Improved Veron Identification and Signature Schemes in the Rank Metric

Emanuele Bellini\textsuperscript{1}, Florian Caullery\textsuperscript{1}, Philippe Gaborit\textsuperscript{2}, Marc Manzano\textsuperscript{1}, and Victor Mateu\textsuperscript{1}

\textsuperscript{1} Darkmatter LLC, Abu Dhabi, UAE
\textsuperscript{2} Université de Limoges, France

Abstract. It is notably challenging to design an efficient and secure signature scheme based on error-correcting codes. An approach to build such signature schemes is to derive it from an identification protocol through the Fiat-Shamir transform. All such protocols based on codes must be run several rounds, since each run of the protocol allows a cheating probability of either $2/3$ or $1/2$. The resulting signature size is proportional to the number of rounds, thus making the $1/2$ cheating probability version more attractive. We present a signature scheme based on double circulant codes in the rank metric, derived from an identification protocol with cheating probability of $2/3$. We reduced this probability to $1/2$ to obtain the smallest signature among signature schemes based on the Fiat-Shamir paradigm, around 22 KBytes for 128 bit security level. Furthermore, among all code-based signature schemes, our proposal has the lowest value of signature plus public key size, and the smallest secret and public key sizes. We provide a security proof in the Random Oracle Model, implementation performances, and a comparison with the parameters of the most important code-based signature schemes.

Keywords: code-based cryptography · signature scheme · identification protocol · Fiat-Shamir transform · rank metric

1 Introduction

Due to the early stage of post-quantum algorithm research, it is of paramount importance to provide the full range of quantum secure cryptographic primitives (signatures, key exchange, etc.) for all the main mathematical problems cryptography relies on. This way, it will be easier to switch from one scheme to the other in the case one of the problems turns out to be insecure in the quantum model. Given that it is the oldest quantum resistant family and, hence, the most thoroughly studied among all the contenders, code-based cryptography is a strong candidate in the NIST competition to standardize quantum resistant cryptographic algorithms [32].

This work focuses on code-based cryptography digital signature schemes. Designing such schemes efficiently has been a grueling challenge and mainly three different approaches have been followed, with very little success. Hash-and-sign was introduced in pioneering work of Courtois, Finiasz, and Sendrier
and is probably the most popular approach of the three. It is based on
the existence of a trapdoor which allows fast decoding, obtained by hiding a
structured code into a random linear code. Different choices of the underlying
code lead to different instantiations of the scheme. All hash-and-sign schemes
yield to small signatures (few thousands bits), but large public keys (order of
MBytes), in some cases even non-practical ones for 128 bit security level and
above. Furthermore, almost all these schemes have been attacked. The other two
approaches avoid the use of trapdoors. The first is usually referred to as the KKS
(Kabatianski-Krouk-Smeets) signature scheme [25], who later evolved in the
BMS (Barreto-Misoczki-Simplicio) scheme [8]. Both of them can be instantiated
on top of general linear codes. KKS and BMS have a good balance between public
key (few tens of thousands of bits) and signature size (few thousands of bits),
but they can only be considered one-time signature schemes. The third approach
uses the Fiat-Shamir transform to turn a zero-knowledge identification scheme
into a signature scheme, as initially proposed by Stern [36] in 1993. The main
drawback of such scheme is the large signature size. Many researchers followed
Stern approach, trying to improve either the signature or the key size of the
scheme.

In this manuscript, we provide a variation of a signature scheme based on
Stern approach, obtaining the smallest signature (\(\text{sgn}\)), secret and public key (\(\text{pk}\))
sizes in the literature. Compared to other approaches used to build code-based
signature schemes, we also have the smallest \(|\text{sgn}| + |\text{pk}|\) value. We derive such
signature from a 5-pass identification protocol with cheating probability \(1/2\). We
provide a security proof in the Random Oracle Model, a detailed pseudo-code,
implementation performances, set of parameters for 80, 128, 192, and 256 bit of
classical security, and a comparison with the parameters of the most important
code-based signature schemes.

The paper is organized as follows: Sect. 2 provides an overview of previous
works and the ideas behind the scheme. In Sect. 3 we provide the notions that are
needed to understand the contribution. Sect. 4 presents our new identification
protocol. Sect. 5 sets the parameters of our signature schemes. Sect. 7 argues
about the theoretical complexity of the key generation, signature and verifica-
tion algorithms, providing also implementation details and performances. Sect. 6
shows a comparison of the parameters of our proposal and other well-known
code-based signature schemes, and Sect. 8 draws the conclusions.

2 Main idea

Commonly, cryptographic signature schemes whose security relies on the diffi-
culty of decoding a linear code are built by converting a 3 or a 5-pass iden-
tification protocol into a signature scheme via the Fiat-Shamir transform or a
generalization of it. The first to propose such paradigm was Stern [36]. In this
work, Stern exhibits a 3-pass identification protocol whose security is based on
the difficulty of decoding a random linear code and finding a hash collision, and
in which a cheater can correctly identify with a probability of \(2/3\). For this last
reason, the protocol should be run an appropriate number of rounds which depends on the security level the scheme needs to reach. Since the corresponding signature is proportional to the number of rounds, this means that this type of approach yields to large signatures, of the order of hundreds of KBytes. The basic idea of the protocol is that, given the parity-check matrix $H$ of a linear code as a public parameter, a random vector $e$ of weight $w$, and a public key $s = eH$, the prover needs to prove the knowledge of two properties, namely the fact that the vector $e$ is generating the syndrome $s$, and that $e$ has Hamming weight $w$. Adding a random commitment there are always two possibilities for cheating among the three cases. In the same work, Stern shows how to reduce the cheating probability to $1/2$, by splitting the challenge step into two challenges, the second of which adds a variation on $e$, forcing the protocol to perform 5 passes. Precisely, $e$ was chosen as a codeword of a Reed-Muller code. Such trick allows to almost halve the corresponding signature size, even though, with this particular solution, there is a loss in efficiency. Stern signature schemes presents very small secret keys (less than a thousand bits) and medium size public keys (one hundred thousand bits).

Subsequent works aim at improving either key or signature sizes, by (1) choosing a structured code rather than a random linear code, (2) changing the variation performed on $e$, (3) working with the dual cryptosystem, or (4) working in a different metric. In [39], Veron presents the dual of the 3-pass Stern proposal, i.e. it uses the generator matrix $G$ of a code, instead of the parity-check matrix as a public parameter, and uses a pair $(x, e)$ as a secret key, and a codeword $y = xG + e$ as a public key. This allows to send less data on average during the response step, implying slightly shorter signatures. Later, Cayrel-Veron-El Yousfi Alaoui (CVE) [14] presented a 5-pass identification protocol with cheating probability of 1/2, using codes over $F_{2^m}$, rather than $F_2$ as done by Stern and Veron, and a scalar multiplication as the variation of $e$. In [16] it is shown how to extend the Fiat-Shamir transform to a $n$-pass protocol (with $n$ odd).

In 2011, Gaborit, Schrek and Zémor [22] presented the rank metric version of the Stern identification protocol, decreasing significantly key and signature sizes, due to the fact that rank metric decoding has quadratic exponential complexity, while Hamming metric decoding is linear exponential. The same year, Aguilar, Gaborit and Schrek [5], used double circulant codes in the Hamming metric to reduce the key size of the Veron scheme, and presented a 5-pass version of it, with cheating probability close to 1/2, performing a variation of $e$ with a circulant rotation of its two halves in the second challenge step. Furthermore, they introduced a compression technique to reduce the signature size. Recently, in [9], a rank metric version of Veron and CVE has been presented, though lacking a security proof. We are not aware of any attack to any of the Fiat-Shamir paradigm constructions, which probably have not received much attention from the cryptographic community yet.

In this work, we present a rank metric version of the 5-pass Veron double circulant signature scheme of [5], with a new variation performed on $e$, which allows us to reach a cheating probability much closer to 1/2. Precisely, we adopt
a random linear combination of all possible rotations of \( e \) in the second challenge step. We also present a compressed version of the scheme, which achieves signature sizes that are comparable to the one of post-quantum hash-based signature schemes.

