Improved Attacks for SNOVA by Exploiting Stability under a Group Action

Daniel Cabarcas[1], Peigen Li[2], Javier Verbel[3], and Ricardo Villanueva-Polanco[3]

[1] Universidad Nacional de Colombia, Colombia
dcabarc@unal.edu.co
[2] Beijing Institute of Mathematical Sciences and Applications, China
lpg22@bimsa.cn
[3] Technology Innovation Institute, UAE
{javier.verbel,ricardo.polanco}@tii.ae

Abstract. SNOVA is a post-quantum digital signature scheme based on multivariate polynomials. It is a second-round candidate in an ongoing NIST standardization process for post-quantum signatures, where it stands out for its efficiency and compactness. Since its initial submission, there have been several improvements to its security analysis, both on key recovery and forgery attacks. All these works reduce to solving a structured system of quadratic polynomials, which we refer to as SNOVA system.

In this work, we propose a polynomial solving algorithm tailored for SNOVA systems, which exploits the *stability* of the system under the action of a commutative group of matrices. This new algorithm reduces the complexity of solving SNOVA systems over generic ones. We show how to adapt the *reconciliation* and *direct* attacks in order to profit from the new algorithm. Consequently, we improve the reconciliation attack for all SNOVA parameter sets with speedup factors ranging between 2 and 2^{20}. We also show how to use similar ideas to carry on a forgery attack. In this case, we use experimental results to estimate its complexity, and we discuss its impact. The empirical evidence suggests that our attack is more efficient than previous attacks, and it takes some SNOVA parameter sets below NIST's security threshold.

Keywords: Cryptanalysis, SNOVA, stable ideals, post-quantum, multivariate.

1 Introduction

Digital signatures are essential to ensure the authenticity and integrity of digital communications. The security of widely used digital signature schemes is threatened by quantum computers [16]. Post-quantum cryptography (PQC) is an active area of research aiming at developing cryptographic algorithms that are resilient against quantum attacks. In light of this, NIST has been leading an effort to evaluate and standardize cryptographic algorithms capable of withstanding quantum adversaries.

Following the success of its first PQC standardization process, in 2023 NIST initiated a second call for submissions, focused on digital signatures. One promising candidate in this process is the SNOVA signature scheme [19], which builds upon the Unbalanced Oil and Vinegar (UOV) signature scheme. SNOVA modifies UOV's structure to reduce the size of the public key and the speed of signing while keeping UOV's short signatures and fast verification. For example, at security level I, SNOVA can have 1000-byte public keys and 232-byte signatures, while the speed of signing and verification are comparable to those of Dilithium, one of the post-quantum signature schemes standardized by NIST. Due to its efficiency and compactness, SNOVA offers an attractive option for real-world applications.

The main concern about SNOVA is its security, which is the main focus of this work. There have been several papers since the beginning of 2024 analyzing SNOVA's security from different perspectives. Since our work builds upon those works, it is important to summarize their main findings.

Key-Recovery and reconciliation. In two independent but concurrent works, Ikematsu-Akiyama [9] and Li-Ding [10] analyzed the security of SNOVA against key-recovery attacks. Both works reached the same conclusion: All known key-recovery attacks for SNOVA with parameters (v, o, l, q) can be seen as attacks to a UOV signature scheme with lo^2 equations and $l(v+o)$ variables over $\mathbb{F}_q$. In particular, for the *reconciliation* attack, the attacker finds a solution of a quadratic polynomial system of the form

$$\mathbf{x}^t \Lambda_{S^i}^{(n)} P_k \Lambda_{S^j}^{(n)} \mathbf{x} = 0, \quad \forall k = 1, \ldots, o, \text{ and } 0 \leq i, j < l, \tag{1}$$

where $n = v + o$, $\mathbf{x} \in \mathbb{F}_q^{ln}$ is a vector of variables, $P_k \in \mathbb{F}_q^{ln \times ln}$ is the public key matrix, $\Lambda_{S^i}^{(n)} \in \mathbb{F}_q^{ln \times ln}$ is a block-diagonal matrix with $S^i \in \mathbb{F}_q^{l \times l}$ along the diagonal, and $S \in \mathbb{F}_q^{l \times l}$ is a symmetric matrix with an irreducible characteristic polynomial. Throughout this paper, we refer to any system whose quadratic part has the form (1) as a *SNOVA system*. More recently, Nakamura, Tani, and Furie also discuss a similar attack in [12].

Forgery Attacks. In a recent preprint, Beullens proposed a new forgery attack on SNOVA [3]. The main observation is that the SNOVA public key has a similar structure to MAYO's [2]. Specifically, denoting by $\mathcal{B}(\mathbf{u}_i, \mathbf{u}_j)$ the bilinear map associated with the SNOVA sequence defined by the public key, the SNOVA public key can be written as

$$\mathcal{P}(U) = \sum_{i=1}^{l} \sum_{j=1}^{l} E_{i,j} \mathcal{B}(\mathbf{u}_i, \mathbf{u}_j),$$

where $U = [\mathbf{u}_1, \ldots, \mathbf{u}_l]$, $\mathbf{u}_i \in \mathbb{F}_q^{nl}$, the $E_{i,j} \in \mathbb{F}_q^{ol^2 \times ol^2}$ are block-diagonal matrices with o copies of a matrix $\tilde{E}_{i,j} \in \mathbb{F}_q^{l^2 \times l^2}$ along the diagonal. Unlike MAYO, there

may be a nontrivial linear combination E of the matrices $E_{i,j}$ with a rank defect. Beullens uses this fact to speed up a forgery attack.

In a quick reaction to [3], and in order to mitigate the impact on the security of SNOVA, the submitters have suggested two potential alternatives to update the scheme [18]. At the time we drafted this manuscript, it was unclear how they would update the NIST submission. We consider that analyzing the impact of our attacks for those alternatives falls outside of the scope of this work, however, we do take into account the new parameters proposed in [18] for the original SNOVA scheme.

1.1 Our Contributions

The main contributions of this paper are a polynomial solving algorithm tailored for SNOVA systems, the analysis of its complexity, and its impact on the security of SNOVA. Our algorithm builds on the observation that the ideal I generated by the quadratic part of Eq. (1) is $\Lambda^{(n)}_{\mathbb{F}_q[S]}$-*stable*, meaning that, for every $f \in I$ and $A \in \Lambda^{(n)}_{\mathbb{F}_q[S]}$, $f(A\mathbf{x})$ belongs to I [4]. In [7], Faugère and Svartz propose a variant of the F5 algorithm to compute a Gröbner basis of a $\mathcal{D}$-stable ideal, where $\mathcal{D}$ is any commutative group of matrices. They provide an asymptotical analysis of the complexity when the degree of the Macaulay matrix tends to infinity.

We adapt the ideas in [7] for SNOVA systems. Applying an appropriate change of variables over a field extension to the ideal I, we obtain a stable ideal under the action of a cyclic diagonal group of matrices. We show that the resulting polynomial system has a multi-homogeneous structure. Unlike [7], we propose an XL-like algorithm to solve the system and leverage the multi-homogeneous structure. In addition, we provide a concrete and tight analysis of our algorithm's complexity, which is supported experimentally. Our complexity estimates show that SNOVA systems can be solved faster than random quadratic systems by a factor of about q^l.

By using our new algorithm to solve SNOVA systems, we improve the reconciliation attack in [9,10], with speedup factors ranging between 2 and 2^{20}.

We also show how to use the techniques developed for the new algorithm in the forgery attack by Beullens [3]. In order to reduce the forgery of a message msg to the problem of solving a SNOVA system, while maintaining the multi-homogeneity of the lifted system, we leverage the low-rank of the matrix E in a different way than Beullens does.

Instead of finding elements in the left kernel of E to obtain linear equations, we first bruteforce a value $\mathsf{salt} \in \{0,1\}^{128}$ such that the target vector $\mathbf{z}_0 = \mathsf{Hash}(\mathsf{msg}\|\mathsf{salt})$ falls in the columns space of E, where $\mathsf{Hash} : \{0,1\}^* \to \mathbb{F}_q^{ol^2}$ is a cryptographic hash function. We then use a basis of the affine subspace of solutions to $E\mathbf{w} = \mathbf{z}_0$, to state the forgery as a system $\mathcal{F}(\mathbf{u}) = \mathbf{w}_0 + W\mathbf{y}$, which is quadratic in $\mathbf{u}$ and linear in the coordinates $\mathbf{y}$ of the affine subspace. By lifting this system via an appropriate change of variables, we arrive at a system

[4] $\Lambda^{(n)}_{\mathbb{F}_q[S]}$ is the group given by all $\mathbb{F}_q$-linear combinations of $\Lambda^{(n)}_{S^0}, \Lambda^{(n)}_{S^1} \ldots, \Lambda^{(n)}_{S^{l-1}}$.

whose quadratic part is multi-homogeneous. However, the system has a linear part coming from the coordinates of $\mathbf{y}$, so we cannot directly use the algorithm to solve SNOVA systems. Instead, we present empirical evidence showing that such a system can be solved faster than a random one using an out-of-the-box Gröbner basis algorithm. We conjecture its complexity based on the experimental results. The experiments suggest that the actual complexity of an efficient implementation is faster than Beullens' and puts most parameter sets below the security threshold defined by NIST. This is particularly relevant for parameters with $l = 4$, since they allow the smallest keys and signatures, and they are not significantly affected by the attack in [3].

This paper is organized as follows. Section 2 describes the SNOVA signature scheme, some of the attacks, and a special XL algorithm for multi-homogeneous systems. In Section 3.1, we describe some of the algebraic properties of SNOVA systems. Section 4 introduces the proposed polynomial solving algorithm tailored for SNOVA systems and gives a detailed analysis of its complexity. Section 5 presents our adaptation of the reconciliation and forgery attacks to profit from the proposed algorithm and its impact on the security of SNOVA.

2 Preliminaries

2.1 Notation

In this paper, we use the following notation:

- v, o, q, l are positive integers defining the parameters of SNOVA.
- We set $n = v + o$ and $m = ol^2$.
- $\mathbb{F}_q$ denotes a finite field with q elements.
- $\mathbb{F}_q^{n_r \times n_c}$ denotes the set of matrices of size $n_r \times n_c$ with entries in $\mathbb{F}_q$.
- $S \in \mathbb{F}_q^{l \times l}$ is symmetric with an irreducible characteristic polynomial.
- $\mathbb{F}_q[S]$ denotes the set $\{a_0 S^0 + a_1 S + \cdots + a_{l-1} S^{l-1} : \ a_0, \ldots, a_{l-1} \in \mathbb{F}_q\}$.
- $[k]$ denotes the set $\{1, \ldots, k\}$.
- $\Lambda_Q^{(n)}$ denotes the block-diagonal matrix with n copies of Q along the diagonal.

2.2 The SNOVA Signature Scheme

SNOVA is a post-quantum digital signature scheme introduced by Wang, Tseng, Kuan and Chou in [19] that aims to provide UOV-like signatures, but has a small public key size. SNOVA is a second-round candidate in the on-ramp standardization process led by NIST.

The public key in SNOVA is given by the quadratic map $\mathcal{P} = (\mathcal{P}_1, \ldots, \mathcal{P}_o) : \mathbb{F}_q^{nl^2} \to \mathbb{F}_q^{ol^2}$, where each $\mathcal{P}_k : \mathbb{F}_q^{nl^2} \to \mathbb{F}_q^{l^2}$ is defined as

$$\mathcal{P}_k(U) = \sum_{\alpha=1}^{l^2} A_\alpha \cdot U^t \left(\Lambda_{Q_{\alpha,1}}^{(n)} P_k \Lambda_{Q_{\alpha,2}}^{(n)} \right) U \cdot B_\alpha,$$

where U is a matrix of variables of size $ln \times l$, and the matrices $Q_{\alpha,1}, Q_{\alpha,2} \in \mathbb{F}_q[S]$, $A_\alpha, B_\alpha \in \mathbb{F}_q^{l\times l}$ and $P_k \in \mathbb{F}_q^{ln\times ln}$ are public.

The secret key is given by an invertible matrix $T \in (\mathbb{F}_q[S])^{n\times n}$ and matrices $F_1, \ldots, F_o \in \mathbb{F}_q^{ln\times ln}$ such that

$$P_k = T^t F_k T, \text{ for each } k \in [o],$$

and each F_k is of the form

$$F_k = \begin{bmatrix} F_{k,1} & F_{k,2} \\ F_{k,3} & 0 \end{bmatrix},$$

where $F_{k,1} \in \mathbb{F}_q^{lv\times lv}$, $F_{k,2} \in \mathbb{F}_q^{lv\times lo}$ and $F_{k,3} \in \mathbb{F}_q^{lo\times lv}$.

A pair $\sigma = (U, \mathsf{salt})$, with $U \in \mathbb{F}_q^{ln\times n}$ and $\mathsf{salt} \in \{0,1\}^{128}$, is a valid signature for a message $\mathsf{msg} \in \{0,1\}^*$ under the public key $\mathcal{P}$ if and only if $\mathcal{P}(U) = \mathsf{Hash}(\mathsf{msg}\|\mathsf{salt})$, where $\mathsf{Hash} : \{0,1\}^* \to \mathbb{F}_q^{ol^2}$ is a cryptographic hash function.

Security level	parameters v	o	q	l	public key size	signature size	private key size
I	37	17	16	2	9826	124	60008
	25	8	16	3	2304	165	37962
	24	5	16	4	1000	248	34112
III	56	25	16	2	31250	178	202132
	49	11	16	3	5990	286	174798
	37	8	16	4	4096	376	128384
V	75	33	16	2	71874	232	515360
	66	15	16	3	15188	381	432297
	60	10	16	4	8000	576	389312

Table 1. Proposed parameters for SNOVA with corresponding signature and key sizes in bytes [19].

