On the Complexity of the Hybrid Approach on HFEv-

Albrecht Petzoldt

National Institute of Standards and Technology,
Gaithersburg, Maryland, USA
albrecht.petzoldt@gmail.com

Abstract. The HFEv- signature scheme is one of the most promising candidates for post-quantum digital signatures. Most notably here is the short signature size of the scheme.
It has long been known that direct attacks against HFEv- systems work more efficiently than against random systems. The reason for this was found by Jintai Ding et al., who proved an upper bound on the degree of regularity of these systems. However, not much is known about the efficiency of the hybrid approach against the HFEv- scheme. In order to find suitable parameter sets for HFEv- for higher levels of security, this topic has to be studied in more detail.
In this article we consider this question by performing a large number of computer experiments. As our experiments show, guessing variables does not help to speed up direct attacks against HFEv- systems. Therefore, in the parameter selection of these schemes, we do not have to consider the hybrid approach. Furthermore, we develop in this article a simple formula to estimate the degree of regularity of a determined HFEv- system. Together with our results on the behavior of the hybrid approach, this formula gives us an easy way to estimate the complexity of direct attacks against HFEv- systems.

Keywords: Multivariate Cryptography, HFEv-, Direct Attack, Hybrid Approach

1 Introduction

The HFEv- signature scheme as proposed by Patarin, Courtois and Goubin in [11] is one of the best studied multivariate schemes and one of the most promising candidates for post-quantum digital signatures. Most notably is the short signature size of the scheme, which allows us to generate signatures of length less than two times the security level. Therefore, HFEv- produces the shortest signatures of all existing signature schemes.

Experiments [7, 10] have shown that direct attacks can solve the public systems of HFEv- much faster than random systems. The reason for this was found by Jintai Ding and Bo Yin Yang in [5], who proved an upper bound on the degree of regularity of HFEv- systems. However, not much is known about the effect of the hybrid approach [1] against HFEv- systems. Guessing variables before applying a direct attack often reduces the overall complexity of the attack, even if this implies to perform the algorithm several times. With respect to a possible future standardization of the HFEv- signature scheme, we therefore have to study the efficiency of the hybrid approach against HFEv- schemes in detail.

In this article we study the efficiency of direct attacks using the hybrid approach against HFEv- schemes and answer the question whether it is sensible to guess variables before applying an algorithm like XL or a Gröbner basis technique such as F_4 [6] or F_5. To do this, we perform a large number of computer experiments with HFEv- systems of different size using MAGMA.

As our experiments show, guessing variables does not help to speed up direct attacks against HFEv- schemes. In detail, when guessing a reasonably small number of variables, the degree of regularity of the system does not change; when guessing more variables, the system behaves exactly like a random system of the same size. Therefore, we do not have to consider the hybrid approach when selecting parameters for the HFEv- scheme. Furthermore, we derive from our experiments a simple formula to estimate the degree of regularity of an HFEv- system in practice. Together with our results on the behavior of the hybrid approach, this formula give us an easy way to estimate the complexity of a direct algebraic attack against an HFEv- scheme.

2 The HFEv- Cryptosystem

2.1 Multivariate cryptography

The basic objects of multivariate cryptography are systems of multivariate quadratic polynomials over a finite field $\mathbb{F} = \mathbb{F}_q$. The security of multivariate schemes is based on the *MQ Problem* of solving such a system. The MQ Problem is proven to be NP-Hard even for quadratic polynomials over the field GF(2) [8] and believed to be hard on average (both for classical and quantum computers).

To build a multivariate public key cryptosystem (MPKC), one starts with an easily invertible quadratic map $\mathcal{F} : \mathbb{F}^n \rightarrow \mathbb{F}^m$ (*central map*). To hide the structure of $\mathcal{F}$ in the public key, we compose it with two invertible affine (or linear) maps $\mathcal{S} : \mathbb{F}^m \rightarrow \mathbb{F}^m$ and $\mathcal{T} : \mathbb{F}^n \rightarrow \mathbb{F}^n$. The *public key* of the scheme is therefore given by $\mathcal{P} = \mathcal{S} \circ \mathcal{F} \circ \mathcal{T} : \mathbb{F}^n \rightarrow \mathbb{F}^m$.
The *private key* consists of the three maps $\mathcal{S}, \mathcal{F}$ and $\mathcal{T}$ and therefore allows to invert the public key.

