5th International Workshop on Code-based Cryptography

Zurich, Switzerland, May 25-26, 2024

https://cb-crypto.org

Book of Abstracts

About

Code-based cryptography is the area of research that focuses on the study of cryptosystems based on error-correcting codes, following the seminal work of McEliece and Niederreiter in the late 1970s - early 1980s. These systems have shown no vulnerabilities to quantum attackers and this research branch is widely regarded as one of the most promising in the so-called area of Post-Quantum Cryptography. Current efforts in code-based cryptography are directed at producing fast, secure and efficient schemes. Research in this area has also been fostered by the recent NIST's Post-Quantum Standardization call.

The goal of this two-day event is to promote this research area to an increasingly larger audience. Besides bringing together the existing community, in fact, CBCrypto aims at providing an opportunity to extend the range of participation to researchers approaching this area for the first time, or simply interested in knowing more about it. The program includes invited talks, contributed talks, posters and dedicated discussion sessions.

After the workshop, we invite all accepted abstracts to submit their full papers to the post-conference proceedings at a special issue in Springer's Lecture Notes in Computer Science series. Full versions will undergo an independent review process. Further instructions about submission of full papers will be given after the conference, together with the invitation to submit.

Organizers

Violetta Weger, Technical University of Munich
Anna-Lena Horlemann, University of St.Gallen
Jean-Christophe Deneuville, Ecole Nationale de l'Aviation Civile

Conference Location

ETH Zurich
Room E1.1
Rämistrasse 101
8006 Zurich, Switzerland

Program

	Saturday	Sunday	
8:30	Registration Opens		
9:00	Opening Speech		
9:10	**Invited Speaker**		
	Algebraic methods in code-based cryptography		
	Simona Samardjiska	**Invited Speaker**	9:30
10:10	**Contributed Talk**	Hints for Codes and Lattices	
	Properties of Quasi-Cyclic MDPC Codes in Post-Quantum Code-Based Cryptosystems	*Alexander May*	
	Gretchen Matthews		
10:40	**Coffee Break**	**Coffee Break and Poster Session**	10:30
11:00	**Contributed Talks**	**Contributed Talks**	11:00
	Dihedral MDPC Quantum Codes	FuLeakage: Breaking FuLeeca by Learning Attacks	
	Najda Willenborg	*Felicitas Hörmann*	
11:30	Breaking HWQCS: a code-based signature scheme from high weight QC-LDPC codes	On Linear Equivalence, Canonical Forms, and Digital Signatures	11:30
	Giovanni Tognolini	*Tung Chou*	
12:00	Tighter DFR analysis and new decoders for HQC	Lattice approach to Lee metric decoding	12:00
	Sebastian Bitzer	*Karan Khathuria*	
12:30	**Lunch**	**Lunch**	12:30
13:30	**Contributed Talks**	**Work in Groups and Poster Session**	13:30
	Breaking Four Code-Based Cryptosystems		
	Stefan Ritterhoff		
14:00	SDitH in Hardware		
	Sanjay Deshpande		
14:30	Public-Key Encryption based on Supercode Decoding		
	Anmoal Porwal		
15:00	**Coffee Break**	**Coffee Break and Poster Session**	15:00
15:30	**Contributed Talks**	**Contributed Talks**	15:30
	On the Rank of Random Binary Sub-Matrices and its Impact for Information Set Decoding Algorithms	Group Factorisation for Smaller Signatures from Cryptographic Group Actions	
	Bénédikt Tran	*Giuseppe D'Alconzo*	
16:00	Extending Interactive Oracle Proofs to General Linear Codes	Complexity of Solving Syndrome Decoding Problems as a System of Multivariate Equations	16:00
	Adrien Pasquereau	*Alex Pellegrini*	
16:30	Asymptotic Cost Comparison of Generic Rank Decoders	Closing Speech	16:30
	Hugo Sauerbier Couvée		
17:00	End of Day	End of Day	17:00
19:00	Social Event	Social Event	19:00

List of Abstracts

Saturday, May 25, 2024

Invited Talk: 09:10-10:10

Algebraic methods in code-based cryptography

Simona Samardjiska
Radboud University, the Netherlands

Code-based cryptography has been around for quite a while with the McEliece cryptosystem using the syndrome decoding problem in the Hamming metric considered one of the best understood cryptosystems with stable record of cryptanalytical advancement. The best attacks are still message recovery attacks that use combinatorial methods against the underlying hard problem.

In recent years, the quest for better performance has made the code-based scene much more colorful with abundance of new metrics, new hard problems and cryptographic constructions. As a result, the cryptanalytic methods are also more varied with algebraic methods becoming more relevant and more creative.

In this talk I will give an overview of algebraic attacks used in code-based cryptography. On a high level, such an attack involves modeling a hard problem or a cryptosystem as a system of equations and then solving it. The challenge lies in finding the best possible algebraic model and the best possible solving method. I will go through several examples of algebraic modeling and solving of hard problems, decryption errors and cryptographic construction. I will further argue that often, a clever combination of algebraic and combinatorial methods yields the best results.

Keywords: code-based cryptography, cryptanalysis, algebraic modelling.

Session 1: 10:10 – 10:35

Properties of Quasi-Cyclic MDPC Codes in Post-Quantum Code-Based Cryptosystems

Gretchen L. Matthews
Virginia Tech, Blacksburg, VA USA

Joint work with: Emily McMillon

Keywords: cycles, cryptosystem, parity-check code, quasi-cyclic code

Abstract Quasi-cyclic low-density parity-check (QC-LDPC) codes are widely studied due to their practical and simple implementation. Recently, quasi-cyclic moderate-density parity-check (QC-MDPC) codes have also become relevant due to their role in code-based cryptography schemes which are important in post-quantum cryptography. In this work, we study structural properties of QC-MDPC codes that have not been thoroughly studied in the LDPC context. We enumerate 4-cycles and compute column intersections for the class of QC-MDPC codes, both of which have been shown to negatively affect iterative decoder performance.

Introduction

Quasi-cyclic low-density parity-check (QC-LDPC) codes are a widely used class of LDPC codes due to their practical and simple implementations. Defined by sparse matrices, LDPC codes paired with an iterative decoder are near capacity-achieving. Consequently, this class of codes has been widely studied. Moderate-density parity-check codes, while defined similarly with denser matrices have not captured as much attention. The higher density makes certain graph substructures that impede decoder performance unavoidable and also increases the decoder complexity. Recently, quasi-cyclic moderate-density parity-check (QC-MDPC) codes have become relevant due to their role in code-based post-quantum cryptography schemes based on the ideas of the McEliece [78] and Niederreiter cryptosystems [81].

In this work, we consider QC-MDPC codes as used in the Bit Flipping Key Encapsulation (BIKE) code-based cryptosystem, a Round 4 candidate in the NIST Post-Quantum Cryptography Standardization process [15]. In 2016, Guo, Johansson, and Stankovski [58] demonstrated that decoder failures can lead to a private key recovery attack in cryptosystems based on QC-MDPC codes. Their observations led to the notion of weak keys in QC-MDPC-based cryptosystems. Our contribution is to capture these weak keys as known structures in the Tanner graph associated with the code, complementing the recent work in [18] focused on empirical results.

Cycles, column intersections, and keys

In the BIKE cryptosystem, the private key in BIKE is an $r \times 2r$ quasi-cyclic binary matrix

$$H = [H_0 \ H_1] \in \mathbb{F}_2^{r \times 2r}$$

composed of two circulant blocks, H_0 and H_1, each of size $r \times r$ with r prime and such that $x^r - 1$ has only two irreducible factors modulo 2. It is useful to consider the distance between two positions $i, j \in \{0, \ldots, r-1\}$ in a circulant block, which is given by

$$d(i,j) = \min\{j - i \mod r, i - j \mod r\}.$$

The multiplicity of a distance is the number of times it appears as the difference of two degrees of nonzero monomials of a polynomial $h \in \mathbb{F}_2[x]/\langle x^r - 1\rangle$,

$$\mu(\delta, h) = |\{(i,j) \mid d(i,j) = \delta, h_i = h_j = 1, 0 \leq i \leq j < r\}|\,.$$

Proposition 1. *Let $h \in \mathbb{F}_2[x]/\langle x^r - 1\rangle$, and $H \in \mathbb{F}_2^{r\times r}$ the parity-check matrix corresponding to h. The number of 4-cycles in the Tanner graph of H is*

$$r \cdot \sum_{\delta=1}^{\lfloor r/2 \rfloor} \binom{\mu(\delta, h)}{2}.$$

We relate this result to weak keys in QC-MDPC-based cryptosystems. To understand the impact on performance of the Black-Gray-Flip (BGF) Decoder, for each $i \in [r]$, we define the support profile of its i^{th} column $Col_i H$ to be the multiset

$$SP(Col_i(H)) := \{\{| Col_i H \cap Col_j H |: j \in [r] \setminus \{i\}\}\}\,.$$

The support profile of H is defined to be the multiset

$$SP(H) := \{\{SP(Col_i H) : i \in [r]\}\}\,.$$

The maximum column intersection of H is

$$MCI(H) := \max SP(H).$$

Proposition 2. *The maximum column intersection of H is*

$$MCI(H) = \max_{i,j}\{\mu(d(i,j), h)\} = \max_{j}\{\mu(d(0,j), h)\}.$$

We discuss the ramifications of these results for weak keys in BIKE.

Acknowledgments

We would like to thank the National Science Foundation for support through NSF DMS-2201075 and NSF DMS-2303380 and the Commonwealth Cyber Initiative.

Session 2: 11:00 – 12:25

Dihedral Quantum MDPC Codes and their Applications to Code-Based Cryptography

Nadja Willenborg
University of St.Gallen, St.Gallen, Switzerland

Joint work with: Anna-Lena Horlemann

Keywords: McEliece, Lifted Product Codes, Dihedral Group Algebra

In the evolving landscape of post-quantum cryptography, the McEliece cryptosystem, which exploits the hardness of decoding linear codes, emerges as a paradigm of resilience against quantum computational attacks. Among the variants of this system, those utilizing moderate-density parity-check (MDPC) codes have emerged as noteworthy for their trade-off between security and efficiency (see e.g. [77]). This achievement represents a pivotal advancement in the ongoing pursuit of cryptographic schemes resilient to quantum computing threats.

Recently it has been shown [87] that there exists a family of asymptotically good quantum low-density parity-check (LDPC) codes, i.e., quantum LDPC codes for which both the code minimum distance and the dimension grow linearly with the block length N, based on the lifted product construction from certain group codes. A generalization of this result holds profound implications for the McEliece cryptosystem, particularly when considering the integration of MDPC codes.

The aim of this talk will be to explore the intersection of quantum error correction and cryptography, emphasizing the role of quantum MDPC codes in the context of McEliece-type systems. In particular, we will focus on the construction of lifted product codes over the group algebra $\mathbb{F}_q[D_{2n}]$, defined over a field $\mathbb{F}_q$ and the dihedral group D_{2n} of order $2n$. Lifted product codes over this group algebra have not yet been considered in this generality and still allow the formulation of useful distance bounds. With this construction which is based on the problem of constructing classical linear codes with certain self-orthogonal properties, we can give explicit examples of new MDPC quantum codes.

Our constructed codes are characterized by their parity-check matrices with weights in $\mathcal{O}(\sqrt{N})$. Since such moderate dense parity-check matrices significantly increase the difficulty of finding low weight codewords in the dual code, this in turn enhances the security of the cryptographic system they can be used in.

Breaking HWQCS: a code-based signature scheme from high weight QC-LDPC codes

Giovanni Tognolini
University of Trento, Italy

Joint work with: Alex Pellegrini

Keywords: Code-Based Cryptography, Hash&Sign, Cryptanalysis

Abstract We analyze HWQCS, a code based signature scheme presented at ICISC 2023, which uses quasi-cyclic low density parity check codes (QC-LDPC). The scheme introduces high Hamming weight errors and signs each message using a fresh ephemeral secret key rather than using only one secret key, so to avoid known attacks on QC-LDPC signature schemes. In this work, we show that the signatures of HWQCS leak substantial information concerning the ephemeral keys and formally describe this behavior. Furthermore, we show how to exploit the leakage to efficiently reconstruct partial secret data from very few signatures, and finally mount a universal forgery attack.

Overview of the Attack

We describe the problem from which HWQCS [103] suffers, taking it out-of-the-box. Given two positive integers k and w, let $R := \mathbb{F}_2[x]/(x^k - 1)$, and $V_{k,w} := \{\mathbf{c} \in \mathbb{F}_2^k \mid \mathrm{wt}(\mathbf{c}) = w\}$, where $\mathrm{wt}(\cdot)$ denotes the Hamming weight. We will identify elements in R with their coefficient vectors over $\mathbb{F}_2^k$. The signatures generated by the signing algorithm of HWQCS requires the publication of a value $\mathbf{s}_i \in R$, for $i \in \{1, 2\}$, defined as $\mathbf{s}_i := \mathbf{u}_i\mathbf{f}_i + \mathbf{c}\mathbf{e}_i$, where $\mathbf{e}_i \in V_{k,w_e}, \mathbf{u}_i \in V_{k,w_u}$ are ephemeral private values, $\mathbf{f}_i \in V_{k,w_f}$ is the secret key, $\mathbf{c} \in V_{k,w_c}$ and $w_w, w_u, w_f, w_c \geq 0$ are public values. For all suggested parameters set, w_w, w_u, w_f, w_c have relatively small positive integers values. According to [103], the reconstruction of the ephemeral values $\mathbf{e}_1$ and $\mathbf{e}_2$ allows mounting a universal forgery attack on the scheme. In our work, we show how to recover these values.

Information Leakage

We show that, for $i \in \{1, 2\}$, $\mathbf{s}_i$ leaks a critical amount of information about the ephemeral value $\mathbf{e}_i$. In order to see that, let $[k] := \{1 \ldots, k\}$ and $\mathrm{supp}(\mathbf{c}) := \{i \in [k] \mid c_i \neq 0\}$ and consider the following procedure:

- let $v \in \mathrm{supp}(\mathbf{c})$ and notice that $x^{-v}\mathbf{s}_i = \mathbf{e}_i + x^{-v}\mathbf{u}_i\mathbf{f}_i + \sum_{l \in \mathrm{supp}(\mathrm{c}) \setminus \{v\}} x^{l-v}\mathbf{e}_i$,
- compute $\mathbf{d}_i := \sum_{v \in \mathrm{supp}(\mathrm{c})} x^{-v}\mathbf{s}_i \in \mathbb{Z}[x]$,

where we stress that the sum of the second step is taken over the integers. Notice that for every $v \in \mathrm{supp}(\mathbf{c})$, the value $x^{-v}\mathbf{s}_i$ is given by $\mathbf{e}_i$ plus some random noise. Therefore, we expect the larger coefficients of $\mathbf{d}_i$ to be associated with the entries of $\mathbf{e}_i$ equal to 1. Theorem 3 formally describes the behaviour of the entries of $\mathbf{d}_i \in \mathbb{Z}^k$.

Theorem 3. *Let* $\mathbf{u}, \mathbf{f}, \mathbf{e}$ *and* $\mathbf{c}$ *be random elements of* $\mathbb{F}_2^k$*, having Hamming weight* w_u, w_f, w_e *and*

w_c, respectively. Let $v \in \mathrm{supp}(\mathbf{c})$ and define

$$\mathbf{d} := \sum_{v \in \mathrm{supp(c)}} x^{-v}\mathbf{s} \in \mathbb{Z}[x].$$

Then, for $j \in [k-1]$, we have that $(\mathbf{d})_j$ is binomially distributed as

$$(\mathbf{d})_j \sim Bin\left(w_c, \prod_{i=1}^{3} p_i + \sum_{i=1}^{3} p_i \prod_{j \in [3], j \neq i} (1-p_j)\right), \tag{0.1}$$

where $p_1 = w_e/k$, $p_2 = (1-(1-2w_e/k)^{w_c-1})/2$ and

$$p_3 = \frac{1}{\binom{k}{w_u}\binom{k}{w_f}} \sum_{\substack{1 \leq l \leq \min(w_u, w_f) \\ l \text{ odd}}} \binom{k}{l}\binom{k-l}{w_u-l}\binom{k-w_u}{w_f-l}.$$

In the settings of our attack, we specialize Theorem 3 to the case where $p_1 \in \{0,1\}$. This allows us to study the effect of the j-th coordinate of the vector $\mathbf{e}_i$ on the distribution of the j-th coordinate of $\mathbf{d}_i$. Theorem 3 has two main consequences: on the one hand, given a threshold value τ, we can estimate the probability p_{succ} that all entries j with $(\mathbf{d}_i)_j < \tau$ are such that $(\mathbf{e}_i)_j = 0$. On the other hand, we can also estimate the expected number N_0 of coordinates j such that $(\mathbf{d}_i)_j < \tau$ and $(\mathbf{e}_i)_j = 0$, which means that with probability p_{succ} we are able to correctly reconstruct N_0 of the k coordinates of $\mathbf{e}_i$. In order for our attack to succeed, we need to correctly guess k zero coordinates of $(\mathbf{e}_1, \mathbf{e}_2)$. If $N_0 < k$ we need to find the missing $\lceil \frac{k}{2} \rceil - N_0$ zero positions of $\mathbf{e}_i$.

According to the behaviour of the $(\mathbf{d}_i)_j$ entries, we aim at finding these values among the j's such that $(\mathbf{d}_i)_j = \tau$, i.e. the set of positions that have the least probability of containing an error, after those we already chose. Let M_0 and M_1 be the expected number of error free and error positions j such that $(\mathbf{d}_i)_j = \tau$, respectively. Then $M_0 = (k - w_e)\mathbb{P}((\mathbf{d}_i)_j = \tau \mid (\mathbf{e}_i)_j = 0)$ and $M_1 = (w_e)\mathbb{P}((\mathbf{d}_i)_j = \tau \mid (\mathbf{e}_i)_j = 1)$. Therefore the probability such that $\lceil \frac{k}{2} \rceil - N_0$ randomly chosen positions j such that $(\mathbf{d}_i)_j = \tau$ will be error free is given by $q_{succ} := (M_0/(M_0 + M_1))^{\lceil \frac{k}{2} \rceil - N_0}$. The overall probability of a correct recovery of $\lceil k/2 \rceil$ zero bits of $\mathbf{e}_i$ is then $p_{tot} := p_{succ} \cdot q_{succ}$. Applying this technique on both $\mathbf{e}_1$ and $\mathbf{e}_2$ we can recover k error free positions of $\mathbf{e}$ with probability of success p_{tot}^2. As a real instance example, Table 1 displays the values we obtain from our analysis for security level $\lambda = 128$, which has parameters $(k, \omega_f, \omega_u, \omega_e, \omega_c) = (12539, 145, 33, 141, 31)$.

$\lceil \frac{k}{2} \rceil$	τ	N_0	p_{succ}	$\lceil \frac{k}{2} \rceil - N_0$	q_{succ}	p_{tot}^2
6270	12	5564.3997	0.6707	706	0.7491	0.2524

Table 1: For $\lambda = 128$, we report the chosen threshold value τ, the value N_0 of zero bits that we expect to find among the j's such that $(\mathbf{d}_i)_j < \tau$, and the probability p_{succ} that this event occurs. The value $\lceil \frac{k}{2} \rceil - N_0$ is the number of positions that we still need to guess from each of $\mathbf{e}_1$ and $\mathbf{e}_2$ to fully reconstruct $\lceil k/2 \rceil$ entries. The value q_{succ} represents the probability to correctly guess these values, and p_{tot}^2 denotes the probability of recovering all of them correctly, for both $i \in \{1, 2\}$.

