# Comparison of XL and Gröbner basis algorithms over Finite Fields

Jean-Charles Faugère, Gwénolé Ars

INRIA

INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

# *Comparison of XL and Gröbner basis algorithms over Finite Fields*

Jean-Charles Faugère

— Gwénolé Ars

**N° 5251**

Juillet 2004

THÈME 2


ISSN 0249-6399 ISRN INRIA/RR--5251--FR+ENG

INRIA
ROCQUENCOURT

# Comparison of XL and Gröbner basis algorithms over Finite Fields

Jean-Charles Faugère*†
, Gwénolé Ars ‡

Thème 2 — Génie logiciel
et calcul symbolique
Projets SALSA

Rapport de recherche n° 5251 — Juillet 2004 — 26 pages

**Abstract:** This paper compares the XL algorithm with Gröbner basis algorithm. We explain the link between XL computation result and Gröbner basis with the well-known notion of $D$-Gröbner basis. Then we compare these algorithms in two cases: in the fields $\mathbb{F}_2$ and $\mathbb{F}_q$ with $q \gg n$. For the field $\mathbb{F}_2$, we have proved that if XL needs to compute polynomial with degree $D$ to terminate, the whole Gröbner basis is computated without exceeding the degree $D$. We have studied the XL algorithm and $F_5$ algorithm on semi-regular sequences introduced in report [1]. We show that the size of matrix constructed by XL is huge compared to the ones of $F_5$. So the complexity of XL is worth than $F_5$ algorithm on these systems. For the field $\mathbb{F}_q$, we introduce an emulated algorithm using Gröbner basis computation to have a comparison between XL and Gröbner basis. We have proved that this algorithm will always reach a lower degree for intermediate polynomials than XL algorithm. A study on semi-regular sequences shows that $F_5$ always has a better behavior than XL algorithm especially when $m$ is near from $n$.

**Key-words:** XL algorithm, Hidden Field Equations (HFE), Multivariate polynomial equations, Gröbner bases, Algebraic Cryptanalysis, Computer Algebra, Semi-regular Sequences.

* LIP6/LORIA CNRS/UPMC/INRIA
† Projet SALSA
‡ LIP6 DGA/UPMC/Université Rennes 1

Unité de recherche INRIA Rocquencourt
Domaine de Voluceau, Rocquencourt, BP 105, 78153 Le Chesnay Cedex (France)
Téléphone : +33 1 39 63 55 11 — Télécopie : +33 1 39 63 53 30

# Comparaison de l'algorithme XL et ceux des bases de Gröbner sur les corps finis

**Résumé :** Ce papier compare l'algorithm XL aux algorithmes de calcul de base de Gröbner. Nous expliquons le lien entre les résultats obtenus par XL et les bases de Gröbner à partir de la notion bien-connue de $D$-base de Gröbner. Puis nous comparons ces algorithmes dans deux cas : sur le corps $\mathbb{F}_2$ et dans le corps $\mathbb{F}_q$ with $q \gg n$. Pour ke corps $\mathbb{F}_2$, nous avons prouver que si XL a besoin du paramètre $D$ pour terminer, alors un base de Gröbner complète sera déterminé sans dépasser le degré $D$ pour les calculs. Nous avons étudié l'algorithme $XL$ et l'algorithme $F_5$ sur les séquences semi-régulières introduites dans le rapport INRIA [1]. Nous avons prouvé que la taille des matrices construites par XL est plus grande que celles de $F_5$. Par conséquent, la complexité de XL est plus mauvaise que celle de $F_5$ sur ces systèmes. Pour le corps $\mathbb{F}_q$, nous avons introduit un algorithme légèrement modifié de base de Gröbner pour permettre la comparaison. Nous avons prouvé que cet algorithme atteint toujours un plus bas degré pour les polynomes intermédiaires que l'algorithme XL. Une étude sur les séquences semi-réguliaires a montré que $F_5$ avait toujours un meilleur comportement que l'algorithme XL spécialement lorsque le système a à peine plus d'équations que de variables.

**Mots-clés :** Algorithme XL, Polynômes multivariés, Bases de Gröbner, Cryptanalyse algébrique, Calcul Formel, séquences semi-régulières.

# Introduction

Several cryptosystems can be boiled down to solving a multivariate equations system on a finite field as HFE [15] or nonlinear filter generators [6]. At Eurocrypt 2000, Courtois, Klimov, Patarin and Shamir have introduced XL algorithm to attack these cryptosystems with linear algebra methods. Severals papers study the algorithm [7, 16, 10, 17].

With a system $f_1 = 0, \ldots, f_m = 0$, where $f_i \in \mathbb{F}_q[x_1, \ldots, x_n]$, and a parameter $D$, XL algorithm tries to find a basis of the $\mathbb{F}_q$ vector space $E_D = \{Pf_i | \forall\ i \in \{1, \ldots, m\}$ and $P \in \mathbb{F}_q[x_1, \ldots, x_n]$ st $deg(Pf_i) \leq\ D\}$ so that an univariate polynomial belongs to it. All this operations are done with linear algebra. Then XL algorithm solves this polynomial in the finite field, substitutes the solution in the initial system and repeats the process. At the end, XL algorithm finds a solution of the system.

Gröbner bases of an ideal can be used to simplify solving systems. A Gröbner basis is a new family of polynomials defining the same system with some properties. For that, we need to introduce a monomial ordering. Different algorithms compute this basis : the first algorithm is the Buchberger algorithm [9], this algorithm chooses a pair of elements of a generator family of the ideal, considered a special polynomial of this pair (the $\mathcal{S}$-polynomials) and reduces it by the family. If this reduction is not zero, we add this reduction to the family and repeat the process. For this algorithm, most of the time is spent in computing zero. B. Buchberger introduces criteria to avoid some computation to zero. D. Lazard presented in 83 [13] another method to compute Gröbner basis with linear algebra: he considered the $\mathbb{F}_q$ vector space $E_D$ defined above and computed Gaussian Elimination on a linear basis of $E_D$ to find the Gröbner basis for a $D$ large enough. The last algorithm presented is the $F_5$ algorithm, this algorithm has a more powerful criterion in order not to spend 90% of the time computing zero. This algorithm has no reduction to zero for random systems. We present a matrix version of this algorithm.

It is difficult to compare XL computation result and a Gröbner basis. First XL algorithm does not introduce monomial ordering. But we have proved that if XL algorithm terminates, it will also terminate with a lexicographic ordering. Moreover as the algorithm XL has the parameter $D$, the well-known notion of $D$-Gröbner basis allows to understand the link between Gröbner basis and XL. So we find that XL computes the dehomogenization of the $D$-Gröbner basis of the homogenized starting system.

Solving systems with solutions in the finite field $\mathbb{F}_q$ introduces new polynomials with degree $q$ in the systems called the field equations $(X^q - X)$. Therefore, if $q$ is very high, these relations will not be used. So we can have two different behaviors in finite fields, little finite field as $F_2$ for which the use of the field equations is very important and finite field $F_q$ with $q$ very large. And we have to distinguish both cases.

For the finite field $F_2$, we have proved that for a system of polynomials, if the XL algorithm finds a solution with a parameter value $D$, then the whole Gröbner basis with Buchberger algorithm is computed without exceeding the degree $D$ for the intermediate polynomials. Moreover, we study the XL algorithm on semi-regular sequences introduced by the report [1]. A conjecture is that a "random" system is a semi-regular sequence. For a semi-regular sequence of $m$ equations of degree $d$ on $n$ variables, it seems that the degree $D$ of

the parameter needed for XL algorithm is almost the same as the degree of the polynomials in the matrix constructed by $F_5$ algorithm. But the complexity of these two algorithms is specified by the size of the matrix: XL algorithm creates matrices with $\sum_{i=1}^{m} \binom{n}{D-deg(f_i)}$ rows and $\binom{n}{D}$ columns, whereas $F_5$ creates square matrices with $\binom{n}{D}$ columns. We complete this study on generic systems with a comparison of XL algorithm and Buchberger algorithm for a cryptosystem HFE. For this cryptosystem, Gröbner basis algorithm finds a structure in the multivariate systems and never exceeds a low degree, whereas, for the XL algorithm, the degree seems to still increase with the number of variables $n$.

The second field studied is the field $\mathbb{F}_q$, with $q$ very large compared to $n$. In this field, we do not take into account the field equations. As XL does not compute a complete Gröbner basis, we can describe an emulated algorithm of XL using Gröbner basis computation. With this algorithm, we prove that we always reach a lower degree than XL. As in $\mathbb{F}_2$, we introduce the notion of semi-regular sequences. For these systems, the univariate polynomial will be computed at last, we show XL algorithm terminates for a degree higher than Gröbner basis algorithms with a DRL order. Moreover, for $n = m$, the univariate polynomial will have as degree the Bezout bound, which is $\prod_{i=1}^{m} deg(f_i)$. This study shows that the algorithms based on Gröbner basis computation with DRL order always have a better behavior than XL algorithm especially when $m$ is near from $n$ on generic systems. And on a cryptosystem based on HFE, we experimentally find a better behavior of Buchberger algorithm compared to XL algorithm.

In the section 1, we present how to solve multivariate systems on finite field and introduce Gröbner basis. Next,in the section 2, we present some algorithms for Gröbner basis computation. The descriptions of $F_5$ algorithm and Semi-regular-sequences are done in section 3 and we introduce the XL algorithm in section 4 with first remarks on this algorithm. Section 5 explains the link between the XL computation and Gröbner basis computation. And then, we give a theoretical and experimental comparison on the degree on the specific field $\mathbb{F}_2$ in the section 6 then on the field $\mathbb{F}_q$ in section 7.

# 1 Solving systems on finite field $\mathbb{F}_q$

In this section, we explain how to solve systems of multivariate equations on finite fields.

Let $\mathcal{A}_m^n$ be a system of $m$ multivariate equations on $\mathbb{F}_q$ $\begin{cases} f_1(X_1, \ldots, X_n) &= 0 \\ \vdots & \\ f_m(X_1, \ldots, X_n) &= 0 \end{cases}$

## 1.1 Systems and Ideals

Let us denote by $\mathcal{S}_{q,n}$ the ring of polynomials $\mathbb{F}_q[X_1, \ldots, X_n]$.

