# Ultra Short Signatures with Dragon $HFE_{LL'}$

Jacques Patarin[1,3], Alexandre Roullet[1,3], and Jan Vacek[2,3]

[1] Thales DIS France SAS, Meudon, France
[2] Thales DIS Czech Republic, s.r.o., Prague, Czech Republic,
[3] Universite Paris–Saclay, UVSQ, CNRS, Laboratoire de mathematiques de Versailles, Versailles, France
{jacques.patarin,jan.vacek}@thalesgroup.com, alexandreroullet3@gmail.com

**Abstract.** We study HFE-based multivariate signatures with an $LL'$ perturbation. The resulting scheme $HFE_{LL'}$ gives short signatures, and its Dragon variant $D\text{-}HFE_{LL'}$ gives ultra-short signatures by separating the hash length from the number of signature variables. We discuss the relation with $\widehat{\text{HFE}^+}$, the applicability of known attacks, and propose explicit parameter sets.

**Keywords:** Post-quantum cryptography · Multivariate cryptography · Hidden Field Equations · HFE · LL' perturbation · Ultra-short signatures

## 1 Introduction

### 1.1 Motivation for ultra-short signatures

Multivariate cryptography is one of the families of schemes currently studied in post-quantum cryptography. Multivariate signature schemes are particularly attractive when signatures are transmitted or stored in large numbers, for example in certificate chains, constrained communication links, embedded systems, software authentication infrastructures, distributed ledgers, or other bandwidth-sensitive protocols. In such settings, reducing the signature length directly reduces communication and storage costs. This is especially relevant in the post-quantum setting, where many secure signature schemes have substantially larger signatures than classical schemes.

For an expected security level of 128 bits, some variants of the Unbalanced Oil and Vinegar (UOV) scheme submitted to the NIST Post-Quantum Cryptography process have signature lengths of 768 bits (UOV, [7]), 1448 bits (MAYO, [6]) or 992 bits (SNOVA, [22]). With variants of Hidden Field Equations (HFE), even shorter signatures have been suggested, for example 258 bits with the GeMSS scheme [8].

Dragon public keys, as described in [16], include the hash variables inside the public equations through mixed terms. Thus the hash length is no longer tied to the number of signature variables. We can keep a long hash value, as required for birthday resistance, while choosing a smaller number of signature variables and hence shorter signatures.

### 1.2 Related work and positioning

HFE has been subject to strong algebraic attacks, especially improved MinRank attacks [5,21]. Due to these attacks, GeMSS is no longer a NIST candidate. Recently, [9] provided a systematic analysis of several modifiers that may be used to enforce the security of HFE-type multivariate schemes, such as minus, internal perturbation, and vinegar-like techniques, but not $LL'$. This led to variants such as $HFE_{IP-}$ and $HFE_{IPv}$, but these variants increase the signature length and/or the signing cost.

A different way to enforce the security of HFE for signatures was suggested by Gouget and Patarin [14]. In that work, the perturbation $LL'$ was introduced, with one $LL'$ perturbation on each equation of the public key, and the scheme was a probabilistic multivariate signature scheme: not all public equations had to be satisfied. This resisted MinRank attacks, but the signatures became much larger. Recently, [19] suggested using the $LL'$ perturbation for encryption instead of signatures.

In this paper, we again use $LL'$ with HFE for signatures, but in a different way from [14]. First, every public equation has to be satisfied. Second, we use a number of $LL'$ perturbations that is different from, and generally smaller than, the number of equations.

We explicitly point out that this $LL'$ construction is a special case of the $\widehat{\text{HFE}^+}$ perturbation introduced in [12]. Therefore, attacks described for $\widehat{\text{HFE}^+}$ also apply to the present construction and must be considered in the security analysis. The inversion method used by the legitimate signer is also not new in itself: it is the same general "guess the perturbation, invert HFE, and check" strategy as the first inversion method of [12]. Our contribution is the signature-oriented specialization, the parameter choices, and the Dragon variant.

Finally, the Dragon scheme was originally introduced by Patarin [16]. We use Dragon public keys to avoid the usual birthday-paradox limitation on ultra-short signatures. This approach is more efficient than the Patarin–Feistel trick used in GeMSS, yielding both smaller signature sizes and faster computation.

### 1.3 Contributions

The contributions of this paper are as follows.

1. We describe $\text{HFE}_{LL'}$, a signature variant of HFE using a specific $LL'$ perturbation.
2. We combine the same perturbation with Dragon public keys, obtaining $D\text{-HFE}_{LL'}$. This gives ultra-short signatures while keeping a long hash value for birthday resistance.
3. We discuss several relevant attacks, including HFE MinRank attacks, attacks inherited from $\widehat{\text{HFE}^+}$, Dragon-specific $xy$ MinRank attacks, direct algebraic attacks, generalization of the Smith–Tone–Valenzuela $LL'$-support-recovery attack, and birthday attacks.
4. We give explicit parameter sets and discuss the trade-off between the HFE rank parameter and the number of $LL'$ perturbations.

## 2 Preliminaries

### 2.1 Finite-field and multivariate notation

We denote by $\mathbb{F}$ a finite field $\mathbb{F}_q = \mathbb{F}_{p^u}$, where $p$ is prime and $u \in \mathbb{N}^*$. The extension field $\mathbb{F}_{q^n}$ of $\mathbb{F}$ is denoted by $\mathbb{E}$. We also use $\varphi$ for a canonical isomorphism from $\mathbb{E}$ to the corresponding vector space $\mathbb{F}^n$:

$$\varphi : \mathbb{E} \longrightarrow \mathbb{F}^n, \qquad \sum_{i=1}^{n} v_i \vartheta_i \longmapsto (v_1, \ldots, v_n),$$

where $(\vartheta_1, \ldots, \vartheta_n)$ is a basis of $\mathbb{E}$ as a vector space over $\mathbb{F}$.

### 2.2 Hidden Field Equations

In the extension field $\mathbb{E}$, an HFE central polynomial is a univariate polynomial over $\mathbb{E}$ of the form

$$H^* : X \longmapsto \sum_{0 \le i \le j \le \delta} \alpha_{i,j} X^{q^i + q^j},$$

where $\alpha_{i,j} \in \mathbb{E}$. Equivalently, one may impose a degree bound $q^i + q^j \le D$. In the usual binary parameter choices considered below, this bound is written $D = 2^d + 1$.

We separate two pieces of notation that may otherwise be confused. The parameter $\delta$ denotes the largest Frobenius exponent appearing in the HFE polynomial. The parameter $d$ denotes the rank parameter relevant to MinRank attacks, i.e. the number of independent HFE quadratic layers after identifying monomials that differ only by a Frobenius automorphism. In the common binary case where the polynomial contains the monomials $X^{2^i+1}$ for $1 \le i \le \delta$, one typically has $d = \delta$. The two roles are nevertheless conceptually different: $\delta$ controls the degree bound and therefore the cost of inversion, while $d$ controls the rank exploited in MinRank attacks.

Since $H^*$ has this specific form, the map

$$H = \varphi \circ H^* \circ \varphi^{-1} : \mathbb{F}^n \longrightarrow \mathbb{F}^n$$

is a multivariate quadratic map over $\mathbb{F}$. The polynomial $H^*$ is selected so that it can be inverted efficiently; therefore, the degree bound $D$ must not be too high.

