A new multivariate primitive from CCZ equivalence

Marco Calderini[1], Alessio Caminata[2] and Irene Villa[2]

[1] Dipartimento di Matematica, Università degli studi di Trento
via Sommarive 14, 38123 Povo, Trento, Italy
[2] Dipartimento di Matematica, Dipartimento di Eccellenza 2023-2027, Università di Genova
via Dodecaneso 35, 16146, Genova, Italy

Abstract. Multivariate Cryptography is one of the main candidates for Post-quantum Cryptography. Multivariate schemes are usually constructed by applying two secret affine invertible transformations $\mathcal{S}, \mathcal{T}$ to a set of multivariate polynomials $\mathcal{F}$ (often quadratic). The secret polynomials $\mathcal{F}$ posses a trapdoor that allows the legitimate user to find a solution of the corresponding system, while the public polynomials $\mathcal{G} = \mathcal{S} \circ \mathcal{F} \circ \mathcal{T}$ look like random polynomials. The polynomials $\mathcal{G}$ and $\mathcal{F}$ are said to be affine equivalent. In this article, we present a more general way of constructing a multivariate scheme by considering the CCZ equivalence, which has been introduced and studied in the context of vectorial Boolean functions.

Keywords: Post-quantum Cryptography · Multivariate Cryptography · Boolean functions · CCZ equivalence

1 Introduction

In the last few decades, many new proposals for public-key cryptosystems have been presented to the scientific community. With the advent of *Post-quantum Cryptography* [BL17] following the development of Shor's algorithm, many cryptographers have focused on finding quantum-resistant public-key systems. *Multivariate public-key Cryptography* is one of the main families of post-quantum cryptosystems. These systems base their security on the difficulty of solving a set of randomly chosen nonlinear multivariate polynomials over a finite field. So far, there is no evidence that quantum computers can solve such sets of multivariate polynomials efficiently.

A multivariate public-key cryptosystem involves a public key comprising multivariate polynomials $f^{(1)}, \dots, f^{(m)}$ in $\mathbb{F}_q[x_1, \dots, x_n]$, where $\mathbb{F}_q$ is a finite field with q elements. In order to keep the public key size not too large, usually, quadratic polynomials are considered. The secret key is some information (about the construction of the polynomials $f^{(i)}$) which allows to solve the system $f^{(1)} = y_1, \dots, f^{(m)} = y_m$ efficiently for some $y_1, \dots, y_m \in \mathbb{F}_q$. To encrypt a message $(x'_1, \dots, x'_n) \in \mathbb{F}_q^n$, the sender computes $y_i = f^{(i)}(x'_1, \dots, x'_n)$, for $i = 1, \dots, m$, and sends $(y_1, \dots, y_m)$ to the receiver. With the secret key, the receiver can solve the system and recover the original message.

One of the main methods to achieve the previous scheme is the *Bipolar Construction*. From a system $\mathcal{F}$ of m equations in n variables (possibly quadratic) relatively easy to invert, the secret key is composed of $\mathcal{F}$ and two randomly chosen affine bijections $\mathcal{S}$ and $\mathcal{T}$; the public key is the system $\mathcal{G} = \mathcal{S} \circ \mathcal{F} \circ \mathcal{T}$. The obtained system $\mathcal{G}$ (still quadratic) is now assumed to be not easy to invert, since it should be hardly distinguishable from a

E-mail: marco.calderini@unitn.it (Marco Calderini), alessio.caminata@unige.it (Alessio Caminata), villa.i@dima.unige.it (Irene Villa)

random system. Notice that, in this case, finding a preimage of $y = (y_1, \ldots, y_m)$ reduces to finding a preimage for $\mathcal{F}$ of $\mathcal{S}^{-1}(y)$ and then apply $\mathcal{T}^{-1}$. Hence, the core idea of the Bipolar Construction is to hide the structure of the *central map* $\mathcal{F}$ by applying two random affine bijections to the input and to the output of $\mathcal{F}$. This corresponds to randomly taking a system in the affine-equivalence class of $\mathcal{F}$.

The Bipolar Construction method has found extensive application in numerous significant multivariate schemes, including Matsumoto–Imai [MI88], HFE [Pat96], Oil and Vinegar [Pat97], and Rainbow [DS05]. Unfortunately, the affine equivalence keeps many properties and structures of a system of equations. This might allow an attacker to use them to break the system. Consequently, public polynomials $\mathcal{G}$ often fail to exhibit true randomness, undermining the scheme's robustness.

In order to better explain this phenomenon, we briefly recall the example of the Matsumoto–Imai (MI) Cryptosystem [MI88]. In this scheme, the authors consider q a power of two and $m = n$. The central system $\mathcal{F}$ is seen as a quadratic function $F : \mathbb{E} \to \mathbb{E}$, $F(x) = x^{q^j+1}$ where $\mathbb{E} = \mathbb{F}_{q^n}$ and j is chosen such that $\gcd(q^n - 1, q^j + 1) = 1$. Then, by applying the standard isomorphism $\phi : \mathbb{F}_q^n \to \mathbb{E}$, the map is expanded as a system of n equations over $\mathbb{F}_q$, $\mathcal{F} = \phi \circ F \circ \phi^{-1} : \mathbb{F}_q^n \to \mathbb{F}_q^n$. The quadratic function $F(x) = x^{q^j+1}$ is a bijection and it is easy to invert. Clearly, after applying the affine bijections $\mathcal{S}$ and $\mathcal{T}$ to the input and to the output of $\mathcal{F}$, the system will not have a so-simple structure easy to invert. However, the function F presents a linear relation between its input and its output. Indeed, setting $y = F(x)$, we have the relation $y^{q^j} x = x^{q^{2j}} y$, in which both variables x and y appear only with "linear exponents". The affine transformations $\mathcal{S}$ and $\mathcal{T}$ preserve the presence of such linear relations between input and output, making systems susceptible to linearization attacks as first demonstrated in [Pat95]. Modifications to the MI cryptosystem have been attempted to mitigate this vulnerability, as seen in [CGP92, Din04]. However, even with these alterations, attacks have been developed that can reduce the modified MI system back to its original form, thus rendering it vulnerable to linearization attacks, as detailed in [DFSS07, FGS05]. This exemplifies just one instance of a property that persists through an affine equivalence relation. Cryptographers have identified many such properties, which have been exploited in successful attacks on multivariate schemes, including the Min Rank attack documented in [BFP13, CG21a, LC00, KS98, VST17].

In this paper, we propose to use a more general notion of equivalence relation between polynomial systems to obtain a public-key function $\mathcal{G}$ which does not inherit the "simple" structures of the secret function $\mathcal{F}$. Specifically, our investigation focuses on the CCZ equivalence transformation which has been introduced by Carlet, Charpin, and Zinoviev [CCZ98]. Given two functions $F, G : \mathbb{F}_q^n \to \mathbb{F}_q^m$ we say that they are *CCZ equivalent* if there exists an affine bijection $\mathcal{A}$ of $\mathbb{F}_q^{n+m}$ such that $\mathcal{G}_G = \mathcal{A}(\mathcal{G}_F)$, where $\mathcal{G}_F$ and $\mathcal{G}_G$ are the graphs of F and G respectively. This equivalence relation has been mainly studied in the context of cryptographic Boolean functions, since it keeps unchanged the value of the differential uniformity and the nonlinearity, two important properties to study when a function is used as a component of a block cipher. Budaghyan, Carlet, and Pott proved that CCZ equivalence is strictly more general than the affine equivalence by exhibiting functions which are CCZ equivalent to $F(x) = x^3$ over $\mathbb{F}_{2^n}$, but not affine equivalent, and also not extended affine (EA) equivalent [BCP06]. Other works focused on further studying the relation between CCZ and EA equivalences, see for example [CP19, BCV20].

We propose constructing multivariate schemes where the secret map and the public map are CCZ equivalent, without necessarily being affine or extended affine equivalent. However, selecting a random element G in the CCZ class of a given polynomial map F presents a challenge. Not every affine bijection of $\mathbb{F}_q^{n+m}$ maps the graph of one function into the graph of another, and the admissible affine bijections depend on the chosen function F. To address this challenge, we leverage a result by Canteaut and Perrin [CP19], demonstrating that any two CCZ-equivalent functions can be connected by applying two

extended affine transformations and another map called a t-twist (see Definition 2). In Section 4, we provide a high-level explanation of how this strategy can be utilized to construct an encryption or a signature scheme. This construction is quite general and applicable to any secret map F that admits a t-twist. To provide a concrete example, we propose selecting a quadratic function F derived from Oil and Vinegar (OV) polynomials [Pat95]. OV polynomials divide variables into oil and vinegar sets, with no quadratic terms in the oil variables, allowing for linear decryption/signing by assigning random values to the vinegar variables. The name comes from the fact that the variables do not truly mix, like oil and vinegar in the salad dressing. We name our scheme `Pesto`, as the CCZ transformation ensures that the variables fully mix, resembling the mixing of ingredients in Pesto Sauce using mortar and pestle. Notably, while the secret polynomials are quadratic, the public polynomials have degree four.

We provide commentary on the proposal, specifically regarding the dimensions of the keys and the computational costs involved, in Section 5. Discussion on potential vulnerabilities and attacks is reserved for the final section of the paper. Given that the public polynomials have degree four, many typical attacks against multivariate schemes, which target degree two, are not immediately applicable. However, partial recovery of the affine transformation may still be possible, albeit at significant computational expense.