It is important to observe that the set of equations $\mathcal{A}_m^n$ is equivalent to any other set $g_1(X_1, \ldots, X_n) = \cdots = g_k(X_1, \ldots, X_n) = 0$ for which $\{g_1, \ldots, g_k\}$ generates the same ideal $I$ as $\{f_1, \ldots, f_m\}$. So solving multivariate equations can be done by giving a new family that generates the same ideal so that this family satisfies special conditions. But solving

systems with ideal give us all the solutions of the system in the algebraic closure of $\mathbb{F}_q$. For cryptographic applications, we only want solutions in the field $\mathbb{F}_q$, and not in the algebraic closure. As $\mathbb{F}_q$ is a finite field, we have simple algebraic relations on the variables $X_i$ which describe exactly the elements of $\mathbb{F}_q^n$. These relations are the field equations :

**Definition 1** The field equations *in* $\mathcal{S}_{q,n}$ *are* $\{X_1^q - X_1, \ldots, X_n^q - X_n\}$.

So, in order to limit the solution in $\mathbb{F}_q$, we do not study the ideal $\langle f_1, \ldots, f_m \rangle$ in $\mathcal{S}_n$ but the ideal $\langle f_1, \ldots, f_m, X_1^q - X_1, \ldots, X_n^q - X_n \rangle$. To simplify the notation, we denote $\mathcal{R}_{q,n}$ the ring $\mathbb{F}_q[X_1, \ldots, X_n]/\langle X_1^q - X_1, \ldots, X_n^q - X_n \rangle$. Then we only consider an ideal $\mathcal{I} = \langle f_1, \ldots, f_m \rangle$ in $\mathcal{R}_{q,n}$ corresponding to the ideal $\langle f_1, \ldots, f_m, X_1^q - X_1, \ldots, X_n^q - X_n \rangle$ in $\mathcal{S}_{q,n}$.

In practice, if the cardinal $q$ of $F_q$ is very large compared to $n$, we do not use the field equations for solving systems. We prefer to compute new generators, as Gröbner bases, of $\langle f_1, \ldots, f_m \rangle$ in $\mathcal{S}_{q,n}$ because this computation stops on a degree $D$ lower than $q$.

So we have distinguished two importante cases: the field $\mathbb{F}_2$(in section 6) where the field equations are very important and the field $\mathbb{F}_q$(in section 7) with $q \gg n$.

## 1.2 Gröbner bases

We refer to [9] for details on Gröbner bases.

With univariate polynomials, finding a new generator of an ideal spanned by $\langle f_1, \ldots, f_m \rangle$ is done with the Euclidean Algorithm. To generalize the Euclidean Algorithm on multivariate polynomials, we need to define a monomial ordering:

**Definition 2** *A* monomial ordering *on* $\mathcal{S}_{q,n}$ *is a relation* $\prec$ *on the set of monomials* $X^\alpha$, $\alpha = (\alpha_1, \ldots, \alpha_n) \in \mathbb{Z}^n$, *so that* $\prec$ *is a total ordering, conserves the product relation ( if* $X^\alpha \prec X^\beta$ *and* $\gamma \in \mathbb{Z}^n$ *then* $X^{\alpha+\gamma} \prec X^{\beta+\gamma}$*), and is a well-ordering (every nonempty subset of the monomial set has a smallest element).*

**Example 1** Let $X^\alpha$ and $X^\beta$ be two monomials.

- The Lexicographic order: We say $X^\alpha > X^\beta$ if $\exists i \in \{1, \ldots, n\}$, *st* $\alpha_i > \beta_i$ and $\forall j < i$, $\alpha_j = \beta_j$.
- The DRL order: We say $X^\alpha \succ X^\beta$ if $|\alpha| = \sum_{i=1}^n \alpha_i > |\beta| = \sum_{i=1}^n \beta_i$ or $|\alpha| = |\beta|$ and $\exists i \in \{1, \ldots, n\}$ s.t. $\alpha_i < \beta_i$ and $\forall i < j$, $\alpha_j = \beta_j$.
- The elimination order $[X_1, \ldots, X_k]$, $[X_{k+1}, \ldots, X_n]$: We say that $m = m_x\, m_y \rhd m' = m'_x\, m'_y$ if and only if $m_x \succ m'_x$ or ($m_x = m'_x$ and $m_y \succ m'_y$) with $m_x$ (resp. $m_y$) depends only on $X_i$, $1 \leq i \leq k$ (resp $x_j$, $k < i \leq n$).

Let us introduce some definitions:

**Definition 3** *Let us fix a monomial order and let* $f, g \in \mathcal{R}_{q,n}$ *be nonzero polynomials.*

- *$LT(f)$ is the leading term of $f$, $LM(f)$ the leading monomial of $f$ and $LC(f) = \frac{LT(f)}{LM(f)}$ the leading coefficient of $f$.*
- *A finite subset $G = \{g_1, \ldots, g_s\}$ of an ideal $\mathcal{I}$ is said to be a* Gröbner basis *if $\forall f \in \mathcal{I}$, $\exists i \in \{1, \ldots, s\}$ such that $LM(g_i)$ divides $LM(f)$.*
- *A* reduced basis *for a polynomial ideal $\mathcal{I}$ is a basis $F$ for $\mathcal{I}$ such that:*
  *(i) $LC(p) = 1$ for all $p \in F$.*
  *(ii) For all $p \in F$, and $m$ monomial of $p$, $\nexists\ f \in F$, st $LM(f)$ divides $m$.*
- *The $\mathcal{S}$-*polynomial *of $f$ and $g$ is $\mathcal{S}pol(f,g) = \frac{Lcm(LM(f),LM(g))}{LT(f)}.f - \frac{Lcm(LM(f),LM(g))}{LT(g)}.g$*

We don't have the uniqueness of the Gröbner basis. In fact, if $G$ is a Gröbner basis of $I$ and $f \in \mathcal{I}$, $G \cup \{f\}$ is also a Gröbner basis. We recall some basic properties of Gröbner basis in $\mathcal{R}_{q,n}$.

**Proposition 1** *Let $\mathcal{I} \neq 0$ be an ideal of $\mathcal{R}_{q,n}$ and $\prec$ a monomial ordering.*

- *There is a Gröbner basis $G$ of $\mathcal{I}$, and $\langle G \rangle = \mathcal{I}$.*
- *$\mathcal{I}$ has a unique reduced Gröbner basis $G$.*
- *$G = \{X_0 + a_0, \ldots, X_{L-1} + a_{L-1}\}$ if $(a_0, \ldots, a_{L-1})$ is the only solution of the system $\mathcal{A}_m^n$.*
- *$G = \{1\}$ if there is no solution in $\mathbb{F}_q^n$.*
- *A basis $G = \{g_1, \ldots, g_s\}$ for $I$ is a Gröbner basis for $I$ if and only if for all pairs $i \neq j$, the reduction of $\mathcal{S}pol(g_i, g_j)$ by $G$ is zero.*
- *Let $G$ a Gröbner basis of $I$ for the Elimination order $[X_1, \ldots, X_k]$, $[X_{k+1}, \ldots, X_n]$ (or the Lexicographic order). Then $G \cap \mathbb{F}_q[X_{k+1}, \ldots, X_n]$ is a Gröbner basis of $I \cap \mathbb{F}_q[X_{k+1}, \ldots, X_n]$.*

The last proposition explains how to compute polynomials depending on severals variables $X_{k+1}, \ldots, X_n$ with Gröbner basis. As we have now the notion of Gröbner bases, we will see in the next section different algorithms how to compute it.

# 2 Gröbner basis algorithms

Before presenting the XL algorithm, we will initially carry out some recalls on the computation of Gröbner's bases by the Buchberger algorithm [3, 4]. For all proofs and further details, we refer to [9].

## 2.1 The Buchberger algorithm

The Buchberger Algorithm is a "generalization" of the Euclidean Algorithm to the multivariate case, where the $\mathcal{S}$-polynomial is used to match the leading monomial of two polynomials from $F = \{f_1, \ldots, f_m\}$. If the reduction of the result is nonzero, then we add it to the equations of $F$.

**Buchberger's Algorithm:**

**Input**: $\begin{cases} F = (f_1, \ldots, f_m) \text{ list of polynomials} \\ < \text{an order} \end{cases}$

**Output**: $G$ a Gröbner basis for $\mathcal{I}$ with $F \subset G$.

1. $G = F$. Find all the possible pairs of polynomials of $G$.
2. Choose a pair $(f, g)$ and reduce the $\mathcal{S}pol(f, g)$ by the family $G$. If the reduction is a non zero polynomial, add this polynomial to $G$ and consider the new possible pairs defined by it.
3. repeat the previous point until we have a possible pair of polynomials.

The selection of the pair $(f_i, f_j)$ is very important for the efficiency. A classical strategy is to choose a pair $(f_i, f_j)$ whose $Lcm(f_i, f_j)$ is minimal.

Futhermore the most expensive operation in the algorithm is the reduction of the $\mathcal{S}$-polynomials modulo $F$. Fortunately, B. Buchberger [4] developed two criteria for detecting 0-reductions a priori.

**Buchberger's Criterion 1:** Let $p, q \in \mathcal{R}_{q,n}$. If $Lcm(LM(p), LM(q)) = LM(p)LM(q)$ then $\mathcal{S}pol(p, q)$ will be reduce to zero by $\{p, q\}$.

**Buchberger's Criterion 2:** If, when considering the pair $\{f_i, f_j\}$, an element $f_k$ of the basis $G$ different from $f_i$ and $f_j$ exists such that $Lcm(LM(f_i), LM(f_j))$ is a multiple of $LM(f_k)$ and $\mathcal{S}pol(f_i, f_k)$ and $\mathcal{S}pol(f_j, f_k)$ having already been considered, then $\mathcal{S}pol(f_i, f_j)$ will reduce to zero by the elements already computed.

These two criteria take advantage of being local (e.g. one needs to examine the leading terms) but they are not perfect: most of $\mathcal{S}$-polynomials still reduce to zero. For more details, refer to [9]

## 2.2 Gröbner bases and Gaussian Elimination

D. Lazard in the articles [13] and [12] gave a relationship between the method of the computation of Gröbner bases and the one based on Gaussian Eliminations in some matrices for the system $\mathcal{A}_m^n$.

The matrices considered are the classic matrices named Macaulay [14]. Let us consider a monomial order $\prec$ and an integer $D$, the sets of polynomials $\mathcal{B}_{i,D}$ a subvector space of $\{P \in \mathcal{R}_{q,n} \,|\, deg(P) \leq deg(f_i)\}$, for $i \in \{1, \ldots, m\}$, and $\mathcal{B}_D$ a subvector space of $\{P \in \mathcal{R}_{q,n} \,|\, deg(P) \leq D\}$ .

