Solving Degree and Degree of Regularity for Polynomial Systems over a Finite Fields

Jintai Ding[1,⋆] and Dieter Schmidt[2]

[1] University of Cincinnati and Chongqing University
jintai.ding@gmail.com
[2] University of Cincinnati
schmiddr@ucmail.uc.edu

Abstract. In this paper, we try to clarify some of the questions related to a key concept in multivariate polynomial solving algorithm over a finite field: the degree of regularity. By the degree of regularity, here we refer to a concept first presented by Dubois and Gama, namely the lowest degree at which certain nontrivial degree drop of a polynomial system occurs. Currently, it is somehow commonly accepted that we can use this degree to estimate the complexity of solving a polynomial system, even though we do not have systematic empirical data or a theory to support such a claim. In this paper, we would like to clarify the situation with the help of experiments. We first define a concept of solving degree for a polynomial system. The key question we then need to clarify is the connection of solving degree and the degree of regularity with focus on quadratic systems. To exclude the cases that do not represent the general situation, we need to define when a system is degenerate and when it is irreducible. With extensive computer experiments, we show that the two concepts, the degree of regularity and the solving degree, are related for irreducible systems in the sense that the difference between the two degrees is indeed small, less than 3. But due to the limitation of our experiments, we speculate that this may not be the case for high degree cases.

Keywords: Solving degree, degree of regularity, HFE, HFEv, random polynomial system, non-degenerate system.

1 Introduction

1.1 Motivations

One way of attacking a symmetric or asymmetric cryptosystem is by solving a set of multivariate polynomial equations over a finite field. This is a rapidly developing area in cryptography. The security analysis of many cryptosystems is very much affected by the complexity to solve the related polynomial systems. Such an attack is studied most intensively in the context of multivariate public key cryptography, since here the public key is a set of multivariate polynomials.

⋆ Partially sponsored by National Science Foundation of China, Grant #60973131 and U1135004, and the Charles Phelps Taft Research Center.

M. Fischlin and S. Katzenbeisser (Eds.): Buchmann Festschrift, LNCS 8260, pp. 34–49, 2013.

The security of such a system depends directly on the complexity of solving the given polynomial equations. Therefore, understanding the complexity to solve multivariate polynomial systems is a critical problem in cryptography. In addition, since polynomial solving is used in many other places, the answer to this question has broad impact on other areas in theory and applications.

Since higher order polynomial equations can be transformed into a set of quadratic polynomial equations, the later can be considered to have the most general form in some sense. We thus focus on the quadratic polynomials

$$p_1(x_1, \ldots, x_n) = \cdots = p_m(x_1, \ldots, x_n) = 0,$$

over a finite field $\mathbb{F}_q$ of order q. We will concentrate on the cases with $m = n$, since these are the hard cases, which occur most frequently in applications.

To solve a multivariate polynomial system, the key method is the Gröbner basis algorithm. In general, solving a set of generic multivariate polynomial equations is a very hard problem. For instance, we know that to solve a set of randomly selected quadratic equations over a finite field with n equations and n variables is NP-complete. It is also known that the complexity of the Gröbner basis algorithms for a generic system is doubly exponential in the number of variables for a field of characteristic zero, as was shown in [15].

In the last two decades, polynomial solving algorithms have undergone a fast development. People realized that computationally (not in terms of storage or memory), the original Gröbner basis algorithm – Buchberger's algorithm – is very inefficient due to the need to perform multivariate polynomial reduction independently on each new S-pairs. The new trend is to improve the algorithm with a far better computational complexity but usually with a much larger storage or memory usage, namely a trade-off between computation cost and storage cost. This is achieved by transforming the polynomial solving process into several steps of solving linear systems. Here the linear systems are derived from the polynomial themselves directly by rewriting each polynomial as a row of a matrix. The reduction of the polynomials is then achieved by Gaussian elimination. The origin of the idea can be traced back to the original XL algorithm (later rediscovered as the XL Method) and was proposed by Lazard [2, 14]. The XL algorithm can be described in simplified terms as follows

1. multiply the equations with monomials to form a collection of relations up to some degree d;
2. linearize (i.e., treat each individual monomial as a variable), and use matrix algorithms (for example Gaussian elimination) over $\mathbb{F}_q$ on the resulting matrix (the *extended Macaulay matrix*).

In this paper, we assume that the Macauley matrix is in the form that rows represent monomials and columns represent polynomials, while the usual Macauley matrix is the other way around.

The newly developed algorithms include F_4, XL algorithm and Mutant XL algorithms [2, 5, 12, 16, 17]. For these algorithms it is clear that the computational complexity is dominated by the step, where performing the linear algebra

computation (Gaussian elimination) takes the longest time. On the other hand, if one is more concerned with memory complexity then the step dealing with the largest matrix will determine the space complexity. We call such a step the *solving step*.

The complexity of the solving step is determined by the form of the multivariate polynomials. The number of monomials determine the size of the rows of the corresponding Macauley matrix and the number of polynomials determines the number of columns of the matrix. The number of monomials is determined by the degree of the polynomial. The corresponding matrices we need to deal with for the solving step are in general almost square. Therefore, the complexity of the solving step is determined by the highest degree of the polynomials involved. We call this degree the *solving degree*.

