MR-DSS – Smaller MinRank-Based (Ring-)Signatures

MR-DSS – Smaller MinRank-based (Ring-)Signatures

Emanuele Bellini[1], Andre Esser[1], Carlo Sanna*[2], and Javier Verbel[1]

[1] Technology Innovation Institute, UAE
{emanuele.bellini,andre.esser,javier.verbel}@tii.ae

[2] Politecnico di Torino, Italy
carlo.sanna@polito.it

Abstract. In the light of NIST's announced reopening of the call for digital signature proposals in 2023 due to lacking diversity, there is a strong need for constructions based on other established hardness assumptions. In this work we construct a new post-quantum secure digital signature scheme based on the *MinRank* problem, a problem with a long history of applications in cryptanalysis that led to a strong belief in its hardness. Initially following a design by Courtois (Asiacrypt '01) based on the Fiat–Shamir transform, we make use of several recent developments in the design of sigma protocols to reduce signature size and improve efficiency. This includes the recently introduced *sigma protocol with helper* paradigm (Eurocrypt '19) and combinations with *cut-and-choose* techniques (CCS '18). Moreover, we introduce several improvements to the core of the scheme to further reduce its signature size.
As a second contribution, we formalize the natural extension of our construction to a ring signature scheme and show that it achieves desired anonymity and unforgeability guarantees. Our ring signature is characterized by a sublinear scaling of the signature size in the number of users. Moreover, we achieve competitive practical signature sizes for moderate amount of users in comparison to recent ring signature proposals.

Keywords: Fiat–Shamir · MinRank · post-quantum signature · ring signature · sigma protocols

1 Introduction

The NIST standardization process for post-quantum secure cryptographic schemes is in transition to its fourth round. While the process for post-quantum secure KEMs is progressing well, the process for digital signatures suffers from a lack of diversity among the hardness assumptions of the remaining candidates. In particular, the remaining signature schemes are either based on structured lattices or symmetric primitives [40]. Although, both foundations have desirable

* C. Sanna is a member of GNSAGA of INdAM, and of CrypTO, the group of Cryptography and Number Theory of Politecnico di Torino.

attributes, the security guarantees of structured lattice-based schemes have recently been challenged [39] and schemes based on symmetric primitives, while generally secure, suffer from performance issues [40]. For this reason, NIST announced that it will reopen the call for digital signature schemes in early 2023, making the study of schemes based on other established hardness assumptions an important and urgent task.

In this work we propose a new post-quantum secure digital signature scheme based on the so-called MinRank problem, which was introduced in 1999 by Buss, Frandsen, and Shallit [21]. Roughly speaking, the MinRank problem asks to find a low-rank linear combination of some given matrices over a finite field. The MinRank problem is an attractive candidate for post-quantum cryptography for multiple reasons. First, it is entirely based on linear algebra computations, allowing constructions to benefit from the long line of research in optimizing the involved operations [3, 36, 47]. Also, the MinRank problem has been extensively studied due to its applications in cryptanalysis [14, 19, 20, 22, 35, 41, 45, 48], where faster algorithms for solving the MinRank problem usually imply improved attacks on the involved schemes. This relation has established a strong belief in the hardness of the MinRank problem over the last decades. Furthermore, there are no quantum attacks known that go beyond straightforward quantum search applications. However, there has been very limited work on cryptographic primitives based on this problem. To the best of our knowledge, the only construction based on the hardness of the MinRank problem is a sigma protocol from 2001 due to Courtois [23].

Sigma protocols implement zero-knowledge proofs of knowledge (PoK). These constructions allow a prover to prove to a verifier that he knows a secret object x, which satisfies a certain relation, without revealing any information about x. For example, x might be the secret key corresponding to a known public key or, as in the protocol of Courtois, the solution to a publicly-known MinRank instance. Sigma protocols do not reach perfect soundness, i.e., a cheating prover not knowing x might be able to convince the verifier that he actually knows x. If a cheating prover is able to convince the verifier with probability p, then p is called the soundness error of the protocol.

It is well known that a sigma protocol that offers security against passive attacks can be transformed into a digital signature scheme, secure in the random oracle model, by using the Fiat–Shamir transform. A straightforward application of this transformation to Courtois' protocol results in a digital signature scheme with not particularly desirable parameters. However, starting from Courtois' initial protocol, we adopt recent techniques in the design of sigma-protocols to reduce the soundness error of the construction. Combining this with several modifications and improvements to the original protocol, we are able to derive a new digital signature scheme with significantly reduced signature and public key sizes.

Another major advantage of our construction is that it naturally extends to ring-signatures. A ring-signature allows a signer to sign a message on behalf of a group of users, called a *ring*. A verifier is then able to verify the signature

as usual, but can not identify the signer among all members of the ring. The performance of those schemes is usually a function depending on the number of users in the ring. There exist constant size ring-signatures [2,25], however, none of those are post-quantum secure. Our scheme is characterized by a sublinear scaling of the signature size in the number of users and allowing to achieve competitive practical parameters.

1.1 Related Work

Signature schemes constructed from sigma protocols have a long standing history (e.g. [1,11,12,30,38,42,46,49]). A main advantage of such constructions is that they do not require a trapdoor-relation. This makes it possible to base their security on presumably hard instances of the underlying problem. However, a common drawback is usually a larger signature size due to necessary repetitions of the protocol when applying the Fiat–Shamir transformation. These repetitions reduce the (high) soundness error of the sigma protocol from p to $p^R \leq 2^{-\lambda}$, where R is the number of repetitions and λ the security parameter.

Multiple recent works try to lower the initial soundness error, to reduce the amount of repetitions. Katz et al. [34] have constructed a zero knowledge PoK using the *MPC-in-the-head* paradigm [33] in combination with a preprocessing stage to distribute some auxiliary information to the participants. The protocol extends to an arbitrary number of users n resulting in a sigma protocol with soundness error $\frac{1}{n}$. Beullens [17] then generalized the approach from Katz, Kolesnikov and Wang [34] by introducing the *sigma protocol with helper* paradigm. Here, the sigma protocol uses a trusted third party, called the *helper*, to provide some auxiliary information to the verifier, which similarly results in a lower soundness error of the construction. Eventually, the helper is removed using a *cut-and-choose* technique. The helper paradigm and independent similar techniques to lower the soundness error have recently led to efficient zero-knowledge-based signature schemes [28,29,32].

1.2 Contribution

We construct a post-quantum secure digital signature scheme based on the MinRank problem, which we call *MR-DSS*. Our construction obtains about half the signature size and the public key size of a straightforward application of the Fiat–Shamir transform to the sigma protocol by Courtois.

From a design point of view, we follow the sigma protocol with helper paradigm of Beullens. By introducing the helper we reduce the soundness error of Courtois' protocol from $\frac{2}{3}$ to $\frac{1}{2}$. We then use a cut-and-choose technique from [34] to remove the helper. This results in a sigma protocol with very low soundness error, mitigating the need for multiple iterations when applying the Fiat–Shamir transform. We further introduce several improvements to Courtois' protocol reducing its communication complexity and, implicitly, the signature size. Overall, we are able to decrease the signature size by a factor of roughly 2.

Further we formalize the natural extension of our scheme to ring signatures. The possibility of such an extension was already observed by Courtois [23]. However, he did neither formalize the resulting scheme nor argue about its security or determine its parameters. We show that the extension of our scheme matches the security definitions of ring-signatures given by Bender, Katz, and Morselli [13]. Moreover, the ring signature scheme is characterized by a sublinear scaling of the signature size in the amount of users, leading to particularly good practical signature sizes, especially for moderate amounts of users.

Outline. In Section 2 we cover used notations and definitions, followed by a recap of basic properties of sigma protocols with helpers and commitment schemes. Subsequently, in Section 3 we recall the initial sigma protocol from Courtois. In the following Section 4 we then describe our new scheme, including an analysis of its public key and signature size as well as suggested parameters. Eventually, Section 5 covers our extension of the scheme to ring-signatures.

Concurrent work Recently, Santoso et al. [44] independently proposed a variation of Courtois' sigma protocol that achieves soundness probability $\frac{1}{2}$. They adapt challenge space and responses to lower the soundness. However, their approach yields only slight improvements (in the magnitude of bytes) over Courtois' signature size. Moreover, the authors disregard the size of the initial commitments in their analysis of the communication complexity. Taking commitment sizes into account they achieve no improvement over Courtois. We give more details on this in Appendix C.

2 Preliminaries

For each prime power q, we let $\mathbb{F}_q$ denote the finite field of q elements. For positive integers m and n, we write $\mathrm{M}_{m,n}(\mathbb{F}_q)$ for the vector space of $m \times n$ matrices with entries in $\mathbb{F}_q$, and $\mathrm{GL}_n(\mathbb{F}_q)$ for the group of invertible matrices in $\mathrm{M}_{n,n}(\mathbb{F}_q)$. We let λ denote the security parameter. We use standard Landau notation for complexity statements and write log for the logarithm in base 2.

Let us define the *MinRank* problem. By MinRank we refer to the search version of the MinRank problem over finite fields defined as follows.

Definition 1 (MinRank problem).

- *Parameters: Positive integers q, m, n, k, r with q a prime power.*
- *Instance: $(k+1)$-tuple $\boldsymbol{M}$ of matrices $M_0; M_1, \ldots, M_k \in \mathrm{M}_{m,n}(\mathbb{F}_q)$.*
- *Solution: $\alpha \in \mathbb{F}_q^k$ such that $E := M_0 + \sum_{i=1}^{k} \alpha_i M_i$ has rank less than or equal to r.*

2.1 Sigma protocols with helper

Sigma protocols with helper were recently introduced by Beullens [17]. Informally, a sigma protocol with helper extends a sigma protocol by adding a trusted party, which is called the *helper*. This trusted party runs a setup algorithm based on a random seed at the beginning of each execution of the protocol. The helper then sends the seed value to the prover and some auxiliary information to the verifier. The formal definition of a sigma protocol with helper is the following.