So we can consider the following linear application :

$$f_{D,m}^{\text{mac}} : \begin{array}{ccc} \mathcal{B}_{1,D} \times \cdots \times \mathcal{B}_{m,D} & \longrightarrow & \mathcal{B}_D \\ (g_1, \ldots g_m) & \longmapsto & g = \sum_{i=1}^{m} g_i f_i, \end{array}$$

The Macaulay matrix $\mathcal{M}_{D,m}^{\text{acaulay}}$ is precisely the transposed matrix of such a transformation on the basis describe above. If we want to compute a Gröbner basis for an order $\prec$, we choose $\mathcal{B}_{i,D}$ and $\mathcal{B}_D$ the linear vector spaces spanned respectively by $\{m_\ell$ monomial of $\mathcal{R}_{q,n} \,|\, deg(m_\ell) \leq D - deg(f_i)\}$ and $\{m_\ell$ monomial of $\mathcal{R}_{q,n} \,|\, deg(m_\ell) \leq D\}$. And we need to sort the linear basis of $\mathcal{B}_D$ with the monomial ordering. Then a Gaussian elimination corresponds to a reduction for the monomial ordering.

In other words, this matrix can be constructed as follows : write down horizontally all the monomials with degree lower than $D$, ordered by the monomial order $\prec$ (the first one is the largest one). Hence each column is indexed by a monomial of degree at most $D$. Multiply each $f_i$, $i \in \{1, \ldots, m\}$ by any monomial $m_\ell$ with a degree lower than $D - deg(f_i)$, and write the coefficients of $m_\ell f_i$ in the columns of their corresponding monomial, thus define a row of the matrix which can be indexed by $m_\ell f_i$.

$$\mathcal{M}_{D,m}^{\text{acaulay}} = \begin{array}{c} \\ m_1' \times f_{i_1} \\ m_2' \times f_{i_2} \\ m_3' \times f_{i_3} \\ \cdots \end{array} \begin{array}{c} m_1 \succ m_2 \succ m_3 \ldots \succ m_{L-2} \succ m_{L-1} \succ m_L \\ \left( \begin{array}{c} \cdots \\ \cdots \\ \cdots \\ \cdots \end{array} \right. \end{array}$$

We are only interested in the image of $f_{D,m}^{\text{mac}}$, note $Im(f_{D,m}^{\text{mac}})$. So we compute a Gaussian elimination on the the rows of the matrix $\mathcal{M}_{D,m}^{\text{acaulay}}$ to find a linear basis of $Im(f_{D,m}^{\text{mac}})$.

For the latter, we must define a bound on the degree $D$ so that the linear basis of $Im(f_{D,m}^{\text{mac}})$ is also a Gröbner basis of the ideal. This problem is summarized with the following theorem:

**Theorem 1** *Let $\mathcal{I}$ be an ideal of $\mathcal{R}_{q,n}$ generated by polynomials $f_1, \ldots, f_n$ of degrees resp. $d_1, \ldots, d_n$. Assume the ideal $I$ is zero-dimensional, i.e., ideals for which the set of solutions of the systems is finite.*

- *Gröbner base for a Lexicographic order is computed with $D \leq d_1 d_2 \ldots d_n$.*

- *Gröbner base for a DRL order is computed with* $D \leq d_1 + d_2 \cdots + d_n - n + 1$.

# 3 $F_5$ algorithm and semi-regular sequences

In this section, we will present the algorithm $F_5$ [11] and particularly $F_5$ in a matrix fashion [1]. The algorithm matrix-$F_5$ constructs matrices incrementally in the degree and the number of polynomials, and uses linear algebra. For this section, we will use the notation introduced in section 2.2.

## 3.1 Explanation

We have seen in section 2.2, that we are only interested in $Im(f_{D,m}^{\text{mac}})$ for $\mathcal{B}_{i,D}$ and $\mathcal{B}_D$ the linear vector spaces spanned respectively by $\{m_\ell \text{ monomial of } \mathcal{R}_{q,n} \mid deg(m_\ell) \leq D - deg(f_i)\}$ and $\{m_\ell \text{ monomial of } \mathcal{R}_{q,n} \mid deg(m_\ell) \leq D\}$. So we try to exclude elements of the kernel of $f_{D,m}^{\text{mac}}$, denoted $Ker(f_{D,m}^{\text{mac}})$. To predict elements in $Ker(f_{D,m}^{\text{mac}})$, we introduce an order on the row: Let $(P_1, \ldots, P_m)$ and $(Q_1, \ldots, Q_m)$ be two elements of $\mathcal{B}_{1,D} \times \cdots \times \mathcal{B}_{m,D}$. We say $(P_1, \ldots, P_m) < (Q_1, \ldots, Q_m)$ if $\exists i \in \{1, \ldots, m\}$ s.t. $LT(P_i) \prec LT(Q_i)$ and $\forall j, i < j \leq m$, $P_j = Q_j = 0$. This order is not a total ordering on $\mathcal{B}_{1,D} \times \cdots \times \mathcal{B}_{m,D}$. But, for example, for the canonical basis $\{(\ldots, 0, m_\ell, O, \ldots), \text{ with } m_\ell \text{ monomial}\}$, of $\mathcal{B}_{1,D} \times \cdots \times \mathcal{B}_{m,D}$, it is a total ordering. So all elements can be decomposed in an ordered linear basis. If this element is in $Ker(f_{D,m}^{\text{mac}})$, a sufficient condition is to exclude the higher element of the basis. The new basis still contains a supplementary vector space of $Ker(f_{D,m}^{\text{mac}})$ and its image is still $Im(f_{D,m}^{\text{mac}})$. But if we want to *keep this order* during the computation, the only elementary operation possible is the addition with lower elements. Thus for the Gaussian elimination on the matrix $\mathcal{M}_{D,m}^{\text{acaulay}}$, the elementary operations on a row are only simplifications with previous rows.

Moreover, an element of $Ker(f_{D-1,m}^{\text{mac}})$ gives several elements in $Ker(f_{D,m}^{\text{mac}})$. So we study the Macaulay matrix $\mathcal{M}_{D,m}^{\text{acaulay}}$ incrementally in the degree $D$ and construct the vector space $\mathcal{B}_{1,D} \times \cdots \times \mathcal{B}_{m,D}$ from a basis of $\mathcal{B}_{1,D-1} \times \cdots \times \mathcal{B}_{m,D-1}$ computed after the Gaussian elimination. Indeed, the basis of $B_{i,D}$ is composed by the basis of $B_{i,D-1}$ and elements with degree $D$ which can be written as $X_\ell P_j$ with $P_j$ in $B_{i,D-1}$. Thus, an element $(P_1, \ldots, P_m)$ of $\mathcal{B}_{1,D} \times \cdots \times \mathcal{B}_{m,D}$ is so that exists an unique $(Q_1, \ldots, Q_m)$ in $\mathcal{B}_{1,D-1} \times \cdots \times \mathcal{B}_{m,D-1}$ with $Q_i$ divides $P_i$ for all $i \in \{1, \ldots, m\}$.

An other problem is to prevent elements of $Ker(f_{D,m}^{\text{mac}})$ which are issued from a trivial relation $fg - gf = 0$. For this order, the matrix $\mathcal{M}_{D,i}^{\text{acaulay}}$, for $i < m$ is only the first rows of $\mathcal{M}_{D,m}^{\text{acaulay}}$. So we want to avoid $gf_j - f_j g = 0$ for $j > 1$ and $g$ in $Im(f_{D-1,j-1}^{\text{mac}})$. As $f_j g$ can be written as $\sum_{i=1}^{j-1} P_i f_i$, we deduce that the relation $gf_j - f_j g = 0$ corresponds to $f_{D,m}^{\text{mac}}(-P_1, \ldots, -P_{j-1}, g, 0, \ldots) = 0$, i.e. $(-P_1, \ldots, -P_{j-1}, g, 0, \ldots) \in Ker(f_{D,m}^{\text{mac}})$. The higher element of the basis in the decomposition of this element has this shape $(*, \ldots, *, P_j, 0, \ldots)$ with $LT(P_j) = LT(g)$. In other words, we do not consider the elements so that

$LT(P_j)$ is reduced by $\mathcal{M}^{\text{acaulay}}_{D,j-1}$. That means that, for a degree $D$, the computation is done incrementally on the number of polynomials.

## 3.2 description

To resume this explanation, we can present the algorithm for a system $(f_1, \ldots, f_m)$ so that $deg(f_i) = d_0$:

$F_5$ **algorithm:** Execute the following steps:

1. **Initialization:** Consider the basis $\{(1, 0, \ldots), \ldots, (\ldots, 0, 1, 0, \ldots), \ldots, (\ldots, 0, 1)\}$ for $\mathcal{B}_{1,d_0} \times \cdots \times \mathcal{B}_{m,d_0}$. Construct the matrix $\mathcal{M}^{\text{acaulay}}_{d_0,i}$ and then perform a Gaussian elimination with only simplification with previous rows. We obtain a matrix $\tilde{\mathcal{M}}^{\text{acaulay}}_{d_0,m}$.
2. **Construction:** Suppose $\mathcal{B}_{1,D} \times \cdots \times \mathcal{B}_{m,D}$ and $\tilde{\mathcal{M}}^{\text{acaulay}}_{D,m}$ computed. Construct a basis for $\mathcal{B}_{1,D+1} \times \cdots \times \mathcal{B}_{m,D+1}$ from $\mathcal{B}_{1,D} \times \cdots \times \mathcal{B}_{m,D}$ so that an element $(\ldots, *, P_j, 0, \ldots)$ satisfied $P_j$ is not reducible by $\tilde{\mathcal{M}}^{\text{acaulay}}_{D,j-1}$ and the matrix $\mathcal{M}^{\text{acaulay}}_{D+1,m}$ from $\tilde{\mathcal{M}}^{\text{acaulay}}_{D-1,i}$.
3. **Gaussian elimination:** Perform a Gaussian elimination on rows with only simplification with previous rows.
4. **Repeat:** repeat the step 2 for $D \leftarrow D + 1$ until having a Gröbner basis.

In fact, we do not need to keep all the basis of $\mathcal{B}_{1,D} \times \cdots \times \mathcal{B}_{m,D}$ as we construct $\mathcal{M}^{\text{acaulay}}_{D+1,m}$ from $\tilde{\mathcal{M}}^{\text{acaulay}}_{D,m}$ but the form of the elements $(*, \ldots, *, P_j, 0, \ldots)$ which is characterized by $(LT(P_j), f_j)$, denote the signature of the row of $\mathcal{M}^{\text{acaulay}}_{D+1,m}$.

The degree $D$ of the signatures reached to compute a Gröbner bases is the same than in section 2.2. But in the matrix $\mathcal{M}^{\text{acaulay}}_{D,m}$, all polynomials can have a lower degree than $D$.