Table 1 shows the latest parameters proposed by the SNOVA designers. Note that the parameters for $l = 2$ shown in Table 1 do not match the original parameters in the NIST submission, since the original ones were updated in response to the attacks in [9,10]. The SNOVA scheme is still competitive and it provides three parameter options for each security level, which yield different public key and signature sizes. This feature gives this scheme flexibility and adaptability to be used in diverse scenarios.

We remark that the SNOVA parameters with $l = 4$ appear to be a good alternative to the standardized scheme FALCON, since they offer similar key sizes and smaller signature sizes than FALCON. Furthermore, for the security levels I, III, and V as specified by the NIST standardization call [14, Section 4B], SNOVA allegedly offers 143, 207, and 272 bits of security respectively.

2.3 Key Recovery Attacks

A key-recovery attack for SNOVA reduces to finding a basis to the secret space

$$\mathcal{O} := \{T^{-1} \cdot (a_1, \ldots, a_{ln})^t \in \mathbb{F}_q^{ln} : a_1 = a_2 = \cdots = a_{lv} = 0\},$$

which has dimension lo [19]. In [9,10], it is shown that for any $\mathbf{u}, \mathbf{v} \in \mathcal{O}$, we have

$$\mathbf{v}^t (\Lambda^{(n)}_{S^{i-1}} P_k \Lambda^{(n)}_{S^{j-1}}) \mathbf{u} = 0, \quad \text{for each } (i,k,j) \in [l] \times [o] \times [l]. \tag{2}$$

The main goal of a key-recovery attack for SNOVA is to find at least one nontrivial element in $\mathcal{O}$. Once such an element is found, recovering a basis of $\mathcal{O}$ is significantly easier than finding that first element in $\mathcal{O}$.

In the *reconciliation* attack, one attempts to find a vector of the form $\mathbf{u}_0 = (u_1, \ldots, u_{lv}, 0, \ldots, 1)^t \in \mathbb{F}_q^{ln}$ such that

$$\mathbf{u}_0^t (\Lambda^{(n)}_{S^{i-1}} P_k \Lambda^{(n)}_{S^{j-1}}) \mathbf{u}_0 = 0, \quad \text{for each } (i,k,j) \in [l] \times [o] \times [l]. \tag{3}$$

Since $\mathcal{O}$ is a random vector space of dimension ol, we expect that such a $\mathbf{u}_0 \in \mathcal{O}$ uniquely exists.

The quadratic system in (3) has $l^2 o$ equation on lv variables. For all SNOVA parameters, it holds that $ol^2 < lv$ [19]. Therefore, the system in (3) has an expected number of $O(q^{lv-ol^2})$ solutions. In [9,10], the authors compute the complexity of this attack by using the hybrid approach [1], assuming that solving an underdetermined system as in (3) is as hard as solving a random one. However, in Section 4, we will introduce an algorithm that solves such systems more efficiently than a generic algorithm.

2.4 Forgery Attacks

Given a public key $\mathcal{P}$, an attacker forges a signature for a message $\mathsf{msg} \in \{0,1\}^*$ by finding $U \in R^n$ and $\mathsf{salt} \in \{0,1\}^{128}$ such that $\mathcal{P}(U) = \mathsf{Hash}(\mathsf{msg}||\mathsf{salt})$.

To the best of our knowledge, the most efficient forgery attack against SNOVA was introduced by Beullens in [3]. The attack builds from the observation that, with $U = [\mathbf{u}_0|\mathbf{u}_1|\cdots|\mathbf{u}_{l-1}]$ and each $\mathbf{u}_i$ being vector of ln variables, the public polynomials $\mathcal{P}(U)$ can be written as

$$\mathcal{P}(U) = \sum_{i=0}^{l-1} \sum_{j=0}^{l-1} E_{i,j} \mathcal{B}(\mathbf{u}_i, \mathbf{u}_j),$$

where each $E_{i,j} \in \mathbb{F}_q^{ol^2 \times ol^2}$ is a block-diagonal matrix that can be efficiently computed from the public key, and $\mathcal{B} : \mathbb{F}_q^{nl} \times \mathbb{F}_q^{nl} \to \mathbb{F}_q^{ol^2}$ is a bilinear map defined by $\mathcal{B}(\mathbf{x}, \mathbf{y}) := (\mathcal{B}_1(\mathbf{x}, \mathbf{y}), \ldots, \mathcal{B}_o(\mathbf{x}, \mathbf{y}))$ and each

$$\mathcal{B}_k(\mathbf{x}, \mathbf{y}) := (\mathbf{x}^t \Lambda^{(n)}_{S^{i-1}} P_k \Lambda^{(n)}_{S^{j-1}} \mathbf{y})^t_{i,j \in [l]},$$

where $P_k \in \mathbb{F}_q^{ln \times ln}$ and $S \in F_q^{l \times l}$ are public matrices.

In [3], Beullens performs the change of variables $\mathbf{u}_0 = \mathbf{u}$ and $\mathbf{u}_i = \Lambda_{R_i}^{(n)}\mathbf{u} + \mathbf{v}_i$ for $i = 1, \ldots, l-1$, with $R_i \in \mathbb{F}_q[S]$, and $\mathbf{v}_i \in \mathbb{F}_q^{ln}$ to obtain a public key $\mathcal{P}$ of the form

$$E \cdot \mathcal{F}(\mathbf{u}) + \sum_{i,i=0}^{l-1} E_{i,j} \left(\mathcal{B}(\Lambda_{R_i}^{(n)}\mathbf{u}, \mathbf{v}_j) + \mathcal{B}(\mathbf{v}_i, \Lambda_{R_j}^{(n)}\mathbf{u}) \right) + \sum_{i,j=0}^{l-1} E_{i,j}\mathcal{B}(\mathbf{v}_i, \mathbf{v}_j),$$

where $\mathcal{F}(\mathbf{u}) = \mathcal{B}(\mathbf{u}, \mathbf{u})$, and $E = \Lambda_{\tilde{E}}^{(o)} \in \mathbb{F}_q^{ol^2 \times ol^2}$ with $\tilde{E} \in \mathbb{F}_q^{l^2 \times l^2}$ depending on the choice of $R_1, \ldots, R_{l-1}$.

Given an integer $1 \leq r \leq l^2$ Beullens' attack works as follows. First, the attacker bruteforces the R_i until finding a corresponding matrix $\tilde{E}$ with rank r. Next, the attacker finds a full rank matrix $N \in \mathbb{F}_q^{p \times l^2}$, where $p = o(l^2 - r)$, such that $N \cdot E = 0$. Following that, it computes the p linear equations $N \cdot \mathcal{P}$. Finally, the attacker finds a full rank matrix $V \in \mathbb{F}_q^{(l^2-p) \times l^2}$ whose rows are not in the row space of N, and it computes the $l^2 - p$ quadratic equations $V \cdot \mathcal{P}$. Thereafter, the attacker uses the p linear equations to replace p variables in the quadratic ones to obtain a quadratic system of $ol^2 - p$ quadratic equations in $ln - p$ variables.

2.5 An XL-Like Algorithm for Multi-Homogeneous Systems

An important ingredient for our algorithm to solve SNOVA systems is an algorithm that adapts XL for multi-homogeneous systems. Such algorithm has been used for other attacks such as [15,2,11,13]. We describe here its main features for completeness.

Definition 1 (Multi-homogeneous polynomials). *Let $P = (X_1, \ldots, X_l)$ be a partition of the set $X = \{x_1, \ldots, x_n\}$. We say a monomial $\mathfrak{m}$ in X has multi-degree $(d_1, \ldots, d_l) \in \mathbb{Z}_{\geq 0}^l$ if the degree of $\mathfrak{m}$ with respect to X_i equals d_i for each $i \in [l]$. We say that a polynomial f in X is multi-homogeneous of multi-degree $\mathbf{d} \in \mathbb{Z}_{\geq 0}^l$ if each monomial in its support has multi-degree $\mathbf{d}$. In the particular case, $l = 2$, we use bi-degree and bi-homogeneous instead of multi-degree and multi-homogeneous.*

Let $f_1, \ldots, f_m \in \mathbb{F}_q[x_1, \ldots, x_n]$ be multi-homogeneous quadratic polynomials. Like the XL algorithm, the special XL algorithm aims at finding a common root of the f_i. Unlike the XL, the special XL takes as input a multi-degree $(d_1, \ldots, d_l)$ and is restricted to work with polynomials in $\langle f_1, \ldots, f_m \rangle$ of multi-degree $(e_1, \ldots, e_l) \leq (d_1, \ldots, d_l)$, where $\leq$ means that $e_i \leq d_i$ for each $i \in [l]$.

In the bi-homogeneous case, i.e. $l = 2$, the special XL has been used to estimate the complexity of some attacks in cryptography. See, for instance, the RBS and intersection attacks on Rainbow (see [15, Sec. 5] [2, Sec. 6]).

In the case of bi-homogeneous ($l = 2$) quadratic polynomials, the complexity analysis introduced in [15] suggests that for systems in $n = n_1 + n_2$ variables with m_1 equations of bi-degree $(2, 0)$, m_{12} of bi-degree $(1, 1)$ and m_2 of bi-degree

$(0,2)$, the special XL algorithm is expected to effectively work on input bi-degree (a_{sol}, b_{sol}) if the coefficient of $t_1^a t_2^b$ in the series

$$\frac{(1-t_1^2)^{m_1}(1-t_1t_2)^{m_{12}}(1-t_2^2)^{m_2}}{(1-t_1)^{n_1+1}(1-t_2)^{n_2+1}},$$

is nonnegative for some $a \leq a_{sol}$ and $b \leq b_{sol}$. Note that the series above is also considered in [11] to analyze the complexity of the RSB attack on Rainbow. In this case, the complexity of the special XL algorithm, in number of field multiplications, is upper bounded by

$$3 \cdot \max(n_1, n_2)^2 \cdot \left[\overline{\mathcal{M}}(a_{sol}, b_{sol})\right]^2,$$

where $\overline{\mathcal{M}}(a_{sol}, b_{sol})$ is the number of monomials with bi-degree $(a,b) \leq (a_{sol}, b_{sol})$. For general multi-homogeneous ($l \geq 2$) quadratic systems, the complexity is upper bounded by

$$3 \cdot \max(n_1, \ldots, n_l)^2 \cdot \left[\overline{\mathcal{M}}(\mathbf{d}_{sol})\right]^2,$$

where $\mathbf{d}_{sol} \in \mathbb{Z}^l_{\geq 0}$, $\overline{\mathcal{M}}(\mathbf{d}_{sol})$ is the number of monomials with multi-degree smaller than $\mathbf{d}_{sol}$, and there exists $(d_1, \ldots, d_l) \leq \mathbf{d}_{sol}$ such that the coefficient of $t_1^{d_1} t_2^{d_2} \cdots t_l^{d_l}$ in the series

$$\frac{\prod_{k=1}^{m}\left(1 - t_1^{d_1^{(k)}} t_2^{d_2^{(k)}} \cdots t_l^{d_l^{(k)}}\right)}{\prod_{i=1}^{l}(1-t_i)^{n_i+1}}, \tag{4}$$

is nonnegative, where $\left(d_1^{(k)}, d_2^{(k)}, \ldots, d_l^{(k)}\right)$ is the multi-degree of f_k and n_i is the size of X_i. Note that the series in Eq. (4) was used in [13] to estimate the complexity of the Kipnis-Shamir attack to the MinRank problem.

3 SNOVA Sequences and Ideals

We start by defining a SNOVA sequence.

Definition 2 (SNOVA sequences and ideals). *Let o, n, l be positive integers. A SNOVA sequence is a tuple of ol^2 quadratic polynomials in the variables $x_1, \ldots, x_{ln}$ given by*

$$\mathbf{x}^t (A^{(n)}_{S^{i-1}} P_k A^{(n)}_{S^{j-1}}) \mathbf{x}, \text{ for some } (i,k,j) \in [l] \times [o] \times [l] \tag{5}$$

where $\mathbf{x} = (x_1, \ldots, x_{ln})^t$, $P_k \in \mathbb{F}_q^{ln \times ln}$, and $S \in F_q^{l \times l}$ is as in the description of SNOVA. We define a SNOVA ideal as an ideal generated by a SNOVA sequence.

According to Section 2, whether we perform a forgery attack or a key-recovery attack against SNOVA, we note that we need to handle a SNOVA sequence.

3.1 $\Lambda^{(n)}_{\mathbb{F}_q[S]}$-Stable Ideals

This section examines some algebraic properties of SNOVA ideals. In particular, here we show that these ideals are stable under the action of a subgroup of matrices denoted by $\Lambda^{(n)}_{\mathbb{F}_q[S]}$.

We use the following definition from [7].

Definition 3 ($\mathcal{G}$-stable ideals). *An ideal $J \subseteq \mathbb{F}_q[x_1, \ldots, x_n]$ is said to be stable under a finite matrix group $\mathcal{G} \subseteq GL_n(\mathbb{F}_q)$ if for all $f \in J$ and $G \in \mathcal{G}$, we have $f^G \in J$, where $f^G(\mathbf{x}) := f(G\mathbf{x})$.*

Faugère and Svartz [7] have explored the problem of computing a Gröbner basis of stable ideals in general. Most of the results in this section, up to Theorem 1, are adaptations from [7], specifically for the case of a SNOVA system. The remainder of the section, starting in Corollary 1, presents new material, including the observation in Proposition 1.

