Practical Attack on All Parameters of the DME Signature Scheme

Pierre Briaud[1,2](✉), Maxime Bros[3], Ray Perlner[3], and Daniel Smith-Tone[3,4]

[1] Sorbonne Université, UPMC Univ Paris 06, Paris, France
[2] Inria, Team COSMIQ, Paris, France
pierre.briaud@inria.fr
[3] National Institute of Standards and Technology (NIST), Gaithersburg, MD, USA
{maxime.bros,daniel.smith}@nist.gov
[4] University of Louisville, Louisville, KY, USA

Abstract. DME is a multivariate scheme submitted to the call for additional signatures recently launched by NIST. Its performance is one of the best among all the candidates. The public key is constructed from the alternation of very structured linear and non-linear components that constitute the private key, the latter being defined over an extension field. We exploit these structures by proposing an algebraic attack which is practical on all DME parameters.

Keywords: Public Key Cryptography · Multivariate Cryptography · NIST Candidates · Algebraic Cryptanalysis

1 Introduction

After selecting a first collection of post-quantum algorithms to standardize, see [2], the National Institute of Standards and Technology (NIST) announced an expansion to their post-quantum cryptography standardization project and released a call for additional signature schemes [9][1].

One of the candidates is DME. Originally specified in [12], the basic idea is to use multiple rounds of the so-called "exponential maps" defined over a finite extension of $\mathbb{F}_q$ composed with affine maps. These two types of functions can be defined in terms of matrices whose structure is publicly known but whose entries are secret, for all rounds. The construction can also be used for encryption and it lead to a NIST submission in the KEM category already in 2017. This initial version employed two layers of exponential maps, hence the name "Double Matrix Exponentiation". Unfortunately, it was quickly observed in [4] that one can apply some generic attacks on the composition of functions [6,7]. Another attack of a different nature was given by the DME designers in [3]. In response to

[1] This work was partially supported by a grant from the Simons Foundation (712530, DCST).

M. Joye and G. Leander (Eds.): EUROCRYPT 2024, LNCS 14657, pp. 3–29, 2024.
https://doi.org/10.1007/978-3-031-58754-2_1

these threats, the authors increased the number of rounds of exponentiation in their new NIST submission [11] to $r = 3$ and they changed some specific design choices in order to avoid [3].

DME can be seen as a multivariate scheme for which signing boils down to inverting the public system over $\mathbb{F}_q$ in n variables obtained by composition of the different rounds. The hope is that its polynomials are complicated enough so that previous structural cryptanalysis does not apply. Interestingly enough, the analysis of Gröbner basis techniques in the DME submission is limited to lower bounding the solving degree of the public system by the degree of the field equations, i.e., q (see [11, Sect. 8.2]). While having equations of large enough degree might be necessary to counteract some attacks, a too high density for these polynomials would also lead to an unmanageable public key size. Thus, there are notable restrictions imposed on both the linear and the exponential maps to obtain a more compact key.

The NIST submission adopts a unique field extension $\mathbb{F}_{q^2}$ as well as the value $n = 8$ for all security levels. The observation on the Gröbner basis complexity explains that the main difference lies in the choice of q, namely $q = 2^{32}$, $q = 2^{48}$ and $q = 2^{64}$ for security levels I, III and V respectively. DME appears to be the fastest candidate for both signing and verification. The public key and signature sizes are also very attractive. This is quite understandable for the signature as it is a vector of length only 8 over $\mathbb{F}_q$ for all security levels. For the common "signature size + public key size" metric, DME ties with MAYO [5] (1481 and 1489 bytes respectively for level I) and it outperforms the rest of the multivariate candidates *à la UOV*. Due to this surprisingly competitive performance and its rather unusual design, it seems crucial to analyze the DME scheme.

Contribution. This paper presents an efficient attack on the version of DME submitted to the NIST competition [9]. At a very high level, we show that the added constraints which lead to a compact public key also induce a weakness for this construction.

More precisely, we are able to recover, at a very low cost, a candidate last round that can be completed into an equivalent key. The core idea of our attack is to view the scheme entirely over $\mathbb{F}_{q^2}$, the very same extension field over which the exponential maps are defined. This description allows us to exploit the specific structure of the linear and exponential layers coming from the added constraints, even though these maps remain private.

Our approach to invert one DME round is as follows. First, we observe that we can easily determine the monomial content of the polynomials that form the state before applying this round. The task of fully recovering them is then reduced to the one of finding their coefficients. By exploiting the design of the scheme, we achieve this by inverting multivariate equations whose variables are precisely these coefficients. While polynomial system solving is usually the bottleneck when attacking multivariate schemes, our situation is different. The shape of the exponential maps makes that our hardest system is a bilinear system in a small number of variables that is solved in degree only 2. Once we have recovered these polynomials, the full recovery of the "equivalent" last round is rather straightforward.

From this point, we can iterate our attack on the previous rounds or even use the known techniques applicable for $r = 2$ since all DME instantiations consider $r = 3$. In fact, as we essentially exploit a specific structure preserved throughout the rounds, our method suggests that DME might not be simply repaired by increasing (again) the value of r. Nonetheless, as is common in multivariate cryptography, there may still exist different ways to patch the scheme. In any case, we believe that such modifications should be analyzed with care as well as their effect on the performance of DME.

Outline. In Sect. 3, we briefly present the DME version submitted to NIST and in particular the added constraints to obtain compact keys. In Sect. 4, we introduce the description over the extension field $\mathbb{F}_{q^2}$ that we adopt and we derive from it some non-trivial and crucial properties. This material is used in Sect. 5 where we describe the steps to recover an equivalent last round.

2 Notation

Matrices will be written in **bold**. For a positive integer k, let $n = 2k$ denote the DME vector length. We have $k = 4$ for all security levels.

Finite Fields. For a positive integer e, let $q = 2^e$. Let $\mathbb{F}_q$ be a finite field with q elements and let $\mathbb{F}_{q^2}$ be a degree 2 extension of $\mathbb{F}_q$. One may construct this extension as $\mathbb{F}_{q^2} = \mathbb{F}_q[Y]/(g(Y))$, where g is a degree 2 irreducible polynomial over $\mathbb{F}_q$. From now on, we fix an element $U \in \mathbb{F}_{q^2} \setminus \mathbb{F}_q$ (a root of g, for example). In the following, we will often write $A \in \mathbb{F}_{q^2}$ in the form $A = a_1 + a_2U$, where $a_1, a_2 \in \mathbb{F}_q$. Finally, we denote by $\phi : \mathbb{F}_q^n \to \mathbb{F}_{q^2}^k$ the isomorphism

$$\phi(x_1, \dots, x_n) = (x_1 + x_2U, \dots, x_{n-1} + x_nU).$$

Polynomial Rings. We will consider the following two quotients

$$\mathcal{R} = \mathbb{F}_q[x_1, \dots, x_n]/\langle x_1^q - x_1, \dots, x_n^q - x_n\rangle,$$

$$\mathcal{S} = \mathbb{F}_{q^2}[X_1, \dots, X_k]/\langle X_1^{q^2} - X_1, \dots, X_k^{q^2} - X_k\rangle.$$

Powers of Two. Let $a \in \mathbb{Z}_{2e}$. In what follows, we will write $x^{[a]}$ (resp. $P^{[a]}$) instead of x^{2^a} (resp. P^{2^a}) when a scalar $x \in \mathbb{F}_{q^2}$ (resp. a polynomial $P \in \mathcal{S}$) is raised to the power 2^a.

3 Concise Description of DME

The main specificity of DME resides in the so-called "exponential" maps defined over $\mathbb{F}_{q^2}$. Any such map refers to a function $E_{\mathbf{A}} : \mathbb{F}_{q^2}^k \to \mathbb{F}_{q^2}^k$ defined from a matrix $\mathbf{A} = (a_{ij}) \in \mathrm{GL}_k(\mathbb{Z}_{q^2-1})$ by

$$E_{\mathbf{A}}(X_1, \ldots, X_k) = (X_1^{a_{11}} \cdots X_k^{a_{1k}}, \ldots, X_1^{a_{k1}} \cdots X_k^{a_{kk}}).$$

It works similarly to left multiplication by the matrix $\mathbf{A}$, except with the multiplication of coordinates replaced by exponentiation and with addition replaced by multiplication. We can naturally extend this definition to a map $F_{\mathbf{A}} : \mathbb{F}_q^n \to \mathbb{F}_q^n$ such that

$$F_{\mathbf{A}}(x_1, \ldots, x_n) = \phi^{-1} \circ E_{\mathbf{A}} \circ \phi(x_1, \ldots, x_n).$$

The similarity to matrix multiplication triggers some nice algebraic properties. For example, if $\mathbf{AB} = \mathbf{C}$, then $F_{\mathbf{A}} \circ F_{\mathbf{B}} = F_{\mathbf{C}}$. In particular, we have that the composition $F_{\mathbf{A}^{-1}} \circ F_{\mathbf{A}}$ is the identity of $\mathbb{F}_q^n$. This means that the inverse of an exponential map is another easy to compute exponential map, i.e. $F_{\mathbf{A}}^{-1} = F_{\mathbf{A}^{-1}}$.

A DME round corresponds the application of an exponential map followed by a linear layer $L : \mathbb{F}_q^n \to \mathbb{F}_q^n$ and an addition of constants $C : \mathbb{F}_q^n \to \mathbb{F}_q^n$. For instance, a DME public key $P : \mathbb{F}_q^n \to \mathbb{F}_q^n$ utilizing r rounds of exponentiation can be expressed as

$$P(x_1, \ldots, x_n) = C_r \circ L_r \circ F_{\mathbf{A}_r} \circ C_{r-1} \circ L_{r-1} \circ \cdots \circ C_1 \circ L_1 \circ F_{\mathbf{A}_1} \circ C_0 \circ L_0,$$

where we apply a prior affine component $C_0 \circ L_0$ before the first exponential map. This construction may look quite similar to the one of substitution-permutation networks (SPNs) from symmetric cryptography but the situation is in fact reversed. Indeed, the exponential components $F_{\mathbf{A}_i}$ apply to the whole state of size n while we will see that the linear maps act *locally.*