Completing the reconstruction of the ephemeral values

We exploit the analysis performed so far to fully reconstruct the ephemeral values $\mathbf{e}_i$. Among the rest, HWQCS publishes $H = \big(\mathrm{circ}(\mathbf{h}), \mathrm{circ}(\mathbf{h})^{-1}\big) \in \mathbb{F}_2^{k\times 2k}$, with $\mathrm{circ}(\mathbf{h})$ and $\mathrm{circ}(\mathbf{h}^{-1})$ being the circulant matrices associated to $\mathbf{h} \in R$, the public key of the scheme. Moreover, each signature contains the value $\mathbf{b} = H\mathbf{e}^\top$, where $\mathbf{e} = (\mathbf{e}_1, \mathbf{e}_2)$ is viewed as an element in $\mathbb{F}_2^{2k}$. Let $J \subset [2k]$ the set of positions recovered using the strategy outlined in the previous phase of the attack, and let $I := [2k] \setminus J$. Finally, let H_I be the submatrix of H which consists of the columns indexed by I. We can treat H_I as a random matrix in $\mathbb{F}_2^{k\times k}$. Assume that H_I is invertible. Then, given the syndrome equation $\mathbf{b} = H\mathbf{e}^\top$, we can compute $\bar{\mathbf{e}} = H_I^{-1}\mathbf{b}$ and thus reconstruct $\mathbf{e} = (e_0, \dots, e_{2k-1})$ as, for each $h \in [2k]$ $e_h = \bar{e}_h$ if $h = i$ for some $i \in I$ and 0 otherwise. For each set of suggested parameters of HWQCS, the value of the probability of H_I to be invertible is easily computed as $p_{inv} \approx k^{2.37} = 0.2888$.

Complexity of the attack

The attack succeeds at reconstructing the data needed for a universal forgery if we can correctly obtain k error free positions of $\mathbf{e}$ and if the matrix H_I is invertible. Therefore, the success probability is given by $p_{break} = p_{tot}^2 \cdot p_{inv}$. For instance, for security $\lambda = 128$ we have $p_{break} \approx 0.0727$, and we expect to recover $\mathbf{e}_1$ and $\mathbf{e}_2$ within $p_{break}^{-1} \approx 14$ signatures, with overall complexity $k^{2.37}/p_{break} \approx 2^{36}$.

A Sage implementation of the code and the full version of this work can be respectively found at `https://github.com/triki96/Cryptanalysis-of-HWQCS` and [88].

Tighter DFR Analysis and New Decoders for HQC

Sebastian Bitzer
Technical University of Munich, Germany

Joint work with: Marco Baldi, Nicholas Lilla, Paolo Santini

Keywords: HQC, Decryption Failure Rate, Decoder

Abstract We analyze the decoder of Hamming Quasi-Cyclic (HQC), a code-based post-quantum key encapsulation mechanism admitted to the fourth round of the ongoing NIST competition. A closer look at the error vector reveals a structure that can be exploited for a twofold purpose: providing tighter estimates of the DFR of current decoding algorithms and devising new decoders that can achieve improved DFR performance.

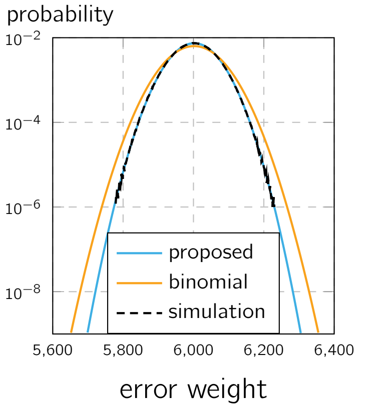
A Quick Overview of HQC

The code-based cryptosystem Hamming Quasi-Cyclic (HQC) [73] is among the round-4 candidates in NIST's competition for the standardization of post-quantum cryptography. HQC is unique in the sense that it uses a public error-correcting code. Decoding of this code recovers the message during decryption. The error that needs to be corrected is given by $\mathbf{e} = \mathbf{u}_1 \cdot \mathbf{v}_1 + \mathbf{u}_2 \cdot \mathbf{v}_2 + \mathbf{v}_3$, where $\mathbf{u}_1, \mathbf{u}_2, \mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3 \in \mathbb{F}_2[x]/(x^n - 1)$ are sparse polynomials. The HQC team claims a precise DFR analysis for HQC is available. However, since the distribution of the Hamming weight of two sparse polynomials over $\mathbb{F}_2$ is generally unknown, they rely on a *binomial approximation*, which models the coefficients of $\mathbf{z}$ as independent Bernoulli random variables. For several lattice-based schemes, this independence assumption is known to be problematic, see e.g. [51]. In the case of HQC, extensive simulations indicate that the approximation yields an upper bound on the actual decryption failure rate (DFR) [11]. Nevertheless, no formal proof exists that the binomial approximation indeed yields an upper bound on the tails of the weight distribution, which are the relevant part for cryptographic applications. Further, it is currently unclear how large the gap between the bound and the actual performance is for practical parameters.

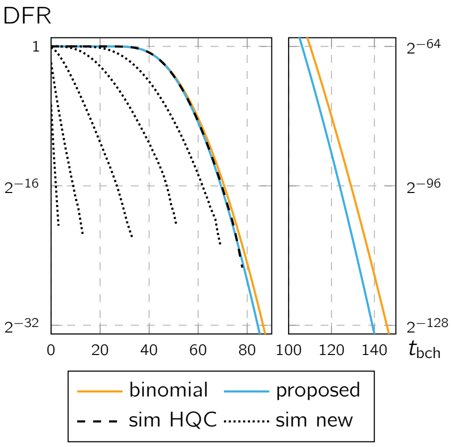
Our contribution

In this work, we extend the binomial approximation by taking further properties of the polynomial multiplication into account. We do not capture the behavior of the multiplication in full, but simulations indicate that we achieve an accurate description of the weight distribution, see Figure 1a. We observe that this refined model predicts lower DFRs than previously assumed, see 1b.

A closer look at the structure of the error vector $\mathbf{e}$ reveals correlations involving also the secret-key polynomials $\mathbf{u}_1$ and $\mathbf{u}_2$. This structure is not utilized by the HQC decoder. We propose a simple modification of the classical decoder that exploits the correlations and achieves a significant decrease in the decryption failure rate, see 1b. We consider a rigorous DFR analysis as well as developing a coding scheme tailored to the particular error structure interesting future work.



(a) Error weight distribution for round-3 HQC parameters.



(b) DFR of HQC decoder and proposed decoder.

Acknowledgements

The work of Marco Baldi has been supported by the Italian Ministry of University's PRIN 2022 program under the "Mathematical Primitives for Post Quantum Digital Signatures" (P2022J4HRR) and "Post quantum Identification and Encryption Primitives: Design and Realization (POINTER)" (2022M2JLF2) projects funded by the European Union - Next Generation EU. Sebastian Bitzer acknowledges the financial support by the Federal Ministry of Education and Research of Germany in the program of "Souverän. Digital. Vernetzt.". Joint project 6G-life, project identification number: 16KISK002.

Session 3: 13:30 – 14:55

Breaking Four Code-Based Cryptosystems

Stefan Ritterhoff
Technical University of Munich, Germany

The following provides cryptanalysis for four different code-based cryptosystems proposed in peer-reviewed articles over the course of the past three years. All analyzed schemes suffer from the same issue in that encryption (or signature generation) is a deterministic, linear function in the private key.

Signature scheme [70]

Signatures in the proposed scheme consist of two parts (the authors name them sig and d), where the former could be viewed as an information vector to be multiplied with the public generator matrix SGP and the latter is a syndrome vector obtained from multiplying the vector $h(h(doc)) + h(doc)$ with a specific secret parity check matrix Q. Unlike the original, we write m instead of doc and σ instead of sig to ease notation. The authors define the vector $d = h(h(m))Q + s$, where $s = h(m)Q$, so

$$d = h(h(m))Q + h(m)Q = (h(h(m)) + h(m))Q.$$

The vector σ satisfies $\sigma SGP = h(m) + h(m)p_2Q = h(m)(I + p_2Q)$, where I is the identity matrix. After solving with the pseudoinverse $(SGP)^\dagger$ we observe that σ can be computed as $\sigma = h(m)(I + p_2Q)(SGP)^\dagger = h(m)(I + p_2Q)p_1^\dagger$.

For a given public key (p_1, p_2, p_3), only the matrix Q needs to be recovered to forge a signature. Knowledge of the remaining parts of the private key (S^{-1}, P^{-1}, G) is not necessary. Thus, we want to recover the secret matrix Q.

By collecting and stacking the (column) vectors d_i for n different messages m_i, we can construct the following matrix equation:

$$\begin{bmatrix} d_1 \\ d_2 \\ \vdots \\ d_n \end{bmatrix} = \underbrace{\begin{bmatrix} (h(h(m_1)) + h(m_1)) \\ (h(h(m_2)) + h(m_2)) \\ \vdots \\ (h(h(m_n)) + h(m_n)) \end{bmatrix}}_{M \in \mathbb{F}_2^{n \times n}} Q.$$

The square matrix M will be invertible with high probability (assuming the bits produced by hashing are close to uniformly random). We can then solve for the private key Q by left-multiplying with the inverse M^{-1}.

$$\begin{bmatrix} (h(h(m_1)) + h(m_1)) \\ (h(h(m_2)) + h(m_2)) \\ \cdots \\ (h(h(m_n)) + h(m_n)) \end{bmatrix}^{-1} \begin{bmatrix} d_1 \\ d_2 \\ \vdots \\ d_n \end{bmatrix} = Q.$$

It may occur that the matrix M is not invertible. However, by collecting just a few additional (message, signature) pairs we can ensure that our success probability quickly converges to 1.

Signature scheme [8]

Here, the public key is a parity check matrix H and some invertible matrix M^{-1}, while private key is a matrix $H' = H^\top M$ and a vector h. The signature c for a message vector m is generated as $c = mH' + h$.

The private key can be recovered as follows. Inverting the third component of the public key yields M. By multiplication with $H^\top$ (also from the public key), we get H', the first part of the private key. Given any valid signature pair (m, c), we recover the rest of the private key (meaning the vector h) by subtracting the syndrome vector mH', so $h = c - mH'$.

The system in [107]

As noted by the author of the scheme, the cost of breaking the system is equal to that of inverting some non-singular matrix $T \in \mathbb{F}_q^{n \times n}$. For most sensible choices of n, however, this is *not* computationally hard (time cost $O(n^{2.373})$). In fact, it is even part of the key generation.

The system in [59]

To encrypt, the authors propose to start with some binary message of length $2n$, e.g. $m = (10111001110100)$ and split it into two halves

$$u = (1011100) \ \text{ and } \ s = (1110100).$$

Then, each pair of bits $(u_i, s_i) \in \mathbb{F}_2^2$ is mapped (bijectively) to an element in the commutative ring $\mathbb{F}_2 + \nu\mathbb{F}_2$ (where $\nu^2 = \nu$) using the relation

$$(u_i, s_i) \mapsto (1 + \nu)u_i + \nu s_i = u_i + (u_i + s_i).\nu$$

In matrix-vector notation over $\mathbb{F}_2$ this is $\begin{bmatrix} u_i & s_i \end{bmatrix} \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} a_i & b_i \end{bmatrix}$. Each of these symbols from $\mathbb{F}_2 + \nu\mathbb{F}_2$ is then transformed back to $\mathbb{F}_2^2$ using another linear isometry (Gray map) $\psi : \mathbb{F}_2 + \nu\mathbb{F}_2 \to \mathbb{F}_2^2, \quad a + b\nu \mapsto (b, a + b)$. Once again, we can write this in matrix-vector notation over $\mathbb{F}_2$ as $\begin{bmatrix} a_i & b_i \end{bmatrix} \begin{bmatrix} 0 & 1 \\ 1 & 1 \end{bmatrix} = \begin{bmatrix} c_i & c_{i+1} \end{bmatrix}$. Both maps are linear and we can write encryption as: $\begin{bmatrix} u_i & s_i \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix} = \begin{bmatrix} c_i & c_{i+1} \end{bmatrix}$. As with encryption, we can "decrypt" pairs of ciphertext symbols (c_{2i-1}, c_{2i}) for $i \in [1 \dots n]$ to plaintext symbols (m_i, m_{n+i}) by multiplying with the inverse of $D = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}$. Over $\mathbb{F}_2$ it holds that $D^2 = I$, which means this matrix is its own inverse (it is involutory). Thus, (right-)multiplication of any ciphertext vector $\begin{bmatrix} c_1 & c_2 & c_3 & c_4 & \cdots & c_{2n-1} & c_{2n} \end{bmatrix}$ with a block-diagonal matrix of the form

$$D = \begin{bmatrix} D & 0 & \dots & 0 \\ 0 & D & \dots & 0 \\ \vdots & \vdots & \ddots & 0 \\ 0 & 0 & 0 & D \end{bmatrix},$$

will produce a sequence of interleaved plaintext bits

$$\begin{bmatrix} u_1 & s_1 & u_2 & s_2 & \cdots & u_n & s_n \end{bmatrix} = \begin{bmatrix} m_1 & m_{n+1} & m_2 & m_{n+2} & \cdots & m_{n-1} & m_{2n} \end{bmatrix}.$$

Encryption and decryption are just permutation/interleaving of the message bits followed by a linear map.

Acknowledgements

Stefan Ritterhoff was supported by the German Research Foundation (Deutsche Forschungsgemeinschaft, DFG) under Grant No. WA3907/4-1, the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (grant agreement no. 801434), and acknowledges the financial support by the Federal Ministry of Education and Research of Germany in the programme of "Souverän. Digital. Vernetzt.". Joint project 6G-life, project identification number: 16KISK002

SDitH in Hardware

Sanjay Deshpande
Yale University, USA and SandboxAQ, USA

Joint work with: James Howe, Jakub Szefer, Dongze Yue

Abstract This work presents the first hardware realisation of the Syndrome-Decoding-in-the-Head (SDitH) signature scheme, which is a candidate in the NIST PQC process for standardising post-quantum secure digital signature schemes. SDitH's hardness is based on conservative code-based assumptions, and it uses the Multi-Party-Computation-in-the-Head (MPCitH) construction. This is the first hardware design of a code-based signature scheme based on traditional decoding problems and only the second for MPCitH constructions, after Picnic. This work presents hardware optimised designs to achieve the best area efficiency, which we evaluate using the Time-Area Product (TAP) metric. This work also proposes a novel hardware architecture by dividing the signature generation algorithm into two phases, namely offline and online phases for optimising the overall clock cycle count. The hardware designs for key generation, signature generation, and signature verification are parameterised for all SDitH parameters, including the NIST security levels, both syndrome decoding base fields (GF256 and GF251), and thus conforms to the SDitH specifications. The hardware design further supports secret share splitting, and the hypercube optimisation which can be applied in this and multiple other NIST PQC candidates. The results of this work result in a hardware design with a drastic reducing in clock cycles compared to the optimised AVX2 software implementation, in the range of 2-4x for most operations. Our key generation outperforms software drastically, giving a 11-17x reduction in *runtime*, despite the significantly faster clock speed. On Artix 7 FPGAs we can perform key generation in 55.1 Kcycles, signature generation in 6.7 Mcycles, and signature verification in 8.6 Mcycles for NIST L1 parameters, which increase for GF251, and for L3 and L5 parameters.

Overview of Our Contributions

The SDitH signature scheme [7] is a relatively new proposal, with this research being the first presentation of its design in hardware. SDitH is based on conservative code-based hardness assumptions and utilises the (Multi-Party Computation in the Head) MPCitH paradigm. Since SDitH is a candidate in the NIST PQC process for additional signatures, this work helps to establish a basis upon which it can be compared to the current NIST PQC signature standards, which have hardware design. Following is the list of contributions of this work:

1. The first hardware design of the SDitH signature scheme for the hypercube variant, and using all proposed parameter sets. All hardware designs are specification compliant, constant-time[1], and also parameterisable in terms of the security level (λ), syndrome decoding field size (q), share splitting size (d), the repetition rate (τ), and the random evaluation points (t) parameters.
2. We design an optimised sample and matrix-vector multiplication core, *syndrome_decoding*, for use in key generation, signature generation, and signature verification. It also includes

[1] The randomness sampling (SampleFieldElements) module has variable runtime, but this only affects public information. This conforms to the specification and the reference implementation. We elaborate on this throughout the paper [49].

two design variations, sample-first-then-multiply (SFTM) and sample-and-multiply-on-the-fly (SaMO).

3. We exploit the nature of the SDitH signature generation design by providing a parametrisable option to split signature generation into two stages, namely *offline* and *online*, in order to hide many of the clock cycles required. Through this we reduce clock cycles by 27-33% with an additional cost of 30%-60% in BRAMs.
4. While we design separate hardware modules for key generation, signature generation, and signature verification operations, we provide the capability that they all could be combined into one design consisting of all operations together.
5. For NIST security levels L3 and L5, we take advantage of d-splitting size syndrome decoding parameter, which adds more parametrisable options and allows the scope of additional parallelism, specifically in SampleWitness, ComputePlainBroadcast, and PartyComputation modules.
6. Many of the sub-modules designed could also be useful in other MPCitH-based schemes or those which employ the hypercube optimisation.
7. We evaluate the resource requirements of our hardware designs on a Xilinx Artix-7 (*xc7a200t*) FPGA, as recommended by NIST for the PQC standardisation process.
8. The hardware code will be made open-source and released under an Apache-2.0 licence, available at: `https://github.com/sandbox-quantum/sdith-impl-hw`.

Hardware Design of SDitH and Results

In most other software and hardware designs of NIST PQC candidates, SHAKE is known to be a bottleneck. But in our area optimised hardware implementation of SDitH primitives we note that the bottleneck is not the SHAKE-256 but the polynomial evaluation module (Evaluate) which contributes to 99% clock cycles in sign (sign_online) and 70%-90% clock cycles in verification depending on the choice of security level and underlying arithmetic field. This adds a distinctive elements to SDitH and its hardware design. Additionally, its feature of being able to be split into offline and online phases illustrates its potential of being useful in many use cases, setting it apart from other NIST PQC candidates.

From Table 2, we highlight that our SDitH-GF256 hardware implementation is of the smallest area footprint when compared to all other designs. Our SDitH-GF251 also uses less area but uses DSP resources for optimising the underlying arithmetic operations. However, our hardware designs use significant BRAM as it is unavoidable due to the nature of the SDitH signature scheme. When comparing the overall performance we note that Dilithium clearly outperforms all other designs. However, it may not be fair to compare the lattice-based schemes against those using MPCitH. A more relevant comparison would be with Picnic, in which case our design uses much less area while implementing all primitives. While we acknowledge that the time taken by the Picnic design to sign and verify is better compared to that of our design, the Picnic implementation uses a reduced data complexity design using a LowMC, compared to the more conservative code-based hardness assumption in SDitH.

We note that this work has been accepted at IACR Transactions on Cryptographic Hardware and Embedded Systems (TCHES)[2] and the accepted full-version version is available on eprint [49] for reviewers' reference.

[2] https://tches.iacr.org/

Table 2: Resource and Performance comparison of our *complete* SDitH hardware design with other related PQC signature hardware designs for different security levels. †Does not include Key Generation and ‡Includes only Signature Generation.