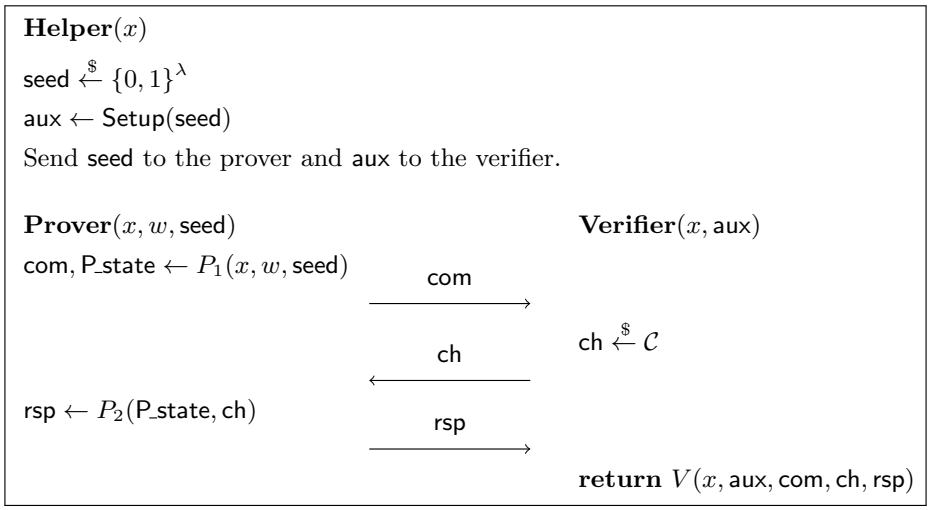

Fig. 1: Structure of a sigma protocol with helper.

Definition 2 (Sigma protocol with helper). *A protocol is a* sigma protocol with helper *for a relation $\mathcal{R}$ with challenge space $\mathcal{C}$ if it is of the form of Fig. 1 and satisfies the following properties:*

- ***Completeness:*** *If all parties follow the protocol on input $(x, w) \in \mathcal{R}$, then the verifier always accepts.*
- ***2-Special soundness:*** *From an adversary that outputs with significant probability two valid transcripts $(x, \mathsf{aux}, \mathsf{com}, \mathsf{ch}, \mathsf{rsp})$ and $(x, \mathsf{aux}, \mathsf{com}, \mathsf{ch}', \mathsf{rsp}')$, with $\mathsf{ch} \neq \mathsf{ch}'$ and where $\mathsf{aux} = \mathsf{Setup}(\mathsf{seed})$ for some seed values (unknown to the extractor), one can efficiently extract a witness w such that $(x, w) \in \mathcal{R}$.*
- ***Special honest-verifier zero-knowledge:*** *There exists a probabilistic polynomial-time simulator that takes as input x, a random seed, and a random challenge ch; and outputs a transcript $(x, \mathsf{aux}, \mathsf{com}, \mathsf{ch}, \mathsf{rsp})$, with*

$\mathsf{aux} = \mathsf{Setup}(\mathsf{seed})$, *that is computationally indistinguishable from the probability distribution of transcripts of an honest execution of the protocol on input* (x, w) *for some* w *such that* $(x, w) \in \mathcal{R}$*, conditioned on the auxiliary information being equal to* aux *and the challenge being equal to* ch*.*

Every sigma protocol with helper can be transformed into a standard sigma protocol by a *cut-and-choose* approach, which we outline in Section 4.2.

2.2 Commitment schemes

In our constructions we assume the existence of a non-interactive commitment function $\mathsf{Com} : \{0,1\}^\lambda \times \{0,1\}^* \to \{0,1\}^{2\lambda}$, which takes as input a pair (r, m) consisting of λ random bits r and an arbitrary message m, and returns a commitment of 2λ bits. The function Com is assumed to be *computational hiding*, which informally means that the commitments do not reveal anything about the committed message, and *computational binding*, which informally states that it should not be possible to find a different message m' that leads to the same commitment. The formal definitions of these properties are given in Appendix A. In practice, Com can be implemented by a cryptographic secure hash function.

3 The sigma protocol of Courtois

Let us briefly recall the sigma protocol of Courtois [23] for a zero-knowledge proof of knowledge of the solution α to an instance of the MinRank problem. It follows a three-pass design with challenge space $\{0, 1, 2\}$ and achieves a soundness error of $\frac{2}{3}$. The protocol is based on an additive masking of the solution vector α with some random vector β. Initially, the prover commits to the matrices

$$N_1 = \sum_{i=1}^{k} \beta_i M_i \quad \text{and} \quad N_2 = M_0 + \sum_{i=1}^{k} (\alpha_i + \beta_i) M_i = N_1 + E,$$

where E is the matrix of rank less than r from the underlying MinRank problem. Then challenges 1 and 2 lead to revealing either β or $\alpha + \beta$, which enables the verifier to check that the prover followed the protocol for the computation of either N_1 or N_2. In case of challenge equal to 0, the prover sends to the verifier two matrices Z_1 and Z_2, which are obtained by multiplicatively and additively masking N_1 and N_2 with the matrices S, T and X. Then the verifier checks that $\mathrm{rank}(Z_2 - Z_1) \leq r$, which implies that $\mathrm{rank}(N_2 - N_1) \leq r$, i.e., that α is a solution to the MinRank problem. See Fig. 2 for a formal description of the protocol.

4 Improved MinRank-based signature scheme

In this section, we present an improved signature scheme based on the MinRank problem. The scheme is constructed in three steps. First, in Section 4.1, we give

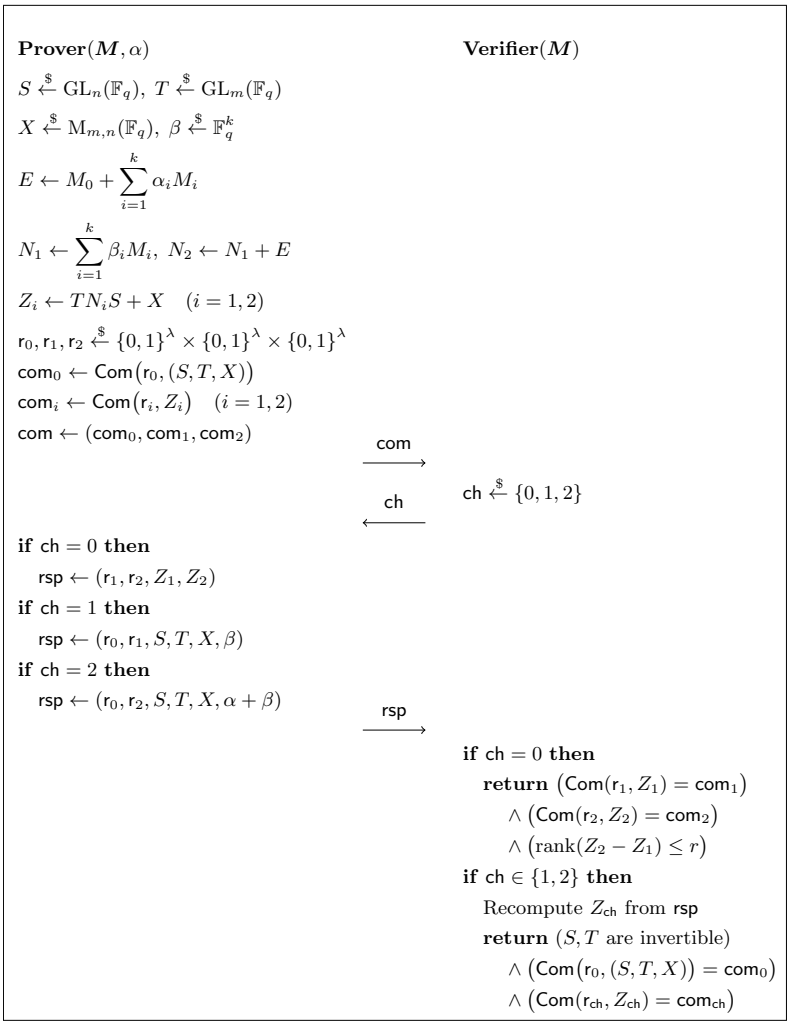

Fig. 2: The sigma protocol of Courtois for ZK proof of MinRank.

a sigma protocol with helper for a zero-knowledge proof of knowledge of the solution to an instance of the MinRank problem. Second, the helper is removed using a cut-and-choose technique, detailed in Section 4.2, to obtain a standard sigma protocol. Eventually, the sigma protocol is converted to a signature scheme by a standard application of the Fiat–Shamir transform.

Furthermore, in Section 4.3 we give several improvements to our initial design, reducing the communication complexity and, implicitly, the signature size.

4.1 Sigma protocol with helper for ZK proof of MinRank

On a high level, we use the helper as a trusted third party to provide the *non*-secret-key dependent commitments. This allows us to decrease the challenge space to $\{0,1\}$, since the prover has to commit only to a single value. For challenge 1 the prover then allows the verifier to check that he followed the protocol when computing the commitment, while for challenge 0 he proves his knowledge of the solution to the MinRank problem.

More precisely, the helper provides the non-secret-key dependent commitments com_0 and com_1 of Fig. 2. He then sends those commitments to the verifier and the used randomness to the prover. The prover only needs to provide the key-dependent commitment com_2, after recomputing the helper-generated data from the given randomness seed. The challenge space reduces to $\{0,1\}$, since the helper-provided commitment does not have to be challenged. Analogously to before, in case of challenge 1 the prover reveals the masked secret $\alpha+\beta$, allowing the re-computation of N_2, while for challenge 0 he answers with Z_1 and Z_2 allowing to verify his knowledge of the MinRank solution. Our full protocol is detailed in Fig. 3.