3 Preliminaries

In this section we provide the essential definition of the objects that are used in our protocol.

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 \( G \in \mathbb{F}_q^{k \times n} \) 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 \( H \in \mathbb{F}_q^{(n-k) \times n} \) is called a parity-check matrix of \( C \) if \( C = \{ x \in \mathbb{F}_q^n : H \cdot x^T = 0 \} \). Our schemes will use a special type of linear codes, called double circulant codes, which are a special case of quasi-cyclic (or circulant) codes (see e.g. [31]).

**Definition 1 (Double Circulant Codes).** Let \( n = 2k \) for an integer \( k \). Consider a vector \( x = (x_1,x_2) \) of \((\mathbb{F}_q)^n\) as a pair of two blocks of length \( k \). An \([n,k]\) linear code \( C \) is Double Circulant (DC) if, for any \( c = (c_1,c_2) \in C \), the vector obtained after applying a simultaneous circular shift to both blocks \( c_1,c_2 \) is also a codeword. More formally, by considering each block \( c_1,c_2 \) as a polynomial in \( R = \mathbb{F}_q[X]/(X^n - 1) \), the code \( C \) is DC if for any \( c = (c_1,c_2) \in C \) it holds that \((X \cdot c_1,X \cdot c_2) \in C \).

A systematic double circulant \([n,k]\) code is a double circulant code with a parity-check matrix of the form \( H = [I_k | A] \), where \( I_k \) is the identity matrix of size \( k \), and \( A \) is a \( k \times k \) circulant matrix.

In this paper we work with codes in the rank metric. Given a fixed basis \( b = \{ b_1, \ldots, b_m \} \) 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 \( b \) in a column \((a_{1,i}, \ldots, a_{m,i})^T \). where \( a_i = \sum_{j=1}^m a_{j,i} b_j, i = 1, \ldots, n \). We define the rank of a vector as the rank of its matrix representation, with respect to \( b \). We denote the previous matrix representation as \( \phi_b(a) \), and by \( \phi_b^{-1} \) the inverse map. In what follows, we will omit \( b \) 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 [22]).

**Definition 2.** Let \( Q \in M_{m,m}(\mathbb{F}_q) \) be a \( q \)-ary matrix of size \( m \times m \), \( P \in M_{n,n}(\mathbb{F}_q) \) be a \( q \)-ary matrix of size \( n \times n \), and \( v \in (\mathbb{F}_q^m)^n \). We define the function

\[ f(Q, P, v) = Q \cdot P \cdot v. \]
Π_{P,Q} such that Π_{P,Q}(v) = φ^{-1}(Q \cdot φ(v) \cdot P), i.e.

\[ \Pi_{P,Q} : (\mathbb{F}_q^n)^n \to (\mathbb{F}_q^m)^n \]
\[ (v_1, \ldots, v_n) \mapsto (\pi_1, \ldots, \pi_n) \]

where for \( h = 1, \ldots, n \),
\[ \pi_h := \beta_1 \sum_{i=1}^{m} \sum_{j=1}^{n} Q_{1,i}v_{i,j}P_{j,h} + \ldots + \beta_m \sum_{i=1}^{m} \sum_{j=1}^{n} Q_{m,i}v_{i,j}P_{j,h} \]

It is proved in [22] that the following properties hold for \( \Pi_{P,Q} \).

- For any \( x, y \in (\mathbb{F}_q^n)^n \), \( P \in M_{n,n}(\mathbb{F}_q) \) and \( Q \in M_{m,m}(\mathbb{F}_q) \) then:
  - (rank preservation) \( w_R(\Pi_{P,Q}(x)) = w_R(x) \);
  - (linearity) \( a\Pi_{P,Q}(x) + b\Pi_{P,Q}(y) = \Pi_{P,Q}(ax + by) \).
- For any \( x, y \in (\mathbb{F}_q^n)^n \) such that \( w_R(x) = w_R(y) \), it is possible to find \( P \in M_{n,n}(\mathbb{F}_q) \) and \( Q \in M_{m,m}(\mathbb{F}_q) \) such that \( x = \Pi_{P,Q}(y) \).

Both in the Hamming and in the rank metric, random codes over \( \mathbb{F}_q \) asymptotically achieve the Gilbert-Varshamov bound [19]. Furthermore, they have close to optimal correction capability [27].

We now define the problems upon which the security of the schemes we present is based.

**Definition 3 (RSD Distribution).** Given the positive integers \( n, k \), and \( r \), the RSD\((n, k, r)\) Distribution chooses \( H \leftarrow (\mathbb{F}_q^m)^{(n-k) \times n} \) and \( x \leftarrow (\mathbb{F}_q^m)^n \) such that \( w_R(x) = r \), and outputs \((H, H \cdot x^T)\).

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

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 [4]. For cryptography, it is also useful to use the Decisional version of the problem. Our scheme security depends on the difficulty of solving the same RSD problem defined with Double Circulant codes, rather than random linear codes. The decisional version of this problem is a special case of the Decisional Rank s-Quasi Cyclic Syndrome Decoding Problem defined for example in [4]. There is no known reduction from the search version of this problem to its decisional version. However, the best known attacks on the decisional version of the problem remain the direct attacks on the search version of the problem.

### 4 Veron Double Circulant identification protocol in the rank metric

The scheme we present in this section, to which we refer to as the Rank Veron Double Circulant (RVDC) identification protocol, mixes the ideas from [22],
where the Stern protocol is converted from Hamming to rank metric and the
function \( P_{\mathcal{F}_Q} \) (see Section 3 above) is introduced, and from [5], where the
cheating probability of the Veron protocol is improved from 2/3 to 1/2 using
the double circulant technique in the Hamming metric. In [5], the intermediate
challenge is a random parallel left rotation. To better exploit the rank metric
properties, and to make it more difficult to guess the challenge for an attacker,
we instead consider a random linear combination of all possible parallel left
rotations.

**Definition 4.** Let \( n = 2k \) and \( x = (x_1, \ldots, x_k) \in (\mathbb{F}_q^m)^k \), \( y = (y_1, \ldots, y_n) \in
(\mathbb{F}_q^m)^n \). We denote with

\[
\text{rot}_i((x_1, \ldots, x_k)) = (x_{i+1}, \ldots, x_k, x_1, \ldots, x_i)
\]

the left rotation of \( i \) positions of the vector \( x \), and with

\[
\text{drot}_i((y_1, \ldots, y_i, y_{i+1}, \ldots, y_k, y_{k+1}, \ldots, y_{k+i}, y_{k+i+1}, \ldots, y_{k+k})) =
(y_{i+1}, \ldots, y_k, y_1, \ldots, y_{i+1}, \ldots, y_{k+k}, y_{k+1}, \ldots, y_{k+i})
\]

the parallel left rotation of \( i \) positions of the two halves of the vector \( y \). Given
\( a = (\alpha_1, \ldots, \alpha_k) \in (\mathbb{F}_q)^k \) we also denote with \( \Gamma'_a(x) \) the linear combination of all
possible \( k \) left rotations of \( k - i \) positions of \( x \), and \( \Gamma_a(y) \) the linear combination
of all possible \( k \) parallel left rotations of \( i \) positions of \( y \)

\[
\Gamma'_a(x) = \sum_{i=1}^{k} \alpha_i \cdot \text{rot}_{k-i}(x) \in (\mathbb{F}_q^m)^k, \quad \Gamma_a(y) = \sum_{i=1}^{k} \alpha_i \cdot \text{drot}_i(y) \in (\mathbb{F}_q^m)^n.
\]

The following lemma, used to prove the completeness of the scheme, can be
easily proven.

**Lemma 1.** Given the \( k \times 2k \) generator matrix \( G \) of a double circulant linear
code and a vector \( x = (x_1, \ldots, x_k) \in (\mathbb{F}_q^m)^k \), the following property holds

\[ \Gamma_a(x \cdot G) = \Gamma'_a(x) \cdot G \]

As we already noted in Section 3 a codeword \( y \) of a \([2k, k]\) double circulant
code can be seen as the concatenation of two blocks, i.e. \( y = (y_1, y_2) \), of length
\( k \). If we consider each block \( y_1, y_2 \) as a polynomial in \( R = \mathbb{F}_q[X]/(X^k - 1) \) then the function
\( (y_1, y_2) \mapsto \text{drot}_i((y_1, y_2)) \) is equal to \( (y_1, y_2) \mapsto (X^i \cdot y_1, X^i \cdot y_2) \),
where the multiplication by \( X^i \) is performed in the ring, i.e. modulo \( X^k - 1 \).