The complexity analysis problem now becomes a problem of finding such a degree. It is clear that the concept of solving degree is very vague, and what we try to do is to find something more mathematically tangible. This leads us to the degree of regularity introduced by Dubois and Gama [11].

We first define the graded ring $B := \mathbb{F}_q[x_1, \ldots, x_n]/\langle x_1^q, \ldots, x_n^q \rangle$ and B_d its degree-d subspace. By B_d^m, we mean the vector space of direct product of m copies of B_d.

Definition 1. *For homogeneous quadratic polynomials* $(\lambda_1, \ldots, \lambda_m) \in B_2^m$, *let* $\psi_d : B_d^m \to B_{d+2}$ *be the map defined as*

$$\psi(b_1, \ldots, b_m) = \sum_{i=1}^{m} b_i \lambda_i.$$

Then

$$R_d(\lambda_1, \ldots, \lambda_m) := \ker \psi_d$$

defines the subspace of relations

$$\sum_{i=1}^{m} b_i \lambda_i = 0.$$

Further let $T_d(\lambda_1, \ldots, \lambda_n)$ *be the subspace of trivial relations generated by the elements*

$$\{b(\lambda_i e_j - \lambda_j e_i) \mid 1 \le i < j \le m,\, b \in B_{d-2}\},$$

and

$$\{b(\lambda_i^{q-1}) e_i \mid 1 \le i \le m,\, b \in B_{d-2(q-1)}\}.$$

Here e_i *means the* i*-th unit vector consisting of all zeros except* 1 *at the* i*-th position.*

$$e_i = (0, \ldots, 0, 1, 0, \ldots, 0).$$

The degree of regularity of a homogeneous quadratic set is then

$$D_{reg}(\lambda_1, \ldots, \lambda_m) := \min\{d \mid R_{d-2}(\lambda_1, \ldots, \lambda_m)/T_{d-2}(\lambda_1, \ldots, \lambda_m) \neq \{0\}\}.$$

For a generic polynomial (non-homogeneous) system $p_1 = \cdots = p_m = 0$,

$$D_{reg}(p_1, \ldots, p_m) := D_{reg}((p_1)^h, \ldots, (p_m)^h),$$

where $(p_i)^h$ *is the highest degree homogeneous component of* p_i.

Clearly the degree of regularity is the lowest degree at which we have a linear combination of multiples of p_i that has a nontrivial cancellation of all of the highest degree components, and therefore a nontrivial degree-drop.

In this definition, we can see that the subspace T_d of trivial syzygies represents a "known-to-be-useless" degree drop in the following sense:

Let

$$p_i = c^{(i)} + \sum_k b_k^{(i)} x_k + \sum_{k \le \ell} a_{k\ell}^{(i)} x_k x_\ell.$$

Let $(p)^h$ represent the homogeneous highest degree part of the polynomial p. Clearly $(p_j)^h (p_i)^h - (p_i)^h (p_j)^h = 0$ is a trivial syzygy, which is equivalent to the combination of degree-4 rows $\left(\sum_{k\ell} a_{k\ell}^{(i)} (x_k x_\ell p_j)\right) - \left(\sum_{k\ell} a_{k\ell}^{(j)} (x_k x_\ell p_i)\right)$ being of degree-3 (or fewer). Equally clearly this "degree-drop" will not give us anything useful since

$$\left(c^{(i)}(p_j) + \sum_k b_k^{(i)} (x_k p_j) + \sum_{k\ell} a_{k\ell}^{(i)} (x_k x_\ell p_j) \right)$$
$$= \left(c^{(j)}(p_i) + \sum_k b_k^{(j)} (x_k p_i) + \sum_{k\ell} a_{k\ell}^{(j)} (x_k x_\ell p_i) \right),$$

given that both give $(p_i p_j)$. Thus we just "found" a linear combination of polynomials we already have at degree 3. So a trivial or *principal* syzygy between the top-degree parts $(p_i)^h$ leads to a *trivial* degree drop useless for generating new equations. We must verify that a degree-drop is nontrivial before we can claim that we have reached the degree of regularity.

This concept is very mathematical (not computational) in the sense that the degree of regularity is invariant under invertible linear transformations in terms of either the variables and the polynomials.

This critical concept, *degree of regularity*, is actually the lowest degree where we find a *nontrivial* degree drop in terms of linear combinations of multiples of the original polynomials that define the system. By now, it is commonly accepted that this degree somehow in general matches the highest degree of polynomials we need to deal with in a polynomial solving algorithm, or in an abused term, the degree at which F_4, Mutant XL, and similar algorithms usually terminate. But we shall see later, this concept of termination degree is not really a good concept that sometimes it can be misleading. Therefore, we will use the new term the solving degree.

