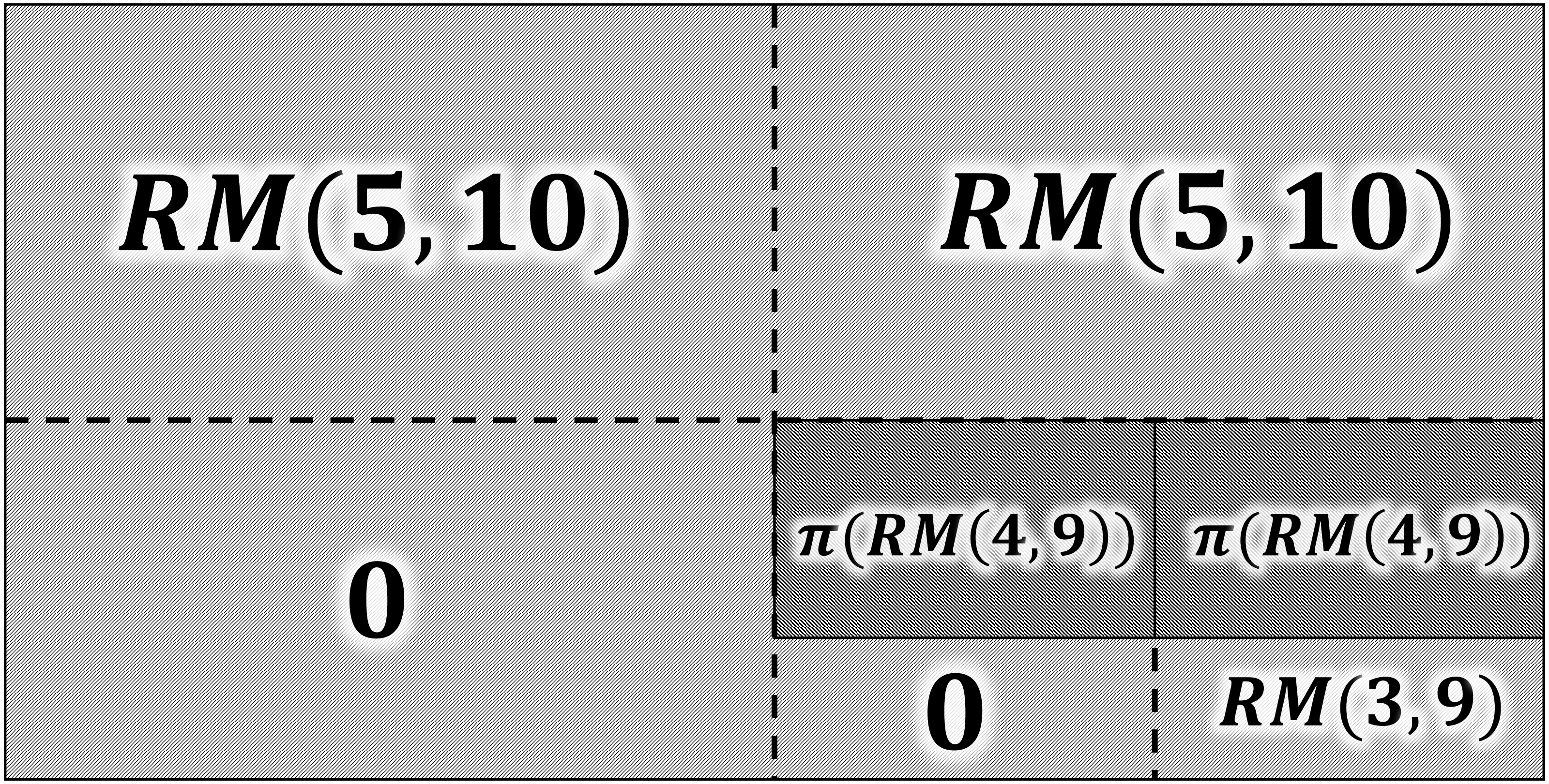

Figure: The generator matrix of the modified pqsigRM public code from RM(5,11) .

Modification of Generator Matrix

- The public key of pqsigRM is a permuted parity check matrix corresponding to the generator matrix of the RM code, in which p columns are replaced by random vectors.
- Here, we will simply replace the generator matrix with permuted generator submatrix of RM code.
- For example, in pqsigRM-5-11, we replace the partial matrices of G, the generator matrix of RM(5,11), with the generator matrix of a permuted RM(4,9).

New Decoding Algorithm for Signing

Algorithm – decoder for pqsigRM-5-11, $\Psi_r^m(y, f, r)$:

If $r = 0$, perform MD decoding for code RM$(0, m)$

Elif $r = m$, perform MD decoding for code RM(r, r)

Else

If $f = 1024$ and $r = 1536$, depermute y

$(y'|y'') \leftarrow y$

$y^v \leftarrow y'y''$

$\widehat{y^v} \leftarrow \Psi_{r-1}^{m-1}\left(y^v, \frac{f+r}{2}, r\right)$

$y^u \leftarrow (y' + y''\widehat{y^v})/2$

$\widehat{y^u} \leftarrow \Psi_r^{m-1}\left(y^u, f, \frac{f+r}{2}\right)$

$\widehat{y^c} \leftarrow (\widehat{y^u}|\widehat{y^u}\widehat{y^v})$

If $f = 1024$ and $r = 1536$, permute $\widehat{y^c}$

Return $\widehat{y^c}$

Performance

Security	Algorithm	Public key size (Byte)	Performance(ms)		
			Key generation	Signing	Verification
Category 1	pqsigRM-5-11	129 K	787	11375	12
Category 3	pqsigRM-6-12	488 K	4009	11013	49
Category 5	pqsigRM-6-13	2055 k	37249	227	331

*Benchmark on Intel(R) i7-6700k 4.00GHz, single core

Conclusion

- There is no all-zero position on the hull of public code.
- The probability for elements 1's in the signature is almost equal.
- Near-minimum Hamming weight codewords are no longer useful to locate the modified columns, because 1/2 elements of each codeword are replaced by partially permuted RM codes.
- Modifying the generator matrix in this way also prevents square code attack, Chizhov-Borodin's attack, and Minder-Shokrollahi's attack.
- Further optimization for key sizes and running times is required.