Although there is no general complexity result for quasi-cyclic codes, their
decoding is considered to be difficult by the community. There exist structural
attacks which uses the cyclic structure of the code [34,24,23,29], but these attacks
have only a very limited impact on the practical complexity of the problem. These
attacks are especially efficient in the case when the polynomial \( X^n - 1 \) has many
small factors. These attacks become inefficient as soon as \( X^n - 1 \) has only two
factors of the form \( (X - 1) \) and \( X^{n-1} + X^{n-2} + \ldots + X + 1 \), which is the case when
\( n \) is primitive in \( \mathbb{F}_q^m \). The conclusion is that in practice, the best attacks are the
same as those for non-circulant codes up to a small factor. Another solution to completely avoid such attacks is to use the ring \( R = \mathbb{F}_q[X]/(X^k - p(X)) \), where \( p(X) \) is a polynomial with coefficients in \( \mathbb{F}_q \), and \( X^k - p(X) \) is irreducible over \( \mathbb{F}_q \).

Recall that we will denote by \( \lambda \) the security level of the scheme. The key generation algorithm is listed in Fig. 1. The RVDC identification protocol is listed in Fig. 2.

\[
\text{RVDC: KGen}(1^\lambda) \\
1: \text{Define } m, n, k, r \text{ as in Sect. 5} \\
2: \quad x \leftarrow (\mathbb{F}_{q^m})^k \\
3: \quad e \leftarrow (\mathbb{F}_{q^m})^n \text{ s.t. } w_R(e) = r \\
4: \quad sk \leftarrow (x, e) \\
5: \quad G \leftarrow (\mathbb{F}_{q^m})^n \\
6: \quad G' \in (\mathbb{F}_{q^m})^{k \times n} \leftarrow \text{Expand } G \text{ in double circulant form} \\
7: \quad y \leftarrow x \cdot G' + e \\
8: \quad pk \leftarrow (y, G, r) \\
9: \quad \text{return } sk, pk
\]

Fig. 1. RVDC key generation algorithm in the rank metric

In Section C, we describe how to convert the identification protocol from Fig. 2 into a signature scheme, to which we will refer to as Rank Veron Double Circulant (RVDC) Signature scheme, using a generalization of the Fiat-Shamir transform, introduced in [16]. The signature size of the scheme can be reduced by applying the commitment compression technique used in [5]. We will call the scheme resulting from this variation compressed Rank Veron Double Circulant (cRVDC) scheme.

In Sect. B we prove that the identification protocol is complete, sound and that the communication leaks no information on the secret key. The security of RVDC scheme is based on a variant of the Rank Syndrome Decoding problem, that we call Differential Rank Decoding Problem, defined as Problem 2 in the same section.

5 Parameters choice

In this section we provide a set parameters for 80, 128, 192, 256 bit of classical security, corresponding to 40, 64, 96, 128 bit of quantum security, the last three falling into category 1, 3, and 5 in the NIST post-quantum competition.

The best generic combinatorial attack to solve the RSD problem has a complexity of \( \mathcal{O} \left( (n - k)^3 m^3 q^{-\frac{(k+1)m}{q(m-k)}} - m \right) \) [7]. If \( k \geq \left\lceil \frac{(r+1)(k+1)-(n+1)}{n+1} \right\rceil \), an algebraic
\begin{align*}
\text{Prover} & \quad \text{Verifier} \\
\text{sk, pk} = (x, e, y, G, r) & \leftarrow \text{KGen} \\
\text{pk} & \\
u & \leftarrow (F_{qm})^k \\
Q & \leftarrow M_{m,m}(F_q), P \leftarrow M_{n,n}(F_q) \\
c_1 & \leftarrow H(P, Q) \\
c_2 & \leftarrow H(\Pi_{P,Q}(u \cdot G)) \\
a & \leftarrow (\alpha_1, \ldots, \alpha_k) \leftarrow (F_q)^k, \\
a_i \text{ not all the same} \\
c_3 & \leftarrow H(\Pi_{P,Q}(u \cdot G + \Gamma_a(e))) \\
\text{if } b = 0 & \\
r_1 & \leftarrow (P, Q), r_2 \leftarrow u + \Gamma'_u(x) \\
r_1, r_2 & \leftarrow (r_1, r_2) \text{ if } c_1 = H(r_1) \land c_3 = H(\Pi_{r_1}(r_2 \cdot G + \Gamma_a(y))) \quad \text{return true} \\
\text{if } b = 1 & \\
r_1 & \leftarrow \Pi_{P,Q}(u \cdot G), r_2 \leftarrow \Pi_{P,Q}(\Gamma_a(e)) \\
r_1, r_2 & \leftarrow (r_1, r_2) \text{ if } c_2 = H(r_1) \land c_3 = H(r_1 + r_2) \land \text{wr}(r_2) = r \quad \text{return true}
\end{align*}

Fig. 2. RVDC identification protocol in the rank metric

approach \cite{20} is also possible to recover the error in \(O\left(\frac{k^3q^r\lceil(k+1)(k+1)-n+1\rceil}{2^m}\right)\) steps. Finally, to avoid specific Gröbner basis attacks, the condition \(n > r(k+1)\) should hold. We choose the values \(m, n, k, r\) accordingly. As far as it concerns post-quantum security, the author of \cite{28}, in line with \cite{11}, presents some arguments showing that the post-quantum complexity of RSD is computed by square-rooting the exponential term in the classical complexity formula.

Recall that for the case of double circulant code we have to choose \(n = 2k\). As suggested in \cite{36}, it is better to choose \(r\) slightly below the theoretical distance \(d\) provided by the Gilbert-Varshamov bound, in order to avoid possible small rank attacks similar to small weight codewords attack such as \cite{35}. We choose \(m\) to be prime, so to have no subfields of \(F_{2^m}\), which in other cases leads to attacks. We also need to choose the number of rounds \(\delta\) in order to decrease the impersonation probability to our needs. As far as it concerns the identification protocols, the impersonation probability of one single round for RVDC is \(p = \frac{q^k+p}{2^m}\) with overwhelming probability. To reach a security level \(l\) with an impersonation probability of \(p\), i.e. to compute the number of round \(\delta\), we
need to set $\delta = \log_p (1/2^t)$. This results in $\delta = 81, 129, 193, 257$, corresponding to 80, 128, 192, 256 bit security level in the classical scenario. In Table 1, we propose 4 sets of parameters, respectively for the 80, 128, 192, and 256 bit security level in the classical scenario, for both RVDC and cRVDC signature schemes.

For all the proposed parameters it holds the condition $k \leq \left\lceil \frac{(r+1)(k+1)-{(n+1)}}{r} \right\rceil$, so the algebraic attack of [20] must be taken into consideration while evaluating the security.

In the table $A = r^3 k^3 q^\left\lceil \frac{(r+1)(k+1)-(n+1)}{2r} \right\rceil$, $B = (n - k)^3 m^3 q^\left\lceil \frac{(k+1)m}{n} \right\rceil$, $C = r^3 k^3 q^\left\lceil \frac{(r+1)(k+1)-(n+1)}{2r} \right\rceil$, $D = (n - k)^3 m^3 q^\left\lceil \frac{(k+1)m}{2n} \right\rceil$.

| Parameters | Classic Attacks WF | Quantum Attacks WF |
|------------|--------------------|--------------------|
| $\lambda$ | $q$ | $m$ | $n$ | $k$ | $r$ | $\rho$ | $\delta$ | $h$ | $\log_2 A$ | $\log_2 B$ | $\log_2 C$ | $\log_2 D$ |
| 96 | 2 | 29 | 22 | 11 | 7 | 10 | 81 | 160 | 95.801 | 106.68 | 60.800 | 51.316 |
| 125 | 2 | 31 | 26 | 13 | 8 | 10 | 129 | 256 | 124.10 | 128.50 | 76.102 | 61.733 |
| 193 | 2 | 41 | 34 | 17 | 10 | 10 | 193 | 384 | 192.23 | 204.39 | 112.23 | 95.864 |
| 257 | 2 | 47 | 38 | 19 | 12 | 12 | 257 | 512 | 251.50 | 279.25 | 143.50 | 130.83 |

Table 1. RVDC and cRVDC parameters.

