Enhancing Code Based Zero-knowledge Proofs using Rank Metric

Emanuele Bellini[1], Philippe Gaborit[3], Alexandros Hasikos[12], and Victor Mateu[1]

[1] Cryptography Research Centre, Technology Innovation Institute
emanuele.bellini@tii.ae, alexandros.hasikos@tii.ae, victor.mateu@tii.ae
[2] Universitat Pompeu Fabra, Barcelona, Spain
[3] University of Limogés
gaborit@unilim.fr

Abstract. The advent of quantum computers is a threat to most currently deployed cryptographic primitives. Among these, zero-knowledge proofs play an important role, due to their numerous applications. The primitives and protocols presented in this work base their security on the difficulty of solving the Rank Syndrome Decoding (RSD) problem. This problem is believed to be hard even in the quantum model. We first present a perfectly binding commitment scheme. Using this scheme, we are able to build an interactive zero-knowledge proof to prove: the knowledge of a valid opening of a committed value, and that the valid openings of three committed values satisfy a given linear relation, and, more generally, any bitwise relation. With the above protocols it becomes possible to prove the relation of two committed values for an arbitrary circuit, with quasi-linear communication complexity and a soundness error of 2/3. To our knowledge, this is the first quantum resistant zero-knowledge protocol for arbitrary circuits based on the RSD problem. An important contribution of this work is the selection of a set of parameters, and an a full implementation, both for our proposal in the rank metric and for the original LPN based one by Jain et. al in the Hamming metric, from which we took the inspiration. Beside demonstrating the practicality of both constructions, we provide evidence of the convenience of rank metric, by reporting performance benchmarks and a detailed comparison.

Keywords: Post Quantum · Code-based cryptography · Rank metric · Zero-knowledge proof · Identification protocol · Commitment scheme

1 Introduction

Due to the results of Grover [21] (1996) and Shor [33] (1997), the advancements in quantum information theory, and the discovery of new technologies, quantum computers are becoming more and more of a threat to the currently deployed cryptosystems, especially to those based on public key cryptography. Among these, *zero-knowledge proofs* (ZKP) are gaining particular attention due to their numerous applications. They can be used to obtain identification and login mechanisms, cryptographic signature schemes, systems to enforce honest behaviour of the users, and to prove statements in public transaction systems such as blockchains. The growing interest from both academia and industry on the ZKP topic, has led to a series of results that improve upon previous theory and allow for the development of practical applications, and a standardization effort for zero-knowledge systems is also being carried on by the cryptographic community [34,37]. On the other hand, most of ZKP schemes are not quantum resistant.

Zero-knowledge proofs were first introduced by Goldwasser, Micali and Rackoff in 1989 [20]. In their work, they created a new proving procedure for communicating a proof, or in modern terms, an efficient *interactive proof system*. An interactive proof is a process in which a prover probabilistically convinces a verifier of the correctness of a mathematical proposition, also called statement. If the proof does not reveal to the verifier any additional information about the mathematical proposition, except if it is true or not, then it is called a *zero-knowledge proof*. A *zero-knowledge proof of knowledge* of a secret information is a special case of zero-knowledge proof, in which the statement consists only of the fact that the prover knows the secret information. Goldreich, Micali and Wigderson [19] showed how to make any proving system in NP (i.e. where the verifier is a deterministic, polynomial-time machine) zero knowledge, meaning that the verifier learns nothing but the correctness of the proposition. Furthermore, Impagliazzo and Yung in 1987 [22], and Ben-Or et al. in 1990 [8], showed that anything that can be proved by an interactive proof system can be proved with zero knowledge. Zero-knowledge proofs therefore provide complete privacy to the prover while convincing the verifier. Further research resulted in the study of non-interactive zero-knowledge proofs (NIZKs), a variant that does not require interaction between the prover and the verifier. Building on top of these, modern NIZK systems have become more efficient, including succinct proofs, sub-linear verifiers and highly efficient provers.

In this work, we will focus on quantum resistant interactive zero-knowledge proofs, with the property of public-coin, i.e. verifier's random coins are made public throughout the proof protocol. Notice that, a public-coin interactive proof of knowledge can always be converted into a non-interactive proof of knowledge by means of the Fiat-Shamir heuristic [14]. Furthermore, if the interactive proof is used as an identification tool, then the non-interactive version can be used directly as a digital signature.

1.1 Our contribution

A commitment scheme is a cryptographic primitive that allows one to commit to a chosen value (or chosen statement) while keeping it hidden to others, with the ability to reveal (or to *open*) the committed value later. Commitment schemes are designed so that a party cannot change the value or statement after they have committed to it: that is, commitment schemes are binding.

In this work, we design and implement a perfectly binding and computationally hiding commitment scheme whose security relies on the hardness of solving the Rank Syndrome Decoding (RSD) problem, i.e. on the hardness of decoding random linear codes in the rank metric. This problem is believed to be hard even in the quantum model. Using this scheme, we are able to build an interactive zero-knowledge proof to prove: the knowledge of a valid opening of a committed value, and that the valid openings of three committed values satisfy a given linear relation, and, more generally, any bitwise relation.

With the above protocols it becomes possible to prove that the committed values c_0, c_1 satisfy $c_0 = C(c_1)$ for an arbitrary circuit C. As proved in [23], the total communication complexity of this protocol is $\mathcal{O}(|C|\mu \log \mu)$ where μ is the length of the committed messages. The soundness error is $2/3$, and thus for most applications must be lowered by (parallel) repetition.

Moreover, we also compute secure parameters, and implement[4] both schemes in the rank and Hamming metric, and compare their performances. Notice that, in [23], no parameters, nor an implementation was provided. Our proposal generates proofs that are 60% smaller and the size of the public parameters required is only a 1% with respect to the public parameters for the Hamming metric.

To our knowledge, this is the first zero-knowledge protocol for arbitrary circuits whose security relies on the difficulty of solving the Rank Syndrome Decoding problem, and the collision resistance of a hash function.

In subsection 1.2, we give an overview of the works related to our result. In section 2, we introduce the basic notions needed to understand our scheme. In section 3, we define a commitment scheme, and below it, in section 4, we build our zero-knowledge protocols. In section 5, we select a set of parameters both for our scheme and for its analogue in the Hamming metric, and we provide benchmarks of our implementations of the corresponding ZKP protocols. Finally, in section 6, we draw the conclusions.

1.2 Related works

This work is an adaptation of the protocols presented by Jain et al. in [23], where they show how to build a zero-knowledge protocol for arbitrary circuits reducing the security of their system to the difficulty of solving the Learning Parity with Noise problem, or, equivalently, to the difficulty of decoding a random linear code in the Hamming metric.

In turn, Jain's work is based on the preliminary identification protocol proposed by Stern in 1993 [35,36], which inspired a long sequence of works improving either the scheme parameter size, or the communication cost. All the subsequent schemes derived from Stern's can be divided in four categories:

- **Type 1**: 3-pass protocols using the parity-check matrix of a code,
- **Type 2**: 3-pass protocols using the generator matrix of a code,
- **Type 3**: 5-pass protocols using the parity-check matrix of a code,
- **Type 4**: 5-pass protocols using the generator matrix of a code.

[4] A C++ implementation of the schemes described in this work can be found at https://github.com/ahasikos/rank_commitments.

Type 1 protocols can be seen as Zero-Knowledge Proof of Knowledge (ZKPoK) of a solution of an instance of the Syndrome Decoding problem for some specific code, where the syndrome is the public key and the corresponding error the private secret. As the original Stern proposal, they are 3-move Σ-protocols with a soundness error of 2/3, and perfect completeness. The original Stern proposal used binary linear codes over the Hamming metric. Also, a second variant minimizing the computing load was presented, but its longer proof renders it unpractical. Double circulant codes, again in the Hamming metric, were proposed in 2007 by Gaborit and Girault in [16]. In 2011, Gaborit et al. adapted their proposal with double circulant codes to rank metric, obtaining the most compact code based identification scheme of Type 1. In 2008, Stern scheme was also adapted to the lattice setting by Kawachi et. al [24], who also extended the initial identification scheme to an *anonymous* identification scheme.

Using a generator matrix rather than the parity-check matrix, allows to reduce the communication cost, at expense of a slightly larger private key. This is why Type 2 protocols were introduced, in 1997, by Veron, in [38]. Type 2 protocols use a secret message and a secret error as the private key, and their encoding under a public generator matrix as the public key. Initially, the advantage in the communication cost was due to the fact that the committed value, which needs to be revealed in the response phase, was in the code plain message space rather than in the encoded message space. In 2012, Jain et al. [23] pointed out that Veron scheme did not reach perfect zero-knowledge, and proposed a variation of it, which they then used to construct zero-knowledge proof of knowledge of linear and multiplicative relations between committed messages. Jain version, though, lost the feature that was reducing the communication cost, as their commitment value was in the error space, which had the same size as the encoded message space. In 2018, Bellini et al. proposed the rank metric version of Veron scheme, thought without providing a security proof, and their scheme was attacked in 2019 in [25]. This is, so far, the only Stern-based scheme that has been attacked.

Notice that Type 1 and Type 2 protocols are 3-pass Σ-protocols, with perfect completeness and a soundness error (often referred to as cheating probability) of 2/3. Type 3 and Type 4 protocols were introduced to reduce the soundness error from 2/3 to almost 1/2, by performing 5 steps instead of 3. This allows to run less parallel execution of the protocol to reach a smaller desired soundness error, and, sometimes, a smaller communication cost at expense of some extra computation.

The first Type 3 protocol was presented in the second variant of Stern's original proposal. However, also this alternative had a larger proof and was not practical. In 2010, Cayrel-Veron-El Yousfi Alaoui (CVE) [12] presented a 5-pass identification protocol with soundness error of $q/(2q-2)$, using codes over $\mathbb{F}_{q^m}$, this time improving significantly the communication cost compared to the initial 5-pass proposal by Stern. A version of CVE scheme in the rank metric is presented in [7], though lacking a security proof. It is worth noting that the parameters proposed for this particular rank version of CVE scheme do not improve key size nor communication cost with respect to the Hamming metric version. A lattice based version of CVE was presented in 2012 by Cayrel et al. [11], reaching a smaller public key, but larger private key and communication cost than CVE. This scheme also improves under all aspects the Type 1 lattice-based scheme of Kawachi et al. [24]

The first Type 4 protocol was presented in 2011 in [2] by Aguilar et al., where double circulant codes were used. The key size, the communication cost and the soundness error of this protocol were later significantly improved in 2019, by Bellini et al. in [6], by replacing the Hamming metric with the rank metric. A lattice based version of the Jain et al. protocol was presented by Martínez and Morillo in 2019 [29], where they also use some ideas from [26] and [40]. The authors do not propose a set of parameters and leave as future work an implementation of their scheme.