As in the description of SNOVA, here $S \in \mathbb{F}_q^{l\times l}$ is a symmetric matrix with irreducible characteristic polynomial, and we define

$$\Lambda^{(n)}_{\mathbb{F}_q[S]} := \left\{ \Lambda^{(n)}_Q \in \mathbb{F}_q^{ln\times ln} : Q \in \mathbb{F}_q[S] \setminus \{0\} \right\}.$$

It is easy to see that SNOVA ideals are $\Lambda^{(n)}_{\mathbb{F}_q[S]}$-stable.

Proposition 1. *Let $C \in \mathbb{F}_q^{ln\times ln}$, $\mathbf{x} = (x_1, \ldots, x_{ln})^t$ and F be a sequence of l^2 quadratic polynomials defined by*

$$\mathbf{x}^t(\Lambda^{(n)}_{S^i} C \Lambda^{(n)}_{S^j})\mathbf{x} = 0, \quad \text{for each } i, j = 0, \ldots, l-1.$$

Then, for any $f \in F$ and $A \in \mathbb{F}_q[S]$, it holds that $f^{\Lambda^{(n)}_A} \in \mathsf{Span}_{\mathbb{F}_q}(F)$. Consequently, SNOVA ideals are $\Lambda^{(n)}_{\mathbb{F}_q[S]}$-stable.

Proof. Suppose $f(\mathbf{x}) = \mathbf{x}^t(\Lambda^{(n)}_{S^i} C \Lambda^{(n)}_{S^j})\mathbf{x} \in F$ and $A = \sum_{h=0}^{l-1} \alpha_h S^h \in \mathbb{F}_q[S]$. Then,

$$\begin{aligned} f^{\Lambda^{(n)}_A}(\mathbf{x}) &= \mathbf{x}^t \Lambda^{(n)}_A \left(\Lambda^{(n)}_{S^i} C \Lambda^{(n)}_{S^j}\right) \Lambda^{(n)}_A \mathbf{x} \\ &= \mathbf{x}^t \left(\sum_{h=0}^{l-1} \alpha_h \Lambda^{(n)}_{S^h}\right) \Lambda^{(n)}_{S^i} C \Lambda^{(n)}_{S^j} \left(\sum_{h=0}^{l-1} \alpha_h \Lambda^{(n)}_{S^h}\right) \mathbf{x} \\ &= \sum_{i=0}^{l-1}\sum_{j=0}^{l-1} \alpha_{i,j} \mathbf{x}^t \Lambda^{(n)}_{S^i} C \Lambda^{(n)}_{S^j} \mathbf{x} \in \mathsf{Span}_{\mathbb{F}_q}(F) \end{aligned}$$

for some $\alpha_{i,j} \in \mathbb{F}_q$. Immediately, it follows that $\langle F \rangle$ is $\Lambda^{(n)}_{\mathbb{F}_q[S]}$-stable. Since SNOVA ideals are generated by several sequences, such as F, each associated with an independent matrix P, they are therefore $\Lambda^{(n)}_{\mathbb{F}_q[S]}$-stable. □

Note that $\Lambda^{(n)}_{\mathbb{F}_q[S]}$ is cyclic but not a diagonal matrix group over the base field. The goal of the following lemma is to describe a matrix P over a field extension $\mathbb{F}_{q^l}$ that allows us to transform a SNOVA ideal into one that is stable under the action of a cyclic and diagonal matrix group, see Proposition 3.

Let $\tau : \mathbb{F}_{q^l} \to \mathbb{F}_{q^l}$ denote the Frobenius map. Abusing notation, we will also denote by τ the function that applies the Frobenius map component-wise to a vector or matrix.

Lemma 1. *The matrix S is diagonalizable on an l-extension of $\mathbb{F}_q$. Specifically, if $\lambda \in \mathbb{F}_{q^l}$ is an eigenvalue of S and $\boldsymbol{\xi} \in \mathbb{F}^l_{q^l}$ its corresponding eigenvector,*

$$P := \left[\boldsymbol{\xi}\ \tau(\boldsymbol{\xi}) \cdots \tau^{l-1}(\boldsymbol{\xi})\right] \in \mathbb{F}^{l\times l}_{q^l} \tag{6}$$

is non-singular and $P^{-1}SP$ is diagonal.

Proof. It suffices to prove that the columns of P are eigenvectors corresponding to distinct eigenvalues. Since the entries of S are in $\mathbb{F}_q$, for $a \in \{0, \ldots, l-1\}$,

$$S\tau^a(\boldsymbol{\xi}) = \tau^a(S\boldsymbol{\xi}) = \tau^a(\lambda\boldsymbol{\xi}) = \lambda^{q^a}\tau^a(\boldsymbol{\xi}),$$

thus $\lambda^{q^a} \in \mathbb{F}_{q^l} \setminus \mathbb{F}_q$ is an eigenvalue of S, corresponding to the eigenvector $\tau^a(\boldsymbol{\xi})$.

Let us show that $\lambda^{q^i} = \lambda^{q^j}$ implies $i \equiv j \pmod{l}$. By [17, Theorem 19.1.], the polynomial $x^{q^l} - x$ is square-free since $x^{q^l} - x$ and its derivative $q^l x^{q^l-1} - 1 = -1$ are relatively prime. For a contradiction, suppose $\lambda^{q^i} = \lambda^{q^j}$ with $i < j$ and let f be the characteristic polynomial of S. Then λ^{q^j} is a root of f with multiplicity at least 2 and $(x - \lambda^{q^j})^2$ divides f. Since f is a monic irreducible polynomial of degree l, f divides $x^{q^l} - x$ [17, Theorem 19.10.] which would imply $x^{q^l} - x$ is not square-free, contradicting the fact that $x^{q^l} - x$ is square-free. It follows that $\lambda^{q^0}, \ldots, \lambda^{q^{l-1}}$ are l distinct eigenvalues of S. □

We can then diagonalize every matrix in $\Lambda^{(n)}_{\mathbb{F}_q[S]}$ to construct a cyclic matrix group.

Proposition 2. *The matrix group $\mathcal{D} := \{\Lambda^{(n)}_{P^{-1}} M \Lambda^{(n)}_P : M \in \Lambda^{(n)}_{\mathbb{F}_q[S]}\}$ is a cyclic diagonal group which is generated by a diagonal matrix $\Lambda^{(n)}_Q \in \mathbb{F}^{ln\times ln}_{q^l}$, where $Q = diag\big(\beta, \beta^q, \ldots, \beta^{q^{l-1}}\big) \in \mathbb{F}^{l\times l}_{q^l}$ for some $\beta \in \mathbb{F}_{q^l}$.*

Proof. Let $\boldsymbol{\xi} \in \mathbb{F}^l_{q^l}$ be an eigenvector of S with corresponding eigenvalue $\lambda \in \mathbb{F}_{q^l}$. For any nonzero $A = a_0S^0 + a_1S + \cdots + a_{l-1}S^{l-1} \in \mathbb{F}_q[S]$,

$$AP = P \cdot \operatorname{diag}\big(\lambda_A, \tau(\lambda_A), \ldots, \tau^{l-1}(\lambda_A)\big) \tag{7}$$

where $\lambda_A = a_0 + a_1\lambda + \cdots + a_{l-1}\lambda^{l-1}$. Therefore, $P^{-1}AP$ is a diagonal matrix.

Let B be a generator of the multiplicative group $\mathbb{F}_q[S]^\times$. Clearly, $\Lambda^{(n)}_{P^{-1}}\Lambda^{(n)}_B\Lambda^{(n)}_P$ is a generator of $\mathcal{D}$ and $\boldsymbol{\xi}$ is an eigenvector of B. Let $\beta \in \mathbb{F}_{q^l}$ be the eigenvalue of B associated with $\boldsymbol{\xi}$. Then, $P^{-1}BP = \operatorname{diag}\big(\beta, \tau(\beta), \ldots, \tau^{l-1}(\beta)\big)$. Therefore, $\Lambda^{(n)}_{P^{-1}}\Lambda^{(n)}_B\Lambda^{(n)}_P = \Lambda^{(n)}_Q$, where $Q = \operatorname{diag}\big(\tau^0(\beta), \ldots, \tau^{l-1}(\beta)\big)$. □

Applying an appropriate change of variable to the ideal I, yields a stable ideal under the action of a cyclic diagonal group.

Proposition 3. *Let $S \in \mathbb{F}_q^{l\times l}$ be a symmetric matrix with irreducible characteristic polynomial. If $J \subset \mathbb{F}_q[x_1,\dots,x_{ln}]$ is $\Lambda^{(n)}_{\mathbb{F}_q[S]}$-stable, then there exist a matrix $P \in \mathbb{F}_{q^l}^{l\times l}$ such that the ideal*

$$J^{\Lambda_P^{(n)}} := \left\{ f^{\Lambda_P^{(n)}} : f \in J \right\} \subseteq \mathbb{F}_{q^l}[x_1,\dots,x_{ln}] \tag{8}$$

is $\mathcal{D}$-stable, where $\mathcal{D}$ is the diagonal group of matrices defined as in Proposition 2.

Proof. Let P be a matrix defined as in Lemma 1 and $\mathcal{D}$ is the diagonal group of matrices defined as in Proposition 2. Let $f \in J$, $g = f^{\Lambda_P^{(n)}} \in J^{\Lambda_P^{(n)}}$, $M \in \Lambda^{(n)}_{\mathbb{F}_q[S]}$, and $D = \Lambda^{(n)}_{P^{-1}} M \Lambda^{(n)}_P \in \mathcal{D}$. Then,

$$g^D(\mathbf{x}) = g(D\cdot\mathbf{x}) = f(\Lambda_P^{(n)}\cdot D\cdot\mathbf{x}) = f(M\cdot\Lambda_P^{(n)}\cdot\mathbf{x}) = f^M(\Lambda_P^{(n)}\cdot\mathbf{x}).$$

Since J is $\Lambda^{(n)}_{\mathbb{F}_q[S]}$-stable, then $f^M \in J$, and $g^D = (f^M)^{\Lambda_P^{(n)}} \in J^{\Lambda_P^{(n)}}$. Therefore $J^{\Lambda_P^{(n)}}$ is $\mathcal{D}$-stable. □

The group action of the cyclic diagonal group induces a new degree, namely $\mathcal{D}$-degree, on $\mathbb{F}_{q^l}[x_1,\dots,x_{nl}]$ that is compatible with the usual degree.

Definition 4 ($\mathcal{D}$-degree). *Let β, $Q = diag(\beta, \beta^q, \dots, \beta^{q^{l-1}}) \in \mathbb{F}_{q^l}^{l\times l}$ and $\mathcal{D} = \langle \Lambda_Q^{(n)} \rangle$ be as in Proposition 2. For a monomial $\mu = x_1^{\alpha_1}\cdots x_{nl}^{\alpha_{nl}}$ in $\mathbb{F}_{q^l}[x_1,\dots,x_{nl}]$, we have*

$$\begin{aligned}\mu^{\Lambda_Q^{(n)}} &= (\beta x_1)^{\alpha_1}\cdots(\beta^{q^{l-1}}x_l)^{\alpha_l}(\beta x_{l+1})^{\alpha_{l+1}}\cdots(\beta^{q^{l-1}}x_{ln})^{\alpha_{ln}}\\ &= \beta^{\sum_{j=1}^{l} q^{j-1}\cdot\sum_{i=1}^{n}\alpha_{(i-1)\cdot l+j}}\mu.\end{aligned}$$

Then, we define the $\mathcal{D}$-degree of μ as

$$\deg_{\mathcal{D}}(\mu) = \sum_{j=1}^{l} q^{j-1}\cdot\sum_{i=1}^{n}\alpha_{(i-1)\cdot l+j} \mod q^l-1.$$

Remark 1 ($\mathcal{D}$-degrees as multi-degrees in $\mathbb{Z}^l_{\geq 0}$). Consider the partition $P = (X_1,\dots,X_l)$ of the set $X = \{x_1,\dots,x_{nl}\}$, where $X_r = \{x_r, x_{l+r}, \cdots, x_{(n-1)l+r}\}$ for each $r \in [l]$. On the one hand, for any $x \in X$, $\deg_{\mathcal{D}}(x) = q^{r-1}$ if and only if $x \in X_r$. On the other hand, the multi-degree of $x \in X_r$ with respect to the partition P is the canonical vector $\mathbf{e}_r \in \mathbb{Z}^l_{\geq 0}$. This establishes a one-to-one correspondence of $\mathcal{D}$-degrees of linear monomials and their corresponding multi-degrees with respect to P. Similarly, for quadratic monomials x_ix_j, with $x_i \in X_{r_1}$ and $x_j \in X_{r_2}$, the multi-degree is given by $\mathbf{e}_{r_1} + \mathbf{e}_{r_2} \in \mathbb{Z}^l_{\geq 0}$ while $\deg_{\mathcal{D}}(x_ix_j) = q^{(r_1-1)} + q^{(r_2-1)} \mod q^l - 1$.

Example 1. With $l = 2$ and $q > 2$, there are exactly three nonzero $\mathcal{D}$-degrees for quadratic polynomials in $x_1, \ldots, x_{ln}$. For example, if $q = 16$, these are 2, 17, and 32. For any $i, j = 1, \ldots, ln$, we have

$$\deg_{\mathcal{D}}(x_i x_j) = \begin{cases} 2 & \text{if } i \equiv j \equiv 1 \mod l \\ 17 & \text{if } i \not\equiv j \mod l \\ 32 & \text{if } i \equiv j \equiv 0 \mod l \end{cases}$$

With $n = 2$, the variables are x_1, x_2, x_3, x_4 and the quadratic monomials of each $\mathcal{D}$-degree and its corresponding multi-degrees in $\mathbb{Z}^l_{\geq 0}$ are given in the Table 2.