6 Key and signature size comparison

In Table 2 we report some key and signature bit sizes for other signature schemes based on codes. In particular, we report the results of hash-and-sign signature schemes such as Parallel-CFS [18], the three NIST competitors for signatures based on codes, RankSign [21], RaCoSS [3], and pqsigRM [1], and Wave [17], which has been proposed very recently. We also add the results from [6] regarding the Hamming variants of Stern, Veron and CVE signature schemes, one entry for the parameters proposed in [5] for the double circulant version of Veron scheme in the Hamming metric, and one entry for the parameters proposed in [22] for the rank version of Stern signature scheme. As far as it concerns the latter, we remark that when the work was published, results from [20], [7], and [28] were not known, so the security was believed to be 83 bits. While for the parameters in [5], according the decoding complexity estimation of $2^{0.097 n}$ given in [30], the security of the scheme is about 68 bits, while in [5] was claimed to be 81. Recall also that for all three NIST competitors some attacks have been found, so either the parameters should be made larger or some modification of the scheme will be proposed in the future.

For completeness, we also report key and signature size of one of the most popular hash-based signature scheme, SPHINCS*, introduced in [12]. The parameters that we consider are from the NIST submission document [2]. We can see that SPHINCS* has signatures and keys that are from 2 to 5 times smaller compared to cRVDC.
| $\lambda$ | Scheme | Metric | Scheme parameters | $|\text{sgn}|$ | $|\text{sk}|$ | $|\text{pk}|$ |
|---|---|---|---|---|---|---|
| 81 | Parallel-CFS | Hamm. | $(m, t, \delta, i)$ | 294 20 971 680 167 746 560 | 196 2 228 394 22 253 140 |
| 80 | Parallel-CFS | Hamm. | $(n, k, \omega, Q)$ | 196 2 228 394 22 253 140 | 196 2 228 394 22 253 140 |
| 177 | RaCoSS | Hamm. | $(2400, 2060, 48, 0.07)$ | 890 5 760 000 816 000 | 2436 1 527 360 155 520 |
| 177 | RaCoSS(Compr.) | Hamm. | $(2400, 2060, 48, 0.07)$ | 890 5 760 000 816 000 | 2436 1 527 360 155 520 |
| 128 | RankSign I | Rank | $(2^{32}, 21, 20, 10, 2, 2, 1, 8)$ | 11 008 540 288 80 640 | 11 008 540 288 80 640 |
| 128 | RankSign II | Rank | $(2^{32}, 24, 24, 12, 2, 2, 10)$ | 12 000 652 032 96 768 | 12 000 652 032 96 768 |
| 128 | RankSign III | Rank | $(2^{32}, 27, 24, 12, 2, 3, 1, 10)$ | 17 280 1 034 208 155 520 | 17 280 1 034 208 155 520 |
| 128 | RankSign IV | Rank | $(2^{32}, 30, 28, 14, 2, 3, 12)$ | 23 424 1 527 360 228 480 | 23 424 1 527 360 228 480 |
| 128 | pq sigRM-4-12 | Hamm. | $(4, 12, 16, 1295)$ | 4 224 27 749 002 2 621 788 | 4 224 27 749 002 2 621 788 |
| 128 | pq sigRM-6-12 | Hamm. | $(6, 12, 8, 311)$ | 4 224 19 326 902 3 980 860 | 4 224 19 326 902 3 980 860 |
| 256 | RankSign IV | Rank | $(6, 13, 16, 1441)$ | 8 320 16 777 216 84 020 992 | 8 320 16 777 216 84 020 992 |

Table 2. Comparison of keys and signature bit sizes between our proposals and the most popular code-based and hash-based signature schemes.

### 7 Performance

The cost of RVDC, and cRVDC key generation algorithm is dominated by the multiplication of a vector to the generator matrix. Only one multiplication is needed to generate the public key, and this makes the key generation particularly fast. On the other hand, the cost of signature and verification algorithms are dominated by the number of rounds and the cost of the underlying hash function. In particular, in the RVDC scheme (see Appendix C), $3\delta + 2$ and $2\delta + 2$ hashes
have to be computed, respectively, for the signature and the verification. In the cRVDC scheme, $3\delta + 3$ and $2\delta + 3$ hashes have to be computed, respectively, for the signature and the verification.

In Table 3, we report the performance of our scheme on a MacBook Pro equipped with a 2.9 GHz Intel Core i7 and a Huawei P20 Pro equipped with a Kirin 970 supporting ARMv8 instructions. The implementation is using AVX2 or NEON instructions sets for the finite field arithmetic but not on any other part of the code. The hash functions used are from the SHA2 family when the digest size matched the requirements and SHAKE256 when a longer output was needed. We also used AES-CTR-DRBG as a PRNG for random number generation. We compared our implementation with the optimized implementation of SPHINCS$^+$-SHAKE256 from SPHINCS$^+$ NIST submission package. As observed, our proposals outperform SPHINCS$^+$ in all cases. The table entries are in operations per second.

| Scheme      | Security Level | KGen | Sign | Vf  | KGen | Sign | Vf  |
|-------------|----------------|------|------|-----|------|------|-----|
| RVDC        | 80             | 122706.66 | 333.27 | 1447.46 | 68023.54 | 153.42 | 607.5 |
| cRVDC       | 80             | 122706.66 | 322.24 | 1420.00 | 68023.54 | 148.07 | 582.97 |
| RVDC        | 128            | 94041.80  | 146.87 | 267.3  | 24982.40 | 32.79  | 130.31 |
| cRVDC       | 128            | 94041.80  | 164.15 | 287.27 | 24982.40 | 31.61  | 129.87 |
| SPHINCS$^+$-128f | 128         | 194.81   | 143.73 | na     | na     | na     | na   |
| RVDC        | 192            | 47343.91  | 62.67  | 267.3  | 24982.40 | 32.79  | 130.31 |
| cRVDC       | 192            | 47343.91  | 64.69  | 287.27 | 24982.40 | 31.61  | 129.87 |
| SPHINCS$^+$-192f | 192        | 132.14   | 9.73   | 93.75  | na     | na     | na   |
| RVDC        | 256            | 28134.23  | 43.49  | 178.53 | 14157.74 | 19.79  | 81.46 |
| cRVDC       | 256            | 28134.23  | 41.74  | 182.27 | 14157.74 | 19.08  | 80.33 |
| SPHINCS$^+$-256f | 256        | 55.72    | 4.7    | 95.45  | na     | na     | na   |

Table 3. RVDC and cRVDC operations per second.