A public key of the above form can be seen as a set of n multivariate polynomials in the ring $\mathcal{R}$. However, such polynomials may be dense and of high degree even after a single round of exponentiation for a generic matrix in $\mathrm{GL}_k(\mathbb{Z}_{q^2-1})$ and a generic affine map, which would lead to an unmanageable key sizes. Thus, restrictions are imposed on both the linear and the exponential maps to obtain a more compact key. Roughly speaking, the idea is to guarantee some collisions among monomials present and to keep having exponents with low Hamming weight binary decomposition (see Definition 4 below). These modifications are quite simple. First, as mentioned above, the linear maps have a local nature. More precisely, each linear map is chosen as the direct sum of linear maps on $\mathbb{F}_q^2$. In other words, such a map can be expressed as a block diagonal matrix with blocks of dimension 2×2. Second, the entries of the matrices defining the exponential maps are restricted to powers of 2 with exponents less than $2e$ (we may view them as belonging to $\mathbb{Z}_{2e}$) and the number of nonzero entries in each row is limited to 1 or 2. Specifically, the submitted DME implementations utilize the following three exponential maps

$$\mathbf{A}_1 = \begin{bmatrix} 2^{a_0} & 0 & 0 & 0 \\ 2^{a_1} & 2^{a_2} & 0 & 0 \\ 0 & 0 & 2^{a_3} & 0 \\ 0 & 0 & 2^{a_4} & 2^{a_5} \end{bmatrix},$$
$$\mathbf{A}_2 = \begin{bmatrix} 2^{b_0} & 0 & 0 & 2^{b_1} \\ 0 & 2^{b_2} & 0 & 0 \\ 0 & 2^{b_3} & 2^{b_4} & 0 \\ 0 & 0 & 0 & 2^{b_5} \end{bmatrix}, \tag{1}$$
$$\mathbf{A}_3 = \begin{bmatrix} 2^{c_0} & 2^{c_1} & 0 & 0 \\ 0 & 2^{c_2} & 0 & 2^{c_3} \\ 0 & 2^{c_4} & 0 & 2^{c_5} \\ 0 & 0 & 2^{c_6} & 2^{c_7} \end{bmatrix},$$

with

$$\begin{aligned} c_1 &= a_0 + b_0 + c_0 - a_1 - b_2 \pmod{2e} \\ c_7 &= a_3 + b_4 + c_6 - a_4 - b_5 \pmod{2e} \\ c_4 &= c_2 + c_5 - c_3 + d \pmod{2e}, \end{aligned} \tag{2}$$

and where the constants e and $d \in \mathbb{Z}_{2e}$ differ with the three security levels. For example, for NIST security level I, we have that $e = 32$ and $d = 57$.

Signing is accomplished by inverting each of the affine shifts C_i, linear maps L_i and exponential maps $F_{\mathbf{A}_i}$ in sequence. Since the inversion of the exponential maps requires the structure of the exponential maps defined over $\mathbb{F}_{q^2}$ (e.g., $E_{\mathbf{A}_i}$), we may naturally view DME as a multi-round big-field cryptosystem, see Fig. 1. Verification is performed by simply evaluating the public key at the signature. We may note here that the component maps $F_{\mathbf{A}_i}$ are not surjective; hence, it is not possible to use the public key without a construction to prevent signature failure. Still, since these maps are bijections of the image of the unit groups of $\mathbb{F}_{q^2}^k$ under ϕ^{-1} (see for instance [12, Theorem 1.2]), failure of inversion is a reasonably low probability event. The simple use of a salt mitigates the problem efficiently, allowing for isochronous signature generation. The specific choice of an analogue of the PSS00 construction of [1] is specified in [11].

4 Structure of DME over $\mathbb{F}_{q^2}$

As described in the previous section, we may view DME as a multi-round big field scheme. There are two important aspects of this identification. First, the maps over the extension field are still multivariate; thus, the scheme is similar in spirit to the so-called "intermediate field schemes" such as HMFEv-, see [14]. The second important aspect is the multi-round nature. This characteristic is obviously recurrent in symmetric cryptography but it is also present in some very old and rather well-known multivariate constructions such as the double

$$\begin{tikzcd}
 & \mathbb{F}_{q^2}^k \arrow[r, "E_{\mathbf{A}_1}"] & \mathbb{F}_{q^2}^k \arrow[d, "\phi^{-1}"] & \mathbb{F}_{q^2}^k \arrow[r, "E_{\mathbf{A}_2}"] & \mathbb{F}_{q^2}^k \arrow[d, "\phi^{-1}"] & \mathbb{F}_{q^2}^k \arrow[r, "E_{\mathbf{A}_3}"] & \mathbb{F}_{q^2}^k \arrow[d, "\phi^{-1}"] & \\
\mathbb{F}_q^n \arrow[r, "C_0 \circ L_0"] \arrow[rrrrrrr, dashed, bend right, "P"'] & \mathbb{F}_q^n \arrow[u, "\phi"] \arrow[r, "F_{\mathbf{A}_1}"] & \mathbb{F}_q^n \arrow[r, "C_1 \circ L_1"] & \mathbb{F}_q^n \arrow[u, "\phi"] \arrow[r, "F_{\mathbf{A}_2}"] & \mathbb{F}_q^n \arrow[r, "C_2 \circ L_2"] & \mathbb{F}_q^n \arrow[u, "\phi"] \arrow[r, "F_{\mathbf{A}_4}"] & \mathbb{F}_q^n \arrow[r, "C_3 \circ L_3"] & \mathbb{F}_q^n
\end{tikzcd}$$

Fig. 1. A 3-round DME signature scheme.

round quadratic cipher, see [13]. We should note that both of these styles of multivariate cryptosystems were broken, see [10,15,17].

Let us expand the structure of DME over this extension field $\mathbb{F}_{q^2}$. Recall that each linear map L_i is the direct sum of maps on $\mathbb{F}_q^2$. We can thus write it as

$$L_i(x_1, \ldots, x_n) = (L_{i1}(x_1, x_2), \ldots, L_{ik}(x_{n-1}, x_n)),$$

where L_{ij} is a linear map on $\mathbb{F}_q^2$ for $j \in \{1..k\}$. Note further that any $\mathbb{F}_q$-linear map L on $\mathbb{F}_q^2$ is isomorphic to an $\mathbb{F}_q$-linear polynomial $L_{\mathbb{F}_2} : \mathbb{F}_{q^2} \to \mathbb{F}_{q^2}$ by $L(x_1, x_2) = \phi^{-1}(L_{\mathbb{F}_2}(\phi(x_1, x_2)))$. In this way, we can easily construct an $\mathbb{F}_q$-linear map $\widehat{L}_i : \mathbb{F}_{q^2}^k \to \mathbb{F}_{q^2}^k$ equivalent to L_i. Similarly, each affine shift C_i consists of adding coordinate-wise constants $d_i \in \mathbb{F}_q$. Let us denote by $\widehat{C}_i$ the equivalent affine shift over $\mathbb{F}_{q^2}$ by adding the vector of constants $(D_{i1}, \ldots, D_{ik}) = \phi(d_1, \ldots, d_n)$. With these two observations, we may complete our commutative diagram for the public key P to consider a calculation path $\widehat{P}$ entirely over the extension field $\mathbb{F}_{q^2}$, see Fig. 2. Given a formal input $(X_1, \ldots, X_k) \in \mathbb{F}_{q^2}^k$, we may implicitly view all big field polynomials as elements of the quotient ring $\mathcal{S} = \mathbb{F}_{q^2}[X_1, \ldots, X_k]/\langle X_1^{q^2} - X_1, \ldots, X_k^{q^2} - X_k\rangle$.

The rest of this section is dedicated to some crucial observations illustrating why the extension field is the correct arena in which to study the structure of DME. These properties are related to certain invariants of the composition of the exponential maps with the linear maps and affine shifts—the same invariants

$$\begin{tikzcd}
\mathbb{F}_{q^2}^k \arrow[r, "\widehat{C}_0 \circ \widehat{L}_0"'] \arrow[rrrrrrr, dashed, bend left, "\widehat{P}"] & \mathbb{F}_{q^2}^k \arrow[r, "E_{\mathbf{A}_1}"'] & \mathbb{F}_{q^2}^k \arrow[r, "\widehat{C}_1 \circ \widehat{L}_1"'] \arrow[d, "\phi^{-1}"] & \mathbb{F}_{q^2}^k \arrow[r, "E_{\mathbf{A}_2}"'] & \mathbb{F}_{q^2}^k \arrow[r, "\widehat{C}_2 \circ \widehat{L}_2"'] \arrow[d, "\phi^{-1}"] & \mathbb{F}_{q^2}^k \arrow[r, "E_{\mathbf{A}_3}"'] & \mathbb{F}_{q^2}^k \arrow[r, "\widehat{C}_3 \circ \widehat{L}_3"'] \arrow[d, "\phi^{-1}"] & \mathbb{F}_{q^2}^k \arrow[d, "\phi^{-1}"] \\
\mathbb{F}_q^n \arrow[u, "\phi"'] \arrow[r, "C_0 \circ L_0"] \arrow[rrrrrrr, dashed, bend right, "P"'] & \mathbb{F}_q^n \arrow[u, "\phi"] \arrow[r, "F_{\mathbf{A}_1}"] & \mathbb{F}_q^n \arrow[r, "C_1 \circ L_1"] & \mathbb{F}_q^n \arrow[u, "\phi"] \arrow[r, "F_{\mathbf{A}_2}"] & \mathbb{F}_q^n \arrow[r, "C_2 \circ L_2"] & \mathbb{F}_q^n \arrow[u, "\phi"] \arrow[r, "F_{\mathbf{A}_4}"] & \mathbb{F}_q^n \arrow[r, "C_3 \circ L_3"] & \mathbb{F}_q^n
\end{tikzcd}$$

Fig. 2. The 3-round DME considering an equivalent "public" and private key over $\mathbb{F}_{q^2}$.

that guarantee efficiency and small key sizes. To establish a consistent language for discussing the structures, we define the DME round function over $\mathbb{F}_{q^2}$.

Definition 1 (Round function over $\mathbb{F}_{q^2}$). *Given an exponential map $E_{\mathbf{A}_i}$, an $\mathbb{F}_q$-linear map $\widehat{L}_i$ and an affine shift map $\widehat{C}_i$, the composition $R_i = \widehat{C}_i \circ \widehat{L}_i \circ E_{\mathbf{A}_i}$ is called the i-th* DME round function over $\mathbb{F}_{q^2}$.

With this notation we have $\widehat{P} = R_3 \circ R_2 \circ R_1 \circ \widehat{C}_0 \circ \widehat{L}_0$.

4.1 Stability by q-Powering

The first observation of importance is that the initial map $\widehat{C}_0 \circ \widehat{L}_0$ establishes a symmetry that is invariant under the application of DME round functions. Let us recall that each coordinate of $\widehat{L}_0$ is an $\mathbb{F}_q$-linear polynomial in $\mathbb{F}_{q^2}$. It is a classical result that such a polynomial is of the form $X \mapsto A_1X + A_2X^q$ for some $A_1,\ A_2 \in \mathbb{F}_{q^2}$. We thus obtain

$$\widehat{C}_0 \circ \widehat{L}_0(X_1, \ldots, X_k) = (A_{0,1}X_1 + A_{0,2}X_1^q + D_1, \ldots, A_{0,2k-1}X_k + A_{0,2k}X_k^q + D_k).$$

In each of the k coordinates, we notice that the relevant variable, i.e., X_i for the i-th one, occurs with the power 1 and q only. In order to generalize this property to more rounds, we introduce the following definition. Recall that $\mathcal{S} = \mathbb{F}_{q^2}[X_1, \ldots, X_k]/\langle X_1^{q^2} - X_1, \ldots, X_k^{q^2} - X_k\rangle$.