Since our primary objective is to establish a connection between the research areas of Boolean functions and Multivariate Cryptography rather than presenting a fully-fledged new proposal, we have opted not to propose specific parameters for the system. We believe that there is considerable scope for further research and development in this area.

Structure of the paper

This work is organized as follows. In Section 2 we recall the definition and some basic facts about the CCZ equivalence. In Section 3, we present the *twisting* and explain how it can be used to produce a random CCZ transformation. In Section 4 we present a proposal for a multivariate cryptographic scheme obtained by hiding the central map with a CCZ transformation. We present it at a high level of generality in Subsection 4.1, and then with more specifics on the functions in Subsection 4.2, i.e., the `Pesto` scheme. Section 5 presents some comments on the form of the constructed maps, an analysis on the sizes of the keys and on the computational cost of applying the proposed procedure. In the last section, we present a preliminary analysis on the security of the scheme.

Acknowledgments

A. Caminata and I. Villa are supported by the Italian PRIN2022 grant P2022J4HRR "Mathematical Primitives for Post Quantum Digital Signatures", by the MUR Excellence Department Project awarded to Dipartimento di Matematica, Università di Genova, CUP D33C23001110001, and by the European Union within the program NextGenerationEU. Additionally, part of the work was done while Caminata was visiting the Institute of Mathematics of the University of Barcelona (IMUB). He gratefully appreciates their hospitality during his visit.

2 Preliminaries

In this section we recall the definition of CCZ equivalence, introduced in [CCZ98], together with some preliminary results and notations.

2.1 CCZ equivalence

Let n, m be positive integers, q a prime power and $\mathbb{F}_q$ a finite field with q elements. We consider a function $F : \mathbb{F}_q^n \to \mathbb{F}_q^m$. Notice that F can be seen as $F = (f_1, \ldots, f_m)$ with $f_i : \mathbb{F}_q^n \to \mathbb{F}_q$ for $1 \leq i \leq m$. We call *coordinate*, or *i-th coordinate*, of F the function f_i. For $\lambda = (\lambda_1, \ldots, \lambda_m) \in \mathbb{F}_q^m$ we call the λ-*component* of F the function $F_\lambda = \lambda \cdot F = \lambda_1 f_1 + \cdots + \lambda_m f_m$. To represent F we can use the *algebraic normal form* (ANF), that is, we can represent the function as a multivariate polynomial over $\mathbb{F}_q^m$:

$$F(x) = F(x_1, \ldots, x_n) = \sum_{u \in \mathbb{N}^n} a_u x_1^{u_1} \cdots x_n^{u_n}, \qquad \text{with } a_u \in \mathbb{F}_q^m.$$

Moreover, in order to have a unique representative for F we adopt the standard convention that $u_1, \ldots, u_n < q$. We say that $x_1^{u_1} \cdots x_n^{u_n}$ is a *term* of F if $a_u \neq 0$. The *algebraic degree* of F is $\deg(F) = \max\{\sum_{i=1}^n u_i : u \in \mathbb{N}^n;\, a_u \neq 0\}$. We call F *linear* if $\deg(F) = 1$ and $F(0) = 0$, *affine* if $\deg(F) \leq 1$, *quadratic* if $\deg(F) \leq 2$. When $m = n$, we say that $F : \mathbb{F}_q^n \to \mathbb{F}_q^n$ is a *bijection* or that it is *invertible* if F induces a permutation over $\mathbb{F}_q^n$, i.e., if $\{F(v) : v \in \mathbb{F}_q^n\} = \mathbb{F}_q^n$.

Definition 1. Let $F, G : \mathbb{F}_q^n \to \mathbb{F}_q^m$ be two functions.

- F is *affine equivalent* to G if there are two affine bijections A_1, A_2 of $\mathbb{F}_q^m$ and $\mathbb{F}_q^n$ respectively such that $G = A_1 \circ F \circ A_2$.
- F is *EA equivalent* (extended affine) to G if there are two affine bijections A_1, A_2 of $\mathbb{F}_q^m$ and $\mathbb{F}_q^n$ respectively and an affine transformation $A : \mathbb{F}_q^n \to \mathbb{F}_q^m$ such that $G = A_1 \circ F \circ A_2 + A$.
- F is *CCZ equivalent* to G if there exists an affine bijection $\mathcal{A}$ of $\mathbb{F}_q^{n+m}$ such that $\mathcal{G}_G = \mathcal{A}(\mathcal{G}_F)$, where $\mathcal{G}_F = \{(x, F(x)) : x \in \mathbb{F}_q^n\} \subseteq \mathbb{F}_q^n \times \mathbb{F}_q^m$ is the graph of F, and $\mathcal{G}_G$ is the graph of G.

Clearly, affine equivalence is a particular case of EA equivalence. Moreover, EA equivalence is a particular case of CCZ equivalence (see [CCZ98, Car20]). Notice that a CCZ transformation might change the algebraic degree of a function and also its bijectivity, whereas both notions are preserved by the affine equivalence and, when the function is not affine, the algebraic degree is also preserved by the EA equivalence.

In the usual set up of multivariate schemes, the central (secret) map F is composed with two randomly chosen affine bijections A_1, A_2 of $\mathbb{F}_q^m$ and $\mathbb{F}_q^n$ respectively to obtain the public map $G = A_1 \circ F \circ A_2$. Thus, the secret and public key F and G are affine equivalent. Modifying this construction by using EA equivalence does not provide any improvement. Indeed, the difference between EA and affine equivalence is just the addition of an affine transformation. So, most of the attacks that can be performed over schemes using an affine transformation can be easily extended to the case of EA transformation. Therefore, our investigation will focus on the CCZ transformation. Unfortunately, given a function F, it seems not so easy to obtain a random CZZ-equivalent function G as we are going to explain next.

2.2 Towards a random CCZ construction

Let F and G be two CCZ-equivalent functions as in Definition 1. We can write the affine bijection $\mathcal{A} : \mathbb{F}_q^n \times \mathbb{F}_q^m \to \mathbb{F}_q^n \times \mathbb{F}_q^m$ as

$$\mathcal{A}(x, y) = \mathcal{L}(x, y) + (a, b),$$

with $a \in \mathbb{F}_q^n$ and $b \in \mathbb{F}_q^m$ and $\mathcal{L} : \mathbb{F}_q^n \times \mathbb{F}_q^m \to \mathbb{F}_q^n \times \mathbb{F}_q^m$ linear bijection. Thus, $\mathcal{L}$ maps the graph of F to the graph of G', with $G'(x) = G(x+a) + b$. Hence, up to a translation of the input and the output, we can consider directly the linear bijection $\mathcal{L}$. Then, we can write $\mathcal{L}$ as a matrix composed by four linear maps,

$$\mathcal{L}(x,y) = \begin{bmatrix} A_1 & A_2 \\ A_3 & A_4 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} A_1(x) + A_2(y) \\ A_3(x) + A_4(y) \end{bmatrix} = \begin{bmatrix} L_1(x,y) \\ L_2(x,y) \end{bmatrix}.$$

Recall that $\mathcal{L}(x, F(x)) = (x', G(x'))$. Set

$$F_1(x) = L_1(x, F(x)) = A_1(x) + A_2(F(x))$$

and

$$F_2(x) = L_2(x, F(x)) = A_3(x) + A_4(F(x)).$$

Clearly, F_1 has to be a bijection and $G = F_2 \circ F_1^{-1}$. Notice that both F_1 and F_2 have degree at most the degree of F, while the bound on the degree of G depends also on the degree of the inverse of F_1.

With this notation, in order to generate a random function G in the CCZ class of F, we need to construct:

1. A random linear map $L_1 : \mathbb{F}_q^n \times \mathbb{F}_q^m \to \mathbb{F}_q^n$ such that $L_1(x, F(x))$ is a bijection;
2. A random linear map $L_2 : \mathbb{F}_q^n \times \mathbb{F}_q^m \to \mathbb{F}_q^m$ such that $\begin{bmatrix} L_1 \\ L_2 \end{bmatrix}$ is a bijection over $\mathbb{F}_q^{n+m}$.

Clearly, L_1 (and consequentially L_2) strongly depends on the choice of the initial function F. So, differently from the affine and the EA equivalence, it appears to be not easy to provide a general construction method for a random CCZ-equivalent function.

3 The twisting

To study a possible way to construct a random CCZ-equivalent map, we consider a particular instance of CCZ equivalence, introduced for the case $q = 2$ by Canteaut and Perrin with the name of twisting. Indeed, in [CP19, Theorem 3] the authors showed that any two CCZ-equivalent functions are connected via the following 3 steps: EA transformation, t-twisting, EA transformation.

3.1 Definition of twisting

We recall the definition of t-twisting from [CP19], generalized to any finite field $\mathbb{F}_q$.

Definition 2. For ℓ a positive integer, we denote with I_ℓ the $\ell \times \ell$ identity matrix. Given $0 \leq t \leq \min(n, m)$, we say that two functions $F, G : \mathbb{F}_q^n \to \mathbb{F}_q^m$ are *equivalent via t-twist (or t-twisting)* if $\mathcal{G}_G = M_t(\mathcal{G}_F)$, where M_t the $(n+m) \times (n+m)$ matrix

$$M_t = \begin{bmatrix} 0 & 0 & I_t & 0 \\ 0 & I_{n-t} & 0 & 0 \\ I_t & 0 & 0 & 0 \\ 0 & 0 & 0 & I_{m-t} \end{bmatrix}.$$

It holds $M_t = M_t^T = M_t^{-1}$, so this is an equivalence relation.