In the field $\mathbb{F}_2$, M. Bardet, J.C. Faugère and B. Salvy [1] have introduced a new criterion to avoid relations $f_i f_i = f_i$ deduced from the Fröbenius application. In step 2, $P_j$ does not have to be reducible by $\tilde{\mathcal{M}}^{\text{acaulay}}_{D,j-1}$.

## 3.3 Semi-regular sequences

In the report [1], the authors introduced the notion of semi-regular sequences for overdefined systems. We have, in section 1.1, distinguished two important cases for finite fields, $\mathbb{F}_2$ and $\mathbb{F}_q$. In the field $\mathbb{F}_2$, we have a criterion deduced from the Fröbenius application. If we are interested by a system $\mathcal{A}^n_m$ on a field $\mathbb{F}_q$, with $q \gg n$, i.e. $q$ is very high compared to $n$. Then the trivial relation issued from the Fröbenius application will not be reached during computation and all computation done is similar to computation on $\mathbb{Q}$.

**Definition 4**

**Homogeneous semi-regular sequence :** *Let $f_1, \ldots, f_m$ be a sequence of $m$ homogeneous polynomials (i.e. for all $m$ monomial of $f_i$, $deg(m) = deg(f_i)$), and $\mathcal{I} = \langle f_1, \ldots, f_m \rangle$.*

- *The* degree of regularity *of $\mathcal{I}$ is the minimal degree $d$ so that $\{LT(f) \mid f \in \mathcal{I}, deg(f) = d\}$ is exactly the set of monomials of degree $d$ in $\mathcal{R}_{q,n}$, denoted $D_{reg}(\mathcal{I})$.*
- *$f_1, \ldots, f_m$ is a* homogeneous semi regular sequence on $\mathbb{F}_2$ *if for $i \in \{1, \ldots, m\}$, if $g_i f_i = 0$ in $\mathcal{R}_{q,n}/\langle f_1, \ldots, f_{i-1} \rangle$ and $deg(g_i f_i) < D_{reg}(\mathcal{I})$ then $g_i = 0$ in $\mathcal{R}_{q,n}/\langle f_1, \ldots, f_{i-1}, f_i \rangle$.*
- *$f_1, \ldots, f_m$ is a* homogeneous semi regular sequence on $\mathbb{Q}$ *if for $i \in \{1, \ldots, m\}$, if $g_i f_i = 0$ in $\mathcal{R}_{q,n}/\langle f_1, \ldots, f_{i-1} \rangle$ and $deg(g_i f_i) < D_{reg}(\mathcal{I})$ then $g_i = 0$ in $\mathcal{R}_{q,n}/\langle f_1, \ldots, f_{i-1}, f_i \rangle$.*

**Affine semi-regular sequence :** *Let $f_1, \ldots, f_m$ be a sequence of $m$ polynomials, and $\mathcal{I} = \langle f_1, \ldots, f_m \rangle$. Let $f_i^h$ the homogeneous part of largest degree of $f_i$.*

- *$f_1, \ldots, f_m$ is a* semi regular sequence *if $f_1^h, \ldots, f_m^h$ is a homogeneous semi-regular sequence.*
- *the* degree of regularity *of $\mathcal{I}$ is the degree of regularity of $\langle f_1^h, \ldots, f_m^h \rangle$, denoted $D_{reg}$.*

With this sequence of polynomials, the matrix generated by $F_5$ algorithm has a full rank for the degree $d < D_{reg}$. i.e., the image of the basis of $\mathcal{B}_{1,d} \times \cdots \times \mathcal{B}_{m,d}$ generated by $F_5$ is a basis of $Im(f_{d,m}^{\text{mac}})$.

This means that, for semi-regular sequences, the number of rows $H_{m,n}(d)$ of the matrix in the homogeneous case, for $d < D_{reg}$, is known, and is given by a recurrence formula $H_{m,n}(d) = H_{m-1,n}(d) + \#\{m_\ell \text{ monomial of degree } d - d_m\} - H_{m,n}(d - d_m)$ with initial conditions $H_{m,n}(d) = 0$ if $m \leq 0$ or $d < min(deg(f_k) \mid k \leq m\}$. Then the number of rows of a matrix for the affine case is $\sum_{d'=1}^{d} H_{m,n}(d')$.

The degree $D_{reg}$ corresponds to the degree $d$ when we will have more rows than columns for the homogeneous part of largest degree. It is the minimal degree so that $H_{m,n}(d) > \#\{m_\ell \text{ monomial of degree } d\}$. If we consider the series $f(t) = \sum_{d \geq 0}(H_{m,n}(d) - \#\{m_\ell \text{ monomial of degree } d\})t^d$, the degree $D_{reg}$ is give when the coefficient of this series is negative. the expression of $f$ for quadratic equations is : $\frac{(1+y)^n}{(1+y^2)^m}$ for $\mathbb{F}_2$ and $\frac{(1-y^2)^m}{(1-y)^n}$ for $\mathbb{F}_q$, with $q \gg n$.

Moreover, in the article [1], the authors have made a conjecture verified on many computer experiments:

**Conjecture 1** *a "generic" sequence of polynomials is semi-regular.*

The degree $D_{reg}$ is the minimal degree reached for a Gröbner basis computation. For example, for the Buchberger algorithm, consider a monomial $m_\ell$ with degree $D_{reg} - 1$ which does not appear in $\{LT(f) \mid f \in \mathcal{I}, deg(f) = D_{reg} - 1\}$. Then for $X_j m_\ell$ and $X_i m_\ell$, there

is $g_j$ and $g_i$ in the Gröbner basis computed so that $LT(g_j)$ and $LT(g_i)$ divide respectively $X_j m_\ell$ and $X_i m_\ell$. So the $\mathcal{S}$-polynomial of $(g_i, g_j)$ will be studied and we have $m_\ell$ divides $Lcm(LT(g_j), LT(g_i))$. So the degree of computation of the $\mathcal{S}$-polynomial is at least $D_{reg}$.

# 4 The XL algorithm

## 4.1 The basic Principle of XL

This subsection refers to the paper [5, 8]. Let $D$ be the parameter of XL algorithm. The problem is to find at least one solution $u = (u_1, \ldots, u_n) \in \mathbb{F}_q^n$ of a system $\mathcal{A}_m^n$. Let be $\mathcal{I}$ the ideal generated by $\{f_1, \ldots, f_m\}$.

**The XL algorithm** Execute the following steps:

1. **Multiply:** Generate all the products $(\prod_{j=1}^{k} X_{i_j}).f_i \in \mathcal{I}$ with $k \leq D - deg(f_i)$.
2. **Linearize:** Consider each monomial in the $X_i$ of degree $\leq D$ as a new variable and perform Gaussian elimination on the equations obtained in 1.

   The ordering on the monomials must be such that all terms containing one variable (say $X_1$) are eliminated last.
3. **Solve:** Assume that step 2 yields at least one univariate equation in the powers of $X_1$. Solve this equation over the finite fields (e.g., with Berlekamp's algorithm).
4. **Repeat:** Simplify the equations and repeat the process to find the values of the other variables.

Step 1 is the construction of the Macaulay matrix $M_{D,m}^{acaulay}$ with such an order that all terms containing one variable are eliminated last. So the step 1 and 2 are **exactly the same method** as the one presented in section 2.2. We deduced that for $D$ big enough and if we have only one solution in the finite field, XL algorithm will terminate because it will have computed the Gröbner basis.

There are different extensions of XL called FXL (which stands for Fixing and XL) and XL' [7]: the algorithm FXL consists of guessing the values of a few variables and then applying XL; the algorithm XL' proposes to modify the *Solve* step of the XL algorithm to find $r$ equations in $r$ variables. In the following we just study the XL algorithm, but all points mentioned can be extended to FXL or XL'.

## 4.2 Polynomials' relations in the XL algorithm

The XL algorithm does not take care of simple relations between polynomials. The authors take some relations, the trivial relations, into account in the article [7] for the behavior of XL. But they do not integrate these relations in the algorithm and the study of the behaviors of XL for a quadratic multivariate system with $D = 5$ could consider other relations deduced from the second criterion too. . So the Gaussian elimination will reduce to zero a lot of

the equations generated by XL. As the step 1 and 2 are *exactly* the same method as the one presented in section 2.2, we can use the same techniques as $F_5$ algorithm to avoid some reductions to zero deduced by the Buchberger criteria and the Fröbenius application. The relation are $f_i f_j - f_j f_i = 0$ for the first Buchberger criterion, $\mathcal{S}pol(f_i, f_j)f_k - f_k \mathcal{S}pol(f_i, f_j) = 0$ for the second criterion and $f^{q-1} f - f = 0$ for the Fröbenius application.

All these relations are still true even if XL algorithm simplifies monomials by the relation $x_i^q = x_i$. This simple proposition gives a criterion that can be applied directly.

**Proposition 2** *Let $\{f_1, \ldots, f_m\}$ be a multivariate system of equations Let, $\forall\, k \in \{1, \ldots, m\}$, $\mathcal{I}_k$ the monomial ideal spanned by the $LM(f_i)$, $LM(\mathcal{S}pol(f_i, f_j)$, $1 \leq i < j < k$ and $LM(f_k^{q-1})$.*

*For all $m_\ell \in \mathcal{I}_k$, $m_\ell f_k$ will be reduced to zero by the lines $m'_\ell f_j$ and $m''_\ell f_k$, $j < k$, $m'_\ell \notin \mathcal{I}_j$ and $m''_\ell \notin \mathcal{I}_k$.*

### 4.3 The choice of the monomial ordering

To compare XL algorithm with Gröbner basis, we need to give an explicit monomial ordering for XL. As the algorithm does not give explicit monomial ordering, we need to introduce the following lemma :

**Lemma 1** *Let $\mathcal{A}_m^n$ be a system of $m$ multivariate equations with $n$ variables.*

$$\textit{XL terminates for a degree } D \iff \begin{array}{l}\textit{XL terminates for a degree } D \\ \textit{with the Lexicographic ordering}\end{array}$$

**Proof.** Let be $M$ (respect. $M'$) the matrix generated by all the products $(\prod_{j=1}^{k} x_{i_j}).f_i \in \mathcal{I}$ with $k \leq D - degree(f_i)$ for XL (respect. with the Lexicographic ordering). So we can write $M = \left( \begin{array}{c|c} A & B \end{array} \right)$ and $M' = \left( \begin{array}{c|c} A' & B' \end{array} \right)$ so that $B$ (respect. $B'$) correspond to the columns for the univariate monomials. Moreover $M'$, $A'$ and $B'$ are only columns permutation of $M$, $A$ and $B$.