$\mathcal{D}$-degree	Multi-degree	Monomials
2	(2,0)	x_1^2, x_1x_3, x_3^2
17	(1,1)	$x_1x_2, x_1x_4, x_2x_3, x_3x_4$
32	(0,2)	x_2^2, x_2x_4, x_4^2

Table 2. Correspondence of $\mathcal{D}$-degrees and multi-degrees of quadratic polynomials for $n = 2$ and $l = 2$.

It readily follows that the multi-degree induces a grading on $\mathbb{F}_{q^l}[x_1, \ldots, x_{nl}]$.

Definition 5 (Grading). *Let R be a ring, G an abelian group, and $R = \oplus_{i \in G} R_i$ a direct sum decomposition of an abelian group. R is graded (G-graded) if $R_i R_j \subset R_{i+j}$ for all $i, j \in G$.*

It is also easy to see that the multi-degree refines the usual degree. We can then refer to the multi-homogeneous components of a polynomial.

Definition 6. *A polynomial f in $\mathbb{F}_{q^l}[x_1, \ldots, x_{nl}]$ is said to be multi-homogeneous if all its monomials have the same multi-degree.*

Example 2. For $l = 4$, the multi-homogeneous components of the polynomial

$$x_1x_2 + x_3x_4 + x_{l+1}x_{l+2} + x_{l+3}x_{l+4} + \cdots + x_{(n-2)l+1}x_{(n-2)l+2} + x_{(n-1)l+3}x_{(n-1)l+4}.$$

are

$$x_1x_2 + x_{l+1}x_{l+2} + \cdots + x_{(n-1)l+1}x_{(n-1)l+2}$$

and

$$x_3x_4 + x_{l+3}x_{l+4} + \cdots + x_{(n-1)l+3}x_{(n-1)l+4}.$$

Definition 7. *An ideal $J \subset \mathbb{F}_{q^l}[x_1, \ldots, x_{ln}]$ is said to be multi-homogeneous if for any polynomial $f \in J$, all its multi-homogeneous components are in J.*

The following theorem was proven by Faugère and Svartz [7] using terminology of $\mathcal{D}$-degrees and for generic $\mathcal{D}$-stable ideal. Here, we rewrite it using multi-degrees for the case $\mathcal{D}$ ideal generated by quadratic polynomials.

Theorem 1. *Let* $Q = diag(\beta, \beta^q, \ldots, \beta^{q^{l-1}}) \in \mathbb{F}_{q^l}^{l\times l}$ *and* $\mathcal{D} = \langle \Lambda_Q^{(n)} \rangle$ *be as in Proposition 2. An ideal* J *generated by quadratic polynomials is* $\mathcal{D}$*-stable if and only if* J *is multi-homogeneous.*

Proof. ($\Leftarrow$) Suppose J is a multi-homogeneous ideal and $g = h_1 + \cdots + h_e \in J$, where the h_i are the multi-homogeneous components of g. Now, for each h_i it holds that, for some $\lambda \in \mathbb{F}_{q^l}$,

$$h_i^{\Lambda_Q^{(n)}} = \lambda \cdot h_i.$$

Since J is multi-homogeneous, we obtain that $h_i^{\Lambda_Q^{(n)}} \in J$. Therefore,

$$g^{\Lambda_Q^{(n)}} = h_1^{\Lambda_Q^{(n)}} + \cdots + h_e^{\Lambda_Q^{(n)}} \in J.$$

($\Rightarrow$) Fix a quadratic generator $g \in J$. For each $r_1, r_2 \in [l]$, let h_{r_1,r_2} be the multi-homogeneous component of g of multi-degree $\mathbf{e}_{r_1} + \mathbf{e}_{r_2} \in \mathbb{Z}_{\geq 0}^l$. Note that

$$h_{r_1,r_2}^{\Lambda_P^{(n)}} = \beta^{d(r_1,r_2)} \cdot h_{r_1,r_2},$$

where $d(r_1, r_2) := q^{r_1-1} + q^{r_2-1}$. Set $D = \{d(r_1, r_2) : r_1, r_2 \in [l]\}$ and $e = q^l - 1$. For each $i = 0, \ldots, e-1$, define $h_i = 0$ if $i \notin D$, otherwise define $h_{d(r_1,r_2)} = h_{r_1,r_2}$. Therefore,

$$\begin{bmatrix} g \\ g^{\Lambda_Q^{(n)}} \\ \vdots \\ g^{(\Lambda_Q^{(n)})^{e-1}} \end{bmatrix} = \begin{bmatrix} 1 & 1 & \cdots & 1 \\ 1 & \beta & \cdots & \beta^{e-1} \\ \vdots & \vdots & \vdots & \vdots \\ 1 & \beta^{e-1} & \ldots & \beta^{(e-1)(e-1)} \end{bmatrix} \begin{bmatrix} h_0 \\ h_1 \\ \vdots \\ h_{e-1} \end{bmatrix}, \tag{9}$$

and since J is stable under $\mathcal{D}$, then $g^{(\Lambda_Q^{(n)})^d} \in J$, for each $d = 0, \ldots, e-1$. Note that the matrix of (9) is a Vandermonde matrix $V = [\beta^{(i-1)(d-1)}]_{1\leq i,d\leq e}$. It is known that $\det(V) = \prod_{1\leq i<d\leq e}(\beta^{d-1} - \beta^{i-1}) \neq 0$, since β is of order e. Therefore, V is nonsingular, and each homogeneous component is in J. □

Corollary 1. *Let* $Q = diag(\beta, \beta^q, \ldots, \beta^{q^{l-1}}) \in \mathbb{F}_{q^l}^{l\times l}$ *and* $\mathcal{D} = \langle \Lambda_Q^{(n)} \rangle$ *be as in Proposition 2. Let* $g_1, \ldots, g_s \in \mathbb{F}_{q^l}[x_1, \ldots, x_{nl}]$ *be quadratic polynomials such that* $\langle g_1, \ldots, g_s \rangle$ *is a* $\mathcal{D}$*-stable ideal and*

$$g_i^{\Lambda_Q^{(n)}} \in \mathsf{Span}_{\mathbb{F}_q}(g_1, \ldots, g_s), \text{ for each } i \in [s]. \tag{10}$$

Then,

$$\mathsf{Span}_{\mathbb{F}_{q^l}}(g_1, \ldots, g_s) = \mathsf{Span}_{\mathbb{F}_{q^l}}(\mathcal{H}),$$

where $\mathcal{H} = \{h_i^{(r_1,r_2)} : i \in [o], r_1, r_2 \in [l]\}$ *and* $h_i^{(r_1,r_2)}$ *denotes the multi-homogeneous component of* g_i *of multi-degree* $\mathbf{e}_{r_1} + \mathbf{e}_{r_2}$*. Consequently, if* $g_1, \ldots, g_s$ *are linearly independent, then there is an invertible* $B \in \mathbb{F}_q^{s\times s}$ *and polynomials* $\tilde{h}_1, \ldots, \tilde{h}_s \in \mathcal{H}$ *such that*

$$(g_1, \ldots, g_s)^t = B \cdot (\tilde{h}_1, \ldots, \tilde{h}_s)^t,$$

Proof. We set $e = q^l - 1$, $\mathcal{V} = \mathsf{Span}_{\mathbb{F}_{q^l}}(g_1, \ldots, g_s)$, and $\mathcal{W} = \mathsf{Span}_{\mathbb{F}_{q^l}}(\mathcal{H})$. Clearly, $\mathcal{V} \subset \mathcal{W}$, so we focus on proving that $\mathcal{W} \subset \mathcal{V}$.

By Eq. (9) in Theorem 1, for any $i \in [s]$, we obtain that

$$\{h_i^{(r_1,r_2)} : r_1, r_2 \in [l]\} \subset \mathsf{Span}_{\mathbb{F}_{q^l}}(g_i^{\Lambda_Q^{(n)}}, \ldots, g_i^{\Lambda_Q^{(n)}}).$$

By hypothesis in Eq. (10), we obtain $\mathcal{H} \subset \mathsf{Span}_{\mathbb{F}_q}(g_1, \ldots, g_s)$. Therefore, $\mathcal{W} \subset \mathcal{V}$. □

3.2 Distribution of the Basis of Multi-Homogeneous Components

In this section, we demonstrate that after the change of variables defined by the matrix $\Lambda_P^{(n)}$, a SNOVA ideal is generated by multi-homogeneous polynomials, as shown in Proposition 4, with the number of polynomials at each multi-degree precisely determined, as detailed in Theorem 2.

Proposition 4. *Let P be a matrix of eigenvectors of S, as described in Lemma 1, and $Q = diag(\beta, \beta^q, \ldots, \beta^{q^{l-1}}) \in \mathbb{F}_{q^l}^{l \times l}$ and $\mathcal{D} = \langle \Lambda_Q^{(n)} \rangle$ as in Proposition 2. Let $F = (f_1, \ldots, f_{ol^2}) \in \mathbb{F}_q[x_1, \ldots, x_{ln}]$ be a SNOVA sequence. Hence, if we set $\mathcal{H} = \{h_i^{(r_1,r_2)} : i \in [o], r_1, r_2 \in [l]\}$, then*

$$\mathsf{Span}_{\mathbb{F}_{q^l}}(f_1^{\Lambda_P^{(n)}}, \ldots, f_{ol^2}^{\Lambda_P^{(n)}}) = \mathsf{Span}_{\mathbb{F}_{q^l}}(\mathcal{H}),$$

where $h_i^{(r_1,r_2)}$ denotes the multi-homogeneous component of $f_i^{\Lambda_P^{(n)}}$ of multi-degree $\mathbf{e}_{r_1} + \mathbf{e}_{r_2}$. Consequently, if $f_1, \ldots, f_{ol^2}$ are linearly independent, then there exists an invertible block-diagonal matrix $B \in \mathbb{F}_q^{ol^2 \times ol^2}$ with blocks of size $l^2 \times l^2$ and multi-homogeneous polynomials $\tilde{h}_1, \ldots, \tilde{h}_{ol^2} \in \mathcal{H}$ such that

$$(f_1^{\Lambda_P^{(n)}}, \ldots, f_{ol^2}^{\Lambda_P^{(n)}})^t = B \cdot (\tilde{h}_1, \ldots, \tilde{h}_{ol^2})^t,$$

Proof. Suppose $\Lambda_Q^{(n)} = \Lambda_{P^{-1}}^{(n)} \Lambda_A^{(n)} \Lambda_P^{(n)}$, where $A \in \mathbb{F}_q[S]$. By Proposition 1, $f^{\Lambda_A^{(n)}} \in \mathsf{Span}_{\mathbb{F}_q}(F)$ for each $f \in F$.

First, we prove the case $o = 1$. Set $m = l^2$, for any $i \in [m]$, it follows that

$$\begin{aligned}
\left(f_i^{\Lambda_P^{(n)}}\right)^{\Lambda_Q^{(n)}} &= \left(f_i^{\Lambda_P^{(n)}}\right)^{\Lambda_{P^{-1}}^{(n)} \Lambda_A^{(n)} \Lambda_P^{(n)}} \\
&= f_i^{\Lambda_A^{(n)} \Lambda_P^{(n)}} \\
&= (a_1 f_1 + \cdots + a_m f_m)^{\Lambda_P^{(n)}} \\
&= a_1 f_1^{\Lambda_P^{(n)}} + \cdots + a_m f_m^{\Lambda_P^{(n)}} \in \mathsf{Span}_{\mathbb{F}_{q^l}}(f_1^{\Lambda_P^{(n)}}, \ldots, f_m^{\Lambda_P^{(n)}}),
\end{aligned}$$

where $a_1, \ldots, a_m \in \mathbb{F}_q$ satisfy that $f_i^{\Lambda_A^{(n)}} = a_1 f_1 + \cdots + a_m f_m$. Hence, for the case $o = 1$, our result follows directly from Corollary 1.

The case $o > 1$ follows from the fact that a SNOVA sequence F with ol^2 polynomials can be written as $F = (F_1, \ldots, F_o)$, where each F_i is a SNOVA sequence with the same number of variables and l^2 polynomials. Indeed, to simplify the notation, we set

$$G = (G_1, \ldots, G_o), \text{ where } G_i = F_i^{\Lambda_P^{(n)}} \text{ for each } i \in [o]. \tag{11}$$

Above we proved that, for each $i \in [o]$, there are multi-homogenous polynomials $h_i^{(1,1)}, \ldots, h_i^{(l,l)}$ such that

$$\mathsf{Span}_{\mathbb{F}_q}(G_i) = \mathsf{Span}_{\mathbb{F}_{q^l}}(h_i^{(1,1)}, \ldots, h_i^{(l,l)}).$$

Moreover, if the polynomials in F_i are linearly independent, then there is an invertible matrix $B_i \in \mathbb{F}_q^{l^2 \times l^2}$ such that

$$F_i = B_i \cdot (\tilde{h}_{i,1}, \ldots, \tilde{h}_{i,l^2})^t.$$

Thus, $\mathsf{Span}_{\mathbb{F}_{q^l}}(G) = \mathsf{Span}_{\mathbb{F}_{q^l}}(h_1^{(1,1)}, \ldots, h_o^{(l,l)})$. Moreover, if the polynomials in F are linearly independent, then the invertible block-diagonal matrix $B = \mathrm{diag}(B_1, \ldots, B_o) \in \mathbb{F}_q^{ol^2 \times ol^2}$ satisfies that $G = B \cdot (\tilde{h}_{1,1}, \ldots, \tilde{h}_{o,l^2})^t$. □

Now, given $r_1, r_2 \in [l]$, we focus on determining the dimension of the $\mathbb{F}_{q^l}$ vector space generated by all the multi-homogeneous components of multi-degree $\mathbf{e}_{r_1} + \mathbf{e}_{r_2}$ of the polynomials in a SNOVA sequence after the change of variables defined by $\Lambda_P^{(n)}$, that is, G in Eq. (11). More precisely, with the notation as in Proposition 4, we want to determine the dimension of

$$\mathcal{H}_{r_1,r_2} := \mathsf{Span}_{\mathbb{F}_{q^l}}(h_1^{(r_1,r_2)}, h_2^{(r_1,r_2)}, \ldots, h_{ol^2}^{(r_1,r_2)}).$$

We will prove that

$$\mathsf{dim}(\mathcal{H}_{r_1,r_2}) \leq \begin{cases} o & \text{if } r_1 = r_2, \\ 2o & \text{otherwise.} \end{cases} \tag{12}$$

An equality in the above equation implies the following theorem.