Theorem 1 (MinRank with Helper). *Let* Com *be a commitment scheme which is computational binding and hiding. Then the protocol detailed in Fig. 3 satisfies Definition 2 for sigma protocols with helper for challenge space* $\mathcal{C}=\{0,1\}$.

Proof. We have to prove that the protocol of Fig. 3 fulfills the notions of completeness, 2-special soundness, and special honest-verifier zero-knowledge given in Definition 2. In the following, we let $\mathsf{rsp}=(\mathsf{r}_0,\mathsf{r}_2,Y,\gamma)$ denote the response for challenge $\mathsf{ch}=1$. In particular, an honest prover sends $Y:=(Y_1,Y_2,Y_3)=(S,T,X)$ and $\gamma=\alpha+\beta$.

Completeness. If all parties follow the protocol, then it is clear that the verifier accepts, since

$$\operatorname{rank}(Z_2-Z_1)=\operatorname{rank}\big(T(N_2-N_1)S\big)=\operatorname{rank}(N_2-N_1)=\operatorname{rank}(E)\leq r.$$

2-Special soundness. Suppose that an adversary knows two valid transcripts

$$(\boldsymbol{M},\mathsf{aux},\mathsf{com},\mathsf{ch},\mathsf{rsp}) \quad \text{and} \quad (\boldsymbol{M},\mathsf{aux},\mathsf{com},\mathsf{ch}',\mathsf{rsp}')$$

with $\mathsf{ch}\neq\mathsf{ch}'$ and where $\mathsf{aux}=\mathsf{Setup}(\mathsf{seed})$ for some value of seed, which is unknown to the adversary. We have to prove that the adversary can efficiently compute a solution to $\boldsymbol{M}$.

Without loss of generality, assume that $\mathsf{ch}=1$ and $\mathsf{ch}'=0$. Since the verifier accepts the response rsp, we have that $\mathsf{com}_0=\mathsf{Com}(\mathsf{r}_0,Y)$ and

$$\mathsf{com}_2=\mathsf{Com}\big(\mathsf{r},Y_2(M_0+\textstyle\sum_{i=1}^{k}\gamma_iM_i)Y_1+Y_3\big).$$

From the computational binding property of the commitment we now conclude that $Y=(S,T,X)$ since $\mathsf{Com}(\mathsf{r}_0,Y)=\mathsf{com}_0=\mathsf{Com}\big(\mathsf{r}_0,(S,T,X)\big)$.

Moreover, from the verifier accepting the response rsp', we know that $\mathsf{com}_2 = \mathsf{Com}(\mathsf{r}, Z_2')$. Thus, we find analogously that

$$\mathsf{Com}\big(\mathsf{r}, T(M_0 + \textstyle\sum_{i=1}^k \gamma_i M_i)S + X\big) = \mathsf{com}_2 = \mathsf{Com}(\mathsf{r}, Z_2'),$$

implying $Z_2' = T(M_0 + \sum_{i=1}^k \gamma_i M_i)S + X$. Further, by the helper behaving honestly we know that $\mathsf{com}_1 = \mathsf{Com}(r_1, T\sum_i^k \beta_i M_i S + X)$ while the verifier only accepts rsp' if $\mathsf{com}_1 = \mathsf{Com}(\mathsf{r}_1, Z_1')$, giving $Z_1' = T(\sum_{i=1}^k \beta_i M_i)S + X$.

In turn, this gives that

$$\begin{aligned} Z_2' - Z_1' &= \big(T(M_0 + \textstyle\sum_{i=1}^k \gamma_i M_i)S + X\big) - \big(T(\sum_{i=1}^k \beta_i M_i)S + X\big) \\ &= T\big(M_0 + \textstyle\sum_{i=1}^k (\gamma_i - \beta_i) M_i\big)S = T\big(M_0 + \sum_{i=1}^k \delta_i M_i\big)S, \end{aligned}$$

where $\delta := \gamma - \beta \in \mathbb{F}_q^k$. Since $Y = (S, T, X)$ is known, we can compute δ by solving the linear system

$$\textstyle\sum_{i=1}^k \delta_i M_i = Y_2^{-1}(Z_2' - Z_1')Y_1^{-1} - M_0.$$

Finally, from the verifier accepting rsp' we know that $\operatorname{rank}(Z_2' - Z_1') \le r$ which implies

$$\operatorname{rank}\big(M_0 + \textstyle\sum_{i=1}^k \delta_i M_i\big) = \operatorname{rank}\Big(T\big(M_0 + \sum_{i=1}^k \delta_i M_i\big)S\Big) = \operatorname{rank}(Z_2' - Z_1') \le r,$$

thus δ is a solution to the instance $\boldsymbol{M}$.

Special honest-verifier zero-knowledge. Define a simulator that takes as input $\boldsymbol{M}$, a random seed seed, and a random challenge ch; and outputs a valid transcript $(\boldsymbol{M}, \mathsf{aux}, \widetilde{\mathsf{com}}_2, \mathsf{ch}, \widetilde{\mathsf{rsp}})$ computed as follows:

1. Generate $S, T, X, \beta, N_1, \mathsf{r}_0, \mathsf{r}_1, \mathsf{com}_0, \mathsf{com}_1, \mathsf{aux}$ from seed as a honest helper would do.
2. If $\mathsf{ch} = 1$ then pick a random $\mathsf{r} \in \{0,1\}^\lambda$ and a random $\widetilde{\gamma} \in \mathbb{F}_q^k$, and set $\widetilde{N}_2 = M_0 + \sum_{i=1}^k \widetilde{\gamma}_i M_i$, $\widetilde{\mathsf{com}}_2 = \mathsf{Com}(\mathsf{r}, T\widetilde{N}_2 S + X)$, and $\widetilde{\mathsf{rsp}} = \big(\mathsf{r}, \mathsf{r}_0, (S, T, X), \widetilde{\gamma}\big)$.
3. If $\mathsf{ch} = 0$ then pick a random $\mathsf{r} \in \{0,1\}^\lambda$ and a random $R \in \mathrm{M}_{m,n}(\mathbb{F}_q)$ of rank r, and set $\widetilde{Z}_2 = T(N_1 + R)S + X$, $\widetilde{\mathsf{com}}_2 = \mathsf{Com}(\mathsf{r}, \widetilde{Z}_2)$, and $\widetilde{\mathsf{rsp}} = (\mathsf{r}, \mathsf{r}_1, TN_1S + X, \widetilde{Z}_2)$.

Then, by construction, $(\boldsymbol{M}, \mathsf{aux}, \widetilde{\mathsf{com}}_2, \mathsf{ch}, \widetilde{\mathsf{rsp}})$ is a valid transcript where $\widetilde{\mathsf{rsp}}$ is uniformly distributed in

$$\begin{aligned} &\{0,1\}^\lambda \times \{0,1\}^\lambda \times \mathrm{GL}_n(\mathbb{F}_q) \times \mathrm{GL}_m(\mathbb{F}_q) \times \mathrm{M}_{m,n}(\mathbb{F}_q) \times \mathbb{F}_q^k \\ \cup\ &\{0,1\}^\lambda \times \{0,1\}^\lambda \times \{(U,V) : U, V \in \mathrm{M}_{m,n}(\mathbb{F}_q),\ \operatorname{rank}(U - V) = r\}. \end{aligned}$$

Since also rsp is uniformly distributed in the above set, we get that rsp and $\widetilde{\mathsf{rsp}}$ follow the same distribution. Moreover, $\widetilde{\mathsf{com}}_2$ is completely determined by $\widetilde{\mathsf{rsp}}$ in the same way that com_2 is completely determined by rsp. This implies that

the transcripts $(\boldsymbol{M}, \mathsf{aux}, \widetilde{\mathsf{com}}_2, \mathsf{ch}, \mathsf{rsp})$ and $(\boldsymbol{M}, \mathsf{aux}, \mathsf{com}, \mathsf{ch}, \mathsf{rsp})$ follow the same distribution.

Eventually, since the commitment $\mathsf{com}_{1-\mathsf{ch}}$ is never opened, by the computational hiding property of the commitment the transcripts are indistinguishable. □

4.2 Removing the Helper

In order to remove the helper, we use a cut-and-choose technique of Katz et al. [34] that proceeds as follows. The prover computes several setups and sends all generated auxiliary information to the verifier. The verifier then chooses a certain amount of the setups to execute, i.e., run the normal protocol based on the seeds of the chosen setups. Additionally, the prover sends all seeds belonging to the setups that are not executed to the verifier, allowing him to check that those setups have been computed honestly.

More precisely, we let the prover compute s setups and the verifier has to choose a subset of τ setups to execute. We illustrate this procedure schematically in Fig. 4. Now, if the soundness error of a single execution of the protocol with helper is p then the soundness error of the whole construction becomes

$$p_\tau := \max_{0 \le i \le \tau} \frac{\binom{s-i}{\tau-i} p^{\tau-i}}{\binom{s}{\tau}}. \tag{1}$$

Therefore assume that the prover computed a total of i setups dishonestly to provide valid responses in the online phase. Since the prover discloses all seeds belonging to setups that have not been executed, the cheating can only be hidden from the verifier if all these i setups are executed. The probability for this to happen is $\binom{s-i}{\tau-i}/\binom{s}{\tau}$. Now, the prover still needs to provide valid responses for the $\tau - i$ executed and honestly computed setups, in which he succeeds with probability $p^{\tau-i}$. For a more formal proof we refer to [10].

4.3 Further improvements

In order to reduce the communication complexity of the sigma protocol of Fig. 4, we apply various improvements outlined in the following.

Merkle-tree. First, we combine the $\mathsf{com}_{2,i}$ in a Merkle-tree, with $\mathsf{com}_{2,i}$ being the i-th leaf of the tree, where we label the root of the tree ρ. Then instead of sending $\mathsf{com}_{2,i}$, $\forall i$, we only send ρ as a commitment. Later we then provide missing nodes of the tree to the verifier to be able to recompute the root ρ.