8 Conclusions

We have presented two code-based signature schemes derived from a 5-pass identification protocol with cheating probability close to 1/2, using double circulant codes in the rank metric. The second scheme optimizes the signature size from the first one, at the cost of few hash computations. The resulting signature scheme has a signature size of approximately 11, 22, 54, and 93 KBytes for a corresponding security level of 96, 125, 193, and 254. When compared to one of the most popular post-quantum hash-based signature schemes, namely SPHINCS+, the key generation algorithm is between 350 and 500 times faster, the signing algorithm is approximately ten times faster, and the verification algorithm is twice as fast.
References

1. Post quantum signature scheme based on modified Reed-Muller code (November 2017), Post-Quantum Cryptography, NIST Round 1 Submissions.
2. SPHINCS+: Submission to the nist post-quantum project (2017), Post-Quantum Cryptography, NIST Round 1 Submissions.
3. Supporting documentation of RaCoSS (2017), Post-Quantum Cryptography, NIST Round 1 Submissions.
4. Aguilar, C., Blazy, O., Deneuville, J.C., Gaborit, P., Zémor, G.: Efficient encryption from random quasi-cyclic codes. arXiv preprint arXiv:1612.05572 (2016)
5. 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)
6. Alaoui, S.M.E.Y., Cayrel, P.L., El Bansarkhani, R., Hoffmann, G.: Code-based identification and signature schemes in software. In: International Conference on Availability, Reliability, and Security. pp. 122–136 (2013)
7. 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
8. Barreto, P.S., Misoczki, R., Simplicio Jr, M.A.: One-time signature scheme from syndrome decoding over generic error-correcting codes. Journal of Systems and Software 84(2), 198–204 (2011)
9. 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)
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. Bernstein, D.J.: Grover vs. McEliece. In: International Workshop on Post-Quantum Cryptography. pp. 73–80 (2010)
12. Bernstein, D.J., Hopwood, D., Hülsing, A., Lange, T., Niederhagen, R., Pa-pachristodoulou, L., Schneider, M., Schwabe, P., Wilcox-O’Hearn, Z.: SPHINCS: practical stateless hash-based signatures. In: Annual International Conference on the Theory and Applications of Cryptographic Techniques. pp. 368–397. Springer (2015)
13. Boneh, D., Dagdelen, Ö., Fischlin, M., Lehmann, A., Schaffner, C., Zhandry, M.: Random oracles in a quantum world. In: International Conference on the Theory and Application of Cryptology and Information Security. pp. 41–69 (2011)
14. 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)
15. Courtois, N.T., Finiasz, M., Sendrier, N.: How to achieve a McEliece-based digital signature scheme. In: International Conference on the Theory and Application of Cryptology and Information Security. pp. 157–174 (2001)
16. Dagdelen, Ö., Galindo, D., Véron, P., Alaoui, S.M.E.Y., Cayrel, P.L.: Extended security arguments for signature schemes. Designs, Codes and Cryptography 78(2), 441–461 (2016)
17. Debris-Alazard, T., Sendrier, N., Tillich, J.P.: Wave: A new code-based signature scheme. arXiv preprint arXiv:1810.07554 (2018)
18. Finiasz, M.: Parallel-CFS. In: International Workshop on Selected Areas in Cryptography. pp. 159–170 (2010)
19. Gabidulin, E.M.: Theory of codes with maximum rank distance. Problemy Peredachi Informatsii 21(1), 3–16 (1985)
20. 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)
21. Gaborit, P., Ruatta, O., Schrek, J., Zémor, G.: RankSign: an efficient signature algorithm based on the rank metric. In: International Workshop on Post-Quantum Cryptography. pp. 88–107 (2014)
22. Gaborit, P., Schrek, J., Zémor, G.: Full cryptanalysis of the chen identification protocol. In: International Workshop on Post-Quantum Cryptography. pp. 35–50 (2011)
23. Guo, Q., Johansson, T., Löndahl, C.: A new algorithm for solving ring-lpn with a reducible polynomial. IEEE Transactions on Information Theory 61(11), 6204–6212 (2015)
24. Hauteville, A., Tillich, J.P.: New algorithms for decoding in the rank metric and an attack on the lrpc cryptosystem. In: Information Theory (ISIT), 2015 IEEE International Symposium on. pp. 2747–2751. IEEE (2015)
25. Kabatianskii, G., Krouk, E., Smeets, B.: A digital signature scheme based on random error-correcting codes. In: IMA International Conference on Cryptography and Coding. pp. 161–167 (1997)
26. Kiltz, E., Lyubashevsky, V., Schaffner, C.: A concrete treatment of fiat-shamir signatures in the quantum random-oracle model. In: Annual International Conference on the Theory and Applications of Cryptographic Techniques. pp. 552–586 (2018)
27. Loidreau, P.: Properties of codes in rank metric. arXiv preprint cs/0610057 (2006)
28. Loidreau, P.: A new rank metric codes based encryption scheme. In: International Workshop on Post-Quantum Cryptography. pp. 3–17 (2017)
29. Löndahl, C., Johansson, T., Shooshtari, M.K., Ahmadian-Attari, M., Aref, M.R.: Squaring attacks on McEliece public-key cryptosystems using quasi-cyclic codes of even dimension. Designs, Codes and Cryptography 80(2), 359–377 (2016)
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. Misoczki, R., Tillich, J.P., Sendrier, N., Barreto, P.S.: MDPC-McEliece: New McEliece variants from moderate density parity-check codes. In: Information Theory Proceedings (ISIT), 2013 IEEE International Symposium on. pp. 2069–2073. IEEE (2013)
32. NIST: Round 1 submissions (2018), available at https://csrc.nist.gov/Projects/Post-Quantum-Cryptography/Round-1-Submissions
33. Schrek, J.: Signatures et authentifications pour les cryptosystèmes basés sur les codes correcteurs en métrique de Hamming et en métrique rang. Ph.D. thesis, Université de Limoges (2013)
34. Sendrier, N.: Decoding one out of many. In: International Workshop on Post-Quantum Cryptography. pp. 51–67. Springer (2011)
35. Stern, J.: A method for finding codewords of small weight. In: International Colloquium on Coding Theory and Applications. pp. 106–113 (1988)
36. Stern, J.: A new identification scheme based on syndrome decoding. In: Annual International Cryptology Conference. pp. 13–21 (1993)
37. Unruh, D.: Non-interactive zero-knowledge proofs in the quantum random oracle model. In: Annual International Conference on the Theory and Applications of Cryptographic Techniques. pp. 755–784 (2015)
A Post-quantum security of the Fiat-Shamir transform

It is well known that the Fiat-Shamir transform is secure in the random oracle model (ROM), see e.g. [26]. However, when the adversary has a quantum access to the oracle, i.e. in the quantum random oracle model (QROM), the situation is somehow more complex, and recently many results have been published (e.g. [13], [37], [26], [38]). Since most of the schemes we compare to do not take into account this scenario, we also omit it, and leave it to future research.

An alternative quantum secure transform by Unruh [37] could be used instead of the Fiat-Shamir one, yielding though a considerably less efficient signature, since multiple executions of the underlying identification scheme are required. In [38], it is proven that if a sigma-protocol has honest-verifier zero-knowledge and statistical soundness with a dual-mode hard instance generator, then the resulting Fiat-Shamir signature scheme is unforgeable in the quantum scenario. It is easy to see that our proposal has a dual-mode hard instance generator and honest-verifier (computational) zero-knowledge, but on the other hand, a computationally unbounded adversary would prevent us from achieving statistical soundness. Thus, we cannot apply the results of [38] to our proposal. Still, to the best of our knowledge, no quantum attack has been published to Veron-like constructions.

B Zero-knowlegde properties of RVDC signature scheme

In this section we prove the security of RVDC scheme by showing how the completeness, soundness and zero-knowledge properties are achieved. In the proofs we follow [5].

Completeness Given \((sk, pk)\) output from \(KGen\) function, it is easy to see that for any possible \(sk = (x, e)\) the Verifier always accepts after interacting with the Prover \(P\) on common input \(pk\). This is because the honest Prover who knows \(sk\) is be able to construct the three commitments \(c_1, c_2, c_3\). Furthermore, the Verifier is always able to identify the Prover because the verifications match with the given commitments.

In particular, the check on the value \(c_3\) when \(b = 0\) is valid because of Lemma 1, i.e. \(\Gamma_a(x \cdot G) = \Gamma'_a(x) \cdot G\). Thanks to this we have that \(u \cdot G + \Gamma_a(e) = u \cdot G + \Gamma_a(x \cdot G) + \Gamma_a(y) = u \cdot G + \Gamma'_a(x) \cdot G + \Gamma_a(y) = (u + \Gamma'_a(x)) \cdot G + \Gamma_a(y)\).