Theorem 2. *Let P be a matrix of eigenvectors of S, as described in Lemma 1, and $Q = diag(\beta, \beta^q, \ldots, \beta^{q^{l-1}}) \in \mathbb{F}_{q^l}^{l \times l}$ and $\mathcal{D} = \langle \Lambda_Q^{(n)} \rangle$ as in Proposition 2. Let $F = (f_1, \ldots, f_{ol^2}) \in \mathbb{F}_q[\mathbf{x}]$ be a SNOVA sequence. Suppose we have an equality in (12). Then, the vector space*

$$\mathsf{Span}_{\mathbb{F}_{q^l}}(f_1^{\Lambda_P^{(n)}}, \ldots, f_{ol^2}^{\Lambda_P^{(n)}})$$

has a basis G of multi-homogeneous polynomials with the following distribution of multi-degrees:

- *For each $r \in [l]$, G contains o polynomials of multi-degree $2\mathbf{e}_r$.*

– *For each* $r_1 \neq r_2 \in [l]$, G *contains* $2o$ *polynomials of multi-degree* $\mathbf{e}_{r_1} + \mathbf{e}_{r_2}$.

Proof. By Proposition 4, $\mathsf{Span}_{\mathbb{F}_{q^l}}(f_1^{\Lambda_P^{(n)}}, \ldots, f_{ol^2}^{\Lambda_P^{(n)}}) = \mathsf{Span}_{\mathbb{F}_{q^l}}(h_1^{(1,1)}, \ldots, h_{ol^2}^{(l,l)})$, where $h_i^{(r_1,r_2)}$ denotes the multi-degree $\mathbf{e}_{r_1} + \mathbf{e}_{r_2}$ homogeneous component of $f_i^{\Lambda_P^{(n)}}$. Hence, we can write

$$\mathsf{Span}_{\mathbb{F}_{q^l}}(f_1^{\Lambda_P^{(n)}}, \ldots, f_{ol^2}^{\Lambda_P^{(n)}}) = \mathsf{Span}_{\mathbb{F}_{q^l}}(\mathcal{H}_{1,1} \cup \mathcal{H}_{1,2} \cup \cdots \cup \mathcal{H}_{l,l}),$$

where $\mathcal{H}_{r_1,r_2} = \mathsf{Span}_{\mathbb{F}_{q^l}}(h_1^{(r_1,r_2)}, h_2^{(r_1,r_2)}, \ldots, h_{ol^2}^{(r_1,r_2)})$.

Now, assuming the equality in (12), we obtain that, for each $r \in [l]$, the vector space $\mathcal{H}_{r,r}$ is generated by o polynomials, namely $\tilde{h}_1^{(r,r)}, \ldots, \tilde{h}_o^{(r,r)}$ of multi-degree $2\mathbf{e}_r$. Likewise, for each $r_1 \neq r_2 \in [l]$, $\mathcal{H}_{r_1,r_2}$ is generated by $2o$ polynomials, namely $\tilde{h}_1^{(r_1,r_2)}, \ldots, \tilde{h}_{2o}^{(r_1,r_2)}$ of multi-degree $\mathbf{e}_{r_1} + \mathbf{e}_{r_2}$. Finally, define G as the set that contains all the $\tilde{h}_i^{(r_1,r_2)}$ polynomials for $i \in [o]$ and $r_1, r_2 \in [l]$. □

Remark 2. We have experimentally verified that the inequality in (12) holds as an equality with overwhelming probability, i.e. with a value very close to 1.

We now focus on proving the statement in Eq. (12). A lifted SNOVA sequence G consisting of ol^2 polynomials is the aggregation of o lifted SNOVA sequences $G_1, \ldots, G_o$, each containing l^2 equations. Therefore, it suffices to prove Eq. (12) for the case $o = 1$.

Let F be a SNOVA sequence with l^2 polynomials, given by

$$\mathbf{x}^t \Lambda_{S^i}^{(n)} C \Lambda_{S^j}^{(n)} \mathbf{x}$$

for $i, j \in \{0, \ldots, l-1\}$.

Let $\lambda \in \mathbb{F}_{q^l}$ be the eigenvalue of S and $P \in \mathbb{F}_{q^l}^{l \times l}$ be as in Lemma 1. Additionally, suppose $f_{i,j}(\mathbf{x}) = \mathbf{x}^t \Lambda_{S^i}^{(n)} C \Lambda_{S^j}^{(n)} \mathbf{x}$. Then,

$$f_{i,j}^{\Lambda_P^{(n)}}(\mathbf{x}) = \mathbf{x}^t \Lambda_{P^t S^i}^{(n)} C \Lambda_{S^j P}^{(n)} \mathbf{x},$$

$P^t S^i = \mathrm{diag}(\lambda^{i \cdot q^0}, \ldots, \lambda^{i \cdot q^{l-1}}) P^t$ and $S^j P = \mathrm{diag}((\lambda^{j \cdot q^0}, \ldots, \lambda^{j \cdot q^{l-1}}) P$, so that with $\Lambda_{P^t}^{(n)} C \Lambda_P^{(n)} = [t_{i,j}]_{i,j \in [ln]} \in \mathbb{F}_{q^l}^{ln \times ln}$, we can write

$$g_{i,j}(\mathbf{x}) := f_{i,j}^{\Lambda_P^{(n)}}(\mathbf{x}) \quad = \sum_{a,b \in \{0,\ldots,n-1\}} \sum_{r_1, r_2 \in [l]} \lambda^{i \cdot q^{r_1 - 1} + j \cdot q^{r_2 - 1}} \cdot t_{al + r_1, bl + r_2} x_{al + r_1} x_{bl + r_2},$$

where $\mathbf{x} = (x_1, \ldots, x_{ln})$.

As we saw in Remark 1, the multi-degree of a quadratic monomial is the sum to two canonical vectors $\mathbf{e}_{r_1} + \mathbf{e}_{r_2} \in \mathbb{Z}_{\geq 0}^l$. Assuming $h_{i,j}^{(r_1,r_2)}$ denotes the multi-homogeneous component of $g_{i,j}$ of multi-degree $\mathbf{e}_{r_1} + \mathbf{e}_{r_2}$, we can write

$$g_{i,j} = \sum_{r_1, r_2 \in [l]} h_{i,j}^{(r_1,r_2)} \text{ and } \mathcal{H}_{r_1,r_2} = \left\{ h_{i,j}^{(r_1,r_2)} : i, j \in \{0, \ldots, l-1\} \right\}.$$

For each $r \in [l]$,

$$h_{i,j}^{(r,r)} = \lambda^{(i+j)\cdot q^{r-1}} \cdot \sum_{a=0}^{n-1}\sum_{b=0}^{n-1} t_{al+r,bl+r} x_{al+r} x_{bl+r} = \lambda^{(i+j)\cdot q^{r-1}} h_{0,0}^{(r,r)}.$$

Thus, $\mathsf{Span}_{\mathbb{F}_{q^l}}(\mathcal{H}_{r,r}) = \mathsf{Span}_{\mathbb{F}_{q^l}}(h_{0,0}^{(r,r)})$, which implies $\mathsf{dim}(\mathcal{H}_{r,r}) \leq 1$.

Likewise, given $r_1 \neq r_2 \in [l]$, for each $i, j \in \{0, \ldots, l-1\}$, we can write

$$h_{i,j}^{(r_1,r_2)} = \hat{h}_{i,j}^{(r_1,r_2)} + \tilde{h}_{i,j}^{(r_1,r_2)},$$

where

$$\hat{h}_{i,j}^{(r_1,r_2)} := \lambda^{i\cdot q^{r_1-1}+j\cdot q^{r_2-1}} \cdot \sum_{a=0}^{n-1}\sum_{b=0}^{n-1} t_{al+r_1,bl+r_2} x_{al+r_1} x_{bl+r_2},$$

and

$$\tilde{h}_{i,j}^{(r_1,r_2)} = \lambda^{i\cdot q^{r_2-1}+j\cdot q^{r_1-1}} \cdot \sum_{a=0}^{n-1}\sum_{b=0}^{n-1} t_{al+r_2,bl+r_1} x_{al+r_1} x_{bl+r_2}.$$

Thus, $\mathsf{Span}_{\mathbb{F}_{q^l}}(\mathcal{H}_{r_1,r_2}) = \mathsf{Span}_{\mathbb{F}_{q^l}}(\hat{h}_{0,0}^{(r_1,r_2)}, \tilde{h}_{0,0}^{(r_1,r_2)})$, which implies $\mathsf{dim}(\mathcal{H}_{r_1,r_2}) \leq 2$. This finalizes the proof of (12) for the case $o = 1$.

4 Solving SNOVA Systems

Now we focus on the complexity of solving SNOVA systems, i.e. systems over $\mathbb{F}_q[x_1, \ldots, x_{ln}]$ of the form

$$\mathbf{x}^t \big(\Lambda_{S^{i-1}}^{(n)} P_k \Lambda_{S^{j-1}}^{(n)}\big)\mathbf{x} = z_k^{(i,j)}, \quad \text{for } (i,k,j) \in [l] \times [o] \times [l], \tag{13}$$

where $\mathbf{x} = (x_1, \ldots, x_{ln})^t$, $P_k \in \mathbb{F}_q^{ln \times ln}$ is a matrix, and $\Lambda_A^{(n)}$ and $S \in \mathbb{F}_q^{l\times l}$ are as in the description of SNOVA.

In the homogeneous case, i.e. $z_k^{(i,j)} = 0$, if $ln \leq l^2 o$, there is a naive way to improve generic solving algorithms. As observed in [10], in this case, if there is a solution $\mathbf{s}$ to the SNOVA system, then $\Lambda_{S^i}^{(n)}\mathbf{s}$ is also a solution for each $i \in [l-1]$. Hence, there is either no solution or an l-dimensional vector space of solutions due to the fact that each P_k has a UOV-like structure. In this scenario, one can remove l variables and search for a solution of the form $(x_1, \ldots, x_{ln-l}, 1, 0, \ldots, 0)$. Thus, the complexity of solving the SNOVA system would be at least $O(q^l)$ times faster than a random quadratic system with the same dimensions. This approach is less effective when $ln > l^2 o$, because we can only specialize $ln - l^2 o$ variables for free in the random case.

The algorithm we propose in this section improves the naive approach in both the underdefined ($ln > l^2 o$) and the overdefined cases ($ln \leq l^2 o$). We focus on estimating the speed up in the underdefined case, because all the proposed parameters for SNOVA fall into this case, see Table 1.

4.1 Solving the System of Homogeneous Components

As we will explain in Section 4.2, our strategy to solve underdefined SNOVA systems reduces to solving a multi-homogeneous system. We start by explaining how to solve such a system. More precisely, it is a system of $m = l^2o$ equations

$$h_1^{(1,1)}(\tilde{\mathbf{x}}) = w_1^{(1,1)}, h_1^{(1,2)}(\tilde{\mathbf{x}}) = w_1^{(1,2)}, \ldots, h_o^{(l,l)}(\tilde{\mathbf{x}}) = w_o^{(l,l)}, \tag{14}$$

over $\mathbb{F}_{q^l}[\tilde{x}_{1,1}, \tilde{x}_{2,1}, \ldots, \tilde{x}_{l,n}]$ with the following properties:

1. Every polynomial $h_k^{(i,j)}(\tilde{\mathbf{x}})$ is multi-homogeneous with respect to the partition of the variables $\tilde{\mathbf{x}}_1, \tilde{\mathbf{x}}_2, \ldots, \tilde{\mathbf{x}}_l$, where $\tilde{\mathbf{x}}_r := (\tilde{x}_{r,1}, \ldots, \tilde{x}_{r,n})$.
2. For each $r \in [l]$, the o polynomials $h_1^{(r,r)}, \ldots, h_o^{(r,r)}$ have multi-degree $2\mathbf{e}_r$, where $\mathbf{e}_i \in \mathbb{F}_q^l$ is the r-th canonical vector, that is, they only involve variables from $\mathbf{x}_r := (x_{r,1}, \ldots, x_{r,n})$.
3. For each $r_1 \neq r_2 \in [l]$, the $2o$ polynomials $h_1^{(r_1,r_2)}, \ldots, h_o^{(r_1,r_2)}, h_1^{(r_1,r_2)}, \ldots, h_o^{(r_1,r_2)}$ have multi-degree $\mathbf{e}_{r_1} + \mathbf{e}_{r_2}$, that is, they are bilinear in the sets $\mathbf{x}_{r_1}$ and $\mathbf{x}_{r_2}$.