Seed-tree. Similarly, we optimize the transmission of the seeds by using a seed-tree that expands an initial root into two seeds via a hash function. From there every node of the tree is expanded in a similar fashion, until the tree reaches a depth of $\lceil \log s \rceil$, i.e., it contains at least s leaves. Now we declare seed_i to be the i-th leaf of that tree. The transmission of the seeds then only requires to reveal a certain (fewer) number of nodes of the tree.

Single initial commitment. Instead of initially sending the root of the Merkle-tree ρ together with all aux_i's as commitment, we just send a single commitment $\mathsf{com} := \mathsf{Com}(\rho, \mathsf{aux}_1, \ldots, \mathsf{aux}_n)$. Later we then provide the missing inputs similar to the missing nodes of the Merkle-tree so that the verifier can recompute com.

Sending a rank-r matrix. In the case of challenge equal to 0 instead of sending Z_1, Z_2 as response, we send $Z_1, Z_1 - Z_2$. Note, that the response still carries the same information. However, the benefit lies in $Z_1 - Z_2$ being a rank-r matrix, which for small r has a shorter description length. Precisely, we can write $Z_1 - Z_2 = XY$, where X is an $m \times r$ matrix and Y an $r \times n$ matrix. Hence, instead of sending the entries of Z_2, we can send the entries of X and Y, which requires to transmit $(m+n)r \log q$ bits instead of $mn \log q$ bits, i.e., we obtain an improvement as long as $r < mn/(m+n)$.

4.4 Public key size

The public key of the scheme is the MinRank instance $\boldsymbol{M}$. Courtois [23] generates $M_0, \ldots, M_{k-1}$ from an initial seed and chooses M_k such that there exists a solution to the MinRank instance. Precisely, for a random matrix $E \in \mathrm{M}_{m,n}(\mathbb{F}_q)$ of rank r and a random secret key $\alpha \in \mathbb{F}_q^k$ with $\alpha_k \neq 0$, he lets

$$M_k := \alpha_k^{-1} \left(-E + M_0 + \textstyle\sum_{i=1}^{k-1} \alpha_i M_i \right).$$

The public key then consists of the seed and M_k and, thus, has a size of $\lambda + mn \log q$ bits. We improve on this by showing that any generic MinRank instance can be transformed into a canonical form which yields a shorter description length for its matrices.

More precisely, let L be the $(k+1) \times mn$ matrix whose i-th row consists of the entries of M_i in row-major order, for $i = 1, \ldots, k$ and whose $(k+1)$-th row is formed by the entries of M_0. Now, row operations on L correspond to linear transformations of the variables α_i, i.e., we can apply elementary row operations without affecting the existence of a solution. Hence, we assume

$$L = \left(\begin{array}{c|c} \boldsymbol{I} & \\ 0 \ldots 0 & L' \end{array} \right), \tag{2}$$

where $\boldsymbol{I}$ is the $k \times k$ identity matrix. Here, we restrict to keys where the first k columns and k rows of L form a matrix of full rank. However, since we consider random instances this is a constant fraction of the whole keyspace.

The public key is now generated as follows. First, from a random seed of λ bits, generate the first k rows of L', from which the matrices $M_1, \ldots, M_k$ can be derived following Eq. (2). Then generate a random $m \times n$ matrix E of rank r, a random $\beta \in \mathbb{F}_q^k$, and compute $F := E - \sum_{i=1}^{k} \beta_i M_i$. Finally, let $f_1, \ldots, f_k$ be the first k entries of F in row-major order, and let $M_0 := F - \sum_{i=1}^{k} f_i M_i$ and $\alpha_i := \beta_i + f_i$ for $i = 1, \ldots, k$. This ensures that the last row of L starts with k zeros. The compressed public key now consists of the seed and the last $mn - k$ entries of M_0 (the first k entries are all zero) and so its size is $\lambda + (mn - k) \log q$ bits.

4.5 Signature size

The signature size after the Fiat–Shamir transform is determined by the communication size of messages send from the prover to the verifier. For our improved version of the protocol (see Section 4.3) this communication includes:

1. initial commitment of size 2λ,
2. missing nodes of the Merkle-tree to compute ρ,
3. seed values seed_i for $i \notin I$,
4. missing auxiliary information aux_i to compute $\mathsf{com} := \mathsf{Com}(\rho, \mathsf{aux}_1, \ldots, \mathsf{aux}_n)$,
5. responses rsp_i for $i \in I$.

In the online phase the verifier can compute all τ values $\mathsf{com}_{2,i}$ for $i \in I$. Hence, due to the usage of a Merkle-tree, the prover needs to send at most $\lceil \tau \log \frac{s}{\tau} \rceil$ tree-nodes, each of size 2λ, to allow the verifier the re-computation of the root ρ. Similarly, the usage of the seed-tree requires the prover to reveal at most $\lceil \tau \log \frac{s}{\tau} \rceil$ nodes of the tree, each of size λ, to enable the verifier to recompute all $s - \tau$ seeds seed_i for $i \notin I$.

These seeds allow to compute $\mathsf{aux}_i := (\mathsf{com}_{0,i}, \mathsf{com}_{1,i})$ for $i \notin I$. Further in the online phase the verifier can compute one of either $\mathsf{com}_{0,i}$ or $\mathsf{com}_{1,i}$ for $i \in I$. In order to finally re-compute com the verifier, now, misses τ values $\mathsf{com}_{j,i}$, not obtained in the online phase, which have to be provided by the prover, corresponding to $\tau \cdot 2\lambda$ bits of communication.

Eventually the average size of each of the τ responses is

$$|\mathsf{rsp}| = \underbrace{\frac{\big(mn + r(m+n)\big)\log q}{2}}_{\mathsf{ch}=0} + \underbrace{\frac{\lambda + k \log q}{2}}_{\mathsf{ch}=1}.$$

Indeed, in the case of $\mathsf{ch} = 0$ the response is composed of one $m \times n$ matrix and one rank-r $(m \times n)$-matrix over $\mathbb{F}_q$; while in the case of $\mathsf{ch} = 1$ it consists of the seed used to derive the matrices (S, T, X) and a vector of length k over $\mathbb{F}_q$.

In total we find a communication complexity of

$$C := \underbrace{2\lambda}_{1)} + \underbrace{3\lambda \left\lceil \tau \log \frac{s}{\tau} \right\rceil}_{2)\ +\ 3)} + \underbrace{\tau \cdot 2\lambda}_{4)} + \underbrace{\tau \cdot |\mathsf{rsp}|}_{5)}, \tag{3}$$

while the soundness of the protocol is p_τ detailed in Eq. (1).

4.6 Parameters

In this section, we propose parameters for our signature scheme targeting NIST's security categories I, III, and V and detail the corresponding signature and public key sizes.

We estimate the security of our parameters by using the recent hybrid-MinRank approach from [6]. Given $0 \leq a \leq n$, this hybrid-MinRank approach

Category	λ	q	m	n	k	r	KS (a)	SM(a)	big-k(a)
I	128	16	16	16	142	4	158(8)	160(8)	223(0)
III	192	16	19	19	167	6	231(8)	234(8)	343(0)
V	256	16	22	22	254	6	303(11)	295(11)	416(0)

Table 1: Estimated bit-security of proposed parameter sets using $\omega = 2$.

reduces the cost of solving a rank-r MinRank problem with K matrices in $\mathrm{M}_{m,n}(\mathbb{F}_q)$ to the cost of solving q^{ar} smaller instances with only $K - am$ matrices in $\mathrm{M}_{m,n-a}(\mathbb{F}_q)$ and rank r. The complexity of the smaller instances is estimated by using the kernel-search algorithm [31], the Support-Minors modeling [4], and the big-k algorithm [24][3]. Notice that we do not consider the Kipnis-Shamir [35], and Minors [27] modelings, since it was recently proven that these modelings are less efficient than Support-Minors [5].

The complexity of the aforementioned algorithms depends on the linear algebra constant $2 \leq \omega \leq 3$, where the complexity of multiplying two $n \times n$ matrices is $O(n^\omega)$. All our bit security estimates are done for the conservative choice of $\omega = 2$. Also, we assume that multiplying two elements in $\mathbb{F}_q$ costs $(\log q)^2$ bit operations. Table 1 states the parameter sets for our scheme targeting the different security categories. The column KS contains the complexity of the kernel-search algorithm, while SM indicates the complexity of the Support-Minors modeling. The value of a inside the parenthesis shows the hybridization parameter of the hybrid-MinRank approach from [6].

Avoiding random solutions. Further, it is known that a set of k' randomly chosen matrices in $\mathrm{M}_{m,n}(\mathbb{F}_q)$, in expectation, does not to span a rank r matrix when $k' < (m-r)(n-r)$ [24, Sec. 24.2]. Hence we enforce $k + 1 < (m-r)(n-r)$ in order to avoid random solutions to the underlying MinRank problem.

Table 2 gives the signature and public key sizes obtained for the proposed parameters. We compare our scheme to the original scheme by Courtois. Here, we obtain an optimal signature size of our scheme for cut-and-choose parameters $\tau = \lambda$ and $s = 2\lambda$ (compare to Section 4.2). For this choice, using Merkle- and seed-trees (as described in Section 4.3) yields signature size improvements (only) on average.

Nevertheless, our scheme improves significantly on Courtois' design. In terms of signature size we, e.g., obtain a reduction by a factor of 2.4 for category I, while achieving a public key reduction by 1.97 using the improvement described in Section 4.4.

Note that the nature of the MinRank problem involves the transmission of matrices between the corresponding parties, which leads in general to larger signatures compared to schemes that only involve vector exchanges. Nevertheless,

[3] The big-k algorithms is called big-m in [24].