Parameter		FPGA Utilisation			Frequency	KeyGen		Sign		Verify	
Sets	LUT	FF	DSP	BRAM	*(MHz)*	Latency	Time	Latency	Time	Latency	Time
						(Mcycles)	*(ms)*	*(Mcycles)*	*(ms)*	*(Mcycles)*	*(ms)*
SDitH-L1-GF256 (Ours, [49])	16,592	8,778	0	164.5	164	0.055	0.34	6.73	41.04	8.688	52.98
SDitH-L1-GF251 (Ours, [49])	17,423	16,336	196	164.5	164	0.057	0.35	21.45	130.77	23.403	142.71
SDitH-L3-GF256 (Ours, [49])	22,569	13,881	0	356.0	164	0.046	0.28	7.74	47.17	10.960	66.85
SDitH-L3-GF251 (Ours, [49])	29,961	25,794	382	356.0	164	0.047	0.29	24.60	149.99	24.600	149.99
SDitH-L5-GF256 (Ours, [49])	23,323	14,962	0	520.5	164	0.083	0.51	17.48	106.60	24.940	152.10
SDitH-L5-GF251 (Ours, [49])	34,456	31,409	472	521.5	164	0.086	0.53	45.28	276.07	30.110	183.57
PICNIC-L1† [63]	90,337	23,105	0	52.5	125	–	–	0.03	0.25	0.030	0.24
PICNIC-L5† [63]	167,530	33,164	0	98.5	125	–	–	0.15	1.24	0.147	1.17
SPHINCS^{+}-128s-simple‡ [10]	48,231	72,514	0	11.5	250 & 500	–	–	–	12.40	–	0.07
SPHINCS^{+}-128f-simple‡ [10]	47,991	72,505	1	11.5	250 & 500	–	–	–	1.01	–	0.16
SPHINCS^{+}-192s-simple‡ [10]	48,725	72,514	0	17.0	250 & 500	–	–	–	21.40	–	0.10
SPHINCS^{+}-192f-simple‡ [10]	48,398	73,476	1	17.0	250 & 500	–	–	–	1.17	–	0.19
SPHINCS^{+}-256s-simple‡ [10]	51,130	74,576	1	22.5	250 & 500	–	–	–	19.30	–	0.14
SPHINCS^{+}-256f-simple‡ [10]	51,009	74,539	1	22.5	250 & 500	–	–	–	2.52	–	0.21
Dilithium-L2 [106]	29,998	10,336	10	11.0	97	0.004	0.04	0.03	0.29	0.004	0.05
Dilithium-L3 [106]	29,998	10,336	10	11.0	97	0.006	0.06	0.04	0.46	0.006	0.06
Dilithium-L5 [106]	29,998	10,336	10	11.0	97	0.009	0.09	0.05	0.51	0.009	0.09
LESS-L1 {b} [31]	54,800	39,900	0	59.5	200	0.029	0.14	5.20	26.02	5.156	25.78
LESS-L1 {i} [31]	54,800	39,900	0	59.5	200	0.077	0.38	5.13	25.63	5.093	25.47
LESS-L1 {s} [31]	54,800	39,900	0	59.5	200	0.174	0.87	4.17	20.83	4.137	20.69
LESS-L3 {b} [31]	76,700	57,900	0	102.5	167	0.072	0.43	39.24	234.95	39.146	234.87
LESS-L3 {s} [31]	76,700	57,900	0	102.5	167	0.132	0.79	46.22	276.75	46.142	276.85
LESS-L5 {b} [31]	104,300	76,700	0	167.5	143	0.134	0.93	129.89	909.20	129.726	908.08
LESS-L5 {s} [31]	104,300	76,700	0	167.5	143	0.247	1.73	87.16	610.13	87.013	609.09

Public-Key Encryption based on Supercode Decoding

Anmoal Porwal
Technical University of Munich, Germany

Joint work with: Anna Baumeister, Violetta Weger, Antonia Wachter-Zeh, Pierre Loidreau

Keywords: Code-based cryptography, Error-erasure decoding

Abstract We describe a public-key encryption framework based on codes with efficient error-erasure decoders. This scheme significantly differs from the McEliece framework, has a different hardness assumption (in particular, the code is public), and potentially better parameters. The security depends crucially on the chosen code family. It is an open question whether a secure choice exists.

Introduction

We discuss a public-key encryption scheme based on codes with efficient error-erasure decoders. The framework is a generalization of specific schemes [19, 54, 67, 93], that have been broken in the past. Unlike the McEliece framework, this scheme does not rely on hiding the structure of a secret code. Further, the scheme has potentially much smaller public keys, thus overcoming a major drawback of the McEliece system. A possible disadvantage of the new scheme is that it requires large field sizes.

Error-erasure decoding here refers to correcting erasures and errors up to a certain radius where it is easier to correct erasures than errors. For example, for many codes, it is possible to decode w errors and w' erasures whenever $2w + w' < d_{\min}$, where $d_{\min}$ denotes the minimum distance of the code.

In general, our framework is parameterized by the choice of a code and its security crucially depends on it. The following sections describe the framework, its underlying hard problem and its past instantiations.

The Framework

We keep the metric unspecified, as one can use either the Hamming or rank metric. We denote the support of a vector $\mathbf{x} \in \mathbb{F}_{q^m}^n$ by $\operatorname{supp}(\mathbf{x})$. For example, in the Hamming metric the support is simply the indices of the non-zero entries of $\mathbf{x}$, while in the rank metric, the support can refer to the row space of $\mathbf{x}$.

Fix a generator matrix $\mathbf{G} \in \mathbb{F}_{q^m}^{k \times n}$ of a code $\mathcal{C}$ that has an efficient error-erasure decoder. This code and decoder are public knowledge. For the key generation (Algorithm 1), Alice draws random vectors $\mathbf{m}' \in \mathbb{F}_{q^m}^k$ and $\mathbf{e}' \in \mathbb{F}_{q^m}^n$ such that $\operatorname{wt}(\mathbf{e}') = w'$ for a large w'. The public key $\boldsymbol{\rho}$ is $\mathbf{m}'\mathbf{G} + \mathbf{e}'$ and the secret key $\boldsymbol{\kappa}$ is $\mathbf{e}'$. To encrypt a message $\mathbf{m} \in \mathbb{F}_{q^m}^k$ (Algorithm 2), Bob picks a random $\alpha \in \mathbb{F}_{q^m}^*$ and a random weight-w vector $\mathbf{e} \in \mathbb{F}_{q^m}^n$ with w being small. He then computes the ciphertext $\mathbf{r} = \mathbf{m}\mathbf{G} + \alpha\boldsymbol{\rho} + \mathbf{e}$. For decryption (Algorithm 3), note that $\mathbf{r}$ is a corrupted codeword with error $\alpha\mathbf{e}' + \mathbf{e}$. Since Alice knows $\operatorname{supp}(\mathbf{e}') = \operatorname{supp}(\alpha\mathbf{e}')$, she can perform error-erasure decoding to obtain $\alpha\mathbf{e}' + \mathbf{e}$ using the known erasures $\operatorname{supp}(\mathbf{e}')$. With $\alpha\mathbf{e}' + \mathbf{e}$, it is possible to find $\mathbf{e}$ and hence, equivalently, $\mathbf{m}$.

For concreteness, assume that the Hamming metric and a Reed-Solomon (RS) code are used. To decrypt, Alice first error-erasure decodes $\mathbf{r}$. For Reed-Solomon codes, this can be done as follows.

Note that $\mathbf{r} = \mathbf{m}\mathbf{G} + \alpha\boldsymbol{\rho} + \mathbf{e} = (\mathbf{m} + \alpha\mathbf{m}')\mathbf{G} + (\alpha\mathbf{e}' + \mathbf{e})$. If Alice punctures this word at $\mathrm{supp}(\mathbf{e}')$, she is then left with a corrupted codeword of error weight at most w in a punctured RS code. Assuming proper choices for w' and w, she can decode this to obtain $\mathbf{m} + \alpha\mathbf{m}'$ or equivalently $\alpha\mathbf{e}' + \mathbf{e}$. Since $w < w'$, several entries of $\alpha\mathbf{e}' + \mathbf{e}$ equal $\alpha e'_i$ and hence reveal α, and in turn $\mathbf{e}$. A more involved (and probabilistic) extraction procedure also exists for Gabidulin codes in the rank metric.

While Alice needs to decode w' erasures and w errors, an attacker needs to solve an error-only decoding problem with error weight at least w'. Assuming suitable parameters and choice of code, the former problem is easy while the latter is assumed to be hard.

```
Algorithm 1: Keygen
  Input: security level
  Output: public key $\boldsymbol{\rho}$, secret key $\boldsymbol{\kappa}$
  $\mathbf{e}' \xleftarrow{\$} \mathbb{F}_{q^m}^n$ s.t. $\mathrm{wt}(\mathbf{e}') = w'$
  $\mathbf{m}' \xleftarrow{\$} \mathbb{F}_{q^m}^k$
  Set $\boldsymbol{\rho} = \mathbf{m}'\mathbf{G} + \mathbf{e}'$
  Set $\boldsymbol{\kappa} = \mathbf{e}'$
  return $(\boldsymbol{\rho}, \boldsymbol{\kappa})$
```

```
Algorithm 2: Encryption
  Input: message $\mathbf{m} \in \mathbb{F}_{q^m}^k$, public
    key $\boldsymbol{\rho}$
  Output: ciphertext $\mathbf{r}$
  $\alpha \xleftarrow{\$} \mathbb{F}_{q^m}^*$
  $\mathbf{e} \xleftarrow{\$} \mathbb{F}_{q^m}^n$ s.t. $\mathrm{wt}(\mathbf{e}) = w$
  Set $\mathbf{r} = \mathbf{m}\mathbf{G} + \alpha\boldsymbol{\rho} + \mathbf{e}$
  return $\mathbf{r}$
```

```
Algorithm 3: Decryption
  Input: ciphertext $\mathbf{r}$, secret key $\boldsymbol{\kappa}$
  Output: message $\mathbf{m}$ or failure
  Set $\mathbf{x} = \mathrm{error-erasure-dec}(\mathbf{r}, \mathrm{supp}(\boldsymbol{\kappa}))$  // $\mathbf{x}$ is $\alpha\mathbf{e}' + \mathbf{e}$
  Extract $\mathbf{e}$ from $\mathbf{x}$
  Solve the system $[\hat{\mathbf{m}}\ \ \hat{\alpha}]\begin{bmatrix}\mathbf{G}\\ \boldsymbol{\rho}\end{bmatrix} = \mathbf{r} - \mathbf{e}$ for $[\hat{\mathbf{m}}\ \ \hat{\alpha}]$
  return $\hat{\mathbf{m}}$
```

Hardness Assumption

Let $\mathcal{S}_w$ denote the set of vectors in $\mathbb{F}_{q^m}^n$ of weight w. One can show the OW-CPA (one-wayness under chosen plaintext attack) security of the scheme is equivalent to the following problem. The problem is parameterized by a public generator matrix $\mathbf{G} \in \mathbb{F}_{q^m}^{k \times n}$ and the integers $w' > w$.

Problem 4 (Supercode Decoding $(\mathbf{G}, w, w')$). *Draw* $\mathbf{c}' \in \langle\mathbf{G}\rangle$ *and* $\mathbf{e}' \in \mathcal{S}_{w'}$ *uniformly at random. Set* $\boldsymbol{\rho} = \mathbf{c}' + \mathbf{e}'$. *Draw* $\mathbf{c}^\dagger \in \left\langle \begin{bmatrix}\mathbf{G}\\ \boldsymbol{\rho}\end{bmatrix} \right\rangle \setminus \langle\mathbf{G}\rangle$ *and* $\mathbf{e} \in \mathcal{S}_w$ *uniformly at random. Set* $\mathbf{r} = \mathbf{c}^\dagger + \mathbf{e}$. *Given only* $\mathbf{G}$, $\boldsymbol{\rho}$ *and* $\mathbf{r}$, *find* $\mathbf{e}$.

The main question that arises is: does there exist a $\mathbf{G}$ and $w' > w$ such that one can error-erasure decode for erasure weight w' and error weight w while the above problem remains hard? If so, then the described framework with these choices is an OW-CPA secure public-key encryption scheme.

Past Schemes and Variations

Our framework can be seen as a generalization of the Augot-Finiasz (AF) system [19] and its rank-metric counterpart the Faure-Loidreau (FL) system [54]. The former employs RS codes, while Gabidulin codes are used in the latter. The hardness assumption 4 asks whether decoding in a certain supercode of $\mathbf{G}$ is difficult. For both RS and Gabidulin, this is in fact not the case, as shown in [47, 56, 38] and hence these schemes were broken. The attack is based on a modified Welch-Berlekamp decoding algorithm in order to incorporate the additional basis ρ of the supercode $\left\langle \begin{bmatrix} \mathbf{G} \\ \boldsymbol{\rho} \end{bmatrix} \right\rangle$. This works precisely because RS codes, respectively Gabidulin codes, are polynomial, respectively q-polynomial, evaluation codes. This suggests that code classes without this particular structure would be viable candidates for this framework.

There is a variation of this framework which employs "interleaving" and implies augmenting the supercode with multiple words in the hard problem instead of just one. This allows using binary codes which has the added benefit that an error decoder for a binary code can always be efficiently converted to an erasure-error decoder. However, this has the disadvantage that the underlying hard problem is more complex to analyse. This variation is the same as the modification presented for the AF/FL system where the trace operator is used in an attempt to (unsuccessfully) thwart the modified Welch-Berlekamp attack.

Finally, we note a small difference in the AF/FL system compared to our framework: the AF/FL system requires the highest order bit of $\mathbf{m}$ to be zero in the encryption step which allows them to use a slightly different decryption procedure. However, we see no benefit of this difference and, further, this has the the drawback of making the hard problem less simple.

Acknowledgements

This work has been supported by the ANR-DFG project "CROWD: Cryptography with Skew Codes", project number: WA3907/9-1. V. Weger is supported by the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie (Grant agreement No. 899987). A. Wachter-Zeh is supported by the Bavarian State Ministry of Science and Arts via the project EQAP.

Session 4: 15:30 – 16:55

On the Rank of Random Binary Sub-Matrices and its Impact for Information Set Decoding Algorithms

Bénédikt Tran
Ecole Polytechnique Fédérale de Lausanne Switzerland

Keywords: Information Set Decoding, Code-based Cryptography, Post-Quantum Cryptography.

Abstract Information set decoding (ISD) is a technique for solving the syndrome decoding problem for random linear codes and many works focused on improving the original Prange algorithm. We present an ongoing research focusing on using an algebraic property of the rank of random submatrices to improve quantum Prange ISD algorithm.

Introduction

The *syndrome decoding problem* (SDP) is a hard problem in code-based cryptography and known to be NP-complete for random binary linear codes [33]. Given a parity-check matrix $\mathbf{H} \in \mathbb{F}_2^{(n-k)\times n}$ of a binary linear $[n, k, d]$-code of length n, dimension k and minimal distance d, a syndrome $\mathbf{s} \in \mathbb{F}_2^{n-k}$ and a target weight $w \geq 0$, the goal is to find an error vector $\mathbf{e} \in \mathbb{F}_2^n$ such that $\mathbf{He} = \mathbf{s}$ and $\mathrm{wt}(\mathbf{e}) = w$. One technique for solving SDP is based on *information set decoding* (ISD), first introduced by Prange [91] in 1962 and later improved by [68, 101, 50, 71, 30, 40] and [34, 62] which essentially consists in finding the position of the error bits. While many time-memory trade-offs, both for classical or quantum algorithms, have been suggested, little research has been done on exploitable algebraic properties of random codes. This work aims to improve Prange's ISD algorithm in practice by relying on a heuristic property of random matrices.

Notations

For $a \leq b \in \mathbb{Z}$, we write $[\![a, b]\!] = \{a, \dots, b\}$ and $[\![a]\!] = [\![1, a]\!]$. For $\mathbf{H} \in \mathbb{F}_2^{m\times n}$, define $\ker \mathbf{H} \triangleq \{\mathbf{x} \in \mathbb{F}_2^n \colon \mathbf{Hx} = \mathbf{0}\}$. Given $(I, J) \subseteq [\![m]\!] \times [\![n]\!]$, we denote by $\mathbf{H}[I|J]$ the $|I| \times |J|$ submatrix of $\mathbf{H}$ formed by the rows and columns of $\mathbf{H}$ indexed by I and J respectively. The uniform distribution over $\mathbb{F}_2^{n\times n}$ is denoted by $\mathcal{U}_n$ and $\mathcal{D}_n^k$ denotes the uniform distribution over the k-sized subsets $\mathcal{P}_n^k = \{I \subseteq [\![n]\!] \colon |I| = k\}$ of $[\![n]\!]$.

For $\mathbf{H} \in \mathbb{F}_2^{n_1\times n_2}$, let $E_{\mathrm{H}}^{k_1,k_2} = \left\{k - 4 \leq \mathrm{rk}(\mathbf{H}[I|J]) \leq 4 \colon (I, J) \sim \mathcal{D}_{n_1}^{k_1} \otimes \mathcal{D}_{n_2}^{k_2}\right\}$, where $k = \min(k_1, k_2)$ and the randomness of E_{H} is characterized by the random coins $\xi_{\mathcal{P}}$ used to sample (I, J). A matrix $\mathbf{H} \in \mathbb{F}_2^{n_1\times n_2}$ is *good* if $\Pr[E_{\mathrm{H}}^{k_1,k_2}]$ is close to 1 for all (k_1, k_2) and a distribution $\mathcal{H}$ over $\mathbb{F}_2^{n_1\times n_2}$ is *good* if it produces good matrices with probability close to 1.

Let $\mathcal{H}$ be a good distribution over $\mathbb{F}_2^{n\times n}$ and $\ell_1, \ell_2 \in [\![n]\!]$. Let $\mathbf{H} \sim \mathcal{H}$ and $\mathbf{S}$ be a random $\ell_1 \times (n - \ell_2)$ submatrix of $\mathbf{H}$. By the rank-nullity theorem, we expect $\dim \ker \mathbf{S} \leq 4 + \max(0, n - (\ell_1 + \ell_2))$. By [66, Eq. 3.2.2], the rank of a uniform random $n \times n$ matrix is in $[n - 4, n]$ and we experimentally observed that the rank of a uniform random $k \times k$ submatrix of a uniform $n \times n$ matrix is in $[k - 4, k]$ with high probability. This motivates the following conjecture.

Conjecture 5. *$\mathcal{U}_n$ is a good distribution.*

In particular, any procedure that finds an element in $\ker \mathbf{S}$ may instead bruteforce the kernel itself if the dimension is small enough and that dimension can be controlled by the choices of ℓ_1 and ℓ_2.