The mutant XL algorithm is an improved XL algorithm as follows: for a fixed degree d, multiply each p_i with all monomials of degree $d - \deg p_i$ to create a large collection of relations of degree d, order the monomials and linearize

these equations to obtain the Macaulay matrix $\text{Mac}^{(d)}(p_1, \ldots, p_m)$, then try to eliminate the highest degree monomials from $\text{Mac}^{(d)}(p_1, \ldots, p_m)$ to create new relations of degree $d-1$ or lower, and once we find such polynomials with degree drop, we try to use them fully before we move on to the next degree. These new polynomial created from nontrivial degree drop are called mutants. This idea allows us to greatly improve the XL algorithm to become one of the most efficient polynomial solving algorithms.

1.2 Questions about the Terminology

There is some confusion about the term "the degree of regularity". The rank of Macaulay matrices at any given degree, which describes the dimension related to the space of the XL algorithm can be computed with certain generating functions and the strong assumption that there are no nontrivial syzygies. A system where this assumption is valid for any degrees is called *regular*. However this can not happen in the case of a finite field. To deal with such a problem, there is a definition of "semi-regular" system [3]. The degree of regularity in such a setting, is the degree at which the system ceases to behave as if it is regular. The degree of regularity as described in Definition 1 is the degree at which the first appearance of "nontrivial degree drop" is observed, that is the system ceases to behave semi-regular.

A heuristic formula for the degree of regularity of most random systems (including asymptotics) is given by Bardet et al [1, 18]. However, this formula is not at all applicable to most systems with a structure that we are interested in.

Certain simple upper bounds to D_{reg} for the multivariate cryptosystems for HFE, HFE-, HFEv and HFEv- [4, 7, 8, 10] were found, and are shown to be good bounds to find the computational complexity.

1.3 The Contribution of This Paper

The question, we would like to clarify: Is it indeed true that the degree of regularity is a good concept to help us to determine the complexity to solve a given polynomial system?

We would like to first point out that this is not true for just any system. We will use an example of a triangular system to demonstrate this first. This leads to new definitions of degenerate systems and partially degenerate systems, and irreducible systems, and we would like to find out that if it is indeed true that the degree of regularity is a good concept to help us to determine the complexity to solve a given irreducible polynomial system.

We would like to show via experiments that the degree of regularity and the solving degree are closely related. Since we were not able to perform experiments on systems with a large number of variables, we are not sure what the relationship will be. We will also discuss briefly the applicability of the two degrees for higher degree systems.

2 A Degenerate System and an Irreducible System

We will first study the concept of degenerate systems.

2.1 An Example of a Degenerate System

We would like to first show an example, where the degree of regularity and solving degree have a big difference.

The constructions of this example is a type of triangular system. The example is a polynomial system, which looks like the following:

$$\begin{aligned} p_1(x_1, \ldots, x_n) &= f_1(x_1, x_2) \\ p_2(x_1, \ldots, x_n) &= f_2(x_1, x_2) \\ p_3(x_1, \ldots, x_n) &= f_3(x_1, \ldots, x_n) \\ &\cdots \\ p_n(x_1, \ldots, x_n) &= f_n(x_1, \ldots, x_n) \end{aligned}$$

The first two equations involves only the first 2 variables, and the rest are much more complicated polynomials. In this case, what a Gröbner basis solver will actually do is to try to solve the subsystem formed by f_1 and f_2 first, where a nontrivial degree drop will occur, and then try to solve the rest. Therefore in this case, the degree of regularity comes from the f_1 and f_2 system, but the solving degree actually comes from $f_3, \ldots, f_n$. We can expect a big difference, or as big a difference as we wish by manipulating the system.

Below we will give a concrete example for such a system. Before we present the example, we would like to say a few words about how we present the experimental results. We used both mutant XL algorithms and Magma implementation of the Gröbner basis algorithms for our experiments, but those mutant XL implementations are not yet publicly available. Therefore, to enable other researchers to check on our results, we will use only the data from the Magma implementation of the Gröbner basis algorithms and will not publish the data from the mutant XL algorithms, which matches well with the data from the Magma implementation of the Gröbner basis algorithms. In the Magma implementation, the algorithm goes through many steps of computations, and the key computation in each step is the Gaussian elimination which performs a reduction of the polynomials at a fixed degree. We call this degree the *step degree*. The degree of regularity is the first step degree at which the step degree starts to go either flat or down. We will present a graph, where the horizontal direction is the step number and the vertical direction is the step degree. Then we will add two vertical line segments to represent the relative metrics of the matrix size (left line) and the time (right line) to each step. In this way, we can read out easily the solving degree and the step which dominates the whole computation process.