All the above mentioned protocols are believed to be secure even against quantum adversaries, thought the situation is more uncertain as far as it concern the analogue of the Fiat-Shamir transform for 5-pass protocols.

In the case of lattices, it is possible to construct zero-knowledge proofs using approaches different from Stern, as it was done, for example, in [31,28,26].

A summary of the above described Stern-like protocols can be found in Table 1.

Name	Ref.	Year	Metric	Setting	Aim	Notes
3-pass, with parity-check matrix						
Stern(1)	[35,36]	1993	Hamm.	Linear codes	Identification	-
Stern(2)	[35,36]	1993	Hamm.	Linear codes	Identification	Minimize computing load, proof not practical
GG	[16]	2007	Hamm.	Double Circulant codes	Identification	-
KTX	[24]	2008	Euclidean	Lattices	Anonymous Identification	-
GSZ	[18]	2011	Rank	Double Circulant codes	Identification	-
3-pass, with generator matrix						
Veron	[38]	1997	Hamm.	Linear codes	Identification	Not perfect ZK
JKPT	[23]	2012	Hamm.	Linear codes	ZKPoK of relations	-
BKLP	[9]	2015	Euclidean	Lattices	ZKPoK of relations	-
BCHMM	[7]	2018	Rank	Linear Codes	Signature	Attacked in [25]
This work	-	-	**Rank**	**Linear codes**	**ZKPoK of relations**	-
5-pass, with parity-check matrix						
Stern(3)	[35,36]	1993	Hamm.	Linear codes	Identification	Proof not practical
CVE	[12]	2010	Hamm.	q-ary Linear Code	Identification	-
CLRS	[11]	2012	Euclidean	Lattices	Identification	-
BCHMM	[7]	2018	Rank	Linear Codes	Signature	-
5-pass, with generator matrix						
AGS	[2]	2011	Hamm.	Double Circulant codes	Identification	-
BCGMM	[6]	2019	Rank	Double Circulant codes	Signature	-
MM	[29]	2019	Euclidean	Ideal Lattices	ZKPoK of relations	-

Table 1. Summary of Stern-like protocols.

2 Preliminaries and notations

2.1 Codes in the rank metric

We use $\mathcal{M}_{r,c}(R)$ and $\mathcal{M}^*_{r,c}(R)$ to denote, respectively, the set of all matrices and the set of all full rank matrices with r rows and c columns with entries over the ring R. Given $M_1 \in \mathcal{M}_{r,c_1}(R)$ and $M_2 \in \mathcal{M}_{r,c_2}(R)$, we indicate with $M_1 \| M_2$ the concatenation of the two matrices.

A linear $(n,k)_q$-code C is a vector subspace of $(\mathbb{F}_q)^n$ of dimension k, where k and n are positive integers such that $k < n$, q is a prime power, and $\mathbb{F}_q$ is the finite field with q elements. Elements of the vector space are called vectors or words, while elements of the code are called codewords. A matrix $G \in \mathcal{M}^*_{k,n}(\mathbb{F}_q)$ is called a generator matrix of C if its rows form a basis of C, i.e. $C = \{x \cdot G : x \in (\mathbb{F}_q)^k\}$. A matrix $H \in \mathcal{M}^*_{n-k,n}(\mathbb{F}_q)$ is called a parity-check matrix of C if $C = \{x \in (\mathbb{F}_q)^n : H \cdot x^T = 0\}$.

In this paper, we work with codes in the *rank metric*. Given a fixed basis $\beta = \{\beta_1, \dots, \beta_m\}$ of $(\mathbb{F}_q)^m$, a vector $a \in (\mathbb{F}_{q^m})^n$ can be represented as a matrix with entries in $\mathbb{F}_q$, by expanding each component of a_i with respect to β in a column $(a_{1,i}, \dots, a_{m,i})^T$, where $a_i = \sum_{j=1}^m a_{j,i}\beta_j, i = 1, \dots, n$. We define the rank $\mathsf{w}_\mathsf{R}(v)$ of a vector v as the rank of its *matrix representation*, with respect to β. We denote the previous matrix representation as $\phi_\beta(a)$, and by ϕ_β^{-1} the inverse map. In what follows, we will omit β as we consider it fixed.

To send a binary vector of a certain Hamming weight to *any* other vector of the same Hamming weight, it is sufficient to apply a random permutation to vector components. The map with the analogue property in the rank metric, i.e. sending a vector of a certain rank to *any* other vector of the same rank, can be defined as follows (see [18]).

Definition 1. *Let $Q \in \mathcal{M}^*_{m,m}(\mathbb{F}_q)$ be a q-ary matrix of size $m \times m$, $P \in \mathcal{M}^*_{n,n}(\mathbb{F}_q)$ be a q-ary matrix of size $n \times n$, and $v = (v_1, \dots, v_n) \in (\mathbb{F}_{q^m})^n$. We define the function $\Pi_{P,Q}$ such that $(\pi_1, \dots, \pi_n) = \Pi_{P,Q}(v) = \phi^{-1}(Q \cdot \phi(v) \cdot P) \in (\mathbb{F}_{q^m})^n$, where for $h = 1, \dots, n$, $\pi_h := \beta_1 \sum_{i=1}^m \sum_{j=1}^n Q_{1,i} v_{i,j} P_{j,h} + \dots + \beta_m \sum_{i=1}^m \sum_{j=1}^n Q_{m,i} v_{i,j} P_{j,h}$, with $\beta = \{\beta_1, \dots, \beta_m\}$ a basis of $(\mathbb{F}_q)^m$.*

In [18], it is proved that, for any $x, y \in (\mathbb{F}_{q^m})^n$, and any full rank $P \in \mathcal{M}^*_{n,n}(\mathbb{F}_q)$ and any full rank $Q \in \mathcal{M}^*_{m,m}(\mathbb{F}_q)$, then $\Pi_{P,Q}$ has the rank preserving property, i.e. $\mathsf{w}_\mathsf{R}(\Pi_{P,Q}(x)) = \mathsf{w}_\mathsf{R}(x)$, and is a linear mapping, i.e. $a\Pi_{P,Q}(x) + b\Pi_{P,Q}(y) = \Pi_{P,Q}(ax + by)$. Furthermore, for any $x, y \in (\mathbb{F}_{q^m})^n$ such that $\mathsf{w}_\mathsf{R}(x) = \mathsf{w}_\mathsf{R}(y)$, it is possible to find $P \in \mathcal{M}^*_{n,n}(\mathbb{F}_q)$ and $Q \in \mathcal{M}^*_{m,m}(\mathbb{F}_q)$ such that $x = \Pi_{P,Q}(y)$. The last property shows that, given a vector of a certain rank, it is possible to associate to it any other vector of the same rank by modifying P and Q. This property will be used in the zero-knowledge proof of the proposed scheme. Notice also that $\Pi_{P,Q}$ is invertible if P and Q are.

We denote by $\begin{bmatrix} n \\ s \end{bmatrix} = \prod_{i=0}^{s-1} \frac{q^n - q^i}{q^s - q^i}$ the number of s-dimensional vector subspaces of $(\mathbb{F}_q)^n$ over $\mathbb{F}_q$. A *ball* $B_R^r(a)$ in the rank metric of radius r centered in a vector $a \in (\mathbb{F}_{q^m})^n$ is the set of all vectors in rank distance at most r from a. It can be shown [39] that $|B_R^r(a)| = \sum_{i=1}^r \begin{bmatrix} m \\ i \end{bmatrix} \prod_{j=0}^{i-1} (q^n - q^j)$, which does not depend on a.

The following bound plays an important role in the choice of the parameters of our schemes.

Theorem 1 **(q-ary Gilbert-Varshamov Bound in rank metric [15]).** *Let $A^R_{q^m}(n,d)$ be the maximum cardinality of a linear block code over $\mathbb{F}_{q^m}$ of length n, size M, and minimum distance d in the rank metric. Then $A^R_{q^m}(n,d) \geq \frac{q^{mn}}{|B_R^{d-1}(0)|}$.*

Both in the Hamming and in the rank metric, random codes over $\mathbb{F}_q$ asymptotically achieve the Gilbert-Varshamov bound [15]. Furthermore, they have close to optimal correction capability [27]. This result is important for the scheme that we propose, as it allows to choose random generator (or parity-check) matrices as long as the code parameters respect the bound.

2.2 Rank Decoding problem

We now define the problem upon which the security of the commitment schemes we present is based. This problem is equivalent to the *decoding problem* for random linear codes, which consists of searching for the closest codeword to a given vector. More precisely, given G, $y = xG+e$, and the weight w, the decoding problem consists in finding the pair (x, e), where the weight of e is w. In the case of linear codes, it can be easily shown that the decoding problem is equivalent to the problem in which the syndrome $s = Hy$ of the received vector is given instead of the received vector itself. In this case we use the term Syndrome Decoding (SD) when referring to linear code in the Hamming metric, and Rank Syndrome Decoding (RSD) when referring to linear code in the rank metric.

Definition 2 (RSD Distribution). *Given the positive integers n, k, and ρ, the $RSD(n, k, \rho)$ Distribution chooses $H \leftarrow_{\$} \mathcal{M}^*_{n-k,n}(\mathbb{F}_{q^m})$ and $x \leftarrow_{\$} (\mathbb{F}_{q^m})^n$ such that $\mathsf{w_R}(x) = \rho$, and outputs $(H, H \cdot x^T)$*

Problem 1 (RSD Problem). On input $(H, y^T) \in \mathcal{M}^*_{n-k,n}(\mathbb{F}_{q^m}) \times (\mathbb{F}_{q^m})^{n-k}$ from the RSD distribution, the Rank Syndrome Decoding problem $\mathrm{RSD}(n, k, \rho)$ asks to find $x \in (\mathbb{F}_{q^m})^n$ such that $H \cdot x^T = y^T$ and $\mathsf{w_R}(x) = \rho$.

The previous problem can be defined correspondingly also in the Hamming metric, in which setting the problem has been proven to be NP-complete [10]. The RSD problem has recently been proven difficult with a probabilistic reduction to the Hamming scenario in [1]. By applying the transformation described in [1] it can be shown that the Decisional version of the RSD problem can be reduced to a search problem for the Hamming metric, providing some evidence on the hardness of the problem.

2.3 Commitment schemes

In this section we define a commitment scheme and the properties which are related to this paper.