To generate a signature for a document (hash value) $\mathbf{h} \in \mathbb{F}^m$, one computes recursively $\mathbf{x} = \mathcal{S}^{-1}(\mathbf{h}) \in \mathbb{F}^m$, $\mathbf{y} = \mathcal{F}^{-1}(\mathbf{x}) \in \mathbb{F}^n$ and $\mathbf{z} = \mathcal{T}^{-1}(\mathbf{y})$.
To check the authenticity of a signature $\mathbf{z} \in \mathbb{F}^n$, one simply computes $\mathbf{h}' =$

$\mathcal{P}(\mathbf{z}) \in \mathbb{F}^m$. If the result is equal to $\mathbf{h}$, the signature is accepted, otherwise rejected. This process is illustrated in Figure 1.

Signature Generation

$$\begin{tikzcd}
\mathbf{h} \in \mathbb{F}^m \arrow[r, "\mathcal{S}^{-1}"] & \mathbf{x} \in \mathbb{F}^m \arrow[r, "\mathcal{F}^{-1}"] & \mathbf{y} \in \mathbb{F}^n \arrow[r, "\mathcal{T}^{-1}"] & \mathbf{z} \in \mathbb{F}^n \arrow[lll, bend left, "\mathcal{P}"]
\end{tikzcd}$$

Signature Verification

Fig. 1. Signature Generation and Verification for Multivariate Signature Schemes

2.2 The HFEv- Signature Scheme

The HFEv- signature scheme was proposed by Patarin, Courtois and Goubin in [11]. It is an example of a multivariate BigField scheme, which means that the central map $\mathcal{F}$ is a univariate polynomial map over a degree n extension field $\mathbb{E}$ of $\mathbb{F}$. In order to switch between the ground field $\mathbb{F}$ and the extension field $\mathbb{E}$, we use an isomorphism $\phi : \mathbb{F}^n \to \mathbb{E}$.

The *central map* $\mathcal{F}$ of the HFEv- scheme has the form

$$\mathcal{F}(X) = \sum_{0 \leq i,j}^{q^i+q^j \leq D} \alpha_{ij} X^{q^i+q^j} + \sum_{i=0}^{q^i \leq D} \beta_i(x_1, \ldots, x_v) X^{q^i} + \gamma(x_1, \ldots, x_v),$$

where β_i and γ are linear and quadratic maps in the vinegar variables $x_1, \ldots, x_v$ respectively. The *public key* has the form

$$\mathcal{P} = \mathcal{S} \circ \phi^{-1} \circ \mathcal{F} \circ \phi \circ \mathcal{T} : \mathbb{F}^{n+v} \to \mathbb{F}^{n-a}$$

with two affine maps $\mathcal{S} : \mathbb{F}^n \to \mathbb{F}^{n-a}$ and $\mathcal{T} : \mathbb{F}^{n+v} \to \mathbb{F}^{n+v}$ and is a multivariate quadratic map with coefficients and variables over $\mathbb{F}$.

The *private key* consists of the three maps $\mathcal{S}, \mathcal{F}$ and $\mathcal{T}$.

Signature Generation: To generate a signature $\mathbf{z}$ for a document d, one uses a hash function $\mathcal{H} : \{0, 1\} \to \mathbb{F}^{n-a}$ to compute a hash value $\mathbf{h} = \mathcal{H}(d) \in \mathbb{F}^{n-a}$ and performs the following four steps

1. Compute a preimage $\mathbf{x} \in \mathbb{F}^n$ of $\mathbf{h}$ under the affine map $\mathcal{S}$ and set $X = \phi(\mathbf{x}) \in \mathbb{E}$.

2. Choose random values for the vinegar variables $x_1, \ldots, x_v$ and substitute them into the central map to obtain the parametrized map $\mathcal{F}_V$.
3. Solve the equation $\mathcal{F}_V(Y) = X$ over the extension field $\mathbb{E}$ by Berlekamp's algorithm.
4. Compute $\mathbf{y} = \phi^{-1}(Y)$ and the signature $\mathbf{z} \in \mathbb{F}^{n+v}$ by $\mathbf{z} = \mathcal{T}^{-1}(\mathbf{y}||x_1|| \ldots ||x_v)$.