Application

Given a permutation matrix $\mathbf{P} \in \mathrm{GL}_n(\mathbb{F}_2)$ and $\mathbf{R} \in \mathrm{GL}_{n-k}(\mathbb{F}_2)$, recall that $\mathrm{SDP}(\mathbf{H}, \mathbf{s}, w)$ and $\mathrm{SDP}(\hat{\mathbf{H}} = \mathbf{RHP}, \boldsymbol{\sigma} = \mathbf{Rs}, w)$ are equivalent. Usually, ISD algorithms rely on solving a linear system of m equations of the form

$$\mathbf{A}_1\mathbf{u}_1 \oplus \mathbf{A}_2\mathbf{u}_2 = \boldsymbol{\sigma} \tag{$\star$}$$

with unknowns $\mathbf{u}_i \in \mathbb{F}_2^{\ell_i}$ subject to $\mathrm{wt}(\mathbf{u}_1) + \mathrm{wt}(\mathbf{u}_2) = w$. For instance, Lee-Brickell's algorithm [68] computes the systematic form $\hat{\mathbf{H}} = \begin{bmatrix}\mathbf{Q} & \mathbf{I}_{n-k}\end{bmatrix}$ of $\mathbf{H}$ and chooses $(\mathbf{A}_1, \mathbf{A}_2) = (\mathbf{Q}, \mathbf{I}_{n-k})$. Instead of guessing the error bits of $\mathbf{u}_1$ and $\mathbf{u}_2$ as for ISD algorithms, we suggest guessing their zero bits positions as follows. Up to applying a Gauss-Jordan elimination, we may assume that $\mathbf{A}_1 = \mathbf{A} \in \mathbb{F}_2^{m\times\ell}$ and $\mathbf{A}_2 = \mathbf{I}_m$, and reformulate $(\star)$ as $\mathbf{A}\mathbf{u}_1 \oplus \mathbf{u}_2 = \boldsymbol{\sigma}$.

Assume that $L_1 \subseteq [\![\ell]\!]$ and $L_2 \subseteq [\![m]\!]$ satisfy $\mathbf{u}_{ij} = 0$ for all $j \in L_i$ and let $m' = |L_2|$ and $\ell' = |L_1^c| = \ell - |L_1|$. Let $\mathbf{A}' \triangleq \mathbf{A}[L_2|L_1^c] \in \mathbb{F}_2^{m'\times\ell'}$. By construction, $\mathbf{v}_1' \triangleq (\mathbf{u}_1)_{L_1^c}$ satisfies $(\mathbf{A}'\mathbf{v}_1')_j = \sigma_j$ for all $j \in L_2$ and $\mathrm{wt}(\mathbf{v}_1') = \mathrm{wt}(\mathbf{u}_1)$. For $\boldsymbol{\sigma} = \mathbf{0}$, namely $\mathbf{v}_1' \in \ker \mathbf{A}'$, choose $\mathbf{v}_1 \in \mathbb{F}_2^{\ell}$ such that $(\mathbf{v}_1)_{L_1^c} = \mathbf{v}_1'$ and 0 otherwise and check that $\mathbf{v}_2 = \mathbf{A}\mathbf{v}_1$ satisfies $\mathrm{wt}(\mathbf{v}_1) + \mathrm{wt}(\mathbf{v}_2) = w$. If L_1 and L_2 are correctly chosen, any solution $(\mathbf{v}_1, \mathbf{v}_2)$ to $(\star)$ would arise from some $\mathbf{v}_1' \in \ker \mathbf{A}'$. While we failed to generalize this technique to $\boldsymbol{\sigma} \neq \mathbf{0}$, we explain the practical relevance of the simplest case for quantum Prange.

In **BIKE** [16], the public key $\mathbf{A} = \mathbf{U}_2\mathbf{U}_1^{-1}$ is the product of private invertible circulant matrices with a small constant row weight $\frac{w}{2}$. Due to the limitations imposed by NIST on quantum resources, only quantum Prange [34] has been considered so far. By [53], the number of qubits for quantum Prange is $\Omega(k(n-k))$. If we search for L_1^c and L_2, storing $\mathbf{A}[L_2|L_1^c]$ and (L_1^c, L_2) require $\Omega(m'\ell' + \log\binom{\ell}{\ell'}\binom{m}{m'})$ qubits. For **BIKE**, we have $k = \frac{n}{2}$ and $m = \ell = k$. With $m' = \lceil m/2 \rceil$ and $\ell' = \ell - \lceil \ell/2 \rceil \leq m'$, finding (L_1, L_2^c) such that an element of $\ker \mathbf{A}[L_2|L_1^c]$ (which we expect to have dimension ≤ 4) gives rise to a solution has a quantum complexity of order $\binom{m}{m'}\binom{m-w/2}{m'}^{-1} \cdot T_G(m', \ell')$ against $\sqrt{\binom{n}{w}/\binom{n-k}{w}} \cdot T_G(k, k)$ for quantum Prange, where $T_G(p, q) = \frac{3}{2}p^2q - \frac{3}{4}p^3$ if $p \leq q$ and $\frac{3}{4}pq^2$ otherwise. For **BIKE-128** parameters $(k, w) = (12323, 142)$, this translates into $\mathcal{O}(2^{104.482})$ and $\mathcal{O}(2^{107.614})$ respectively, both requiring $\Omega(n^2/4)$ qubits.

A Generalized Framework for Folding Codes and Applications to Proofs of Proximity

Adrien Pasquereau
University of St.Gallen, Switzerland

Keywords: Proof of Proximity, Code Equivalence, Group Actions.

Abstract Deciding whether a vector belongs to a given error-correcting code or whether it is far from the code is a problem known as proximity testing. Over the last decade, two-party proximity testing protocols to prove the knowledge of a codeword have found a particular interest in designing succinct non-interactive arguments, along with applications in verifiable computation fulfilling post-quantum security. The framework of Interactive Oracle Proof of Proximity (IOPP) models the semantics of such probabilistic proof protocols and provides metrics to evaluate their efficiency. In particular, building protocols with very efficient, *i.e.*, poly-logarithmic verification time, is considered a central challenge to achieving scalable IOPPs. We follow a line of work that started with the FRI protocol of Ben-Sasson *et al.* [32] for Reed-Solomon codes, and more recently with the work of Bordage *et al.* [39] for Algebraic Geometry codes over Kummer curves and the Hermitian tower. We propose a generalization of these protocols for a wider class of codes satisfying some simple conditions associated with pairs of linear representations. In this talk, we will extend the core idea of recursively folding a code into a sequence of codes of reduced length and dimensions by translating the behavior of the protocol to the language of linear algebra and group actions. Based on this correspondence, we aim to identify a set of necessary conditions on the code's suitability for folding in the previous fashion by reformulating the construction of the code's folding sequence and the execution of the protocol in terms of the tested code's generator matrix. Moreover, we investigate how some sufficient properties (in particular, the assumptions for foldable Reed-Solomon codes and Algebraic Geometry codes) impact the soundness and efficiency of the protocol and classify tradeoffs achievable on the more generic folding-friendly codes. As a work in progress, we discuss the feasibility of folding algorithms in other metrics and provide insights for when the protocol fails to generalize.

Extended Abstract

We start from the same observation as in [39]; the action of a cyclic subgroup Γ of the automorphism group of a curve $\mathcal{C}$ lifts to an $\mathbb{F}_q$-linear action of Γ on $\mathcal{L}_{\mathcal{C}}(D)$, provided that D is a Γ-invariant divisor. If $\mathcal{C}$ has an appropriate geometric structure, Γ can be chosen to be a cyclic subgroup whose order divides $q-1$, *i.e.*, Γ can be seen as a subgroup of $\mathbb{F}_q^\times$. As a direct consequence, given a generator of the group acting on the space $\mathcal{L}_{\mathcal{C}}(D)$, this space splits as a direct sum of the eigenspaces associated with the corresponding primitive roots of unity. The authors of [39] express these eigenspaces as Riemann-Roch spaces over a quotient curve with compatible divisors, showing how to lift a composition sequence of groups and quotient curves to a series of folding operators between folded subcodes (with respect to some well-chosen evaluation points).

We aim to generalize the heuristic of lifting a composition series of groups to a family of folded codes, but we focus on the groups' representations instead of the function fields. For example, in the previous functional evaluation setting, the pullback of an automorphism of the curve could be seen as a linear operator on the Riemann-Roch space, that can, up to the choice of a basis, be represented as a $k \times k$ matrix over $\mathbb{F}_q$. On the other hand, the construction of the folding also relies on identifying points within the same orbits together to decrease the next code's length. For the IOPP protocols with respect to Reed-Solomon or Algebraic Geometry codes, the central assumption

is that the set of evaluation points can be partitioned into a collection of full orbits. We will identify the action of the automorphism on this set with a permutation of $\{1, \ldots, n\}$, which is represented by an $n \times n$ matrix over $\mathbb{F}_q$ (which is, in particular, a Hamming-isometry of the ambient space). As a consequence, we obtain a pair of $\mathbb{F}_q$-linear representations $L : \Gamma \to \mathrm{GL}_k(\mathbb{F}_q)$ and $R : \Gamma \to \mathrm{GL}_n(\mathbb{F}_q)$ satisfying the following adjunction relation: if G is the generator matrix of the code associated with the evaluation map (up to the basis and the order of the evaluation points fixed earlier), then for every $\gamma \in \Gamma$, we have $L(\gamma) \cdot G = G \cdot R(\gamma)$.
We present an alternative way of constructing the folded codes from a pair of representations satisfying this adjunction relation. Our observation is that even though some geometric information about the curve is forgotten (Γ is simply a subgroup of the multiplicative group of the field, the curve and divisors do not need to be explicitly known), one can still computationally execute the protocol. This results in a strategy for folding more general codes: given the generator matrix $G \in \mathbb{F}_q^{k \times n}$ of a code C (not necessarily arising from functional evaluation), find a non-trivial element of the stabilizer subgroup of G under the action of $\mathrm{GL}_k(\mathbb{F}_q) \times \mathrm{GL}_n(\mathbb{F}_q)$ on $\mathbb{F}_q^{k \times n}$ via $(L, R) \cdot G = L^{-1}GR$. We show that folded subcodes can be derived from the study of such pairs; in particular, if both L and R are diagonalizable over $\mathbb{F}_q$, then C can be quasi-isometrically mapped to a direct sum of codes playing the role of local components. We provide a theorem to compute their number, dimensions, lengths, and generator matrices from the pair (L, R) and discuss how they naturally relate to the recursion step in FRI-alike IOPP protocols. What's more, we give a way of controlling the soundness error within this framework by investigating the metric properties of the mapping induced by the decomposition.

Asymptotic Cost Comparison of Generic Rank Decoders

Hugo Sauerbier Couvée
Technical University of Munich, Germany

Joint work with: Alberto Ravagnani, Antonia Wachter-Zeh, Violetta Weger

Abstract Cryptosystems based on rank-metric codes constitute a significant contender for new future standards as they promise competitive performances compared to Hamming-based systems. Many rank-metric-based systems rely on the hardness of variants of the decoding problem, and therefore, the cost of solving the Rank Decoding Problem (RDP) has to be well understood. We investigate the asymptotic cost of several algebraic attacks on the RDP proposed in [22, 23, 24] for different parameter regimes, showing that often they perform asymptotically comparable to combinatorial attacks.

Classical code-based cryptography, which was initiated with the seminal work of McEliece, is strongly connected to the NP-complete problem of decoding a random linear code in the Hamming metric. As many proposed Hamming-based cryptosystems suffer from large key sizes, systems based on decoding problems in alternative metrics, such as the rank metric, have gained considerable attention in the last decade. In the NIST Post-Quantum Standardization Process, encryption schemes like ROLLO [5] and RQC [6] demonstrated the potential of rank-based cryptography, and recently, digital signature schemes like RYDE [12], MIRA [13] and MiRitH [3] promise relatively small key and signature sizes. Many of these and other schemes like Durandal and LowMS rely on the hardness of solving (variants of) the Rank Decoding Problem (RDP) or MinRank problem. Thus, an extensive analysis of the costs of solving these problems is necessary to properly understand the security of rank-based cryptosystems.

The main question we investigate will be to determine the cost of solving the RDP: given an extension field $\mathbb{F}_{q^m}$ over a finite field $\mathbb{F}_q$, a random parity-check matrix $\boldsymbol{H} \in \mathbb{F}_{q^m}^{(n-k)\times n}$ of an $\mathbb{F}_{q^m}$-linear code of dimension k and length n, a syndrome vector $\boldsymbol{s} \in \mathbb{F}_{q^m}^{n-k}$ and a positive integer t, find an error vector $\boldsymbol{e} \in \mathbb{F}_{q^m}^{n}$ with rank weight at most t, which satisfies the syndrome equation $\boldsymbol{e}\boldsymbol{H}^\top = \boldsymbol{s}$. Several generic decoding algorithms attacking the RDP have been proposed [57], either of combinatorial or algebraic nature. Some notable combinatorial attacks developed in [17, 42, 86], each preferred in different parameter regimes, have costs up to polynomial factors of $O\big(q^{t\lceil\frac{(k+1)m}{n}\rceil-m}\big)$, $O\left(q^{(m-t)(t-1)}\right)$ and $O\left(q^{(t-1)(k+1)}\right)$ respectively, and their asymptotic costs are easily derived from these formulas.

On the other hand, recent algebraic attacks on RDP and the related MinRank problem proposed by [22, 23, 24] have been celebrated as new benchmarks with lower costs in certain parameter regimes compared to the mentioned combinatorial attacks. However, the asymptotic cost of these algorithms is understood less, and more generally, it is an important open question, which algorithm (algebraic or combinatorial) has the lowest asymptotic costs of solving RDP in each parameter regime.

For the asymptotic analysis of the cost, we consider the parameters $m = m(n)$, $k = k(n)$, $t = t(n)$ as functions in n with the following limits: $M = \lim_{n\to\infty} \frac{m(n)}{n} \in \mathbb{R}_{>0}$ and $R = \lim_{n\to\infty} \frac{k(n)}{n} \in (0, 1)$. Furthermore we consider $t(n)$ in the case that $\lim_{n\to\infty} \frac{t(n)}{\log(n)}$ is finite and larger than 0 (i.e. t grows logarithmically) and the case that it is infinite (i.e. t grows faster than logarithmically). Note that RYDE [12] decodes up to the minimum rank distance given by the Gilbert-Varshamov bound [69] and is thus using $t(n) \sim Tn$ for some $T > 0$. On the other hand, schemes based on LRPC codes such as ROLLO [5] have $t(n) \sim T\sqrt{n}$. When we consider the above parameter regimes for the combinatorial algorithms in [17, 57, 86], we find that asymptotically the cost is given by $q^{R\min\{M,1\}\,n\,t(n)}$, up to linear terms in the exponent.

The algebraic algorithms in [22, 23, 24] exploit different algebraic modelings of the RDP as a system of equations that can be solved with computer algebra methods. In case we need to reduce the number of variables before solving such a system, a hybrid approach is used where we reduce an RDP instance with parameters (q, m, n, k, t) to an instance with parameters $(q, m, n-a, k-a, t)$ for some parameter a, at a price of multiplying the cost by $q^{a\,t(n)}$. When studying the asymptotic behaviour of a, we get the following result.

Theorem 6.

- *If $t(n)$ grows faster than logarithmically, i.e. $\lim_{n\to\infty} \frac{t(n)}{\log(n)} = \infty$, all (hybrid) MaxMinors and Support-Minors modelings in [22, 23, 24] give an asymptotic cost of $q^{R\,n\,t(n)}$, up to lower order terms in the exponent.*
- *If $t(n) \sim T\log_q(n)$ for some $T > 0$, the MaxMinors modeling [22, 24],[23, Modeling 1 and 2] can be used in the overdetermined case with an asymptotic cost given by $q^{\omega T\log_q(n)^2}$ up to lower order terms, where ω denotes the linear algebra constant.*

In the second case, i.e., $t(n) \sim T\log_q(n)$, the algebraic algorithms outperform the best combinatorial algorithms. However, as soon as we let the rank weight $t(n)$ grow faster, e.g. when using random codes, the algebraic approach and the combinatorial approach have the same asymptotic cost, up to linear terms in the exponent. These and other lower order terms are yet to be investigated in future research.

Acknowledgements

H. Sauerbier Couvée is supported by the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (Grant agreement No. 801434) and the Bavarian State Ministry of Science and Arts via the project EQAP. A. Ravagnani is supported by the Dutch Research Council (Grants VI.Vidi.203.045, OCENW.KLEIN.539) and by the Royal Academy of Arts and Sciences of the Netherlands. V. Weger is supported by the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie (Grant agreement No. 899987). A. Wachter-Zeh is supported by the Bavarian State Ministry of Science and Arts via the project EQAP.

Sunday, May 26, 2024

Invited Speaker: 09:30 – 10:25

Hints for Codes and Lattices

Alexander May
Ruhr University Bochum, Germany

Implementations may leak partial information of cryptographic secret key, e.g. via side-channel analysis. Such partial information is usually called a hint. It is of crucial importance to understand to which extent hints decrease the security of cryptographic constructions. As a consequence, the security loss of cryptographic keys under various hints has been intensively studied within the last decade.

This talk gives a survey of hints considered in the coding as well as in the lattice world. For codes we review the hint framework introduced by Horlemann, Puchinger, Renner, Schamberger, Wachter-Zeh (CBCrypto 21), and compare it to the lattice-based frameworks of Dachman-Soled, Ducas, Gong, Rossi (Crypto 20), Dachman-Soled, Gong, Hanson, Kippen (Crypto 23) and May, Nowakowski (Asiacrypt 23). We also provide some practical applications of hints, e.g. their use in the cryptanalysis of McEliece-1284 by Esser, May and Zweydinger (Eurocrypt 22).

Acknowledgements

Alexander May is funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) – grant 465120249 and grant 390781972.

Keywords: code-based cryptography, side-channel attacks, lattice-based cryptography

Session 1: 11:00 – 12:25

FuLeakage: Breaking FuLeeca by Learning Attacks

Felicitas Hörmann
German Aerospace Center, Germany
University of St. Gallen, Switzerland

Joint work with: Wessel van Woerden

Keywords: Lee metric, lattices, cryptanalysis, learning attack.

Abstract We show that the *code-based* signature scheme **FuLeeca** is vulnerable to *lattice-based* cryptanalysis. A classical attack using lattice-basis reduction lowers the claimed security levels significantly, whereas learning techniques allow to recover the secret key from the leakage of less than 175 000 signatures in practice. The exploitation of ideal structures and efficient quantum algorithms further yields a full quantum break.

FuLeeca is the first signature scheme based on *Lee-metric* codes and was presented at CBCrypto 2023 [95]. Moreover, **FuLeeca** was submitted [94] to the additional call for digital signatures, that NIST announced in 2022 after three rounds of their first standardization project for post-quantum cryptography had led to little diversity in the used security primitives. Even though **FuLeeca** is *code*-based, we show that it is closely related to known *lattice* schemes such as NTRUSign. This proximity allows us to mount multiple key-recovery attacks that exploit techniques from lattice-based cryptography and fully break the system for all proposed parameter sets.

The *Lee weight* of an element $x \in \mathbb{F}_p$ can be defined as $\mathrm{wt}_L(x) := |x|$, if we identify the finite field $\mathbb{F}_p$ of odd prime order with the set $\left\{-\frac{p-1}{2}, \dots, \frac{p-1}{2}\right\}$. The Lee weight extends additively to a vector $\boldsymbol{x} \in \mathbb{F}_p^n$ and induces the *Lee metric* between two vectors $\boldsymbol{x}, \boldsymbol{y} \in \mathbb{F}_p^n$ as $d_L(\boldsymbol{x}, \boldsymbol{y}) = \mathrm{wt}_L(\boldsymbol{x} - \boldsymbol{y}) = \sum_{i=1}^n \mathrm{wt}_L(x_i - y_i)$. Let us consider a Lee-metric code $\mathcal{C} \subset \mathbb{F}_p^n$ and define the full-rank lattice $\mathcal{L}_1 := \mathcal{C} + p\mathbb{Z}^n \subset \mathbb{R}^n$. Any codeword $\boldsymbol{c} \in \mathcal{C}$ can implicitly be lifted to $\widetilde{\boldsymbol{c}} \in \mathcal{L}_1$ and, in particular, $\mathrm{wt}_L(\boldsymbol{c}) = |\widetilde{\boldsymbol{c}}| := \sum_{i=1}^n |\widetilde{c}_i|$ applies. The ℓ_1-norm is further closely related to the Euclidean ℓ_2-norm, and therefore, shortness and closeness in terms of the Lee metric on $\mathcal{C}$ translate more or less directly into shortness and closeness in terms of the Euclidean ℓ_2-metric on $\mathcal{L}_1$.