Definition 2 (q-symmetric orbit). *For a monomial $X_1^{\alpha}X_2^{\beta} \cdots X_k^{\gamma} \in \mathcal{S}$, the* q-symmetric orbit *is defined to be the set*

$$\{X_1^{\alpha}X_2^{\beta} \cdots X_k^{\gamma}, X_1^{q\alpha}X_2^{\beta} \cdots X_k^{\gamma}, X_1^{\alpha}X_2^{q\beta} \cdots X_k^{\gamma},$$
$$X_1^{q\alpha}X_2^{q\beta} \cdots X_k^{\gamma}, \ldots, X_1^{q\alpha}X_2^{q\beta} \cdots X_k^{q\gamma}\}.$$

In other words, each of its elements is obtained by q-powering one or several variables present in the monomial.

Remark 1 *For the term "orbit" to make sense it might be more natural to authorize an arbitrary number of q-powerings. The above definition would not change as we have $X_i^{\alpha q^2} = X_i^{\alpha} \in \mathcal{S}$ for any $i \in \{1..k\}$ and any $\alpha \in \mathbb{N}$.*

Definition 3 (q-symmetric polynomial). *A polynomial $p \in \mathcal{S}$ is said to be* q-symmetric *if its set of monomials is a disjoint union of q-symmetric orbits.*

One may verify some simple properties of q-symmetric polynomials. We summarize the ones we require in the following. Their proofs are easy and the only subtlety lies in coefficient cancellations, see Appendix A.

Lemma 1. *The following statements on q-symmetric polynomials hold with high probability (easily estimated assuming a bound on the number of monomials),*

1. *Let $D \in \mathbb{F}_{q^2} \subseteq \mathcal{S}$. Then D is q-symmetric (with probability* 1*).*

2. *Let $p_1,\ p_2 \in \mathcal{S}$ two q-symmetric polynomials. Then $p_1 + p_2$ is q-symmetric.*
3. *Let $p_1,\ p_2 \in \mathcal{S}$ two q-symmetric polynomials. Then $p_1 p_2$ is q-symmetric.*
4. *Let $p \in \mathcal{S}$ be a q-symmetric polynomial and let $r \in \mathbb{N}$. Then p^r is q-symmetric.*
5. *Let $p \in \mathcal{S}$ be q-symmetric and let $L : \mathbb{F}_{q^2} \to \mathbb{F}_{q^2}$ be an $\mathbb{F}_q$-linear map. Then the composition $L(p) \in \mathcal{S}$ is q-symmetric. Moreover, the monomial content of $L(p)$ is identical to that of p.*

From these results, we see that all of the linear and affine layers preserve q-symmetry. This property is in fact critical in maintaining control on the growth of the number of monomials in the key as the number of rounds increases. Furthermore, since each coordinate of the exponential map $E_{\mathbf{A}_i}$ simply raises q-symmetric polynomials to powers and multiplies them together, Lemma 1 shows that the exponential maps preserve q-symmetry as well. Thus, we obtain

Corollary 1. *Given a k-tuple of q-symmetric polynomials as input, the DME round function over $\mathbb{F}_{q^2}$ produces another k-tuple of q-symmetric polynomials with high probability.*

4.2 Multi-hamming Weight

To analyze the specific structure of the polynomials produced by each round function, we will also use the following definition. As we work over $\mathcal{S}$, each variable exponent can be seen as an element in $\mathbb{Z}_{q^2-1}$. In the following, we may implicitly consider the unique representative in $\{0..q^2-2\}$.

Definition 4 (Multi-Hamming weight). *A monomial $X_1^{\alpha} X_2^{\beta} \cdots X_k^{\gamma} \in \mathcal{S}$ has* multi-Hamming weight $(a, b, \ldots, c)$ *if the binary representations of $\alpha, \beta, \ldots, \gamma$ are of Hamming weight $a, b, \ldots, c$, respectively.*

Clearly, raising a polynomial to a power of the form 2^s does not change the multi-Hamming weights of its monomials. More generally, since q is a power of two, we also observe that the multi-Hamming weight remains constant within one q-symmetric orbit. Note however that an arbitrary q-symmetric polynomial may contain distinct orbits having the same multi-Hamming weight.

We will restrict our notation to consider the parameters of DME provided in the submission package [11]. For the case of NIST security level I defined in [9], recall that we had $e = 32$ (thus, $q = 2^{32}$), $k = 4$ and 3 rounds. From the definition of DME, we see that the coordinates in the multi-Hamming weight can only grow by product between different components over $\mathbb{F}_{q^2}$. Due to the shape of the exponential maps $\boldsymbol{A}_i$, we also see that such a product involves at most 2 components. Finally, since there are only 3 rounds, we need to consider at most 3 such products. For these reasons the coordinates will remain small (we compute them explicitly in the next section). Since they will never be as large as 10, there will be no ambiguity in abbreviating $(a, b, \ldots, c)$ via concatenation: $ab \ldots c$. For example, we will denote the multi-Hamming weights of the input vector (X_1, X_2, X_3, X_4) as $(1000, 0100, 0010, 0001)$ instead of the much heavier

$$((1,0,0,0), (0,1,0,0), (0,0,1,0), (0,0,0,1)) .$$

4.3 Monomial Content over $\mathbb{F}_{q^2}$

Using these notions, we can now describe more precisely the DME structure after each round. In our presentation, we will assume that no cancellation between coefficients occurs. We note that such an event should hold with very low probability under the choice of the linear and non-linear DME components.

Affine Layer $\widehat{\boldsymbol{C}}_{\mathbf{0}} \circ \widehat{\boldsymbol{L}}_{\mathbf{0}}$. As observed above, the application of the first linear layer $\widehat{L}_0$ produces polynomials of the form $AX + BX^q$. The relevant weight vector remains $(1000, 0100, 0010, 0001)$ but this time we have created q-symmetric polynomials. Each of these equations contains one orbit with two monomials. Applying $\widehat{C}_0$ would then add the orbit $\{1\}$ but this first affine shift is in fact omitted in [11].

In the subsequent rounds, we may focus on the exponential maps. This is because each affine map can be seen as acting coordinate-wise over $\mathbb{F}_{q^2}$ and because the monomial content of a q-symmetric polynomial does not change after applying an $\mathbb{F}_q$-linear map (statement 5 in Lemma 1). For $j \in \{1..3\}$, we will denote the j-th round output by $(G_1^{(j)}, G_2^{(j)}, G_3^{(j)}, G_4^{(j)})$.

Round Function $\boldsymbol{R_1}$. The monomial content of the state $(G_1^{(1)}, G_2^{(1)}, G_3^{(1)}, G_4^{(1)})$ obtained at the end of R_1 is presented in Table 1. For each polynomial, we give the variables involved, the q-symmetric orbits (in the form "associated multi-Hamming weights: cardinality") and the total number of monomials.

Table 1. Variables, q-symmetric orbits and monomial counts for the output coordinates of R_1.

	Variables	Orbits	#Monomials
$G_1^{(1)}$	X_1	1000:2, 0000:1	3
$G_2^{(1)}$	X_1, X_2	1100:4, 0000:1	5
$G_3^{(1)}$	X_3	0010:2, 0000:1	3
$G_4^{(1)}$	X_3, X_4	0011:4, 0000:1	5

This table can be easily obtained from the structure of $\mathbf{A}_1$ in Eq. (1). Raising a polynomial to a power of two does not change the multi-Hamming weights of its monomials and it does not affect the number of orbits. Then, each output is the product of at most two such 2-powered polynomials. In this first step we multiply polynomials with no variables in common: in this case, the multi-Hamming weight of a product is equal the sum of their multi-Hamming weights. Finally, as we apply $\widehat{C}_1$, we need to add the constant monomial.

Round Function $\boldsymbol{R_2}$. We may continue in the same fashion to reveal the structure over $\mathbb{F}_{q^2}$ of the output of the round function R_2, see Table 2. The numbers of (complete) orbits and monomials present are still derived assuming no cancellation. We note that, once again, the polynomials being multiplied share no variables in common; thus, the multi-Hamming weight of the product of two monomials is equal to the sum of their multi-Hamming weights in all cases. We also note that given the monomial count from Table 1, the numbers of monomials in the output coordinates of the exponential layer correspond to products of numbers of monomials in the factors due to the fact that all of those monomials are distinct. In particular, the number of monomials in the output coordinates of R_2 will be these products unless there is cancellation of the coefficients due to the application of the linear layer, $\widehat{L}_2$, which is a low probability event.

Table 2. Variables, q-symmetric orbits and monomial counts for the output coordinates of R_2.

	Variables	Orbits	#Monomials
$G_1^{(2)}$	X_1, X_3, X_4	1011:8, 0011:4, 1000:2, 0000:1	15
$G_2^{(2)}$	X_1, X_2	1100:4, 0000:1	5
$G_3^{(2)}$	X_1, X_2, X_3	1110:8, 1100:4, 0010:2, 0000:1	15
$G_4^{(2)}$	X_3, X_4	0011:4, 0000:1	5

Round Function R_3. Deriving the q-symmetric orbits and the monomial support of the output coordinates of R_3 is more intricate.

The third round is the first in which products of monomials containing the same variables occur. Note that the product of a pair of orbits from R_2 no longer necessarily yields a unique q-symmetric orbit in R_3. Indeed, consider two orbits Ω and Ω' where a variable X appears with Hamming weight 1 and let us take representatives for these orbits with exponents 2^u and 2^v respectively. In other words, there exist monomials μ_Ω and $\mu_{\Omega'}$ not involving X such that $X^{[u]}\mu_\Omega \in \Omega$ and $X^{[v]}\mu_{\Omega'} \in \Omega'$. By doing the product we get monomials falling into two distinct orbits, namely $X^{[u]}X^{[v]}\mu_\Omega\mu_{\Omega'}$, $X^{[u+e]}X^{[v+e]}\mu_\Omega\mu_{\Omega'}$ on one side and $X^{[u+e]}X^{[v]}\mu_\Omega\mu_{\Omega'}$, $X^{[u]}X^{[v+e]}\mu_\Omega\mu_{\Omega'}$ on the other side.

Such a behaviour already occurs in the case when there are no imposed relations on the exponents in the exponential maps, see Table 3. For example, the product between the orbit of multi-Hamming weight 1011 in $G_1^{(2)}$ and the one of multi-Hamming weight 1000 in $G_2^{(2)}$ gives two orbits 2111(a) and 2111(b) of the same multi-Hamming weight and same cardinality 16 in $G_1^{(3)}$. The same phenomenon happens for the monomials of multi-Hamming weight 2100. Overall, a counting based on the multi-Hamming weights of all orbits would give 75 monomials for $G_1^{(3)}$ and $G_4^{(3)}$.

However, with the extra constraints of Eq. (2) imposed in the specification of DME, these polynomials actually contain only 65 monomials, see Lemma 2. More complete information on the monomial content is given in Table 4 below.

Table 3. Variables, q-symmetric orbits and monomial counts for the output coordinates when the matrices $\mathbf{A}_i \in \mathrm{GL}_k(\mathbb{Z}_{q^2-1})$ are chosen without the constraints of Eq. (2).