We aim at finding a solution to Eq. (14) in the big field that corresponds to a solution in the small field. Assuming that the system is underdefined, i.e. $ln > l^2o$, we can "guess" $ln - l^2o$ variables for "free". We then sample k entries in the small field and lift them to k/l entries in the big field, which we use to partially evaluate the polynomials and try to solve the resulting system. If we find a solution for the remaining variables corresponding to a vector in the small field, we are done; otherwise, we sample again. The following steps precisely describe the algorithm, which is parameterized by a nonnegative integer k that is a multiple of l.

1. Sample $(s_{ol^2+1}, \ldots, s_{ln})^t \in \mathbb{F}_q^{(n-ol)l}$.
2. Sample $(s_{ol^2-k+1}, \ldots, s_{ol^2})^t \in \mathbb{F}_q^k$ and compute

$$\tilde{\mathbf{s}}_2 := \Lambda_{P^{-1}}^{(n-ol+\frac{k}{l})} \cdot (s_{ol^2-k+1}, \ldots, s_{ln})^t.$$

3. Use the XL algorithm [5] as described in Section 2.5 to find $\tilde{\mathbf{s}}_1 \in \mathbb{F}_{q^l}^{ol^2-k}$ such that [6]

$$h_1(\tilde{\mathbf{s}}_1, \tilde{\mathbf{s}}_2) = w_1, \ldots, h_m(\tilde{\mathbf{s}}_1, \tilde{\mathbf{s}}_2) = w_m.$$

   If no solution is found, go back to Step 2.
4. Set $\mathbf{s} = \Lambda_P^{(n)}(\tilde{\mathbf{s}}_1^t, \tilde{\mathbf{s}}_2^t)^t$. If $\mathbf{s} \in \mathbb{F}_q^{ln}$, output $\mathbf{s}$. Otherwise, go to Step 2.

[5] It might be possible to achieve similar performance using other Groebner basis algorithms. For example, one could use Faugere's F4 with a monomial ordering compatible with the multi-degree and design a criterion to pick S-polynomials based on the multi-degree. However, the complexity analysis of the XL approach is cleaner.

[6] We abuse notation by enumerating the $h_k^{(r_1,r_2)}$ as $h_1, \ldots, h_m$.

Note that, if $\tilde{\mathbf{s}}_2$ is part of a solution that corresponds to a vector in the small field, since the system $h_1(\mathbf{y}, \tilde{\mathbf{s}}_2) = w_1, \ldots, h_m(\mathbf{y}, \tilde{\mathbf{s}}_2) = w_m$ is overdefined, one expects it to have only a few solutions, hence we expect to find $\tilde{\mathbf{s}}_1$ by solving such a partially evaluated system only a few times.

The number of $\mathbb{F}_{q^l}$-multiplications of the algorithm described above is upper bounded by

$$\min_{k \in [ol^2],\ l \mid k} q^k \cdot 3\Big(ol - \frac{k}{l}\Big)^2 \cdot \big[\overline{\mathcal{M}}(\mathbf{d}_{sol})\big]^2, \tag{15}$$

where $\mathbf{d}_{sol} \in \mathbb{Z}^l_{\geq 0}$ minimizes $\overline{\mathcal{M}}(\mathbf{d}_{sol})$, which is the number of monomials of multi-degree smaller than $\mathbf{d}_{sol}$, and there exists $(d_1, \ldots, d_l) \leq \mathbf{d}_{sol}$ such that the coefficient of $t_1^{d_1} t_2^{d_2} \cdots t_l^{d_l}$ in the series

$$\frac{\prod_{1 \leq i < j \leq l} \left(1 - t_i t_j\right)^{2o} \cdot \prod_{i=1}^{l} \left(1 - t_i^2\right)^o}{\prod_{i=1}^{l} (1 - t_i)^{ol - k/l + 1}} \tag{16}$$

is nonnegative.

In order for (16) to accurately predict the corank of the Macaulay matrix, we need to make genericity assumptions akin to Assumptions 1 and 2 in [15]. Although it is outside of the scope of this paper to formulate or test such assumptions, we have run the algorithm for several random instances to check whether the submatrix of the Macaulay matrix produced in Step 3 has the corank predicted by (16). The results are presented towards the end of Section 4.2 (see Table 3), and they suggest that the prediction is accurate with high probability.

4.2 Solving Underdefined SNOVA Systems

To simplify the notation, let us write the SNOVA system given in Eq. (13) as

$$f_1(x_1, \ldots, x_{ln}) = z_1, \ldots, f_{ol^2}(x_1, \ldots, x_{ln}) = z_{ol^2}.$$

By using Theorem 2, we transform this system into a multi-homogeneous system of the form (14) over the extension field and use the procedure described in Section 4.1 to find a solution that corresponds to a solution in the small field. The following steps precisely describe the algorithm, which is parameterized by a nonnegative k multiple of l.

1. For each $i \in [ol^2]$, compute $f_i^{\Lambda_P^{(n)}}$, where P is the matrix of eigenvectors of S, as described in Lemma 1.
2. Compute a basis $(h_1^{(1,1)}, h_2^{(1,2)}, \ldots, h_o^{(l,l)}) \subset \mathcal{H}$ of the vector space spanned by the $f_i^{\Lambda_P^{(n)}}$, where $\mathcal{H}$ is the set of multi-homogeneous components of the $f_i^{\Lambda_P^{(n)}}$. By Proposition 4, such a basis exists.
3. Compute the invertible matrix $B \in \mathbb{F}_{q^l}^{m \times m}$ such that

$$(f_1^{\Lambda_P^{(n)}}, \ldots, f_m^{\Lambda_P^{(n)}})^t = B \cdot (h_1^{(1,1)}, h_1^{(1,2)}, \ldots, h_o^{(l,l)})^t,$$

and define $(w_{1,1,1}, \ldots, w_{l,l,o})^t = B^{-1}(z_1, \ldots, z_{ol^2})^t$.

4. Use the algorithm described in Section 4.1 to find $\mathbf{s} \in \mathbb{F}_q^{ln}$ such that

$$h_1^{(1,1)}(\Lambda_{P^{-1}}^{(n)}\mathbf{s}) = w_{1,1,1}, \ldots, h_o^{(l,l)}(\Lambda_{P^{-1}}^{(n)}\mathbf{s}) = w_{l,l,o}. \tag{17}$$

5. Output $\mathbf{s}$.

In Theorem 2, we showed that the system Eq. (17) satisfies the first three properties of the system described in Section 4.1 with high probability. Hence, it is correct to apply the algorithm described in Section 4.1 to solve the system at Step 4. Moreover, we expect such an algorithm to successfully output a solution $\mathbf{s} \in \mathbb{F}_q^{ln}$ because after each specialization of the last $ln - ol^2$ variables to any vector $(s_{ol^2+1}, \ldots, s_{ln})$, the specialized SNOVA system

$$f_i(x_1, \ldots, x_{ol^2}, s_{ol^2+1}, \ldots, s_{ln}) = z_i, \quad \text{for each } i \in [ol^2].$$

has one solution $s_1, \ldots, s_{ol^2} \in \mathbb{F}_q^{ol^2}$ with high probability, since it is a well-defined system. In such a case, the vector $\tilde{\mathbf{s}} = \Lambda_{P^{-1}}^{(n)} \cdot (s_1, \ldots, s_{ln})^t$ is a solution to the system in Eq. (17).

We performed experiments to verify that the series in (16) accurately predicts the corank of the Macaulay matrix in the modified XL algorithm. We did this for several random systems and random SNOVA public keys with a planted solution. Table 3 summarizes the experimental results. One can see that in every experiment and for both systems, the corank of the Macaulay matrix matches the corank predicted by (16). Note that a negative coefficient in (16) points to the target multi-degree, where the Macaulay matrix is full rank and corresponds to a corank 1 when the system has a solution.

The complexity of the algorithm described above is clearly dominated by the complexity of step 4. Hence, we use the complexity formula given in Eq. (15) to estimate the complexity of solving underdefined SNOVA systems.

Fig. 1 shows bit complexity estimates of the algorithm presented in this section to solve underdefined SNOVA systems with $l(o+v)$ variables and ol^2 equations over $\mathbb{F}_{16}$, where v is chosen in the same regime of the parameters proposed for SNOVA, see Table 1. For comparison, we include the bit complexity estimates of solving random quadratic systems of the same dimensions.

The estimates for SNOVA systems are computed using Eq. (15). In this case, for every $\mathbb{F}_{16^l}$-multiplication we assign a cost of $2(\log_2(16^l))^2 + \log_2(16^l)$ bit operations. For a random system, we found that the best strategy for the specific regime of parameters is to fix the $ln - ol^2$ extra variables and then use the hybrid-XL algorithm. These complexity estimates are computed using the MQEstimator [6], which assigns a cost of $2(\log_2 16)^2 + \log_2(16)$ bit operations per $\mathbb{F}_{16}$-multiplication. From our estimates, we observe that solving an underdefined SNOVA system is easier than its corresponding random version by a factor of $O\left(q^l\right)$. For example, for $(l, o) = (5, 4)$, the SNOVA systems are 22 bits easier, while for $(l, o) = (4, 6), (3, 10)$, and $(2, 18)$ the differences are 15, 10, and 5, respectively.

Parameters v o q l k	target multi-degree	corank sequences
7 2 16 3 9	$[2,1,1]$	srs: $12,12,12,16,8,16,16,-16$ rdm: $12,12,12,16,8,16,16,1$ sno: $12,12,12,16,8,16,16,1$
4 1 16 4 8	$[1,1,1,1]$	srs: $7,7,7,9,7,7,9,7,9,9,-15$ rdm: $7,7,7,9,7,7,9,7,9,9,1$ sno: $7,7,7,9,7,7,9,7,9,9,1$
15 5 16 2 6	$[4,2]$	srs: $31,54,168,31,168,366,80,330,360,160,480,-45$ rdm: $31,54,168,31,168,366,80,330,360,160,480,1$ sno: $31,54,168,31,168,366,80,330,360,160,480,1$
7 4 16 3 15	$[2,2,2]$	srs: $32,56,192,32,192,540,56,192,56,320,832,192,832,1248,32,192,540,192,832,1248,540,1248,-1920$ rdm: $32,56,192,32,192,540,56,192,56,320,832,192,832,1248,32,192,540,192,832,1248,540,1248,1$ sno: $32,56,192,32,192,540,56,192,56,320,832,192,832,1248,32,192,540,192,832,1248,540,1248,1$
15 3 16 2 4	$[3,2]$	srs: $12,19,30,12,30,9,20,28,-45$ rdm: $12,19,30,12,30,9,20,28,1$ sno: $12,19,30,12,30,9,20,28,1$
4 2 16 4 16	$[2,1,1,1]$	srs: $21,21,21,65,21,21,65,21,65,65,73,13,45,45,89,45,89,89,-175$ rdm: $21,21,21,65,21,21,65,21,65,65,73,13,45,45,89,45,89,89,1$ sno: $21,21,21,65,21,21,65,21,65,65,73,13,45,45,89,45,89,89,1$

Table 3. Experimental results to test the effectiveness of our algorithm for solving underdefined SNOVA systems. Each block of three rows corresponds to one parameter set. The target multi-degree is the optimal that allows to solve the system according to (16). The three corank sequences are the coefficients of (16) (srs), the corank sequence of a random system (rdm), and the corank sequence of a random SNOVA public key (sno). All sequences are sorted in graded lexicographic order starting from a sequence whose entries sum two and ending at the target multi-degree.

5 Attacking the Original Version of SNOVA

5.1 Improved Reconciliation Attack

The reconciliation attack is a key-recovery attack for SNOVA that involves finding a vector in the secret space $\mathcal{O}$ that is a solution of an underdefined SNOVA system with ln variables and ol^2 equations. For more details, see Section 2.3.

Our improved reconciliation attack consists of repeatedly applying the algorithm in Section 4.2 by systematically sampling all possible vectors at Steps 1 and 2 in the subroutine described in Section 4.1.

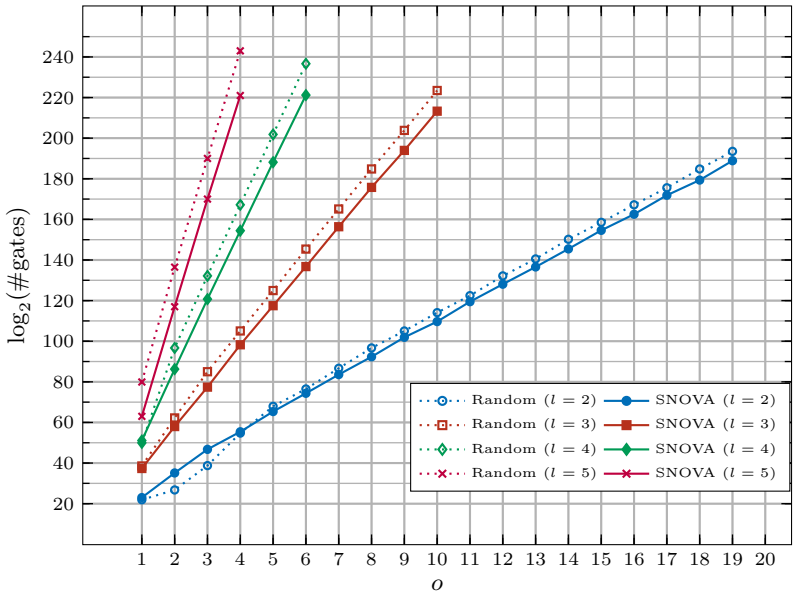

Fig. 1. Comparison of the bit complexity of solving underdefined random systems and SNOVA systems.