Category	Signature (KB)		Public key (B)	
	Courtois	MR-DSS	Courtois	MR-DSS
I	65	27	144	73
III	135	60	205	121
V	248	106	274	147

Table 2: Signature sizes (in kilobytes) and public key sizes (in bytes) for suggested parameters of our new scheme in comparison to Courtois' scheme. The signature size of our scheme is computed by setting $\tau = \lambda$ and $s = 2\lambda$ in Eq. (3)

the signature size of our construction gets close to being competitive to other NIST PQC candidates that are not based on structured problems. As for example to those of SPHINCS$^+$, which achieves roughly 17kB signatures for category I.

5 MinRank-based ring signatures

In the following we formalize the extension of our MinRank-based signature scheme to ring-signatures. We follow the formalism and the security definitions for ring signatures given by Bender, Katz, and Morselli [13]. We refer as a *ring* (of users) to a list of public keys $\mathsf{R} = (\mathsf{pk}_1, \ldots, \mathsf{pk}_u)$. The formal definition of a ring signature scheme is given in Appendix B. An essential property of a ring signature scheme is that no coordination between the potential users of the scheme is needed. First, anyone can generate keys independently using Gen. Second, at the time of signing a message msg, a particular user holding a secret key sk uses its own public key along with any set of $u-1$ public keys from other users to create a ring R and computes $\sigma \leftarrow \mathsf{Sign}_{\mathsf{sk}}(\mathsf{msg}, \mathsf{R})$. Anyone knowing R can verify the signature σ of the message msg, and guarantee that msg was signed by someone holding a secret key with corresponding public key in R. In the following we refer to the holder of sk as the *signer*.

A desired property of a ring signature scheme is to preserve the *anonymity* of the signer, i.e., informally speaking, the verifier can not identify the signer among all members of R. Another fundamental security property is the *unforgeability* for fixed rings. Roughly speaking, for a given ring R, without knowing any of the secret keys corresponding to public keys in R, an adversary is not able to produce a valid signature. Formal definitions of those security properties are given in Appendix B.1

5.1 Extending to ring signatures

Let us briefly outline the idea of how to extend our signature scheme to a ring-signature scheme. The public key of each user is a matrix R, while the instance $\boldsymbol{M}$ (the public key of our regular signature scheme) is now a public parameter of the ring-signature scheme. Each user crafts R, such that he knows a linear

combination of the M_i's that added to R yields a low-rank matrix, i.e., he knows a solution to the instance $(\boldsymbol{M}, R)$, which defines his secret key. Recall that a ring is defined as u public keys $\mathsf{R} := (R_1, R_2, \ldots, R_u)$. A ring signature is obtained by invoking the signing function of our regular scheme with $(\boldsymbol{M}, \mathsf{R})$ as public key and the known solution as private-key.

Formal definition of the scheme In the following we let MR-Sign and MR-Verify denote the signing and verification function of our signature scheme outlined in Section 4. Further, let $\boldsymbol{M} := (M_0, M_1, \ldots, M_k) \in \left(M_{m,n}(\mathbb{F}_q)\right)^{k+1}$ be a public parameter of the scheme (generated from some public $\mathsf{Initseed} \in \{0,1\}^\lambda$). In the following R is the public-key corresponding to secret-key α and the ring is $\mathsf{R} = (R_1, \ldots, R_u)$.

$\mathsf{Gen}(1^\lambda)$:

1. Choose random secret key $\alpha := (\alpha_1, \ldots, \alpha_k) \in \mathbb{F}_q^k$,
2. Set the public key to $R = -\left(M_0 + \sum_{i=1}^k \alpha_i M_i\right) + E$, where $E \in \mathbb{F}_q^{m \times n}$ is a randomly chosen rank r matrix.
3. Output (R, α)

$\mathsf{Sign}_\alpha(\mathsf{msg}, \mathsf{R})$:

1. Set $\gamma \leftarrow (\alpha, \boldsymbol{\varepsilon}_j)$, where $\boldsymbol{\varepsilon}_j \in \mathbb{F}_q^u$ denotes the j-th canonical vector.
2. Output $\mathsf{MR\text{-}Sign}_\gamma(\mathsf{msg})$

$\mathsf{Verify}_\mathsf{R}(\mathsf{msg}, \sigma)$:

1. Output $\mathsf{MR\text{-}Verify}_{\widetilde{\boldsymbol{M}}}(\mathsf{msg}, \sigma)$, where $\widetilde{\boldsymbol{M}} := (\boldsymbol{M}, \mathsf{R})$

The proof of correctness as well as the proofs of our scheme fulfilling the security notions of anonymity with regard to adversarially-chosen keys and unforgeability against fixed rings is given in Appendix B.2

5.2 Parameters of the scheme

Next we derive parameter sets for our constructed ring signature scheme. Therefore, we need to make some observations on the security of the constructed instances. Let us start with a remark on the amount of users a certain parameter set can support.

Limitation on the number of users A given parameter set for our MinRank-based ring signature scheme can not afford an unlimited number of users. This is because for a ring of size u we can forge a signature by solving a MinRank instance with $u + k + 1$ matrices in $\mathrm{M}_{m,n}(\mathbb{F}_q)$. Such an instance turns easy if $u+k+1$ is big enough. By using the big-k algorithm [23] one solves any MinRank problem with parameters (m, n, k', r) in polynomial time $\mathrm{Poly}(m, n, k')$ as long

# users	8	16	32	64	128	256	512	1024	4096
q	16	16	16	16	16	16	16	16	16
m	16	16	18	20	23	29	36	46	81
n	16	16	18	20	23	29	36	46	81
k	102	102	102	124	158	216	320	340	560
r	5	5	6	6	6	7	7	9	12
bit-security	144	144	154	164	174	205	198	190	145

Table 3: Suggested parameters for our ring signature an their estimated bit-security.

as $k' \geq m(n-r)$. Hence in both cases, i.e., for the one-user and the ring version of our scheme we make sure that

$$k' < m(n-r). \tag{4}$$

Still, in the case $k' < m(n-r)$, the attacker can succeed with probability $q^{k'-m(n-r)}$. Hence, the complexity of the algorithm becomes

$$q^{m(n-r)-k'} \cdot \mathrm{Poly}(m, n, k').$$

We take this attack into account when deriving parameters. Further, we enforce $u + k + 1 \leq (m-r)(n-r)$ in order to avoid random solutions to the underlying MinRank problem.

Attack scenarios To forge a signature for a given ring $\mathsf{R} := (R_1, R_2, \ldots, R_u)$ one has to solve an instance of the rank-r MinRank problem with matrices $(\boldsymbol{M}, \mathsf{R}) \subset \mathrm{M}_{m,n}(\mathbb{F}_q)$, where $\boldsymbol{M} := (M_0; M_1, \ldots, M_k)$ is the fixed set of matrices of the scheme. We consider two attack scenarios. First, due to the construction of our ring signature, one can fix to zero the coefficients of all but one matrix in R and still the remaining problem has a solution. That is, for any $1 \leq i \leq u$, the rank-r MinRank problem defined on the $k+1$ matrices $M_0 + R_i, M_1, \ldots, M_k$ has a solution. Finding this solution corresponds to solving a MinRank instance with parameters $(q, m, n, k+1, r)$. In the second scenario the attacker aims at finding a solution to the instance $M_0, M_1, \ldots, M_k, R_1, \ldots, R_i$, for $2 \leq i \leq u$ which has i solutions. We then take the minimum time complexity obtained in both scenarios to derive the bit complexity.

Table 3 shows a list of parameters for our ring signature achieving NIST category I security.

5.3 Public key and signature size

Suppose we have a ring with u users. The public key size for a ring of u users is given by

$$\lambda + u \cdot mn \log q.$$

#users	2^3	2^4	2^5	2^6	2^7	2^8	2^{10}	2^{12}	**Assumption**	**Security**
MRr-DSS	27	27	32	36	45	64	145	422	MinRank	Cat. I
KKW [34]	-	-	-	250	-	-	-	456	LowMC	Cat. V
Raptor [37]	10	-	-	81	-	333	1290	5161	MSIS / MLWE	100 bit
EZSLL [26]	19	-	-	31	-	-	-	148	MSIS / MLWE	Cat. II
Falafl [15]	30	-	-	32	-	-	-	35	MSIS / MLWE	Cat. I
Calamari [15]	5	-	-	8	-	-	-	14	CSIDH	128 bit (60bit)
LESS [8]	11	-	-	14	-	-	-	20	Code Equiv.	128 bit

Table 4: Ring signature size (in kilobytes) of our ring signature in comparison to recent proposals.

This means that the public key size is linear in the number of users u.

The signature size is given by $f(m, n, k+u, r, q)$, where $f(m, n, k, r, q)$ denotes the signature size with one user and parameters (m, n, k, r, q). Asymptotically we find

$$\frac{f(m,n,k+u,r,q)}{f(m,n,k,r,q)} = \mathcal{O}\left(\frac{\lambda/\log q + mn + k + u}{\lambda/\log q + mn + k}\right) = \mathcal{O}\left(\frac{mn+k+u}{mn+k}\right),$$

assuming that $\frac{\lambda}{\log q} = \mathcal{O}(mn)$. Since we know from Eq. (4) that $k < mn$ we achieve a signature size that scales with the number of users u roughly as $\mathcal{O}\left(\frac{u}{mn}\right)$.Note that as long as mn is a function in u that tends to infinity for growing u, this corresponds to a sublinear scaling. Moreover, for practical parameters the large denominator allows us to achieve a competitive signature size for low to moderate amounts of users.

Table 4 states the signature sizes of our ring signature *MRr-DSS* achieved for different amounts of ring sizes using the parameters detailed in Table 3. We compare our parameters to various recent developments. Note that parameters for NIST category I are not available for all designs, so we also indicate the achieved security level.