### 2.3 Vanilla HFE signatures

To hide the structure, vanilla HFE uses two bijective linear maps $S$ and $T$. They may be represented as linearized polynomials over $\mathbb{E}$ or as matrices over $\mathbb{F}$. The public map is

$$P = T \circ H \circ S,$$

more precisely,

$$P = T \circ \varphi \circ H^* \circ \varphi^{-1} \circ S.$$

It is interpreted as a multivariate quadratic system of $n$ equations in $n$ variables. The public key is $PK = P = \{P_1, \ldots, P_n\}$ and the secret key is $SK = \{H^*, S, T\}$.

Let $M$ be the message to be signed. The signing procedure is as follows.

1. Sample a salt $\tau$ and compute
$$B = \mathsf{Hash}(\tau, M) \in \mathbb{F}^n.$$
The value $B$ is the target value that the public map must output.
2. Compute
$$Y = T^{-1}(B).$$
3. Find any solution $X$ to
$$H^*(X) = Y.$$
If no solution exists, return to Step 1 and try again with a different salt.
4. Compute
$$A = S^{-1}(X),$$
which is the signature of the message.

The verifier accepts if and only if
$$P(A) = \mathsf{Hash}(\tau, M).$$
This vanilla version is recalled only as a basis for the constructions below.

### 2.4 Hashing and salts

In a signature scheme, the message is first mapped to a target value by a cryptographic hash function. We denote this value by
$$B = \mathsf{Hash}(\tau, M),$$
where $M$ is the message and $\tau$ is a salt, also called a diversification value.

The salt is sampled during signing. Its purpose is to give the signer several possible hash targets for the same message. This is useful because, for a fixed target value, the hidden HFE equation may have no solution. If this happens, the signer samples a different salt and tries again.

The salt may either be included explicitly as part of the signature, or it may be recovered by the verifier by trying a bounded set of possible salt values. If the second option is used, the bound must be part of the scheme specification, so that verification remains efficient.

The input to the hash function must be encoded unambiguously. In particular, the salt and the message must be separated in a prefix-free way. For example, one may define
$$\mathsf{Hash}(\tau, M) = \mathsf{XOF}(\texttt{domain} \parallel \mathrm{len}(\tau) \parallel \tau \parallel \mathrm{len}(M) \parallel M),$$
where $\mathsf{XOF}$ is an extendable-output function and `domain` is a fixed domain-separation string for the scheme. The resulting bit string is then interpreted as an element or vector of the appropriate space, depending on the concrete construction.

# 3 The $HFE_{LL'}$ Signature Scheme

## 3.1 The $LL'$ perturbation

Assume that the public map without perturbation is

$$P = T \circ H \circ S.$$

Let $1 \leq r \leq n$, and let

$$L_1, \ldots, L_r, \qquad L'_1, \ldots, L'_r$$

be $2r$ secret linear forms over $\mathbb{F}$ in the variables $x_1, \ldots, x_n$. For $q = 2$, we have

$$\mathbb{P}[L_i(x_1, \ldots, x_n)L'_i(x_1, \ldots, x_n) \neq 0] = \frac{1}{4}.$$

We define the perturbed public polynomials by

$$P'_j = P_j + \sum_{i=1}^{r} a_{i,j} L_i L'_i, \qquad j \in \{1, \ldots, n\},$$

where the coefficients $a_{i,j}$ are secret. The public key consists of the polynomials $P'_1, \ldots, P'_n$, while the private key consists of $T$, $H^*$, $S$, the linear forms $L_i, L'_i$, and the coefficients $a_{i,j}$.

## 3.2 Relation to HFE$^{\widehat{+}}$

The construction above is a particular HFE$^{\widehat{+}}$perturbation [12]: the added perturbation space is spanned by products of pairs of secret linear forms. Consequently, all attacks that apply to HFE$^{\widehat{+}}$also apply to this special case. In particular, the general HFE$^{\widehat{+}}$attack that reduces the problem to an HFE instance with enlarged rank must be considered in Section 5.

We also do not claim the inversion procedure below as new. It is the same general strategy of fixing the perturbation values, inverting an HFE instance, and checking consistency.

## 3.3 Signing and verification

A signature is computed as follows.

1. Compute $B = \mathsf{Hash}(\tau, M)$, where $M$ is the message and $\tau$ is a salt sampled as described in Subsection 2.4.
2. Compute $Y = T^{-1}(B)$.
3. Select values for the terms $L_i L'_i$. This choice is called the hypothesis $\mathcal{H}$.

4. Invert the HFE map for the selected values of $L_i L_i'$, that is, find $X$ such that
$$H^*(X) + \Lambda = Y,$$
where $\Lambda$ is the perturbation vector corresponding to the hypothesis. If no solution exists, try again with a different salt in Step 1 or a different hypothesis in Step 3.
5. Check whether the hypothesis $\mathcal{H}$ is satisfied, i.e. whether the selected values of $L_i L_i'$ agree with the values obtained from the computed solution. If yes, then we have a valid signature candidate. Otherwise, restart. We denote by $N_{\text{att}}$ the average number of attempts required to obtain a valid signature.
6. Compute $A = S^{-1}(X)$, which is the signature of the message.

The verifier checks whether $P'(A) = \mathsf{Hash}(\tau, M)$.

The hypothesis distribution must not always select all values $L_i L_i'$ equal to zero, because this special case gives an attacker additional degree-2 relations; see Subsection 5.5. In the simplest estimates, we choose one product $L_i L_i'$ equal to 1 and the others equal to 0. A more robust variant is to randomize the Hamming weight of the hypothesis. In practice, the signer may set one product $L_i L_i'$ to 1 with high probability, two products to 1 with smaller probability, three products to 1 with still smaller probability, and so on.

### 3.4 Expected number of signing attempts

For a fixed target value, an HFE-type equation does not necessarily have a solution. Therefore, during signing, the signer may have to try several salts and several hypotheses before obtaining a valid signature.

As a first heuristic estimate, we use the following standard approximation. For a random function from a finite set to itself, the probability that a fixed target value belongs to the image is
$$1 - \left(1 - \frac{1}{N}\right)^N,$$
where $N$ is the size of the set. For large $N$, this value is close to
$$1 - \frac{1}{e}.$$
Thus, ignoring the additional cost of the $LL'$ hypothesis, one expects about
$$\frac{e}{e-1} \approx 1.582$$
trials before the target value has a preimage. This standard estimate is discussed, for example, in [13].

In our setting, the public maps are not random functions. Nevertheless, our computer simulations indicate that the same approximation gives a good estimate for the probability that a solution exists for a fixed hash target. A proof of this heuristic for the structured maps considered here is left for future work.