	Orbits	#Monomials
$G_1^{(3)}$	2111(a):16, 2111(b):16, 1111:16, 2100(a):4, 2100(b):4, 1011:8, 1100:4, 0011:4, 1000:2, 0000:1	75
$G_2^{(3)}$	1111:16, 1100:4, 0011:4, 0000:1	25
$G_3^{(3)}$	1111:16, 1100:4, 0011:4, 0000:1	25
$G_4^{(3)}$	1121(a):16, 1121(b):16, 1111:16, 0021(a):4, 0021(b):4, 1110:8, 1100:4, 0011:4, 0010:2, 0000:1	75

Table 4. DME case.

	Orbits	#Monomials
$G_1^{(3)}$	$\underline{\text{2111:8}}$, $\underline{\text{2100:2}}$, 1111:16, $\underline{\text{1111(fall):16}}$, 1011:8, 1100:4, $\underline{\text{1100(fall):4}}$, 0011:4, 1000:2, 0000:1	65
$G_2^{(3)}$	1111:16, 1100:4, 0011:4, 0000:1	25
$G_3^{(3)}$	1111:16, 1100:4, 0011:4, 0000:1	25
$G_4^{(3)}$	$\underline{\text{1121:8}}$, $\underline{\text{0021:2}}$, 1111:16, $\underline{\text{1111(fall):16}}$, 1110:8, 1100:4, 0011:4, $\underline{\text{0011(fall):4}}$, 0010:2, 0000:1	65

Lemma 2. *Under the* 3 *constraints on the exponential maps given in Eq.* (2)*, the number of monomials in* $G_1^{(3)}$ *and* $G_4^{(3)}$ *is generically equal to* 65*.*

Proof. Let us consider $G_1^{(3)}$. The case of $G_4^{(3)}$ is similar by replacing the first condition of Eq. (2) by the second one. First, notice from the definition of $\mathbf{A}_3$ given in Eq. (1) that the first output coordinate of $E_{\mathbf{A}_3}$ is

$$\left(G_1^{(2)}\right)^{[c_0]}\left(G_2^{(2)}\right)^{[c_1]}. \tag{3}$$

Thus, by Lemma 1, $G_1^{(3)}$ has the monomial structure of a product of polynomials with q-symmetric orbits and multi-Hamming weights as given in the first two rows of Table 2 since it is an affine function of this output. Moreover, without the 3 constraints of Eq. (2), its monomials would all be distinct and the total number would merely be the product of the number of monomials in $G_1^{(2)}$ and $G_2^{(2)}$ which is $15 \cdot 5 = 75$.

We now show that the constraints of Eq. (2) eliminate a total of 10 monomials. Most of the q-symmetric orbits of $G_1^{(2)}$ and $G_2^{(2)}$ involve disjoint variable sets and thus produce q-symmetric orbits of the expected multi-Hamming weight and number of monomials. The exceptions are products of the q-symmetric orbits of multi-Hamming weight 1011 with 1100 and 1000 with 1100, each of which giving a nontrivial interaction on the variable X_1.

Considering the sequence of operations that gives monomials of multi-Hamming weight 1011 and 1100 in $G_1^{(2)}$, there exists a representative for both of these orbits that contains the factor $X_1^{[a_0+b_0]}$. Similarly, we may trace the calculation of the q-symmetric orbit 1100 in $G_2^{(2)}$ and find that a representative includes a factor $X_1^{[a_1+b_2]}$. By definition of $\mathbf{A}_3$, we observe that the product of Eq. (3) has monomials including factors of the form

$$\begin{gathered}
X_1^{[a_0+b_0+c_0]}X_1^{[a_1+b_2+c_1]},\\
X_1^{[a_0+b_0+c_0+e]}X_1^{[a_1+b_2+c_1]},\\
X_1^{[a_0+b_0+c_0]}X_1^{[a_1+b_2+c_1+e]},\\
X_1^{[a_0+b_0+c_0+e]}X_1^{[a_1+b_2+c_1+e]},
\end{gathered}$$

exactly as was considered above in the discussion on the number of orbits. Considering the first restriction from Eq. (2), we see that $a_0 + b_0 + c_0 = a_1 + b_2 + c_1 \pmod{2e}$ and thus $2^{a_0+b_0+c_0}$ and $2^{a_1+b_2+c_1}$ are equal in $\mathbb{Z}_{q^2-1}$. Therefore, the above monomial set simplifies in the form

$$\underbrace{X_1^{[a_0+b_0+c_0+1]},\ X_1^{[a_0+b_0+c_0+e+1]}}_{\text{Hamming weight 1}} \text{ and } \underbrace{X_1^{[a_0+b_0+c_0]}X_1^{[a_0+b_0+c_0+e]}}_{\text{Hamming weight 2}}.$$

Thus, we see that for actual parameters, the product of orbits involving X_1 does not split into two orbits of multi-Hamming weight 2111 (resp. 2100) as in the general case, but bifurcates into an orbit of multi-Hamming weight 2111 (resp. 2100), including factors of $X_1^{[a_0+b_0+c_0]}X_1^{[a_0+b_0+c_0+e]}$, and another orbit of multi-Hamming weight 1111 (resp. 1100), including factors of $X_1^{[a_0+b_0+c_0+1]}$ or $X_1^{[a_0+b_0+c_0+e+1]}$. These orbits are the ones which are underlined in Table 4.

To determine their sizes, we note that

$$\left(X_1^{[a_0+b_0+c_0]}X_1^{[a_0+b_0+c_0+e]}\right)^q = X_1^{[a_0+b_0+c_0+e]}X_1^{[a_0+b_0+c_0]},$$

and so they are determined by the powers of the remaining variables. For the 2111 orbit there are then 8 monomials (instead of 16) and for the 2100 orbit there are 2 monomials (instead of 4). Thus, there is a reduction of 10 monomials from the general case.

Finally, for the q-symmetric orbits that experience a multi-Hamming weight fall, the number of monomials is actually the same as in the generic case in which there is no such fall. In particular, $\left(X_1^{[a_0+b_0+c_0+1]}\right)^q = X_1^{[a_0+b_0+c_0+e+1]}$, so there are two possible powers of X_1 in such monomials. Thus, the atypical second instance of a multi-Hamming weight 1111 (resp. 1100) orbit contains the same 16 (resp. 4) monomials as the corresponding multi-Hamming weight 2111 (resp. 2100) orbit in the general case. □

5 Algebraic Attack on DME

In this section, we describe our attack by recovering an *equivalent key* (recall that two secret keys are equivalent if they correspond to the same public key). More precisely, we will explain how to recover an *equivalent last round function*, i.e., whose knowledge allows to complete the attack in the same way as on a 2-round version of DME. Since all rounds have the same structure and since there are only 3 rounds, we argue that this latter step is not more costly (also, recall that 2-round DME has already been shown to be weak, see [4]).

More details on this completion will be given in Subsect. 5.5. Prior to that, Subsect. 5.1 presents tools that are used throughout this section and Subsects. 5.2, 5.3 and 5.4 are dedicated to the recovery of such an equivalent last round function. For the sake of clarity, our description will be limited to the level I parameter set with $e = 32$ (the values $k = 4,\ n = 8$ are common to all levels).

5.1 Using the Big Field Representation

In order to apply the results of Sect. 4, we start by deriving a public key $\widehat{P}$ whose 4 components lie in the ring

$$\mathcal{S} = \mathbb{F}_{q^2}[X_1, \ldots, X_4]/\langle X_1^{q^2} - X_1, \ldots, X_4^{q^2} - X_4\rangle.$$

Recall that the initial public key P is an 8-tuple of polynomials in the quotient ring $\mathcal{R} = \mathbb{F}_q[x_1, \ldots, x_8]/\langle x_1^q - x_1, \ldots, x_8^q - x_8\rangle$. To make the transformation, let ι be the inclusion $\iota : \mathcal{R} \to \mathbb{F}_{q^2}[x_1, \ldots, x_8]/\langle x_1^q - x_1, \ldots, x_8^q - x_8\rangle$ and let $\psi : \mathcal{S} \to \mathbb{F}_{q^2}[x_1, \ldots, x_8]/\langle x_1^q - x_1, \ldots, x_8^q - x_8\rangle$ be the unique ring morphism satisfying $\psi(X_i) = x_{2(i-1)+1} + Ux_{2i}$ for $i \in \{1..4\}$ and $\psi(\lambda) = \lambda$ for $\lambda \in \mathbb{F}_{q^2}$. The latter has inverse

$$\begin{array}{rccc} \psi^{-1} : & \mathbb{F}_{q^2}[x_1, \ldots, x_8]/\langle x_1^q - x_1, \ldots, x_8^q - x_8\rangle & \to & \mathcal{S} \\ & f & \mapsto & f(\xi_1, .., \xi_8), \end{array}$$

where $\forall i \in \{1..8\}, \xi_{2(i-1)+1} = \frac{UX_i^q - U^q X_i}{U - U^q}$, and $\xi_{2i} = \frac{X_i^q - X_i}{U^q - U}$. Concretely, our approach starts by building the set of polynomials $\widehat{P} = (\widehat{P}_1, \ldots, \widehat{P}_4)$ defined by

$$\forall i \in \{1..4\},\ \widehat{P}_i = \psi^{-1}\left(\iota(P_{2(i-1)+1}) + U\iota(P_{2i})\right).$$

Note that the above morphisms are used implicitly in the key generation of DME. In particular, constructing $\widehat{P}$ from P is extremely efficient.

From there, we may consider a calculation path entirely over $\mathbb{F}_{q^2}$. As observed in Sect. 4, the linear maps from the big field representation act *coordinate-wise*. This allows to exploit the following result, stating that such maps "nearly commute" coordinate-wise with power maps of exponent a power of two (the result is presented for general parameters).