Signature Verification: To check the authenticity of a signature $\mathbf{z}$, the verifier computes $\mathbf{h} = \mathcal{H}(d)$ and $\mathbf{h}' = \mathcal{P}(\mathbf{z})$. If $\mathbf{h}' = \mathbf{h}$ holds, the signature is accepted, otherwise it is rejected.

2.3 Gui

For performance reasons, the HFEv- signature scheme is mostly used over the field GF(2). However, in order to prevent colission attacks against the scheme, this would lead to a very large public key (In order to reach a security level of k bits against colission attacks, we would need $2k$ equations over GF(2)). To avoid this, Petzoldt et al. developed in [12] a specially designed signature generation process for HFEv- based schemes. The scheme was named Gui.

In the signature generation process of Gui, k HFEv- signatures are created (for different hash values of the same message d). These k HFEv- signatures are then combined to a single Gui signature of length $(n-a)+k\cdot(a+v)$ bit (see Algorithm 1). Analogously, the verification algorithm of Gui evaluates the HFEv- public key k times (see Algorithm 2).

Algorithm 1 Signature Generation Process of Gui

```
Input: Gui private key ($\mathcal{S}$, $\mathcal{F}$, $\mathcal{T}$), message $\mathbf{d}$, repetition factor $k$
Output: signature $\sigma \in \mathrm{GF}(2)^{(n-a)+k(a+v)}$
$\mathbf{h} \leftarrow$ SHA-256($\mathbf{d}$)
$S_0 \leftarrow \mathbf{0}$
for $i = 1$ to $k$ do
    $D_i \leftarrow$ first $n-a$ bits of $\mathbf{h}$
    $(S_i, X_i) \leftarrow \mathrm{HFEv-}^{-1}(D_i \oplus S_{i-1})$
    $\mathbf{h} \leftarrow$ SHA-256($\mathbf{h}$)
end for
$\sigma \leftarrow (S_k||X_k||\ldots||X_1)$
return $\sigma$
```

Algorithm 2 Signature Verification Process of Gui

```
Input: Gui public key $\mathcal{P}$, message $\mathbf{d}$, signature $\sigma \in \mathrm{GF}(2)^{(n-a)+k(a+v)}$, repetition factor $k$
Output: TRUE or FALSE
$\mathbf{h} \leftarrow$ SHA-256($\mathbf{d}$)
$(S_k, X_k, \ldots, X_1) \leftarrow \sigma$
for $i = 1$ to $k$ do
    $D_i \leftarrow$ first $n-a$ bits of $\mathbf{h}$
    $\mathbf{h} \leftarrow$ SHA-256($\mathbf{h}$)
end for
for $i = k-1$ to $0$ do
    $S_i \leftarrow \mathcal{P}(S_{i+1}||X_{i+1}) \oplus D_{i+1}$
end for
if $S_0 = \mathbf{0}$ then
    return TRUE
else
    return FALSE
end if
```

The Gui signature scheme solves the above problem with the colission resistance of the hash function by using a hash of effective length $k \cdot |\mathcal{H}|$. Therefore, with regard to the hash function, it would be possible to use in Gui an HFEv- scheme with an arbitrary small number of equations (and adjusting the repetition factor k appropriately). However, there are other attacks which put a lower bound on the number of equations. Despite the new signature generation process of Gui, its security is equivalent to that of HFEv- and all known attacks against Gui are attacks against the underlying HFEv- scheme.

2.4 Attacks against HFEv-

The main attacks against the HFEv- cryptosystem are

- (quantum) brute force attacks
- direct attacks
- Rank attacks of the Kipnis-Shamir type

While the first two of these attacks are signature forgery attacks, which have to be performed for each message separately (and, due to the specially designed signature generation process of Gui, for every message k times), the Kipnis-Shamir attack is a key recovery attack. After having recovered the HFEv- private key using this method, it is possible to generate Gui signatures in the same way as a legitimate user.