FuLeeca can be interpreted as a hash-and-sign scheme which uses the hashed message as an erroneous codeword of a quasi-cyclic code and a decoded low-weight codeword as the signature. The secret key is a generator matrix $\boldsymbol{G}_{\mathrm{sec}} \in \mathbb{F}_p^{k \times n}$ with $n = 2k$ and can be fully described by means of the secret vector $\boldsymbol{g} = (\boldsymbol{a} \mid \boldsymbol{b})$ with $\boldsymbol{a}, \boldsymbol{b} \in \mathbb{F}_p^k$. The i-th row of $\boldsymbol{G}_{\mathrm{sec}}$ is $\left(\mathrm{shift}^{i-1}(\boldsymbol{a}) \mid \mathrm{shift}^{i-1}(\boldsymbol{b})\right)$, where $\mathrm{shift}(\boldsymbol{x}) = (x_k, x_1, \dots, x_{k-1})$ denotes the circular shift of a vector $\boldsymbol{x} = (x_1, \dots, x_k)$. The goal for key-recovery attacks is thus to find the secret vector $\boldsymbol{g}$ or any of its quasi-circular shifts, which we consider equivalent for simplicity.

It is vital to the signing procedure that $\boldsymbol{g}$ has low Lee weight and is thus short in the lattice $\mathcal{L}_1$. We can use lattice-reduction algorithms such as BKZ [97] to find a short vector in $\mathcal{L}_1$ of similar Lee weight and potentially use it as an equivalent secret key to forge signatures. For these parameters BKZ has a heuristic runtime of $2^{0.292\beta+o(n)}$ for $\beta \geq 0.95n$. This attack was already considered in the FuLeeca specification and performs worse than code-based approaches. However, the same idea can be improved by considering a sublattice of lower dimension in which the secret vector $\boldsymbol{g}$ is *unusually* short and thus uniquely and more efficiently recoverable. The key observation is that no wrapping modulo p takes place during FuLeeca's signing step, due to the very large modulus $p = 65\,521$ that is chosen for all proposed parameter sets. As a result, all FuLeeca signatures lie in the sublattice $\mathcal{L}_2 \subset \mathcal{L}_1$ that is generated by the rows of the secret generator matrix $\boldsymbol{G}_{\text{sec}}$. Even though we have a-priori no access to a basis of $\mathcal{L}_2$, a small sample of FuLeeca signatures of size, say, 100 is enough to construct one and proceed with the BKZ attack. Note further that $\mathcal{L}_2$ has rank $n/2$ and that the Euclidean length of $\boldsymbol{g}$ is approximately 15% of the Gaussian heuristic of $\mathcal{L}_2$ and thus unusually short. This allows to heuristically reduce the cost of BKZ to find $\boldsymbol{g}$ down to $2^{\frac{0.292n}{4}+o(n)}$ and hence the security levels of the parameter sets FuLeeca-I, FuLeeca-III, and FuLeeca-V from 160, 224, and 288 bits to 111, 155, and 199 bits, respectively. A full break of FuLeeca with the same amount of signatures is feasible in quantum-polynomial time, when the ideal structure of $\mathcal{L}_2$ is exploited even further.

We further derived a polynomial-time learning attack that recovers the secret key with less than 175 000 available FuLeeca signatures for every parameter set. Learning attacks originate from the break of the lattice-based GGH and NTRUSign schemes [80]. There, they abuse the fact that all signatures lie in the parallelepiped spanned by the short vectors of the secret basis. With enough signatures at hand, one can thus learn the outline of this parallelepiped and hence recover the secret.

In the FuLeeca setting, a similar bias is introduced by the *concentration step* within the signing algorithm. This part tries to alter the signature such that its Lee weight and the number of sign matches with the message hash lie in prescribed intervals. The trial-and-error process successively adds or subtracts the rows of $\boldsymbol{G}_{\text{sec}}$ and checks for improvements. However, the first row is always considered first, then the second row, and so on. This introduces a bias in the signature distribution that is visualized for two dimensions in Figure 1. We average over the outer product of the signature vectors and exploit some properties of the scheme to recover the FuLeeca keys in polynomial time. A sample of 175 000 FuLeeca signatures is enough to break instances of every parameter set.

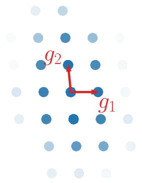


Figure 1: Signature bias in dimension 2.

In summary, FuLeeca is not secure for any parameter set proposed in [94]. The described attacks show once more that code- and lattice-based cryptography are indeed closely related, even though they *seem* different at first sight. Thus, attacks from both sides should be taken into account whenever a new scheme is suggested to ensure reliable security estimates.

Acknowledgements

This abstract is based on [61] and all experiments presented therein were carried out using the PlaFRIM experimental testbed, supported by Inria, CNRS (LABRI and IMB), Université de Bordeaux, Bordeaux INP and Conseil Régional d'Aquitaine. W. van Woerden was supported by the CHARM ANR-NSF grant (ANR-21-CE94-0003).

On Linear Equivalence, Canonical Forms, and Digital Signatures

Tung Chou
Academia Sinica, Taiwan

Joint work with: Edoardo Persichetti, Paolo Santini

Introduction

LESS is a post-quantum signature scheme first introduced in [37]. The scheme is usually considered part of code-based cryptography, although it departs from the traditional methodology of this area. In fact, rather than exploiting the difficulty of decoding, LESS relies on the idea of finding some kind of isomorphism between linear codes. This notion is well-known in coding theory under the name of *code equivalence*, and has been studied for a very long time. Indeed, determining whether two linear codes are equivalent is considered a hard task, in general, and thus constitutes a natural problem to construct cryptographic protocols. Interestingly, such a concept of equivalence can be seen as a *group action*, akin to the ubiquitous one behind the Discrete Logarithm Problem (DLP), although showing more similarities to settings such as the isomorphisms between polynomials, or graphs. It is in In this way that LESS is constructed, following in the steps of well-trodden paths to construct a Sigma protocol based on the Code Equivalence Problem; this is then turned into a signature scheme using the Fiat-Shamir transform.

Subsequent works followed, mainly trying to improve the efficiency of the protocol: for instance, the authors in [26] show that the signature size can be reduced by having a public key comprised of more than two equivalent codes, as well as by selecting challenges according to a fixed-weight distribution. These optimizations are generic, in the sense that they can be applied to any scheme following the same framework, and have appeared in literature in other works, such as for instance [55]. In [89], instead, the authors investigate optimizations that are specific to the chosen setting, i.e. that apply only to code equivalence. In the paper, it is shown that it is possible to further reduce the signature size, by (approximately) a factor of 2, as some pieces of information in the commitments are redundant. The idea of [89] is later used for the specification of LESS [21], as submitted to NIST's call for additional post-quantum signatures [83]. The specification shows that the smallest signature sizes are around 5.0 KiB, 13.4 KiB, and 26.6 KiB for security categories 1, 3 and 5, respectively. These sizes are achieved by using more than 2 generator matrices in each public key, which considerably increases the public key sizes.

Our Contributions

In this paper, we introduce the concept of *canonical forms* for matrices and show how it can be applied to code equivalence. By canonical form, we refer to the representative of a certain equivalence class; in our case, the equivalence relation is derived from the linear equivalence between codes. We show that canonical forms turn out to be a rather useful tool: apart from allowing for a new perspective on the code equivalence group action, they open up the way to new attack avenues, and have a strong impact on cryptographic applications such as digital signatures. We investigate all these aspects, as follows.

First, we formalize a new notion of equivalence between codes, which we call Canonical Form Code Equivalence. We then show that, when canonical forms possess some desirable properties (e.g. they exist with high probability and are efficiently computable), this new notion of equivalence reduces to the standard one, and viceversa. Secondly, we describe a new attack on the code equivalence problem, which relies on having access to efficiently computable canonical forms; the attack has asymptotic cost $2^{\frac{1}{2}\cdot n\cdot h(R)\left(1+o(1)\right)}$, with h being the binary entropy function. Despite being an initial attempt at incorporating canonical forms into the cryptanalysis of the code equivalence problem, this already yields an algorithm which is faster than many other approaches. For instance, for codes with rate $1/2$, the attack runs in time $2^{\frac{n}{2}\left(1+o(1)\right)}$: if q is large enough, this algorithm is currently poised to be the fastest solver for code equivalence. Remarkably, our algorithm is different from previously known algorithms, as for instance our solver does not depend on the type of considered equivalence (permutation or monomial, see e.g. [36] and [100]), nor on the hull dimension (unlike [99] and [25]) or the finite field size q.

Finally, we show how to apply this new notion of equivalence to the LESS scheme. Namely, we apply more powerful notions of canonical forms and, de facto, replace the rudimentary ones used in the original works [37, 26] as well the more sophisticated one proposed in [89].The resulting scheme, which we call CF-LESS, achieves extremely compact signatures, much smaller than its predecessors: for instance, considering the same code and protocol parameters as in the "balanced" parameter sets from the LESS submission [21] (which uses only 2 generator matrices and aims to minimize the public key size), we obtain signatures of only 2.4 KiB, 5.7 KiB, and 9.8 KiB for NIST security categories 1, 3 and 5, respectively. If 4 generator matrices are used, these sizes are further reduced to 1.8 KiB, 4.3 KiB and 7.7 KiB, respectively.

Lattice approach to Lee metric decoding

Karan Khathuria
Quantinuum, United Kingdom

Joint work with: Anna-Lena Horlemann, Marc Newman, Carlos Vela

Keywords: Lee metric, Generic decoding problem, ℓ_1-norm, Lattice theory, Norm embeddings

Abstract The Lee metric has recently gained interest in the construction of code-based cryptosystems, where the security is based on the hard problem of decoding a generic Lee metric code. In this work, we study the lattice-based approach for solving the Lee decoding problem. We first prove a reduction from the Lee decoding problem to fundamental lattice problems in ℓ_1-norm, including the bounded decoding problem and unique shortest vector problem. Using this theoretical framework, we explore the advantage of using ℓ_2-norm based lattice methods to solve the Lee decoding problem.

The field of code-based cryptography has witnessed continuous evolution, with researchers actively exploring novel methods to construct public-key cryptosystems. In recent years, there has been growing interest in utilizing various metrics, such as the rank metric or the Lee metric. For instance, Horlemann and Weger [60] introduced the application of the Lee metric[1] in code-based cryptography, prompting further study on algorithms to solve the generic Lee metric decoding problem, commonly referred to as the Lee decoding problem. Notably, in the recent NIST's call for standardization of post-quantum signature schemes, the first Lee metric-based scheme [96] was proposed, highlighting the increasing relevance of this metric in cryptographic applications.

One intuitive approach to visualize the Lee metric is by linking it to the ℓ_1-norm[2], which, in a certain sense, resides between the Hamming metric and the ℓ_2-norm. Consequently, it is natural to explore established methods tailored for both the Hamming metric and the ℓ_2-norm when addressing decoding problems in the Lee metric. While considerable efforts have been devoted to adapting Hamming metric based Information Set Decoding (ISD) algorithms for the Lee metric (see for e.g. [105, 43, 29]), limited attention has been paid to the adaptation of lattice reduction and enumeration algorithms for this purpose.

In this work, we study the lattice-based approach to solve the Lee decoding problem. To lay the groundwork, we establish the theoretical framework by proving a reduction from the Lee decoding problem to fundamental lattice problems in ℓ_1-norm, including the bounded decoding problem and unique shortest vector problem. Building upon this foundation, we aim to explore the advantage of using ℓ_2-norm based lattice methods to solve the Lee decoding problem.

Reduction from Lee decoding problems to lattice problems

Given a linear code $\mathcal{C}$ in $\mathbb{Z}_q^n$, we associate a lattice to it in the following way. Let G be a $k \times n$ generator matrix of $\mathcal{C}$. Then the associated lattice is given by

$$\mathcal{L}(G) = \left\{ \mathbf{c} \in \mathbb{Z}^n : \mathbf{c} = G^{\intercal}\mathbf{x} \mod q \text{ for some } \mathbf{x} \in \mathbb{Z}^k \right\}. \tag{0.1}$$

[1] The Lee weight of a vector $\mathbf{x} = (x_1, \dots, x_n) \in \mathbb{Z}_q^n$ is given by $\mathrm{wt}_L(\mathbf{x}) = \sum_{i=1}^n \min(x_i, q - x_i)$.

[2] For $1 \leq p < \infty$, the ℓ_p-norm of $\mathbf{x} = (x_1, \dots, x_n) \in \mathbb{R}^n$ is $\|\mathbf{x}\|_p = \left(\sum_{i=1}^n |x_i|^p\right)^{1/p}$.

It is easy to see that the lattice $\mathcal{L}(G)$ is independent from the choice of the generator matrix G.

We assume that the space $\mathbb{R}^n$ is equipped with ℓ_1-norm. The ℓ_1 distance between two vectors $\mathbf{v}, \mathbf{w} \in \mathbb{R}^n$ is denoted by $\text{dist}(\mathbf{v}, \mathbf{w}) := \|\mathbf{v} - \mathbf{w}\|_1$. For a vector $\mathbf{r} \in \mathbb{R}^n$, the distance between $\mathbf{r}$ and $\mathcal{L}$ is given by $\text{dist}(\mathbf{r}, \mathcal{L}) := \inf \{\text{dist}(\mathbf{r}, \mathbf{v}) : \mathbf{v} \in \mathcal{L}\}$. The shortest vector of a lattice $\mathcal{L}$ is the vector in $\mathcal{L}$ having smallest ℓ_1-norm. The length of the shortest vector is denoted by $\lambda_1(\mathcal{L})$.

To state the reduction results, we first recall the definition of the Lee decoding problem and the bounded distance decoding problem.

Problem 7 (Lee decoding problem)**.** *Given a linear code $\mathcal{C}$ over $\mathbb{Z}_q$ of length n, a vector $\mathbf{r} \in \mathbb{Z}_q^n$ and a positive integer $t \in \mathbb{N}$, find a codeword $\mathbf{c} \in \mathcal{C}$ such that $wt_L(\mathbf{r} - \mathbf{c}) \leq t$.*

Problem 8 (α-Bounded distance decoding problem (BDD_α))**.** *Given a lattice $\mathcal{L}$ in $\mathbb{R}^n$ and a vector $\mathbf{r} \in \mathbb{R}^n$ such that $dist(\mathbf{r}, \mathcal{L}) < \alpha\lambda_1(\mathcal{L})$, find a lattice vector $\mathbf{v} \in \mathcal{L}$ such that $dist(\mathbf{v}, \mathbf{r})$ is minimum.*

Using the above connection between a linear code $\mathcal{C}$ and a lattice $\mathcal{L}(G)$, we prove the following two results assuming certain restriction of the parameters:

1. The minimum Lee distance of $\mathcal{C}$ is equal to the norm of the shortest vector in the lattice $\mathcal{L}(G)$.
2. Lee decoding problem reduces to the α-Bounded distance decoding problem.

Effectiveness of ℓ_2-norm lattice methods for solving the Lee decoding problem

As discussed in the previous section, we can reduce the Lee decoding problem to a lattice problem in ℓ_1-norm. However, there aren't many techniques known for solving lattice problems in ℓ_1-norm. This prompts us to explore the potential of using ℓ_2-norm techniques in solving the Lee decoding problem or the corresponding ℓ_1-norm lattice problem.

We first note that there is a direct way of applying ℓ_2-norm methods for ℓ_1-norm, by using the following basic relationship between norms: for any $\mathbf{x} \in \mathbb{R}^n$,

$$\|\mathbf{x}\|_2 \leq \|\mathbf{x}\|_1 \leq \sqrt{n}\, \|\mathbf{x}\|_2 .$$

This relationship does not help much because of the $\sqrt{n}$ factor. Even if we find a short ℓ_2-norm lattice vector, it does not guarantee a small ℓ_1-norm.

The above relationship holds for all the vectors in the space. However, in cryptographic applications, we typically confront average-case scenarios where we are given a random instance of a hard problem. Thus, it becomes relevant to study the norm relationships for a random instance rather than an arbitrary one.

In this ongoing research, we study the effectiveness of using ℓ_2-norm lattice algorithms in solving such average case scenario. Bariffi et al. [28] studied the distribution of a random vector with a given Lee weight. We compare the distribution of a constant Lee weight vector to the Gaussian distribution commonly used in lattice-based cryptography.

While concrete results are yet to be obtained, our preliminary exploration suggests a close alignment between these two distributions within certain parameter domains. Using tools like norm embeddings [92] and statistical distances or Renyi divergence [20], we aim to identify the conditions where ℓ_2-norm lattice methods outperform existing ISD methods in solving the Lee decoding problem.

Poster Session: 13:30 – 14:55, Foyer EO Nord

McEliece Parameter Sets Optimized for Processing in Memory Architectures

Cyrius Nugier
Université de Toulouse, France

Joint work with: Jean-Christophe Deneuville

Keywords: Cyptography, Processing in Memory, Optimization

Abstract Parameter sets for post-quantum primitives have to be designed with their whole lifecycle in mind. Computers will likely evolve to include larger computing spaces, for instance with processing in memory. This architecture change is already known to improve cryptographic performance. In this paper we propose new parameter sets for Classic McEliece to benefit further from this change.

Classic McEliece on Processing In Memory Computers

Computers can be made capable of processing in memory with minimal increase in chip surface. For instance, in [4], the SRAM of a cache is improved by using bit-lines as sensors and adding column peripherals, allowing to compute any binary operation between two lines of the cache at the same time. Even on L1 caches a 4096-bit parallelism can be reached with these operations.

During previous work [84], it has been shown that these architectures improve significantly the performances of classic McEliece [72], notably the Gaussian elimination that was the bottleneck of the key generation. The results in table 1 are given by simulation of a 32-bit RISC-V core (no AVX)[76] with and without a computing cache.

Table 1: Cycle counts of Classic McEliece 348864 with cache computing, and encryption in transposed form.

Architecture	Gaussian	PKgen	KeyGen	Encrypt	Tr. Enc.
RISC-V	793 189 167	840 551 389	1 316 632 015	3 012 219	796 970
RISC-V + PIM	17 098 761	377 406 084	542 756 873	734 918	47 311

This study showed that the improvements provided by Processing in Memory architecture are creating new design spaces. For example, storing the public key in transposed form allows for a significantly faster encryption algorithm. Therefore we aim to adapt the primitives as well as possible to these architectures, notably by rethinking the parameter sets.

Parameter Sets Adaptation

The two main parameters for Classic McEliece n the length of the code and t the error correction capacity do not have the same impact with high parallelism architectures. n has sweetspots when at or below multiples of the parallelism level, while the complexity of Gaussian elimination is in $\mathcal{O}(t^2)$. Following this logic we determined the following parameter sets (n, t): in category I (4096,48) instead of (3488,64), in category III (8192,64) instead of (4608,96), in category V (8192, 112) instead of (6688,128), (6960,119) and (8192,128).