If XL terminates for a degree $D$, it means that $rank(M) > rank(A)$. Then $rank(M') > rank(A')$ and then XL will find an univariate polynomial with the lexicographic ordering. □

## 5 XL computation: $D$-Gröbner basis

According to lemma 1, we assume that XL computation is done with a lexicographic order. XL deals with polynomials of degree smaller than an integer $D$. We can look more precisely how this limitation affects the computation of Gröbner bases. It is the well known notion of $D$-Gröbner basis. See also [2] for further details.

## 5.1 $D$-Gröbner basis

**Definition 5** *Let $G$ be a finite subset of $\mathcal{S}_{q,n}$, and let $d \in \mathbb{N}$. Then we call $G$ a* $D$-Gröbner basis *if $\forall\ f, g \in G$ with $degree(Lcm(LM(f), LM(g))) \leq D$, $\mathcal{S}pol(f,g)$ reduces to zero by $G$.*

With a selection strategy based on the degree, the $D$-Gröbner basis is simply the first element of the computed Gröbner basis. The most significant results for $D$-Gröbner basis are based on sets of homogeneous polynomials (see [2]). These results can be used to obtain a deeper understanding of the behavior of degrees in any non-homogeneous Gröbner basis with homogenization and (dehomogenization). For $f \in \mathcal{S}_{q,n}$ with $degree(f) = d$, we now define $f^* = Z^d f(\frac{X_1}{Z}, \ldots, \frac{X_n}{Z})$ the *homogenization* of $f$ in $\mathbb{F}_q[X_1, \ldots, X_n, Z]$. For $g \in \mathbb{F}_q[X_1, \ldots, X_n, Z]$, we define $g_* \in \mathcal{S}_{q,n}$ by $g_*(X_1, \ldots, X_n) = g(X_1, \ldots, X_n, 1)$. $g_*$ is called the *dehomogenization* of $g$.

**Theorem 2** *Let $F$ be a finite subset of $\mathcal{S}_{q,n}$, let $D \in \mathbb{N}$, and suppose $G \subset \mathbb{F}_q[X_1, \ldots, X_n, Z]$ is a $D$-Gröbner basis of $F^*$ w.r.t. some term order. Furthermore, let $p \in \mathcal{S}_{q,n}$ with $degree(p) = d'$. Then the following are equivalent :*

*(i) There is $q_f \in \mathcal{S}_{q,n}$ such that $p = \sum_{f \in F} q_f f$ and $max\{degree(q_f f) | f \in F\} \leq D$.*

*(ii) The reduction of $Z^{d-d'} p^*$ by $G$ is zero.*

**Corollary 1** *Let $F$ be a finite subset of $\mathcal{S}_{q,n}$, let $D \in \mathbb{N}$, and suppose $G \subset \mathcal{S}_{q,n}$ is a $D$-Gröbner basis of $F$ w.r.t. some term order. Furthermore, let $p \in \mathcal{S}_{q,n}$.*

*If there is $q_f \in \mathcal{S}_{q,n}$ such that $p = \sum_{f \in F} q_f f$ and $max\{degree(q_f f) | f \in F\} \leq D$ then the reduction of $p$ by $G$ is zero.*

We do not have the reciprocity: let us consider $\mathcal{I} = \langle X_2^2 + X_3 = 0,\ X_1 X_2 - X_2 = 0,\ X_3^2 + X_1 = 0 \rangle$ and let be $G$ the 3-Gröbner basis for the DRL order $[X_3, X_2, X_1]$. We have $G = < X_2^2 + X_3, X_1 X_2 - X_2, X_3^2 + X_1, X_1 X_3 - X_3, X_1^2 - X_1 >$ but $X_1^2 - X_1 = (X_1 - 1)(X_3^2 + X_1) - (X_3 X_1 - X_3)(X_2^2 + X_3) + X_3 X_2 (X_1 X_2 - X_2)$. To see this, the reduction of the $\mathcal{S}$-polynomial of $g$ and $h$ in the homogenized computation keeps the degree of $Gcd(LM(g), LM(h))$ whereas in ordinary computation, we can gain a degree.

## 5.2 Link between the results of XL and $D$-Gröbner basis

Let be $F = \{f_1, \ldots, f_m\}$ a polynomial system. With lemma 1, we assume that XL algorithm uses the lexicographic order for its linearization. We consider the set $F_{XL}$ of all the polynomials computed by the Gaussian elimination. Let us consider $G$ a $D$-Gröbner basis of $F^*$ for the lexicographic order. As the XL algorithm result is not based on homogeneous polynomial, we will find a link between $F_{XL}$ and $G_*$. As we do not have the unicity of Gröbner basis, to compare the two results, we need to reduce the bases.

**Proposition 3** *Let be $F_1$ the reduced basis for $F_{XL} \cup \{X_1^q - X_1, \ldots, X_n^q - X_n\}$ and $G_1$ for $G_*$ where $G$ is the $D$-Gröbner basis of $(F \cup \{X_1^q - X_1, \ldots, X_n^q - X_n\})^*$.*

*Then $F_1 = G_1$.*

# 6 Comparison of XL algorithm and Gröbner basis on $\mathbb{F}_2$

## 6.1 Theoretical comparison of degree

In the ring $\mathcal{R}_{2,n}$, the only univariate monomials in the variable $X_1$ are 1 and $X_1$. They are the first two monomials of DRL monomial ordering $[X_n, \ldots, X_1]$.

More over the reduction of an $\mathcal{S}$-polynomial of $f$ and $X_i + a_i$ corresponds to the substitution step and does not exceed the degree of $f$. So we have the following theorem:

**Theorem 3** *Let $\mathcal{A}_m^n$ be a polynomial system with $n$ variables, $m$ equations on $\mathbb{F}_2$.*

*If XL algorithm terminates on the system $\mathcal{A}_m^n$ for a degree $D$ eliminated in order $X_1, \ldots, X_n$, the Gröbner basis computation for a DRL monomial ordering $[X_n, \ldots, X_1]$ does not overreach the degree $D$ during the computation.*

### 6.1.1 Comparison between $F_5$ and $XL$

For a random sequence $f_1, \ldots, f_m$, this sequence is a semi-regular sequence. XL algorithm computes $Im(f_{D,m}^{\text{mac}})$ for an homogeneous system. So we need to consider the homogenization of $f_1, \ldots, f_m$. The homogenization of $f_1, \ldots, f_m$ gives an homogeneous random sequence, so it is also a semi-regular sequence. With $F_5$ algorithm, we have the explicit number of rows which do not reduce to zero. So if we want to find an univariate polynomial with XL, we need to have a number of rows higher than the number of monomial with degree $D$ minus the number of univariate monomial in $X_1$ (i.e., $X_1$ and 1). This means that XL algorithm will finish for a degree $D > 1$ if $H_{m,n+1}(D) > \binom{n+1+D}{n+1} - 2$.

We have compared this degree for a system $\mathcal{A}_m^n$ of quadratic equations. The degree $D$ corresponds to the first coefficient of the serie defined by $\frac{(1+y)^n}{(1-x)(1+y^2)^m} - \frac{1+y}{1-y}$ which is negative. The Figure 1 present a comparison of the degree reached between XL algorithm and Gröbner basis computation for a variation of the number of variables $n$ and Figure 2 for a variation of the number of equations $m$.

With these figures, we do not have a noticeable difference between the degree reached by the two algorithms. So we can say that for random systems, the methods of XL and Gröbner basis are almost the same.

For the complexity point of view, if $N_D$ is the size of the matrix constructed, then the whole complexity is the cost of linear algebra on this matrix, which is $N_D^w$ where $w \leq 3$ is the coefficient of linear algebra. XL algorithm creates matrices with $\sum_{i=1}^{m} \sum_{k=0}^{D-deg(f_i)} \binom{n}{k}$ rows and $\sum_{k=0}^{D} \binom{n}{k}$ columns, whereas $F_5$ creates square matrices with $\sum_{k=0}^{D} \binom{n}{k}$ columns. For example, for a quadratic multivariate polynomials with $n = 128$ and $m = 130$, both algorithm reached the same degree 17 and the matrices generated by XL algorithm will have about $19,7.10^{20}$ rows and $7,2.10^{20}$ columns compared to squared matrices with only $7,2.10^{20}$ rows and columns for $F_5$ algorithm. So the number of columns for $F_5$ algorithm matrices is lower or egal to the one for $XL$ algoritm matrices whereas the number of rows of the matrices constructed is very different, Figure 3 presents the number of rows of each

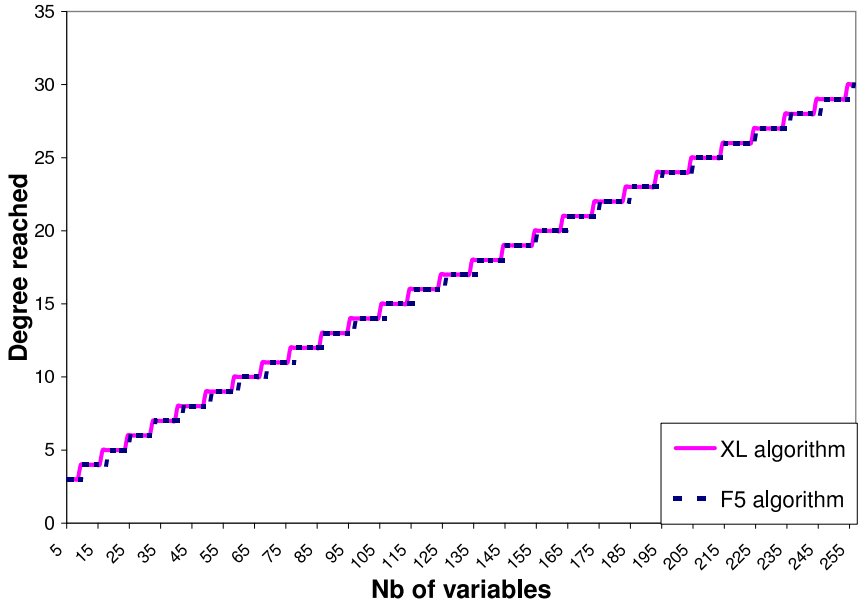


Figure 1: Behavior of XL and $F_5$ on $\mathbb{F}_2$ with respect the number of variables

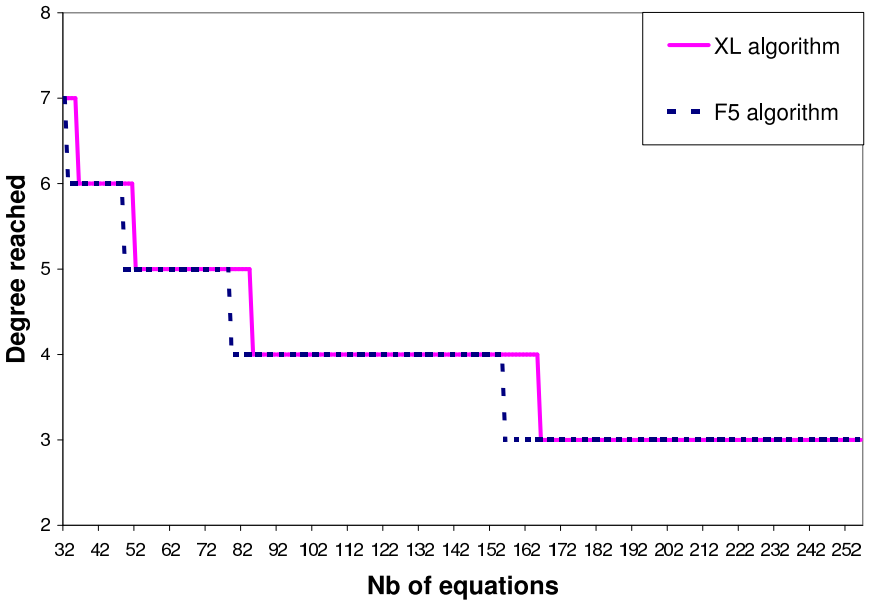


Figure 2: Behavior of XL and $F_5$ on $\mathbb{F}_2$ with respect the number of equations

matrices with a logarithm scale. As we can see, the difference between the two curves give us a multiplicative constant.