Assume that $F, G : \mathbb{F}_q^n \to \mathbb{F}_q^m$ are equivalent via t-twist. Then, we can split the input and the output of the function F in the first t entries and the remaining $n-t$ (resp. $m-t$) entries for input (resp. for output). That is, for $x \in \mathbb{F}_q^t, y \in \mathbb{F}_q^{n-t}$ we write

$$F(x,y) = (T(x,y), U(x,y)) = (T_y(x), U_x(y))$$

with $T : \mathbb{F}_q^t \times \mathbb{F}_q^{n-t} \to \mathbb{F}_q^t$ and $U : \mathbb{F}_q^t \times \mathbb{F}_q^{n-t} \to \mathbb{F}_q^{m-t}$. Then, we can write

$$M_t(\mathcal{G}_F) = M_t \cdot \begin{bmatrix} x \\ y \\ T(x,y) \\ U(x,y) \end{bmatrix} = \begin{bmatrix} T(x,y) \\ y \\ x \\ U(x,y) \end{bmatrix}.$$

To clarify the notation, here and in the rest of the paper we identify x with the t variables $x_1, \ldots, x_t$, and y with the $n-t$ variables $y_1, \ldots, y_{n-t}$. The following observation is important.

Remark 1. We have that $M_t(\mathcal{G}_F)$ is the graph of some function (we equivalently say that M_t *is admissible for* F) if $T(x,y) = T_y(x)$ is a bijection for every fixed $y \in \mathbb{F}_q^{n-t}$. Then, the CCZ-equivalent function G ($\mathcal{G}_G = M_t(\mathcal{G}_F)$) has the form $G(x,y) = (T_y^{-1}(x), U_{T_y^{-1}(x)}(y))$. Notice that, if the degree of $T_y^{-1}(x)$ is d (i.e. the algebraic degree with respect to both x and y), then the degree of $U(T_y^{-1}(x), y)$ is at most $2d$.

3.2 Twisting of quadratic functions

As mentioned in the introduction, the majority of multivariate cryptographic schemes appearing in the literature deals with quadratic functions. Therefore, we consider the case when the central map is a quadratic function and we present a preliminary study of the twisting for quadratic maps.

Let F be a quadratic function admitting a t-twist. We keep the notation introduced in §3.1 and we write $F(x,y) = (T(x,y), U(x,y))$. Since F is quadratic, both functions T and U have degree at most 2. Moreover, by Remark 1 we have that for every $y \in \mathbb{F}_q^{n-t}$, $T(x,y) = T_y(x)$ is invertible. Then, the bijection $T_y(x)$ is either affine or quadratic. We deal with both cases separately.

3.2.1 T affine

We assume that T is affine, that is $T(x,y) = \ell(x) + \phi(y)$ with $\ell : \mathbb{F}_q^t \to \mathbb{F}_q^t$ linear bijection and $\phi : \mathbb{F}_q^{n-t} \to \mathbb{F}_q^t$ affine transformation. In this case, the map is easily invertible as $T_y^{-1}(x) = \ell^{-1}(x) - \ell^{-1}(\phi(y))$. However this map will produce a function G that is EA-equivalent to F. Indeed, $G = F \circ A$, with A the affine bijection of the form $A(x,y) = (\ell^{-1}(x) - \ell^{-1}(\phi(y)), y)$. So, we are not interested in this case.

3.2.2 T quadratic

The general case $T_y(x)$ quadratic is quite difficult to analyse. By Remark 1, to apply a CCZ transformation, we need the map $T(x,y) = T_y(x)$ to be invertible for every fixed y. A possible way to achieve this is to consider $T(x,y)$ as a function of the form

$$T(x,y) = \ell(x) + \mathfrak{q}(y), \tag{1}$$

with $\ell : \mathbb{F}_q^t \to \mathbb{F}_q^t$ a linear bijection and $\mathfrak{q} : \mathbb{F}_q^{n-t} \to \mathbb{F}_q^t$ a quadratic function. In this case, the inverse of $T_y(x)$ has the form $T_y^{-1}(x) = \ell^{-1}(x) - \ell^{-1}(\mathfrak{q}(y))$. Indeed, we have

$$T(\ell^{-1}(x) - \ell^{-1}(\mathfrak{q}(y)), y) = \ell(\ell^{-1}(x) - \ell^{-1}(\mathfrak{q}(y))) + \mathfrak{q}(y) = x.$$

The map $T_y^{-1}(x)$ has degree at most two. Thus, Remark 1 implies that the map G constructed with the t-twist has degree at most four.

4 Multivariate CCZ Scheme

We propose a scheme where the central map is hidden by an application of a CCZ transformation. We present two versions of this proposal at increasing level of details. First, we present a generic instance of the scheme which can be applied to any central map admitting a t-twist (§4.1). Then, we present some restrictions on the choice of the quadratic secret map used (§4.2.1 and §4.2.2), summarized in a concrete instance with a specific choice for the central map (§4.2.3).

4.1 Generic CCZ Scheme

As "central map" we consider $F : \mathbb{F}_q^n \to \mathbb{F}_q^m$ a function admitting a t-twist for an integer $1 \leq t \leq \min(n, m)$. We remove the value $t = 0$ since no modification is obtained in this way. With the usual notation, we write F as

$$F(x, y) = (T(x, y), U(x, y))$$

with $x \in \mathbb{F}_q^t$, $y \in \mathbb{F}_q^{n-t}$, $T : \mathbb{F}_q^t \times \mathbb{F}_q^{n-t} \to \mathbb{F}_q^t$ such that $T(x, y) = T_y(x)$ is invertible in x for every possible y, and $U : \mathbb{F}_q^t \times \mathbb{F}_q^{n-t} \to \mathbb{F}_q^{m-t}$. The equivalent function G has then the form

$$G(x, y) = (T_y^{-1}(x), U(T_y^{-1}(x), y)).$$

Finally, we construct the public map as $G_{pub} = A_1 \circ G \circ A_2$, for A_1, A_2 random affine bijections of $\mathbb{F}_q^m$ and $\mathbb{F}_q^n$ respectively. In the secret key, we need to store the information needed to invert the map G_{pub}, consisting of $\langle A_1, A_2, t, T, U\rangle$. Equivalently, we can directly consider $\langle A_1^{-1}, A_2^{-1}, t, T, U\rangle$.

In the following, we describe in more details the steps to use this pair of public and secret keys for an encryption scheme and for a signature scheme.

4.1.1 Proposal as an encryption scheme

In an encryption scheme, a sender encrypts a message by using the public key. The receiver recovers the original message by knowing the secret key.

Encryption. The sender encrypts the message $m \in \mathbb{F}_q^n$ by evaluating the public key G_{pub} and sends $c = G_{pub}(m) \in \mathbb{F}_q^m$.

Decryption. From $c \in \mathbb{F}_q^m$, the receiver has to compute its preimages, i.e. the set of solutions $\bar{m}$ such that $G_{pub}(\bar{m}) = c$. This can be obtained in the following way.

1. Given c, compute $c' = A_1^{-1}(c)$. To simplify the notation, for $c = G_{pub}(m)$, call $m' = A_2(m)$, so $c' = G(m')$. Moreover, write c' as $(c'_T, c'_U) \in \mathbb{F}_q^t \times \mathbb{F}_q^{m-t}$ and similarly consider $m' = (x, y) \in \mathbb{F}_q^t \times \mathbb{F}_q^{n-t}$. So it holds

$$\begin{aligned} c'_T &= T_y^{-1}(x), \\ c'_U &= U(T_y^{-1}(x), y), \end{aligned}$$

   implying $c'_U = U(c'_T, y)$.

2. From $c'_U = U(c'_T, y)$, find the set of possible solution $\mathcal{Y} = \{y \in \mathbb{F}_q^{n-t} : c'_U = U(c'_T, y)\}$.

3. For $\bar{y} \in \mathcal{Y}$, compute $\bar{x} = T_{\bar{y}}(c'_T)$.

4. For every possible pair of solutions $(\bar{x}, \bar{y})$, compute $\bar{m} = A_2^{-1}(\bar{x}, \bar{y})$.

4.1.2 Proposal as a signature scheme

In a signature scheme, a sender produces a signature for a document, knowing the secret information. The receiver checks the validity of the signature for the received document. Consider a hash function $\mathcal{H} : \{0,1\}^* \to \mathbb{F}_q^m$.

Signature. Given a document $d \in \{0,1\}^*$, the sender wants to create a valid signature, knowing the secret information.

1. Compute the hash value $w = \mathcal{H}(d)$. Compute $w' = A_1^{-1}(w)$ and write it as $w' = (w_T, w_U) \in \mathbb{F}_q^t \times \mathbb{F}_q^{m-t}$.
2. We need to find $x \in \mathbb{F}_q^t$ and $y \in \mathbb{F}_q^{n-t}$ such that $G(x,y) = w$. So $T_y^{-1}(x) = w_T$ and $U(T_y^{-1}(x), y) = w_U$.
3. Solve $U(w_T, y) = w_U$, pick randomly one of the solutions and call it $\bar{y}$.
4. Solve $T_{\bar{y}}^{-1}(x) = w_T$, that is compute $\bar{x} = T_{\bar{y}}(w_T)$.
5. Given the solution $(\bar{x}, \bar{y})$, compute $(x', y') = A_2^{-1}(\bar{x}, \bar{y})$.
6. Output (x', y') as signature.

Verification. The receiver wants to verity that the signature (x', y') is valid for the document d.