The parameter set least adapted to Processing in Memory seems to be (4608,96), for which no set with equivalent security have been found with $n \leq 4096$. The larger n and smaller t generally cause an increase in PK size and a decrease in ciphertext size which were already Classic McEliece's weakest and strongest characteristics (see table 2).

Table 2: Comparison of characteristics and cycles for arithmetical operations between our proposed sets and the corresponding NIST official ones.

Parameter Sets Proposed / Original	PK bytes	SK bytes	CT bytes	Gaussian elimination	KeyGen time
(4096, 48) / (3488, 64)	-2.94%	+0.68%	-18.75%	-43.76%	-4.74%
(8192, 64) / (4608, 96)	+46.03%	+2.83%	-27.66%	-55.56%	-5.62%
(8192, 112) / (6688, 128)	+17.32%	+1.12%	-10.83%	-23.44%	-4.66%
(8192, 112) / (6960, 119)	+17.12%	+1.01%	-5.31%	-11.42%	-1.18%
(8192, 112) / (8192, 128)	-9.74%	-0.23%	-10.83%	-23.44%	-6.89%

The main remaining bottlenecks of the key generation for architectures with processing in memory are: the *controlbits* subroutine (82% of the total KeyGen time), the secret key generation (28% of the rest) even when already improved with lower t, and the filling of the matrix (41% of the public key generation).

Acknowledgements

The authors would like to thank the Institut Cybersécurité Occitanie for supporting this research.

Classic McEliece and RLCE algorithms enabled for OpenSSL key exchange

Jonathan Wagner
UNC Charlotte, USA

Joint work with: Yongge Wang

Keywords: Classic McEliece, RLCE, Transport Layer Security, OpenSSL

Abstract An OpenSSL 1.1.1 implementation that utilizes both Classic McEliece and RLCE algorithms for TLS 1.3 key exchange is introduced in this paper. In order for the TLS connections to be successful for both groups of algorithms and other s_server/s_client tests, major modifications of the post-quantum OpenSSL fork from GitHub (see [85]) are needed and are discussed in this paper; this results in this paper's introduced OpenSSL implementation, and these include changes to code involving the use of both the tls_construct_client_hello and tls_process_client_hello functions along with revisions to functions within statem_srvr.c, statem_clnt.c, and extensions_srvr.c. This paper's OpenSSL implementation is also proven to be a real-world application, as successful connections are made to google.com and bing.com, and also TLS connections are established with another normal OpenSSL s_server and s_client. TLS connections using the post-quantum algorithm kyber768 and the classical algorithm X25519 for key exchange were also conducted in an experiment. Ultimately, this paper proves that large public key algorithms of Classic McEliece and RLCE are possible for TLS 1.3 key exchanges of OpenSSL.

Theory of Operation

Modifications to code involving the tls_construct_client_hello function

The construction of a ClientHello message, including its key share extension, must be modified in order to hold large public key sizes of the RLCE and Classic McEliece algorithms; that is the size limits must be extended well beyond the approximate 65K Byte limit. Key exchanges that choose RLCE or Classic McEliece will work when such modifications are considered, but TLS connections to a regular OpenSSL server or even google.com, will fail. Therefore, code must be added where if the requested key exchange NID from the client requesting a TLS connection is either a Classic McEliece or RLCE algorithm, then the construction of the ClientHello message is then changed to include the large public key of the chosen encryption algorithm; otherwise, the ClientHello message construction will function normally for all other NIDs of key exchange algorithms requested (X25519, etc.).

Modifications to code involving the tls_process_client_hello and tls_process_server_hello functions

Not only does the construction of ClientHello message needs to be modified, but the processing of it as well. Otherwise, the server will throw a "length mismatch" or even a "bad extension" error when trying to process a ClientHello message during a TLS 1.3 handshake. Therefore, the code for processing such a large ClientHello message (which includes either a RLCE or a Classic McEliece public key in its key share extension) needs to be modified to handle such circumstances. When the tls_process_client_hello function invokes the tls_collect_extensions function, in regards to collecting all extensions from a ClientHello message, the functioning of this tls_collect_extensions function is then modified to help process large key shares. Specifically, all extensions are processed normally, but when an Extension Type 51 is processed, then a large key share extension including a Classic McEliece or RLCE public key algorithm, is taken into consideration when collecting all extensions from the ClientHello message.

The changed functionality of the tls_collect_extensions function will work for processing of ClientHello messages. However, this fails for clients for the processing of a ServerHello message, when it has a key exchange length containing a rlcel1 algorithm's ciphertext of 988 Bytes (when this type of RLCE algorithm is requested from the client at the start of a TLS handshake). Therefore, the original functionality of tls_collect_extensions must be preserved for the tls_process_server_hello function for the client where it processes the ServerHello message received. Keeping the tls_collect_extensions functionality intact for the processing of ServerHello messages will allow TLS connections to initiate for both Classic McEliece and RLCE algorithms requested as key exchange algorithms.

The new way that tls_process_client_hello functions will result in a "length mismatch" error, when a normal OpenSSL client requests a TLS connection from the server. The functioning of the tls_process_client_hello function is then modified for the server to take into consideration if a client requests a Classic McEliece or RLCE algorithm, or another algorithm otherwise for key exchange. Making this decision based on the algorithm's NID will not work in this circumstance. Instead, the "message_size" of the ClientHello message, and comparing it to the rlcel1 algorithm's ClientHello message's payload length (188317 Bytes), must be considered in order for the server to act appropriately [48]. This works as follows: if the server receives a ClientHello "message_size" shorter than 188317 Bytes, then it will process the ClientHello message normally. Otherwise, the server will process the ClientHello message and will take into consideration that this ClientHello message contains a large key share; this large key share contains either a Classic McEliece or RLCE algorithm in these types of situations.

An algorithmic optimization for HQC on the ARM Cortex-M4 MCU

Ridwane Aissaoui
University of Toulouse, France

Joint work with: Jean-Christophe Deneuville, Christophe Guerber, Alain Pirovano

Keywords: HQC: Hamming Quasi-Cyclic, PQC: Post-Quantum Cryptography, KEM: Key Encapsulation Mechanism, Optimization

Abstract Constrained systems with limited computational and memory resources, such as IoT devices, small plUAV, or medical devices require the implementation of cryptographic primitives to ensure information security. An encryption scheme can provide confidentiality and authentication. plKEM are vital in this process. Several propositions for quantum-resistant plKEM have already been chosen by the NIST for standardization. Among the remaining schemes in the fourth round, HQC can be implemented with algorithmic optimization using properties in polynomial operations to reduce computational load and memory usage. Performance tests of this implementation on the ARM m4 MCU show considerable improvements in time complexity and memory usage.

HQC Cryptosystem

HQC is one of the code-based candidates still under consideration in the fourth round of the NIST standardization process (alongside BIKE and Classic McEliece). This algorithm ensures a secure key exchange between parties using a public-key architecture. The pqm4 project [64] has provided optimized versions of BIKE, but not of HQC. This work implements a known algorithmic optimization of HQC on an ARM Cortex-M4 processor. The test bed consists of the the STM32F4DISCOVERY development board [2], providing 192kB of RAM and 1MB of flash memory. The environment is completed by the ChibiOS RTOS [1]. Using an RTOS impedes pure performance, but it allows us to closely emulate real-world applications.

We evaluated the performance of the KEM algorithms that are still in the standardization process on this test bed : HQC, BIKE and CRYSTALS-Kyber. Classic McEliece requires more memory than we have available, so we cannot compare its performance on this test bed. With readily available implementations, the performance of HQC seemed much worse than its competitors. BIKE and CRYSTALS-Kyber were used in the m4f optimizations by pqm4 [64] wehereas HQC was used in the PQClean version [65].

Algorithmic optimization of HQC

The most expensive parts of the HQC algorithm are multiplications of vectors with n components in $\mathbb{F}_2$ which are computed with polynomial multiplications. In HQC polynomial multiplications one of the polynomials is sparse. This allows us to store only the indices of non-zero values instead of the entire polynomial. With this representation, the polynomial multiplication can be simplified by left shifting the arbitrary polynomial with each index of the sparse polynomial and then xoring each of these rotations. This strategy on the polynomial multiplication has a complexity of $\mathcal{O}(n * w)$ (or $\mathcal{O}(n\sqrt{n})$ if we consider that the Hamming weight of the sparse vector grows with the square root of n). A similar process is used in polynomial addition, further improving performance. Also, the implementation of the Karatsuba algorithm proposed in the reference implementation uses a considerable space in the RAM, more than $16n$ bits, whereas our implementation only uses $2n$ bits for its variables during polynomial multiplication.

Algorithm	Max Cycles Keygen	Max Cycles Encapsulation	Max Cycles Decapsulation	RAM usage (bytes)	Flash memory usage (bytes)
HQC-128 (PQClean)	48,030,414 (285.9 ms)	96,874,954 (576.7 ms)	145,737,544 (867.5 ms)	85,000	33,692
HQC-128 (optimized)	1,837,507 (11 ms)	4,878,515 (29.1 ms)	7,502,580 (44.7 ms)	50,000	29,484
BIKE-1	36,895,891 (219.6 ms)	4,309,361 (25.7 ms)	73,127,121 (435.8 ms)	90,000	121,668
Kyber-2	747,259 (4.5 ms)	898,939 (5.4 ms)	798,862 (4.8 ms)	10,000	18,332
ECDH x25519	3,587,785 (21.3 ms)	3,564,808 (21.3 ms)	3,564,808 (21.3 ms)	2,048	19,932
RSA 2048	∞	11,216,240 (66.8ms)	118,744,894 (706.8 ms)	2,048	15,784

Table 3: Global performance comparison with current standards and PQC competitors

The new sparse vector generation, polynomial multiplication and polynomial addition algorithms were designed to remain constant time. The numbers of elemental operations only depends on the Hamming weight of the sparse operator, which is constant and publicly known.

We tested our optimized version of HQC against BIKE, CRYSTALS-Kyber, RSA and ECDH. The results of these performance tests are summed up in Table 3. We see a substantial improvement in computational complexity and memory consumption, which make HQC viable for constrained systems.

Increasing Index Sizes in Information Set Decoding Algorithms

Zachary Welch
Carleton University, Canada

Joint work with: joint work

Keywords: Information Set Decoding, Stern's Algorithm, Row Reduction

Abstract This paper introduce a modification to Stern's algorithm that reduces average decryption time by approximately 14.5% for a code of length 1024, dimension 524 capable of correcting 50 errors.

Information Set Decoding

There exist many post-quantum cryptosystems, with one important category being that of Hamming weight code-based cryptography. Hamming weight code-based cryptosystems are three of the four Round 4 cryptosystems in the NIST PQC standardization competition [82], those being BIKE [14], HQC [74] and Classic McEliece [9]. One of the most effective types of attacks against Hamming weight code-based cryptosystems are information set decoding attacks, notably the variations of Stern's algorithm [102] and including the most recent attacks [52, 79]. For an attack parameter l, these attacks involve row reducing the parity check matrix H into the following form, up to permutation of rows and columns:

$$H = \begin{bmatrix} \begin{matrix} I_{n-k-l} \\ 0_{l\times(n-k-l)} \end{matrix} & H_1 \end{bmatrix} \tag{0.2}$$

We call these first $n-k-l$ columns the index of H. In this paper, we show that with a different row reduction it is possible to replace I_{n-k-l} with a wider matrix that still allows the attack to be performed. We detail one such algorithm that allows 3 columns to be added to the index, which gives an average 16.2% reduction in decryption times depending on cryptosystem and attack parameters.

Increasing the Index Size

In Stern's Algorithm, for an attack parameter p, we check if the Hamming weight of sums of the syndrome vector and $2p$ columns from H_1 has Hamming weight $t-2p$. The property this uses is that the sum of $t-2p$ columns outside of H_1 has Hamming weight $t-2p$. If instead we replaced I_{n-k-l} with a matrix whose columns all had maximum Hamming weight u, then the sum of $t-2p$ of its columns would have weight at most $u(t-2p)$. Algorithm 1 details one such way of minimizing the Hamming weight of columns in a matrix.

Algorithm 1: Low Column Weight Row Reduction

Input: A binary $(n-k) \times n$ matrix H in RREF with its first $n-k$ columns linearly independent.

Output: $H' \in \mathbb{F}_2^{(n-k)\times n}$ with the weight of the first $n-k+3$ columns of H' at most 2 for all but 6 columns, those 6 columns having weight of at most 3.

```
Set $r_i = -1$ for all $i \in \{0,1\}^3$
for $d$ from 0 to $n-k-1$:
    Let $i$ =concat($H_{d,n-k}, H_{d,n-k+1}, H_{d,n-k+2}$)
    if $i = 000$:
        $H'[d] := H[d]$
    if $i \neq 000$:
        if $r_i \neq -1$ then set $H'[r_i] := H[r_i] + H[d]$
        $r_i := d$
Set $H'[r_i] := H[r_i]$ for all $i \in \{0,1\}^3$
$H'[r_{111}] := H[r_{111}] + H[r_{110}]$
$H'[r_{101}] := H[r_{101}] + H[r_{001}]$
$H'[r_{011}] := H[r_{011}] + H[r_{010}]$
return $H'$
```

If at least l values of d had $i = 000$ in Algorithm 1, then H' is in the form described in Equation 0.2 up to row and column permutation. The specific permutations move l rows that corresponded with $i = 000$ to the bottom l rows of H' and the l non-zero columns of those rows into H_1.

Results

There are three factors with using Algorithm 1 that slow down each iteration; those factors are the potential of false positives, the runtime of the algorithm itself and the reduced effectiveness of early aborts. We use the CPU cycle counts and parameter set given in [35] to obtain theoretical values for each of these.

False positives occur when a vector of low weight is identified that does not correspond to a solution. The probability of a vector being a false positive is approximately $2^{-143.8}$; this is low enough to have no practical effect on the runtime of the algorithm. Comparing the number of row operations in Algorithm 1 to the Gaussian elimination step of Stern's Algorithm, we find that Algorithm 1 should be between 1.55% and 1.8% of an iterations runtime. For the version of ISD being used in [35] Algorithm 1 needs to be run an average of 2.36 times per iteration. The column sum threshold being increased from $t-2p$ to $2(t-2p)+6$ slows down early aborts when checking the Hamming weight of vectors. While we do not know the exact effect of increasing the threshold on the runtime of an iteration, we do know that incorporating early aborts saves 8.8% of an iterations runtime. This 8.8% is an upper bound of the actual slow down percentage which we use in place of a more accurate percentage for the slowdown.

The effects of increasing the index size by 3 is a reduction in the average number of iterations by 24.2%. With each iteration taking approximately 12.8% longer to perform, this gives 14.5% less time on average to identify a solution.

These changes add 3 columns to the index, the ideal number of columns to add to index as well as their corresponding speedups is an area of future study.

Session 2: 15:30 – 16:55

Group Factorisation for Smaller Signatures from Cryptographic Group Actions

Giuseppe D'Alconzo
Politecnico di Torino, Italy

Joint work with: Alessio Meneghetti, Edoardo Signorini

Keywords: Digital signatures, Post-quantum, Code equivalence.

Abstract Cryptographic group actions have gained significant attention in recent years for their application on post-quantum sigma protocols and digital signatures. In NIST's recent additional call for post-quantum signatures, three relevant proposals are based on group actions: LESS, MEDS, and ALTEQ. This work explores signature optimisations leveraging a group's factorisation. We show that if the group admits a factorisation as a semidirect product of subgroups, the group action can be restricted on a quotient space under the equivalence relation induced by the factorisation. If the relation is efficiently decidable, we show that it is possible to construct an equivalent sigma protocol for a relationship that depends only on one of the subgroups. Moreover, if a special class of representative of the quotient space is efficiently computable via a canonical form, the restricted action is effective and does not incur in security loss. Finally, we apply these techniques to LESS and MEDS, showing how they will affect the length of signatures and public keys.

Group Action and Digital Signatures

The topic of cryptographic group action has raised a lot of interest in recent years. They represent a generalisation of the Discrete Logarithm Problem, and the underlying problem, called *Group Action Inversion Problem*, can be stated as follows: given a group action $(G, X, \star)$ and two elements x, y in X, find, if any, an element g of G such that $y = g \star x$. The most impactful application is the one related to sigma protocols and digital signatures. For instance, three candidates to the NIST's call for the post-quantum standardisation are based on group actions: LESS [27], MEDS [45] and ALTEQ [104]. These three signatures share the following general structure: the public key contains elements in X, the secret key in G, and the signature mainly consists of a sequence of bits and elements in G. Shortening the dimension of elements in G is crucial to achieve smaller signatures, while shortening the dimension of elements in X allows for smaller public keys.

Equivalence Relations from Groups Factorisations

Given a group action $(G, X, \star)$, suppose that we can write G as $G_1 \times G_2$. Hence, we assume that we can efficiently represent every element of G as a pair (g_1, g_2). It is natural to define the following equivalence relation on $X \times X$

$$x \sim y \iff \exists g_1 \in G_1 \text{ such that } y = (g_1, e) \star x.$$

Given the quotient space $X_\sim$ with respect to the equivalence $\sim$, we can define a new group action $(G_2, X_\sim, \star_\sim)$ as follows

$$g_2 \star_\sim [x]_\sim \mapsto [(e, g_2) \star x]_\sim.$$

To obtain an *effective* group action, we want that two elements in $X_\sim$ must have a unique representation. To prove that two orbits of $X_\sim$ are the same, we use a special class of representatives computable via a canonical form. The map $\mathsf{CF}_\sim : X \to X \cup \{\perp\}$ is a *canonical form with failures* if, for any $x, y \in X$ such that $x \sim y$, then $\mathsf{CF}_\sim(x) = \mathsf{CF}_\sim(y)$ and if $\mathsf{CF}_\sim(x) \neq \perp$ then $\mathsf{CF}_\sim(x) \sim x$. If $\mathsf{CF}_\sim(x) = \perp$ we say that $\mathsf{CF}_\sim$ fails on the element x. Notice that when $\mathsf{CF}_\sim(x) = \mathsf{CF}_\sim(y) \neq \perp$, then $x \sim y$ and if $\mathsf{CF}_\sim(x) = \perp$ then $\mathsf{CF}_\sim(y) = \perp$ for every $y \sim x$.

If there exists an efficiently computable canonical form CF with low failure probability, i.e. with overwhelming probability $x \sim y$ if and only if $\mathsf{CF}(x) = \mathsf{CF}(y)$, then the above action is efficiently computable identifying the orbits of $X_\sim$ with their representatives

$$g_2 \star_\sim x \mapsto \mathsf{CF}((e, g_2) \star x).$$

This leads to an effective group action.

One might wonder about the hardness of the Group Action Inverse Problem of this new action. We can prove the following.

Proposition 9. *If there exists a polynomial-time computable canonical form* CF *for the equivalence* $\sim$ *such that* CF *also returns the element* g_1 *such that* $\mathsf{CF}(x) = (g_1, e) \star x$*, the Group Action Inverse problems for* $(G, X, \star)$ *and* $(G_2, X_\sim, \star_\sim)$ *are polynomially equivalent.*

Observe that most canonical forms used in practice satisfy the condition of Proposition 9. Moreover, the above result also works for the semidirect product $G = G_1 \rtimes G_2$ with respect to the relation induced by G_1.

Proposition 9 implies that, if one is able to factorise G and there exists a polynomial-time computable canonical form with respect to the relation for a factor G_1, then the induced action $(G_2, X_\sim, \star_\sim)$, where G_2 is the remaining factor, can be used without introducing new computational assumptions. This means that, instead of using elements from G and X, one can use elements from G_2 and $X_\sim$, potentially reducing the sizes of the elements involved. This is implicitly used in the Linear Code Equivalence Problem when the systematic form is employed.