We will show via computer experiments that the degree of regularity and solving degree can be far apart. We use an HFE system (whose definition is given in the section below) and then replace the first two equations by quadratic

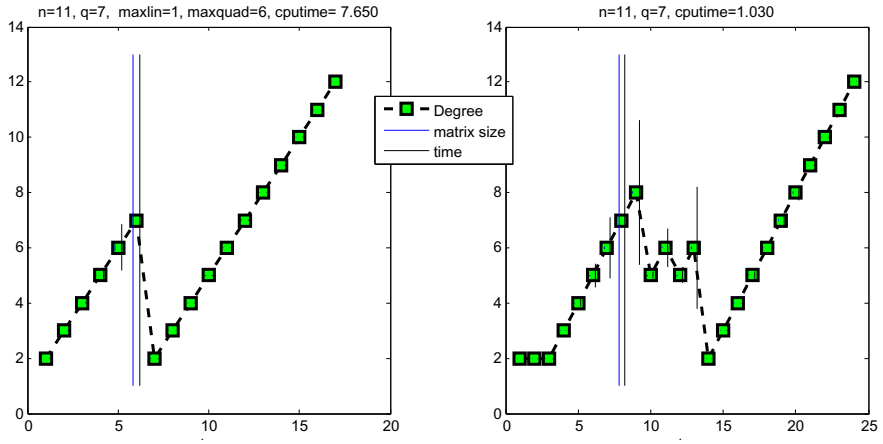

Fig. 1. The degrees at each step. The (relative) size of the matrix is given by the length of the left vertical line and the size of the line on the right gives the (relative) time for each step.
a) A HFE system over $GF(7)$ b) A system of same size but first two equations are replaced by quadratic equations in x_1 and x_2.

equations involving only the first two variables x_1, x_2. From Figure 1, we see that the degree of regularity for the HFE system is 7 and occurred in step 6. Most of the time was spent in this step and the largest matrix was encountered there, so that the solving degree is 7. When the first two equations were replaced by quadratic equations in x_1 and x_2 the degrees encountered by the Gröbner basis algorithm looks different. The degree of regularity is now 2 but the solving degree remains 7 and was encountered in step 8. The reason in the second case that the degree of regularity is low is exactly because the solver is actually first working on solving the system made of the first two equations.

We can actually create systems where the difference between the degree of regularity and the solving degree can be as large as we desire.

Also we know that for linear algebra based Gröbner basis algorithms the change of basis or the mixing of polynomials through invertible linear transformations does not change the degree of regularity and the solving degree. This leads to our definition of a degenerate system.

2.2 Definition of a Degenerate System

To simplify the exposition, we would like to deal with quadratic system where $m = n$ and where the solution space is of zero dimension.

Let us assume that we are dealing with a set of polynomials:

$$p_1(x_1, \ldots, x_n) = \cdots = p_n(x_1, \ldots, x_n) = 0.$$

We would like to first define a set of degenerate system.

Definition 2. *A quadratic system* $p_1(x_1, \ldots, x_n) = \cdots = p_n(x_1, \ldots, x_n) = 0$ *is degenerate, if we can find* m' *linearly independent polynomials* $h_i(x_1, \ldots, x_{n'})$, $i = 1, \ldots, m'$, $n' < n$ *and* $m' \geq n'$ *such that*

$$h_i(x_1, \ldots, x_{n'}) = (\sum a_{ij}(p_j)^h) \circ L(x_1, \ldots, x_n).$$

It is clear that the example above is an example of a degenerate system, where $m' = n' = 2$, while $n = 7$.

Now the question is: Is it true that for all non-degenerate systems the degree of regularity and the solving degree are close?

For this, we are actually not sure. The reason for this is that we are still not at all sure what happens for a system that is partially degenerate, namely what happens if we can find such a m' polynomials such that $m' < n'$. Since such a case is very complicated, what we would like to do is to concentrate on what we call an irreducible system.

Definition 3. *A quadratic system* $p_1(x_1, \ldots, x_n) = \cdots = p_n(x_1, \ldots, x_n) = 0$ *is an irreducible system, if we can not find a non-zero polynomial* $\sum a_{ij}(p_j)^h$*, such that the corresponding quadratic form is not of full rank (for the case, q=2, it should be full rank-2, when n is even).*

Here we would like to remark that for the case when n is odd and $q = 2$, the full rank can only be $n - 1$ not n. Another remark is that we do not know what happens in the case of the partially degenerate system and we speculate that the differences between the solving degree and the degree of regularity can be anything.

3 Solving Degree and Degree of Regularity

What we will do is to systematically perform testing on irreducible systems to check on the connection of degree of regularity and the solving degree. An easy way to construct an irreducible system is to use a random systems, whose coefficient are generated independently and uniformly. But how can we generate other type of irreducible systems? The trick here is that we will use HFE or a related cryptosystem from multivariate public key cryptography to construct irreducible systems that behave differently from random systems.

3.1 The HFE, HFEv, and IPHFE Cryptosystems

In the standard formulation of a multivariate public-key cryptosystem over a finite field $\mathbb{F}$, the public-key $P : \mathbb{F}^n \mapsto \mathbb{F}^m = T \circ Q \circ S$ is a composition of two invertible affine maps $S : \mathbb{F}^n \mapsto \mathbb{F}^n$ and $T : \mathbb{F}^m \mapsto \mathbb{F}^m$, and a quadratic map (possibly with some parameters) $Q : \mathbb{F}^n \mapsto \mathbb{F}^m$ which is easily invertible when all parameters are given. The maps S and T are part of the secret key, and properties of the central map Q determines most of the properties of the cryptosystem.