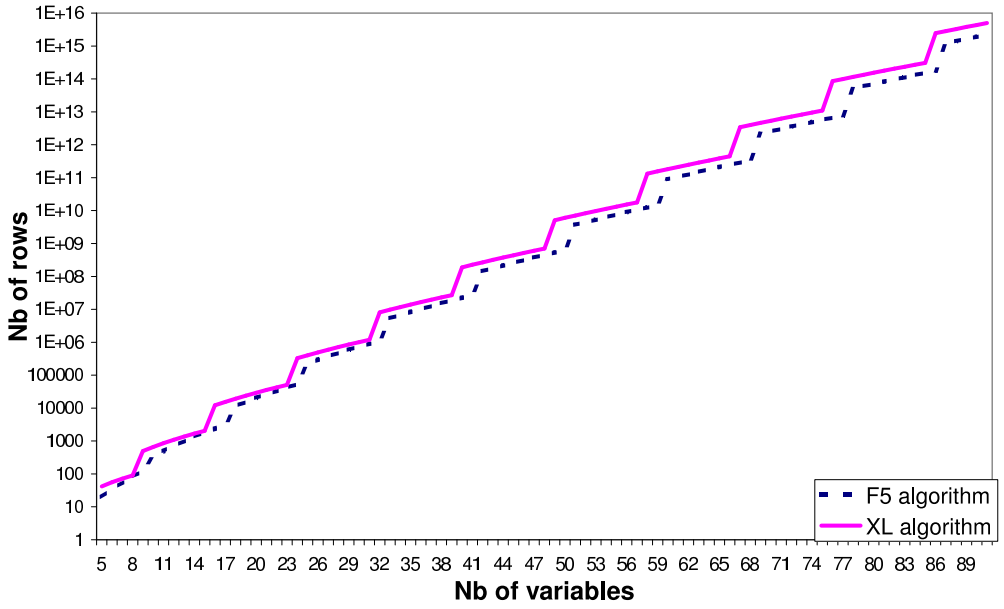


Figure 3: Matrices of XL algorithm and $F_5$ algorithm on $\mathbb{F}_2$

In cryptography, the systems studied seem to be random but have a structure behind them. So we need to make experimental tests on cryptosystems to have an idea of the efficiency of both algorithms.

## 6.2 Example on the cryptosystem HFE

Hidden Field Equations (HFE) is an asymmetric cryptosystem. It does not use the number theory but it is based on multivariate polynomials over a finite field (cf [15]). The idea of HFE is to take a secret univariate polynomial (the private key) on an extension of the finite field, then to express this polynomial on the finite field. We thus obtain an algebraic system (the public key). This system is composed with polynomials of degree 2.

We have implemented the XL algorithm in Magma to test on the examples. Moreover as XL algorithm has a better behavior for $m > n$, we have fixed some variables to be in the case $m = n + 2$.

With Figure 4, we see that XL algorithm's maximal degree increases whereas for Gröbner basis computation, the degree of resolution does not change and does not exceed 3. In fact, XL algorithm seems to follow the Figure 1. So XL does not seem to find a difference between a random system and the HFE cryptosystem contrary to Gröbner basis computation.

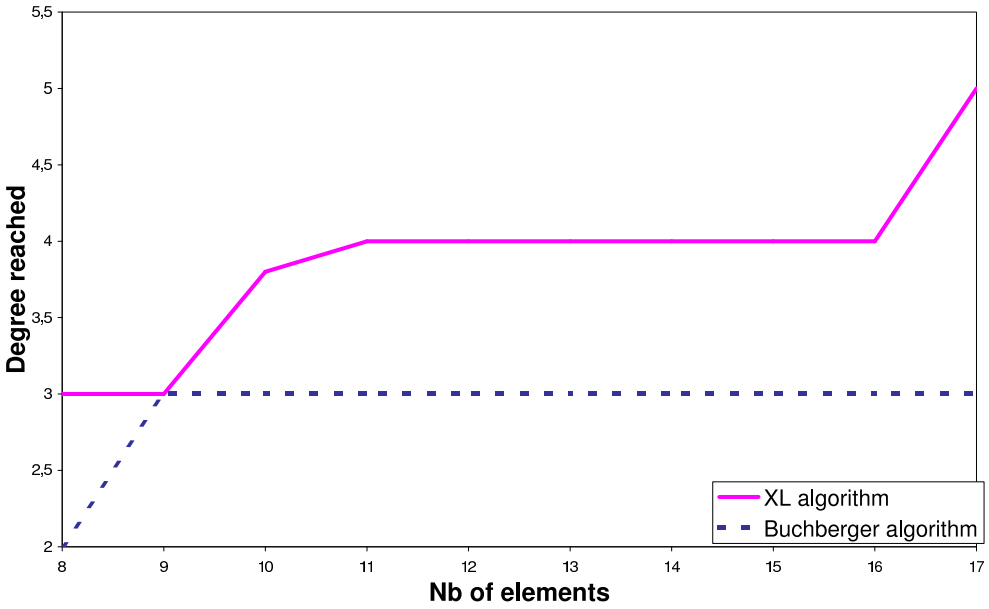


Figure 4: Comparison between XL and Gröbner algorithms on HFE over $\mathbb{F}_2$

# 7 Comparison between XL and Gröbner bases on $\mathbb{F}_q$

As the XL algorithm does not compute a completed Gröbner basis, we first present emulated algorithms for which we can give theoretical results.

## 7.1 Presentation of an emulated algorithm

Let us consider $F = \{f_1, \ldots, f_m\}$ a system of equations and $G = F \cup \{X_1^q - X_1, \ldots, X_n^q - X_n\}$.

The problem of the XL algorithm is that we do not keep the computation we have done in step 1. So when we repeat the step 1, many computations will be done again. We have seen in the proof of theorem 5 that the substitution step of $X_i$ by $a_i$ can be replaced by adding the polynomial $X_i - a_i$ in the Gröbner basis. We could emulated XL algorithm with replacing step 1 and 2 by a Gröbner basis computation with an elimination order $[X_n, \ldots, X_{j+1}]$,$[X_j]$ to find the univariate polynomial. But we will loose computation results at each step. In the new algorithm, we combine a Gröbner basis computation on a fixed order, the DRL order, and use FGLM methods to verify if we do not have an univariate polynomial.

**An Algorithm**

Execute the following steps:

1. **Initialization:** Initialize the variable $i$ at 1.

2. **Gröbner Basis:** Compute a Gröbner basis $G'$ for $G$ with the DRL order and a selection strategy based on the degree so that at each degree $d$,we compute the reduction $\overline{X_i^j}$ of $X_i^j$ by the pretend basis $G'$ and $\{1, \overline{X_i^k}, k < j\}$, for $j \in \{1, \ldots, d\}$. We stop the computation if we find a $j$ so that $\tilde{X_i^j} = 0$ and find a pretend basis $G'$.
3. **Solve:** Find the explicit relation $\sum_{j=0}^{d} X_i^j$ which is reduced to zero by the pretend basis $G'$ and Solve this equation over the finite fields (e.g., with Berlekamp's algorithm). We find $a_i$ a solution.
4. **Repeat:** Add the polynomial $X_i + a_i$ in the pretend basis $G'$, $i \leftarrow i + 1$ and Continue the computation of step 2.

**Theorem 4** *Let be $\mathcal{A}_m^n$ a polynomial system with $n$ variables, $m$ equations.*
*If XL algorithm terminates on the system $\mathcal{A}_m^n$ for a degree $D$, the above algorithm terminates too and the Gröbner basis computation satisfies that all computed polynomials have a degree lower than $D$.*

This algorithm does not depend on a parameter $D$. Contrary to the XL algorithm, this algorithm always gives out a result. In the worst of cases, it computes the Gröbner basis and then sends back the field equation to the univariate polynomial. Moreover, this algorithm has the advantage to keep all the intermediate computations. So contrarily to the XL algorithm,we do not waste time to redo the same computations.

### 7.1.1 Comparison between $F_5$ and $XL$

For a random sequence $f_1, \ldots, f_m$ on $\mathbb{F}_q$, $q \gg n$, this sequence is a semi-regular sequence on $\mathbb{Q}$. XL algorithm computes $Im(f_{D,m}^{\text{mac}})$ for an homogeneous system. So we need to consider the homogenization of $f_1, \ldots, f_m$. The homogenization of $f_1, \ldots, f_m$ gives an homogeneous random sequence, so it is also a semi-regular sequence. With $F_5$ algorithm, we have the explicit number of rows which do not reduce to zero. So if we want to find an univariate polynomial with XL, we need to have a number of rows higher than the number of monomial with degree $D$ minus the number of univariate monomials in $X_1$ (i.e., $X_1^D, \ldots, 1$).

This means that XL algorithm will finish for a degree $D > 1$ if $H_{m,n+1}(D) > \binom{n+D}{n} - (D+1)$.