We must also take into account the probability that the chosen $LL'$ hypothesis is correct. In the degree-2 case, assume for simplicity that the hypothesis sets exactly one product $L_iL'_i$ to 1, and all other products to 0. Since, over $GF(2)$,

$$\Pr[L_iL'_i = 1] = \frac{1}{4} \qquad \text{and} \qquad \Pr[L_iL'_i = 0] = \frac{3}{4},$$

the probability that the hypothesis is satisfied is

$$\frac{1}{4}\left(\frac{3}{4}\right)^{r-1}.$$

Combining this with the probability that the HFE equation has a solution gives the estimate

$$N_{\text{att}} = \frac{e}{e-1} \cdot \frac{4^r}{3^{r-1}}.$$

For degree 3, the analogous perturbation is $L_iL'_iL''_i$. In this case,

$$\Pr[L_iL'_iL''_i = 1] = \frac{1}{8} \qquad \text{and} \qquad \Pr[L_iL'_iL''_i = 0] = \frac{7}{8}.$$

Under the same simplified assumption that exactly one perturbation product is set to 1, we obtain

$$N_{\text{att}} = \frac{e}{e-1} \cdot \frac{8^r}{7^{r-1}}.$$

These estimates are used in the parameter tables. They are heuristic estimates: they rely on the random-function approximation for the existence of a solution and on the simplified assumption that exactly one perturbation product is set to 1. More general hypothesis distributions are discussed in Subsection 5.5.

## 4 The Dragon $HFE_{LL'}$ Signature Scheme

### 4.1 Dragon public keys

The Dragon scheme, originally proposed in [16], is a public-key construction whose public key consists of equations in signature variables $a_1, \ldots, a_n$ and hash variables $b_1, \ldots, b_m$ of the form

$$\begin{aligned} P_1(a_1, \ldots, a_n) + Q_1(a_1, \ldots, a_n, b_1, \ldots, b_m) &= F_1(b_1, \ldots, b_m), \\ &\vdots \\ P_k(a_1, \ldots, a_n) + Q_k(a_1, \ldots, a_n, b_1, \ldots, b_m) &= F_k(b_1, \ldots, b_m). \end{aligned} \tag{1}$$

Here each $P_i$ is quadratic in the signature variables, and each $Q_i$ is bilinear in the signature variables and the hash variables. The functions $F_i$ are public functions of the hash variables only. They do not depend on the secret information, and consequently they do not introduce an additional attack surface related to the hidden structure.

To verify a signature $(a_1, \ldots, a_n)$ for a message $M$, the verifier computes $(b_1, \ldots, b_m) = \mathsf{Hash}(\tau, M)$ and checks Equation (1). The advantage is that the hash length $m$ can be chosen large enough to resist birthday attacks, while the signature length is controlled by $n$.

### 4.2 Dragon-HFE central equations

Such public equations can be obtained from secret equations of the form

$$\sum_{0\leq i,j\leq d} \alpha_{i,j} X^{q^i+q^j} + \sum_{i=0}^{d}\sum_{j=1}^{m} \beta_{i,j} X^{q^i} Y_j = 0, \tag{2}$$

where each $Y_j$ is a secret random linear form in the hash variables $y_1, \ldots, y_m$. After fixing the hash value, the terms $Y_j$ become constants and the signer obtains an HFE inversion problem in $X$.

As in HFE, the public key is hidden by secret linear bijections. In the Dragon setting, we typically have $k \approx n$ and $m \approx 2n$.

### 4.3 Adding $LL'$ perturbations

We construct secret linear forms $L_i, L'_i$ in the variables $(a_1, \ldots, a_n, b_1, \ldots, b_m)$ for $i \in \{1, \ldots, r\}$. For $\mathbb{F} = GF(2)$ we again have

$$\mathbb{P}(L_i L'_i \neq 0) = \frac{1}{4}.$$

We then construct secret linear combinations of the products $L_i L'_i$, denoted by $V_i$ for $i \in \{1, \ldots, k\}$. We define $R_i$ as the part of $V_i$ that contains no $a_j$ term, i.e. the pure hash-variable part. We then set

$$V_i^* = V_i - R_i.$$

Thus the public key has the Dragon form

$$\begin{aligned} P_1(a) + Q_1(a,b) + V_1^*(a,b) &= F_1(b), \\ &\vdots \\ P_k(a) + Q_k(a,b) + V_k^*(a,b) &= F_k(b). \end{aligned} \tag{3}$$

The polynomials $R_i$ remove all terms of the form $b_i b_j$ arising from the $LL'$ perturbation, except those that may arise from the public functions $F_i$.

### 4.4 Signing and verification

The signing mechanism proceeds as follows.

1. Compute $B = \mathsf{Hash}(\tau, M)$.
2. Derive the values $Y_i$ and $R$ from $B$.
3. Choose a hypothesis $\mathcal{H}$ specifying which products $L_i L'_i$ are set to 0 and which are set to 1. In the simplest estimates, one product is set to 1 and the others are set to 0.

4. Once the hypothesis is fixed, compute the corresponding vector $V$ and solve

$$\sum_{0\leq i,j\leq d} \alpha_{i,j} X^{2^i+2^j} + \sum_{i=0}^{d}\sum_{j=1}^{m} \beta_{i,j} X^{2^i} Y_j + V + R = F \tag{4}$$

   for $X$. If no solution exists, choose another hypothesis or restart with another salt.
5. Compute $A = S^{-1}X$.
6. Check whether the hypothesis $\mathcal{H}$ is satisfied. If it is satisfied, then $A$ is a valid signature of $M$. Otherwise, repeat the process.

If there is more than one solution $X$, one may choose one of them pseudo-randomly.

### 4.5 Small examples

We give two small examples to illustrate the principle.

*Example 1.*

$$a_9X^9 + a_6X^6 + a_5X^5 + a_3X^3 + b_4X^8Y_4 + b_3X^4Y_3 + b_2X^2Y_2 + b_1XY_1 = 0.$$

Here, $Y_1, Y_2, Y_3, Y_4$ are four vectors whose components are secret linear terms in $y_1, y_2, \ldots, y_m$, and the coefficients $a_i$ and $b_j$ belong to the extension field $\mathbb{E} = GF(2^n)$.

In this example, the largest exponent is 9, so the degree bound is $D = 9$. Since $9 = 2^3 + 1$, the largest Frobenius exponent appearing in the HFE part is $\delta = 3$. The HFE rank parameter is also $d = 3$. Indeed, the independent HFE layers are represented by the terms

$$X^3 = X^{2^1+1}, \qquad X^5 = X^{2^2+1}, \qquad X^9 = X^{2^3+1}.$$

These correspond respectively to the layers 1, 2, and 3.

The term $X^6$ does not increase $d$. Although it is present in the polynomial, we have

$$X^6 = (X^3)^2.$$

Over $GF(2)$, squaring is a Frobenius map and is linear over the base field. Therefore $X^6$ is a Frobenius transform of the same layer as $X^3$, not a new independent HFE layer. Equivalently,

$$X^3 = X^{2^1+2^0} \quad \text{and} \quad X^6 = X^{2^2+2^1}$$

both have the same difference between Frobenius exponents, namely 1. By contrast,

$$X^5 = X^{2^2+2^0} \quad \text{and} \quad X^9 = X^{2^3+2^0}$$

give the new differences 2 and 3, respectively. Thus this example has

$$D = 9, \qquad \delta = 3, \qquad d = 3.$$

*Example 2.*

$$a_3X^3 + b_2X^2Y_2 + b_1XY_1 = 0.$$

This is a minimal example with $D = 3$ and $d = 1$. Since roots of cubic equations can be computed efficiently and one generally wants $d$ to be small, this is an attractive toy example.

## 5 Security Analysis

### 5.1 Overview of the attack surface

The security analysis must consider several types of attacks. First, HFE-based schemes are subject to MinRank attacks on the hidden linear transformations. Second, since our $LL'$ perturbation is a special HFE$^{\widehat{+}}$instance, HFE$^{\widehat{+}}$attacks also apply. Third, Dragon introduces mixed $xy$ terms that require a dedicated MinRank discussion. Fourth, direct algebraic attacks may exploit the degree of regularity of the structured public system. Finally, the signing procedure itself may leak information if the hypothesis distribution is too restricted.