Let $\mathbb{F} \cong \mathbb{F}_q$ be a finite field of order q and $\mathbb{K}$ a degree-n extension of $\mathbb{F}$, with a "canonical" isomorphism ϕ identifying $\mathbb{K}$ with the vector space $\mathbb{F}^n$. That is, $\mathbb{F}^n \xrightarrow{\phi} \mathbb{K}$, $\mathbb{K} \xrightarrow{\phi^{-1}} \mathbb{F}^n$. Any function or map F from $\mathbb{K}$ to $\mathbb{K}$ can be expressed *uniquely* as a polynomial function with coefficients in $\mathbb{K}$ and degree less than q^n, namely

$$F(X) = \sum_{i=0}^{q^n-1} a_i X^i, \quad a_i \in \mathbb{K}.$$

Denote by $\deg_{\mathbb{K}}(F)$ the degree of $F(X)$ for any map F. Using ϕ, we can build a new map $F' : \mathbb{F}^n \to \mathbb{F}^n$

$$P(x_1, \ldots, x_n) = (p_1(x_1, \ldots, x_n), \ldots, p_n(x_1, \ldots, x_n)) = \phi^{-1} \circ F \circ \phi(x_1, \ldots, x_n),$$

which is essentially F but viewed from the perspective of $\mathbb{F}^n$. We will denote F' also by F unless there is a chance of confusion.

An $\mathbb{F}$-degree-2 or $\mathbb{F}$-quadratic function from $\mathbb{K}$ to $\mathbb{K}$ can in this framework be seen to be a polynomial all of whose monomials have exponent $q^i + q^j$ or q^i or 0 for some i and j. The general form of this $\mathbb{F}$-quadratic function is $Q(X) = \sum_{i,j=0}^{n-1} a_{ij} X^{q^i+q^j} + \sum_{i=0}^{n-1} b_i X^{q^i} + c.$, the *extended Dembowski-Ostrom polynomial map*. Such a $Q(X)$ with a fixed low $\mathbb{K}$-degree is used to build the HFE multivariate public key cryptosystems, as in the following

$$Q(X) = \sum_{i,j=0, j\le i}^{q^i+q^j \le D} a_{ij} X^{q^i+q^j} + \sum_{i=0}^{q^i \le D} b_i X^{q^i} + c;$$

Note that the coefficients are values in $\mathbb{K}$, and all coefficients $a_{ii} = 0$ if $q = 2$, since those are covered by the b-part of the coefficients.

For an overview of multivariate cryptosystems, including all the common modifiers such as "minus", "internal perturbation", and "vinegar" see [6,9]. It gives this formulation of HFEv, which uses the vinegar modification [13], built from the polynomial:

$$Q(X, \bar{X}) = \sum_{i,j} a_{ij} X^{q^i+q^j} + \sum_{i,j} b_{ij} X^{q^i} \bar{X}^{q^j} + \sum_{i,j} \alpha_{ij} \bar{X}^{q^i+q^j} + \sum_i b_i X^{q^i} + \sum_i \beta_i \bar{X}^{q^i} + c \tag{1}$$

where the auxiliary variable $\bar{X}$ occupies only a subspace of small rank v in $\mathbb{K} \cong \mathbb{F}^n$. The function Q is quadratic in the components of X and $\bar{X}$, and so is

$P = T \circ Q \circ S$ for affine bijections T and S in $\mathbb{F}^n$ and $\mathbb{F}^{n+v}$. We hope that P is hard to invert to the adversary, while the legitimate user, with the knowledge of (S, T) can compute X by substituting a random $\bar{X}$, then solving for X via root-finding algorithms such as Berlekamp (or Cantor-Zassenhaus, if $q \neq 2$). To limit the effort of Berlekamp, we restrict the maximum degree D of the polynomial.

Another closely related scheme to HFEv is IPHFE (internally perturbed HFE). Suppose in Eq. 1, $\bar{X}$ is not a free variable, but is instead the image of ℓ, a map from $\mathbb{F}^n$ onto $\mathbb{F}^v$. So the central map is really $Q'(X) := Q(X, \ell(X))$.

For our experiments, we will use a more generalized version of HFE, namely we allow the maximum degree of the quadratic part (the terms in the form of $X^{q^i+q^j}$) to be different from the linear part (the terms in the form of X^{q^i}). We call the highest degree of the quadratic part the quadratic degree and the highest degree of the linear part the linear degree of the HFE polynomial.

Here we would like to make one remark that we only look at systems with relative large n, since when n is small, special combinatorial identity could occur.

3.2 The Experimental Results

We will first present a few graphs of systems from many experiments we have done to give the reader a basic idea of what happens in the solving process. All computations were performed on a PC with the 64 bit version of Magma for Unix. The first example (Fig. 2) is an example, where the differences between the corresponding two degrees, the degree of regularity and the solving degree, are the same and occur at the same step.

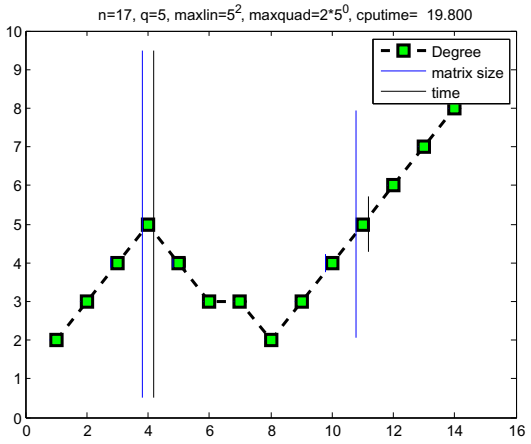

Fig. 2. This is an example of HFE with q=5, n=17, the quadratic degree is 2 and the linear degree 25. This is a case where both degrees are 5 and both occur at the step 4.