Definition 3. *A triple of algorithms (*Setup,Com,Ver*) is called a* commitment scheme *if it satisfies the following:*

- *On input* 1^λ*, the setup algorithm* Setup *outputs the public commitment parameters* pp*.*
- *The commitment algorithm* Com *takes as inputs a message* m *from a message space* M *and a the the public commitment parameters* pp*, and outputs a commitment/opening pair* (c, d)*.*
- *The verification algorithm* Ver *takes the parameters* pp*, a message* m*, a commitment* c *and an opening* d *and outputs* true *or* false*.*

The commitment scheme we describe satisfies these security properties:

- *Correctness*: Ver evaluates to true if the inputs are honestly computed, i.e.,

$$\Pr[\mathsf{Ver}(\mathsf{pp}, \mathsf{m}, \mathsf{c}, \mathsf{d}) = \mathsf{true}; \mathsf{pp} \leftarrow_{\$} \mathsf{Setup}(1^\lambda), \mathsf{m} \in M, (\mathsf{c}, \mathsf{d}) \leftarrow_{\$} \mathsf{Com}(\mathsf{m}, \mathsf{pp})] = 1$$

- *Perfect binding*: With overwhelming probability over the choice of the public commitment parameters $\mathsf{pp} \leftarrow_{\$} \mathsf{Setup}(1^\lambda)$, no commitment c can be opened in two different ways, i.e.,

$$(\mathsf{Ver}(\mathsf{pp}, \mathsf{m}, \mathsf{c}, \mathsf{d}) = \mathsf{true}) \textbf{ and } (\mathsf{Ver}(\mathsf{pp}, \mathsf{m}', \mathsf{c}, \mathsf{d}') = \mathsf{true}) \implies \mathsf{m} = \mathsf{m}'$$

- *Computational hiding*: A commitment c computationally hides the committed message if, with overwhelming probability over the choice of the value $\mathsf{pp} \leftarrow_{\$} \mathsf{Setup}(1^\lambda)$, for every $\mathsf{m}, \mathsf{m}' \in M$, and for $(\mathsf{c}, \mathsf{d}) \leftarrow_{\$} \mathsf{Com}(\mathsf{m}, \mathsf{pp})$ and $(\mathsf{c}', \mathsf{d}') \leftarrow_{\$} \mathsf{Com}(\mathsf{m}', \mathsf{pp})$ the distributions c and c' are computationally indistinguishable.

2.4 Zero-knowledge proof of knowledge

A zero-knowledge proof of knowledge is a protocol in which P wants to prove to a V the knowledge of some secret information without revealing anything about it, except the fact that he knows it. More formally, in a zero-knowledge proof for a binary relation R, the two parties have a common input y and P has a private input w such that $(y, w) \in R$. To be defined as zero-knowledge, the protocol must then satisfy the following three properties:

- *Completeness*: for an honest prover, the verifier always accepts.
- *Zero-knowledge*: for every potentially malicious verifier V' there exists a PPT simulator only taking y as an input whose output is indistinguishable from conversations of V' with an honest prover.
- *Proof of knowledge*: from every prover P which can make the verifier accept with a probability larger than a threshold k (the *knowledge error*), a w' satisfying $(y, w') \in R$ can be extracted efficiently in a rewindable black-box way.

For a more formal definition we refer to Bellare and Goldreich [5].

3 A commitment scheme in the rank metric

In this section we describe a perfectly binding commitment scheme whose security depends on the difficulty of solving the Rank Syndrome Decoding (RSD) problem. This commitment scheme follows the structure of the commitment scheme presented [23], based on the LPN problem.

The scheme is parameterized by the following values: the prime characteristic q (in our implementation we set $q = 2$) and the degree m of a q-ary extension field $\mathbb{F}_{q^m}$, the bit length μ of a message $\mathsf{m} \in \mathbb{F}_q^{\mu}$, the bit length π of the randomness $\mathsf{s} \in \mathbb{F}_q^{\pi}$, the length n of the linear code C, and the rank weight ρ of an error $\mathsf{e} \in \mathbb{F}_{q^m}^n$. The dimension k of the code C is given by $k = (\mu+\pi)/m$ (we require μ and π to be both multiples of m, so that $(\mathsf{s}\|\mathsf{m})$ can be seen as an element of $\mathbb{F}_{q^m}^k$). Notice also that an instance of the RSD problem is hard if the weight ρ is taken close to the Gilbert-Varshamov bound. Once the scheme public parameters q, m, μ, π, n, ρ are chosen accordingly with the security parameter λ (see subsection 5.1 for an example of actual values), then the commitment scheme is defined by the following three algorithms (Setup, Com, Ver):

$\mathsf{Setup}(1^\lambda)$	$\mathsf{Com}_G(\mathsf{m})$	$\mathsf{Ver}_G(\mathsf{c}, \mathsf{m}', \mathsf{s}')$
$G_\mathsf{m} \leftarrow_\$ \mathcal{M}^*_{\frac{\mu}{m},n}(\mathbb{F}_{q^m})$	$\mathsf{s} \leftarrow_\$ \mathbb{F}_2^\pi$	$\mathsf{e}' = \mathsf{c} + (\mathsf{s}'\|\mathsf{m}') \cdot G$
$G_\mathsf{s} \leftarrow_\$ \mathcal{M}^*_{\frac{\pi}{m},n}(\mathbb{F}_{q^m})$	$\mathsf{e} \leftarrow_\$ \mathbb{F}_{q^m}^n$, s.t. $\mathsf{w_R}(\mathsf{e}) = \rho$	**if** $\mathsf{w_R}(\mathsf{e}') = \rho$ **return** True
return $G = (G_\mathsf{s}^\intercal \| G_\mathsf{m}^\intercal)^\intercal$	$\mathsf{c} = (\mathsf{s}\|\mathsf{m}) \cdot G + \mathsf{e}$	**else return** False
	return c, s	

The matrix G is called the *public commitment key*. We will write Com and Ver, omitting G, when clear from the context. The second output s of the Com algorithm is needed by the party generating the commitment, in order to prove that it was the one generating the commitment.

Theorem 2. *Let us fix q, m, μ, π, n, ρ so that the RSD problem is hard. Let $G \in \mathcal{M}^*_{k,n}(\mathbb{F}_{q^m})$ be the generator matrix of a random linear code C of dimension k and length n. Then the above defined commitment scheme is perfectly binding and computationally hiding.*

Proof. We first prove that the scheme is perfectly binding. First, let us recall that random linear codes over $\mathbb{F}_{q^m}$ asymptotically achieve the Gilbert-Varshamov bound. Thus, with overwhelming probability, the code C has minimum rank distance greater than $d_C = 2\rho$. This means that no codeword in C can have rank weight less than or equal to d_C. Now, let us assume, by contraposition, that there exists two different openings m, m' for a commitment c. This means that $\mathsf{e} = \mathsf{c} + (\mathsf{s}\|\mathsf{m}) \cdot G$ and $\mathsf{e}' = \mathsf{c} + (\mathsf{s}'\|\mathsf{m}') \cdot G$ are such that $\mathsf{w_R}(\mathsf{e}) = \mathsf{w_R}(\mathsf{e}') = \rho$. Since $\mathsf{e} + \mathsf{e}' = ((\mathsf{s}\|\mathsf{m}) + (\mathsf{s}'\|\mathsf{m}')) \cdot G \in C$, and because of the metric properties, we have that $\mathsf{w_R}(\mathsf{e} + \mathsf{e}') \leq \mathsf{w_R}(\mathsf{e}) + \mathsf{w_R}(\mathsf{e}') = 2\rho = d_C$. This means that the codeword $(\mathsf{e} + \mathsf{e}')$ has minimum rank weight less than the code distance. Since this is impossible, than m must be different from m'.

To prove that the scheme is computationally hiding, we first notice that $\mathsf{c} = \mathsf{s} \cdot G_\mathsf{s} + \mathsf{m} \cdot G_\mathsf{m} + \mathsf{e}$. Then we conclude that c is indistinguishable from a random vector of the same length, since both s and e are sampled from a random distribution, and we are assuming that $\mathsf{s} \cdot G_{\mathsf{s}} + \mathsf{e}$ is also indistinguishable from random. □

4 Zero Knowledge Proof protocols

In this section we describe three Σ-protocol. The first protocol is a proof of knowledge of a valid opening. It is a variant of Stern protocol [35], or, more precisely, of the dual of it due to Veron [38]. The second protocol allows to prove that committed strings satisfy any linear relation. Finally, the third protocol allows to prove that committed strings satisfy any bitwise relations, as bitwise AND, NAND, OR, or NOR. Since NAND is functionally complete, using this protocol we can construct Σ-protocols for any relation amongst committed messages. For all three protocols, we follow the ideas and proofs of [23], and adapt them to rank metric.

4.1 Proving knowledge of a valid opening

The following Σ-protocol proves knowledge of a valid opening for commitments of the form described in section 3, i.e., it shows possession of $\mathsf{s}, \mathsf{m}, \mathsf{e}$ such that $\mathsf{y} = (\mathsf{s} \| \mathsf{m}) \cdot G + \mathsf{e}$ for an error satisfying $\mathsf{w_R}(\mathsf{e}) = \rho$. For the sake of notation convenience, we will sometimes write x to denote the vector $(\mathsf{s} \| \mathsf{m})$. The protocol is described in Figure 1. The inputs for P are $\mathsf{x} \in (\mathbb{F}_{q^m})^k$ and $\mathsf{e} \in (\mathbb{F}_{q^m})^n$ s.t. $\mathsf{w_R}(\mathsf{e}) = \rho$. The pair (x, e) is the secret P wants to prove the knowledge of. Both P and V share as input the public parameters: the generator matrix G and the error rank weight ρ. The function $\mathsf{E}()$ takes a sequence of inputs and converts it to a binary string of size μ (a collision resistant hash or XOF function can be used), suitable to be used as input message for the Com or Ver algorithm. Notice that, the protocol uses Π as defined in subsection 2.1 with P and Q being invertible. This allows us to operate with f and $f + \mathsf{e}$ in a way that preserves the rank of the error but still hides it. The Π operation preserves linearity and is the key on the adaptation from Hamming to rank metric.

Theorem 3. *The protocol described in Figure 1 is a Σ-protocol for the following relation:* $\mathcal{R}_{RSD} = \{((G, \mathsf{y}), (\mathsf{s}, \mathsf{m}, \mathsf{e})) : \mathsf{y} = (\mathsf{s} \| \mathsf{m}) \cdot G + \mathsf{e}$ ***and*** $\mathsf{w_R}(\mathsf{e}) = \rho\}$

The proof of Theorem 3 can be found in section B.

4.2 Proving linear relations

As it is introduced in Jain et al. paper, our adaptation into rank metric is also suitable to prove linear relations of several valid openings. The main idea is to provide a method by which a prover P can prove knowledge of a bitwise relation between the committed messages without showing the messages. The whole construction is similar in the sense that the relation is still holding in the message space of the commitments.