Lemma 3 ("Nearly-commuting" trick). *Let $q = 2^e$, let $\widehat{L} : \mathbb{F}_{q^2}^k \to \mathbb{F}_{q^2}^k$ be a coordinate-wise $\mathbb{F}_q$-linear map and let $p = (p_1, \ldots, p_k) : \mathbb{F}_{q^2}^k \to \mathbb{F}_{q^2}^k$ be a map* *such that p_i raises the i-th input to a power of the form 2^{u_i}, $u_i \in \mathbb{Z}_{2e}$. Then there exists another coordinate-wise $\mathbb{F}_q$-linear map $\widehat{M} : \mathbb{F}_{q^2}^k \to \mathbb{F}_{q^2}^k$ such that*

$$p \circ \widehat{L} = \widehat{M} \circ p.$$

Proof. We only need to examine one single index i. To avoid confusion with the notation used for the linear layers, we may write the linear map at this coordinate as $\mathcal{L}_i(X) = AX + BX^q$ (in place of $\widehat{L}_i$). Then we observe that

$$p_i(\mathcal{L}_i(X)) = A^{[u_i]}X^{[u_i]} + B^{[u_i]}X^{[u_i+e]} = \mathcal{M}_i(p_i(X)),$$

where $\mathcal{M}_i(X) = A^{[u_i]}X + B^{[u_i]}X^q$. □

Lemma 3 is instrumental to better grasp the set of possible equivalent keys. For example, assume that a secret key contains the composition of two round functions $R := \widehat{C} \circ \widehat{L} \circ E_{\boldsymbol{A}}$ and $R' := \widehat{C'} \circ \widehat{L'} \circ E_{\boldsymbol{A}'}$, where the matrices $\boldsymbol{A}$ and $\boldsymbol{A}'$ are of the same shape as in Eq. (1). The exponential map $E_{\boldsymbol{A}}$ may be written in a non-unique way as the composition of a power-of-two map p as in Lemma 3 followed by another exponential map $E_{\boldsymbol{A}''}$. We then have

$$\begin{aligned} R \circ R' &= \widehat{C} \circ \widehat{L} \circ E_{\boldsymbol{A}} \circ \widehat{C'} \circ \widehat{L'} \circ E_{\boldsymbol{A}'} \\ &= \widehat{C} \circ \widehat{L} \circ E_{\boldsymbol{A}''} \circ p \circ \widehat{C'} \circ \widehat{L'} \circ E_{\boldsymbol{A}'}. \end{aligned}$$

By linearity, we can write $p \circ \widehat{C'}$ as $\widehat{D'} \circ p$ where $\widehat{D'}$ is still an affine shift and eventually apply Lemma 3 to the composition $p \circ \widehat{L'}$. Namely, there exists a coordinate-wise $\mathbb{F}_q$-linear map $\widehat{M'}$ such that $p \circ \widehat{L'} = \widehat{M'} \circ p$. This gives

$$R \circ R' = \widehat{C} \circ \widehat{L} \circ E_{\boldsymbol{A}''} \circ \widehat{D'} \circ \widehat{M'} \circ \underbrace{p \circ E_{\boldsymbol{A}'}}_{:= E_{\boldsymbol{A}'''}} := R'' \circ R''',$$

which is another composition of two round functions. Our attack will not require the full classification of equivalent keys but we already note that we can obtain other compositions by starting from a different factorization of $E_{\boldsymbol{A}}$.

We will now see that Lemma 3 also plays a crucial role in the recovery of the equivalent last round. We will focus our attention on the 4 secret polynomials that constitute the input of the equivalent last round. These polynomials are entirely private but the analysis of Sect. 4.3 will allow to very efficiently identify the monomials present, see Subsect. 5.2. The associated coefficients, still unknown, will then be found in Subsect. 5.3 by solving polynomial systems. The recovery can be completed easily once these coefficients are known. Thus, polynomial system solving should represent the most costly part of the attack. We provide a complexity analysis of this step in Sect. 5.4.

5.2 Finding the Monomial Content of the Last Round Input

First, let us recall the relation with the public polynomials from $\widehat{P}$. We have

$$\widehat{P} = \widehat{C}_3 \circ \widehat{L}_3 \circ E_{\boldsymbol{A}_3} \circ G,$$

where G is the input of the *genuine* last round R_3 and where

$$\mathbf{A}_3 = \begin{bmatrix} 2^{c_0} & 2^{c_1} & 0 & 0 \\ 0 & 2^{c_2} & 0 & 2^{c_3} \\ 0 & 2^{c_4} & 0 & 2^{c_5} \\ 0 & 0 & 2^{c_6} & 2^{c_7} \end{bmatrix}.$$

If we were to invert the last linear layer, we would obtain

$$\widehat{L}_3^{-1} \circ \widehat{C}_3^{-1} \circ \left(\widehat{P}_1, \widehat{P}_2, \widehat{P}_3, \widehat{P}_4\right) = \left(G_1^{[c_0]} G_2^{[c_1]}, G_2^{[c_2]} G_4^{[c_3]}, G_2^{[c_4]} G_4^{[c_5]}, G_3^{[c_6]} G_4^{[c_7]}\right)$$

$$= E_{\boldsymbol{B}} \circ \left(G_1^{\alpha} G_2^{\beta}, G_2^{\beta} G_4^{[c_5]}, G_2^{[c_4]} G_4^{[c_5]}, G_3^{\gamma} G_4^{[c_5]}\right), \tag{4}$$

where $\alpha = [c_0 - (c_1 + c_3 - c_2 - c_5)]$, $\beta = [c_2 - c_3 + c_5]$, $\gamma = [c_5 + c_6 - c_7]$ and

$$\boldsymbol{B} = \begin{bmatrix} 2^{c_1+c_3-c_2-c_5} & 0 & 0 & 0 \\ 0 & 2^{c_3-c_5} & 0 & 0 \\ 0 & 0 & 2^0 & 0 \\ 0 & 0 & 0 & 2^{c_7-c_5} \end{bmatrix}. \tag{5}$$

The goal here is to have a right-hand-side $\left(G_1^{\alpha} G_2^{\beta}, G_2^{\beta} G_4^{[c_5]}, G_2^{[c_4]} G_4^{[c_5]}, G_3^{\gamma} G_4^{[c_5]}\right)$ such that each of the 4 coordinates shares a factor in common with another. We will use this property later on in this subsection.

To recover the map $E_{\boldsymbol{B}}$, we will use two facts. First, the maps $\widehat{L}_3$ and $\widehat{C}_3$ do not alter the monomial content of q-symmetric polynomials; thus, the monomial content of $\left(G_1^{[c_0]} G_2^{[c_1]}, G_2^{[c_2]} G_4^{[c_3]}, G_2^{[c_4]} G_4^{[c_5]}, G_3^{[c_6]} G_4^{[c_7]}\right)$ is public. Second, the discussion of Sect. 4.3 shows that the q-symmetric orbits in G are known. For instance, the following Table 5 is just Table 2 with the shorthand notation $(G_1, G_2, G_3, G_4) = (G_1^{(2)}, G_2^{(2)}, G_3^{(2)}, G_4^{(2)})$.

Table 5. Known q-symmetric orbits in G.

	Variables	Orbits	#Monomials
G_1	X_1, X_3, X_4	1011:8, 0011:4, 1000:2, 0000:1	15
G_2	X_1, X_2	1100:4, 0000:1	5
G_3	X_1, X_2, X_3	1110:8, 1100:4, 0010:2, 0000:1	15
G_4	X_3, X_4	0011:4, 0000:1	5

In the following, let QSymOrbit be a trivial procedure that computes the q-symmetric orbit of a monomial. For a given monomial μ and a set of indices $S \subset \{1..4\}$, we will also call *truncation of* μ *on* S the highest degree monomial ν in the variables $X_s,\ s \in S$ such that $\nu | \mu$. Applied to a polynomial p and such a set S, the procedure MonomialContent will return the set of truncations of the monomials of p on S. For example,

$$\text{MonomialContent}(X_1X_2X_3X_4 + X_1X_4, \{3,4\}) = \{X_3X_4, X_4\}.$$

We now explain how to recover the difference $c_3 - c_5$ from the public components $\widehat{P}_2$ and $\widehat{P}_3$, by using the peculiar property that G_2 and G_4 have disjoint variable supports; namely, their supports are $\{X_1, X_2\}$ and $\{X_3, X_4\}$, respectively. This difference is the power of two at which we have to raise $\widehat{P}_2$ so that the monomials in the result truncated on the variables $\{X_3, X_4\}$ match the truncation of those of $\widehat{P}_3$ on the same variables (they will correspond to the common factor $G_4^{[c_5]}$). More precisely, we apply the following Algorithm 1, on input $\widehat{P}_3$, $\widehat{P}_2$ and $\{3,4\}$.

Algorithm 1. RetrieveDifference(H_1, H_2, S)

Input: H_1 and H_2 are polynomials and $S \subset \{1..4\}$ is a subset of indices.
Output: Difference Δ

```
$\Delta \leftarrow 0$
for $\mu \in$ MonomialContent$(H_1, S)$ do
    for $r$ in $\{0..e-1\}$ do
        if QSymOrbit$(\mu^{[r]}) \subseteq$ MonomialContent$(H_2, S)$ then
            $\Delta \leftarrow r$
        end if
    end for
end for
return $\Delta$
```

The difference we are looking for belongs to $\mathbb{Z}_{2e}$ and the algorithm will output this difference modulo e. The two possibilities for this difference will both yield equivalent keys (this can be seen from the discussion after Lemma 5). With the same ambiguity, we may recover the other exponents modulo e in an analogous way by

$$c_1 + c_3 - c_2 - c_5 = \text{RetrieveDifference}(\widehat{P}_2^{[c_5-c_3]}, \widehat{P}_1, \{2\}),$$
$$c_7 - c_5 = \text{RetrieveDifference}(\widehat{P}_3, \widehat{P}_4, \{4\}).$$

Once the matrix $\boldsymbol{B}$ is recovered, we use Lemma 3 on Eq. (4) to verify the existence of a coordinate-wise $\mathbb{F}_q$-linear map $\widehat{M}$ and affine shift $\widehat{D}$ such that

$$E_{\boldsymbol{B}^{-1}} \circ \widehat{P} = \widehat{D} \circ \widehat{M} \circ \left(G_1^{\alpha} G_2^{\beta}, G_2^{\beta} G_4^{[c_5]}, G_2^{[c_4]} G_4^{[c_5]}, G_3^{\gamma} G_4^{[c_5]} \right),$$

where the left-hand side is now entirely known and where the monomial content of G_1^{α}, G_2^{β}, G_3^{γ} and $G_4^{[c_5]}$ is also known. This follows from the fact that the variable supports of G_2 and G_4 are disjoint and that each of the four coordinates shares a factor in common with another. The remainder of the recovery of an equivalent round 3 will consist in recovering the associated coefficients in G_1^{α}, G_2^{β}, G_3^{γ} and $G_4^{[c_5]}$ and then the ones of the maps $\widehat{D}$ and $\widehat{M}$, sequentially.

5.3 Finding the Unknown Coefficients

We start from the equation

$$\widehat{M}^{-1} \circ \widehat{D}^{-1} \circ E_{\boldsymbol{B}^{-1}} \circ \widehat{P} = \left(G_1^{\alpha} G_2^{\beta}, G_2^{\beta} G_4^{[c_5]}, G_2^{[c_4]} G_4^{[c_5]}, G_3^{\gamma} G_4^{[c_5]}\right), \tag{6}$$

that corresponds to 4 polynomial equalities. Our approach consists in viewing the unknown coefficients of all polynomials G_1^{α}, G_2^{β}, G_3^{γ}, $G_4^{[c_5]}$ and those of the 4 polynomials in $\widehat{M}^{-1}$ as formal variables in a multivariate polynomial ring and then in deriving equations in these coefficients. The equations will be solved using standard algebraic techniques.