For all SNOVA parameters, it holds that $ol^2 < lv$ [19], therefore, the system in Eq. (3) is expected to have $O(q^{lv-ol^2})$ solutions. However, only one solution lies in the secret space $\mathcal{O}$. After sampling $(s_{ol^2+1}, \ldots, s_{ln})^t \in \mathbb{F}_q^{(n-ol)l}$ in Step 1, we can obtain a solution using the algorithm in Section 4.2 if one exists. However, the solution in secret space $\mathcal{O}$ appears with probability q^{ol^2-lv}. Therefore, we expect to iterate at most q^{lv-ol^2} times in Step 1 of the subroutine. Consequently, the complexity of our reconciliation attack is given by q^{lv-ol^2} multiplied by the complexity of one iteration of the algorithm to solve SNOVA systems. That is, the complexity of our reconciliation attack, as the number of $\mathbb{F}_{q^l}$-multiplications, is given by

$$q^{lv-ol^2} \cdot \min_{k \in [ol^2],\ l \mid k} q^k \cdot 3\Big(ol - \frac{k}{l}\Big)^2 \cdot \big[\overline{\mathcal{M}}(\mathbf{d}_{sol})\big]^2, \tag{18}$$

where $\mathbf{d}_{sol} \in \mathbb{Z}_{\geq 0}^l$ minimizes $\overline{\mathcal{M}}(\mathbf{d}_{sol})$, which is the number of monomials of multi-degree smaller than $\mathbf{d}_{sol}$, and there exists $(d_1, \ldots, d_l) \leq \mathbf{d}_{sol}$ such that the coefficient of $t_1^{d_1} t_2^{d_2} \cdots t_l^{d_l}$ in the series (16) is nonnegative.

Table 4 shows the bit complexity estimates of our improved reconciliation attack compared with the state-of-the-art [9,10,12]. Hence, our newly proposed attack improves over previous reconciliation attacks against SNOVA.

Security level	parameters v o q l	previous best reconciliation attack	our attack
I	37 17 16 2	197	195
	25 8 16 3	196	187
	24 5 16 4	269	252
III	56 25 16 2	289	288
	49 11 16 3	438	424
	37 8 16 4	387	367
V	75 33 16 2	379	378
	66 15 16 3	574	560
	60 10 16 4	695	675

Table 4. Bit complexities of our reconciliation attack compared with the previous best reconciliation attack for SNOVA.

5.2 A New Forgery Attack

In this section, we describe a new forgery attack against SNOVA, which uses some ideas introduced in [3] and exploits the multi-homogeneous structure of the lifted SNOVA sequence.

The new attack starts from the following observation, described in Section 2.4 and introduced in [3]. With $U = [\mathbf{u}_0|\cdots|\mathbf{u}_{l-1}]$ and $R_i \in \mathbb{F}_q[S]$, after the change of variables $\mathbf{u}_0 = \mathbf{u}$ and $\mathbf{u}_i = \Lambda^{(n)}_{R_i}\mathbf{u}$ for $i \in [l-1]$, a public key $\mathcal{P}$ will have the form

$$E \cdot \mathcal{F}(\mathbf{u}), \tag{19}$$

where $E = \Lambda^{(o)}_{\tilde{E}}$, and $\tilde{E} \in \mathbb{F}_q^{l^2 \times l^2}$ is public and depends on the choice of $R_1, \ldots, R_{l-1}$.

Our goal is to invert the above public key exploiting both a potential rank defect of E and the multi-homogeneous structure of the sequence $\mathcal{H}(\mathbf{v}) = B^{-1} \cdot \mathcal{F}(\Lambda^{(n)}_P \mathbf{v})$, which is obtained by performing the change of variables given by $\mathbf{v} = \Lambda^{(n)}_{P^{-1}}\mathbf{u}$ with B as in Theorem 2 and P as in Lemma 1. Note that if we substitute p variables by linear equations as in [3], we loose the multi-homogeneous structure.

In order to preserve as much as possible the multi-homogeneous structure of the system, we are forced to use the rank defect of E in a different way to the one proposed by Beullens [3]. We bruteforce the salt so that $\mathbf{z}_0 = \mathsf{Hash}(\mathsf{msg}\|\mathsf{salt}) \in \mathbb{F}_q^{ol^2}$ falls into the column space of E. Then a solution to $E \cdot \mathcal{F}(\mathbf{u}) = \mathbf{z}_0$ can be obtained by finding $\mathbf{u}$ such that $\mathcal{F}(\mathbf{u})$ belongs to the affine subspace of solutions to $E\mathbf{w} = \mathbf{z}_0$, which we describe using a particular solution $\mathbf{w}_0$ and a basis W. The resulting system $\mathcal{F}(\mathbf{u}) = \mathbf{w}_0 + W\mathbf{y}$ is quadratic in $\mathbf{u}$ and linear in the coordinates of the affine subspace $\mathbf{y}$. We can then do the change of variables $\mathbf{v} = \Lambda^{(n)}_{P^{-1}}\mathbf{u}$ to expose the multi-homogeneous structure of the system and solve in the extension field.

We now describe in detail our proposed forgery attack. Given $1 \leq r \leq l^2$, an attacker performs the following steps:

1. Bruteforce $R_1, \ldots, R_{l-1} \in \mathbb{F}_q[S]$ until finding a matrix E of rank $or \leq ol^2$ in Eq. (19).
2. Repeatedly sample $\mathsf{salt} \in \{0,1\}^{128}$ until $\mathbf{z}_0 = \mathsf{Hash}(\mathsf{msg}\|\mathsf{salt}) \in \mathbb{F}_q^{ol^2}$ falls into the column space of E, with Hash as in Section 2.2.
3. Solve a linear system to find $\mathbf{w}_0 \in \mathbb{F}_q^{ol^2}$ such that $\mathbf{z}_0 = E\mathbf{w}_0$.
4. Find a full-rank matrix $W \in \mathbb{F}_q^{ol^2 \times p}$, with $p := o(l^2 - r)$, such that $E \cdot W = 0$.
5. Build the quadratic system
$$\mathbf{w}_0 = \mathcal{F}(\mathbf{u}) + W \cdot \mathbf{y}, \tag{20}$$
where the variables are the coordinates of $\mathbf{u}$ and $\mathbf{y}$.
6. In Eq. (20), perform the change of variables $\mathbf{u} = \Lambda_P^{(n)}\mathbf{v}$ and multiply by the matrix B^{-1} to obtain
$$\tilde{\mathbf{w}_0} = \mathcal{H}(\mathbf{v}) + \tilde{W} \cdot \mathbf{y}, \tag{21}$$
where $\tilde{\mathbf{w}_0} = B^{-1}\mathbf{w}_0 \in \mathbb{F}_{q^l}^{ol^2}$ and $\tilde{W} = B^{-1}W \in \mathbb{F}_{q^l}^{ol^2 \times p}$.
7. Find $(\mathbf{s}, \mathbf{a}) \in \mathbb{F}_q^{ln} \times \mathbb{F}_q^p$ such that $(\Lambda_P^{(n)}\mathbf{s}, \mathbf{a}) \in \mathbb{F}_{q^l}^{nl} \times \mathbb{F}_q^p$ is a solution to the system in Eq. (21).
8. Output a forged signature $\sigma = (U, \mathsf{salt})$ to the message msg, where $U = [\mathbf{s}|\Lambda_{R_1}^{(n)}\mathbf{s}| \cdots |\Lambda_{R_{l-1}}^{(n)}\mathbf{s}]$.

Remark 3. If the rank r of $\tilde{E}$ at Step 1 of the forgery attack described above is too small, then it might be possible we cannot find value salt satisfying the requirements at Step 2. We expect to find such a value salt whenever $o \cdot (l^2 - r)\log_2(q) \leq 128$.

Remark 4. Even though the system in Eq. (20) has the same size as the one in [3] [7], our attack leads to the system Eq. (21), which has a multi-homogeneous quadratic part. This is obtained at the price of finding a particular salt in the first step of the attack.

The complexity of our forgery attack, namely C_{forge}, is dominated by the sum of the complexity of finding salt at Step 2 and the complexity C_{solve} of solving the quadratic system given in Eq. (21). That is,

$$C_{forge} = b \cdot q^{(l^2-r)\cdot o} \cdot l^6 + C_{solve}, \tag{22}$$

[7] Indeed, first we select a number of p quadratic equations such that its corresponding set of homogeneous linear components in the coordinates of $\mathbf{y}$ are linearly independent. We use these equations to eliminate the variables in $\mathbf{y}$ from the rest of the $ol^2 - p$ equations. Since we have added a total of $p = o(l^2 - r)$ new variables, we can specialize the same number of coordinates of $\mathbf{u}$, which yields a set of $ol^2 - p$ quadratic equations in $ln - p$ variables.

where $b := 2(\log_2 q)^2 + \log_2 q$ indicates the bit complexity of performing one $\mathbb{F}_q$-multiplication.

Note that the system in Eq. (21) has ol^2 equations and $ln+p$ variables, so the attacker can specialize up to $ln+p-ol^2$ variables and still expect a solution to the system. For a random system, if the attacker specializes $ln-ol^2+k \geq ln-ol^2+p$ variables, we expect a solution with probability q^{p-k}, hence he must try q^{k-p} values in order to expect a solution. In order to preserve the multi-homogeneous structure, we choose k to be a multiple of l greater or equal to p and specialize $ln-ol^2+k$ variables. More precisely, the attacker runs the following steps, similar to the algorithm described in Section 4.2:

1. Sample $(s_{ol^2-k+1}, \ldots, s_{ln})^t \in \mathbb{F}_q^{(n-ol)l+k}$ and compute
$$\tilde{\mathbf{v}}_2 := \Lambda_{P^{-1}}^{((n-ol)+\frac{k}{l})} \cdot (s_{ol^2-k+1}, \ldots, s_{ln})^t;$$
2. Solve the system of equations
$$\tilde{\mathbf{w}}_0 = \mathcal{H}(\tilde{\mathbf{x}}_1, \tilde{\mathbf{v}}_2) + \tilde{W} \cdot \mathbf{y}, \tag{23}$$
for $(\tilde{\mathbf{x}}_1, \mathbf{y}) \in \mathbb{F}_{q^l}^{ol^2-k} \times \mathbb{F}_q^p$; if no solution is found, go back to Step 1.
3. Set $\mathbf{s} = \Lambda_P^{(n)}(\tilde{\mathbf{x}}_1^t, \tilde{\mathbf{v}}_2^t)^t$. If $\mathbf{s} \in \mathbb{F}_q^{ln}$, output $\mathbf{s}$. Otherwise, go to Step 1.

We propose two ways to solve the system in Eq. (23). In the case of $p = 0$, i.e. $r = l^2$, we use the multi-homogeneous XL algorithm from Section 2.5. On the other hand, if $p > 0$, i.e. $r = l^2$, we cannot directly use the multi-homogeneous XL algorithm because the polynomials in (23) are not multi-homogeneous. However, they do have a special structure that we can use to estimate the complexity of solving the system. Each polynomial in (23) is quadratic. Its quadratic part is multi-homogeneous and only involves variables from $\tilde{\mathbf{x}}_1$. It follows that the first fall degree is bounded from above by $D_{md}^h := \sum_{i=1}^l d_i$, where $d_1, \ldots, d_l \in \mathbb{Z}_{\geq 0}$ are such that $[\mathbf{t}^{(d_1,\ldots,d_l)}]H(t_1, \ldots, t_l) < 0$ with

$$H(t_1, \ldots, t_l) := \frac{\prod_{1 \leq i < j \leq l} (1 - t_i t_j)^{2o} \cdot \prod_{i=1}^l (1 - t_i^2)^o}{\prod_{i=1}^l (1 - t_i)^{ol-k/l}}. \tag{24}$$

One may be tempted, as it is common in the literature, to estimate that the solving degree of a Gröbner Basis algorithm is close to the first fall degree, see e.g. [5]. Such estimates are reasonable for regular or semi-regular sequences, but in general, the two degrees can be far away, see e.g. [4]. Since the polynomials in (23) are not regular and have a lot of structure, we performed experiments to estimate the complexity of solving such a system using a Gröbner basis algorithm such as F4 [8]. We observed that the solving degree in most cases was equal to the degree predicted by the series that accumulates the corank for smaller multi-degrees, namely

$$H'(t_1, \ldots, t_l) := \frac{H(t_1, \ldots, t_l)}{\prod_{i=1}^l (1 - t_i)}. \tag{25}$$

We will call this degree $D_{md} := \sum_{i=1}^{l} d_i$, where $d_1, \ldots, d_l \in \mathbb{Z}_{\geq 0}$ are such that $[\mathbf{t}^{(d_1,\ldots,d_l)}]H'(t_1, \ldots, t_l) < 0$. In all the experiments we ran, the first fall degree was bounded from above by D^h_{md}, and in most cases, the solving degree was less or equal to D_{md}. Hence, the rows and columns of the largest matrix in the Gröbner basis computation were bounded by

$$\binom{ol^2 - k + p + D_{md}}{D_{md}}. \tag{26}$$

Parameters					eqns	vars	D^h_{md}	D_{md}	D_{ff}	D_{sol}	D^h_{reg}	D_{reg}
l	o	r	p	k								
2	8	1	24	24*	32	32	3	3	3	6	23	27
				26		30	2	3	2	3	15	17
				28*		28	2	2	2	3	12	13
				30		26	2	2	2	2	10	11
2	8	2	16	18*	32	30	4	4	4	5	15	17
				20		28	3	4	3	4	12	13
				22		26	3	3	3	3	10	11
				24		24	3	3	3	3	8	9
				26		22	2	3	2	3	7	8
				28		20	2	2	2	2	6	6
				30		18	2	2	2	2	5	6
3	3	5	12	15	27	24	3	4	3	4	11	12
				18		21	3	3	3	3	8	9
				21		18	2	3	2	3	6	7
				24		15	2	2	2	2	5	5
3	3	6	9	15	27	21	3	4	3	4	8	9
				18		18	3	3	3	3	6	7
				21		15	2	3	2	3	5	5
				24		12	2	2	2	2	4	4
3	4	5	16	21	36	31	4	4	4	4	12	13
				24		28	3	4	3	3	10	10

Table 5. Experimental results to test the validity of the complexity bound of our forgery attack in the non-homogeneous case. Each row corresponds to one experiment where we generate a random SNOVA key for a parameter set and run the forgery attack described in Section 5.2. The column D_{md} is our theoretical upper bound for the first degree based on Eq. (24), D_{md} is as defined in (25). The next two columns report the first fall and the solving degree of the F4 algorithm while solving Eq. (21). Vars and eqns give the total number of variables and equations of each system and D^h_{reg}, D_{reg} are the degree of regularity of a semi-regular sequence of that size in the homogeneous and non-homogeneous cases.