Notice also that the components of the first challenge \(a\) cannot be all the same, otherwise \(w_R(\Gamma_a(e)) = 0\) or 2, depending of \(a\) being equal to \((0, \ldots, 0)\), \((\tilde{a}, \ldots, \tilde{a})\) respectively, and the check when \(b = 1\) would fail.
Soundness We will show that if someone can be successfully identified by \( V \) with the protocol, then it is able to retrieve the secret in polynomial time with a certain probability. To do so, we introduce a specific problem which is easier to be solved than the syndrome decoding, \(^3\) except when there is only one solution, in which case the two problems are the same. The way in which we assure the security is by choosing the parameters which allow to decrease the size of the solutions of the new problem to one with a probability exponentially close to 1 (in practice, this probability to have more than one solution is \( 2^\lambda \)).

Problem 2 (Differential Rank Decoding Problem). Consider \( H \) a random double circulant matrix, \( Y \) a random codeword in \((\mathbb{F}_q)^n\) of rank weight \( r \), and \( A = \{a_1, \ldots, a_\rho\} \subseteq (\mathbb{F}_q)^k \), with \( a_j \) all distinct for \( j = 1, \ldots, \rho \), and \( a_j = (a_1, \ldots, a_k) \), with \( a_1, \ldots, a_k \) all distinct. Let \( H \cdot Y^T \) be a syndrome. The problem \( \mathcal{P}(H, Y^T, A, r) \) consists in finding \( \rho \) words \( z_j \) and a constant \( C \) such that \( H \cdot \Gamma_{a_j}(Y)^T - H \cdot z_j^T = C \), and \( w_R(z_j) = r \) for all \( j < \rho \).

The above mentioned problem is easier than the independent syndrome decoding problem, because of the addition of the unknown \( C \). However, it still seems to be hard to be solved. Note that we can suppose that there exist a particular solution \( Z_1, \ldots, Z_\rho, C \) to the problem \( \mathcal{P}(H, Y^T, A, r) \), such that \( C \) is equal to 0. In this case, we have to solve the usual rank syndrome decoding problem \( H \cdot \Gamma_{a_j}(Y)^T = H \cdot z_j^T \) for all \( j < i \).

Lemma 2 gives the probability to find a solution of Problem 2

**Lemma 2.** Consider \( \rho, A, r \) fixed. Let \( Z_C = (Z_1, \ldots, Z_\rho, C) \) be a random vector with \( Z_j, 1 \leq j \leq \rho \) a random variable with uniform distribution over the words of rank weight \( r \), and \( C \) a random variable with uniform distribution over \((\mathbb{F}_q^m)^{n-k}\). Let \( S_\rho \) be a random variable equal to the set of the solutions of the problem \( \mathcal{P}(H, Y^T, A, r) \), \( \rho, A, r \) as in Problem 2. Note that \( S_\rho \) is a random variable, in the sense that \( S_\rho \) is defined relatively to \( H \) a random double circulant matrix and \( Y \) a random word of weight \( r \). We have \( \Pr[Z_C \in S_\rho] = \frac{1}{(q^{m(n-k)})^{\rho}} \).

**Proof.**

\[
\Pr[Z_C \in S_\rho] = \\
= \Pr[H \cdot Z_1^T = C + H \cdot \Gamma_{a_1}(X)^T \land \ldots \land H \cdot Z_\rho^T = C + H \cdot \Gamma_{a_\rho}(X)^T],
\]

which, by the conditional probability formula, is the product of the following two probabilities

\[
\Pr[H \cdot Z_1^T = C + H \cdot \Gamma_{a_1}(X)^T \land \ldots \land H \cdot Z_\rho^T = C + H \cdot \Gamma_{a_\rho}(X)^T].
\]

\[
\Pr[\bigcap_{j=2}^{\rho} H \cdot Z_j^T = C + H \cdot \Gamma_{a_j}(X)^T].
\]

\(^3\) This problem is the analog of the Differential Syndrome Decoding Problem (denoted Problème de décodage par syndrome différentiel) in [33], for the Hamming metric. The same problem is used in [5].
In the case where the words of rank weight $r$ do not have a common image for $H$, we have that:

$$
\Pr[\mathbf{Z}_C \in S_\rho] = \Pr\left[H \cdot \mathbf{Z}_1^T = C + H \cdot \Gamma_{a_1}(X)^T \left| \bigcap_{j=2}^{\rho} Z_1 \neq Z_j \right.\right].
$$

$$
\Pr\left[\bigcap_{j=2}^{\rho} H \cdot \mathbf{Z}_j^T = C + H \cdot \Gamma_{a_j}(X)^T \right].
$$

These variables are independent, so

$$
\Pr[\mathbf{Z}_C \in S_\rho] = \Pr[H \cdot \mathbf{Z}_1^T = C + H \cdot \Gamma_{a_1}(X)^T] \cdot \Pr\left[\bigcap_{j=2}^{\rho} H \cdot \mathbf{Z}_j^T = C + H \cdot \Gamma_{a_j}(X)^T \right].
$$

Using a recursive argument, we have that

$$
\Pr[\mathbf{Z}_C \in S_\rho] = \Pr[H \cdot \mathbf{Z}_1^T = C]^\rho.
$$

The hardness of the Decisional Rank Double Circulant Syndrome Decoding (DRDCSD) Problem (Defined in [4]) assures that the syndromes associated to codewords of given rank are indistinguishable from random syndromes, i.e. they are uniformly distributed among the syndrome space $(\mathbb{F}_{q^m})^{n-k}$. Thus, we can conclude that

$$
\Pr[\mathbf{Z}_C \in S_\rho] = \frac{1}{(q^{m(n-k)})^\rho}.
$$

**Lemma 3.** The distribution of $N_\rho$ describing the size of $S_\rho$ is the same of the variable $1 + Y$, with $Y$ a binomial distribution with parameters $N = (q^{m(n-k)} - 1)[n \choose r]$ and $p = \frac{1}{(q^{m(n-k)})^\rho}$. Furthermore

$$
\mathbb{E}[N_\rho] = Np + 1 = (q^{m(n-k)} - 1) \left( \frac{[n \choose r]}{q^{m(n-k)}} \right)^\rho.
$$

**Proof.** Let $\mathbf{Z}_C = (Z_1, \ldots, Z_\rho, C)$ be the random vector defined in Lemma 2, with $C \neq 0$ and $T_C$ the variable equal to 1 when $\mathbf{Z}_C \in S_\rho$ and 0 otherwise. $N_\rho = \sum_{C \neq 0} T_C + T_0$, with $T_0$ the number of solutions when $C = (0, \ldots, 0)$. The variable $T_0$ is equal to 1 since for a given $C$ and $\rho$ distinct codewords of rank weight $r$ only one solution can be found. The number of words of given rank weight $r$ is given by the number of vector subspaces of length $n$ and dimension $r$, which is indicated with $[n \choose r]$ (defined in Section 3), while the number of all possible $C \neq (0, \ldots, 0)$ is $q^{m(n-k)} - 1$. So, we have $N_\rho = 1 + Y$ with $Y$ a binomial distribution with parameters $N = (q^{m(n-k)} - 1)[n \choose r]$ and $p = \frac{1}{(q^{m(n-k)})^\rho}$. $\square$
Lemma 4. Let $Y'$ be a random variable with Poisson distribution with parameter $Np$. Then we have $\Pr[N_p = 1] \approx \Pr[Y' = 0] \approx 1 - \frac{[n]^p}{q^{m(n-k)(x-1)}}$.

Proof. For a sufficiently large $N$ and sufficiently small $p$, the binomial distribution of $Y'$ is approximated by the Poisson distribution with parameter $\lambda = Np$. We can deduce that the probability $\Pr[N_p = 1] \approx \Pr[Y' = 0] = e^{-Np}Np^0 \approx e^{-\frac{x}{q^{m(n-k)(x-1)}}}$. When $x$ is closed to 0, we have that $e^x \approx 1 - x$, and thus $\Pr[N_p = 1] \approx 1 - \frac{[n]^p}{q^{m(n-k)(x-1)}}$. □

Let us call $\epsilon$ the value $1 - \frac{[n]^p}{q^{m(n-k)(x-1)}}$.

Lemma 5. If someone is able to solve the problem $\mathcal{P}(H, Y, \rho, A, r)$ with probability $\epsilon'$, then he is also able to find the secret key of the protocol from the public key with a probability of about $\epsilon \epsilon'$.