Bilinear Modeling. Our first modeling is obtained from the second and third coordinates of $E_{\boldsymbol{B}^{-1}} \circ \widehat{P}$. Specifically, we know that

$$\begin{aligned} \widehat{M_2}^{-1}\left(\widehat{P}_2^{[c_5-c_3]} - D_2\right) &= G_2^{\beta} G_4^{[c_5]} \\ \widehat{M_3}^{-1}\left(\widehat{P}_3 - D_3\right) &= G_2^{[c_4]} G_4^{[c_5]}. \end{aligned} \tag{7}$$

Using Table 5, the number of formal variables we need to introduce is 5 for each of the polynomials $G_4^{[c_5]}$, $G_2^{[c_4]}$ and G_2^{β} (we arrange these variables as vectors $\boldsymbol{x}$, $\boldsymbol{y}$ and $\boldsymbol{z}$ respectively, all of length 5). Since these maps are linear, we also introduce 2 variables $s_1,\ s_2$ for $\widehat{M_2}^{-1}$ and 2 variables $t_1,\ t_2$ for $\widehat{M_3}^{-1}$, namely

$$\begin{aligned} \widehat{M_2}^{-1}(X) &= s_1 X + s_2 X^q, \\ \widehat{M_3}^{-1}(X) &= t_1 X + t_2 X^q. \end{aligned}$$

We consider the equations obtained by matching coefficients in front of the same monomial in the two polynomial equalities from (7). In both of them, the number of monomials present is 25 by using Table 4. Thus, we can obtain a total of $25+25 = 50$ equations. However, we cannot use the 2 equations corresponding to the constant terms since we do not include the secret coefficients D_2 and D_3 in our variables. Finally, we can divide the two equations in (7) by s_1 and t_1, respectively, to further reduce the number of variables.

Modeling 1. *We obtain in this way an affine bilinear system in* $\mathbb{F}_{q^2}[\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z}, s, t]$ *with* $s = s_2/s_1$ *and* $t = t_2/t_1$ *that contains* 48 *polynomials.*

- *the first equation in* (7) *gives a bilinear system in* $\mathbb{F}_{q^2}[\boldsymbol{x}, \boldsymbol{z}, s]$ *containing polynomials of the form*

$$x_i z_j + \ell_{i,j}(s),$$

  where $\ell_{i,j}(s)$ *linear affine in* s*, for any* $(i,j) \in \{1..5\}^2 \setminus \{(5,5)\}$*;*
- *the second equation in* (7) *gives a bilinear system in* $\mathbb{F}_{q^2}[\boldsymbol{x}, \boldsymbol{y}, t]$ *containing polynomials of the form*

$$x_i y_j + m_{i,j}(t),$$

  where $m_{i,j}(t)$ *linear affine in* t*, for any* $(i,j) \in \{1..5\}^2 \setminus \{(5,5)\}$*.*

Remark 2. *Anecdotally, Modeling 1 can be seen as a subset of* 48 *out of* 50 *equations which model two rank one MinRank problems in* $\mathbb{F}_{q^2}^{5\times 5}$ *with matrices* $(\mathbf{M}_1, \mathbf{M}_2)$ *and* $(\mathbf{N}_1, \mathbf{N}_2)$ *respectively correlated in that we look for solutions* $\boldsymbol{x}^\mathsf{T}\boldsymbol{y} = \mathbf{M}_1 + s\mathbf{M}_2$ *and* $\boldsymbol{x}^\mathsf{T}\boldsymbol{z} = \mathbf{N}_1 + t\mathbf{N}_2$ *for the same* $\boldsymbol{x}$*. We simply do not consider the two equations coming from entry* $(5,5)$*.*

In Lemma 4, we study the variety of Modeling 1 intersected with the coordinate ring of $\mathbb{F}_{q^2}^{17}$. This is also the variety (over the algebraic closure) of the ideal J generated by Modeling 1 together with the field equations from $\mathbb{F}_{q^2}$. For practical purposes, we do not, in practice, add equations of such a high degree into the system, since a Gröbner basis over an algebraic closure of $\mathbb{F}_{q^2}$ can be computed so easily on Modeling 1 alone.

Lemma 4. *Let* J *be the ideal generated by Modeling 1 along with the field equations from* $\mathbb{F}_{q^2}$*. The variety* $V(J)$ *has* 1 *degree of freedom over* $\mathbb{F}_{q^2}$*. Moreover, by fixing one variable different from* s *and* t*, we get a variety of size at least* 2*.*

Proof. If $(\boldsymbol{x}, \boldsymbol{z}, \boldsymbol{y}, s, t)$ is a solution in $V(J)$ then $(\lambda\boldsymbol{x}, \lambda^{-1}\boldsymbol{z}, \lambda^{-1}\boldsymbol{y}, s, t)$ is another solution for any non-zero $\lambda \in \mathbb{F}_{q^2}^*$. There is an additional symmetry coming from q-powering. Indeed, let us consider a solution $(\boldsymbol{x}, \boldsymbol{z}, \boldsymbol{y}, s, t)$ with a prescribed coordinate in $\boldsymbol{x}$, $\boldsymbol{y}$ or $\boldsymbol{z}$ (so that it is a solution to Modeling 1 specialized with this constraint). To simplify the notation, let us write S_i for the known polynomial $(E_{\mathbf{B}_1} \circ \widehat{P})_i$, $i \in \{2, 3\}$. Let also $(\overline{G}_4, \overline{G}_{2,1}, \overline{G}_{2,2})$ be the triple of polynomials corresponding to $(\boldsymbol{x}, \boldsymbol{z}, \boldsymbol{y})$. By q-powering the equalities

$$\begin{aligned} S_2^q + s^q S_2 &= \overline{G}_{2,1}\overline{G}_4, \\ S_3^q + t^q S_3 &= \overline{G}_{2,2}\overline{G}_4 \end{aligned}$$

we get

$$\begin{aligned} S_2 + sS_2^q &= (\overline{G}_{2,1})^q(\overline{G}_4)^q, \\ S_3 + tS_3^q &= (\overline{G}_{2,2})^q(\overline{G}_4)^q. \end{aligned}$$

By dividing the first equality by s and the second one by t to adhere to the restriction in our modeling that the linear maps have a coefficient of 1 in front of X, we finally obtain

$$\begin{aligned} (1/s)S_2 + S_2^q &= (\overline{G}_{2,1}^q/s)(\overline{G}_4)^q, \\ (1/t)S_3 + S_3^q &= (\overline{G}_{2,2}^q/t)(\overline{G}_4)^q. \end{aligned}$$

This yields the new solution $(\boldsymbol{x}^q, (\boldsymbol{z}^q/s), (\boldsymbol{y}^q/t), 1/s, 1/t)$ to Modeling 1. Note that, in general, the coordinate prescribed in the former solution $(\boldsymbol{x}, \boldsymbol{z}, \boldsymbol{y}, s, t)$ will here have a different value. However, by $\mathbb{F}_{q^2}$-linearity, exactly one of the other solutions $(\lambda\boldsymbol{x}^q, \lambda^{-1}(\boldsymbol{z}^q/s), \lambda^{-1}(\boldsymbol{y}^q/t), 1/s, 1/t)$ will have the right value at this coordinate. □

Due to Lemma 4, we have the freedom to specialize one of the variables other than s and t in Modeling 1 and retain a consistent system. Any choice of such a variable produces linear equations. However, choosing to fix a variable x_i for some $i \neq 5$ generates the greatest number of linear equations. In the following, we suppose that we fix the value of variable x_1.

The solution to the specialized system will provide candidates for the maps $\widehat{M}_2^{-1}, \widehat{M}_3^{-1}, G_2^{\beta}$, $G_2^{[c_4]}$ and $G_4^{[c_5]}$. There are, however, two complications. First, it is necessary to enforce the restriction that $\beta = [c_4 - d]$, i.e., $G_2^{\beta}/G_2^{[c_4-d]} = 1$, by computing this quotient $\tau \in \mathbb{F}_{q^2}$ and replacing G_2^{β} with τG_2^{β}. Second, the proof of Lemma 4 shows that, in particular, the candidate for $G_4^{[c_5]}$ might instead correspond to $\lambda G_4^{[c_5]}$ or $\lambda G_4^{[c_5+e]}$ for some $\lambda \in \mathbb{F}_{q^2}$. Since both multiplication by λ or τ and exponentiation by q are $\mathbb{F}_q$-linear, this is a not an issue as these operations can both be absorbed into the linear layer of the previous round (using the reasoning sketched right after Lemma 3). Thus, we may assume that we have the correct candidates. From these candidates, we can also recover the affine shift constants D_2 and D_3. Indeed, these solutions reveal all the quantities in Eq. (7) other than D_2 and D_3. In particular, these equations are readily solved for the correct values of the affine shift. In the following, we denote by $(\overline{G}_4, \overline{G}_{2,1}, \overline{G}_{2,2})$ the solutions found for $(G_4^{[c_5]}, G_2^{\beta}, G_2^{[c4]})$.

Remaining Coefficients by Solving Linear Systems. The coefficients that we still have to recover of those of $\widehat{M}_i^{-1}$, $\widehat{D}_i$ for $i \in \{1, 4\}$ and those of the polynomials G_1^{α}, G_3^{γ}. To do so, we come back to Eq. (6) and we plug the solutions previously found into coordinates 1 and 4:

$$\begin{aligned} \widehat{M}_1^{-1}\left(\widehat{P}_1^{[c_2+c_5-c_1-c_3]} - D_1\right) &= G_1^{\alpha}\overline{G}_{2,1} \\ \widehat{M}_4^{-1}\left(\widehat{P}_1^{[c_5-c_7]} - D_4\right) &= G_3^{\gamma}\overline{G}_4. \end{aligned} \tag{8}$$

By introducing formal variables for $\widehat{M}_1^{-1}$ and G_1^{α} in the first equation and for $\widehat{M}_4^{-1}$ and G_3^{γ} in the second equation, the reasoning used to obtain Modeling 1 yields two linear systems. These two systems contain $64 = 65 - 1$ equations (by using Table 4 and dropping the constant term) in $17 = 2 + 15$ variables (by using Table 2 or Table 5). These systems are much easier to solve than Modeling 1. From their solutions we can readily recover G_1^{α}, G_3^{γ} and $\widehat{M}_i^{-1}$ for $i \in \{1, 4\}$. Finally, we can retrieve D_1 and D_4 by coming back to Eq. (8) since all the other values are now known.

To unify notation, in the following let $\widetilde{L}_3$ be the recovered value for $\widehat{M}$ and let $\widetilde{C}_3$ be the recovered value for $\widehat{D}$. These values will be used in Subsect. 5.5 to derive an equivalent last round function (it remains to precise the exponential component). Prior to that, Subsect. 5.4 examines the complexity of recovering $\widetilde{L}_3$ and $\widetilde{C}_3$. Our discussion so far shows that we can restrict ourselves to the solving of Modeling 1.

5.4 Complexity of Solving Specialized Modeling 1

Recall that the equations in Modeling 1 are of the form $x_i z_j + \ell_{i,j}(s) = 0$ or $x_i y_j + m_{i,j}(t) = 0$ for $(i, j) \neq (5, 5)$, were $\ell_{i,j}$, $m_{i,j}$ are univariate polynomials of degree 1. Recall also that we decided to set a constraint $x_1 = a$ in order to obtain a variety of expected size 2.