Applications to Code Equivalence Problems

Here, we show an application of the above technique to reduce the sizes of MEDS [45], a digital signature scheme based on the equivalence of matrix codes. Recall that the group action underlying MEDS uses $G = \mathrm{GL}(n, q) \times \mathrm{GL}(m, q) \times \mathrm{GL}(k, q)$ and X as the set of matrix codes of length $n \times m$ and dimension k. In [45], using the systematic form for the action of the last factor $\mathrm{GL}(k, q)$, they implicitly use only the action of the remaining part of the group $\mathrm{GL}(n, q) \times \mathrm{GL}(m, q)$. Here we go further, quotienting on the factors $\mathrm{GL}(m, q) \times \mathrm{GL}(k, q)$. To obtain this new protocol, we need to exhibit a canonical form for the relation induced by the group $\mathrm{GL}(m, q) \times \mathrm{GL}(k, q)$, and it requires a slightly involved process. Since this canonical form is polynomial-time but inefficient, in the signature, in addition to the invertible matrix in $\mathrm{GL}(n, q)$, we also transmit a hint to speed up the computation. Observe that this hint does not give any additional information to an attacker since it can be computed in polynomial time from the other data via the polynomial-time but inefficient canonical form.
Concerning the version of MEDS that uses the action of $\mathrm{GL}(n, q) \times \mathrm{GL}(m, q)$ from [45], our proposal allows to reduce the size of the signature from 38.6% to 47.7% for the last version of the parameter sets given in [44], as reported in Table 4.

Parameter set	Sec. Level	MEDS [44]	This work	Gain
MEDS-9923	I	9896	6074	38.6%
MEDS-13220	I	12976	7516	42.1%
MEDS-41711	III	41080	23062	43.9%
MEDS-69497	III	54736	29788	45.6%
MEDS-134180	V	132424	70284	46.9%
MEDS-167717	V	165332	86462	47.7%

Table 4: Signature sizes in B.

A similar result can be translated in the setting of LESS. Here the group action uses $G = (\mathrm{GL}_k(\mathbb{F}_q) \times (\mathbb{F}_q^*)^n) \rtimes \mathcal{S}_n$. We set $G_1 = \mathrm{GL}_k(\mathbb{F}_q) \times (\mathbb{F}_q^*)^n$ and $G_2 = \mathcal{S}_n$. Moreover, the authors of LESS recently presented in [46] a new notion of equivalence for codes and proved that it reduces to linear equivalence. This leads to an even more significant reduction in the size of responses. This last variant can partially be framed within our framework, even if they use subsets instead of subgroups to factor G. Unlike in our framework, this variant must be explicitly shown to be equivalent to the original assumption. Further research should consider the possibility of extending the results of this work to a generic factorisation involving a subset of G. See Table 5 for a comparison.

Parameter set	Sec. Level	LEP	IS-LEP [90]	CF-LEP [46]	This work
LESS-1b	I	15726	8646	2496	9096
LESS-3b	III	30408	17208	5658	18858
LESS-5b	V	53896	30616	10056	34696

Table 5: Signature sizes in B.

Complexity of Solving Syndrome Decoding Problems as a System of Multivariate Equations

Alex Pellegrini
Eindhoven University of Technology, The Netherlands

Joint work with: Alessio Caminata, Ryann Cartor, Alessio Meneghetti

Keywords: Code-based cryptosystems, Cryptanalysis, Polynomial systems

Abstract We introduce a framework for the study of the Syndrome Decoding Problem by multivariate polynomials, and thus for the analysis of code-based cryptosystems.

With the imminent arrival of quantum computing, the need for accurate cryptanalysis of post-quantum cryptography, particularly for code-based cryptosystems (like McEliece), is crucial. While it is often assumed that Information Set Decoding is the most efficient attack against a generic code-based cryptosystem, it is important to also consider attacks from various other post-quantum cryptography realms, such as multivariate or lattice-based schemes. For instance, MinRank attacks, commonly used in multivariate cryptography, have successfully compromised rank-metric code-based schemes like RQC and Rollo.

In this paper, we improve upon the description of [75] and introduce a conversion from an instance of the syndrome decoding problem into a system of multivariate polynomial equations. We also study the complexity of solving this system with a direct algebraic attack via Gröbner bases algorithms.

Our approach can be described as follows. Let $n \geq 2$, $H = (h_{i,j})$ be a $(n-k) \times n$ parity-check matrix of an $[n, k, d]$-code over $\mathbb{F}_2$, $s \in \mathbb{F}_2^{n-k}$, $0 \leq t \leq \lfloor \frac{d-1}{2} \rfloor$, so that (H, s, t) is an instance of the syndrome decoding problem, i.e. the problem of finding a solution $\bar{\mathbf{x}} \in \mathbb{F}_2^n$ of weight t of the linear system $H\mathbf{x} = s$. Let $\mathbf{x} = (x_1, \dots, x_n)$ and $\mathbf{y} = (y_1, \dots, y_{n\ell})$ be two sets of variables, where $\ell = \lfloor \log_2(t) \rfloor + 1$.

The reduction of an instance of a syndrome decoding problem consists of four steps, each of which produces a set of polynomials in $\mathbb{F}_2[\mathbf{x}, \mathbf{y}]$: *parity check encoding* (pce), *hamming-weight computation encoding* (hwce), *weight constraint encoding* (wce), and *finite-field equations* (ffe). The system $\mathcal{S}$ consists of the union of the following four sets:

$$
\begin{aligned}
\mathrm{pce} &= \left\{ \sum_{j=1}^{n} h_{i,j} x_j \mid i \in [n-k] \right\}, \\
\mathrm{hwce} &= \left\{ x_1 + y_1, y_2, \dots, y_\ell \right\} \cup \left\{ x_i + y_{(i-1)\ell+1} + y_{(i-2)\ell+1} \mid i \in [2, n] \right\} \\
&\quad \cup \left\{ y_{(i-1)\ell+j-1} y_{(i-2)\ell+j-1} + y_{(i-1)\ell+j} + y_{(i-2)\ell+j-1} + y_{(i-2)\ell+j} \right. \\
&\qquad \left. \mid i \in [2, n]\ j \in [2, \ell] \right\}, \\
\mathrm{wce} &= \left\{ y_{(n-1)\ell+j} + v_j \mid j \in [\ell] \right\}, \text{ for a given } v \in \mathbb{F}_2^\ell, \\
\mathrm{ffe} &= \left\{ x_i^2 - x_i \mid i \in [n] \right\} \cup \left\{ y_j^2 - y_j \mid j \in [n\ell] \right\}.
\end{aligned}
$$

Any solution of $\mathcal{S}$ is a vector $(\bar{\mathbf{x}}, \bar{\mathbf{y}})$, where $\bar{\mathbf{x}}$ solves the original instance of the syndrome decoding problem. The modeling proposed in [75] consists of a set of high-degree polynomials that is then modified to obtain a quadratic system with $\mathcal{O}(n \log_2(n)^2)$ equations and variables. As shown in Table 6, our modeling improves over [75] by a factor of around $\log_2(n) \cdot \log_t(n)$.

	# Equations	# Variables
[75]	$\mathcal{O}(n \cdot \log_2(n)^2)$	$\mathcal{O}(n \cdot \log_2(n)^2)$
This work	$2n(\ell+1) - k + \ell$	$n(\ell+1)$

Table 6: Comparison with the asymptotic size of the polynomial system in [75, Theorem 13], where n is the length of the code, k its dimension and $\ell = \lfloor \log_2(t) \rfloor + 1$.

Goppa code		[75]			This work	
(n, k, t, ℓ)		# EQS	# VARS		# EQS	# VARS
(8,2,2,2)		~ 128	~ 128		48	24
(16,8,2,2)		~ 400	~ 400		156	48
(64,16,8,4)		~ 3136	~ 3136		628	256

Table 7: Comparison between the size of the polynomial systems associated to decoding Goppa codes obtained via the reduction in [75] and by this work.

We study the complexity of solving the system $\mathcal{S}$ via Gröbner bases methods. The fastest known Gröbner bases algorithms are the linear-algebra-based algorithms (such as $F4$) which reduce the problem of solving the system to several instances of Gaussian elimination. Their complexity is dominated by that of Gaussian elimination on the largest Macaulay matrix encountered during this process. The size of this matrix depends on the number of variables and on the *solving degree* $\mathrm{sd}(\mathcal{S})$ of the system [41]. Since the solving degree is usually hard to estimate without actually solving the system, we consider the *degree of regularity* $d_{\mathrm{reg}}(\mathcal{S})$ which (under suitable assumptions) provides an upper bound on the solving degree [98].

We write $\mathcal{S} = \mathcal{L} \cup \mathcal{F}$, where $\mathcal{L}$ and $\mathcal{F}$ denote respectively the sets of linear equations associated to pce, and the other fixed (linear and quadratic) polynomials associated to hwce, wce and ffe, respectively. We are able to compute the degree of regularity of the fixed part of the system.

Theorem 10. *The degree of regularity of $\mathcal{F}$ is*

$$d_{\mathrm{reg}}(\mathcal{F}) = n + \left\lceil \frac{n-1}{2} \right\rceil + (\ell - 2) \left\lceil \frac{n-2}{2} \right\rceil - 1.$$

We point out that $d_{\mathrm{reg}}(\mathcal{S}) \leq d_{\mathrm{reg}}(\mathcal{F})$. Moreover, based on several computer experiments we performed with MAGMA, we conjecture the following.

Conjecture 11. $d_{\mathrm{reg}}(\mathcal{F}) - \ell \leq d_{\mathrm{reg}}(\mathcal{S})$.

By what we said above, our theorem directly gives an upper bound for the complexity of solving the polynomial system $\mathcal{S}$ via the usual linear-algebra-based Gröbner bases algorithms. On the other hand, the story seems to be much more involved than this. In fact, we performed several experiments taking as input a random Goppa code, and we obtained a solving degree which is much smaller than the expected degree of regularity. This results in a much lower complexity estimate. To give a flavour of our results, we report a small table with a selection of our experiments here.

Goppa code		This work						
(n, k, t, ℓ)		# EQS	# VARS	$d_{\text{reg}}(\mathcal{F})$	$d_{\text{reg}}(\mathcal{F}) - \ell$	$\text{sd}(\mathcal{S})$	Comp $(\text{sd}(\mathcal{S}))$	Prange
(8,2,2,2)		48	24	11	9	2	$2^{16.69}$	2^{9}
(16,8,2,2)		156	48	23	21	3	$2^{28.69}$	$2^{12.9}$
(64,16,8,4)		628	256	157	154	4	$2^{54.94}$	2^{19}

Table 8: The values $d_{\text{reg}}(\mathcal{F})$ and $d_{\text{reg}}(\mathcal{F}) - \ell$ are given by Theorem 10. They give an upper bound and a conjectural lower bound on the degree of regularity of $\mathcal{S}$ respectively. The value $\text{sd}(\mathcal{S})$ is the solving degree of $\mathcal{S}$, computed as the highest step degree achieved when directly computing the Gröbner basis of the system in MAGMA. The value Prange is the computational cost of decoding with Prange's Information Set Decoding (ISD) algorithm.

Future Directions

Our results showed that in this set-up the degree of regularity might not be a good estimate for the solving degree of the system. Therefore, we will investigate further the solving degree of $\mathcal{S}$ by using different approaches. First, we will perform more experiments by considering different classes of codes rather than Goppa codes in order to see how the input code affects the complexity of the system.

Bibliography

[1] ChibiOS RTOS.

[2] Discovery kit with STM32f407vg MCU - STMicroelectronics.

[3] Gora Adj, Luis Rivera-Zamarripa, Javier Verbel, Emanuele Bellini, Stefano Barbero, Andre Esser, Carlo Sanna, and Floyd Zweydinger. MiRitH. In *First Round Submission to the additional NIST Postquantum Cryptography Call*, 2023.

[4] Shaizeen Aga, Supreet Jeloka, Arun Subramaniyan, Satish Narayanasamy, David Blaauw, and Reetuparna Das. Compute caches. In *2017 IEEE International Symposium on High Performance Computer Architecture (HPCA)*, pages 481–492. IEEE, 2017.

[5] Carlos Aguilar Melchor, Nicolas Aragon, Magali Bardet, Slim Bettaieb, Loic Bidoux, Olivier Blazy, Jean-Christophe Deneuville, Philippe Gaborit, Adrien Hauteville, Ayoub Otmani, Olivier Ruatta, Jean-Pierre Tillich, and Gilles Zémor. ROLLO- Rank-Ouroboros, LAKE & LOCKER. *NIST PQC Call for Proposals*, 2020. Round 2 Submission.

[6] Carlos Aguilar Melchor, Nicolas Aragon, Slim Bettaieb, Loic Bidoux, Olivier Blazy, Maxime Bros, Alain Couvreur, Jean-Christophe Deneuville, Philippe Gaborit, Adrien Hauteville, and Gilles Zémor. Rank Quasi-Cyclic (RQC). *NIST PQC Call for Proposals*, 2020. Round 2 Submission.

[7] Carlos Aguilar Melchor, Thibauld Feneuil, Nicolas Gama, Shay Gueron, James Howe, David Joseph, Antoine Joux, Edoardo Persichetti, Tovohery H. Randrianarisoa, Matthieu Rivain, and Dongze Yue. SDitH. Technical report, National Institute of Standards and Technology, 2023. available at `https://csrc.nist.gov/Projects/pqc-dig-sig/round-1-additional-signatures`.

[8] Adel Alahmadi, Selda Çalkavur, Patrick Solé, Abdul Nadim Khan, Mohd Arif Raza, and Vaneet Aggarwal. A new code based signature scheme for blockchain technology. *Mathematics*, 11(5):1177, 2023.

[9] Martin R Albrecht, Daniel J Bernstein, Tung Chou, Carlos Cid, Jan Gilcher, Tanja Lange, Varun Maram, Ingo Von Maurich, Rafael Misoczki, Ruben Niederhagen, et al. Classic McEliece: conservative code-based cryptography. 2022.

[10] Dorian Amiet, Lukas Leuenberger, Andreas Curiger, and Paul Zbinden. FPGA-based SPHINCS+ implementations: Mind the glitch. In *2020 23rd Euromicro Conference on Digital System Design (DSD)*, pages 229–237. IEEE, 2020.

[11] N. Aragon, P. Gaborit, and G. Zémor. HQC-RMRS, an instantiation of the HQC encryption framework with a more efficient auxiliary error-correcting code. *arXiv preprint arXiv:2005.10741*, 2020.

[12] Nicolas Aragon, Magali Bardet, Loïc Bidoux, Jesús-Javier Chi-Domínguez, Victor Dyseryn, Thibauld Feneuil, Philippe Gaborit, Antoine Joux, Matthieu Rivain, Jean-Pierre Tillich, and Adrien Vinçotte. RYDE. In *First Round Submission to the additional NIST Postquantum Cryptography Call*, 2023.

[13] Nicolas Aragon, Magali Bardet, Loïc Bidoux, Jesús-Javier Chi-Domínguez, Victor Dyseryn, Thibauld Feneuil, Philippe Gaborit, Romaric Neveu, Matthieu Rivain, and Jean-Pierre Tillich. MIRA. In *First Round Submission to the additional NIST Postquantum Cryptography Call*, 2023.

[14] Nicolas Aragon, Paulo Barreto, Slim Bettaieb, Loic Bidoux, Olivier Blazy, Jean-Christophe Deneuville, Philippe Gaborit, Santosh Ghosh, Shay Gueron, Tim Güneysu, et al. BIKE: bit flipping key encapsulation. 2022.

[15] Nicolas Aragon, Paulo Barreto, Slim Bettaieb, Loïc Bidoux, Olivier Blazy, Jean-Christophe Deneuville, Philippe Gaborit, Santosh Ghosh, Shay Gueron, Tim Güneysu, Carlos A Melchor, Rafael Misoczki, Edoardo Persichetti, Jan Richter-Brockmann, N Sendrier, Jean-Pierre Tillich, Valentin Vasseur, and Gilles Zemor. BIKE: bit flipping key encapsulation. 2022.

[16] Nicolas Aragon, Paulo Barreto, Slim Bettaieb, Loïc Bidoux, Olivier Blazy, Jean-Christophe Deneuville, Philippe Gaborit, Shay Gueron, Tim Guneysu, Carlos Aguilar Melchor, et al. BIKE: Bit Flipping Key Encapsulation. 2020.

[17] Nicolas Aragon, Philippe Gaborit, Adrien Hauteville, and Jean-Pierre Tillich. A new algorithm for solving the rank syndrome decoding problem. In *2018 IEEE International Symposium on Information Theory (ISIT)*, pages 2421–2425, 2018.

[18] Sarah Arpin, Tyler Raven Billingsley, Daniel Rayor Hast, Jun Bo Lau, Ray Perlner, and Angela Robinson. A study of error floor behavior in QC-MDPC codes. In *International Conference on Post-Quantum Cryptography*, pages 89–103. Springer, 2022.

[19] Daniel Augot, Matthieu Finiasz, and Pierre Loidreau. Using the Trace Operator to repair the Polynomial Reconstruction based Cryptosystem presented at Eurocrypt 2003.

[20] Shi Bai, Tancrède Lepoint, Adeline Roux-Langlois, Amin Sakzad, Damien Stehlé, and Ron Steinfeld. Improved security proofs in lattice-based cryptography: using the Rényi divergence rather than the statistical distance. *Journal of Cryptology*, 31:610–640, 2018.

[21] Marco Baldi, Alessandro Barengh, Luke Beckwith, Jean-François Biasse, Andre Esser, Kris Gaj, Kamyar Mohajerani, Gerardo Pelosi, Edoardo Persichetti, Markku-Juhani O. Saarinen, Paolo Santini, and Robert Wallace. LESS: Linear equivalence signature scheme, 2023.

[22] Magali Bardet and Manon Bertin. Improvement of algebraic attacks for solving superdetermined minrank instances. In Jung Hee Cheon and Thomas Johansson, editors, *Post-Quantum Cryptography*, pages 107–123, Cham, 2022. Springer International Publishing.

[23] Magali Bardet, Pierre Briaud, Maxime Bros, Philippe Gaborit, and Jean-Pierre Tillich. Revisiting algebraic attacks on minrank and on the rank decoding problem. *Designs, Codes and Cryptography*, 91(11):3671–3707, Nov 2023.

[24] Magali Bardet, Maxime Bros, Daniel Cabarcas, Philippe Gaborit, Ray Perlner, Daniel Smith-Tone, Jean-Pierre Tillich, and Javier Verbel. Improvements of algebraic attacks for solving the rank decoding and minrank problems. In Shiho Moriai and Huaxiong Wang, editors, *Advances in Cryptology – ASIACRYPT 2020*, pages 507–536, Cham, 2020. Springer International Publishing.

[25] Magali Bardet, Ayoub Otmani, and Mohamed Saeed-Taha. Permutation code equivalence is not harder than graph isomorphism when hulls are trivial. In *2019 IEEE International Symposium on Information Theory (ISIT)*, pages 2464–2468. IEEE, 2019.

[26] Alessandro Barenghi, Jean-François Biasse, Edoardo Persichetti, and Paolo Santini. LESS-FM: fine-tuning signatures from the code equivalence problem. In *Post-Quantum Cryptography: 12th International Workshop, PQCrypto 2021, Daejeon, South Korea, July 20–22, 2021, Proceedings 12*, pages 23–43. Springer, 2021.