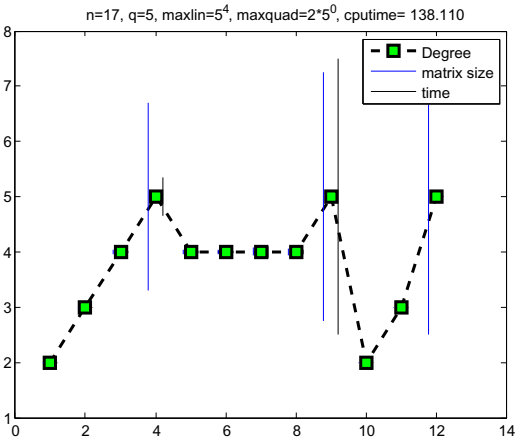

Fig. 3. This is an example of HFE with q=5, n=17, the quadratic degree is 2 and the linear degree to be 625. This is a case where the two degrees have the same value 5 but they occur at different steps.

But the degree of regularity and the solving degree can occur at very different steps despite the fact that they have the same values. Fig. 3 is such an example.

The degree of regularity and the solving degree can occur at very different steps and the difference now is 1. Fig. 4 is an example.

We have seen cases where the degree of regularity and the solving degree differ by 2, but we could not reproduce it for this paper. We are not sure if it was caused by the choice of the coefficients, which are selected at random, or by a programming or another error from our side.

Below are some of the tables from many we made for the degree of regularity and the solving degree. In the tables 'deg-reg' stands for the degree of regularity, 'deg-size' is the solving degree where the largest matrix size was encountered, and 'deg-time' the solving degree where the longest time was spent. The entry 'at step' gives the step where each occurred. In all cases displayed the difference is at most 1.

Table 1 was created by an HFE system with different values of n. The degree of regularity and the solving degree are always the same, but sometimes the largest matrix is encountered at a different step.

In Tables 2 and 3 the degree of the quadratic terms is fixed, but the degree of the linear terms are allowed to increase. Whereas the degree of regularity remains the same, the solving degree increases by 1 when the linear degree reaches a certain threshold. Table 4 shows that internal perturbation of an HFE system has no effect on the difference of the degrees.

For a quadratic system with the coefficients selected at random there will be some difference between the degrees as seen in Table 5 and it also occurs for other finite fields.

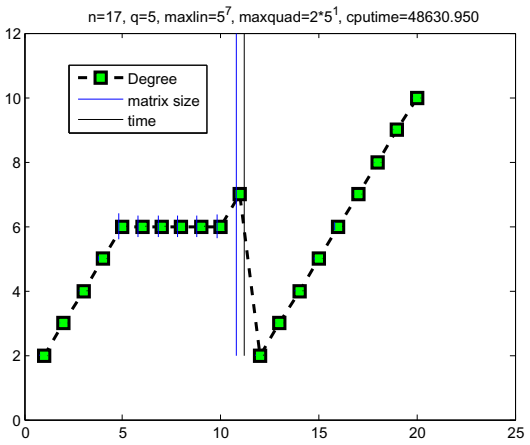

Fig. 4. This is an example of HFE with q=5, n=17 the quadratic degree is degree 10 and the linear degree is 78125. This is a case where the two degrees differ by 1 and are 6 and 7 respectively.

Table 1. HFE systems over $GF(3)$ with quadratic degree 6 and linear degree 9 for different values of n

n	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32
deg-reg	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4
at step	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
deg-size	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4
at step	3	3	3	3	3	3	3	7	3	8	8	3	8	8	7	3	7	3	8	3	8	7	8
deg-time	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4
at step	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3

Table 2. HFE systems of size $n = 17$ over $GF(5)$ with quadratic degree 2 and the linear degree 5^{r_1}

r_1	0	1	2	3	4	5	6	7	8	9	10	11	12
deg-reg	5	5	5	5	5	5	5	5	5	5	5	5	5
at step	4	4	4	4	4	4	4	4	4	4	4	4	4
deg-size	5	5	5	5	5	5	5	5	6	6	6	6	6
at step	4	11	4	4	12	4	4	4	9	8	8	8	8
deg-time	5	5	5	5	5	5	5	5	6	6	6	6	6
at step	4	4	4	4	9	8	8	9	9	8	8	8	8

Table 3. HFE systems of size $n = 17$ over $GF(3)$ with quadratic degree 6 and the linear degree 3^{r_1}

r_1	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
deg-reg	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	
at step	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	
deg-size	4	4	4	4	4	4	4	5	5	5	5	5	5	5	5	5	
at step	8	3	7	8	5	6	6	7	6	6	6	6	6	6	6	6	
deg-time	4	4	4	4	4	4	4	5	5	5	5	5	5	5	5	5	
at step	3	3	3	4	5	5	6	7	6	6	6	6	6	6	6	6	