Brute Force Attacks Since the public system of Gui is defined over the field GF(2) with two elements, the parameters of the scheme have to be chosen in a way that prevents brute force attacks against binary MQ systems. In the classical world, we have to mention here the Gray Code enumeration of [3]. In order to solve a public HFEv- system using this technique, one first fixes $a + v$ variables to get a determined system. The resulting system of $n - a$ equations in $n - a$ variables can then be evaluated for every possible input using $2^{n-a+2} \cdot \log_2(n-a)$ bit operations. In order to forge a Gui signature, we have to perform this step k times. Therefore, the complexity of this attack can be estimated as

$$\text{Complexity}_{\text{brute force; classical}} = k \cdot 2^{n-a+2} \cdot \log_2(n-a)$$

bit operations.

In the quantum world, brute force attacks can be additionally sped up using Grover's algorithm. As shown in [13], we can find the solution of a binary MQ system of $n - a$ equations in $n - a$ variables using $2^{(n-a)/2} \cdot 2 \cdot (n-a)^3$ quantum bit operations. The complexity of forging a Gui signature using this technique can be estimated as

$$\text{Complexity}_{\text{brute force; quantum}} = k \cdot 2^{(n-a)/2} \cdot 2 \cdot (n-a)^3$$

(quantum) bit operations.

Direct Attacks Besides brute force attacks, direct algebraic attacks are the most straightforward way to attack a multivariate scheme such as HFEv-. One considers the public equation

$$\mathcal{P}(\mathbf{z}) = \mathbf{w} \tag{1}$$

as an instance of the MQ Problem and tries to solve it using a system solver like XL or a Gröbner basis technique such as F_4 [6] or F_5. In the case of a multivariate signature scheme such as HFEv-, the equation (1) is an underdetermined multivariate quadratic system (i.e .the number n of variables is larger than the number m of equations). In this case it is the best strategy to fix $n - m$ of the variables before applying the Gröbner basis algorithm. One can assume that the so projected system has exactly one solution. The complexity of a direct attack against a determined system of n quadratic equations in n variables can be estimated by

$$\text{Complexity}_{\text{direct}} = 3 \cdot \binom{n + d_{\text{reg}}}{d_{\text{reg}}}^2 \cdot \binom{n}{2}, \tag{2}$$

where d_{reg} is the so called degree of regularity of the system.
For GF(2) as the underlying field, we have to consider the field equations $\{x_i^2 - x_i\}$, which reduces the number of monomials in the extended systems produced by the F_4 algorithm. Therefore, we have to change the above formual slightly. We get

$$\text{Complexity}_{\text{direct; GF(2)}} = 3 \cdot \binom{n}{d_{\text{reg}}}^2 \cdot \binom{n}{2}, \tag{3}$$

Since HFEv- is mainly used over the field GF(2), we use in the following this formula.

The Hybrid Approach The idea of the hybrid approch [1] is to guess some (say ℓ) additional variables (therefore creating an overdetermined system) before applying the Gröbner basis algorithm. Even if this implies to run the algorithm q^ℓ times, it often leads to better results. The complexity of solving a binary system by the hybrid approach can be estimated by

$$\text{Complexity}_{\text{hybrid; classical}} = \min_\ell \; q^\ell \cdot 3 \binom{n-\ell}{d_{\text{reg}}}^2 \cdot \binom{n-\ell}{2}. \tag{4}$$

In the presence of quantum computers, we can use Grover's algorithm to reduce this complexity to

$$\text{Complexity}_{\text{hybrid; quantum}} = \min_\ell \; q^{\ell/2} \cdot 3 \binom{n-\ell}{d_{\text{reg}}}^2 \cdot \binom{n-\ell}{2}. \tag{5}$$

In order to forge a Gui signature, we have to perform this attack k times.