Prover P		**Verifier** V
$Q \leftarrow_\$ \mathcal{M}^*_{m,m}(\mathbb{F}_q)$, $P \leftarrow_\$ \mathcal{M}^*_{n,n}(\mathbb{F}_q)$ invertibles		
$v \leftarrow_\$ (\mathbb{F}_{q^m})^k$		
$f \leftarrow_\$ (\mathbb{F}_{q^m})^n$		
$\mathsf{y} \leftarrow \mathsf{x} \cdot G + \mathsf{e}$		
$\mathsf{c}_0, \mathsf{s}_0 \leftarrow \mathsf{Com}(\mathsf{E}(P, Q, v \cdot G + f))$		
$\mathsf{c}_1, \mathsf{s}_1 \leftarrow \mathsf{Com}(\mathsf{E}(\Pi_{P,Q}(f)))$		
$\mathsf{c}_2, \mathsf{s}_2 \leftarrow \mathsf{Com}(\mathsf{E}(\Pi_{P,Q}(f + \mathsf{e})))$	$\xrightarrow{\mathsf{y}, \mathsf{c}_0, \mathsf{c}_1, \mathsf{c}_2}$	
	$\xleftarrow{\mathsf{ch}}$	$\mathsf{ch} \leftarrow_\$ \{0, 1, 2\}$
$\mathsf{r}_{P,Q} \leftarrow (P, Q)$		
$\mathsf{r}_0 \leftarrow v \cdot G + f$		
$\mathsf{r}_1 \leftarrow \Pi_{P,Q}(f)$		
$\mathsf{r}_2 \leftarrow \Pi_{P,Q}(f + \mathsf{e})$		
if $\mathsf{ch} = 0$	$\xrightarrow{\mathsf{r}_{P,Q}, \mathsf{r}_0, \mathsf{r}_1, \mathsf{s}_0, \mathsf{s}_1}$	**if** $\begin{cases} \mathsf{Ver}(\mathsf{c}_0, \mathsf{E}(\mathsf{r}_{P,Q}, \mathsf{r}_0), \mathsf{s}_0)) = \mathsf{true} \textbf{ and} \\ \mathsf{Ver}(\mathsf{c}_1, \mathsf{E}(\mathsf{r}_1), \mathsf{s}_1) = \mathsf{true} \textbf{ and} \\ \mathsf{r}_0 + \Pi^{-1}_{r_{P,Q}}(\mathsf{r}_1) \in \mathsf{Img}(G) \textbf{ and} \\ \mathsf{r}_{P,Q} \in \mathcal{M}^*_{m,m}(\mathbb{F}_q) \times \mathcal{M}^*_{n,n}(\mathbb{F}_q) \end{cases}$ **return** true
if $\mathsf{ch} = 1$	$\xrightarrow{\mathsf{r}_{P,Q}, \mathsf{r}_0, \mathsf{r}_2, \mathsf{s}_0, \mathsf{s}_2}$	**if** $\begin{cases} \mathsf{Ver}(\mathsf{c}_0, \mathsf{E}(\mathsf{r}_{P,Q}, \mathsf{r}_0), \mathsf{s}_0)) = \mathsf{true} \textbf{ and} \\ \mathsf{Ver}(\mathsf{c}_2, \mathsf{E}(\mathsf{r}_2), \mathsf{s}_2) = \mathsf{true} \textbf{ and} \\ \mathsf{r}_0 + \Pi^{-1}_{r_{P,Q}}(\mathsf{r}_2) + \mathsf{y} \in \mathsf{Img}(G) \end{cases}$ **return** true
if $\mathsf{ch} = 2$	$\xrightarrow{\mathsf{r}_1, \mathsf{r}_2, \mathsf{s}_1, \mathsf{s}_2}$	**if** $\begin{cases} \mathsf{Ver}(\mathsf{c}_1, \mathsf{E}(\mathsf{r}_1), \mathsf{s}_1)) = \mathsf{true} \textbf{ and} \\ \mathsf{Ver}(\mathsf{c}_2, \mathsf{E}(\mathsf{r}_2), \mathsf{s}_2) = \mathsf{true} \textbf{ and} \\ \mathsf{w_R}(\mathsf{r}_1 + \mathsf{r}_2) = \rho \end{cases}$ **return** true

Fig. 1: A Σ-protocol proving valid opening of a commitment in the rank metric.

Given the three q-ary vectors m_1, m_2, m_3 and two q-ary matrices $X_1, X_2 \in \mathcal{M}_{\mu,\mu}(\mathbb{F}_q)$ such that $\mathsf{m}_3 = X_1\mathsf{m}_1 + X_2\mathsf{m}_2$, a prover can prove in zero knowledge the existence of this relation by running the protocol detailed in Figure 2. P first commits to the values obtaining $\mathsf{y}_i = (\mathsf{s}_i||\mathsf{m}_i)G + \mathsf{e}_i$, and then generates v_1 and v_2 at random to later compute $v_3 = X_1v_1 + X_2v_2$. With this second set of values sharing the same linear relations the prover proceeds by proving valid opening of the v_i values using the proof from subsection 4.1 but now the verifier validates different computations regarding the linear relation and how it applies to either v_i or $v_i + \mathsf{m}_i$ depending on the challenge. The protocol protects the values m_i by masking them with the random values v_i which, given that they share the linear relations, can be evaluated without disclosing their values. It is worth noting that, both prover P and verifier V know the public parameters y_1, y_2, y_3, G, ρ, and the relations X_1 and X_2. On the other hand, only the prover P knows $\mathsf{x} = (\mathsf{s}_i||\mathsf{m}_i)$ and e.

Prover P		Verifier V
$Q_1, Q_2, Q_3 \leftarrow\$ \mathcal{M}^*_{m,m}(\mathbb{F}_q)$		
$P_1, P_2, P_3 \leftarrow\$ \mathcal{M}^*_{n,n}(\mathbb{F}_q)$		
$u_1, u_2, u_3 \leftarrow\$ (\mathbb{F}_q^{\pi})$		
$f_1, f_2, f_3 \leftarrow\$ (\mathbb{F}_{q^m}^{n})$		
$v_1, v_2 \leftarrow\$ (\mathbb{F}_q^{\mu})$		
$v_3 = X_1 v_1 + X_2 v_2$		
$\mathsf{c}_{i,0}, \mathsf{s}_{i,0} \leftarrow \mathsf{Com}(\mathsf{E}(P_i, Q_i, (u_i \| v_i) \cdot G + f_i))$		
$\mathsf{c}_{i,1}, \mathsf{s}_{i,1} \leftarrow \mathsf{Com}(\mathsf{E}(\varPi_{P_i,Q_i}(f_i)))$		
$\mathsf{c}_{i,2}, \mathsf{s}_{i,2} \leftarrow \mathsf{Com}(\mathsf{E}(\varPi_{P_i,Q_i}(f_i + \mathsf{e}_i)))$	$\xrightarrow{\mathsf{c}_{i,0}, \mathsf{c}_{i,1}, \mathsf{c}_{i,2}}$	
	$\xleftarrow{\mathsf{ch}}$	$\mathsf{ch} \leftarrow\$ \{0, 1, 2\}$
$\mathsf{r}_{i,P,Q} \leftarrow (P_i, Q_i)$		
$\mathsf{r}_{i,0} \leftarrow (u_i \| v_i) \cdot G + f_i$		
$\mathsf{r}_{i,1} \leftarrow \varPi_{P_i,Q_i}(f_i)$		
$\mathsf{r}_{i,2} \leftarrow \varPi_{P_i,Q_i}(f_i + \mathsf{e}_i)$		
if $\mathsf{ch} = 0$	$\xrightarrow{\mathsf{r}_{i,P,Q}, \mathsf{r}_{i,0}, \mathsf{r}_{i,1}, \mathsf{s}_{i,0}, \mathsf{s}_{i,1}}$	**if** $\begin{cases} \mathsf{Ver}(\mathsf{c}_{i,0}, \mathsf{E}(\mathsf{r}_{i,P,Q}, \mathsf{r}_{i,0}), \mathsf{s}_{i,0})) = \mathsf{true} \textbf{ and} \\ \mathsf{Ver}(\mathsf{c}_{i,1}, \mathsf{E}(\mathsf{r}_{i,1}), \mathsf{s}_{i,1}) = \mathsf{true} \textbf{ and} \\ \exists a_i, b_i \mid \mathsf{r}_{i,0} + \varPi^{-1}_{r_{i,P,Q}}(\mathsf{r}_{i,1}) = (a_i \| b_i) \cdot G \textbf{ and} \\ b_3 = b_1 X_1 + b_2 X_2 \textbf{ and} \\ \mathsf{r}_{i,P,Q} \in \mathcal{M}^*_{m,m}(\mathbb{F}_q) \times \mathcal{M}^*_{n,n}(\mathbb{F}_q) \end{cases}$ **return** true
if $\mathsf{ch} = 1$	$\xrightarrow{\mathsf{r}_{i,P,Q}, \mathsf{r}_{i,0}, \mathsf{r}_{i,2}, \mathsf{s}_{i,0}, \mathsf{s}_{i,2}}$	**if** $\begin{cases} \mathsf{Ver}(\mathsf{c}_{i,0}, \mathsf{E}(\mathsf{r}_{i,P,Q}, \mathsf{r}_{i,0}), \mathsf{s}_{i,0})) = \mathsf{true} \textbf{ and} \\ \mathsf{Ver}(\mathsf{c}_{i,2}, \mathsf{E}(\mathsf{r}_{i,2}), \mathsf{s}_{i,2}) = \mathsf{true} \textbf{ and} \\ \exists \mathsf{c}_i, d_i \mid \mathsf{r}_{i,0} + \varPi^{-1}_{r_{i,P,Q}}(\mathsf{r}_{i,2}) + \mathsf{y}_i = (\mathsf{c}_i \| d_i) \cdot G \textbf{ and} \\ d_3 = d_1 X_1 + d_2 X_2 \end{cases}$ **return** true
if $\mathsf{ch} = 2$	$\xrightarrow{\mathsf{r}_{i,1}, \mathsf{r}_{i,2}, \mathsf{s}_{i,1}, \mathsf{s}_{i,2}}$	**if** $\begin{cases} \mathsf{Ver}(\mathsf{c}_{i,1}, \mathsf{E}(\mathsf{r}_{i,1}), \mathsf{s}_{i,1})) = \mathsf{true} \textbf{ and} \\ \mathsf{Ver}(\mathsf{c}_{i,2}, \mathsf{E}(\mathsf{r}_{i,2}), \mathsf{s}_{i,2}) = \mathsf{true} \textbf{ and} \\ \mathsf{w_R}(\mathsf{r}_{i,1} + \mathsf{r}_{i,2}) = \rho \end{cases}$ **return** true

Fig. 2: A $\varSigma$-protocol proving linear relations of valid opening in the rank metric.