Tables 5 and 6 compare our theoretical estimates for the first fall and solving degree with the experimental results. The few cases in which the solving degree

was greater than D_{md}, are marked with a *. Notice that these are cases in which the number of variables is very close to the number of variables. The table also presents the number of rows and columns of the system and the degree of regularity for a semi-regular sequence of this size. The large gap between the solving degree and the degree of regularity shows that the Gröbner basis algorithm indeed takes advantage of the multi-homogeneous structure of the system.

Fig. 2 compares the complexity estimates of the Gröbner step with the complexity observed in experiments solving that step. The experimental cost is the number of clock cycles used by the F4 implementation of Magma, and it is obtained using the function `ClockCycles()`. The theoretical estimates are calculated as the complexity of performing Gaussian elimination in a matrix of size given by Eq. (26), where ω is the linear algebra constant.

Table 7 shows the estimated complexity of our approach in comparison to [3]. The estimates of our approach sum the cost of steps 1, 2 and 6, which dominate the complexity of the attack. The estimates for Step 1 are taken from [3]. The estimates for Step 2 are computed as indicated in Eq. (22). We estimate the complexity of Step 6 as the cost of solving a linear system over $\mathbb{F}_{q^l}$ of the size given by (26). We present two estimates with different linear algebra constants $\omega = 2$ and $\omega = 2.81$ to illustrate the complexity range depending on the implementation. The experiments we presented in Fig. 2 suggest that the actual complexity of an efficient implementation is closer to the values estimated with $\omega = 2$. In that case, our attack is always faster than Beullens' and puts most parameter sets below the security threshold defined by NIST. This is particularly relevant for parameters with $l = 4$, since they allow the smallest keys and signatures, and they are not significantly affected by the attack in [3].

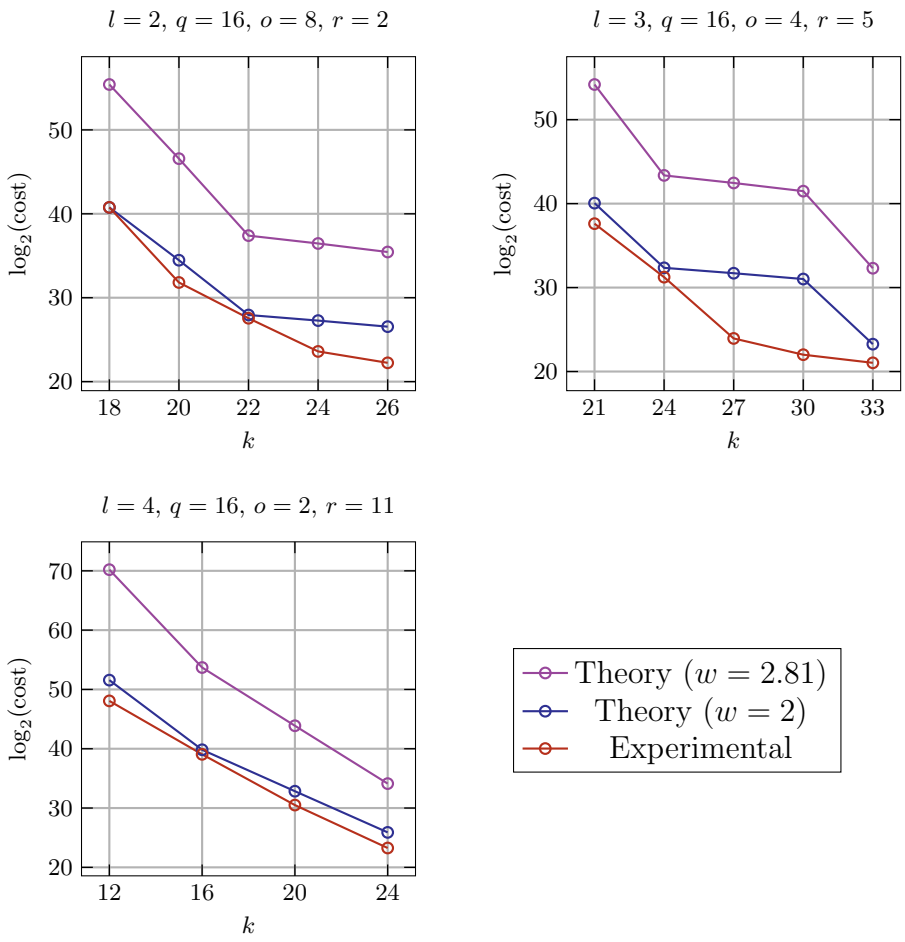

Fig. 2. Complexity of the Gröbner basis computation step; Theory vs Practice.

Parameters					eqns	vars	D^h_{md}	D_{md}	D_{ff}	D_{sol}	D^h_{reg}	D_{reg}
l	o	r	p	k								
				27		25	3	3	3	3	8	8
3	4	5	16	30	36	22	2	3	2	3	6	7
				33		19	2	2	2	2	5	5
				21		27	4	4	4	4	9	10
				24		24	3	4	3	3	7	8
3	4	6	12	27	36	21	3	3	3	3	6	6
				30		18	2	3	2	2	5	5
				33		15	2	2	2	2	4	4
				12		30	5	7	5	7	15	17
				16		26	4	5	4	5	10	11
4	2	11	10	20	32	22	3	4	3	4	7	8
				24		18	3	3	3	3	5	6
				28		14	2	3	2	2	4	4
				16		24	4	5	4	4	8	9
4	2	12	8	20	32	20	3	4	3	3	6	6
				24		16	3	3	3	3	5	5
				28		12	2	3	2	2	3	4
				28		35	4	5	4	5	10	11
4	3	11	15	32	48	31	3	4	3	4	8	9
				36		27	3	3	3	3	6	7
				28		32	4	5	4	4	9	9
4	3	12	12	32	48	28	3	4	3	4	7	7
				36		24	3	3	3	3	6	6
4	5	11	25	56	80	49	4	4	4	4	10	11

Table 6. Experimental results to test the validity of the complexity bound of our forgery attack in the non-homogeneous case. Each row corresponds to one experiment where we generate a random SNOVA key for a parameter set and run the forgery attack described in Section 5.2. The column D_{md} is our theoretical upper bound for the first degree based on Eq. (24), D_{md} is as defined in (25). The next two columns report the first fall and the solving degree of the F4 algorithm while solving Eq. (21). Vars and eqns give the total number of variables and equations of each system and D^h_{reg}, D_{reg} are the degree of regularity of a semi-regular sequence of that size in the homogeneous and non-homogeneous cases.

Parameters					attack in [3]	our attack	
v	o	q	l	r		$w = 2.81$	$w = 2$
				3	137	145	109
37	17	16	2	2	97	N.A.	N.A.
				1	45	N.A.	N.A.
				7	150	171	123
25	8	16	3	6	130	131	110
				5	112	142	142
				13	167	184	139
24	5	16	4	12	156	166	125
				11	145	155	117
				3	189	205	149
56	25	16	2	2	132	N.A.	N.A.
				1	68	N.A.	N.A.
				7	194	216	158
49	11	16	3	6	169	N.A.	N.A.
				5	143	N.A.	N.A.
				13	253	264	199
37	8	16	4	12	235	250	185
				11	218	N.A.	N.A.
				3	240	N.A.	N.A.
75	33	16	2	2	167	N.A.	N.A.
				1	88	N.A.	N.A.
				7	253	276	206
66	15	16	3	6	221	N.A.	N.A.
				5	187	N.A.	N.A.
				13	307	347	256
60	10	16	4	12	285	N.A.	N.A.
				11	264	N.A.	N.A.

Table 7. Bit complexity estimates of the new forgery attack against SNOVA in Section 5.2. N.A. indicates that our algorithm is not expected to work for the particular parameters, as explained in Remark 3. Beullens' attack cost is taken from [3].

References

1. Luk Bettale, Jean-Charles Faugère, and Ludovic Perret. Hybrid approach for solving multivariate systems over finite fields. *Journal of Mathematical Cryptology*, 3(3):177–197, 2009.
2. Ward Beullens. Improved cryptanalysis of UOV and Rainbow. In Anne Canteaut and François-Xavier Standaert, editors, *Advances in Cryptology – EUROCRYPT 2021*, pages 348–373, Cham, 2021. Springer International Publishing.
3. Ward Beullens. Improved cryptanalysis of SNOVA. Cryptology ePrint Archive, Paper 2024/1297, 2024.
4. Alessio Caminata and Elisa Gorla. Solving degree, last fall degree, and related invariants. *Journal of Symbolic Computation*, 114:322–335, 2023.
5. Jintai Ding and Dieter Schmidt. *Solving Degree and Degree of Regularity for Polynomial Systems over a Finite Fields*, pages 34–49. Springer Berlin Heidelberg, Berlin, Heidelberg, 2013.
6. Andre Esser, Javier Verbel, Floyd Zweydinger, and Emanuele Bellini. Sok: Cryptographicestimators – a software library for cryptographic hardness estimation. In *Proceedings of the 19th ACM Asia Conference on Computer and Communications Security*, ASIA CCS '24, page 560–574, New York, NY, USA, 2024. Association for Computing Machinery.
7. Jean-Charles Faugère and Jules Svartz. Gröbner bases of ideals invariant under a commutative group: the non-modular case. In *Proceedings of the 38th International Symposium on Symbolic and Algebraic Computation*, ISSAC '13, page 347–354, New York, NY, USA, 2013. Association for Computing Machinery.
8. Jean-Charles Faugére. A new efficient algorithm for computing gröbner bases (f4). *Journal of Pure and Applied Algebra*, 139(1):61–88, 1999.
9. Yasuhiko Ikematsu and Rika Akiyama. Revisiting the security analysis of SNOVA. *Proceedings of the 11th ACM Asia Public-Key Cryptography Workshop*, 2024.
10. Peigen Li and Jintai Ding. Cryptanalysis of the SNOVA signature scheme. In *International Conference on Post-Quantum Cryptography*, pages 79–91. Springer, 2024.
11. Shuhei Nakamura, Yasuhiko Ikematsu, Yacheng Wang, Jintai Ding, and Tsuyoshi Takagi. New complexity estimation on the Rainbow-Band-Separation attack. *Theoretical Computer Science*, 896:1–18, 2021.
12. Shuhei Nakamura, Yusuke Tani, and Hiroki Furue. Lifting approach against the SNOVA scheme. Cryptology ePrint Archive, Paper 2024/1374, 2024.
13. Shuhei Nakamura, Yacheng Wang, and Yasuhiko Ikematsu. A new analysis of the Kipnis-Shamir method solving the minrank problem. *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, E106.A(3):203–211, 2023.
14. National Institute of Standards and Technology. Call for additional digital signature schemes for the post-quantum cryptography standardization process. NIST Web Page, 2022. https://csrc.nist.gov/csrc/media/Projects/pqc-dig-sig/documents/call-for-proposals-dig-sig-sept-2022.pdf.
15. Ray Perlner and Daniel Smith-Tone. Rainbow band separation is better than we thought. Cryptology ePrint Archive, Paper 2020/702, 2020.
16. Peter W Shor. Algorithms for quantum computation: discrete logarithms and factoring. In *Proceedings 35th annual symposium on foundations of computer science*, pages 124–134. Ieee, 1994.

17. Victor Shoup. *A Computational Introduction to Number Theory and Algebra*. Cambridge University Press, 2005. https://shoup.net/ntb/ntb-v2.pdf.
18. Lih-Chung Wang, Chun-Yen Chou, Jintai Ding, Yen-Liang Kuan, Jan Adriaan Leegwater, Ming-Siou Li, Bo-Shu Tseng, Po-En Tseng, and Chia-Chun Wang. A note on the SNOVA security. Cryptology ePrint Archive, Paper 2024/1517, 2024.
19. Lih-Chung Wang, Po-En Tseng, Yen-Liang Kuan, and Chun-Yen Chou. A simple noncommutative UOV scheme. Cryptology ePrint Archive, Paper 2022/1742, 2022.