We have compared these degrees for a system $\mathcal{A}_m^n$ of quadratic equations. The degree $D$ corresponds to the first coefficient of the serie defined by $\frac{(1-y^2)^m}{(1-y)^{n+1}} - \frac{1}{(1-y)^2}$ which is negative. The Figure 5 presents a comparison of the degree reached between XL algorithm and Gröbner basis computation for a variation of the number of variables $n$ with $m = n + 2$ and Figure 6 for a variation of the number of equations $m$. First we can see that for random polynomials we always have computed a Gröbner basis before finding the univariate polynomial for a degree $D$. Moreover, we can see the behavior of the degree of XL algorithm does not seem to follow the formula $\frac{n}{\sqrt{m}}$ as it was said in [8].

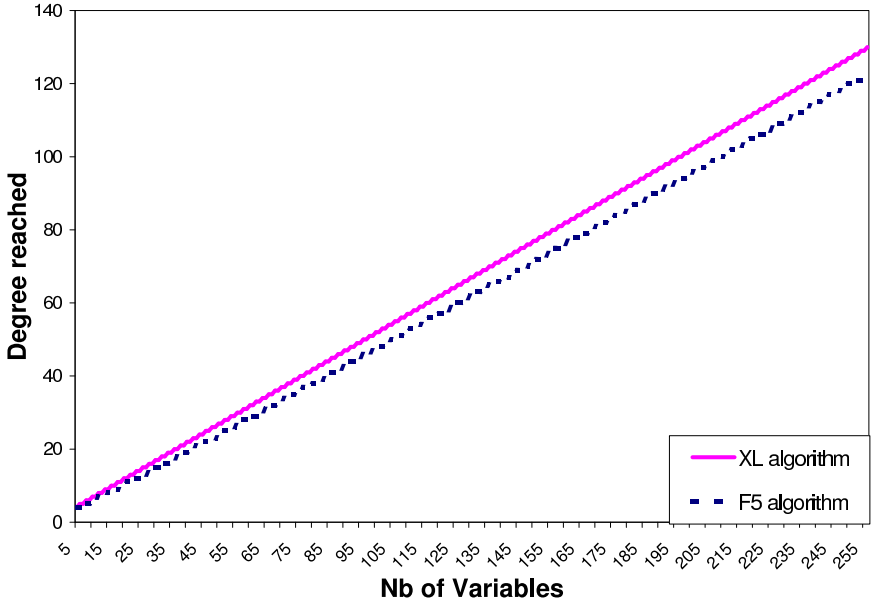


Figure 5: Behavior of XL and $F_5$ on $\mathbb{F}_q$ with respect the number of variables

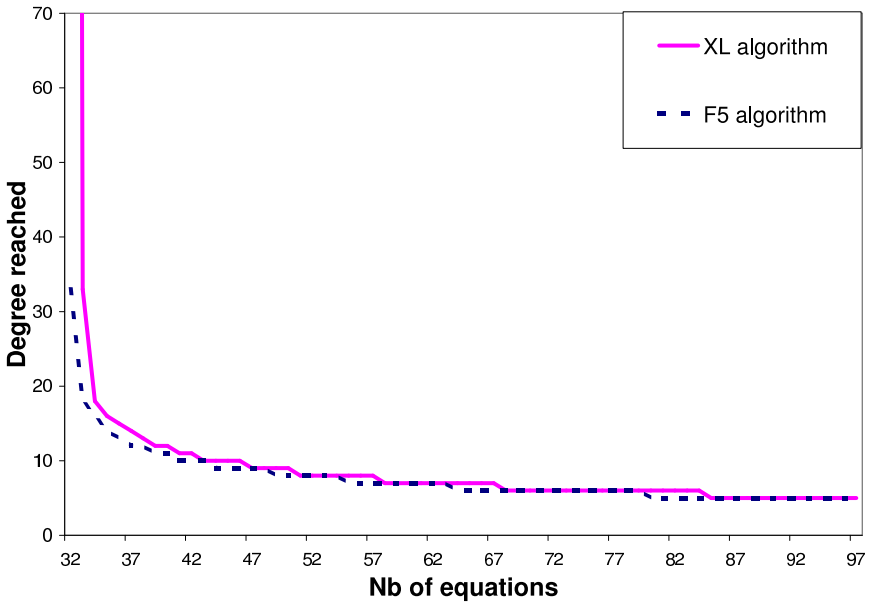


Figure 6: Behavior of XL and $F_5$ on $\mathbb{F}_q$ with respect the number of equations

We have said that the complexity is $N_D^w$, where $N_D$ is the size of the matrix constructed and $w$ the coefficient of linear algebra. As XL algorithm has a higher degree $D$ than $F_5$ algorithm, the difference of the size of constructed matrices is very important. For exemple, for quadratic multivariate polynomials with $n = 128$ and $m = 130$, XL algorithm reached a degree 66 whereas $F_5$ algorithm reached a degree 61. So the matrices generated by XL algorithm will have about $94317.10^{49}$ rows and $6332.10^{49}$ columns compared to squared matrices with only $8,4.10^{49}$ rows and columns for $F_5$ algorithm.

For the case $m = n$, the number of solution with multiplicity of a random system with quadratic equations is $\prod_{i=1}^{m} deg(f_i) = 2^n$, which is the Bezout bound. So the univariate polynomial has this degree and XL will terminate for this degree. Whereas, the computation of the Gröbner basis will not exceed $1 + \sum_{i=1}^{n}(deg(f_i) - 1) = n + 1$ for any ordering. This computation is done with a DRL ordering and then we use FGLM algorithm [9] to find the wanted ordering.

All this study is still true if $D < q$ and not only for $q \gg n$.

## 7.2 Examples on HFE systems

As in section 6, we can not only consider random polynomials, we need to look at cryptosystems to see the efficiency of the algorithm.

We have tested the four algorithms on the field $F_{16}$. Moreover as in section 6, we have fixed some variables to be in the case $m = n + 2$.

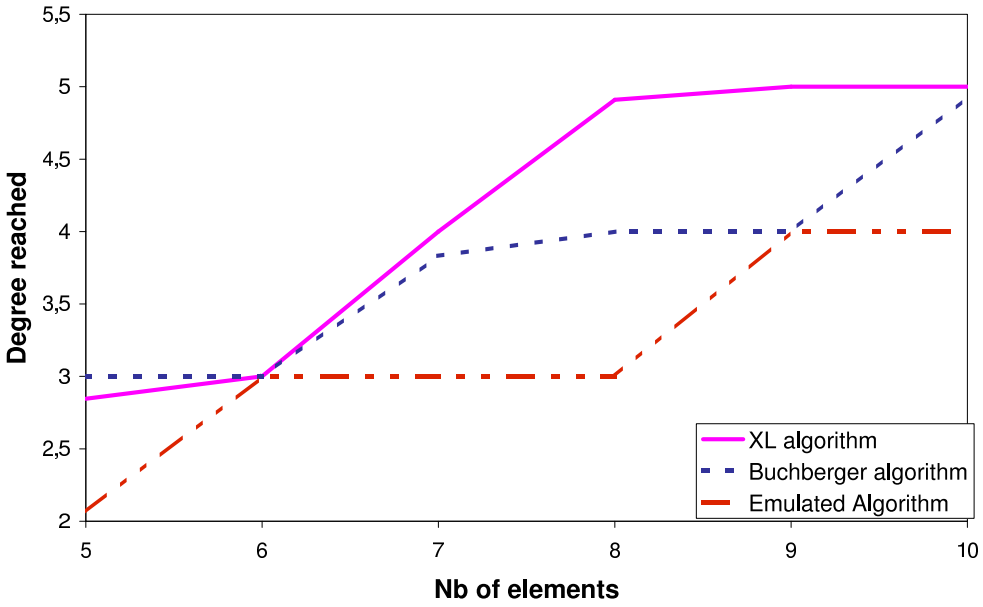


Figure 7: Comparison between XL and Gröbner algorithms on HFE on $\mathbb{F}_{16}$

On the experimental results presented Figure 7, the Buchberger algorithm is still better than XL algorithm for a number of elements higher than 6. The emulated algorithms still give a better results. Nevertheless, the degree reached by the emulated algorithm can be the degree reached for another Gröbner basis algorithm.

## Conclusion

This study proves that on $\mathbb{F}_2$, Gröbner basis algorithms will always have a better behavior than XL algorithm. For semi-regular sequences, even if XL and $F_5$ algorithm reached a similar degree, the matrix constructed by XL is huger than the matrix of $F_5$. So the complexity of XL is higher than Gröbner basis complexity. For specific systems, as HFE cryptosystems, we found the same results.

For field $\mathbb{F}_q$ with $q \gg n$, we show that, for semi-regular sequences, $F_5$ algorithm will have a better complexity and we have tested on the HFE cryptosystem and still have the same results.

## References

[1] M. Bardet, J. C. Faugère, and B. Salvy. Complexity of Gröbner basis computation for semi-regular sequences over $\mathbb{F}_2$ with solutions in $\mathbb{F}_2$. Technical report, INRIA Rocquencourt, 2003.

[2] T. Becker and V. Weispfenning. *Gröbner Basis : A Computational Approach to Commutative Algebra*. Springer-Verlag, NY, 1993. 574 pages.

[3] B. Buchberger. *Ein Algorithmus zum Auffinden der Basiselemente des Restklassenringes nach einem nulldimensionalen Polynomideal*. PhD thesis, Innsbruck, 1965.

[4] B. Buchberger. Ein algorithmishes Kriterium für die Lösbarkeit eines algebraischen Gleichungssystems. *Aequationes Math.*, 4:374–383, 1970.

[5] N. Courtois, A. Klimov, J. Patarin, and A. Shamir. Efficient algorithms for solving overdefined systems of multivariate polynomial equations. In *Advances in cryptology—EUROCRYPT 2000 (Bruges)*, volume 1807 of *Lecture Notes in Computer Science*, pages 392–407. Springer, Berlin, 2000.

[6] N. Courtois and W. Meier. Algebraic attacks on stream ciphers with linear feedback. *Eurocrypt, LNCS*, 2656:345ff, 2003.

[7] N. Courtois and J. Patarin. About the XL algorithm over GF(2). In *Topics in Cryptology—CT-RSA 2003*, volume 2612 of *Lecture Notes in Computer Science*, pages 141–157. Springer, Berlin, 2003.

[8] N. Courtois and J. Pieprzyk. Cryptanalysis of block ciphers with overdefined systems of equations. In *Proceedings of the 8th International Conference on the Theory and Application of Cryptology and Information Security*, pages 267–287. Springer-Verlag, 2002.

[9] D. A. Cox, J. B. Little, and D. O'Shea. *Ideals, Varieties, and Algorithms : An introduction to computational algebraic geometry and commutative algebra.* Springer-Verlag, NY, 2nd edition, 1996. 536 pages.