4.3 Proving multiplicative relations

When the properties we want to prove are multiplicative such as $\mathsf{m}_3 = \mathsf{m}_1 \circ \mathsf{m}_2$, we will follow the original idea and try to reduce the multiplicative relation into a linear relation in order to use the construction from subsection 4.2. In a nutshell, the prover P will have the commitments y_1 y_2, and y_3 of the messages m_1, m_2, and m_3. In order to prove the $\circ$ relation, P will begin sampling vectors m'_i sharing the same multiplicative relation and adding restrictions to its structure. After this, P will generate a random permutation matrix R such that $R \cdot m'_i = \mathsf{m}_i$. Finally, P will use the proof of linear relation with the linear relation R but, given that R is not known by V, it will send a commitment to R and also commitments to m'_i for $i = 1, 2, 3$. The detailed version of the protocol is presented in Figure 3 (commitment) and Figure 4 (challenge and response). The inputs for P are $\mathsf{m}_i \in \mathbb{F}_2^{\mu}$, $\mathsf{e}_i \in (\mathbb{F}_{q^m})^n$ s.t. $\mathsf{w_R}(\mathsf{e}_i) = \rho$, and $\mathsf{s}_i \in \mathbb{F}_2^{\pi}$ for $i = 1, 2, 3$. Both P and V share as input the public parameters: the generator matrix G, the error rank weight ρ, and the commitments y_i for $i = 1, 2, 3$. Besides this, they also share knowledge of the multiplicative relation $\circ$. For the purpose of readability, we use similar notation to the original Jain et al. protocol, which includes the use of m_i^j to denote the j-th bit block of m_i. Following this reasoning, R^j would be the submatrix resulting from taking only the columns from $(j-1)\mu m+1$ to $j\mu m$. We use the same notation for Q_i, Q'_i, P_i, and P'_i. Notice that, the conversion from Hamming to rank metric, is again made possible by the use of the functions $\Pi_{P,Q}$, which are linear mappings preserving the rank.

5 Implementation

In the original proposal from Jain et al. [23], no set of parameters was provided, and consequently no implementation to prove the efficiency of the scheme in a real scenario. In this section, we first fix a set of parameters for a quantum security level of 128 bit, and then we provide benchmarks of our implementation of the commitment schemes in the Hamming and the rank metric.

5.1 Parameters

For the Jain et al. commitment scheme we have to choose a proper set of parameters n, k, w such that the syndrome decoding problem in the Hamming metric can be solved with at least 2^{128} operations with a quantum or a classical computer. The difficulty of solving the syndrome decoding problem in the Hamming metric and in the *full distance decoding* scenario[5] and when $n \approx 2k$ (the hardest case), is given by $2^{0.097n}$ [30]. For quantum security, the exponent should be divided by two. To obtain a security level of λ bit in the quantum scenario, then $n = 2\lambda/0.097$. For $\lambda = 128$ we obtain $n = 2640, k = 1320$. Since the Gilbert-Varshamov bound for the given n, k is $d = 294$, we choose w close to this value, e.g. $w = 284$. We recall that, in [23], the dimension k is split in two values ℓ and v (in this paper corresponding to π and μ, respectively), where ℓ is the security parameter, resulting in $\ell = 128$ and $v = 1192$.

For our commitment scheme, we choose a proper set of parameters q, m, n, k, r such that the SD problem in the rank metric can not be solved with less than 2^{128} operations using a quantum or a classical computer. To obtain a security level of $\lambda = 128$ bit in the quantum scenario, we selected the following parameters: $q = 2, m = 43, n = 38, k = 17, \rho = 13$. Notice that the Gilbert-Varshamov bound for the given q, m, n, k is $d = 15$. Finally, we have $\pi = 129$ (greater than $\lambda = 128$)and $\mu = mk - \pi = 602$. The work factor (i.e. the base 2 logarithm of the attack time complexity) of the known attacks for the chosen parameters is summarized in Table 2. Since the cheating probability of the scheme is $2/3$, to reach 128 bit of security, we need to repeat the protocol δ times, where δ is the least integer such that $(2/3)^\delta < 2^{-128}$. This gives us $\delta = 219$.

[5] In the full distance decoding, the attacker receives an arbitrary point and aims at decoding the closest codeword. In the *half distance decoding*, the attacker knows that the error vector is within the error correction distance, i.e. $\mathsf{w}_\mathsf{H}(e) \leq \lfloor (d-1)/2 \rfloor$. In this case the decoding complexity is $2^{0.0473n}$.

Reference	Attack type	Complexity	Work factor
[13]	combinatorial	$(n\rho + m)^3 q^{(m-\rho)(\rho-1)/2}$	207.21
[32] (1)	combinatorial	$(m\rho)^3 q^{(\rho-1)(k+1)/2}$	135.38
[32] (2)	combinatorial	$(k+\rho)^3 \rho^3 q^{(m-\rho)(\rho-1)/2}$	205.82
[17] (1)	combinatorial	$(n-k)^3 m^3 q^{\rho\lfloor k*m/(2n)\rfloor}$	146.46
[17] (2)	combinatorial	$(n-k)^3 m^3 q^{(\rho-1)\lfloor (k-1)m/(2n)\rfloor}$	137.46
[3]	combinatorial	$(n-k)^3 m^3 q^{\rho\lceil (k+1)m/(2n)\rceil - m}$	**129.46**
[17] (3)	algebraic	$\rho^3 k^3 q^{\rho\lceil ((\rho+1)(k+1)-(n+1))/\rho\rceil}$	244.36
[4]	algebraic	$\left(\frac{((m+n)\rho)^{\rho+1}}{(\rho+1)!}\right)^w, w = 2.807$	292.55

Table 2. Work factor of the known attacks to the rank syndrome decoding problem for $q = 2, m = 43, n = 38, k = 17, \rho = 13$.

5.2 Sizes and communication cost comparison

Table 3 shows a comparison of the secret and public parameter sizes and the average communication cost of one round of the Σ-protocol of Figure 1, for a quantum security level of 128 bits, for both Hamming and rank metric. In the rank metric, the average communication cost is about 60% lower, while the public parameters size is two orders of magnitude smaller. However, the size of the secret in the ZKP is also 40% smaller. The size of the secret can be evaluated as both a benefit and a drawback. On one side, the proof is limited on the size of the secret, therefore, bigger secrets would also require bigger proofs. On the other side, this also means the proof is able to provide security to a smaller value. A common application of this last argument is a signature scheme, where the secret is the private key of the signer. In that case, the size of the private key could be smaller and, therefore, the scheme would be more efficient in terms of size. Notice that the communication cost could be reduced by using techniques similar to the ones presented in [6]. Also, secret and public parameters sizes could be reduced in both Hamming and rank metric by using ideal or quasi-cyclic codes instead of random linear codes.

		Parameters	|Secret|	|Public Param.|	Average Communication
Hamming [23]	Formula	(n, k, w)	$k + n$	$n + kn + \log_2(w)$	$5n + \lceil 2/3(n\log_2(n))\rceil + 2\lambda$
	Bits	(2640,1320,284)	3960	3487449	33461
Rank (this work)	Formula	(q, m, n, k, ρ)	$mk + mn$	$mn + mkn + \log_2(\rho)$	$5mn + \lceil 2/3(m^2 + n^2)\rceil + 2\lambda$
	Bits	(2,43,38,17,13)	2365	29416	10622

Table 3. Communication cost and parameters bit sizes of the Σ-protocol of Figure 1 for a quantum security level of 128 bits.

5.3 Performance comparison

We have implemented both Jain et. al. [23] schemes and ours. In both implementations we have used the NTL library from Victor Shoup. The implementations were written in C++ and the benchmarks were conducted on a 2.9 GHz Quad-Core Intel Core i7 with 16GB of LPDDR3 RAM clocked at 2133MHz.

Commitment Scheme					
Jain et. al.			**This work**		
Routine	**Subroutine**	**Time [ms]**	**Routine**	**Subroutine**	**Time [ms]**
Setup	Generate matrix A	1.303	Setup	Generate matrix G	0.030
Commitment	Generate random vector r	negl.	Commitment	Generate random vector s	negl.
	Generate error vector e	0.168		Generate error vector e	1.800
	Compute commitment c	0.029		Compute commitment c	0.025
	Total	0.197		**Total**	1.825
Verification	Recover error vector e	0.029	Verification	Recover error vector e	0.0250
	Compute hamming weight of e	0.001		Compute rank of e	0.0160
	Total	0.030		**Total**	0.041

Table 4. Commitment scheme performance comparison.

Table 4 depicts the performance in milliseconds of the two commitment schemes. In Table 5, we compare the performance of both Hamming and rank metric variants for the knowledge of a valid opening. Table 6 show the performance of the linear and multiplicative relations. For the latter two modes the comparison is more brief as the subroutines are mostly the same as in Table 5. The key outcomes of this comparison are the following. For the commitment scheme, the generation of the commitment is slower in the rank metric because of the algorithm that generates an error of a given rank. The verification of the commitment is slower in the rank metric because we have to compute the rank of a matrix rather than the Hamming weight of a binary vector. The generation of matrix A is slower than matrix G due to their different size. The generation of the proof of Knowledge of a Valid opening, Linear relations and Multiplicative Relations achieve similar timings for both variants. For the verification of the proofs, the performance of the rank metric is around 100 times better than the Hamming metric. This happens because of the large linear systems that have to be solved in the Hamming case.

6 Conclusion

We showed that quantum resistant zero-knowledge proof protocols can be built upon the Rank Syndrome Decoding problem in an efficient way. In particular, we implemented the building blocks needed for a zero-knowledge protocol to prove the relation among two committed values for any circuit. Our protocol is quasi-linear in the size of the circuit, has a soundness error of 2/3, and is quantum resistant. We hope this work to be a starting point to build even more efficient zero-knowledge protocols based on the RSD problem.