The most compact ring signatures are obtained by the *Calamari* construction of Beullens, Katsumata, and Pintore [15], which follows a group-action-based construction similar to classical discrete logarithm based schemes. However, there is some doubt about the quantum security of its hardness assumption. Moreover, the chosen parameters offer at most 60 bits of quantum security employing NIST metrics [43]. Recently, Barenghi et al. [8] adapted the same idea but instantiated the group action via the code equivalence problem. However, despite recent efforts [9, 16] motivated by cryptographic constructions [7, 8, 18], the code equivalence problem has not yet reached the same level of cryptanalytic maturity as the MinRank problem.

Apart from group action based constructions, for a large number of users the *Falafl* scheme [15] yields the best signature size, due to its logarithmic dependence on the ring size.

However, for low to moderate amounts of users ($\leq 2^7$) our scheme yields competitive performance. Even though some of the considered schemes might achieve (slightly) lower signature sizes in this regime, those are all based on structured lattice-based assumptions. Our scheme yields a solid alternative to this trend by being based on the hardness of random instances of a non-structured problem.

References

1. Abdalla, M., An, J.H., Bellare, M., Namprempre, C.: From identification to signatures via the Fiat-Shamir transform: minimizing assumptions for security and forward-security. In: Advances in cryptology—EUROCRYPT 2002, vol. 2332, pp. 418–433 (2002)
2. Au, M.H., Liu, J.K., Susilo, W., Yuen, T.H.: Secure ID-based linkable and revocable-iff-linked ring signature with constant-size construction. Theoret. Comput. Sci. **469**, 1–14 (2013)
3. Bard, G.V.: Accelerating cryptanalysis with the method of four Russians. Cryptology ePrint Archive (2006)
4. Bardet, M., Bros, M., Cabarcas, D., Gaborit, P., Perlner, R., Smith-Tone, D., Tillich, J.P., Verbel, J.: Improvements of algebraic attacks for solving the rank decoding and MinRank problems. In: Advances in cryptology—ASIACRYPT 2020. Part I, vol. 12491, pp. 507–536 (2020)
5. Bardet, M., Bertin, M.: Improvement of algebraic attacks for solving superdetermined MinRank instances. CoRR **abs/2208.01442** (2022). https://doi.org/10.48550/arXiv.2208.01442
6. Bardet, M., Briaud, P., Bros, M., Gaborit, P., Tillich, J.P.: Revisiting algebraic attacks on MinRank and on the rank decoding problem. Cryptology ePrint Archive, Paper 2022/1031 (2022), https://eprint.iacr.org/2022/1031
7. Barenghi, A., Biasse, J.F., Persichetti, E., Santini, P.: LESS-FM: fine-tuning signatures from the code equivalence problem. In: Post-quantum cryptography, Lecture Notes in Comput. Sci., vol. 12841, pp. 23–43. Springer, Cham ([2021] ©2021). https://doi.org/10.1007/978-3-030-81293-5_2, https://doi.org/10.1007/978-3-030-81293-5_2
8. Barenghi, A., Biasse, J.F., Ngo, T., Persichetti, E., Santini, P.: Advanced signature functionalities from the code equivalence problem. International Journal of Computer Mathematics: Computer Systems Theory **7**(2), 112–128 (2022)
9. Barenghi, A., Biasse, J.F., Persichetti, E., Santini, P.: On the computational hardness of the code equivalence problem in cryptography. Cryptology ePrint Archive (2022)
10. Baum, C., Nof, A.: Concretely-efficient zero-knowledge arguments for arithmetic circuits and their application to lattice-based cryptography. In: Public-key cryptography—PKC 2020. Part I, vol. 12110, pp. 495–526 (2020)
11. Bellini, E., Caullery, F., Gaborit, P., Manzano, M., Mateu, V.: Improved veron identification and signature schemes in the rank metric. In: IEEE International Symposium on Information Theory. pp. 1872–1876 (2019)

12. Bellini, E., Gaborit, P., Hasikos, A., Mateu, V.: Enhancing code based zero-knowledge proofs using rank metric. In: Cryptology and Network Security - 19th International Conference, CANS. vol. 12579, pp. 570–592 (2020)
13. Bender, A., Katz, J., Morselli, R.: Ring signatures: stronger definitions, and constructions without random oracles. J. Cryptology **22**(1), 114–138 (2009)
14. Beullens, W.: Improved cryptanalysis of UOV and Rainbow. In: Advances in cryptology—EUROCRYPT 2021. Part I, vol. 12696, pp. 348–373 (2021)
15. Beullens, W., Katsumata, S., Pintore, F.: Calamari and Falafl: logarithmic (linkable) ring signatures from isogenies and lattices. In: Advances in cryptology—ASIACRYPT 2020. Part II, vol. 12492, pp. 464–492 (2020)
16. Beullens, W.: Not enough LESS: An improved algorithm for solving code equivalence problems over $\mathbb{F}_q$. In: International Conference on Selected Areas in Cryptography. pp. 387–403. Springer (2020)
17. Beullens, W.: Sigma protocols for MQ, PKP and SIS, and fishy signature schemes. In: Advances in cryptology—EUROCRYPT 2020. Part III, vol. 12107, pp. 183–211 (2020)
18. Biasse, J.F., Micheli, G., Persichetti, E., Santini, P.: LESS is more: code-based signatures without syndromes. In: Progress in cryptology—AFRICACRYPT 2020, Lecture Notes in Comput. Sci., vol. 12174, pp. 45–65. Springer, Cham ([2020] ©2020). https://doi.org/10.1007/978-3-030-51938-4_3, https://doi.org/10.1007/978-3-030-51938-4_3
19. Billet, O., Gilbert, H.: Cryptanalysis of Rainbow. In: De Prisco, R., Yung, M. (eds.) Security and Cryptography for Networks. pp. 336–347 (2006)
20. Briaud, P., Tillich, J.P., Verbel, J.: A polynomial time key-recovery attack on the Sidon cryptosystem. In: AlTawy, R., Hülsing, A. (eds.) Selected Areas in Cryptography. pp. 419–438 (2022)
21. Buss, J.F., Frandsen, G.S., Shallit, J.O.: The computational complexity of some problems of linear algebra. J. Comput. System Sci. **58**(3), 572–596 (1999)
22. Cabarcas, D., Smith-Tone, D., Verbel, J.A.: Key recovery attack for ZHFE. In: Post-quantum cryptography, vol. 10346, pp. 289–308 (2017)
23. Courtois, N.T.: Efficient zero-knowledge authentication based on a linear algebra problem MinRank. In: Advances in cryptology—ASIACRYPT 2001 (Gold Coast), vol. 2248, pp. 402–421 (2001)
24. Courtois, N.T.: La sécurité des primitives cryptographiques basées sur des problèmes algébriques multivariables: MQ, IP, MinRank, HFE. Ph.D. thesis, Université de Paris 6 - Pierre et Marie Curie (2001)
25. Dodis, Y., Kiayias, A., Nicolosi, A., Shoup, V.: Anonymous identification in *ad hoc* groups. In: Advances in cryptology—EUROCRYPT 2004, vol. 3027, pp. 609–626 (2004)
26. Esgin, M.F., Zhao, R.K., Steinfeld, R., Liu, J.K., Liu, D.: MatRiCT: efficient, scalable and post-quantum blockchain confidential transactions protocol. In: Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security. pp. 567–584 (2019)
27. Faugère, J.C., Safey El Din, M., Spaenlehauer, P.J.: Computing loci of rank defects of linear matrices using Gröbner bases and applications to cryptology. In: ISSAC 2010—Proceedings of the 2010 International Symposium on Symbolic and Algebraic Computation. pp. 257–264 (2010)
28. Feneuil, T., Joux, A., Rivain, M.: Shared permutation for syndrome decoding: New zero-knowledge protocol and code-based signature. Cryptology ePrint Archive (2021)

29. Feneuil, T., Joux, A., Rivain, M.: Syndrome decoding in the head: Shorter signatures from zero-knowledge proofs. Cryptology ePrint Archive (2022)
30. Gaborit, P., Schrek, J., Zémor, J.: Full cryptanalysis of the chen identification protocol. In: Post-Quantum Cryptography - 4th International Workshop, PQCrypto 2011. vol. 7071, pp. 35–50 (2011)
31. Goubin, L., Courtois, N.T.: Cryptanalysis of the TTM cryptosystem. In: Advances in cryptology—ASIACRYPT 2000, vol. 1976, pp. 44–57 (2000)
32. Gueron, S., Persichetti, E., Santini, P.: Designing a practical code-based signature scheme from zero-knowledge proofs with trusted setup. Cryptography **6**(1), 5 (2022)
33. Ishai, Y., Kushilevitz, E., Ostrovsky, R., Sahai, A.: Zero-knowledge proofs from secure multiparty computation. SIAM J. Comput. **39**(3), 1121–1152 (2009)
34. Katz, J., Kolesnikov, V., Wang, X.: Improved non-interactive zero knowledge with applications to post-quantum signatures. Proceedings of the ACM Conference on Computer and Communications Security pp. 525–537 (2018)
35. Kipnis, A., Shamir, A.: Cryptanalysis of the HFE public key cryptosystem by relinearization. In: Advances in cryptology—CRYPTO '99, Lecture Notes in Comput. Sci., vol. 1666, pp. 19–30. Springer, Berlin (1999)
36. Linton, S., Nebe, G., Niemeyer, A., Parker, R., Thackray, J.: A parallel algorithm for Gaussian elimination over finite fields. arXiv preprint arXiv:1806.04211 (2018)
37. Lu, X., Au, M.H., Zhang, Z.: Raptor: a practical lattice-based (linkable) ring signature. In: Applied cryptography and network security, vol. 11464, pp. 110–130 (2019)
38. Lyubashevsky, V.: Fiat-Shamir with aborts: applications to lattice and factoring-based signatures. In: Advances in cryptology—ASIACRYPT, vol. 5912, pp. 598–616 (2009)
39. MATZOV: Report on the security of LWE: Improved dual lattice attack
40. Moody, D., Alagic, G., Apon, D.C., Cooper, D.A., Dang, Q.H., Kelsey, J.M., Liu, Y.K., Miller, C.A., Peralta, R.C., Perlner, R.A., et al.: Status report on the second round of the NIST post-quantum cryptography standardization process (2020)
41. Moody, D., Perlner, R., Smith-Tone, D.: Key recovery attack on the cubic ABC simple matrix multivariate encryption scheme. In: Selected areas in cryptography—SAC 2016, vol. 10532, pp. 543–558 (2017)
42. Ohta, K., Okamoto, T.: A digital multisignature scheme based on the fiat-shamir scheme. In: International Conference on the Theory and Application of Cryptology. pp. 139–148 (1991)
43. Peikert, C.: He gives C-sieves on the CSIDH. In: Advances in cryptology—EUROCRYPT 2020. Part II, vol. 12106, pp. 463–492 (2020)
44. Santoso, B., Ikematsu, Y., Nakamura, S., Yasuda, T.: Three-pass identification scheme based on MinRank problem with half cheating probability, https://arxiv.org/abs/2205.03255
45. Smith-Tone, D., Verbel, J.: A rank attack against extension field cancellation. In: Post-quantum cryptography, vol. 12100, pp. 381–401 (2020)
46. Stern, J.: A new identification scheme based on syndrome decoding. In: Advances in Cryptology - CRYPTO '93. vol. 773, pp. 13–21 (1993)
47. Strassen, V., et al.: Gaussian elimination is not optimal. Numerische mathematik **13**(4), 354–356 (1969)
48. Tao, C., Petzoldt, A., Ding, J.: Efficient key recovery for all HFE signature variants. In: Advances in cryptology—CRYPTO 2021. Part I, vol. 12825, pp. 70–93 (2021)
49. Véron, P.: Improved identification schemes based on error-correcting codes. Appl. Algebra Eng. Commun. Comput. **8**(1), 57–69 (1996)