### 5.2 MinRank and HFE$^{\widehat{+}}$ attacks

A MinRank attack is based on the fact that the signer's secret is a low-degree polynomial over an extension field, obfuscated by linear transformations to produce a public system of quadratic equations. There are two relevant forms of the attack. A MinRank-$T$ attack searches for low-rank linear combinations related to the output transformation $T$; this type of attack was described in [5]. A MinRank-$S$ attack targets the input transformation $S$; this type of attack was described in [21] and was later strengthened using support-minors techniques. In particular, GeMSS was broken by a MinRank-$S$ support-minors attack, while it was not broken by the MinRank-$T$ attack [1]. This distinction is important, because both directions have to be considered for HFE-based signatures.

Since our $LL'$ perturbation is a specific HFE$^{\widehat{+}}$perturbation [12], the first part of the MinRank analysis should be understood as a HFE$^{\widehat{+}}$attack applied to our special perturbation space. The $LL'$ perturbation adds $r$ rank-one quadratic components to the HFE structure. Consequently, for the known MinRank-$S$ and MinRank-$T$ attacks, the relevant rank parameter is raised from approximately $d$ to approximately $d + r$. Thus, the same combined parameter $d + r$ is used to estimate the cost of both types of MinRank attacks. A method for solving the corresponding MinRank problem using the support-minor approach is described in [1,3]. The complexity estimate is

$$\mathcal{O}\left((d+r)(n-1)^4\binom{2(d+r)+1}{d+r}^2\right).$$

We denote the combined MinRank parameter by $\mu = d + r$ when helpful. For the proposed parameters, both known MinRank-$S$ and MinRank-$T$ attacks are

therefore covered by the security estimates in Table 2. As usual for multivariate schemes, this does not exclude the possibility of future attacks exploiting additional structure.

### 5.3 MinRank attacks on Dragon *xy* terms

We now consider the Dragon-specific question of whether the mixed terms $x_i y_j$ can lead to a more efficient MinRank attack than the usual $x_i x_j$ terms. This question was first studied in [18] within the $HFE^-_{\mathrm{IP}}$ framework.

For the purpose of the MinRank analysis, it is useful to write the Dragon central map in an extension-field Frobenius form. We use the notation

$$H(X,Y) = \sum_{q^i+q^j \le D} \alpha_{i,j} X^{q^i+q^j} + \sum_{\substack{q^i \le D \\ j<n}} \beta_{i,j} X^{q^i} Y^{q^j}.$$

Here $X$ denotes the extension-field variable corresponding to the signature variables, and $Y$ denotes the extension-field variable corresponding to the hash-dependent part. The index $j < n$ in the second sum indexes the Frobenius powers of $Y$, not the number of hash variables. In the concrete public equations, these Frobenius terms are expanded into bilinear terms involving the signature variables and the hash-derived variables.

Let $(a_0, \ldots, a_{n-1})$ denote the signature variables, and let $(b_0, \ldots, b_{m-1})$ denote the additional variables derived from the hash value. The public key consists of $n$ quadratic equations of the form

$$\widetilde{P}_k(a,b) = \sum_{i,j} \lambda^{(k)}_{i,j} a_i a_j + \sum_{i,j} \mu^{(k)}_{i,j} a_i b_j, \qquad 0 \le k < n.$$

As in [5], we can express the public maps as matrices, which admit the following tensor representation:

$$\left(\widetilde{P}_0, \ldots, \widetilde{P}_{n-1}\right) U = \left(\widetilde{W}\widetilde{H}^{*0}\widetilde{W}^t, \ldots, \widetilde{W}\widetilde{H}^{*(n-1)}\widetilde{W}^t\right), \tag{5}$$

where

$$\widetilde{H} = \begin{pmatrix} (\alpha_{i,j}) & (\beta_{i,j}) \\ 0 & 0 \end{pmatrix}, \qquad \widetilde{W} = \begin{pmatrix} SM_n & 0 \\ 0 & M_{m\times n} \end{pmatrix}, \qquad U = T^{-1} M_n,$$

and

$$\left(\widetilde{H}^{*k}\right)_{i,j} = \left(\widetilde{H}_{i-k,j-k}\right)^{q^k}.$$

Thus, the only structural change introduced by the Dragon terms is the addition of the $\beta$ block. Importantly, this block adds columns to the central matrix $\widetilde{H}$, but it does not add nonzero rows. Consequently, only the first $d$ rows of $\widetilde{H}$ can be nonzero, and therefore

$$\mathrm{rank}(\widetilde{H}) \le d.$$

This observation is the key point for both MinRank attacks. In particular, since $\widetilde{W}$ has full column rank,

$$\operatorname{rank}\left(\widetilde{W}\widetilde{H}\widetilde{W}^t\right) \leq d.$$

Hence, the same MinRank-$T$ attack described in [15] applies to the Dragon public key. Its target rank is unchanged compared with ordinary HFE, while the dimension of the matrices increases from $n$ to $n+m$.

For the MinRank-$S$ attack of [21], the relevant rank is

$$\operatorname{rank}\begin{pmatrix} \widetilde{h}_0 \\ \vdots \\ \widetilde{h}_{n-1} \end{pmatrix},$$

where $\widetilde{h}_k$ denotes the first row of $\widetilde{H}^{*k}$. Again, only the first $d$ rows of $\widetilde{H}$ can be nonzero. Therefore,

$$\operatorname{rank}\begin{pmatrix} \widetilde{h}_0 \\ \vdots \\ \widetilde{h}_{n-1} \end{pmatrix} \leq d.$$

The MinRank-$S$ attack therefore has the same target rank as in ordinary HFE, with the ambient matrix dimension increased from $n$ to $n+m$.

*Discussion.* The derivation above shows explicitly that the Dragon $x_iy_j$ terms do not reduce the target rank of the known MinRank constructions. They modify the public matrices by adding the mixed $\beta$ block, but this block does not introduce additional nonzero rows in the central matrix. Thus, for both the MinRank-$T$ and MinRank-$S$ attacks, the target rank remains unchanged.

Therefore, Dragon terms do not obviously reduce the target rank below the ordinary HFE value. They preserve the known MinRank structure rather than creating a simpler one.

The $LL'$ perturbation raises the relevant target rank from $d$ to approximately $d+r$ for this attack as well. Under the same heuristic used for the $x_ix_j$ terms, the complexity is therefore governed by $d+r$. Nevertheless, Dragon terms are less studied than ordinary HFE terms. The analysis above rules out a reduction of the target rank for the known MinRank strategies, but it does not prove the absence of Dragon-specific attacks. If a future attack exploiting the joint structure of the $x_ix_j$ and $x_iy_j$ blocks is found, the non-Dragon HFE$_{LL'}$construction remains the conservative fallback.

### 5.4 Direct algebraic attacks

Direct algebraic attacks on HFE signatures view signature forgery as the problem of solving the public multivariate quadratic system for the preimage of a target value. The attacker may use Gröbner-basis algorithms such as F4 or F5 [10,11].