1. Compute $w = \mathcal{H}(d)$.
2. Check that $G_{pub}(x', y') = w$.

Correctness of the procedures For the signature scheme, the correctness is verified by the following relation,

$$\begin{aligned} G_{pub}(x', y') =& A_1 \circ G \circ A_2(x', y') = A_1 \circ G(\bar{x}, \bar{y}) \\ =& A_1 \circ (T_{\bar{y}}^{-1}(\bar{x}), U(T_{\bar{y}}^{-1}(\bar{x}), y)) = A_1(w_T, w_U) = w. \end{aligned}$$

The correctness for the encryption scheme $G_{pub}(\bar{m}) = c$ is verified in the same way.

Conditions for signature scheme and encryption scheme For both encryption scheme and signature scheme, we need $T(x,y) = T_y(x)$ to be an invertible function with respect to x, for every possible value of y.

Regarding the function $U(x,y)$, different conditions must be satisfied. If we want to use F as a central map for a signature scheme, we need that for any possible $(a,b) \in \mathbb{F}_q^t \times \mathbb{F}_q^{m-t}$, the system $U(a,y) = b$ has always a solution. Instead, for the encryption scheme, we need the system to have few solutions. Indeed, in this case, the receiver has to consider all possible pre-images of the map G_{pub} and find the correct one.

4.2 Our proposal

Assume that $F : \mathbb{F}_q^n \to \mathbb{F}_q^m$ is a quadratic function which admits a t-twist, so $F(x,y) = (T(x,y), U(x,y))$ where both T and U have at most degree 2. In what follows we present our proposal for the choice of the maps T an U.

4.2.1 The choice of T

Given the analysis presented in §3.2, we choose the map T as in Equation (1), which allows us to easily invert $T_y(x)$. Therefore, we consider $T(x,y) = \ell(x) + \mathfrak{q}(y)$, where $\ell : \mathbb{F}_q^t \to \mathbb{F}_q^t$ is a linear invertible tranformation and $\mathfrak{q} : \mathbb{F}_q^{n-t} \to \mathbb{F}_q^t$ is a quadratic map. We show in the following that we do not loose generality by restricting to $\ell(x) = x$ the identity map.

Remark 2. The t-twisted map G and the public map G_{pub} are

$$G(x,y) = (T_y^{-1}(x), U(T_y^{-1}(x), y)) = (\ell^{-1}(x) - \ell^{-1}(\mathfrak{q}(y)), U(\ell^{-1}(x) - \ell^{-1}(\mathfrak{q}(y)), y))$$

and $G_{pub} = A_1 \circ G \circ A_2$, with A_1, A_2 affine bijections of $\mathbb{F}_q^m$ and $\mathbb{F}_q^n$ respectively. We can always write the affine bijection A_2 as $A_2 = L \circ A_2'$ with $L(x,y) = (\ell(x), y)$ and $A_2' = L^{-1} \circ A_2$. Then, we set $G' = G \circ L$, so we have $G_{pub} = A_1 \circ G' \circ A_2'$ with

$$G'(x,y) = G \circ L(x,y) = (x - \ell^{-1}(\mathfrak{q}(y)), U(x - \ell^{-1}(\mathfrak{q}(y)), y)).$$

This means that, since A_2 is chosen at random, we can assume without loss of generality that $\ell(x) = x$.

Given the previous consideration, our choice is

$$T(x,y) = x + \mathfrak{q}(y),$$

leading to a t-twist of the form $G(x,y) = (x - \mathfrak{q}(y), U(x - \mathfrak{q}(y), y))$. Observe that with this choice of T, to compute $\bar{x} = T_{\bar{y}}(c_T)$ simply corresponds to computing $\bar{x} = c_T + \mathfrak{q}(\bar{y})$.

4.2.2 The choice of U

Now, we propose a possible choice for the quadratic map U. We recall that we need $U(x,y)$ to be such that, fixed x, it is easy to get the preimages (or a preimage) with respect to y. A possible way to achieve this is to use Oil and Vinegar (OV) maps. Therefore, chosen a parameter s with $0 \leq s \leq n-t$, we propose to construct the map U as a system of OV equations with $t+s$ vinegar variables and $n-t-s$ oil variables. To be more specific, in this proposal U consists of a system of $m-t$ equations of the form

$$f^{(i)} = \sum_{j,k \in V} \alpha_{jk}^{(i)} z_j z_k + \sum_{j \in V, k \in O} \beta_{jk}^{(i)} z_j z_k + \sum_{j \in V \cup O} \gamma_j^{(i)} z_j + \delta^{(i)} \tag{2}$$

with $\{z_j : j \in V\} = \{x_1, \ldots, x_t, y_1, \ldots, y_s\}$ and $\{z_j : j \in O\} = \{y_{s+1}, \ldots, y_{n-t}\}$ respectively the sets of vinegar and oil variables, and with the coefficients $\alpha_{jk}^{(i)}, \beta_{jk}^{(i)}, \gamma_j^{(i)}, \delta^{(i)}$ randomly chosen over $\mathbb{F}_q$. Notice that, fixed the vinegar variables, the system is linear in the oil variables, hence it is easy to solve, for example with a simple Gaussian reduction. The legitimate user can get the preimages of U with respect to y (fixed x) using classical techniques from OV systems [DY09, DPS20]. Notice that the easiest system is obtained for $s = 0$, however from the analysis presented in §6, this is not a good choice.

4.2.3 `Pesto`

We sum up all the previous choices in the following definition: the scheme `Pesto`.

Definition 3 (`Pesto` primitive)**.** Fix positive integer parameters n, m, t, s with $t \leq \min(n,m)$ and $s \leq n-t$, consider the following maps:

- $\mathfrak{q} : \mathbb{F}_q^{n-t} \to \mathbb{F}_q^t$ random quadratic map (so $T(x,y) = x + \mathfrak{q}(y)$);
- $U : \mathbb{F}_q^t \times \mathbb{F}_q^{n-t} \to \mathbb{F}_q^{m-t}$ a system of $m-t$ random OV maps with $x_1, \ldots, x_t, y_1, \ldots, y_s$ vinegar variables and $y_{s+1}, \ldots, y_{n-t}$ oil variables as in Equation (2);

- $A_1 : \mathbb{F}_q^m \to \mathbb{F}_q^m$ a random affine bijection;
- $A_2 : \mathbb{F}_q^n \to \mathbb{F}_q^n$ a random affine bijection.

Set $G = (x - \mathfrak{q}(y), U(x - \mathfrak{q}(y), y))$. Then the map $G_{pub} = A_1 \circ G \circ A_2$ is the public key, and $\langle A_1, A_2, \mathfrak{q}, U \rangle$ (and eventually t) constitutes the secret key.

We do not suggest specific values for the size of the parameters. However, in the following remark we stress the role of s.

Remark 3. The amount of possible signatures for a document, or the amount of possible plaintexts for a given ciphertext, depends on the value of s.

- For an encryption scheme, we need to find all possible solutions of $c'_U = U(c'_T, y)$. Here we need to try all possible values for $y_1, \ldots, y_s$ (that is, q^s possibilities) and then solve a linear system of $m - t$ equations in $n - t - s$ variables.
- For a signature scheme, we need to find only one solution of $w'_U = U(w'_T, y)$. Hence, we pick random values for $y_1, \ldots, y_s$ and then we solve a linear system of $m - t$ equations in $n - t - s$ variables. If the system does not have a solution, we pick other random values for $y_1, \ldots, y_s$.

Hence, when choosing the parameters n, m, t, s one has also to consider if the linear system to solve, $m - t$ equations in $n - t - s$ variables, should be:

- determined with high probability ($m - t = n - t - s$, that is $s = n - m$);
- overdetermined ($m - t > n - t - s$, that is $s > n - m$);
- underdetermined ($m - t < n - t - s$, that is $s < n - m$).

Finally, based on the security analysis performed in §6, we also recommend that t is around $n/3$.

4.2.4 Toy Example

We provide here a toy example of `Pesto` over $\mathbb{F}_5$. We take $n = 5$, $m = 4$, $t = 2$, and $s = 1$. Therefore we have the following set of variables: $x = \{x_1, x_2\}$ and $y = \{y_1, y_2, y_3\}$. To define the map $T(x, y) : \mathbb{F}_5^2 \times \mathbb{F}_5^4 \to \mathbb{F}_5^2$, we need to define a quadratic map $\mathfrak{q} : \mathbb{F}_5^4 \to \mathbb{F}_5^2$. Set

$$\mathfrak{q}(y) = \begin{bmatrix} y_1^2 + 2y_1y_2 + 4y_2^2 + 3y_2y_3 + y_2 + 3y_3^2 + 4 \\ \\ 3y_1^2 + 3y_1y_2 + 2y_1y_3 + 2y_2y_3 + 2y_2 + y_3^2 + 2y_3 \end{bmatrix},$$

therefore $T(x, y) = x + \mathfrak{q}(y)$. The map U is an OV system of 2 equations with x_1, x_2, y_1 oil variables and y_2, y_3 vinegar variables. Hence we consider

$$U(x, y) = \begin{bmatrix} x_1^2 + 2x_1x_2 + 3x_1y_1 + x_1y_2 + 4x_1y_3 + 2x_1 + x_2^2 + x_2y_1 + x_2y_2 \\ \quad +3x_2y_3 + 3x_2 + 2y_1^2 + 2y_1y_2 + 2y_1y_3 + y_1 + 4y_2 + y_3 \\ \\ x_1x_2 + x_1 + 4x_2^2 + 2x_2y_2 + 3x_2y_3 + 3x_2 + 2y_1y_3 + 3y_1 + 3y_3 + 1 \end{bmatrix}.$$