Table 4. IPHFE system of size $n = 17$ over $GF(3)$ with v internal perturbation variables. The linear degree is 9 and the quadratic degree is 6.

v	0	1	2	3	4	5	6	7	8	9	10	11	12
deg-reg	4	5	6	6	7	7	7	7	7	7	7	7	7
at step	3	4	5	5	6	6	6	6	6	6	6	6	6
deg-size	4	5	6	6	7	7	7	7	7	7	7	7	7
at step	3	4	5	5	6	6	6	6	6	6	6	6	6
deg-time	4	5	6	6	7	7	7	7	7	7	7	7	7
at step	3	4	5	5	6	6	6	6	6	6	6	6	6

Table 5. Random quadratic system of size n over $GF(2)$

n	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29
deg-reg	3	3	3	3	4	4	4	4	4	4	4	5	5	5	5	5	5	5	5	6	6	6	6	6	6
at step	2	2	2	2	3	3	3	3	3	3	3	4	4	4	4	4	4	4	4	5	5	5	5	5	5
deg-size	3	3	3	3	4	4	4	4	4	4	4	5	5	5	5	5	5	5	5	6	6	6	6	6	6
at step	2	2	4	3	3	3	3	3	3	3	4	4	4	4	4	4	4	5	4	5	5	5	5	5	5
deg-time	3	3	3	3	4	4	4	4	4	4	4	5	5	5	5	5	5	5	5	6	6	6	6	6	6
at step	2	2	4	2	3	3	3	3	3	3	3	4	4	4	4	4	4	4	4	5	5	5	5	5	5

From all the experiments, we conclude that it seems that it is indeed true that the differences between the degree of regularity and the solving degree for irreducible systems are small. But again, we like to emphasize that the experiments we have done is relatively small in terms of number of variables, and therefore our experiments, though very systematic but are limited by our computing capacity. Some experiments indicate that the situation may not be true for large n. The reason is due to the experiments listed in Table 3 and illustrated in Figure 5.

In the two cases of Figure 5, the quadratic parts are exactly the same (therefore the degree of regularities remains the same), and only the linear parts are different. But in the second case, the linear part is more complicated. This shows that the linear part has a substantial impact. It increases the solving degree by 1 and not just for the case shown in the figure, but in all cases when the linear part had a degree $\geq 3^7$, see Table 3. Therefore, we believe we need more

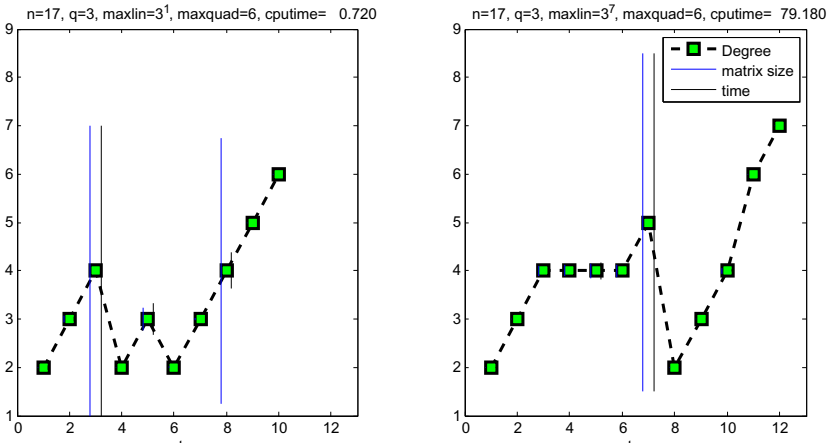

Fig. 5. This is an example of HFE with q=3, n=17, the quadratic degree is at most 6. On the left the linear terms have only degree 1, whereas on the right they are limited by 3^7. This a case where the solving degree can be increased by 1 due to additional linear terms.

experiments with larger n. It will not be a surprise if we find the differences to be bigger than 1 when n is large enough. Therefore we speculate the differences of the two degrees could be dependent on n. For this, we need much more powerful computers to do the experiments, which are now beyond our reach.

3.3 Higher Degree Cases

We performed some examples with higher degree polynomials, in particular, degree 3 polynomials with random coefficients in $GF(2)$ or in $GF(3)$. The overall impression is that in these cases the degree of regularity and the solving degree are the same and occur at the same step. Modifying the equations we have seen examples where the difference was greater than 1. For this case, we need more studies to come to a reasonable conclusion.

4 Conclusion and Discussion

From the experiments, we conclude that indeed for an irreducible quadratic system the difference between the degree of regularity and the solving degree is small. But our experiments are preliminary so far and are limited since they were run on a personal computer with a 64 bit Unix system. The results in Fig. 5 however force us to suspect that maybe the difference between the two degrees can become bigger due to the influence of the linear part. The next step would require to find some way to prove the claim, but for this we need to set up reasonable additional assumptions. This would be a big breakthrough in terms

of understanding what really is going on with the complexity to solve polynomial systems.