This attack must be considered separately from generic random-system estimates. HFE-derived systems may have a lower degree of regularity than random quadratic systems. Therefore, the fact that a random system would be hard to solve is not sufficient by itself. In the proposed schemes, the $LL'$ perturbation is intended to increase the apparent rank and destroy the low-degree relations exploited by known HFE attacks. The parameter choices are based on the best known structural attacks, including the MinRank estimates above.

### 5.5 $LL'$ support recovery

The known attack recovering the $LL'$ perturbation support is the attack of Smith–Tone and Valenzuela [20]. It was proposed against HFE-$LL'$ in encryption mode. In that setting, the intended use is to encrypt many random candidates and decrypt one of them: with significant probability, one candidate is not perturbed, or is only weakly perturbed, by the $LL'$ construction. Such candidates give the legitimate receiver a faster decryption path, because they can be handled essentially as ordinary HFE instances. Smith–Tone and Valenzuela exploit this skewed distribution to recover the noise support generated by the $LL'$ quadratic forms, then filter out the perturbation, recover a compatible HFE map, and finally obtain an equivalent private key.

The signature setting considered here is different from this encryption setting. The signer does not intentionally output signatures corresponding to the all-zero $LL'$ hypothesis. If all products $L_iL'_i$ were always equal to 0 in valid signatures, then valid signatures would provide clean samples where the $LL'$ contribution vanishes. This is precisely the kind of situation that we want to avoid. Therefore, the signer must sometimes use nonzero $LL'$ products.

In the simplest analysis, one chooses exactly one product $L_iL'_i$ equal to 1 and all other products equal to 0. This prevents the direct transcript analogue of the Smith–Tone–Valenzuela attack: the signature oracle does not provide valid signatures lying in the all-zero $LL'$ support. However, always using exactly one nonzero product also creates a rigid distribution. In that case, an attacker may search for higher-degree relations, for example products of two $LL'$ pairs that always vanish. This leads to attacks of higher degree, with complexity roughly $\mathcal{O}\left(n^{12}\right)$ for the degree-four case.

For this reason, a more robust variant is to randomize the Hamming weight of the hypothesis. In practice, the signer may set one product $L_iL'_i$ to 1 with high probability, two products to 1 with smaller probability, three products to 1 with still smaller probability, and so on. This keeps the average signing complexity close to the simple one-nonzero-product estimate, while avoiding deterministic low-degree relations that hold for all signatures. A complete analysis of the optimal hypothesis distribution remains an open problem.

### 5.6 Generic random-system baseline

In this part, we discuss attacks that would apply even to a completely random system. This subsection is only a baseline and should not be confused with the

structural direct attacks discussed above. Since HFE-derived systems may have smaller degree of regularity than generic systems, the random-system estimate alone cannot justify the parameters.

We use the random-system model to understand the minimum signature length one could hope for if the hidden structure gave no advantage to the attacker. In that case, the attacker must solve a generic polynomial system, for example with a Gröbner-basis algorithm. The complexity is commonly estimated as

$$\mathcal{O}\left(\binom{n+d_{reg}}{d_{reg}}^{\omega}\right),$$

where $d_{reg}$ is the degree of regularity and $2 \leq \omega \leq 3$ is a linear-algebra constant [2,11]. Hybrid attacks combine exhaustive search on some variables with Gröbner-basis solving on the remaining variables [4].

**Table 1.** Minimum signature length for $q = 2$ and $\omega = 2.37$.

| | bits of security | | | | |
|---|---|---|---|---|---|
| | 80 | 100 | 128 | 192 | 256 |
| degree 2 | 86 | 112 | 145 | 218 | 290 |
| degree 3 | 82 | 103 | 131 | 196 | 262 |

### 5.7 Birthday attacks and hash length

Since the Dragon $\mathrm{HFE}_{LL'}$ scheme has ultra-short signatures, i.e. signatures whose length is smaller than twice the security level, we must be careful about birthday attacks.

A first birthday attack precomputes candidate signatures and message hashes and searches for a collision between public-key images and hash values. This attack applies to ordinary systems of the form $P(A) = \mathsf{Hash}(M)$ when the signature length is too short. Dragon systems prevent this separation because the equations contain mixed terms involving both signature variables and hash variables.

A second birthday attack targets collisions among hash values. We avoid it by choosing the number $m$ of hash variables so that the hash value has at least twice the target security level, even though the signature itself may be shorter. For example, at the 128-bit security level, the Dragon variant may have about 128 to 160 signature variables but uses about 256 or more hash bits.

The short signature space also implies a possible collision issue for a signer who generates many signatures. Since the signature contains only $n$ bits of signature variables, a signer who produces about $2^{n/2}$ valid signatures may find two different messages with the same signature value by the birthday paradox. This does not give an external forgery attack: without the private key, an adversary

still has to find a valid signature for a target message. However, it means that the signer himself could in principle produce two messages with the same signature, which may be relevant in discussions of non-repudiation.

A technique used in [17] considers signatures that are even shorter than the security parameter by transmitting only part of a value used during verification. We do not use this technique here. Although it can reduce the transmitted signature size, it changes the verification cost and introduces an additional trade-off between computational complexity, space complexity, bandwidth complexity, and the probability that random submitted signatures are accepted.

Finally, if we fix the signature variables $X$, then the Dragon equations become linear in the hash variables $Y$. Therefore, if $Y$ were not a hash value but simply the message itself, it would be easy to find a message with a valid signature. Since the verifier uses a preimage-resistant, domain-separated hash function, this attack does not directly apply.

### 5.8 Degree-3 variants and polarization attacks

We may also consider public equations of degree 3 instead of degree 2. In this case, we can use perturbations of the form

$$L_i L'_i L''_i,$$

where $L_i$, $L'_i$, and $L''_i$ are secret linear forms. The public key then grows considerably. At the same time, over $GF(2)$, the probability that such a perturbation is equal to 0 increases from $\frac{3}{4}$ to $\frac{7}{8}$, since

$$\Pr[L_i L'_i L''_i = 1] = \frac{1}{8}.$$

Therefore, degree 3 may reduce the expected number of nonzero perturbation products, but at the cost of a much larger public key.

A standard way to attack multivariate schemes of degree 3 is to apply polarization, or equivalently to study differentials, in order to derive quadratic equations. We now explain what this gives for the perturbation terms.

Let $X = (x_1, \ldots, x_n)$ be the variable vector, and let $A = (a_1, \ldots, a_n)$ be a fixed shift vector. The coordinates $a_i$ are constants during the differential computation. As a simplified model for one cubic perturbation term, consider

$$f(X) = x_1 x_2 x_3.$$

Over characteristic 2, the differential in direction $A$ is

$$f(X + A) + f(X) + f(A).$$

We compute

$$\begin{aligned} f(X + A) &= (x_1 + a_1)(x_2 + a_2)(x_3 + a_3) \\ &= x_1 x_2 x_3 + a_1 x_2 x_3 + x_1 a_2 x_3 + x_1 x_2 a_3 \\ &\quad + a_1 a_2 x_3 + a_1 x_2 a_3 + x_1 a_2 a_3 + a_1 a_2 a_3. \end{aligned}$$

Therefore,

$$f(X+A)+f(X)+f(A) = a_1x_2x_3 + x_1a_2x_3 + x_1x_2a_3 \\ + a_1a_2x_3 + a_1x_2a_3 + x_1a_2a_3.$$

The last three terms are linear in $X$, and the constant term cancels. Since we are interested in the quadratic part obtained by polarization, we keep

$$a_1x_2x_3 + x_1a_2x_3 + x_1x_2a_3.$$

If $a_1 = a_2 = a_3 = 0$, then the quadratic part is zero, so the rank contribution is 0. If exactly one of $a_1, a_2, a_3$ is nonzero, then the quadratic part is a single product of two linear forms, and the rank contribution is 1. If exactly two of them are nonzero, the rank contribution is still 1, for example

$$x_2x_3 + x_1x_3 = x_3(x_1 + x_2).$$

If all three are nonzero, then the quadratic part is

$$x_2x_3 + x_1x_3 + x_1x_2 = x_1x_2 + x_3(x_1 + x_2),$$

so the rank contribution is 2, not 3.