Knowledge of Valid Opening							
Jain et.al.				**This work**			
Routine	**Subroutine**		**Time [ms]**	**Routine**	**Subroutine**		**Time[ms]**
Proof gen.	Generate π		0.552	Proof gen.	Generate $\Pi_{P,Q}$		0.135
	Generate random vectors		negl.		Generate random vectors		negl.
	Comm. 0	t_0	0.032		Comm. 0	r_0	0.020
		$\mathsf{E}(t_\pi, t_0)$	0.400			$\mathsf{E}(\mathsf{r}_{P,Q}, \mathsf{r}_0)$	0.035
		$\mathsf{Com}(\mathsf{E}(t_\pi, t_0))$	0.200			$\mathsf{Com}(\mathsf{E}(\mathsf{r}_{P,Q}, \mathsf{r}_0))$	1.860
	Comm. 1	t_1	0.038		Comm. 1	r_1	0.044
		$\mathsf{E}(t_1)$	0.391			$\mathsf{E}(\mathsf{r}_1)$	0.019
		$\mathsf{Com}(\mathsf{E}(t_1))$	0.203			$\mathsf{Com}(\mathsf{E}(\mathsf{r}_1))$	1.809
	Comm. 2	t_2	0.040		Comm. 2	r_2	0.044
		$\mathsf{E}(t_2)$	0.396			$\mathsf{E}(\mathsf{r}_2)$	0.018
		$\mathsf{Com}(\mathsf{E}(t_2))$	0.197			$\mathsf{Com}(\mathsf{E}(\mathsf{r}_2))$	1.736
	Total		1.897		**Total**		5.585
Proof ver.	Verif. 0	$\mathsf{Ver}(\mathsf{c}_0, \mathsf{E}(t_\pi, t_0), \mathsf{s}_0))$	0.423	Proof ver.	Verif. 0	$\mathsf{Ver}(\mathsf{c}_0, \mathsf{E}(\mathsf{r}_{P,Q}, \mathsf{r}_0), \mathsf{s}_0))$	0.077
		$\mathsf{Ver}(\mathsf{c}_1, \mathsf{E}(t_1), \mathsf{s}_1)$	0.426			$\mathsf{Ver}(\mathsf{c}_1, \mathsf{E}(\mathsf{r}_1), \mathsf{s}_1)$	0.064
		$t_0 + \pi^{-1}(t_1) \in \mathsf{Img}(A)$	170.888			$\mathsf{r}_0 + \Pi_{r_0}^{-1}(\mathsf{r}_1) \in \mathsf{Img}(G)$	2.559
	Verif. 1	$\mathsf{Ver}(\mathsf{c}_0, \mathsf{E}(t_\pi, t_0), \mathsf{s}_0))$	0.424		Verif. 1	$\mathsf{Ver}(\mathsf{c}_0, \mathsf{E}(\mathsf{r}_{P,Q}, \mathsf{r}_0), \mathsf{s}_0))$	0.080
		$\mathsf{Ver}(\mathsf{c}_2, \mathsf{E}(t_2), \mathsf{s}_2)$	0.444			$\mathsf{Ver}(\mathsf{c}_2, \mathsf{E}(\mathsf{r}_2), \mathsf{s}_2)$	0.066
		$t_0 + \pi^{-1}(t_2) + y \in \mathsf{Img}(A)$	175.526			$\mathsf{r}_0 + \Pi_{r_0}^{-1}(\mathsf{r}_2) + \mathsf{y} \in \mathsf{Img}(G)$	2.47
	Verif. 2	$\mathsf{Ver}(\mathsf{c}_1, \mathsf{E}(t_1), \mathsf{s}_1)$	0.459		Verif. 2	$\mathsf{Ver}(\mathsf{c}_1, \mathsf{E}(\mathsf{r}_1), \mathsf{s}_1)$	0.064
		$\mathsf{Ver}(\mathsf{c}_2, \mathsf{E}(t_2), \mathsf{s}_2)$	0.446			$\mathsf{Ver}(\mathsf{c}_2, \mathsf{E}(\mathsf{r}_2), \mathsf{s}_2)$	0.064
		$\mathsf{w_H}(t_1 + t_2)$	0.001			$\mathsf{w_R}(\mathsf{r}_1 + \mathsf{r}_2)$	0.018
	Total		349.037		**Total**		5.462

Table 5. Knowledge of Valid Opening performance comparison.

References

1. Aguilar, C., Blazy, O., Deneuville, J.C., Gaborit, P., Zémor, G.: Efficient encryption from random quasi-cyclic codes. arXiv preprint arXiv:1612.05572 (2016)
2. Aguilar, C., Gaborit, P., Schrek, J.: A new zero-knowledge code based identification scheme with reduced communication. In: Information Theory Workshop (ITW), 2011 IEEE. pp. 648–652. IEEE (2011)
3. Aragon, N., Gaborit, P., Hauteville, A., Tillich, J.P.: Improvement of Generic Attacks on the Rank Syndrome Decoding Problem (Oct 2017), https://hal.archives-ouvertes.fr/hal-01618464, working paper or preprint
4. Bardet, M., Briaud, P., Bros, M., Gaborit, P., Neiger, V., Ruatta, O., Tillich, J.P.: An algebraic attack on rank metric code-based cryptosystems. arXiv preprint arXiv:1910.00810 (2019)
5. Bellare, M., Goldreich, O.: On defining proofs of knowledge. In: Annual International Cryptology Conference. pp. 390–420. Springer (1992)
6. Bellini, E., Caullery, F., Gaborit, P., Manzano, M., Mateu, V.: Improved veron identification and signature schemes in the rank metric. In: Information Theory (ISIT), 2019 IEEE International Symposium on. pp. 1872–1876. IEEE (2019). https://doi.org/10.1109/ISIT.2019.8849585
7. Bellini, E., Caullery, F., Hasikos, A., Manzano, M., Mateu, V.: Code-based signature schemes from identification protocols in the rank metric. In: International Conference on Cryptology and Network Security. pp. 277–298. Springer (2018)
8. Ben-Or, M., Goldreich, O., Goldwasser, S., Håstad, J., Kilian, J., Micali, S., Rogaway, P.: Everything provable is provable in zero-knowledge. In: Conference on the Theory and Application of Cryptography. pp. 37–56. Springer (1988)
9. Benhamouda, F., Krenn, S., Lyubashevsky, V., Pietrzak, K.: Efficient zero-knowledge proofs for commitments from learning with errors over rings. In: Computer Security - ESORICS 2015 - 20th European Symposium on Research in Computer Security. pp. 305–325 (2015)

Linear Relations					
Jain et.al.			**This work**		
Routine	**Subroutine**	**Time [ms]**	**Routine**	**Subroutine**	**Time[ms]**
Proof Gen.	Generate permutation P	3.039	Proof Gen.	Generate matrices P and Q	0.869
	Generate random vectors	0.030		Generate random vectors	0.568
	Generate commitments	4.502		Generate commitments	10.539
	Total	7.571		**Total**	11.976
Proof Ver.	Verification 1	524.682	Proof Ver.	Verification 1	4.780
	Verification 2	525.985		Verification 2	4.765
	Verification 3	2.344		Verification 3	0.512
	Total	1053.011		**Total**	10.057
Multiplicative Relations					
Jain et.al.			**This work**		
Routine	**Subroutine**	**Time [ms]**	**Routine**	**Subroutine**	**Time[ms]**
Proof Gen.	Generate permutation P	72.462	Proof Gen.	Generate matrices P and Q	28.432
	Generate random vectors	0.177		Generate random vectors	0.120
	Generate commitments	16.130		Generate commitments	47.430
	Total	88.769		**Total**	75.982
Proof Ver.	Verification 1	2634.96	Proof Ver.	Verification 1	22.402
	Verification 2	2580.93		Verification 2	22.760
	Verification 3	6.508		Verification 3	2.463
	Total	5222.398		**Total**	47.625

Table 6. Linear and multiplicative relations performance comparison.

10. Berlekamp, E., McEliece, R., Van Tilborg, H.: On the inherent intractability of certain coding problems (corresp.). IEEE Transactions on Information Theory **24**(3), 384–386 (1978)
11. Cayrel, P.L., Lindner, R., Rückert, M., Silva, R.: Improved zero-knowledge identification with lattices. Tatra Mountains Mathematical Publications **53**(1), 33–63 (2012)
12. Cayrel, P.L., Véron, P., Alaoui, S.M.E.Y.: A zero-knowledge identification scheme based on the q-ary syndrome decoding problem. In: International Workshop on Selected Areas in Cryptography. pp. 171–186 (2010)
13. Chabaud, F., Stern, J.: 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. pp. 368–381 (1996)
14. Fiat, A., Shamir, A.: How to prove yourself: Practical solutions to identification and signature problems. In: Conference on the Theory and Application of Cryptographic Techniques. pp. 186–194 (1986)
15. Gabidulin, E.M.: Theory of codes with maximum rank distance. Problemy Peredachi Informatsii **21**(1), 3–16 (1985)
16. Gaborit, P., Girault, M.: Lightweight code-based identification and signature. In: 2007 IEEE International Symposium on Information Theory. pp. 191–195. IEEE (2007)
17. Gaborit, P., Ruatta, O., Schrek, J.: On the complexity of the rank syndrome decoding problem. IEEE Transactions on Information Theory **62**(2), 1006–1019 (2016)
18. Gaborit, P., Schrek, J., Zémor, G.: Full cryptanalysis of the chen identification protocol. In: International Workshop on Post-Quantum Cryptography. pp. 35–50 (2011)

19. Goldreich, O., Micali, S., Wigderson, A.: Proofs that yield nothing but their validity or all languages in np have zero-knowledge proof systems. Journal of the ACM (JACM) **38**(3), 690–728 (1991)
20. Goldwasser, S., Micali, S., Rackoff, C.: The knowledge complexity of interactive proof systems. SIAM Journal on computing **18**(1), 186–208 (1989)
21. Grover, L.K.: A fast quantum mechanical algorithm for database search. In: Proceedings of the Twenty-Eighth Annual ACM Symposium on the Theory of Computing, Philadelphia, Pennsylvania, USA. pp. 212–219. ACM (1996)
22. Impagliazzo, R., Yung, M.: Direct minimum-knowledge computations. In: Conference on the Theory and Application of Cryptographic Techniques. pp. 40–51. Springer (1987)
23. Jain, A., Krenn, S., Pietrzak, K., Tentes, A.: Commitments and efficient zero-knowledge proofs from learning parity with noise. In: International Conference on the Theory and Application of Cryptology and Information Security. pp. 663–680 (2012)
24. Kawachi, A., Tanaka, K., Xagawa, K.: Concurrently secure identification schemes based on the worst-case hardness of lattice problems. In: International Conference on the Theory and Application of Cryptology and Information Security. pp. 372–389. Springer (2008)
25. Lau, T.S.C., Tan, C.H., Prabowo, T.F.: Key recovery attacks on some rank metric code-based signatures. In: IMA International Conference on Cryptography and Coding. pp. 215–235. Springer (2019)
26. Ling, S., Nguyen, K., Stehlé, D., Wang, H.: Improved zero-knowledge proofs of knowledge for the isis problem, and applications. In: International Workshop on Public Key Cryptography. pp. 107–124. Springer (2013)
27. Loidreau, P.: Properties of codes in rank metric. arXiv preprint cs/0610057 (2006)
28. Lyubashevsky, V.: Lattice-based identification schemes secure under active attacks. In: International Workshop on Public Key Cryptography. pp. 162–179. Springer (2008)
29. Martínez, R., Morillo, P.: RLWE-based zero-knowledge proofs for linear and multiplicative relations. In: IMA International Conference on Cryptography and Coding. pp. 252–277. Springer (2019)
30. May, A., Ozerov, I.: On computing nearest neighbors with applications to decoding of binary linear codes. In: Annual International Conference on the Theory and Applications of Cryptographic Techniques. pp. 203–228 (2015)
31. Micciancio, D., Vadhan, S.P.: Statistical zero-knowledge proofs with efficient provers: Lattice problems and more. In: Annual International Cryptology Conference. pp. 282–298. Springer (2003)
32. Ourivski, A.V., Johansson, T.: New technique for decoding codes in the rank metric and its cryptography applications. Problems of Information Transmission **38**(3), 237–246 (2002)
33. Shor, P.W.: Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. SIAM Journal on Computing **26**(5), 1484–1509 (1997)
34. Smart, N.: Letter to NIST on standardizing new cryptographic standards (2016), available at https://zkp.science/docs/Letter-to-NIST-20160613-Advanced-Crypto.pdf
35. Stern, J.: A new identification scheme based on syndrome decoding. In: Annual International Cryptology Conference. pp. 13–21 (1993)
36. Stern, J.: A new paradigm for public key identification. IEEE Transactions on Information Theory **42**(6), 1757–1768 (1996)