Proof. We have from Lemma 4 that the probability that the solution of $\mathcal{P}(H, Y, \rho, A, r)$ is unique is $\epsilon$.

□

Theorem 1. If a prover $P$ is able to be authenticated by a a verifier $V$ with a probability greater than $\frac{a^x + a^y}{2q}$, then $P$ is able to retrieve the secret key of the protocol from the public key with a probability greater than $1 - \frac{[n]^p}{q^{m(n-k)(x-1)}}$ in polynomial time or to find a collision on the underlying hash function in polynomial time.

Proof. The prover $P$ is able to correctly answer more than $k + \rho$ challenge queries. In this case, let us call $a, b$, respectively, the first and the second challenge of $V$. First $P$ randomly chooses $P \in M_{n,n}(\mathbb{F}_q)$, $Q \in M_{m,m}(\mathbb{F}_q)$, and $v \in (\mathbb{F}_{q^m})^n$, and sends the first two commitments $c_1 = H(P, Q)$ and $c_2 = H(v)$ to $V$.

We call $c_3$ the second commitment sent to $V$.

We also call $(u_a, P_a, Q_a)$ and $(v_a, z_a)$ the last response, respectively, when $b = 0$, and when $b = 1$. For the Pigeonhole principle, $P$ is able to answer to the challenge $(a, b = 0)$ and $(a, b = 1)$ for at least $\rho$ different $a$, which we call $a_1, \ldots, a_\rho$. $V$ must verify that $H(P_{a_j}, Q_{a_j}) = c_1$ and $H(v_{a_j}) = c_2$. Thus, for any $j \in \{1, \ldots, \rho\}$, either $P$ finds a collision of the hash function, or $(P_{a_j}, Q_{a_j}) = (P, Q)$ and $v_{a_j} = v$ for all $a_j$. $V$ must also verify that the rank weight of $z_{a_j}$ equals $r$ and that the commitment $c_3$ is correct. To meet this last condition, the values $P, Q, u_{a_j}, v, z_{a_j}$ generated by $P$ must satisfy the condition $H(P_{a_j}, Q_{a_j}) = c_1$, and thus $H(v_{a_j}) = c_2$. Thus, $P$ finds a collision of the hash function, or $H(P, Q)(u_{a_j}, G + \Gamma_{a_j}(y)) = H(v + z_{a_j})$, since both side of the equation must be equal to $c_3$. Thus, either $P$ finds a collision of the hash function, or $H(P, Q)(u_{a_j}, G + \Gamma_{a_j}(y)) = v + z_{a_j}$.

In this case, we deduce that $u_{a_j}G + \Gamma_{a_j}(y) = \Pi_{P, Q}^{-1}(v) + \Pi_{P, Q}^{-1}(z_{a_j})$, and then $H \cdot \Gamma_{a_j}(y)^T - H \cdot \Pi_{P, Q}^{-1}(z_{a_j}) = H \cdot \Pi_{P, Q}^{-1}(v)^T$. Since $H \cdot \Gamma_{a_j}(y)^T = H \cdot \Gamma_{a_j}(e)^T$, the previous equation corresponds to the problem $\mathcal{P}(H, Y, A, r)$. We deduce from Lemma 5 that $P$ is able to find the secret key with a probability greater than $\epsilon$.

□
Theorem 2. If a prover $P$ is able to be authenticated by a verifier $V$ with a probability greater than $\left(\frac{q^k + \rho}{2q^k}\right)^N$, then $P$ is able to retrieve the secret key of the protocol from the public key with a probability greater than $1 - \frac{[n]^p}{q^{m(n-k)(r-1)}}$ in polynomial time or to find a collision on the underlying hash function in a polynomial time.

Proof. $P$ is able to build $c_{1,1}, \ldots, c_{1,N}$ and $c_{2,1}, \ldots, c_{2,N}$ such that it can be authenticated with a probability greater than $\left(\frac{q^k + \rho}{2q^k}\right)^N$. For the Pigeonhole principle, we can deduce the existence of an integer $j$ such that $P$ can be authenticated by the first protocol with a probability greater than $q^k + \rho$. Theorem 1 allows to conclude the proof. \qed

Zero-Knowledge We need to prove that, beside the public parameters, no information can be deduced in polynomial time from an execution of the protocol.

We need to construct a polynomial time simulator $S$ of the protocol that, by interacting with the verifier $V$, provides a transcript which is indistinguishable from the one of the original protocol.

The simulator $S$ should perform the following steps

- if $b = 0$:
  - choose random $P' \in M_{n,n}(\mathbb{F}_q)$, $Q' \in M_{m,m}(\mathbb{F}_q)$, and $v \in (\mathbb{F}_{q^m})^n$;
  - choose random $a' \in (\mathbb{F}_q)^k$;
  - compute $h_1 = H(P', Q')$, and $h_3 = H(\Pi_{P', Q'}(v \cdot G + \Gamma_a(y)))$.
  Note that $P', Q', v$ are indistinguishable from $P, Q, u + \Gamma_a(x)$;
- if $b = 1$:
  - choose random $P' \in M_{n,n}(\mathbb{F}_q)$, $Q' \in M_{m,m}(\mathbb{F}_q)$, $v \in (\mathbb{F}_{q^m})^n$, and $z \in (\mathbb{F}_{q^m})^n$ such that $w_R(z) = r$;
  - compute $h_2 = H(\Pi_{P', Q'}(v))$, and $h_3 = H(\Pi_{P', Q'}(v) + z)$.
  Note that $\Pi_{P', Q'}(v), z$ are indistinguishable from $\Pi_{P, Q}(u \cdot G), \Pi_{P, Q}(\Gamma_a(e))$, since, if $P, Q$ are random matrices, then the function $\Pi_{P, Q}$ can map a vector of a certain rank to any vector of the same rank. Furthermore, the function $\Gamma_a$ preserves the rank.

The simulator just described runs in polynomial time.

C Veron Double Circulant Signature schemes in the rank metric

In this section we provide the description of the signature scheme derived from Veron identification protocol using double circulant codes (Section 4), which we will refer to as Rank Veron Double Circulant (RVDC) Signature scheme. We also consider a version of the scheme with a signature compression, and we refer to it as compressed Rank Veron Double Circulant (cRVDC) signature scheme.
RVDC: Sign\((sk, pk, msg, \delta)\) \hspace{2cm} RVDC: Verify\((pk, msg, \delta, sgn)\)

\[
\begin{array}{ll}
\text{sk} = (x, c) & \leftarrow KGen \\
\text{pk} = (y, G, r) & \leftarrow KGen \\
msg & \text{message} \\
\delta, \text{number of rounds as defined in Sect. 5} & \\
\end{array}
\]

\[
\begin{array}{ll}
\text{sgn} & \text{signature} \\
\end{array}
\]

\[
\begin{array}{ll}
1: & \text{for } i = 1..\delta \text{ do} \\
2: & u_i \leftarrow (\mathbb{F}_q^m)^k \\
3: & P_i \leftarrow M_{m,n}(\mathbb{F}_q), Q_i \leftarrow M_{m,n}(\mathbb{F}_q) \\
4: & c_{i,1} \leftarrow H(P_i, Q_i) \\
5: & c_{i,2} \leftarrow H(\Pi_{P_i, Q_i} (u_i \cdot G)) \\
6: & cmt_0 \leftarrow c_{1,2}[c_{1,2}] | \ldots | c_{i,1}[c_{i,2}] \\
7: & ch_1 \leftarrow H(cmt_0) \\
8: & \text{Truncate rightmost bits in } ch_1 \\
9: & \text{so that it has } \delta k \log_2(q) \text{ bits} \\
10: & \text{and there is no block of length } k \log_2(q) \text{ with all equal components over } \mathbb{F}_q, \\
11: & \text{for } i = 1..\delta \text{ do} \\
12: & a_i \leftarrow (ch_1(c_{i-1}) k \log_2(q) + 1, \ldots, ch_1(c_{i-1}) k \log_2(q)) \\
13: & c_{i,3} \leftarrow H(\Pi_{P_i, Q_i} (u_i \cdot G + \Gamma_{G_i}(c))) \\
14: & cmt_1 \leftarrow c_{i,3}[c_{i,3}] | \ldots | c_{i,3}[c_{i,3}] \\
15: & \text{for } i = 1..\delta \text{ do} \\
16: & \text{if } ch_{2,i} = 0 \\
17: & r_{i,1} \leftarrow (P_i, Q_i) \\
18: & r_{i,2} \leftarrow u_i + \Gamma_{P_i}(x) \\
19: & \text{if } ch_{2,i} = 1 \\
20: & r_{i,1} \leftarrow H(P_i, Q_i) (u_i \cdot G) \\
21: & r_{i,2} \leftarrow H(P_i, Q_i) (\Gamma_{G_i}(c)) \\
22: & \text{return sgn} \leftarrow [cmt_0, cmt_1, r] \\
\end{array}
\]