Thus the average rank contribution is

$$0 \cdot \frac{1}{8} + 1 \cdot \frac{3}{8} + 1 \cdot \frac{3}{8} + 2 \cdot \frac{1}{8} = 1.$$

Consequently, after polarization, one cubic perturbation term contributes on average like one quadratic perturbation term. In other words, the differential produces approximately the same number of effective terms $\widehat{L}_i\widehat{L}'_i$ as the number of original cubic terms $L_iL'_iL''_i$. For this reason, in the degree-3 variant, we keep the same value of $r$ as in the degree-2 security estimates.

## 6 Parameters and Performance

### 6.1 Parameter-selection methodology

The parameter choices are guided by the following constraints.

1. The hash length must be large enough to resist birthday attacks; for a target security level $\lambda$, the Dragon variant uses approximately $m \geq 2\lambda$ hash bits.
2. The signature length is controlled mainly by $n$.
3. The MinRank estimates depend on the combined parameter $d + r$.
4. The signing cost depends on both $D$ and $r$: increasing $d$ increases the cost of solving the HFE equation, while increasing $r$ increases the number of hypotheses to try.

For this reason, small values of $d$ are attractive, and the remaining MinRank security is obtained by increasing $r$.

Table 2 summarizes the values of $d + r$ used for the security estimates.

**Table 2.** Required value of the combined MinRank parameter $d+r$ for the considered security levels.

| security level | $d+r$ |
|---|---|
| 80 | 12 |
| 100 | 17 |
| 128 | 24 |
| 192 | 40 |
| 256 | 56 |

### 6.2 Parameters for $HFE_{LL'}$

For 128 bits of security, we suggest the following parameters for the non-Dragon variant:

$$q = 2, \qquad n = 256, \qquad d + r = 24.$$

For example, one may take $d = 2$ and $r = 22$. With these parameters, the signature size is 256 bits. Table 3 gives parameters for different security levels.

**Table 3.** Parameters for $\mathrm{HFE}_{LL'}$ of degree 2; b = bits, B = bytes.

| security bits | $n$ | $d+r$ | signature size | pk size | $d$ | $D$ | $r$ | $N_{\mathrm{att}}$ | $C_{\mathrm{sig}}$ | time |
|---|---|---|---|---|---|---|---|---|---|---|
| 80 | 160 | 12 | 160 b | 252 KiB | 2 | 5 | 10 | 84 | $8 \cdot 10^4$ | 0.023 s |
| | | | | | 3 | 9 | 9 | 63 | $1 \cdot 10^5$ | 0.032 s |
| 100 | 200 | 17 | 200 b | 491 KiB | 2 | 5 | 15 | 355 | $4 \cdot 10^5$ | 0.124 s |
| | | | | | 3 | 9 | 14 | 266 | $7 \cdot 10^5$ | 0.167 s |
| 128 | 256 | 24 | 256 b | 1.0 MiB | 2 | 5 | 22 | 2661 | $4 \cdot 10^6$ | 1.2 s |
| | | | | | 3 | 9 | 21 | 1995 | $6 \cdot 10^6$ | 1.6 s |
| 192 | 384 | 40 | 384 b | 3.38 MiB | 2 | 5 | 38 | $3 \cdot 10^5$ | $5 \cdot 10^8$ | 3.0 min |
| | | | | | 3 | 9 | 37 | $2 \cdot 10^5$ | $8 \cdot 10^8$ | 4.0 min |
| 256 | 512 | 56 | 512 b | 8.02 MiB | 2 | 5 | 54 | $3 \cdot 10^7$ | $7 \cdot 10^{10}$ | 6.6 h |
| | | | | | 3 | 9 | 53 | $2 \cdot 10^7$ | $1 \cdot 10^{11}$ | 8.9 h |

The public key size can be computed as

$$|PK| = n \cdot C'(n, 2) = \frac{n^3 + n^2}{2},$$

where $C'(n, k)$ denotes the number of combinations with repetition of $k$ elements chosen from $n$ elements. Since the Berlekamp algorithm has complexity

$$\mathcal{O}\left(D^\omega + D(\log D \log\log D)n\right),$$

for $2 < \omega \leq 3$, and since we perform on average $N_{\text{att}}$ attempts while considering relatively small values of $D$, we use the estimate

$$\mathcal{O}\left(C_{\text{sig}}\right) = \mathcal{O}\left((D^3 + nD)N_{\text{att}}\right).$$

We denote by $C_{\text{att}}$ the estimated number of cycles required for one attempt to solve the HFE equation. The total signing cost is then estimated as

$$C_{\text{sig}} = C_{\text{att}} \cdot N_{\text{att}}.$$

**Table 4.** Estimated number of cycles $C_{\text{att}}$ for one attempt to find a solution of the system for degree 2.

| $D \backslash n$ | 160 | 200 | 256 | 384 | 512 |
|---|---|---|---|---|---|
| 5 | $8 \cdot 10^5$ | $9 \cdot 10^5$ | $1 \cdot 10^6$ | $2 \cdot 10^6$ | $2 \cdot 10^6$ |
| 9 | $1 \cdot 10^6$ | $2 \cdot 10^6$ | $2 \cdot 10^6$ | $3 \cdot 10^6$ | $4 \cdot 10^6$ |

### 6.3 Parameters for Dragon $HFE_{LL'}$, degree 2

Table 5 gives the parameters for $D$-HFE$_{LL'}$of degree 2.

**Table 5.** Parameters for $D$-HFE$_{LL'}$of degree 2; b = bits, B = bytes.

| security bits | $n$ | $m$ | $d+r$ | signature size | pk size | $d$ | $D$ | $r$ | $N_{\text{att}}$ | $C_{\text{sig}}$ | time |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 80 | 100 | 200 | 12 | 100 b | 306 KiB | 2 | 5 | 10 | 84 | $5 \cdot 10^4$ | 0.015 s |
| | | | | | | 3 | 9 | 9 | 63 | $1 \cdot 10^5$ | 0.02 s |
| 100 | 128 | 256 | 17 | 128 b | 641 KiB | 2 | 5 | 15 | 355 | $3 \cdot 10^5$ | 0.079 s |
| | | | | | | 3 | 9 | 14 | 266 | $5 \cdot 10^5$ | 0.107 s |
| 128 | 160 | 320 | 24 | 160 b | 1.22 MiB | 2 | 5 | 22 | 2661 | $2 \cdot 10^6$ | 0.742 s |
| | | | | | | 3 | 9 | 21 | 1995 | $4 \cdot 10^6$ | 1.0 s |
| 192 | 230 | 460 | 40 | 230 b | 3.63 MiB | 2 | 5 | 38 | $3 \cdot 10^5$ | $3 \cdot 10^8$ | 1.8 min |
| | | | | | | 3 | 9 | 37 | $2 \cdot 10^5$ | $6 \cdot 10^8$ | 2.4 min |
| 256 | 300 | 600 | 56 | 300 b | 8.05 MiB | 2 | 5 | 54 | $3 \cdot 10^7$ | $4 \cdot 10^{10}$ | 3.8 h |
| | | | | | | 3 | 9 | 53 | $2 \cdot 10^7$ | $7 \cdot 10^{10}$ | 5.2 h |

For degree 2, the public key size is

$$|PK| = n\left(C'\left(n, 2\right) + nm\right) \overset{m=2n}{=} \frac{5}{2}n^3 + \frac{1}{2}n^2.$$

The values of $C_{\text{att}}$ used for the Dragon degree-2 estimates are given in Table 6.