A Commitment scheme

In this section we give the formal definition of a computation hiding and computation binding commitment scheme.

Definition 3 (Computational hiding). *We say that* Com *is computationally hiding if for all polynomial time algorithms* $\mathcal{A}$*, and every pair of messages* m*,* m' *the advantage* $\mathsf{Adv}^{\text{hiding}}_{\mathsf{Com}}(\mathcal{A}, m, m')$ *is a negligible function of the security parameter* λ*, where*

$$\mathsf{Adv}^{\text{hiding}}_{\mathsf{Com}}(\mathcal{A}, m, m') := \left| \Pr_{\mathsf{bits} \xleftarrow{\$} \{0,1\}^\lambda} \left[\mathcal{A}\big(\mathsf{Com}(\mathsf{bits}, m)\big) = 1\right] - \Pr_{\mathsf{bits} \xleftarrow{\$} \{0,1\}^\lambda} \left[\mathcal{A}\big(\mathsf{Com}(\mathsf{bits}, m')\big) = 1\right] \right|.$$

Definition 4 (Computational binding). *We say that* Com *is computationally binding if for all polynomial time algorithms* $\mathcal{A}$*, the advantage* $\mathsf{Adv}^{\text{binding}}_{\mathsf{Com}}(\mathcal{A})$ *is a negligible function of the security parameter* λ*, where*

$$\mathsf{Adv}^{\text{binding}}_{\mathsf{Com}}(\mathcal{A}) = \Pr\left[\mathsf{Com}(\mathsf{bits}, m) = \mathsf{Com}(\mathsf{bits}', m') \mid (\mathsf{bits}, m, \mathsf{bits}', m') \leftarrow \mathcal{A}(1^\lambda)\right].$$

B Ring signatures

In the following we give the formal definition of a ring signature scheme.

Definition 5 (Ring signature scheme). *A ring signature scheme is a triple of polynomial time algorithms* $(\mathit{Gen}, \mathit{Sign}, \mathit{Verify})$ *that generates keys, sign a message, and verify the signature of a message, respectively. Formally:*

- $\mathit{Gen}(1^\lambda)$ *outputs a key pair* $(\mathsf{pk}, \mathsf{sk})$*, where* pk *denotes the public key and* sk *its corresponding secret key.*
- $\mathit{Sign}_{\mathsf{sk}_i}(\mathit{msg}, R)$ *outputs a signature* σ *of the message* msg *with respect to the ring* $R = (\mathsf{pk}_1, \ldots, \mathsf{pk}_u)$*. Here it is assumed that: (1)* $(\mathsf{pk}_i, \mathsf{sk}_i)$ *is a valid key-pair output by* Gen*; (2)* $|R| \geq 2$*; and (3) each public key in the ring is distinct.*
- $\mathit{Verify}_R(\mathit{msg}, \sigma)$ *verifies a signature* σ *of the message* msg *with respect to* R*.*

We say that a ring signature scheme is correct if it satisfy the following correctness condition: for every λ *and for every set of outputs* $\{(\mathsf{pk}_i, \mathsf{sk}_i)\}_{i=1}^u$ *of* $\mathit{Gen}(1^\lambda)$ *it holds*

$$\mathit{Verify}_R(\mathit{msg}, \mathit{Sign}_{\mathsf{sk}_i}(\mathit{msg}, R)) = 1,$$

where $R = (\mathsf{pk}_1, \ldots, \mathsf{pk}_u)$*.*

B.1 Security definitions

Next we give the security definitions for ring signatures following Bender, Katz, and Morselli [13].

Definition 6 (Anonymity w.r.t adversarially-chosen keys). *Let* $(\mathsf{Gen}, \mathsf{Sign}, \mathsf{Verify})$ *be a ring signature scheme,* $u(\cdot)$ *a polynomial, and let* $\mathcal{A}$ *be a PPT adversary. Consider the following game:*

1. *The key pairs* $\{(\mathsf{pk}_i, \mathsf{sk}_i)\}_{i=1}^{u(\lambda)}$ *are generated using* $\mathsf{Gen}(1^\lambda)$*, and the set of public keys* $S := \{\mathsf{pk}_i\}_{i=1}^{u(\lambda)}$ *is given to* $\mathcal{A}$*.*
2. $\mathcal{A}$ *is given access to an oracle* $\mathsf{OSign}(\cdot,\cdot,\cdot)$ *such that for every* R *and* $1 \leq i \leq u(\lambda)$ *it holds* $\mathsf{OSign}(i, \mathsf{msg}, \mathsf{R}) := \mathsf{Sign}_{\mathsf{sk}_i}(\mathsf{msg}, \mathsf{R})$*, where* $\mathsf{pk}_i \in \mathsf{R}$*.*
3. $\mathcal{A}$ *outputs a message* msg *and a ring* R *that contains at least two public keys* $\mathsf{pk}_{i_0}, \mathsf{pk}_{i_1} \in S$*.*
4. *A challenge signature* $\sigma \leftarrow \mathsf{Sign}_{\mathsf{sk}_{i_b}}(\mathsf{msg}, \mathsf{R})$*, where* $b \xleftarrow{\$} \{0,1\}$ *is a random bit, is given to* $\mathcal{A}$*.*
5. $\mathcal{A}$ *outputs a bit* b'*, and it succeeds if* $b' = b$*.*

We say $(\mathsf{Gen}, \mathsf{Sign}, \mathsf{Verify})$ *achieves Anonymity w.r.t adversarially-chosen keys if, for any PPT* $\mathcal{A}$ *and any polynomial* $u(\cdot)$*, the success probability of* $\mathcal{A}$ *in the aforementioned game is negligibly close to* $\frac{1}{2}$*.*

Note that in contrast to the weaker security notion of *basic anonymity* the property of *anonymity w.r.t adversarially-chosen keys* allows the adversary to inject own public keys in R. This holds for the usage of the oracle in step 2 as well as when providing the challenge data in step 3.

Definition 7 (Unforgeability against fixed-ring attacks). *We say that a ring signature* $(\mathsf{Gen}, \mathsf{Sign}, \mathsf{Verify})$ *is unforgeable against fixed-ring attacks if for any PPT adversary* $\mathcal{A}$ *and for any polynomial* $u(\cdot)$*, the probability that* $\mathcal{A}$ *succeeds in the following game is negligible:*

1. *The key pairs* $\{(\mathsf{pk}_i, \mathsf{sk}_i)\}_{i=1}^{u(\lambda)}$ *are generated using* $\mathsf{Gen}(1^\lambda)$*, and the set of public keys* $R := \{\mathsf{pk}_i\}_{i=1}^{u(\lambda)}$ *is given to* $\mathcal{A}$*.*
2. $\mathcal{A}$ *is given access to a signing oracle* $\mathsf{OSign}(\cdot,\cdot)$*, where* $\mathsf{OSign}(i, \mathsf{msg})$ *outputs* $\mathsf{Sign}_{\mathsf{sk}_i}(\mathsf{msg}, R)$*.*
3. $\mathcal{A}$ *outputs* $(\mathsf{msg}^*, \sigma^*)$*, and succeeds if* $\mathsf{Verify}(\mathsf{msg}^*, \sigma^*) = 1$ *and also* $\mathcal{A}$ *never made a query of the form* $\mathsf{OSign}(*, \mathsf{msg}^*)$*.*

B.2 Proofs

In the following we prove the correctness, anonymity, and unforgeability of our ring-signature scheme defined in Section 5.1.