The Gröbner basis computation of the specialized system proceeds very simply. Setting $x_1 = a$ yields linear relations of the form $z_j + a^{-1}\ell_{1,j}(s) = 0$ and $y_j + a^{-1}m_{1,j}(t) = 0$. Thus, by substitution into the other equations that remain of degree 2, the entire system implicitly reduces to an overdetermined bilinear system of $38 = 48 - 10$ equations in the 6 variables s, t and x_i for $i \neq 1$ in which each relation contains a single quadratic monomial of the form $x_i s$ or $x_i t$. As outlined in detail in Appendix B, such a system is solved in degree 2. As suggested in Lemma 4, we find that the Gröbner basis consists of a single univariate quadratic equation and an otherwise linear system of equations producing precisely two solutions.

In fact, even a generic system consisting of 38 quadratic (not necessarily bilinear) equations and 10 linear equations in 16 variables[2] is solved at degree 2 (using standard arguments, e.g., Hilbert series). We may therefore provide a rather gross overestimate of the complexity of solving the system of Modeling 1 over $\mathbb{F}_{q^2}$ with the formula

$$\text{Complexity}_{\text{Modeling 1}} = \mathcal{O}\left(\binom{16+2}{2}^{\omega}\right),$$

in which ω is the linear algebra constant, typically assumed to be $\omega = 2.81$ for Strassen's Algorithm [16]. Finally, by using the standard formula

$$\#\text{ gates per } \mathbb{F}_q\text{-multiplication} = 2\left(\log_2 q\right)^2 + \log_2 q,$$

we obtain the following results on the NIST parameter sets (Table 6):

Table 6. Conservative estimate for the cost of solving specialized Modeling 1.

Level	Value of q	Gate Count
I	2^{32}	2^{31}
III	2^{48}	2^{32}
V	2^{64}	2^{33}

[2] For simplicity, our implementation does not use $17 - 1 = 16$ variables. Instead, it adds an equation of the form $x_1 - a$, resulting in a system of 49 equations in 17 variables.

5.5 Completing an Equivalent Round Function

We finally explain that we can efficiently recover an equivalent key from the maps $\widetilde{L}_3$ and $\widetilde{C}_3$ that have just been retrieved. Lemma 5 below shows that we can actually construct one which is identical to the genuine key in its first round.

Lemma 5. *Let $\widetilde{L}_3$ (resp. $\widetilde{C}_3$) denote the retrieved linear map (resp. affine shift), let $\boldsymbol{B}$ be the diagonal matrix of Eq.* (5) *and let*

$$\mathbf{C} = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 2^{57} & 0 & 1 \\ 0 & 0 & 1 & 1 \end{bmatrix}.$$

Then the round function $\widetilde{R}_3$ given by the composition of $\widetilde{C}_3$, $\widetilde{L}_3$ and $E_{\boldsymbol{B}^{-1}\boldsymbol{C}}$ is an equivalent round function, in the sense that there exists a round function R'_2 satisfying

$$\widehat{P} = \widetilde{R}_3 \circ R'_2 \circ R_1 \circ \widehat{C}_0 \circ \widehat{L}_0.$$

Proof. Let $(\widetilde{G}_1, \widetilde{G}_2, \widetilde{G}_3, \widetilde{G}_4)$ be the components recovered in Subsect. 5.4. By the last equation in this subsection, they satisfy $\widehat{P} = \widetilde{C}_3 \circ \widetilde{L}_3 \circ E_{\mathbf{B}^{-1}\boldsymbol{C}}(\widetilde{G}_1, \widetilde{G}_2, \widetilde{G}_3, \widetilde{G}_4)$. We then obtain

$$\begin{aligned}\widehat{P} &= \widetilde{C}_3 \circ \widetilde{L}_3 \circ E_{\mathbf{B}^{-1}\boldsymbol{C}}(G_1^{\alpha}, G_2^{\beta}, G_3^{\gamma}, G_4^{[c_5]}) = \widetilde{R}_3(G_1^{\alpha}, G_2^{\beta}, G_3^{\gamma}, G_4^{[c_5]}) \\ &= E_{\mathbf{A}_3}(G_1, G_2, G_3, G_4),\end{aligned}$$

where the top equality is by definition of the $\widetilde{G}_i$'s and where the bottom one follows from the definition of $\boldsymbol{B}$ in Eq. (5) and from the condition on the exponents given by Eq. (2). By definition of (G_1, G_2, G_3, G_4) we also have

$$(G_1, G_2, G_3, G_4) = R_2 \circ R_1 \circ \widehat{C}_0 \circ \widehat{L}_0(X_1, \ldots, X_4).$$

Thus, it remains to find a round function R'_2 such that

$$R'_2 \circ R_1 \circ \widehat{C}_0 \circ \widehat{L}_0 = E_{\boldsymbol{T}} \circ R_2 \circ R_1 \circ \widehat{C}_0 \circ \widehat{L}_0,$$

where $\boldsymbol{T}$ is the diagonal matrix with diagonal entries α, β, γ, and $[c_5]$. By Lemma 3, there exists a linear map $L'_2 : \mathbb{F}_{q^2}^4 \to \mathbb{F}_{q^2}^4$ such that $E_{\boldsymbol{T}} \circ \widehat{L}_2 = L'_2 \circ E_{\boldsymbol{T}}$. If we denote by C'_2 the affine shift derived from $\widehat{C}_2$ by raising the constants to the powers α, β, γ and $[c_5]$, then the round function

$$R'_2 = C'_2 \circ L'_2 \circ E_{\boldsymbol{T}\boldsymbol{A}_2}$$

satisfies the criterion. □

Lemma 5 guarantees that we can view $(\widetilde{G}_1, \widetilde{G}_2, \widetilde{G}_3, \widetilde{G}_4)$ as the public key of a 2-round DME scheme. From there, we can apply the attack of Beullens [4] or iterate our procedure. In the latter case, the modelings derived to recover equivalent functions $\widetilde{R}_2$ and $\widetilde{R}_1$ will involve different components than the ones we described for $\widetilde{R}_3$. However, the shape of the exponent matrices $\boldsymbol{T}\boldsymbol{A}_2$ and $\boldsymbol{A}_1$ ensure that these systems will not be harder to solve than the bilinear system of above.

6 Experimental Results

A Magma implementation of the attack on the Level I parameters is available at https://github.com/ppbriaud/DMEattack. For this parameter set, the attack takes between roughly 500 ms and 1 s. Perhaps surprisingly, the main cost in practice corresponds to the application of Algorithm 1 and not to polynomial system solving. The reason is probably a poor implementation of this part.

The code can be easily adapted to the other security levels by changing the values of d and e, with no computational overhead other than the increased cost of the field arithmetic.

A Proof of Lemma 1

Any scalar $D \in \mathbb{F}_{q^2}$ can clearly be viewed as a q-symmetric polynomial with monomial support reduced to the unique orbit $\{1\}$ if $D \neq 0$ and empty monomial support if $D = 0$. This proves Statement 1.

Statement 2 is similarly trivial. The monomial support of the sum $p_1 + p_2$ is perfectly controlled except if there is a cancellation when adding two coefficients for the same monomial. Assuming that the monomial support of both p_1 and p_2 are fixed and that their coefficients are randomly sampled, such a cancellation occurs with a probability bounded by the minimum of the numbers of monomials in the polynomials divided by $q^2 - 1$, which is assumed to be small.

The monomial support of a polynomial product is also perfectly understood with very high probability (which has nothing to do with q-symmetry). If we let M_i be monomial support of p_i for $\{1, 2\}$, the monomial support M of p_1p_2 is included in the set $M_{\max}$ containing the distinct elements of the list

$$[\mu_1\mu_2 : \mu_1 \in M_1,\ \mu_2 \in M_2]\,.$$

This inclusion is strict precisely when there exists $\mu \in M_{\max}$ with cancellation

$$\sum_{\substack{\mu_1 \in M_1 \\ \mu_2 \in M_2 \\ \mu_1\mu_2 = \mu}} \operatorname{coef}(\mu_1, p_1)\operatorname{coef}(\mu_2, p_2) = 0.$$

Just as in the previous case, this event is of very low probability. If now p_1 and p_2 are q-symmetric with $M = M_{\max}$, let us consider an arbitrary element $\mu = \mu_1\mu_2 \in M_{\max}$ which is thus a monomial appearing in p_1p_2. Any monomial $\widetilde{\mu}$ obtained by q-powering variables in μ can clearly be written as $\widetilde{\mu_1}\widetilde{\mu_2}$, where $\widetilde{\mu_i}$ belongs[3] to the q-symmetric orbit of μ_i for $\{1, 2\}$. By q-symmetry of p_1 (resp. p_2) we have $\widetilde{\mu_1} \in M_1$ (resp. $\widetilde{\mu_2} \in M_2$), hence $\widetilde{\mu} \in M_{\max}$ and this monomial necessarily appears in p_1p_2. This shows Statement 3.

Statement 4 is obviously a particular case of Statement 3.

[3] Possibly $\widetilde{\mu_i} = \mu_i$.

Finally, Statement 5 is a consequence of the previous results along with the fact that every $\mathbb{F}_q$-linear map on $\mathbb{F}_{q^2}$ has a linearized polynomial form. Specifically, the reason that $L(p)$ has the same monomial content as p is because q-powering simply permutes the monomials of a q-symmetric polynomial and cannot create coefficient cancellations.

B Gröbner Bases for Specialized Modeling 1

We detail the behaviour of the Gröbner basis algorithm on Modeling 1 when we fix one variable x_i for some $i \neq 5$ to a nonzero value $a \in \mathbb{F}_{q^2}^*$. This specialization may represent the most favorable case as we maximize the number of linear equations produced at the first step. Our description is made for a graded order $<$ such that $s < t < \boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z}$ and our goal is mainly to describe the experimental steps reported in Fig. 3 below[4].

STEP 1. Former equations with leading terms divisible by x_i now become equations with leading terms z_j and y_k for $j,\ k \in \{1..5\}$. More precisely, we get $z_j - a^{-1}\ell_{i,j}(s) = 0$ for $j \in \{1..5\}$ and $y_k - a^{-1}m_{i,k}(t) = 0$ for $k \in \{1..5\}$. This should explain the 10 linear degree fall polynomials that are observed.

STEP 2. The second step is in degree 2. There, we use the degree fall polynomials found in STEP 1 to "remove" the $\boldsymbol{y}$ and $\boldsymbol{z}$ blocks in the initial modeling.

1. From the former equations with leading terms x_5z_j for $j \neq 5$, we now get 4 affine equations in $\mathbb{F}_{q^2}[x_5, s, 1]$ whose unique quadratic monomial is x_5s. Similarly, we obtain 4 affine equations in $\mathbb{F}_{q^2}[x_5, t, 1]$ whose unique quadratic monomial is x_5t from the previous equations with leading terms x_5y_k for $k \neq 5$. In addition to this new leading term x_5s (resp. x_5t) we thus expect 3 linear equations in $\mathbb{F}_{q^2}[x_5, s, 1]$ (resp. 3 linear equations in $\mathbb{F}_{q^2}[t, x_5, 1]$). By doing linear combinations between these degree 1 polynomials we may generate one degree 1 polynomial with leading term x_5 and one in $\mathbb{F}_{q^2}[s, t, 1]$ with leading term t. We cannot create a degree 1 equation with leading term s because we have > 1 solution (see Lemma 4).
2. From the former equations with leading terms x_uz_j for $u \neq \{i, 5\}$, we now get 5 affine equations with leading monomial x_us and degree ≤ 1 part in x_u and s. Similarly, we obtain 5 affine equations with leading monomial x_ut and degree ≤ 1 part in x_u and t from the previous equations with leading terms x_uy_k. This time we produce the leading terms $\{x_us, x_ut\}$ as well as x_u for $u \neq \{i, 5\}$.