37. Various: Zero knowledge proof standardization: An open industry/academic initiative (Last visited 20-01-2020), available at https://zkproof.org/index.html
38. Véron, P.: Improved identification schemes based on error-correcting codes. Applicable Algebra in Engineering, Communication and Computing **8**(1), 57–69 (1997)
39. Wachter-Zeh, A.: Decoding of block and convolutional codes in rank metric. Ph.D. thesis, Universität Ulm (2013)
40. Xie, X., Xue, R., Wang, M.: Zero knowledge proofs from ring-lwe. In: International Conference on Cryptology and Network Security. pp. 57–73. Springer (2013)

Prover P **Verifier** V

$m'_1, m'_2, m'_3 \leftarrow\$ \mathbb{F}_2^{4\mu}$ such that
$m'_3 = m'_1 \circ m'_2$
$\forall (a,b) \in (\mathbb{F}_q)^2, \#[(m'_1[j], m'_2[j]) = (a,b)] = \mu$
$R \leftarrow\$ \mathcal{M}_{\mu,4\mu}(\mathbb{F}_q)$ s.t. $R \cdot m'_i = \mathsf{m}_i$ **and** $\mathsf{Rank}(R) = \mu$
for $i = 1,2,3$
for $j = 1,2,3,4$
$s'^j_i \leftarrow\$ \mathbb{F}_q^{\pi}$
$e'^j_i \leftarrow\$ (\mathbb{F}_{2^m}^n)$ s.t. $\mathsf{w_R}(e'^j_i) = \rho$
$y'^j_i = (s'^j_i || m'^j_i) \cdot G + e'^j_i$
$Q'^j_i \leftarrow\$ \mathcal{M}^*_{m,m}(\mathbb{F}_2)$
$P'^j_i \leftarrow\$ \mathcal{M}^*_{n,n}(\mathbb{F}_2)$
$v'^j_i \leftarrow\$ \mathbb{F}_2^{\mu}$
$u'^j_i \leftarrow\$ \mathbb{F}_2^{\pi}$
$f'^j_i \leftarrow\$ \mathbb{F}_{2^m}^n$
$\mathsf{c}'^j_{i,0}, \mathsf{s}'^j_{i,0} \leftarrow \mathsf{Com}(\mathsf{E}(P'^j_i, Q'^j_i, (u'^j_i || v'^j_i) \cdot G + f'^j_i)$
$\mathsf{c}'^j_{i,1}, \mathsf{s}'^j_{i,1} \leftarrow \mathsf{Com}(\mathsf{E}(\Pi_{P'^j_i, Q'^j_i}(f'^j_i)))$
$\mathsf{c}'^j_{i,2}, \mathsf{s}'^j_{i,2} \leftarrow \mathsf{Com}(\mathsf{E}(\Pi_{P'^j_i, Q'^j_i}(f'^j_i + e'^j_i)))$
endfor
$v_i = \sum_{j=1}^{4} R^j \cdot v'^j_i$
$Q_i, \leftarrow\$ \mathcal{M}^*_{m,m}(\mathbb{F}_2)$
$P_i, \leftarrow\$ \mathcal{M}^*_{n,n}(\mathbb{F}_2)$
$u_i, \leftarrow\$ \mathbb{F}_2^{\pi}$
$f_i, \leftarrow\$ \mathbb{F}_{2^m}^n$
$\mathsf{c}_{i,0}, \mathsf{s}_{i,0} \leftarrow \mathsf{Com}(\mathsf{E}(P_i, Q_i, (u_i || v_i) \cdot G + f_i))$
$\mathsf{c}_{i,1}, \mathsf{s}_{i,1} \leftarrow \mathsf{Com}(\mathsf{E}(\Pi_{P_i,Q_i}(f_i)))$
$\mathsf{c}_{i,2}, \mathsf{s}_{i,2} \leftarrow \mathsf{Com}(\mathsf{E}(\Pi_{P_i,Q_i}(f_i + \mathsf{e}_i)))$
endfor
$\mathsf{c}, \mathsf{s} \leftarrow \mathsf{Com}(\mathsf{E}(y'_1, y'_2, y'_3))$
$\mathsf{c}_R, \mathsf{s}_R \leftarrow \mathsf{Com}(\mathsf{E}(R))$

$\xrightarrow{\mathsf{c}, \mathsf{c}_R, \mathsf{c}_{i,0}, \mathsf{c}_{i,1}, \mathsf{c}_{i,2} \;\; \mathsf{c}'_{i,0}, \mathsf{c}'_{i,1}, \mathsf{c}'_{i,2}}$

Fig. 3: Commitment step of the Σ-protocol proving multiplicative relations in the rank metric.

Prover P		**Verifier** V
	$\xleftarrow{\mathsf{ch}}$	$\mathsf{ch} \leftarrow_\$ \{0,1,2\}$

for $i = 1,2,3$
  for $j = 1,2,3,4$
    $\mathsf{r}'^j_{i,P',Q'} = (P'^j_i, Q'^j_i)$
    $\mathsf{r}'^j_{i,0} = (u'^j_i || v'^j_i) \cdot G + f'^j_i$
    $\mathsf{r}'^j_{i,1} = \Pi_{P'^j_i, Q'^j_i}(f'^j_i)$
    $\mathsf{r}'^j_{i,2} = \Pi_{P'^j_i, Q'^j_i}(f'^j_i + e'^j_i)$
  endfor
  $\mathsf{r}_{i,P,Q} = (P_i, Q_i)$
  $\mathsf{r}_{i,0} = (u_i || v_i) \cdot G + f_i$
  $\mathsf{r}_{i,1} = \Pi_{P_i,Q_i}(f_i)$
  $\mathsf{r}_{i,2} = \Pi_{P_i,Q_i}(f_i + \mathsf{e}_i)$
endfor

---

if $\mathsf{ch} = 0$ $\quad \xrightarrow{R,\ \mathsf{r}_{i,P,Q},\ \mathsf{r}'^j_{i,P',Q'},\ \mathsf{r}'^j_{i,0},\ \mathsf{r}_{i,0},\ \mathsf{r}'^j_{i,1},\ \mathsf{r}_{i,1},\ \mathsf{s}_R,\ \mathsf{s}'^j_{i,0},\ \mathsf{s}_{i,0},\ \mathsf{s}'^j_{i,1},\ \mathsf{s}_{i,1}}$

if
$$\begin{cases} \mathsf{Ver}(\mathsf{c}_R, \mathsf{E}(R), \mathsf{s}_R) = \mathsf{true} \textbf{ and} \\ \mathsf{Ver}(\mathsf{c}_{i,0}, \mathsf{E}(\mathsf{r}_{i,0}), \mathsf{s}_{i,0}) = \mathsf{true} \textbf{ and} \\ \mathsf{Ver}(\mathsf{c}'^j_{i,0}, \mathsf{E}(\mathsf{r}'^j_{i,0}), \mathsf{s}'^j_{i,0}) = \mathsf{true} \textbf{ and} \\ \mathsf{Ver}(\mathsf{c}_{i,1}, \mathsf{E}(\mathsf{r}_{i,1}), \mathsf{s}_{i,1}) = \mathsf{true} \textbf{ and} \\ \mathsf{Ver}(\mathsf{c}'^j_{i,1}, \mathsf{E}(\mathsf{r}'^j_{i,1}), \mathsf{s}'^j_{i,1}) = \mathsf{true} \textbf{ and} \\ \exists a_i, b_1 \mid \mathsf{r}_{i,0} + \Pi^{-1}_{r_{i,P,Q}}(\mathsf{r}_{i,1}) = (a_i || b_i) \cdot G \textbf{ and} \\ \exists a'^j_i, b'^j_i \mid \mathsf{r}'^j_{i,0} + \Pi^{-1}_{r'^j_{i,P',Q'}}(\mathsf{r}'^j_{i,1}) = (a'^j_i || b'^j_i) \cdot G \textbf{ and} \\ b_i = \sum_{j=1}^4 R^j . b'^j_i \textbf{ and} \\ \mathsf{r}_{i,P,Q}, r'^j_{i,P',Q'} \in \mathcal{M}^*_{m,m}(\mathbb{F}_q) \times \mathcal{M}^*_{n,n}(\mathbb{F}_q) \end{cases}$$
return true

---

if $\mathsf{ch} = 1$ $\quad \xrightarrow{y'_1,\ y'_2,\ y'_3,\ R,\ \mathsf{r}_{i,P,Q},\ \mathsf{r}'^j_{i,P',Q'},\ \mathsf{r}'^j_{i,0},\ \mathsf{r}_{i,0},\ \mathsf{r}'^j_{i,2},\ \mathsf{r}_{i,2},\ \mathsf{s}_R,\ \mathsf{s}'^j_{i,0},\ \mathsf{s}_{i,0},\ \mathsf{s}'^j_{i,2},\ \mathsf{s}_{i,2}}$

if
$$\begin{cases} \mathsf{Ver}(\mathsf{c}_R, \mathsf{E}(R), \mathsf{s}_R) = \mathsf{true} \textbf{ and} \\ \mathsf{Ver}(\mathsf{c}_{i,0}, \mathsf{E}(\mathsf{r}_{i,0}), \mathsf{s}_{i,0}) = \mathsf{true} \textbf{ and} \\ \mathsf{Ver}(\mathsf{c}'^j_{i,0}, \mathsf{E}(\mathsf{r}'^j_{i,0}), \mathsf{s}'^j_{i,0}) = \mathsf{true} \textbf{ and} \\ \mathsf{Ver}(\mathsf{c}_{i,2}, \mathsf{E}(\mathsf{r}_{i,2}), \mathsf{s}_{i,2}) = \mathsf{true} \textbf{ and} \\ \mathsf{Ver}(\mathsf{c}'^j_{i,2}, \mathsf{E}(\mathsf{r}'^j_{i,2}), \mathsf{s}'^j_{i,2}) = \mathsf{true} \textbf{ and} \\ \mathsf{Rank}(R) = \mu \textbf{ and } \mathsf{w_H}(R_i) \leq 1 \textbf{ and} \\ \exists c_i, d_1 \mid \mathsf{r}_{i,0} + \Pi^{-1}_{r_{i,P,Q}}(\mathsf{r}_{i,2}) + \mathsf{y}_i == (c_i || d_i) \cdot G \textbf{ and} \\ \exists c'^j_i, d'^j_i \mid \mathsf{r}'^j_{i,0} + \Pi^{-1}_{r'^j_{i,P',Q'}}(\mathsf{r}'^j_{i,2}) + y'^j_i == (c'^j_i || d'^j_i) \cdot G \textbf{ and} \\ d_i == \sum_{j=1}^4 R^j . d'^j_i \end{cases}$$
return true