[10] C. Diem. The XL-algorithm and a conjecture from commutative algebra. unpublished work, Institute for Experimental Mathematics.

[11] J.-C. Faugère. A new efficient algorithm for computing Gröbner bases without reduction to zero ($F_5$). In T. Mora, editor, *ISSAC 2002*, pages 75–83, 2002.

[12] D. Lazard. Résolution des systèmes d'équations algébriques. *Theoretical Computer Science*, 15(1):77–110, 1981.

[13] D. Lazard. Gröbner bases, Gaussian elimination and resolution of systems of algebraic equations. In *Computer algebra (London, 1983)*, volume 162 of *Lecture Notes in Computer Science*, pages 146–156. Springer, Berlin, 1983.

[14] F. Macaulay. Some formulae in elimination. In *proceedings of the London Mathematical Society*, number 33 in 1, pages 3–27, 1902.

[15] J. Patarin. Hidden fields equations (HFE) and isomorphisms of polynomials (IP): Two new families of asymmetric algorithms. *Lecture Notes in Computer Science*, 1070:33–48, 1996.

[16] M. Sugita, M. Kawazoe, and H. Imai. Relation between xl algorithm and groebner bases algorithms. Cryptology ePrint Archive, Report 2004/112, 2004. `http://eprint.iacr.org/`

[17] B.-Y. Yang and J.-M. Chen. Theorical analysis of xl over small fileds. In *Advances in Cryptology, ACISP 2004*, Lecture Notes in Computer Science. Springer, 2004.

# A Proof of the link between the results of XL and $D$-Gröbner basis

Let be $F = \{f_1, \ldots, f_m\}$ a polynomial system. With lemma 1, we assume that XL algorithm uses the lexicographic order for its linearization. We consider the set $F_{XL}$ of all the polynomials computed by the Gaussian elimination. Let us consider $G$ a $D$-Gröbner basis of $F^*$ for the lexicographic order. As the XL algorithm result is not based on homogeneous polynomial, we will find a link between $F_{XL}$ and $G_*$. As we do not have the unicity of Gröbner basis, to compare the two results, we need to reduce the bases.

**Proposition 4** *Let be $F_1$ the reduced basis for $F_{XL}$ and $G_1$ for $G_*$.*
*Then $F_1$=$G_1$.*

**Proof.**

- First, we will prove that $LT(F_1) = \{LT(f) | f \in F_1\} = LT(G_1) = \{LT(g) | g \in (G_1)\}$.
  Let $f \in F_1$, as $F_{XL}$ is a $\mathbb{F}_q$-vector basis of $< \sum_i q_i f_i$ with $degree(q_i f_i) \leq D >$, $\exists q_i, i \in \{1, \ldots, m\}$ such that $f = \sum_i q_i f_i$ and $degree(q_i f_i) \leq D$.
  From the theorem.2, $Z^{D-degree(f)} f^*$ is reduced to zero by $G$. So $f$ is reduced to zero by $G_1$.
  $\exists g \in G_1$ such that $LT(g)$ divides $LT(f)$.
  Moreover, as $g \in G_1$, $g$ is reduced to zero by $G_*$.
  So, from theorem.2, $\exists q_i, i \in \{1, \ldots, m\}$ such that $g = \sum_i q_i f_i$ and $degree(q_i f_i) \leq D$.
  Then $\exists f' \in F_1$ such that $LT(f')$ divides $LT(g)$.
  As $LT(f')$ divides $LT(f)$ and $F_1$ reduced basis, $f' = f$ and $LT(f) = LT(g)$.
  We have $LT(G_1) \subset LT(F_1)$.
  We prove, with the same methods, that $LT(F_1) \subset LT(G_1)$.
  We conclude that $LT(F_1) = LT(G_1)$.
- Let assume $F_1 \neq (G_1)$, $\exists f, g \in F_1, G_1$ such that $f \neq g$.
  Let consider $f - g$, we have two cases :
  1. $HT(f - g) \in M(f)$.
     As $\exists q_i, i \in \{1, \ldots, m\}$ such that $g = \sum_i q_i f_i$ and $degree(q_i f_i) \leq D$. $f - g$ belongs to the vector space $< \sum_i q_i f_i$ with $degree(q_i f_i) \leq D >$.
     So $\exists f' \in F_1$ st $LT(f')$ divides $LT(f - g_*)$. Contradiction with the reduced properties.
  2. $HT(f - g) \in M(g)$.
     As $f$ is reduced to zero by $G_1$, $f - g$ too. Contradiction with the reduced properties of $G_1$.

  We conclude that $F_1 = G_1$.

□

**Remark 1** In this proof, we do not take care of the monomials simplification $X_i^q = X_i$in the first step.

But we have seen that the monomials simplification can be interpreted by the Gaussian elimination of the system with the field equations.

So the reduced bases from the result of two systems'Gaussian elimination are the same if we add for the first (i.e. with monomial simplification) the field equation.

Then we have the proposition :

**Proposition 5** *Let be $F_1$ the reduced basis for $F_{XL} \cup \{X_1^q - X_1, \ldots, X_n^q - X_n\}$ and $G_1$ for $G_*$ where $G$ is the $D$-Gröbner basis of $(F \cup \{X_1^q - X_1, \ldots, X_n^q - X_n\})^*$.*

*Then $F_1$=$G_1$.*

# B Proof of the comparisons of degree of XL and Gröbner basis

## B.1 On the field $\mathbb{F}_2$

**Lemma 2** *Let us $\mathcal{A}_m$ be a system of $m$ multivariate equations with $n$ variables.*

*Let consider $X_1, X_2, \ldots, X_n$ the variables which are eliminated in this order and $\prec$ a monomial ordering on $\mathcal{R}_{2,n}$ so that $X_1 \prec X_2 \prec \cdots \prec X_n$.*

$$\begin{array}{lcl} \textit{XL terminates} & & \textit{XL terminates} \\ & \iff & \textit{for a degree } D \\ \textit{for a degree } D & & \textit{with the order } \prec \end{array}$$

**Proof.**

In the ring $\mathcal{R}_{2,n}$, the only univariate monomials in the variable $X_1$ are 1 and $X_1$. They are the first two monomials of this ordering.

as lemma 1, we can prove that if XL finds a univariate polynomial in $X_1$, it will find it with the monomial ordering $\prec$. □

**Theorem 5** *Let $\mathcal{A}_m^n$ be a polynomial system with $n$ variables, $m$ equations on $\mathbb{F}_2$.*

*If XL algorithm terminates on the system $\mathcal{A}_m^n$ for a degree $D$ eliminated in order $X_1, \ldots, X_n$, the Gröbner basis computation for a DRL monomial ordering $[X_n, \ldots, X_1]$ does not overreach the degree $D$ during the computation.*

**Proof.** The DRL ordering $[X_n, \ldots, X_1]$ satisfied the condition of lemma 2.

Let $g$ be the first univariate polynomial found by XL algorithm for the degree $D$. Assuming the corollary.1, the reduction of $g$ by the $D$-Gröbner basis is zero.

As $g = aX_1 + b$, $a, b \in \mathbb{F}_2$, $g$ can be only reduced by a univariate polynomial

So the $D$-Gröbner basis computed contains a univariate polynomial $X_1 + a_1$.

Moreover the reduction of an $\mathcal{S}$-polynomial of $f$ and $X_i + a_i$ corresponds to the substitution step and does not exceed the degree of $f$.

So, these remarks prove that Gröbner basis computation does not exceed the degree $D$. □

## B.2 On the field $\mathbb{F}_q$

**Theorem 6** *Let be $\mathcal{A}_m^n$ a polynomial system with $n$ variables, $m$ equations.*

*If XL algorithm terminates on the system $\mathcal{A}_m^n$ for a degree $D$, the emulated algorithm terminates too and the Gröbner basis computation satisfies that all computed polynomials have a degree lower than $D$.*

**Proof.**

Suppose that the theorem has been satisfied for the first variable $X_1, \ldots, X_{i-1}$. Let $\tilde{f}_j$ be the substitution of variable $X_k$ by $a_k$ in the polynomial $f_j$, for $k < i$ and $j \in \{1, \ldots, m\}$.

As XL algorithm terminates for the degree $D$, $\exists P_1, \ldots, P_m \in \mathcal{R}_{q,n}$ st $deg(P_j) \leq D - d_j$ and $g \in \mathbb{F}_q[X_i]$ such that

$$g(X_i) = \sum_{j=1}^{m} P_j \tilde{f}_j \text{ in } \mathcal{R}_{q,n}$$

Or $\tilde{f}_j$ is the reduction of all monomial of $f_j$ by $X_j - a_j$.

Then $\exists Q_1, \ldots, Q_{i-1} \in \mathcal{R}_{q,n}$ st $deg(Q_k) \leq D - 1$,

$$g(X_i) = \sum_{j=1}^{m} P_j f_j + \sum_{k=1}^{i-1} Q_k (X_k - a_k) \text{ in } \mathcal{R}_{q,n}$$

Assuming the corollary 1, the reduction of $g$ by the $D$-Gröbner basis $G'$ is zero.

So $\exists \ell \leq D$, the reduction of $X_j^\ell$ by $G'$ and $\{\tilde{X_1^k}, k < \ell\}$ is zero.

Step 2 will stop for the degree $D$ and step 3 finds a polynomial $h \in \mathbb{F}_q[X_1]$, solves it in $\mathbb{F}_q$ and find a solution $a_i$.

This algorithm has found a solution $a_i$ with a computation degree lower than $D$.

We deduce the theorem from above. □

Unité de recherche INRIA Rocquencourt
Domaine de Voluceau - Rocquencourt - BP 105 - 78153 Le Chesnay Cedex (France)

Unité de recherche INRIA Futurs : Domaine de Voluceau - Rocquencourt - BP 105 - 78153 Le Chesnay Cedex (France)
Unité de recherche INRIA Lorraine : LORIA, Technopôle de Nancy-Brabois - Campus scientifique
615, rue du Jardin Botanique - BP 101 - 54602 Villers-lès-Nancy Cedex (France)
Unité de recherche INRIA Rennes : IRISA, Campus universitaire de Beaulieu - 35042 Rennes Cedex (France)
Unité de recherche INRIA Rhône-Alpes : 655, avenue de l'Europe - 38330 Montbonnot-St-Martin (France)
Unité de recherche INRIA Sophia Antipolis : 2004, route des Lucioles - BP 93 - 06902 Sophia Antipolis Cedex (France)

Éditeur
INRIA - Domaine de Voluceau - Rocquencourt, BP 105 - 78153 Le Chesnay Cedex (France)
http://www.inria.fr
ISSN 0249-6399