\[
\begin{array}{ll}
1: & ch_1 \leftarrow H(cmt_0 || msg) \\
2: & ch_2 \leftarrow H(cmt_1) \\
3: & \text{for } i = 1..\delta \text{ do} \\
4: & a_i \leftarrow (ch_1(c_{i-1}) k \log_2(q) + 1, \ldots, ch_1(c_{i-1}) k \log_2(q)) \\
5: & c_{i,3} \leftarrow H(cmt_1, h((i-1)+1,\ldots,h)) \\
6: & \text{if } ch_{2,i} = 0 \\
7: & c_{i,1} \leftarrow cmt_0, [2h(i-1)+1,\ldots,2h(i-1)+h] \\
8: & \text{if } c_{i,1} \neq H(r_{i,1}) \\
9: & c_{i,3} \leftarrow H(\Pi_{r_{i,1}} (r_{i,2} \cdot G + \Gamma_{G_i}(y)) \\
10: & \text{return false} \\
11: & \text{if } ch_{2,i} = 1 \\
12: & c_{i,2} \leftarrow cmt_0, [2h(i-1)+h+1,\ldots,2h] \\
13: & \text{if } c_{i,2} \neq H(r_{i,1}) \\
14: & c_{i,3} \leftarrow H(r_{i,1} + r_{i,2}) \vee \text{wr}(r_{i,2}) \neq r \\
15: & \text{return false} \\
16: & \text{return true} \\
\end{array}
\]

**Fig. 3.** RVDC signature and verification algorithms.

Signature and verification algorithm for the RVDC and cRVDC schemes can be observed, respectively, in Fig. 3 and Fig. 4. Key generation is the same as in Sect. 4.

In the algorithm in Fig. 3, if \(\delta k \log_2(q)\) is greater than \(h\), then it is possible to compute the challenge as \(ch \leftarrow H(cmt || msg || 1) \ldots || H(cmt || msg || l) \in (\mathbb{F}_q)^{1,h}\), where \(l = [\delta k \log_2(q)/h] + 1\). Alternatively, one may use an Extended Output Function (XOF), as shown in Fig. 4, where XOF\((x)_l\) means that we take \(l\) bits from the output of the function XOF with input \(x\).
cRVDC: Sign(sk, pk, msg, δ)

\[ \text{sk} = (x, c) \leftarrow \text{KGen} \]
\[ \text{pk} = (y, G, r) \leftarrow \text{KGen} \]
\[ \text{msg}, \text{message} \]
\[ \delta, \text{number of rounds as defined in Sect. 5} \]

// Step 1
1: for \( i = 1..\delta \) do
2: \( u_i \leftarrow (P_y)^k \)
3: \( \text{seed}_i \leftarrow \{0, \ldots, 2^\lambda - 1\} \)
4: \( P_i \leftarrow \text{XOF}(1, \text{seed}_i)_{m_2} \)
5: \( Q_i \leftarrow \text{XOF}(2, \text{seed}_i)_{m_2} \)
6: \( c_{1,i} \leftarrow \text{XOF}(P_i, Q_i)_{2\lambda} \)
7: \( c_{2,i} \leftarrow \text{XOF}(H_{P_i, Q_i}(u_i \cdot G))_{2\lambda} \)
8: \( cmt_0 \leftarrow \text{XOF}(c_{1,1} || c_{1,2} || \ldots || c_{\delta,1} || c_{\delta,2})_{2\lambda} \)

// Step 2
9: \( \text{ch}_1 \leftarrow \text{XOF}(\text{cmt}_0 || \text{msg})_{\delta k \log_2(q)} \)

// Step 3
10: for \( i = 1..\delta \) do
11: \( a_i \leftarrow (\text{ch}_1, (i-1)k \log_2(q)+1, \ldots, \text{ch}_1, i k \log_2(q)) \)
12: \( c_{1,i} \leftarrow \text{XOF}(H_{P_i, Q_i}(u_i \cdot G + \Gamma_{u_i}(c)))_{2\lambda} \)
13: \( \text{cmt}_1 \leftarrow c_{1,1} || \ldots || c_{\delta,3} \)

// Step 4
14: \( \text{ch}_2 \leftarrow \text{XOF}(\text{cmt}_1)_{2\lambda} \)

// Step 5
15: for \( i = 1..\delta \) do
16: if \( \text{ch}_{2,i} = 0 \)
17: \( r_{1,1} \leftarrow u_i + \Gamma_{u_i}(x) \)
18: \( r_{1,2} \leftarrow \text{seed}_i \)
19: \( r_{1,3} \leftarrow c_{1,i} \)
20: if \( \text{ch}_{2,i} = 1 \)
21: \( r_{1,1} \leftarrow H_{P_i, Q_i}(u_i \cdot G) \)
22: \( r_{1,2} \leftarrow \text{Coordinates of } H_{P_i, Q_i}(\Gamma_{u_i}(c)) \)
23: \( r_{1,3} \leftarrow c_{1,i} \)
24: \( \text{sgn} \leftarrow [\text{cmt}_0, \text{cmt}_1, r] \)
25: \( \text{return sgn} \)

---

cRVDC: Verify(pk, msg, δ, sgn)

\[ \text{pk} = (y, G, r) \leftarrow \text{KGen} \]
\[ \text{msg}, \text{message} \]
\[ \delta, \text{number of rounds as defined in Sect. 5} \]
\[ \text{sgn} = [cmt_0, cmt_1, r], \text{signature} \]

// Step 1
1: \( \text{ch}_1 \leftarrow \text{XOF}(cmt_0 || \text{msg})_{2\lambda} \)
2: \( \text{ch}_2 \leftarrow \text{XOF}(cmt_1)_{2\lambda} \)
3: for \( i = 1..\delta \) do
4: \( a_i \leftarrow (\text{ch}_1, (i-1)k \log_2(q)+1, \ldots, \text{ch}_1, i k \log_2(q)) \)
5: \( c_{1,i} \leftarrow \text{cmt}_1 [h(i-1)+1, \ldots, h_i] \)
6: if \( \text{ch}_{2,i} = 0 \)
7: \( P' \leftarrow \text{XOF}(1, r_{1,2})_{m_2} \)
8: \( Q' \leftarrow \text{XOF}(2, r_{1,2})_{m_2} \)
9: \( c_{1,i} \leftarrow \text{XOF}(P', Q')_{2\lambda} \)
10: \( c_{2,i} \leftarrow r_{1,3} \)
11: if \( c_{1,3} \neq \text{XOF}(H_{P', Q'}(r_{1,1} \cdot G + \Gamma_{u_i}(y)))_{2\lambda} \)
12: return false
13: if \( \text{ch}_{2,i} = 1 \)
14: \( c_{1,1} \leftarrow r_{1,3} \)
15: \( c_{1,2} \leftarrow \text{XOF}(r_{1,1})_{2\lambda} \)
16: if \( c_{1,3} \neq \text{XOF}(r_{1,1} + r_{1,2}) \lor w_{R_i}(r_{1,2})_{2\lambda} \neq r \)
17: return false
18: if \( cmt_0 = \text{XOF}(c_{1,1} || c_{1,2} || \ldots || c_{\delta,1} || c_{\delta,2})_{2\lambda} \)
19: return true

---

**Fig. 4.** cRVDC signature and verification algorithms.