---

if $\mathsf{ch} = 2$ $\quad \xrightarrow{y'_1,\ y'_2,\ y'_3,\ s'_1,\ s'_2,\ s'_3,\ m'_1,\ m'_2,\ m'_3,\ e'_1,\ e'_2,\ e'_3,\ \mathsf{r}'^j_{i,1},\ \mathsf{r}_{i,1},\ \mathsf{r}'^j_{i,2},\ \mathsf{r}_{i,2},\ \mathsf{s}'^j_{i,1},\ \mathsf{s}_{i,1},\ \mathsf{s}'^j_{i,2},\ \mathsf{s}_{i,2}}$

if
$$\begin{cases} \mathsf{Ver}(\mathsf{c}_{i,1}, \mathsf{E}(\mathsf{r}_{i,1}), \mathsf{s}_{i,1}) = \mathsf{true} \textbf{ and} \\ \mathsf{Ver}(\mathsf{c}'^j_{i,1}, \mathsf{E}(\mathsf{r}'^j_{i,1}), \mathsf{s}'^j_{i,1}) = \mathsf{true} \textbf{ and} \\ \mathsf{Ver}(\mathsf{c}_{i,2}, \mathsf{E}(\mathsf{r}_{i,2}), \mathsf{s}_{i,2}) = \mathsf{true} \textbf{ and} \\ \mathsf{Ver}(\mathsf{c}'^j_{i,2}, \mathsf{E}(\mathsf{r}'^j_{i,2}), \mathsf{s}'^j_{i,2}) = \mathsf{true} \textbf{ and} \\ \mathsf{w_R}(\mathsf{r}'^j_{i,1} + \mathsf{r}'^j_{i,2}) = \mathsf{w_R}(\mathsf{r}_{i,1} + \mathsf{r}_{i,2}) = \rho \textbf{ and} \\ \mathsf{w_R}(e'^j_i) = \rho \textbf{ and} \\ y'^j_i = (s'^j_i || m'^j_i) \cdot G + e'^j_i \textbf{ and} \\ m'^j_1 \circ m'^j_2 = m'^j_3 \end{cases}$$
return true

Fig. 4: Challenge and response steps of the Σ-protocol proving multiplicative relations in the rank metric.

A Sigma Protocol

We give the definition of Σ-protocol, which is the basis of the protocols we present. This definition might help understanding the security proof in section B.

Definition 4 (Σ-protocol). *Let* (P, V) *be a two-party protocol, where* V *is PPT, and let* R *be a binary relation. Then* (P, V) *is called a* Σ*-protocol for* R *with challenge set* C*, public input* y *and private input* w*, if and only if it satisfies the following conditions:*

- 3-move form*: The protocol is of the following form:*
  - P *computes a commitment* t *and sends it to* V*.*
  - V *draws a challenge* $c \leftarrow_{\$} C$ *and sends it to* P*.*
  - P *sends a response* s *to* V*.*

  Depending on the protocol transcript (t, c, s)*,* V *accepts or rejects the proof. The protocol transcript* (t, c, s) *is called* accepting*, if* V *accepts the protocol run.*
- Completeness*:* V *accepts whenever* $(y, w) \in R$*.*
- Special soundness*: There exists a PPT algorithm* E *(the* knowledge extractor*) which takes a set* $\{(t, c, s_c)$ *s.t.* $c \in C\}$ *of accepting transcripts with the same commitment as inputs, and outputs* w' *such that* $(y, w')R$*.*
- Special honest-verifier zero-knowledge*: There exists a PPT algorithm* S *(the* simulator*) taking* y *and* $c \in C$ *as inputs, and which outputs triples* (t, c, s) *whose distribution is (computationally) indistinguishable from accepting protocol transcripts generated by real protocol runs.*

B Proof of Theorem 3

Proof. We need to prove that the protocol is 3-move, complete, sound and zero-knowledge.

- *3-move*: the protocol is 3-move by design.
- *Completeness*: it is easy to see that the if the protocol is honestly run by a prover, then it always returns true.
  - If $\mathsf{ch} = 0$ then $\mathsf{r}_0 + \Pi_{r_0}^{-1}(\mathsf{r}_1) = v \cdot G + f + \Pi_{P,Q}^{-1}(\Pi_{P,Q}(f)) = v \cdot G \in \mathsf{Img}(G)$ and P, Q are two binary matrices of size $m \times m$ and $n \times n$ respectively.
  - If $\mathsf{ch} = 1$ then $\mathsf{r}_0 + \Pi_{r_0}^{-1}(\mathsf{r}_2) + \mathsf{y} = v \cdot G + f + \Pi_{P,Q}^{-1}(\Pi_{P,Q}(f + \mathsf{e})) + \mathsf{x} \cdot G + \mathsf{e} = (v + \mathsf{x}) \cdot G \in \mathsf{Img}(G)$.
  - If $\mathsf{ch} = 2$ then $\mathsf{w_R}(\mathsf{r}_1 + \mathsf{r}_2) = \mathsf{w_R}(\Pi_{P,Q}(f) + \Pi_{P,Q}(f + \mathsf{e})) = \mathsf{w_R}(\Pi_{P,Q}(f + f + \mathsf{e})) = \mathsf{w_R}(\mathsf{e}) = \rho$.
- *Special soundness*: we first assume that the values $\mathsf{c}_0, \mathsf{c}_1, \mathsf{c}_2$ and openings for all challenges $\mathsf{ch} \in \{0, 1, 2\}$ have been fixed in such a way that that V accepts on all of them. Since the underlying commitment scheme Com is perfectly binding and the compression function E collision resistant, then the openings to identical commitments have to be identical when different challenges are given, or a collision for E should be found. We have that $\Pi_{P,Q}^{-1}(\mathsf{r}_1 + \mathsf{r}_2) + \mathsf{y} \in$

$\mathsf{Img}(G)$ thanks to the verification equations for $\mathsf{ch} = 0$ and $\mathsf{ch} = 1$, and thus that $\mathsf{y} = \mathsf{x}' \cdot G + \Pi^{-1}_{\mathsf{r}_{P,Q}}(\mathsf{r}_1 + \mathsf{r}_2)$, where $\mathsf{x}' = (\mathsf{s}' \| \mathsf{m}')$ can be easily computed. Now, a valid witness of (G, y) is given by $(\mathsf{s}', \mathsf{m}', \Pi^{-1}_{\mathsf{r}_{P,Q}}(\mathsf{r}_1 + \mathsf{r}_2))$, since $\mathsf{w_R}(\mathsf{r}_1 + \mathsf{r}_2) = \rho$. It is important to highlight that the input of the commitment scheme is the result of a collision resistant function, therefore, the probability for the above mentioned equations to not be correct is negligible, as it is the probability of a collision in a collision resistant compression function.

– *Honest Verifier Zero-knowledge*: we need to prove that there exist an efficient simulator Sim, which, for each challenge $\mathsf{ch} \in \{0, 1, 2\}$, outputs an accepting protocol transcript that is computationally indistinguishable from a real protocol transcript performed by an honest prover for the given challenge ch. The simulator can be described as follows:
  - $\mathsf{ch} = 0$: Sim computes $\mathsf{c}_0, \mathsf{c}_1$ as in the protocol, and c_2 as a commitment to 0. It is straightforward that the distribution of $\mathsf{c}_0, \mathsf{c}_1, \mathsf{r}_{P,Q}, \mathsf{r}_0, \mathsf{r}_1$ is identical to the one of a real transcript. Furthermore, the fact that the commitment scheme Com is computationally hiding implies that the distribution of c_2 is computationally indistinguishable from the real protocol runs.
  - $\mathsf{ch} = 1$: Sim selects uniformly at random the values $Q \leftarrow_\$ \mathcal{M}^*_{m,m}(\mathbb{F}_q)$, $P \leftarrow_\$ \mathcal{M}^*_{n,n}(\mathbb{F}_q)$, $b \leftarrow_\$ (\mathbb{F}_{q^m})^k$, $a \leftarrow_\$ (\mathbb{F}_{q^m})^n$. Then, computes the commitments $\mathsf{c}_0 = \mathsf{Com}(\mathsf{E}(P, Q, b \cdot G + \mathsf{y} + a))$ and $\mathsf{c}_2 = \mathsf{Com}(\mathsf{E}(\Pi_{P,Q}(a)))$. The value of c_1 is computed as commitment to 0. The openings of $\mathsf{c}_0, \mathsf{c}_2$ are verified correctly by the verifier. The distribution of the openings is correct because of the perfect uniform distribution of r_2 in the real protocol run and $\Pi_{P,Q}(a)$ in the simulated run in $\mathbb{F}^n_{q^m}$, and of the permutations in the set of permutations. Regarding the opening of c_0, notice that in the real protocol run, it holds $\mathsf{r}_{P,Q} = v \cdot G + f$, where v is uniformly at random, and $f = \Pi^{-1}_{P,Q}(\mathsf{r}_2 + \mathsf{e})$. In the simulated transcript the content of c_0 is $b \cdot G + \mathsf{y} + a = (b + \mathsf{x}) \cdot G + (a + \mathsf{e})$. The distributions of c_0 and c_2 and their openings are perfectly simulated, since v and $b + \mathsf{x}$ are both uniformly random, and the terms f and $a + \mathsf{e}$ are uniquely determined by the contents of c_0 and c_2. Finally, the distribution of c_1 is computationally indistinguishable by the assumed hiding property of Com.
  - $\mathsf{ch} = 2$: Sim selects uniformly at random $a \leftarrow_\$ (\mathbb{F}_{q^m})^n$, $b \leftarrow_\$ (\mathbb{F}_{q^m})^n$ such that $\mathsf{w_R}(b) = \rho$. It computes c_0 as commitment to 0. $\mathsf{c}_1 = \mathsf{Com}(\mathsf{E}(a))$ $\mathsf{c}_2 = \mathsf{Com}(\mathsf{E}(a+b))$. As in the case of $\mathsf{ch} = 1$, the binding property of Com implies that the distributions of c_0 is computationally indistinguishable from real protocol runs. Furthermore, the behavior of an honest prover can be perfectly simulated by c_1 and c_2 and their openings.

□