Correctness Let $\boldsymbol{\varepsilon}_i$ be the i-th canonical vector in $\mathbb{F}_q^u$ and sk_i denote the secret key of the i-th user in the ring R. Clearly, $\gamma_i := (\mathsf{sk}_i, \boldsymbol{\varepsilon}_i)$ is a solution to the MinRank problem defined on $\widetilde{\boldsymbol{M}} := (\boldsymbol{M}, \mathsf{R})$. The correctness of the ring signature scheme now follows from the correctness of our basic signature scheme by observing that

$$\mathsf{Verify}_{\mathsf{R}}\big(\mathsf{msg}, \mathsf{Sign}_{\mathsf{sk}_i}(\mathsf{msg}, \mathsf{R})\big) = \mathsf{MR\text{-}Verify}_{\widetilde{\boldsymbol{M}}}\big(\mathsf{msg}, \mathsf{MR\text{-}Sign}_{\gamma_i}(\mathsf{msg})\big).$$

Anonymity w.r.t adversarially-chosen keys We proof anonymity w.r.t adversarially-chosen keys in the random oracle model by showing the existence of a simulator that, without knowing any of the secret keys corresponding to one of the public keys in the ring, can produce signatures that are indistinguishable from signatures build by a legitimate user.

First note that from the HVZK property of our sigma protocol in the random oracle model it follows that there exists a simulator $\mathcal{S}'$ which is able to provide values σ' indistinguishable from legitimate signatures produced with MR-Sign. To construct $\mathcal{S}'$ we simply follow the Fiat–Shamir transform but using the simulator $\mathcal{S}$ of our sigma protocol whenever a valid transcript is needed.

Now, recall that the signing operation of our ring signature is a call to MR-Sign with adapted public-key $(\boldsymbol{M}, \mathsf{R})$, where

$$\mathsf{Sign}_{\mathsf{sk}_i}(\mathsf{msg}, \mathsf{R}) = \mathsf{MR\text{-}Sign}_{\mathsf{sk}'_i}(\mathsf{msg}).$$

Therefore we can use $\mathcal{S}'$ as a simulator to obtain values σ' which are indistinguishable from legitimate ring signatures.

Now, let G_0 denote the game described in Definition 6. We modify step 4 in G_0 to define a new game G_1. Instead of $\sigma \leftarrow \mathsf{Sign}_{\mathsf{sk}_{i_b}}(\mathsf{msg}, \mathsf{R})$, the output of step 4 in G_1 is $\sigma' \leftarrow \mathcal{S}'(\mathsf{msg}, \mathsf{R})$. Notice G_0 and G_1 are indistinguishable games. Hence, the advantage of any adversary $\mathcal{A}$ against G_0 and G_1 is the same. Also, the challenge σ' given in G_1 does not depend on the bit b chosen in step 3. Therefore, the advantage of an adversary $\mathcal{A}$ against game G_1 is zero.

Unforgeability against fixed-ring attacks Forging a signature for a fixed ring R, i.e., winning the game given in Definition 7, directly reduces to forging a signature for MR-Sign with public-key $(\boldsymbol{M}, \mathsf{R})$. The unforgeability for MR-Sign now follows from the Fiat–Shamir transform applied to the sigma protocol and its HVZK property.

C A note on Santoso et al.'s scheme

The parameters given by Santoso et al. [44] to obtain a security level of λ bits are shown in Table 5.

Parameter set	λ	q	n	m	k	r
A	128	2	26	26	208	13
B	192	2	33	33	330	17
C	256	2	39	39	468	20

Table 5: Parameter sets proposed in [44].

Missing commitments in the signature size The authors of [44] disregard the size of the initial commitments in their analysis of the communication complexity. Taking commitment sizes into account (2λ bits for each hash, to be collision-resistant) the signature size of [44] is given by

$$\lambda \left(\frac{29}{2}\lambda + mn \log q + \frac{k}{2} \log q \right). \tag{5}$$

While the signature size of Courtiois' scheme is given by

$$\frac{\lambda}{\log(3/2)} \left(\frac{20}{3}\lambda + \frac{2}{3}mn \log q + \frac{2}{3}k \log q \right). \tag{6}$$

Random solutions As stated in Section 4.6, a random instance of the MinRank problem with parameters (q, n, m, k, r) has, in expectation, $n_{sol} := q^{k-(m-r)(n-r)}$ solutions. Some algorithms, as e.g., the kernel search algorithm, can directly benefit from multiple solutions by obtaining a speed-up of magnitude $n_{sol} > 1$ in those cases. It turns out that the parameter sets given in [44] contain a large amount of solutions, affecting security.

New security estimates and signature size Table 6 shows the bit-security of the kernel search algorithm for parameters suggested in [44]. Note that all the parameter sets are far below the claimed bit-security, which is 128 for set A, 192 for set B, and 256 for set C. Also, observe that the signature size is larger than the one of standard Courtois for all suggested parameters.

Parameters set	Algorithm	Bit-security	Courtois' signature size using Eq. (6)	Santoso et al.'s signature size given in [44]	Santoso et al.'s signature size using Eq. (5)
A	kernel search	88	38.54 KB	18.81 KB	41.19 KB
B	kernel search	121	89.19 KB	44.50 KB	94.64 KB
C	kernel search	159	162.01 KB	82.15 KB	170.84 KB

Table 6: Bit-security and signature size for parameter sets proposed in [44].

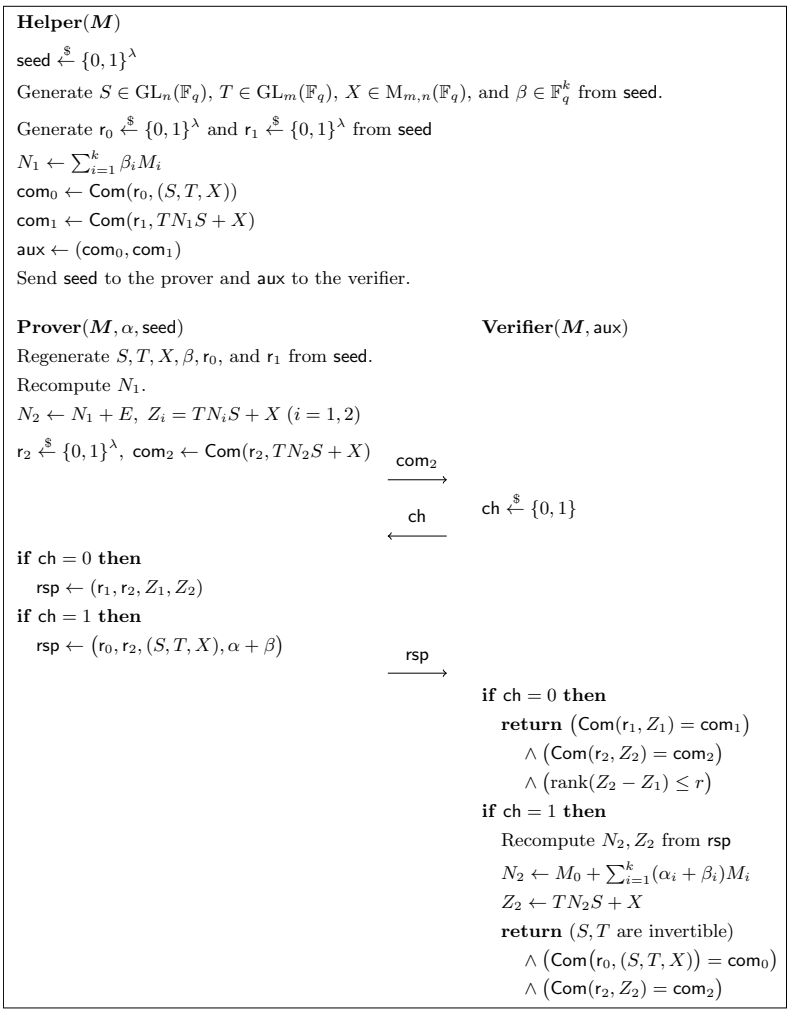

Fig. 3: Structure of a sigma protocol with helper for ZK proof of MinRank.

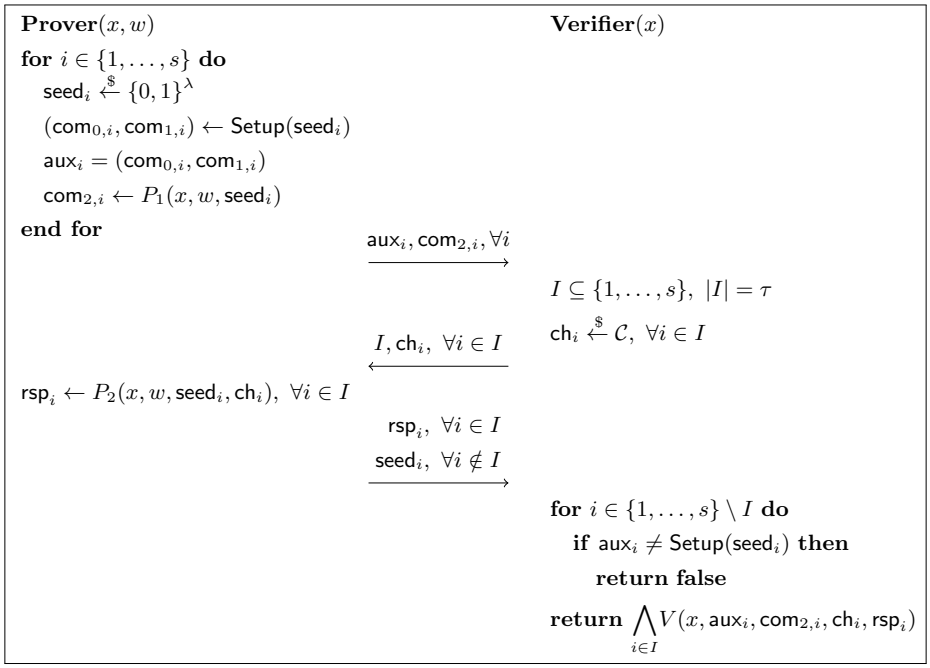

Fig. 4: Sigma protocol obtained by removing the helper from the protocol in Fig. 3. P_1, P_2 and V relate to the actions performed by the prover and verifier in Fig. 3 respectively.