Overall, we believe that the speculation about the connection between the solving degree and the degree of regularity works in the case of degree 2 irreducible polynomial systems with rather limited number of variables, but for high degree cases, it could be different. Therefore, much more work is still needed to be done to understand the complexity of polynomial solving algorithms.

Acknowledgment. We would like to thank Albrecht Petzoldt for useful comments.

J.D. first met Johannes Buchmann in 2004 in Japan. Since then, his work and his life are very much impacted and inspired by the influences of Johannes, which include a year in Darmstadt as a Humboldt Fellow, a meditation trip to the Plum Village and a trip to Tibet. J. D. would like to express his deep appreciation as a colleague and as a friend. Happy 60th Birthday and Namaste.

References

1. Bardet, M., Faugère, J.-C., Salvy, B., Yang, B.-Y.: Asymptotic expansion of the degree of regularity for semi-regular systems of equations. In: Gianni, P. (ed.) MEGA 2005, Sardinia, Italy (2005)
2. Courtois, N.T., Klimov, A., Patarin, J., Shamir, A.: Efficient algorithms for solving overdefined systems of multivariate polynomial equations. In: Preneel, B. (ed.) EUROCRYPT 2000. LNCS, vol. 1807, pp. 392–407. Springer, Heidelberg (2000), `http://www.minrank.org/xlfull.pdf`
3. Diem, C.: The XL-algorithm and a conjecture from commutative algebra. In: Lee, P.J. (ed.) ASIACRYPT 2004. LNCS, vol. 3329, pp. 323–337. Springer, Heidelberg (2004)
4. Ding, J.: Inverting the square systems is exponential. Cryptology ePrint Archive, Report 2011/275 (2011), `http://eprint.iacr.org/`
5. Ding, J., Buchmann, J., Mohamed, M.S.E., Mohamed, W.S.A.E., Weinmann, R.-P.: Mutant XL. In: Talk at the First International Conference on Symbolic Computation and Cryptography (SCC 2008), Beijing (2008)
6. Ding, J., Gower, J., Schmidt, D.: Multivariate Public-Key Cryptosystems. In: Advances in Information Security. Springer (2006) ISBN 0-387-32229-9
7. Ding, J., Hodges, T.J.: Inverting hfe systems is quasi-polynomial for all fields. In: Rogaway, P. (ed.) CRYPTO 2011. LNCS, vol. 6841, pp. 724–742. Springer, Heidelberg (2011)
8. Ding, J., Kleinjung, T.: Degree of regularity for HFE−. Journal of Math-for-Industry 4(2012B-3), 97–104 (2012), `http://eprint.iacr.org/`
9. Ding, J., Yang, B.-Y.: Post-Quantum Cryptography. Springer, Berlin (2009) ISBN: 978-3-540-88701-0, e-ISBN: 978-3-540-88702-7
10. Ding, J., Yang, B.-Y.: Degree of regularity for hfev and hfev-. In: Gaborit, P. (ed.) PQCrypto 2013. LNCS, vol. 7932, pp. 52–66. Springer, Heidelberg (2013)
11. Dubois, V., Gama, N.: The degree of regularity of hfe systems. In: Abe, M. (ed.) ASIACRYPT 2010. LNCS, vol. 6477, pp. 557–576. Springer, Heidelberg (2010)
12. Faugère, J.-C.: A new efficient algorithm for computing Gröbner bases (F_4). Journal of Pure and Applied Algebra 139, 61–88 (1999)

13. Kipnis, A., Patarin, J., Goubin, L.: Unbalanced Oil and Vinegar signature schemes. In: Stern, J. (ed.) EUROCRYPT 1999. LNCS, vol. 1592, pp. 206–222. Springer, Heidelberg (1999)
14. Lazard, D.: Gröbner-bases, Gaussian elimination and resolution of systems of algebraic equations. In: ISSAC 1983 and EUROCAL 1983. LNCS, vol. 162, pp. 146–156. Springer (March 1983)
15. Mayr, E.W., Meyer, A.: The complexity of the word problems for commutative semigroups and polynomial ideals. Adv. in Math. 46(3), 305–329 (1982)
16. Mohamed, M.S.E., Cabarcas, D., Ding, J., Buchmann, J., Bulygin, S.: MXL_3: An efficient algorithm for computing Gröbner bases of zero-dimensional ideals. In: Lee, D., Hong, S. (eds.) ICISC 2009. LNCS, vol. 5984, pp. 87–100. Springer, Heidelberg (2010)
17. Mohamed, M.S.E., Mohamed, W.S.A.E., Ding, J., Buchmann, J.: MXL2: Solving polynomial equations over GF(2) using an improved mutant strategy. In J. Buchmann and J. Ding, editors, *PQCrypto*. In: Buchmann, J., Ding, J. (eds.) PQCrypto 2008. LNCS, vol. 5299, pp. 203–215. Springer, Heidelberg (2008)
18. Yang, B.-Y., Chen, J.-M.: Theoretical analysis of XL over small fields. In: Wang, H., Pieprzyk, J., Varadharajan, V. (eds.) ACISP 2004. LNCS, vol. 3108, pp. 277–288. Springer, Heidelberg (2004)