Rank Attacks In [9] Kipnis and Shamir proposed a rank attack against the HFE cryptosystem. The key idea of this attack is to consider the public and private maps of HFE as univariate polynomial maps over the extension field. Due to the special structure of the HFE central map, the rank of the corresponding matrix is limited by $r = \lfloor \log_2(D-1) \rfloor + 1$. It is therefore possible to reconstruct the affine transformation $\mathcal{S}$ by solving an instance of the MinRank problem.
In [2], Bouillaget et al. improved this attack by showing that the map $\mathcal{S}$ can be found by computing a Gröbner Basis over the base field GF(2). By doing so, we can estimate the complexity of a Rank attack of the Kipnis Shamir type against the basic HFE scheme by

$$\text{Complexity}_{\text{KS attack; HFE}} = \binom{n+r}{r}^{\omega},$$

where $r = \lfloor \log_2(D-1) \rfloor + 1$ is the rank of the matrix corresponding to the central map and $2 < \omega \leq 3$ is the linear algebra factor.
In the case of HFEv-, the rank of the matrix is given by $r + a + v$. Therefore, we can estimate the complexity of our attack by

$$\text{Complexity}_{\text{KS attack; HFEv}-} = \binom{n+r+a+v}{r+a+v}^{\omega}.$$

Since the quadratic systems to be solved during this attack are highly overdetermined, the attack can not be sped up with the help of Grover's algorithm.

3 Our Experiments

3.1 The Direct Attack on HFEv-

Experiments [7, 10] have shown that the public systems (1) of HFE and its variants can be solved significantly faster than random systems. The reason for this is the smaller degree of regularity of these systems. In [5] it was shown that for HFEv- systems, the degree of regularity is upper bounded by

$$\mathrm{d_{reg}} \leq \begin{cases} \frac{(q-1)\cdot(r+a+v-1)}{2} + 2 & \text{for } q \text{ even and } r+a \text{ odd}, \\ \frac{(q-1)\cdot(r+a+v)}{2} + 2 & \text{otherwise.} \end{cases}, \quad (6)$$

where $r = \lfloor \log_q(D-1) \rfloor + 1$.
However, when estimating the actual complexity of a direct attack against HFEv- (or Gui), this upper bound on d_{reg} does not really help. We therefore performed a number of experiments to estimate the degree of regularity of HFEv- systems in practice. For this, we created for $D \in \{5, 9, 17, 33, 65\}$, different values of $a+v$ and increasing values of n the public systems of HFEv-(n, D, a, v) and solved these systems using the F_4 algorithm integrated in MAGMA. Table 1 shows for $D \in \{5, 9, 17, 33, 65\}$ the minimal value of $a+v$ needed to reach various degrees of regularity [1] . Note that the values of $a+v$ listed in the table are not necessarily

[1] Note that parts of this analysis was already done in [12].

d_{reg}	minimal number of $a+v$				
	$D=5$	$D=9$	$D=17$	$D=33$	$D=65$
4	2	1	0	0	0
5	5	4	3	2	1
6	8	7	6	5	4
7	11	10	8	7	6

Table 1. Minimal value of $a+v$ needed to reach given degrees of regularity

the smallest possible. However, with our constrained computing resources, we could not perform experiments with more than 37 equations. Therefore, for larger values of n, the values of $a+v$ listed in Table 1 can be seen as sufficient but not necessarily minimal values to reach the given degree of regularity.
From Table 1 we find

$$a+v \lesssim \begin{cases} 3 \cdot d_{\mathrm{reg}} - 10 & \text{for } D=5 \\ 3 \cdot d_{\mathrm{reg}} - 11 & \text{for } D=9 \\ 3 \cdot d_{\mathrm{reg}} - 12 & \text{for } D=17 \\ 3 \cdot d_{\mathrm{reg}} - 13 & \text{for } D=33 \text{ and} \\ 3 \cdot d_{\mathrm{reg}} - 14 & \text{for } D=65 \end{cases} . \tag{7}$$

Again we note that, for large values of n, this equation yields sufficient, but not necessarily minimal values of $a+v$.
By substituting $r = \lfloor \log_2(D-1) \rfloor + 1$ into equation (7), we get

$$a+v \lesssim 3 \cdot d_{\mathrm{reg}} - r - 7 \tag{8}$$

or

$$d_{\mathrm{reg}} \geq \lfloor \frac{r+a+v+7}{3} \rfloor. \tag{9}$$

In contrast to equation (6), formula (9) yields a lower bound for the degree of regularity of an HFEv- system over GF(2). By estimating

$$d_{\mathrm{reg}}^{\mathrm{HFEv-;\ GF(2)}} = \lfloor \frac{r+a+v+7}{3} \rfloor,$$

we therefore get a lower bound for the complexity of direct attacks against HFEv- .