[27] Alessandro Barenghi, Jean-François Biasse, Edoardo Persichetti, and Paolo Santini. LESS-FM: fine-tuning signatures from the code equivalence problem. In *International Conference on Post-Quantum Cryptography*, pages 23–43. Springer, 2021.

[28] Jessica Bariffi, Hannes Bartz, Gianluigi Liva, and Joachim Rosenthal. On the properties of error patterns in the constant Lee weight channel. In *International Zurich Seminar on Information and Communication (IZS 2022)*, pages 44–48. ETH Zürich, 2021.

[29] Jessica Bariffi, Karan Khathuria, and Violetta Weger. Information set decoding for lee-metric codes using restricted balls. In *Code-Based Cryptography Workshop*, pages 110–136. Springer, 2022.

[30] Anja Becker, Antoine Joux, Alexander May, and Alexander Meurer. Decoding Random Binary Linear Codes in $2^{n/20}$: How $1+1=0$ Improves Information Set Decoding. In David Pointcheval and Thomas Johansson, editors, *Advances in Cryptology - EUROCRYPT 2012*, volume 7237 of *Lecture Notes in Computer Science*, pages 520–536. Springer, 2012.

[31] Luke Beckwith, Robert Wallace, Kamyar Mohajerani, and Kris Gaj. A high-performance hardware implementation of the less digital signature scheme. In Thomas Johansson and Daniel Smith-Tone, editors, *Post-Quantum Cryptography*, pages 57–90, Cham, 2023. Springer Nature Switzerland.

[32] Eli Ben-Sasson, Iddo Bentov, Yinon Horesh, and Michael Riabzev. Fast Reed-Solomon Interactive Oracle Proofs of Proximity. *Electron. Colloquium Comput. Complex.*, TR17, 2017.

[33] Elwyn Berlekamp, Robert McEliece, and Henk Van Tilborg. On the inherent intractability of certain coding problems. *IEEE Transactions on Information Theory*, 24(3):384–386, 1978.

[34] Daniel J Bernstein. Grover vs. McEliece. In *International Workshop on Post-Quantum Cryptography*, pages 73–80. Springer, 2010.

[35] Daniel J Bernstein, Tanja Lange, and Christiane Peters. Attacking and defending the McEliece cryptosystem. In *Post-Quantum Cryptography: Second International Workshop, PQCrypto 2008 Cincinnati, OH, USA, October 17-19, 2008 Proceedings 2*, pages 31–46. Springer, 2008.

[36] Ward Beullens. Not enough less: An improved algorithm for solving code equivalence problems over $\mathbb{F}_q$. In *Selected Areas in Cryptography: 27th International Conference, Halifax, NS, Canada (Virtual Event), October 21-23, 2020, Revised Selected Papers*, pages 387–403. Springer, 2021.

[37] Jean-François Biasse, Giacomo Micheli, Edoardo Persichetti, and Paolo Santini. LESS is more: code-based signatures without syndromes. In *Progress in Cryptology - AFRICACRYPT 2020: 12th International Conference on Cryptology in Africa, Cairo, Egypt, July 20–22, 2020, Proceedings 12*, pages 45–65. Springer, 2020.

[38] Maxime Bombar and Alain Couvreur. Decoding supercodes of gabidulin codes and applications to cryptanalysis. In *Post-Quantum Cryptography: 12th International Workshop, PQCrypto 2021, Daejeon, South Korea, July 20–22, 2021, Proceedings*, page 3–22, Berlin, Heidelberg, 2021. Springer-Verlag.

[39] Sarah Bordage, Mathieu Lhotel, Jade Nardi, and Hugues Randriam. Interactive Oracle proofs of Proximity to Algebraic Geometry Codes. *Proceedings of the 37th Computational Complexity Conference*, 2020.

[40] Leif Both and Alexander May. Decoding Linear Codes with High Error Rate and Its Impact for LPN Security. In Tanja Lange and Rainer Steinwandt, editors, *Post-Quantum Cryptography - 9th International Conference, PQCrypto 2018*, volume 10786 of *Lecture Notes in Computer Science*, pages 25–46. Springer, 2018.

[41] Alessio Caminata and Elisa Gorla. Solving multivariate polynomial systems and an invariant from commutative algebra. In *Arithmetic of finite fields*, volume 12542 of *Lecture Notes in Comput. Sci.*, pages 3–36. Springer, Cham, [2021] ©2021.

[42] Florent Chabaud and Jacques Stern. The cryptographic security of the syndrome decoding problem for rank distance codes. In *International Conference on the Theory and Application of Cryptology and Information Security*, pages 368–381. Springer, 1996.

[43] André Chailloux, Thomas Debris-Alazard, and Simona Etinski. Classical and quantum algorithms for generic syndrome decoding problems and applications to the Lee metric. In *Post-Quantum Cryptography: 12th International Workshop, PQCrypto 2021, Daejeon, South Korea, July 20–22, 2021, Proceedings 12*, pages 44–62. Springer, 2021.

[44] Tung Chou, Ruben Niederhagen, Edoardo Persichetti, Lars Ran, Tovohery Hajatiana Ran-

drianarisoa, Krijn Reijnders, Simona Samardjiska, and Monika Trimoska. Matrix equivalence digital signature. 2023.

[45] Tung Chou, Ruben Niederhagen, Edoardo Persichetti, Tovohery Hajatiana Randrianarisoa, Krijn Reijnders, Simona Samardjiska, and Monika Trimoska. Take your MEDS: Digital signatures from matrix code equivalence. In *International Conference on Cryptology in Africa*, pages 28–52. Springer, 2023.

[46] Tung Chou, Edoardo Persichetti, and Paolo Santini. On linear equivalence, canonical forms, and digital signatures. *Cryptology ePrint Archive*, 2023.

[47] Jean-Sébastien Coron. Cryptanalysis of a Public-Key Encryption Scheme Based on the Polynomial Reconstruction Problem. In Feng Bao, Robert Deng, and Jianying Zhou, editors, *Public Key Cryptography – PKC 2004*, pages 14–27. Springer, 2004.

[48] Joshua Davies. A walkthrough of a TLS 1.3 handshake, 2019. Accessed: Feb. 19, 2024.

[49] Sanjay Deshpande, James Howe, Jakub Szefer, and Dongze Yue. Sdith in hardware. Cryptology ePrint Archive, Paper 2024/069, 2024. https://eprint.iacr.org/2024/069.

[50] Ilya Dumer. On minimum distance decoding of linear codes. In *Proc. 5th Joint Soviet-Swedish Int. Workshop Inform. Theory*, pages 50–52. Moscow, 1991.

[51] J.-P. D'Anvers, F. Vercauteren, and I. Verbauwhede. The impact of error dependencies on Ring/Mod-LWE/LWR based schemes. In *PQCrypto*. Springer, 2019.

[52] Andre Esser, Alexander May, and Floyd Zweydinger. McEliece needs a break–solving McEliece-1284 and quasi-cyclic-2918 with modern ISD. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 433–457. Springer, 2022.

[53] Andre Esser, Sergi Ramos-Calderer, Emanuele Bellini, José Ignacio Latorre, and Marc Manzano. Hybrid Decoding – Classical-Quantum Trade-Offs for Information Set Decoding. Cryptology ePrint Archive, Paper 2022/964, 2022.

[54] Cédric Faure and Pierre Loidreau. A New Public-Key Cryptosystem Based on the Problem of Reconstructing p–Polynomials. In Øyvind Ytrehus, editor, *Coding and Cryptography*, Lecture Notes in Computer Science, pages 304–315. Springer, 2006.

[55] L. De Feo and S. Galbraith. SeaSign: Compact isogeny signatures from class group actions. In Y. Ishai and V. Rijmen, editors, *Advances in Cryptology – EUROCRYPT 2019*, volume 11478 of *Lecture Notes in Computer Science*, pages 759–789. Springer, 2019.

[56] Philippe Gaborit, Ayoub Otmani, and Hervé Talé Kalachi. Polynomial-time key recovery attack on the Faure–Loidreau scheme based on Gabidulin codes. *Designs, Codes and Cryptography*, 86(7):1391–1403, 2018.

[57] Philippe Gaborit, Olivier Ruatta, and Julien Schrek. On the complexity of the rank syndrome decoding problem. *IEEE Transactions on Information Theory*, 62(2):1006–1019, 2015.

[58] Qian Guo, Thomas Johansson, and Paul Stankovski. A key recovery attack on mdpc with cca security using decoding errors. In *Advances in Cryptology–ASIACRYPT 2016: 22nd International Conference on the Theory and Application of Cryptology and Information Security, Hanoi, Vietnam, December 4-8, 2016, Proceedings, Part I 22*, pages 789–815. Springer, 2016.

[59] Murat Güzeltepe and Selda Çalkavur. Skew-cyclic codes based public-key cryptosystem approach. *Security and Privacy*, 5(6):e255, 2022.

[60] Anna-Lena Horlemann-Trautmann and Violetta Weger. Information set decoding in the Lee metric with applications to cryptography. *Advances in Mathematics of Communications*, 15(4), 2021.

[61] Felicitas Hörmann and Wessel van Woerden. FuLeakage: Breaking FuLeeca by learning attacks. Cryptology ePrint Archive, Paper 2024/353, 2024.

[62] Ghazal Kachigar and Jean-Pierre Tillich. Quantum Information Set Decoding Algorithms. Cryptology ePrint Archive, Paper 2017/213, 2017.

[63] Daniel Kales, Sebastian Ramacher, Christian Rechberger, Roman Walch, and Mario Werner. Efficient FPGA implementations of LowMC and Picnic. pages 417–441.

[64] M. J. Kannwischer, R. Petri, J. Rijneveld, P. Schwabe, and K. Stoffelen. PQM4: Post-quantum crypto library for the ARM Cortex-M4.

[65] M. J. Kannwischer, P. Schwabe, D. Stebila, and T. Wiggers. Improving software quality in cryptography standardization projects. In *IEEE European Symposium on Security and Privacy Workshops*, 2022.

[66] V. F. Kolchin. *Random Graphs*. Encyclopedia of Mathematics and its Applications. Cambridge University Press, 1998.

[67] Julien Lavauzelle, Pierre Loidreau, and Ba-Duc Pham. RAMESSES, a Rank Metric Encryption Scheme with Short Keys.

[68] Pil Joong Lee and Ernest F Brickell. An observation on the security of McEliece's public-key cryptosystem. In *Workshop on the Theory and Application of of Cryptographic Techniques*, pages 275–280. Springer, 1988.

[69] Pierre Loidreau. Properties of codes in rank metric. *arXiv preprint cs/0610057*, 2006.

[70] Farshid Haidary Makoui, Thomas Aaron Gulliver, and Mohammad Dakhilalian. A new code-based digital signature based on the McEliece cryptosystem. *IET Communications*, 2023.

[71] Alexander May, Alexander Meurer, and Enrico Thomae. Decoding Random Linear Codes in $\tilde{\mathcal{O}}(2^{0.054n})$. In Dong Hoon Lee and Xiaoyun Wang, editors, *Advances in Cryptology - ASIACRYPT 2011*, volume 7073 of *Lecture Notes in Computer Science*, pages 107–124. Springer, 2011.

[72] Robert J McEliece. A public-key cryptosystem based on algebraic coding theory. *Coding Theory*, 4244:114–116, 1978.

[73] C. A. Melchor, N. Aragon, S. Bettaieb, L. Bidoux, O. Blazy, J. Bos, J.-C. Deneuville, A. Dion, P. Gaborit, J. Lacan, E. Persichetti, J.-M. Robert, P. Véron, and G. Zémor. Hamming quasi-cyclic (HQC). https://pqc-hqc.org/, 2023.

[74] Carlos Aguilar Melchor, Nicolas Aragon, Slim Bettaieb, Loıc Bidoux, Olivier Blazy, Jean-Christophe Deneuville, Philippe Gaborit, Edoardo Persichetti, Gilles Zémor, and IC Bourges. Hamming quasi-cyclic (HQC). *NIST PQC Round*, 2(4):13, 2018.

[75] Alessio Meneghetti, Alex Pellegrini, and Massimiliano Sala. On the equivalence of two post-quantum cryptographic families. *Annali di Matematica Pura ed Applicata (1923 -)*, 202:967–991, 2021.

[76] Vincent Migliore. Crypto RISCV, 2022.

[77] Rafael Misoczki, Jean-Pierre Tillich, Nicolas Sendrier, and Paulo S. L. M. Barreto. MDPC-McEliece: New McEliece variants from moderate density parity-check codes. In *2013 IEEE International Symposium on Information Theory*, pages 2069–2073, 2013.

[78] Rafael Misoczki, Jean-Pierre Tillich, Nicolas Sendrier, and Paulo SLM Barreto. MDPC-McEliece: New McEliece variants from moderate density parity-check codes. In *2013 IEEE International Symposium on Information Theory*, pages 2069–2073, 2013.

[79] Shintaro Narisada, Kazuhide Fukushima, and Shinsaku Kiyomoto. Multiparallel MMT: Faster ISD algorithm solving high-dimensional syndrome decoding problem. *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, 106(3):241–252, 2023.

[80] Phong Q. Nguyen and Oded Regev. Learning a parallelepiped: Cryptanalysis of GGH and NTRU signatures. In *EUROCRYPT 2006*, volume 4004 of *LNCS*, pages 271–288, 2006.

[81] H. Niederreiter. Knapsack-type cryptosystems and algebraic coding theory. *Prob. Contr. Inform. Theory*, 15(2):157–166, 1986.

[82] NIST. Submission Requirements and Evaluation Criteria for the Post-Quantum Cryptography Standardization Process. https://csrc.nist.gov/CSRC/media/Projects/Post-Quantum-Cryptography/documents/call-for-proposals-final-dec-2016.pdf, 2016. [last accessed 2-Feb-2024].

[83] NIST. Call for Additional Digital Signature Schemes for the Post-Quantum Cryptography Standardization Process, 2023. https://csrc.nist.gov/projects/pqc-dig-sig/standardization/call-for-proposals.

[84] Cyrius Nugier and Vincent Migliore. Acceleration of classic mceliece post-quantum cryptosystem with cache processing. *IEEE Micro*, 2023.

[85] Open Quantum Safe project. OQS-OpenSSL_1_1_1, 2023. Accessed: Apr. 10, 2024.

[86] Alexei V Ourivski and Thomas Johansson. New technique for decoding codes in the rank metric and its cryptography applications. *Problems of Information Transmission*, 38:237–246, 2002.

[87] Pavel Panteleev and Gleb Kalachev. Asymptotically good quantum and locally testable classical LDPC codes. In *Proceedings of the 54th Annual ACM SIGACT Symposium on Theory of Computing*, pages 375–388, 2022.

[88] Alex Pellegrini and Giovanni Tognolini. Breaking hwqcs: a code-based signature scheme from high weight qc-ldpc codes. *Cryptology ePrint Archive*, 2024.

[89] Edoardo Persichetti and Paolo Santini. A new formulation of the linear equivalence problem and shorter less signatures, 2023.

[90] Edoardo Persichetti and Paolo Santini. A new formulation of the linear equivalence problem and shorter less signatures. In Jian Guo and Ron Steinfeld, editors, *Advances in Cryptology – ASIACRYPT 2023*, pages 351–378, Singapore, 2023. Springer Nature Singapore.

[91] Eugene Prange. The use of information sets in decoding cyclic codes. *IRE Trans. Information Theory*, 8:5–9, 1962.

[92] Oded Regev and Ricky Rosen. Lattice problems and norm embeddings. In *Proceedings of the thirty-eighth annual ACM symposium on Theory of Computing*, pages 447–456, 2006.

[93] Julian Renner, Sven Puchinger, and Antonia Wachter-Zeh. LIGA: A cryptosystem based on the hardness of rank-metric list and interleaved decoding. *Designs, Codes and Cryptography*, 89(6):1279–1319, 2021.

[94] Stefan Ritterhoff, Georg Maringer, Sebastian Bitzer, Violetta Weger, Patrick Karl, Thomas Schamberger, Jonas Schupp, and Antonia Wachter-Zeh. FuLeeca. *Round 1 of the NIST call for additional digital signatures*, 2023.

[95] Stefan Ritterhoff, Georg Maringer, Sebastian Bitzer, Violetta Weger, Patrick Karl, Thomas Schamberger, Jonas Schupp, and Antonia Wachter-Zeh. FuLeeca: A Lee-based signature scheme. In *CBCrypto 2023*, volume 14311 of *LNCS*, pages 56–83, 2023.

[96] Stefan Ritterhoff, Georg Maringer, Sebastian Bitzer, Violetta Weger, Patrick Karl, Thomas

Schamberger, Jonas Schupp, and Antonia Wachter-Zeh. FuLeeca: A Lee-based signature scheme. In *Code-Based Cryptography Workshop*, pages 56–83. Springer, 2023.

[97] Claus-Peter Schnorr and M. Euchner. Lattice basis reduction: Improved practical algorithms and solving subset sum problems. *Math. Program.*, 66:181–199, 1994.

[98] Igor Semaev and Andrea Tenti. Probabilistic analysis on Macaulay matrices over finite fields and complexity of constructing Gröbner bases. *Journal of Algebra*, 565:651 – 674, 2021. Cited by: 5; All Open Access, Hybrid Gold Open Access.

[99] Nicolas Sendrier. Finding the permutation between equivalent linear codes: The support splitting algorithm. *IEEE Transactions on Information Theory*, 46(4):1193–1203, 2000.

[100] Nicolas Sendrier and Dimitris E Simos. The Hardness of Code Equivalence over F_q and Its Application to Code-Based Cryptography. In *Post-Quantum Cryptography: 5th International Workshop, PQCrypto 2013, Limoges, France, June 4-7, 2013. Proceedings 5*, pages 203–216. Springer, 2013.

[101] Jacques Stern. A method for finding codewords of small weight. In *International Colloquium on Coding Theory and Applications*, pages 106–113. Springer, 1988.

[102] Jacques Stern. A method for finding codewords of small weight. In *Coding Theory and Applications: 3rd International Colloquium Toulon, France, November 2–4, 1988 Proceedings 3*, pages 106–113. Springer, 1989.

[103] Chik How Tan and Theo Fanuela Prabowo. High weight code-based signature scheme from qc-ldpc codes. In Hwajeong Seo and Suhri Kim, editors, *Information Security and Cryptology – ICISC 2023*, pages 306–323, Singapore, 2024. Springer Nature Singapore.

[104] Gang Tang, Dung Hoang Duong, Antoine Joux, Thomas Plantard, Youming Qiao, and Willy Susilo. Practical Post-Quantum Signature Schemes from Isomorphism Problems of Trilinear Forms. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 582–612. Springer, 2022.

[105] Violetta Weger, Karan Khathuria, Anna-Lena Horlemann, Massimo Battaglioni, Paolo Santini, and Edoardo Persichetti. On the hardness of the Lee syndrome decoding problem. *Advances in Mathematics of Communications*, 18(1):233–266, 2024.

[106] Cankun Zhao, Neng Zhang, Hanning Wang, Bohan Yang, Wenping Zhu, Zhengdong Li, Min Zhu, Shouyi Yin, Shaojun Wei, and Leibo Liu. A compact and high-performance hardware architecture for CRYSTALS-dilithium. *TCHES*, 2022(1):270–295, 2022.

[107] Selda Çalkavur. A new public-key cryptosystem based on LCD codes. *Avrupa Bilim ve Teknoloji Dergisi*, (28):320–324, 2021.