**Table 6.** Estimated number of cycles $C_{\text{att}}$ for one attempt to find a solution of the system for Dragon degree 2.

| $D\backslash n$ | 100 | 128 | 160 | 230 | 300 |
|---|---|---|---|---|---|
| 5 | $5 \cdot 10^5$ | $6 \cdot 10^5$ | $8 \cdot 10^5$ | $1 \cdot 10^6$ | $1 \cdot 10^6$ |
| 9 | $8 \cdot 10^5$ | $1 \cdot 10^6$ | $1 \cdot 10^6$ | $2 \cdot 10^6$ | $3 \cdot 10^6$ |

### 6.4 Parameters for Dragon $HFE_{LL'}$, degree 3

Table 7 gives the parameters for the degree-3 variant. The value $d$ in this table denotes the rank of the differential, which has degree 2.

**Table 7.** Parameters for $D$-HFE$_{LL'}$of degree 3; b = bits, B = bytes.

| security bits | $n$ | $m$ | $d+r$ | signature size | pk size | $d$ | $D$ | $r$ | $N_{\text{att}}$ | $C_{\text{sig}}$ | time |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 80 | 82 | 164 | 12 | 82 b | 17.2 MiB | 2 | 7 | 10 | 42 | $4 \cdot 10^4$ | 0.008 s |
| | | | | | | 3 | 11 | 9 | 37 | $8 \cdot 10^4$ | 0.012 s |
| | | | | | | 4 | 19 | 8 | 32 | $3 \cdot 10^5$ | 0.017 s |
| 100 | 103 | 206 | 17 | 103 b | 42.8 MiB | 2 | 7 | 15 | 82 | $9 \cdot 10^4$ | 0.021 s |
| | | | | | | 3 | 11 | 14 | 72 | $2 \cdot 10^5$ | 0.028 s |
| | | | | | | 4 | 19 | 13 | 63 | $6 \cdot 10^5$ | 0.043 s |
| 128 | 131 | 262 | 24 | 131 b | 112 MiB | 2 | 7 | 22 | 209 | $3 \cdot 10^5$ | 0.067 s |
| | | | | | | 3 | 11 | 21 | 183 | $5 \cdot 10^5$ | 0.092 s |
| | | | | | | 4 | 19 | 20 | 160 | $1 \cdot 10^6$ | 0.139 s |
| 192 | 196 | 392 | 40 | 196 b | 559 MiB | 2 | 7 | 38 | 1770 | $3 \cdot 10^6$ | 0.847 s |
| | | | | | | 3 | 11 | 37 | 1549 | $5 \cdot 10^6$ | 1.2 s |
| | | | | | | 4 | 19 | 36 | 1355 | $1 \cdot 10^7$ | 1.8 s |
| 256 | 262 | 524 | 56 | 262 b | 1.74 GiB | 2 | 7 | 54 | $1 \cdot 10^4$ | $3 \cdot 10^7$ | 9.6 s |
| | | | | | | 3 | 11 | 53 | $1 \cdot 10^4$ | $6 \cdot 10^7$ | 13.2 s |
| | | | | | | 4 | 19 | 52 | $1 \cdot 10^4$ | $1 \cdot 10^8$ | 19.9 s |

For degree 3, the public key size is

$$|PK| = n\left(C'(n,3) + mC'(n,2) + nC'(m,2)\right) \overset{m=2n}{=} \frac{19}{6}n^4 + \frac{5}{2}n^3 + \frac{1}{3}n^2.$$

For degree 3, assuming exactly one product $LL'L''$ is equal to 1, the expected number of attempts is estimated as

$$N_{\text{att}} = \frac{e}{e-1} \cdot \frac{8^r}{7^{r-1}}.$$

The values of $C_{\text{att}}$ used for the Dragon degree-3 estimates are given in Table 8.

**Table 8.** Estimated number of cycles $C_{\text{att}}$ for one attempt to find a solution of the system for Dragon degree 3.

| $D\backslash n$ | 82 | 103 | 131 | 196 | 262 |
|---|---|---|---|---|---|
| 7 | $5 \cdot 10^5$ | $7 \cdot 10^5$ | $9 \cdot 10^5$ | $1 \cdot 10^6$ | $2 \cdot 10^6$ |
| 11 | $8 \cdot 10^5$ | $1 \cdot 10^6$ | $1 \cdot 10^6$ | $2 \cdot 10^6$ | $3 \cdot 10^6$ |
| 19 | $1 \cdot 10^6$ | $2 \cdot 10^6$ | $2 \cdot 10^6$ | $4 \cdot 10^6$ | $5 \cdot 10^6$ |

### 6.5 Trade-off between rank and perturbation number

The parameter $d$ depends on the total degree $D$ of the polynomial over the extension field. For degree-2 equations over the small field, each exponent can be written as $q^i + q^j$, where $i, j \leq d$. The parameter $r$ describes the number of perturbations $L_i L'_i$ used in the system.

From the security point of view, the important value in the MinRank estimates is $d + r$. From the performance point of view, a larger value of $r$ leads to a higher expected number of attempts needed to obtain a valid signature. On the other hand, a smaller value of $r$ requires a larger value of $d$, and as $d$ increases, the complexity of finding a solution increases as well. From the tables, the optimal choice for the considered parameters is to take $d$ as small as possible, typically $d = 2$.

## 7 Conclusion

We have presented two HFE-based signature schemes using $LL'$ perturbations. The first scheme, HFE$_{LL'}$, gives short signatures, while the Dragon variant, $D$-HFE$_{LL'}$, gives ultra-short signatures by mixing signature variables and hash variables in the public equations. For the proposed 128-bit security parameters, the signature sizes are 256, 160, and 131 bits for the non-Dragon, Dragon degree-2, and Dragon degree-3 variants, respectively.

The security discussion includes MinRank attacks inherited from HFE and $\widehat{\text{HFE}^+}$, attacks on the Dragon $xy$ terms, direct algebraic attacks, birthday attacks, and generalization of the Smith–Tone–Valenzuela attack on HFE-$LL'$ in encryption mode. The proposed schemes provide exceptionally short signatures while maintaining practical computational efficiency.