Another important point when studying the complexity of direct attacks against HFEv- systems is that, for efficiency reasons, the HFEv- signature scheme is defined over very small fields (in this paper we consider HFEv- schemes over GF(2)). This means that the guessing part of the hybrid approach is very cheap. In order to estimate the complexity of direct attacks against HFEv- systems, we therefore have to carefully study the behavior of the hybrid approach against these systems.

3.2 Experiments with the Hybrid Approach against HFEv-

In this section we present the results of our computer experiments with the hybrid approach against HFEv- systems. Our goal is to decide whether it is sensible to guess variables (and creating so an overdetermined system) before applying a Gröbner basis algorithm. Note that guessing implies to run the Gröbner basis algorithm several times.

For our experiments, we generated for $n - a \in \{30, 35\}$ and different values of D, a and v public HFEv- systems in MAGMA code. After adding the field equations $\{x_i^2 - x_i = 0 : i = 1, \ldots, n+v\}$, we appended $k \in \{0, \ldots, n+v-10\}$ randomly chosen linear equations to the system (thus projecting down the quadratic system to $n - a - k \in \{10, \ldots, n - a\}$ variables) and solved the systems using the F_4 algorithm integrated in MAGMA. The experiments were performed on a server with 16 AMD Opteron processors (2.5 GHz) and 128 GB memory. However, for each experiment, we used only one single core.

Figures 2 and 3 show (some of) the results of our experiments. We found:

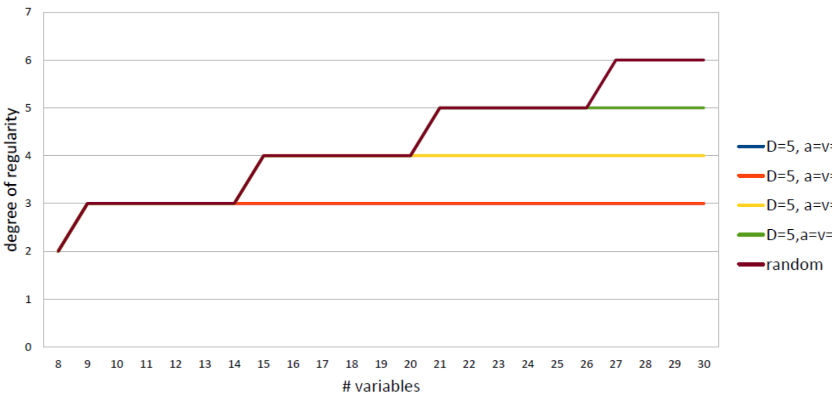

Fig. 2. Hybrid attack against HFEv- systems with $(n - a) = 30$ equations and $D = 5$

- when guessing only a few variables, the degree of regularity of the HFEv- system does not change; memory requirements and the execution time of a single run of the algorithm decrease slightly, but not in an amount that would justify guessing of variables.
- when guessing many variables, the HFEv- systems behave exactly like random systems of the same size; this holds not only with respect to the degree of regularity but also regarding matrix sizes and execution times.
- similar to the results of [12] we found that, as long as the number $n - a$ of equations, D and $s = a + v$ are constant, the concrete choice of a and v has

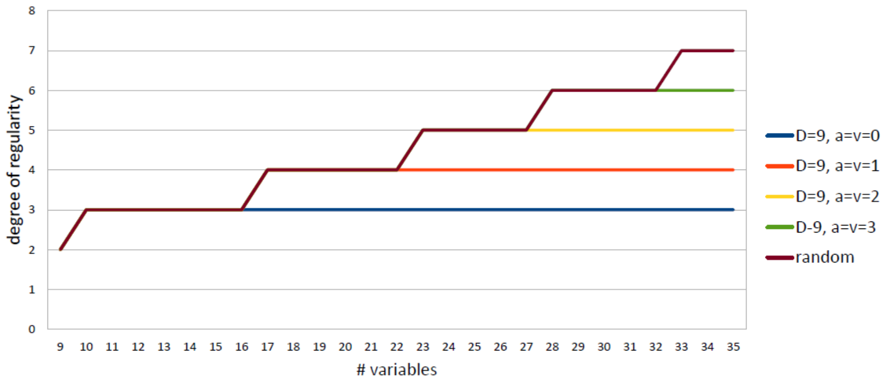

Fig. 3. Hybrid attack against HFEv- systems with $(n-a) = 35$ equations and $D = 9$

> no significant effect on the behavior of direct attacks against the (projected) systems

3.3 Estimating the location of the first degree drop

An important question when studying the efficiency of the hybrid approach is the following:

> How many variables do we have to guess in order to observe a drop of the degree of regularity?