Having constructed these maps, we have that $G(x,y)$ has the following form

$$\begin{bmatrix} x_1 + 4y_1^2 + 3y_1y_2 + y_2^2 + 2y_2y_3 + 4y_2 + 2y_3^2 + 1 \\ \\ x_2 + 2y_1^2 + 2y_1y_2 + 3y_1y_3 + 3y_2y_3 + 3y_2 + 4y_3^2 + 3y_3 \\ \\ \begin{array}{c} x_1^2 + 2x_1x_2 + 2x_1y_1^2 + x_1y_1y_3 + 3x_1y_1 + 2x_1y_2^2 + 2x_1y_3^2 + 4x_1 + x_2^2 + 2x_2y_1^2 + x_2y_1y_3 \\ +x_2y_1 + 2x_2y_2^2 + 2x_2y_3^2 + 4x_2y_3 + y_1^4 + y_1^3y_3 + 4y_1^3 + 2y_1^2y_2^2 + y_1^2y_2 + y_1^2y_3^2 + y_1^2y_3 + 3y_1^2 \\ +y_1y_2^2y_3 + 3y_1y_2^2 + 2y_1y_2y_3 + 4y_1y_2 + y_1y_3^3 + 2y_1y_3^2 + 4y_1 + y_2^4 + 2y_2^2y_3^2 + 2y_2y_3^2 \\ +3y_2y_3 + y_2 + y_3^4 + y_3^3 + y_3^2 + 3 \end{array} \\ \\ \begin{array}{c} x_1x_2 + 2x_1y_1^2 + 2x_1y_1y_2 + 3x_1y_1y_3 + 3x_1y_2y_3 + 3x_1y_2 + 4x_1y_3^2 + 3x_1y_3 + x_1 + 4x_2^2 + 4x_2y_1y_2 \\ +4x_2y_1y_3 + x_2y_2^2 + x_2y_2y_3 + 4x_2y_3^2 + 2x_2y_3 + 4x_2 + 4y_1^4 + y_1^3y_2 + 4y_1^2y_2^2 + y_1^2y_2y_3 + 2y_1^2y_2 + \\ y_1^2y_3 + 2y_1^2 + 2y_1y_2^3 + 4y_1y_2^2y_3 + 4y_1y_2^2 + 3y_1y_2y_3^2 + 3y_1y_2y_3 + y_1y_2 + 2y_1y_3^3 + y_1y_3^2 \\ +4y_1y_3 + 3y_1 + 3y_2^3y_3 + 3y_2^3 + y_2^2y_3^2 + 4y_2^2y_3 + 3y_2y_3^2 + 3y_2y_3 + y_2 + 2y_3^4 + 4y_3^3 + 3y_3^2 + 2 \end{array} \end{bmatrix}.$$

Finally, we consider the following affine bijections of $\mathbb{F}_5^5$ and $\mathbb{F}_5^4$

$$A_2(x,y) = \begin{bmatrix} 1 & 4 & 3 & 2 & 1 \\ 2 & 0 & 1 & 1 & 4 \\ 3 & 2 & 2 & 0 & 2 \\ 1 & 2 & 2 & 2 & 3 \\ 2 & 3 & 4 & 4 & 2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ y_1 \\ y_2 \\ y_3 \end{bmatrix} + \begin{bmatrix} 2 \\ 1 \\ 3 \\ 2 \\ 2 \end{bmatrix}, \qquad A_1(z) = \begin{bmatrix} 2 & 3 & 2 & 1 \\ 4 & 2 & 3 & 1 \\ 1 & 2 & 1 & 3 \\ 1 & 4 & 3 & 1 \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \\ z_3 \\ z_4 \end{bmatrix} + \begin{bmatrix} 1 \\ 0 \\ 0 \\ 4 \end{bmatrix}.$$

All the mentioned functions, namely $\mathfrak{q}, U, A_1, A_2$, were randomly generated with the help of the MAGMA software [BCP97]. The public map $G_{pub} = A_1 \circ G \circ A_2$ consists of 4 dense polynomials of degree 4. Given their sizes, we opt for not reporting them here.

5 Computational remarks

From now on, we focus on our proposal `Pesto` of Definition 3. We study the form of the secret and public key, their size, and the cost of the procedure.

5.1 The form of G and G_{pub}

Since the map $T(x,y) = x + \mathfrak{q}(y)$ and its inverse $T_y^{-1}(x) = x - \mathfrak{q}(y)$ have degree 2, by Remark 1 the degree of the t-twisted map G is at most 4. We analyze the monomials appearing in G (and in G_{pub}) more closely.

Remark 4. First, we notice that the terms appearing in $T(x,y)$ are of the form x_i, y_i, y_iy_j. Clearly, the same terms will appear in $T_y^{-1}(x)$, and so the degrees of the first t coordinates of F are the same of the degrees of the first t coordinates of G. Now, we consider the last $m-t$ coordinates of G, corresponding to $U(T_y^{-1}(x),y) = U(x - \mathfrak{q}(y), y)$. Since U consists of OV quadratic polynomials $f^{(i)}$ of the form of Equation (2), in $U(x,y)$ we can find terms of the form $x_i, y_i, x_ix_j, x_iy_j, y_iy_j$. By evaluating $x \mapsto x - \mathfrak{q}(y)$, the variable x_i might produce terms of the form x_j, y_j and y_jy_k. Therefore, potentially, in the last $m-t$ coordinates of G we can have terms of the form x_i, y_i, x_ix_j, x_iy_j, y_iy_j, $x_iy_jy_k$, $y_iy_jy_k$, $y_iy_jy_ky_l$. Notice that, even if we replace U with a dense quadratic map, the possible terms in G have the same form. To sum up, even if the degree of G is up to 4, the terms of degree 4 involve only variables in y, while terms of degree 3 involve only variables in y or 2 variables in y and 1 variable in x.

The public map $G_{pub} = A_1 \circ G \circ A_2$ consists of dense polynomials of degree up to 4, since the random map A_2 will remove the above-mentioned restrictions. However, the

affine transformation keeps invariant the amount of components of a fixed degree, so to analyse the possible degrees of the components of G_{pub}, we can directly consider the map $G(x, y) = (x - \mathfrak{q}(y), U(x - \mathfrak{q}(y), y))$. So, G_{pub} has at least $q^t - 1$ quadratic components, since the components $\lambda \cdot G$ with $\lambda = (v, 0_{m-t})$ with nonzero $v \in \mathbb{F}_q^t$ are quadratic.

We present in Table 1 some computational experiments on the possible degree of the system $U(x - \mathfrak{q}(y), y)$. We observed that all the constructed systems $U(x - \mathfrak{q}(y), y)$ do not have quadratic components.

Table 1: For each choice of parameters, number of components $\mathtt{d}_3$ of degree 3 and $\mathtt{d}_4$ of degree 4 for 50 randomly generates systems $U(x - \mathfrak{q}(y), y)$

q	n	m	t	s	# of systems	$[\mathtt{d}_3, \mathtt{d}_4]$
5	5	4	2	1	4	$[4, 20]$
					46	$[0, 24]$
5	6	5	2	2	16	$[3, 120]$
					34	$[0, 124]$
5	10	8	3	2	1	$[4, 3120]$
					49	$[0, 3124]$
2^6	5	4	2	1	50	$[0, 4095]$
2^6	6	5	2	2	1	$[36, 262080]$
					49	$[0, 262143]$

5.2 Dimensions of the keys

In the following, we report an analysis of the sizes of the public and private keys.

Proposition 1. *Consider a* `Pesto` *scheme as proposed in Definition 3. Then the public key consists of* $m \cdot \binom{n+4}{4}$ *coefficients over* $\mathbb{F}_q$*, and the secret key consists of* $m^2 + m + n^2 + n + t\binom{n-t+2}{2} + (m-t)\binom{t+s+2}{2} + (m-t)(n-t-s)(t+s+1)$ *coefficients over* $\mathbb{F}_q$*.*

Proof. Recall that a polynomial of degree r in n variables with coefficients over $\mathbb{F}_q$ has $M_n(r) = \binom{n+r}{r}$ terms. Notice that when $q \leq r$, we can use the field equations $x_i^q - x_i$ to lower the degrees of the polynomials over $\mathbb{F}_q$ and store less coefficients. G_{pub} is a system of m equations of degree at most 4 in n variables, where each equation has $M_n(4) = \binom{n+4}{4}$ possible terms. Hence, for G_{pub} we need to specify $m \cdot M_n(4)$ coefficients over $\mathbb{F}_q$.

Regarding the secret key, we want to store the information to recover $A_1, A_2, \mathfrak{q}, U$.

- A_1 and A_2 are an affine bijections over $\mathbb{F}_q^m$ and $\mathbb{F}_q^n$ respectively: They correspond to invertible $m \times m$ and $n \times n$ matrices over $\mathbb{F}_q$ plus a constant vector. So this means $m^2 + m + n^2 + n$ coefficients over $\mathbb{F}_q$.

- $\mathfrak{q} : \mathbb{F}_q^{n-t} \to \mathbb{F}_q^t$ quadratic: It corresponds to t quadratic equations in $n - t$ variables. So, we need to store $t \cdot M_{n-t}(2)$ coefficients over $\mathbb{F}_q$.

- $U : \mathbb{F}_q^n \to \mathbb{F}_q^{m-t}$ a quadratic OV system of $m-t$ equations with $t+s$ vinegar variables and $n - t - s$ oil variables. We can see each equation as a quadratic map in $t + s$ variables, $M_{t+s}(2)$ terms, plus a map with oil variables multiplied by a vinegar variable or multiplied by 1, $(n - t - s)(t + s + 1)$ terms. Thus, we need to store $(m - t) \cdot \big(M_{t+s}(2) + (n - t - s)(t + s + 1)\big)$ coefficients over $\mathbb{F}_q$.