## References

1. Baena, J., Briaud, P., Cabarcas, D., Perlner, R.A., Smith-Tone, D., Verbel, J.A.: Improving support-minors rank attacks: Applications to GeMSS and rainbow. In: Dodis, Y., Shrimpton, T. (eds.) Advances in Cryptology – CRYPTO 2022, Part III. Lecture Notes in Computer Science, vol. 13509, pp. 376–405. Springer, Cham, Switzerland, Santa Barbara, CA, USA (Aug 15–18, 2022). https://doi.org/10.1007/978-3-031-15982-4_13

2. Bardet, M.: Étude des systèmes algébriques surdéterminés. Applications aux codes correcteurs et à la cryptographie. Ph.D. thesis, Université Pierre et Marie Curie-Paris VI (2004)
3. Bardet, M., Briaud, P., Bros, M., Gaborit, P., Tillich, J.P.: Revisiting algebraic attacks on MinRank and on the rank decoding problem. Designs, Codes and Cryptography **91**(11), 3671–3707 (2023). https://doi.org/10.1007/s10623-023-01265-x
4. Bettale, L., Faugère, J.C., Perret, L.: Hybrid approach for solving multivariate systems over finite fields. Journal of Mathematical Cryptology **3**(3), 177 (2009)
5. Bettale, L., Faugère, J.C., Perret, L.: Cryptanalysis of multivariate and odd-characteristic HFE variants. In: Catalano, D., Fazio, N., Gennaro, R., Nicolosi, A. (eds.) PKC 2011: 14th International Conference on Theory and Practice of Public Key Cryptography. Lecture Notes in Computer Science, vol. 6571, pp. 441–458. Springer Berlin Heidelberg, Germany, Taormina, Italy (Mar 6–9, 2011). https://doi.org/10.1007/978-3-642-19379-8_27
6. Beullens, W., Campos, F., Celi, S., Hess, B., Kannwischer, M.J.: MAYO - algorithm specifications and supporting documentation. NIST PQC Round 2 Submission (2025), https://pqmayo.org/assets/specs/mayo-round2-white-back.pdf
7. Beullens, W., Chen, M.S., Ding, J., Gong, B., Kannwischer, M.J., Patarin, J., Peng, B.Y., Schmidt, D., Shih, C.J., Tao, C., Yang, B.Y.: UOV: Unbalanced oil and vinegar - algorithm specifications. NIST PQC Round 2 Submission (2025), https://csrc.nist.gov/csrc/media/Projects/pqc-dig-sig/documents/round-2/spec-files/uov-spec-round2-web.pdf
8. Casanova, A., Faugère, J.C., Macario-Rat, G., Patarin, J., Perret, L., Ryckeghem, J.: GeMSS: A great multivariate short signature. Sub-mission to the NIST Post-Quantum Cryptography Standardization Process (2025), https://csrc.nist.gov/projects/post-quantum-cryptography/post-quantum-cryptography-standardization/round-2-submissions
9. Cogliati, B., Macario-Rat, G., Patarin, J., Varjabedian, P.: State of the art of HFE variants - is it possible to repair HFE with appropriate modifiers? In: Saarinen, M.J., Smith-Tone, D. (eds.) Post-Quantum Cryptography - 15th International Workshop, PQCrypto 2024, Part II. pp. 144–167. Springer, Cham, Switzerland, Oxford, UK (Jun 12–14, 2024). https://doi.org/10.1007/978-3-031-62746-0_7
10. Faugère, J.C.: A new efficient algorithm for computing Gröbner bases (F4). Journal of Pure and Applied Algebra **139**(1-3), 61–88 (1999)
11. Faugère, J.C.: A new efficient algorithm for computing Gröbner bases without reduction to zero (F5). In: Proceedings of ISSAC 2002. pp. 75–83. ACM (2002)
12. Faugère, J.C., macario Rat, G., Patarin, J., Perret, L.: A new perturbation for multivariate public key schemes such as HFE and UOV. Cryptology ePrint Archive, Report 2022/203 (2022), https://eprint.iacr.org/2022/203
13. Fusco, G., Bach, E.: Phase transition of multivariate polynomial systems. In: Proceedings of the 4th International Conference on Theory and Applications of Models of Computation. p. 632–645. TAMC'07, Springer-Verlag, Berlin, Heidelberg (2007)
14. Gouget, A., Patarin, J.: Probabilistic multivariate cryptography. In: Nguyen, P.Q. (ed.) Progress in Cryptology - VIETCRYPT 06: 1st International Conference on Cryptology in Vietnam. Lecture Notes in Computer Science, vol. 4341, pp. 1–18. Springer Berlin Heidelberg, Germany, Hanoi, Vietnam (Sep 25–28, 2006). https://doi.org/10.1007/11958239_1
15. Kipnis, A., Shamir, A.: Cryptanalysis of the HFE public key cryptosystem by relinearization. In: Wiener, M.J. (ed.) Advances in Cryptology – CRYPTO'99. Lecture

Notes in Computer Science, vol. 1666, pp. 19–30. Springer Berlin Heidelberg, Germany, Santa Barbara, CA, USA (Aug 15–19, 1999). https://doi.org/10.1007/3-540-48405-1_2
16. Patarin, J.: Asymmetric cryptography with a hidden monomial. In: Koblitz, N. (ed.) Advances in Cryptology – CRYPTO'96. Lecture Notes in Computer Science, vol. 1109, pp. 45–60. Springer Berlin Heidelberg, Germany, Santa Barbara, CA, USA (Aug 18–22, 1996). https://doi.org/10.1007/3-540-68697-5_4
17. Patarin, J., Macario-Rat, G., Bros, M., Koussa, E.: Ultra-short multivariate public key signatures. Cryptology ePrint Archive, Report 2020/914 (2020), https://eprint.iacr.org/2020/914
18. Patarin, J., Roullet, A.: D-james: Ultra short multivariate signatures. Cryptology ePrint Archive, Paper 2026/1650 (2026), https://eprint.iacr.org/2026/1650
19. Patarin, J., Varjabedian, P.: Multivariate encryptions with LL' perturbations - - is it possible to repair HFE in encryption? In: Dutta, R., Feo, L.D., Gangopadhyay, S. (eds.) Progress in Cryptology - INDOCRYPT 2025: 26th International Conference in Cryptology in India. Lecture Notes in Computer Science, vol. 16372, pp. 263–284. Springer, Cham, Switzerland, Bhubaneshwar, India (Dec 14–17, 2025). https://doi.org/10.1007/978-3-032-13301-4_12
20. Smith-Tone, D., Valenzuela, C.: Cryptanalysis of the best HFE-LL' constructions. Cryptology ePrint Archive, Report 2025/1362 (2025), https://eprint.iacr.org/2025/1362
21. Tao, C., Petzoldt, A., Ding, J.: Efficient key recovery for all HFE signature variants. In: Malkin, T., Peikert, C. (eds.) Advances in Cryptology – CRYPTO 2021, Part I. Lecture Notes in Computer Science, vol. 12825, pp. 70–93. Springer, Cham, Switzerland, Virtual Event (Aug 16–20, 2021). https://doi.org/10.1007/978-3-030-84242-0_4
22. Wang, L.C., Chou, C.Y., Ding, J., Kuan, Y.L., Leegwater, J.A., Li, M.S., Tseng, B.S., Tseng, P.E., Wang, C.C.: SNOVA - proposal for NIST PQC: Additional digital signature schemes (2025), https://csrc.nist.gov/csrc/media/Projects/pqc-dig-sig/documents/round-2/spec-files/snova-spec-round2-web.pdf