In this section we consider this question for HFEv- systems over GF(2).

Let the HFEv- parameters n, D, a and v be fixed. According to formula (9), we can estimate the degree of regularity of solving the determined HFEv- system (without guessing) as

$$d_{\text{reg}}^{\text{HFEv}-;\text{GF}(2)} = \lfloor \frac{a+v+r+7}{3} \rfloor. \tag{10}$$

In order to estimate the number of variables we have to guess to see a degree drop, we consider a determined random system of $n-a$ equations over GF(2). The degree of regularity of solving this system (after guessing k variables) is given as the smallest index t for which the coefficient of X^t in

$$\frac{1}{1-X} \cdot \left(\frac{1-X^2}{1-X}\right)^{n-a-k} \cdot \left(\frac{1-X^2}{1-X^4}\right)^{n-a} \tag{11}$$

is non positive [14].

For most HFEv- parameters and $k = 0$, the so computed degree of regularity will be much higher than the value computed by equation (10). However, by increasing k, the degree of regularity needed to solve the random system will decrease, while the degree of regularity of the HFE system stays constant (see Section 3.2). We succesively increase k until the degree of regularity of the projected random system drops below the bound given by equation (10). The minimal value of k for which this happens corresponds to the minimal number of variables we have to guess to observe a degree drop of the HFEv- system.

In the following, we illustrate this process using the HFEv- parameters $(q, n, D, a, v) = (2, 95, 9, 5, 5)$. Note that these are exactly the parameters of the scheme Gui-95 [12].
By equation (10) we find that the degree of regularity of the HFEv- system is given by

$$d_{\text{reg}}^{\text{HFEv}-} = \lfloor \frac{5+5+4+7}{3} \rfloor = 7. \tag{12}$$

Note that the Gui-95 scheme is assumed in [12] to have exactly this degree of regularity leading to the claimed (pre-quantum) security level of 80 bit.

The degree of regularity of a determined system with $n - a = 90$ equations over GF(2) can be estimated (c.f. formula (11)) to be 13. We increase the number of guessed variables in the random system (such creating overdetermined systems) to decrease its degree of regularity (see Figure 4).

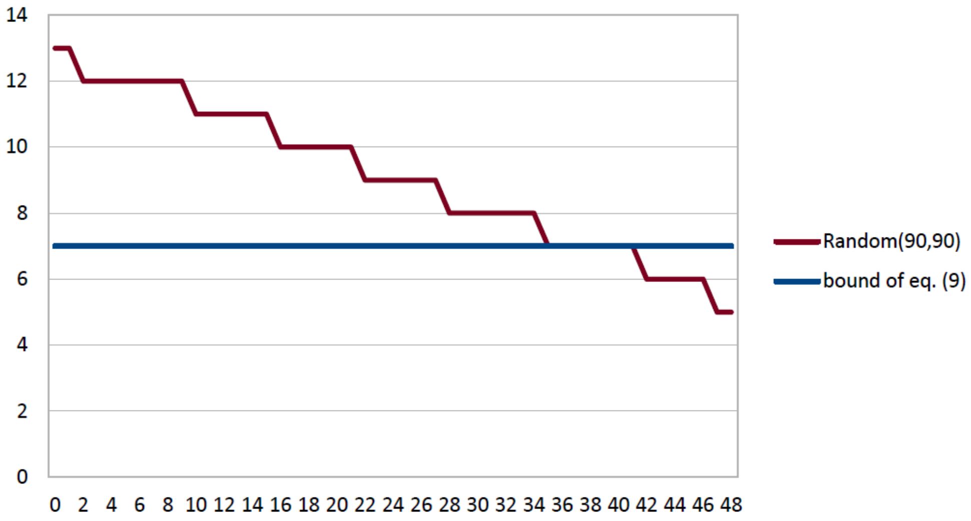

Fig. 4. Estimating the location of the first degree drop

As the figure shows, we meet the bound given by equation (12) for $k = 35$ and go below it for $k = 42$. Therefore, when attacking the given HFEv- instance using the hybrid approach, we would have to perform the F_4 algorithm 2^{42} times (for a system, whose degree of regularity is only one less than that of the original system). We therefore come to the conclusion