□

To give a more precise idea to the reader, in Table 2 we report the amounts for public and secret keys for some small parameters. The sizes are in terms of the amount of coefficients in $\mathbb{F}_q$ to store.

Table 2: Amount of coefficients of $\mathbb{F}_q$ to store

n	m	t	s	amount for pk	amount for sk
5	4	2	1	504	106
6	5	2	2	1050	177
10	8	3	2	8008	545

5.3 Computational cost of the procedure

In this subsection, we present a tentative analysis on the computational cost of the proposed multivariate scheme. Notice that the application of this procedure reduces to the evaluation of polynomials of degree at most four and computing the solution of linear systems. Considering the direct evaluation of a polynomial, we have the following estimates.

- To evaluate affine polynomials we perform $\mathfrak{m}_1(n) = 2\binom{n+1}{1} - n - 2 = n$ multiplications and $\binom{n+1}{1} - 1 = n$ additions.
- To evaluate quadratic polynomials we perform $\mathfrak{m}_2(n) = 2\binom{n+2}{2} - n - 2 = n(n+2)$ multiplications and $\binom{n+2}{2} - 1$ additions.
- To evaluate quartic polynomials we perform $\mathfrak{m}_4(n) = 2\binom{n+4}{4} - n - 2$ multiplications and $\binom{n+4}{4} - 1$ additions.

This is clearly an upper bound, since more efficient techniques might be used, see for example [BES13]. We indicate with $\mathfrak{M}(r, n)$ the number of multiplications needed to solve a linear system of r equations in n variables. Recall that $\mathfrak{M}(r, n)$ is of the order $rn \cdot \min(r, n)$, see for example [BV18, Appendix B].

In the following, we present an estimate on the cost of applying the proposed procedure in terms of multiplications, since to multiply is more expensive than to add.

Proposition 2. *Consider a signature scheme based on the* `Pesto` *primitive as in Definition 3. Set* $\mathfrak{m}_i(r)$ *and* $\mathfrak{M}(r, k)$ *as defined before. Excluding the computation of the hash value, the computational cost, in terms of multiplications over* $\mathbb{F}_q$*, to verify the validity of a signature is* $m \cdot \mathfrak{m}_4(n)$*, whereas the computational cost to produce a valid signature is* $m \cdot \mathfrak{m}_1(m) + t \cdot \mathfrak{m}_2(n-t) + n \cdot \mathfrak{m}_1(n) + (m-t)\left(\mathfrak{m}_2(t) + \mathfrak{m}_2(s)\right) + \mathfrak{M}(m-t, n-t-s)$.

Proof. Assume that the scheme was already initialized, a secret key and the corresponding public key were already constructed. To verify the validity of a signature $\mathfrak{s} \in \mathbb{F}_q^n$ we need to evaluate $G_{pub}(\mathfrak{s})$. G_{pub} is a system of m equations in n variables of degree at most 4. This corresponds to a total of $m \cdot \mathfrak{m}_4(n) = m\left(2\binom{n+4}{4} - n - 2\right)$ multiplications.
To create a valid signature for $w \in \mathbb{F}_q^m$, we have to perform the following.

1. Compute $w' = A_1^{-1}(w)$. Since A_1^{-1} is an affine bijection, this means $m \cdot \mathfrak{m}_1(m) = m^2$ multiplications.
2. Evaluate $U(w'_T, y)$. This is an evaluation of quadratic polynomials in t variables and corresponds to $(m-t) \cdot \mathfrak{m}_2(t) = (m-t)t(t+2)$ multiplications.
3. Find a preimage of $U(w'_T, y) = w'_U$.

4. For one preimage $\bar{y}$ found, evaluate $\bar{x} = w'_T + \mathfrak{q}(\bar{y})$. Since $\mathfrak{q}$ is a system of t quadratic equations in $n-t$ variables, this means $t \cdot \mathfrak{m}_2(n-t) = t \cdot (n-t)(n-t+2)$ multiplications.

5. From the solution pair $(\bar{x}, \bar{y})$, compute $A_2^{-1}(\bar{x}, \bar{y})$. This corresponds to $n \cdot \mathfrak{m}_1(n) = n^2$ multiplications.

We are left with analysing the cost for the third step, where we have to find a preimage of $U(w'_T, y) = w'_U$. Hence we need to set $y_1, \ldots, y_s$ to random values, evaluate those variables and then solve a linear system. The cost of evaluating s variables corresponds to $(m-t) \cdot \mathfrak{m}_2(s) = (m-t)s(s+2)$ multiplications. The cost of solving the final system of $m-t$ equations in $n-t-s$ variables is $\mathfrak{M}(m-t, n-t-s)$ multiplications. □

Remark 5. If we consider an encryption scheme, the cost of encrypting a message and decrypting a valid ciphertext can be deduced from the analysis just reported. The fundamental difference is that we will need to compute the entire set of preimages of $U(c'_T, y) = c'_U$. This means that, in the second step, we need to evaluate $y_1, \ldots, y_s$ in every possible value. Hence, we need to compute q^s evaluations and then solve q^s linear systems.

6 Security Analysis

In this section, we provide some considerations on the security of the scheme `Pesto` of Definition 3. Among them, we consider attacks that have been exploited for the MI cryptosystem (and its generalizations) and we analyze under which conditions it would be possible to extend these also to our system.

6.1 The importance of A_2

We start this analysis by presenting a first observation on the security of the scheme, which stresses the importance of the affine bijection A_2. Indeed, if an attacker is able to recover A_2, then they can operate as follows. First, compute $\bar{G} = G_{pub} \circ A_2^{-1}$ and isolate its quadratic components: t of them are of the form $x_i - \mathfrak{q}_i(y)$, for $i = 1, \ldots, t$. From this, it is rather easy to solve the system. Suppose a cyphertext c is given, and the attacker wants to find a message m such that $G_{pub}(m) = c$. Write $G_{pub}(m) = \bar{G} \circ A_2(m) = c$. From the isolated quadratic equations, recover $x = \mathfrak{q}(y) + c_T$. By substituting this into the remaining equations of $\bar{G}$, solve the system, which has at most s variables appearing in quadratic terms. Notice that this last step is equivalent to what a legitimate user has to do. Once the solution (x, y) of this system is found, then $m = A_2^{-1}(x, y)$.

In order to perform this attack, the attacker has to completely recover A_2 but only partially A_1 (it is enough to isolate the quadratic components of the form $x_i - \mathfrak{q}_i(y)$). The cost of isolate these quadratic components will be treated in the following section.

6.2 A partial recover of A_1 isolating the quadratic components

The goal of this attack is to isolate in G_{pub} the components which correspond to (a linear combination of) the first t coordinates of G. Notice that these components form an OV system which can be then attacked using other known techniques, later analyzed in §6.5.

We try to estimate the cost of isolating the quadratic components of G_{pub}. Let $r \geq t$ be such that the number of quadratic components of G_{pub} is $q^r - 1$. Notice that when $r = t$, then $U(x - \mathfrak{q}(y), y)$ does not have quadratic components. We consider the list Δ of all possible terms of degree 3 and degree 4 in n variables appearing in G_{pub}. We have

$$|\Delta| = \binom{n+4}{4} - \binom{n+2}{2} = (n+2)(n+1)\frac{n^2+7n}{24}.$$

We construct the matrix A with n rows and $|\Delta|$ columns, with at position (i,j) the coefficients of the j-th term of Δ in the i-th equation of G_{pub}. So, to recover the quadratic equations of G_{pub} it is enough to solve the linear system $x^T A = A^T x = 0_{|\Delta|}$. The set of solutions corresponds to the set of quadratic components of G_{pub}. The size of this set is q^r. Recall that the cost of solving a linear system with n equations in $|\Delta|$ variables $\mathfrak{M}(n,|\Delta|)$ is of the order $O(n^6)$. Let us note that in this way we can recover the space of the quadratic components of G_{pub}, but not the exact quadratic coordinates of G_{pub}.

6.3 Differential attack: a partial recover of A_2 using the linear structures

The core idea of this analysis is the following.

1. G_{pub} has (at least) $q^t - 1$ quadratic components with (at least) q^t linear structures in common (see Definition 4 below).
2. If we are able to isolate these components in G_{pub} (corresponding to the mentioned components of G), we take t of them (linearly independent) and then we determine the intersection of the linear structures on these t components of G_{pub}, which contains a t-dimensional vector space V.
3. If we are able to identify this subspace, we know that the linear part of A_2 maps V into $\mathbb{F}_q^t$. So, we can partially recover this linear part.

We present now in more detail the analysis roughly explained above step by step. We start by recalling the definition and some useful properties of linear structures.

Definition 4. Let $f : \mathbb{F}_q^n \to \mathbb{F}_q$ be a function. We say that $a \in \mathbb{F}_q^n$ is a *linear structure* of f if the derivative $D_a f(x) := f(x+a) - f(x)$ is constant.