Overall, we create $2 + 3 \cdot 2 = 8$ new quadratic leading monomials and $2 + 3 \cdot 1 = 5$ new degree 1 leading monomials. This is in accordance with the behaviour observed in Magma.

[4] This figure was generated using the level I parameters but the behaviour would be analogous for the other levels since the only difference is the value of q.

```
*******
STEP 1
Basis length: 49, queue length: 310, step degree: 2, num pairs: 10
Basis total mons: 146, average length: 2.980
Number of S-polynomials: 10, different lcms: 10
Number of pair polynomials: 10, at 23 column(s), 0.000
Average length for reductees: 2.00 [10], reductors: 3.00 [10]
Symbolic reduction time: 0.000, column sort time: 0.000
10 + 10 = 20 rows / 23 columns out of 171 (13.450%)
Density: 10.87% / 13.854% (2.5/r), total: 50 (0.0MB)
Matrix construction time: 0.000
Matrix size: 20 by 23
Current max memory usage: 32.1MB (=max)
Before ech memory: 32.1MB (=max)
Row sort time: 0.000
    0.000 + 0.000 + 0.000 = 0.000 [10]
    Echelonization time: 0.000
After ech memory: 32.1MB (=max)
New rules time: 0.000
Num new polynomials: 10 (100.0%), min deg: 1 [10], av deg: 1.0
Degree counts: 1:10
Queue insertion time: 0.000
Number of linears: 10
New max step: 1, time: 0.000
Step 1 time: 0.000, [0.001], mat/total: 0.000/0.000, mem: 32.1MB (=max)

*******
STEP 2
Basis length: 59, queue length: 338, step degree: 2, num pairs: 38
Basis total mons: 176, average length: 2.983
Number of S-polynomials: 38, different lcms: 38
Number of pair polynomials: 38, at 53 column(s), 0.000
Average length for reductees: 3.00 [38], reductors: 3.00 [38]
Symbolic reduction time: 0.000, column sort time: 0.000
38 + 38 = 76 rows / 53 columns out of 171 (30.994%)
Density: 5.6604% / 9.7791% (3/r), total: 228 (0.0MB)
Matrix construction time: 0.000
Matrix size: 76 by 53
Current max memory usage: 32.1MB (=max)
Before ech memory: 32.1MB (=max)
Row sort time: 0.000
    0.000 + 0.000 + 0.000 = 0.000 [13]
    Echelonization time: 0.000
After ech memory: 32.1MB (=max)
New rules time: 0.000
Num new polynomials: 13 (34.2%), min deg: 1 [5], av deg: 1.6
Degree counts: 1:5 2:8
Queue insertion time: 0.000
Number of linears: 15
Step 2 time: 0.000, [0.001], mat/total: 0.000/0.000, mem: 32.1MB (=max)
```

Fig. 3. Trace of Magma's F4 algorithm [8] on Modeling 1 with one variable x_i, $i \neq 5$, fixed to a nonzero value (for security level I).

```
*******
STEP 3
Basis length: 72, queue length: 308, step degree: 2, num pairs: 8
Basis total mons: 215, average length: 2.986
Number of S-polynomials: 8, different lcms: 8
Number of pair polynomials: 8, at 15 column(s), 0.000
Average length for reductees: 3.00 [8], reductors: 3.00 [12]
Symbolic reduction time: 0.000, column sort time: 0.000
8 + 12 = 20 rows / 15 columns out of 171 (8.772%)
Density: 20% / 36.011% (3/r), total: 60 (0.0MB)
Matrix construction time: 0.000
Matrix size: 20 by 15
Current max memory usage: 32.1MB (=max)
Before ech memory: 32.1MB (=max)
Row sort time: 0.000
    0.000 + 0.000 + 0.000 = 0.000 [1]
    Echelonization time: 0.000
After ech memory: 32.1MB (=max)
New rules time: 0.000
Num new polynomials: 1 (12.5%), min deg: 2 [1], av deg: 2.0
Degree counts: 2:1
Queue insertion time: 0.000
Number of linears: 15
Step 3 time: 0.000, [0.001], mat/total: 0.000/0.000, mem: 32.1MB (=max)

*******
STEP 4
Basis length: 73, queue length: 300, step degree: 3, num pairs: 300
Basis total mons: 218, average length: 2.986
300 pairs eliminated
No pairs to reduce
Pair elimination time: 0.000

Do extern interreduction (length 25)
    INTERREDUCE 17 polynomial(s)
        Symbolic reduction time: 0.000
        Column sort time: 0.000
        17 + 0 = 17 rows / 19 columns
        Density: 15.48% / 35.791% (2.9412/r), total: 50 (0.0MB)
        Row sort time: 0.000
            0.000 + 0.000 = 0.000 [17]
            Echelonization time: 0.000
        Total reduction time: 0.000
    Reduction time: 0.000
Final extern interreduction time: 0.000

Final basis length: 17
Number of pairs: 56
Total pair setup time: 0.000
Max step: 2, time: 0.010
Max num entries matrix: 76 by 53
Max num rows matrix: 76 by 53
Approx mat cost: 8321.32, sym red cost: 338
Approx mat time: 0.000, sym red time: 0.000, total 0.000
Total symbolic reduction time: 0.000
Total column sort time: 0.000
Total row sort time: 0.000
Total matrix time: 0.010
Total new polys time: 0.000
Total queue update time: 0.000
Total Faugere F4 time: 0.010, real time: 0.004
```

Fig. 3. (*continued*)

STEP 3. We are left with the unique variable s if we simplify the system using the linear equations generated at STEP 2. The only degree 2 polynomial occurring at this step is univariate and it has leading monomial s^2.

References

1. IEEE standard specifications for public-key cryptography: IEEE Std 1363-2000, pp. 1–228 (2000). https://doi.org/10.1109/IEEESTD.2000.92292
2. Alagic, G., et al.: Status report on the third round of the NIST post-quantum cryptography standardization process. Technical report. NIST Interagency or Internal Report (IR) 8413, National Institute of Standards and Technology, Gaithersburg, MD, July 2022. https://doi.org/10.6028/NIST.IR.8413-upd1
3. Avendaño, M., Marco, M.: A structural attack to the DME-(3,2,q) cryptosystem. Finite Fields Appl. **71**, 101810 (2021). https://doi.org/10.1016/j.ffa.2021.101810, https://www.sciencedirect.com/science/article/pii/S1071579721000046
4. Beullens, W.: Round 1 official comments on DME. NIST CSRC (2017). https://csrc.nist.gov/CSRC/media/Projects/Post-Quantum-Cryptography/documents/round-1/official-comments/DME-official-comment.pdf
5. Beullens, W.: MAYO: practical post-quantum signatures from oil-and-vinegar maps. In: AlTawy, R., Hülsing, A. (eds.) Selected Areas in Cryptography: 28th International Conference, Virtual Event, 29 September–1 October 2021, Revised Selected Papers, pp. 355–376. Springer, Cham (2021). https://doi.org/10.1007/978-3-030-99277-4_17
6. Bouillaguet, C., Fouque, P., Véber, A.: Graph-theoretic algorithms for the "isomorphism of polynomials" problem. In: Johansson, T., Nguyen, P.Q. (eds.) Advances in Cryptology - EUROCRYPT 2013, 32nd Annual International Conference on the Theory and Applications of Cryptographic Techniques, Athens, Greece, 26–30 May 2013, Proceedings. LNCS, vol. 7881, pp. 211–227. Springer, Cham (2013). https://doi.org/10.1007/978-3-642-38348-9_13
7. Faugère, J., Perret, L.: An efficient algorithm for decomposing multivariate polynomials and its applications to cryptography. J. Symb. Comput. **44**(12), 1676–1689 (2009). https://doi.org/10.1016/j.jsc.2008.02.005
8. Faugére, J.C.: A new efficient algorithm for computing Gröbner bases (F4). J. Pure Appl. Algebra **139**(1), 61–88 (1999). https://doi.org/10.1016/S0022-4049(99)00005-5, https://www.sciencedirect.com/science/article/pii/S0022404999000055
9. Group, C.T.: Call for additional digital signature schemes for the post-quantum cryptography standardization process. NIST CSRC (2022). https://csrc.nist.gov/csrc/media/Projects/pqc-dig-sig/documents/call-for-proposals-dig-sig-sept-2022.pdf
10. Hashimoto, Y.: High-rank attack on HMFEv. JSIAM Lett. **10**, 21–24 (2018). https://doi.org/10.14495/jsiaml.10.21
11. Luengo, I., Avendaño, M.: DME: multivariate signature public key scheme. NIST CSRC (2023). https://csrc.nist.gov/Projects/pqc-dig-sig/round-1-additional-signatures
12. Luengo, I., Avendaño, M., Marco, M.: DME: a public key, signature and KEM system based on double exponentiation with matrix exponents. NIST CSRC (2017). https://csrc.nist.gov/Projects/post-quantum-cryptography/round-1-submissions

13. Patarin, J., Goubin, L.: Trapdoor one-way permutations and multivariate polynomials. In: Han, Y., Okamoto, T., Qing, S. (eds.) Information and Communication Security, First International Conference, ICICS 1997, Beijing, China, 11–14 November 1997, Proceedings. LNCS, vol. 1334, pp. 356–368. Springer, Cham (1997). https://doi.org/10.1007/BFb0028491
14. Petzoldt, A., Chen, M., Ding, J., Yang, B.: HMFEv - an efficient multivariate signature scheme. In: Lange, T., Takagi, T. (eds.) Post-Quantum Cryptography - 8th International Workshop, PQCrypto 2017, Utrecht, The Netherlands, 26–28 June 2017, Proceedings. LNCS, vol. 10346, pp. 205–223. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-59879-6_12
15. Scemama, A.: A cryptanalysis of the double-round quadratic cryptosystem. In: Nam, K.H., Rhee, G. (eds.) Information Security and Cryptology - ICISC 2007, pp. 27–36. Springer, Heidelberg (2007). https://doi.org/10.1007/978-3-540-76788-6_3
16. Volker, S.: Gaussian elimination is not optimal. Numer. Math. **13**, 354–356 (1969)
17. Ye, D., Lam, K., Dai, Z.: Cryptanalysis of “2 r” schemes. In: Wiener, M.J. (ed.) Advances in Cryptology - CRYPTO 1999, 19th Annual International Cryptology Conference, Santa Barbara, California, USA, 15–19 August 1999, Proceedings. LNCS, vol. 1666, pp. 315–325. Springer, Cham (1999). https://doi.org/10.1007/3-540-48405-1_20