> **The hybrid approach does not speed up direct attacks against HFEv- systems.**

4 Conclusion

In this paper we studied the complexity of direct attacks using the hybrid approach against HFEv- systems. We found that

- when guessing only a few variables, guessing does not decrease the degree of regularity of the HFEv- system
- when guessing a large number of variables, the HFEv- system behaves just like a random system

Based on these observations we conclude that

> **The hybrid approach does not speed up direct attacks against HFEv- systems.**

Furthermore, we developed a simple formula to estimate the degree of regularity of a determined HFEv- system over GF(2) in practice. Together with our findings on the behavior of the hybrid approach against HFEv- systems, this formula gives us an easy way to estimate the complexity of direct algebraic attacks against HFEv- systems.

References

1. L. Bettale, J.C. Faugére, L. Perret: Hybrid approach for solving multivariate systems over finite fields. J. Math. Cryptol. 3, pp. 177 - 197 (2009). 1 - 52.
2. L. Bettale, J.C. Faugére, L. Perret: Cryptanalysis of HFE, multi-HFE and variants for odd and even characteristic. Designs Codes and Cryptography 69 (2013), pp.
3. C. Bouillaguet, H.-C. Chen, C.-M. Cheng, T. Chou, R. Niederhagen, A. Shamir, B.-Y. Yang: Fast exhaustive search for polynomial systems in F2. CHES 2010, LNCS vol. 6225, pp. 203 - 218. Springer, 2010.
4. R. Cartor, R. Gipson, D. Smith-Tone, J. Vates: On the Differential Security of the HFEv- Signature Primitive. PQCrypto 2016, LNCS vol. 9606, pp. 162 - 181. Springer, 2016.
5. J. Ding, B.Y. Yang: Degree of Regularity for HFEv and HFEv-. PQCrypto 2013, LNCS vol. 7932, pp. 52-66. Springer, 2013.
6. J.C. Faugére: A new efficient algorithm for computing Gröbner bases (F4). J. Pure Appl. Algebra 139, pp. 6188 (1999).
7. J.C. Faugére: Algebraic cryptanalysis of Hidden Field Equations (HFE) using Gröbner bases. CRYPTO 2003, LNCS vol. 2729, pp. 44 - 60. Springer, 2003.

8. M. R. Garey, D. S. Johnson: Computers and Intractability: A Guide to the Theory of NP-Completeness. W.H. Freeman and Company, 1979.
9. A. Kipnis, A. Shamir: Cryptanalysis of the HFE public key cryptosystem by Relinearization. CRYPTO 1999, LNCS vol. 1666, pp. 19 - 30. Springer, 1999.
10. M.S.E. Mohamed, J. Ding, J. Buchmann: Towards algebraic cryptanalysis of HFE challenge 2. ISA 2011, CCIS vol. 200, pp. 123131. Springer, 2011.
11. J. Patarin, N.T. Courtois, L. Goubin: QUARTZ, 128-bit long digital signatures. CT-RSA 2001, LNCS, vol. 2020, pp. 282 297. Springer, 2001.
12. A. Petzoldt, M.S. Chen, B.Y. Yang, C. Tao, J. Ding: Design principles for HFEv-based multivariate signature schemes. ASIACRYPT 2015 (Part 1), LNCS vol. 9742 , pp. 311 -334. Springer, 2015.
13. P. Schwabe, B. Westerbaan: Solving Binary MQ with Grover's Algorithm. SPACE 2016, LNCS vol. 10076, pp. 303 - 322. Springer 2016.
14. B.-Y. Yang, J.-M. Chen: Theoretical Analysis of XL over Small Fields. ACISP 2004, LNCS vol. 3108, pp.277-288. Springer 2004.