Lemma 1. *The set of linear structures of a function f forms a vector subspace of $\mathbb{F}_q^n$.*

Proof. Indeed, $D_{a+b}f(x) = f(x+a+b) - f(x) = f(x+a+b) - f(x+a) + f(x+a) - f(x) = D_b f(x+a) + D_a f(x)$. Instead, if a is a linear structure of f, to show that τa is also a linear structure, for any $\tau \in \mathbb{F}_q$, we proceed as follows. Consider L a linear bijection of $\mathbb{F}_q^n$ such that $L(e_1) = a$, where e_1 is the first element of the canonical basis of $\mathbb{F}_q^n$. Then, for $g = f \circ L$, e_1 is a linear structure of g since $D_b g(x) = g(x+b) - g(x) = f(L(x) + L(b)) - f(L(x)) = D_{L(b)} f(L(x))$, and for $b = e_1$ the derivative is constant. Since we represent functions with polynomials of degrees $< q$ in each variable, this implies that g is of the form $g(x_1, \dots, x_n) = \alpha x_1 + h(x_2, \dots, x_n)$, for $\alpha \in \mathbb{F}_q$ and $h : \mathbb{F}_q^{n-1} \to \mathbb{F}_q$. Therefore for any $\tau \in \mathbb{F}_q$ τe_1 is also a linear structure of g, implying $D_{\tau e_1} g(x) = D_{L(\tau e_1)} f(L(x))$. Since $L(\tau e_1) = \tau L(e_1) = \tau a$, this concludes the proof. □

Lemma 2. *The multiset of the number of linear structures of the components of G and of G_{pub} is the same. Moreover, $a \in \mathbb{F}_q^n$ is a linear structure of $\lambda \cdot G_{pub}$ if and only if $L_2(a)$ is a linear structure of $L_1^T(\lambda) \cdot G$ where L_1 and L_2 are the linear parts of the affine bijections A_1 and A_2, and L_1^T is the linear function corresponding to the transpose of the matrix defining L_1, that is $x \cdot L_1(y) = L_1^T(x) \cdot y$ for every x, y.*

Proof. It is known that functions in a given EA-class have the same number of components with the same number of linear structures. Indeed, for each nonzero $\lambda \in \mathbb{F}_q^m$ there exists a nonzero $\gamma \in \mathbb{F}_q^m$ such that $\lambda \cdot G_{pub}(x) = \gamma \cdot G(A_2(x))$. Then clearly the number of linear structures of $\lambda \cdot G_{pub}$ equals the number of linear structures of $\gamma \cdot G$.
The second statement can be deduced from the following. From the definition of G_{pub} and since A_1 is affine we have $D_a(\lambda \cdot G_{pub})(x) = \lambda \cdot G_{pub}(x+a) - \lambda \cdot G_{pub}(x) = \lambda \cdot A_1 \circ G \circ A_2(x+a) - \lambda \cdot A_1 \circ G \circ A_2(x) = \lambda \cdot L_1 \circ G \circ A_2(x+a) - \lambda \cdot L_1 \circ G \circ A_2(x)$. Then

$D_a(\lambda \cdot G_{pub})(x) = \lambda \cdot L_1(G(A_2(x) + L_2(a)) - G(A_2(x))) = \lambda \cdot L_1(D_{L_2(a)}G(A_2(x))) = L_1^T(\lambda) \cdot D_{L_2(a)}G(A_2(x))$. □

Now we explain the three steps.

Step 1. By Lemma 2 we can directly count the number of linear structures for the components of G instead of G_{pub}. Notice that for $\lambda = (\lambda', 0) \in \mathbb{F}_q^t \times \mathbb{F}_q^{m-t}$ with any nonzero $\lambda' \in \mathbb{F}_q^t$, the λ-component of G has the form $G_\lambda(x, y) = \lambda' \cdot (x - \mathfrak{q}(y))$. Then, picking any element of the form $a = (a', 0) \in \mathbb{F}_q^t \times \mathbb{F}_q^{n-t}$, it holds $G_\lambda(x + a', y) - G_\lambda(x, y) = \lambda' \cdot (x + a' - \mathfrak{q}(y) - (x - \mathfrak{q}(y))) = \lambda' \cdot a'$. So, a is a linear structure of the function G_λ for any such λ. The number of a of this form is q^t and the number of λ of this form is $q^t - 1$. Therefore, G_{pub} admits at least $q^t - 1$ nonzero components having at least q^t linear structures in common. These common linear structures form a t-dimensional vector space V.

Step 2. To determine V, an attacker has to do the following:

- isolate the components in G_{pub} corresponding to the λ-components of G (with $\lambda = (\lambda', 0) \in \mathbb{F}_q^t \times \mathbb{F}_q^{m-t}$, for any $\lambda' \in \mathbb{F}_q^t$);
- among the isolated components, select t linearly independent, compute the intersection of the linear structures of these components and recover the t-dimensional vector space V.

The desired components of G_{pub} are quadratic; therefore by isolating all the quadratic components of the public function, we are isolating the desired components plus (eventually) the quadratic components of $U(x - \mathfrak{q}(y), y)$. Notice that in all the computational experiments reported in Table 1 $U(x - \mathfrak{q}(y), y)$ did not have components of degree 2.

Step 3. We partially recover the linear part of A_2 in the following way. If we are able to determine V, then we know that $L_2(V) = \mathbb{F}_q^t \times \{0_{n-t}\}$, where we used the notation $\mathbb{F}_q^t \times \{0_{n-t}\} = \{(a_1, \ldots, a_t, 0, \ldots, 0) \in \mathbb{F}_q^n : a_i \in \mathbb{F}_q, 1 \leq i \leq t\}$. Equivalently, $V = L_2^{-1}(\mathbb{F}_q^t \times \{0_{n-t}\})$. We observe that if $t = 1$, then we can recover the first column of L_2^{-1}, up to a multiplication by a nonzero element of $\mathbb{F}_q$. On the other hand, for larger values of t, we only know that there exist t linearly independent elements in V which form the first t columns of L_2^{-1} (corresponding to the evaluation of the first t elements of the canonical basis), but finding the precise value of the columns is still not easy.

6.4 A linearization attack for $s = 0$

We present here an attack which follows the idea of the linearization attack proposed against the MI cryptosystem, see [MI88]. This attack of Patarin [Pat95] works because there exists a bilinear equation in the input $\mathfrak{i}$ and in the output $\mathfrak{o}$ of the public map, namely $B(\mathfrak{i}, \mathfrak{o}) = 0$. This allows first to reconstruct the map B with a relatively small number of input-output pairs, and then given a targeted output, to recover the corresponding input.

First, we consider a `Pesto` primitive with $t = 1$ (so $x = (x_1)$), $s = 0$ and such that $U(x, y) = U(x_1, y)$ does not have the quadratic term x_1^2. In this case we have $U(x_1, y) = x_1\alpha(y) + \beta(y)$, with $\alpha, \beta : \mathbb{F}_q^{n-1} \to \mathbb{F}_q^{m-1}$ affine maps. Hence, for $G(x_1, y) = (c_T, c_U)$ it holds that

$$c_U = U(x_1 - \mathfrak{q}(y), y) = U(c_T, y) = c_T\alpha(y) + \beta(y).$$

The above is a bilinear equation in the input y and the output (c_T, c_U) of G, implying the existence of a bilinear equation in the input and the output of G_{pub}, $B(\mathfrak{i}, \mathfrak{o}) = 0$.

Now, we consider a generic OV map U with $s = 0$ and $t = 1$, then in U we have also the term $x_1^2\delta$, with $\delta \in \mathbb{F}_q^{m-1}$. Thus we have $c_U = c_T\alpha(y) + \beta(y) + c_T^2\delta$, implying that, in order to reconstruct the map for G, and so for G_{pub}, we need more pairs of (input,output)

since the output appears also in quadratic terms. However, once we have reconstructed the equation $B(\mathfrak{i}, \mathfrak{o}) = 0$ for G_{pub}, we have that given a possible output $\bar{\mathfrak{o}}$, the equation $B(\mathfrak{i}, \bar{\mathfrak{o}}) = 0$ is linear in the input.

This same analysis can be generalized to the case $t > 1$ and $s = 0$. Indeed, the map U results to have terms of the form $x_i x_j$, $x_i y_j$, x_i, y_j. So, given $G(x, y) = (c_T, c_U)$ we have $m - t$ quadratic equations in y, c_T, c_U, with the variables in y appear only in degree 1. Hence, also from G_{pub} it is possible to recover an equation $B(\mathfrak{i}, \mathfrak{o}) = 0$ which is quadratic but with the variables of the input appearing only in degree 1. Hence the same attack can be performed. For this reason, we choose $s > 0$ in Definition 3.

6.5 Known attacks on Oil and Vinegar systems

The secret map F is formed by two OV systems $T(x, y)$ and $U(x, y)$. So one may wonder whether the known attacks to OV systems can be performed also to the scheme `Pesto`.

After the twisting transformation is applied, the first t coordinates (of the map G) remain an OV system $x - \mathfrak{q}(y)$, while the last $m - t$ coordinates might increase up to degree 4. So, if one is able to isolate from G_{pub} the components corresponding to $x - \mathfrak{q}(y)$, known attacks to OV systems can be applied to these components only. Therefore, given the well-known Kipnis–Shamir attack [KS98] on balanced Oil and Vinegar signature schemes, we recommend to keep the system $x - \mathfrak{q}(y)$ unbalanced by setting, for example, $t \approx n/3$.

Since the second part of the system G is formed by polynomials of degree up to 4, we believe that the usual OV attacks cannot easily be applied to the whole map G_{pub}.

6.6 Algebraic attack with Gröbner bases

In this section, we consider algebraic attacks using Gröbner bases. The scenario is that an attacker wants to forge a signature, hence to find a preimage of a random element. We consider a random value $w \in \mathbb{F}_q^m$ (the hash of a document), the goal is to find $v \in \mathbb{F}_q^n$ such that $G_{pub}(v) = w$. We can do this by finding a Gröbner basis of the polynomial system G_{pub} by the usual strategy (see e.g. [CG21b]) with linear-algebra-based-algorithms such as F4 [Fau99], F5 [Fau02], XL [CKPS00], etc. The complexity of these algorithms is bounded from above by

$$O\left(\binom{n + \mathrm{sd}(G_{pub})}{n}^{\omega}\right)$$

where n is the number of variables, $\mathrm{sd}(G_{pub})$ is the solving degree, and $2 < \omega < 3$. In a nutshell, the solving degree represents the highest degree of polynomials that need to be considered during the process of solving the system [CG22]. In order to estimate this for the system G_{pub} of `Pesto`, we performed some computational experiments with MAGMA software [BCP97] for different values of the parameters n, m, t, s, q. We also computed the Castelnuovo–Mumford regularity of the homogenized system G^h_{pub}, which gives an upper bound for the solving degree [CG21b]. The results are summarized in the following table.

Table 3: Mean values of solving degree and regularity for 50 randomly generated systems

	n	m	t	s	$\mathrm{sd}(G_{pub})$	$\mathrm{reg}(G^h_{pub})$
$q = 5$	5	4	2	1	4.54	7.6
$q = 2^6$	5	4	2	1	4.06	7.96
$q = 3761$	5	4	2	1	4	8
$q = 5$	6	5	2	2	7.64	8.74
$q = 2^6$	6	5	2	2	7.94	9
$q = 3761$	6	5	2	2	8	9

For higher parameters, we were not able to compute the above values. Therefore, we added to G_{pub} the field equations $x_i^q - x_i = 0$, $y_i^q - y_i = 0$ and then we computed the solving degree for this larger set of equations. Clearly, this strategy is not efficient for large values of q, hence we only considered $q = 5$. As for the previous experiments, we considered some different randomly generated systems G_{pub} and then we display the mean value for $\mathrm{sd}(G_{pub})$, see Table 4.

Table 4: Mean value of the solving degree for 25 randomly generated systems G_{pub} over $\mathbb{F}_5$

n	m	t	s	$\mathrm{sd}(G_{pub})$
10	6	2	1	9.05
10	8	2	2	8.8
10	8	3	2	7.36

7 Conclusions

In this work, we propose applying a CCZ transformation in the construction of a multivariate scheme, instead of the usual affine transformation. This has the advantage of hiding linear relations between the input and output that would occur with an affine transformation alone. However, this approach may increase the public key size.

With this work, we hope to build a fruitful bridge between the areas of cryptographic Boolean functions and Multivariate Cryptography. Since both areas work with functions/polynomials defined over finite fields, we believe that many techniques used in one area could be studied and applied in the other. We see significant potential for further exploration in this direction.

References

[BCP97] Wieb Bosma, John Cannon, and Catherine Playoust. The Magma algebra system I: The user language. *Journal of Symbolic Computation*, 24(3-4):235–265, 1997.

[BCP06] Lilya Budaghyan, Claude Carlet, and Alexander Pott. New classes of almost bent and almost perfect nonlinear polynomials. *IEEE Transactions on Information Theory*, 52(3):1141 – 1152, 2006. All Open Access, Green Open Access. doi:10.1109/TIT.2005.864481.

[BCV20] Lilya Budaghyan, Marco Calderini, and Irene Villa. On relations between CCZ-and EA-equivalences. *Cryptography and Communications*, 12:85–100, 2020.

[BES13] Edoardo Ballico, Michele Elia, and Massimiliano Sala. On the evaluation of multivariate polynomials over finite fields. *Journal of Symbolic Computation*, 50:255–262, 2013.

[BFP13] Luk Bettale, Jean-Charles Faugère, and Ludovic Perret. Cryptanalysis of HFE, multi-HFE and variants for odd and even characteristic. *Designs, Codes, and Cryptography*, 69(1):1 – 52, 2013. All Open Access, Green Open Access. doi:10.1007/s10623-012-9617-2.

[BL17] Daniel J. Bernstein and Tanja Lange. Post-quantum cryptography. *Nature*, 549(7671):188 – 194, 2017. All Open Access, Green Open Access. doi:10.1038/nature23461.

[BV18] Stephen Boyd and Lieven Vandenberghe. *Introduction to applied linear algebra: vectors, matrices, and least squares*. Cambridge university press, 2018.

[Car20] Claude Carlet. *Boolean functions for cryptography and coding theory*. Cambridge University Press, New York, 2020. doi:10.1017/9781108606806.

[CCZ98] Claude Carlet, Pascale Charpin, and Victor Zinoviev. Codes, bent functions and permutations suitable for DES-like cryptosystems. *Designs, Codes and Cryptography*, 15:125–156, 1998.

[CG21a] Alessio Caminata and Elisa Gorla. The complexity of MinRank. *Association for Women in Mathematics Series*, 24:163 – 169, 2021. doi:10.1007/978-3-030-77700-5_5.

[CG21b] Alessio Caminata and Elisa Gorla. Solving multivariate polynomial systems and an invariant from commutative algebra. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 12542 LNCS:3 – 36, 2021. doi:10.1007/978-3-030-68869-1_1.

[CG22] Alessio Caminata and Elisa Gorla. Solving degree, last fall degree, and related invariants. *Journal of Symbolic Computation*, 114:322 – 335, 2022. doi:10.1016/j.jsc.2022.05.001.

[CGP92] Nicolas Courtois, Louis Goubin, and Jacques Patarin. SFLASH, a fast asymmetric signature scheme for low-cost smartcards–primitive specification and supporting documentation. In *URL https://www. cosic. esat. kuleuven. ac. be/nessie/workshop*. Citeseer, 1992.

[CKPS00] Nicolas Courtois, Alexander Klimov, Jacques Patarin, and Adi Shamir. Efficient algorithms for solving overdefined systems of multivariate polynomial equations. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 1807:392 – 407, 2000. doi:10.1007/3-540-45539-6_27.

[CP19] Anne Canteaut and Léo Perrin. On CCZ-equivalence, extended-affine equivalence, and function twisting. *Finite Fields and Their Applications*, 56:209–246, 2019.

[DFSS07] Vivien Dubois, Pierre-Alain Fouque, Adi Shamir, and Jacques Stern. Practical cryptanalysis of SFLASH. In *Annual International Cryptology Conference*, pages 1–12. Springer, 2007.

[Din04] Jintai Ding. A new variant of the Matsumoto-Imai cryptosystem through perturbation. In *International Workshop on Public Key Cryptography*, pages 305–318. Springer, 2004.

[DPS20] Jintai Ding, Albrecht Petzoldt, and Dieter S. Schmidt. *Oil and Vinegar*, pages 89–151. Springer US, New York, NY, 2020. doi:10.1007/978-1-0716-0987-3_5.

[DS05] Jintai Ding and Dieter Schmidt. Rainbow, a new multivariable polynomial signature scheme. In *International conference on applied cryptography and network security*, pages 164–175. Springer, 2005.

[DY09] Jintai Ding and Bo-Yin Yang. *Multivariate Public Key Cryptography*, pages 193–241. Springer Berlin Heidelberg, Berlin, Heidelberg, 2009. doi:10.1007/978-3-540-88702-7_6.

[Fau99] Jean-Charles Faugére. A new efficient algorithm for computing Gröbner bases (F4). *Journal of Pure and Applied Algebra*, 139(1):61 – 88, 1999. doi:10.1016/S0022-4049(99)00005-5.

[Fau02] Jean-Charles Faugère. A new efficient algorithm for computing Gröbner bases without reduction to zero (F5). page 75 – 83, 2002. doi:10.1145/780506.780516.

[FGS05] Pierre-Alain Fouque, Louis Granboulan, and Jacques Stern. Differential cryptanalysis for multivariate schemes. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 341–353. Springer, 2005.

[KS98] Aviad Kipnis and Adi Shamir. Cryptanalysis of the oil and vinegar signature scheme. In *Annual international cryptology conference*, pages 257–266. Springer, 1998.

[LC00] Louis Goubin Louis and Nicolas T. Courtois. Cryptanalysis of the TTM cryptosystem. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 1976:44 – 57, 2000.

[MI88] Tsutomu Matsumoto and Hideki Imai. Public quadratic polynomial-tuples for efficient signature-verification and message-encryption. In *Advances in Cryptology—EUROCRYPT'88: Workshop on the Theory and Application of Cryptographic Techniques Davos, Switzerland, May 25–27, 1988 Proceedings 7*, pages 419–453. Springer, 1988.

[Pat95] Jacques Patarin. Cryptanalysis of the Matsumoto and Imai public key scheme of Eurocrypt'88. In *Advances in Cryptology—CRYPT0'95: 15th Annual International Cryptology Conference Santa Barbara, California, USA, August 27–31, 1995 Proceedings 15*, pages 248–261. Springer, 1995.

[Pat96] Jacques Patarin. Hidden fields equations (HFE) and isomorphisms of polynomials (IP): Two new families of asymmetric algorithms. In *International Conference on the Theory and Applications of Cryptographic Techniques*, pages 33–48. Springer, 1996.

[Pat97] Jacques Patarin. The oil and vinegar signature scheme. *Dagstuhl Workshop on Cryptography September, 1997*, 1997. URL: https://cir.nii.ac.jp/crid/1571417125795682304.

[VST17] Jeremy Vates and Daniel Smith-Tone. Key recovery attack for all parameters of HFE-. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 10346 LNCS:272 – 288, 2017. doi:10.1007/978-3-319-59879-6_16.