Comparison between XL and Gröbner Basis Algorithms

G. Ars[1], J.-C. Faugère[2], H. Imai[3], M. Kawazoe[4], M. Sugita[5]

1 University of Rennes, 2 University of Paris 6, 3 University of Tokyo, 4 Osaka Prefecture University, 5 IPA Japan

Algebraic Attack

- Algebraic attacks are among the most efficient attacks for public key cryptosystems, block ciphers and stream ciphers. They try to recover a secret key by solving a system of algebraic equations.
- J. Patarin '95 (applied to Matsumoto-Imai Public Key Scheme)
- For Eurocrypt 2000, N. Courtois, A. Klimov, J. Patarin and A. Shamir presents a new algorithm to solve polynomial systems on finite fields: XL.
- Courtois-Pieprzyk '02 (applied to block ciphers): XSL
- etc.

Goal of this talk

- Goal of this talk : Find a link between XL and Gröbner basis methods.

Goal of this talk

- Goal of this talk : Find a link between XL and Gröbner basis methods.
- Motivations :
  Complexity bound of well-studied Gröbner bases can be extended to XL algorithm.

Goal of this talk

- Goal of this talk : Find a link between XL and Gröbner basis methods.
- Motivations :
  Complexity bound of well-studied Gröbner bases can be extended to XL algorithm.
- Cryptographic results of XL algorithm gave results with Gröbner bases algorithm and conversely.

Goal of this talk

- Goal of this talk : Find a link between XL and Gröbner basis methods.
- Motivations :
  Complexity bound of well-studied Gröbner bases can be extended to XL algorithm.
- Cryptographic results of XL algorithm gave results with Gröbner bases algorithm and conversely.
- Gröbner bases computation is implemented on many programs : very efficient implementation in lastest version of Magma (Magma V2.11 : http://magma.maths.usyd.edu.au/users/allan/gb/)

Gröbner basis algorithm and XL

- Gröbner basis algorithm = a general method to solve a system of algebraic equations
- XL : proposed as an efficient algorithm for algebraic attacks
- A special condition: In cryptographic scheme, a system of algebraic equations we are interested in has a unique solution over its defining field. (XL was proposed as a powerful technique to solve such special systems.)

Gröbner basis algorithm and XL (2)

- Recently, by using the algorithms $\boldsymbol{F_4}$ and $\boldsymbol{F_5}$, $\mathbf{80}$-bit HFE was first cryptanalyzed. (Faugère-Joux ’03)

Gröbner basis algorithm and XL (2)

- Recently, by using the algorithms $\boldsymbol{F_4}$ and $\boldsymbol{F_5}$, $\mathbf{80}$-bit HFE was first cryptanalyzed. (Faugère-Joux ’03)
- Time results with an implementation under Magma are presented on A. Steel’s web page (http://magma.maths.usyd.edu.au/users/allan/gb/).

Gröbner basis algorithm and XL (2)

- Recently, by using the algorithms F_4 and F_5, 80-bit HFE was first cryptanalyzed. (Faugère-Joux '03)
- Time results with an implementation under Magma are presented on A. Steel's web page (http://magma.maths.usyd.edu.au/users/allan/gb/).
- Why did algebraic cryptanalysis based on these Gröbner basis algorithms exceed XL?

Gröbner basis algorithm and XL (2)

- Recently, by using the algorithms F_4 and F_5, 80-bit HFE was first cryptanalyzed. (Faugère-Joux '03)
- Time results with an implementation under Magma are presented on A. Steel's web page (http://magma.maths.usyd.edu.au/users/allan/gb/).
- Why did algebraic cryptanalysis based on these Gröbner basis algorithms exceed XL?
- We give an answer for this question in this presentation.

Main results

- If the XL algorithm terminates, it will also terminate with a lexicographic ordering.

Main results

- If the XL algorithm terminates, it will also terminate with a lexicographic ordering.
- XL can be viewed as a redundant variant of a Gröbner basis algorithm $\boldsymbol{F_4}$.

Main results

- If the XL algorithm terminates, it will also terminate with a lexicographic ordering.
- XL can be viewed as a redundant variant of a Gröbner basis algorithm $\boldsymbol{F_4}$.
- We study the XL algorithm on semi-regular sequences.

Main results

- If the XL algorithm terminates, it will also terminate with a lexicographic ordering.
- XL can be viewed as a redundant variant of a Gröbner basis algorithm F_4.
- We study the XL algorithm on semi-regular sequences.
- We complete this study on generic systems with a comparison of the XL algorithm and the Buchberger algorithm for a cryptosystem HFE.

Gröbner Basis

Need a monomial ordering

Lexicographic Order

$$x_1^{\alpha_1}\dots x_n^{\alpha_n} > x_1^{\beta_1}\dots x_n^{\beta_n}$$

$$\Updownarrow$$

$$\exists i \in \{1,\dots,n\}, \text{st } \forall j < i,$$

$$\alpha_j = \beta_j \ \ \& \ \alpha_i > \beta_i$$

DRL Order

$$x_1^{\alpha_1}\dots x_n^{\alpha_n} \succ x_1^{\beta_1}\dots x_n^{\beta_n}$$

$$\Updownarrow$$

$$\sum_i \alpha_i > \sum_i \beta_i \text{ or}$$

$$\sum_i \alpha_i = \sum_i \beta_i \ \& \ \exists i \in \{1,\dots,n\},$$

$$\text{st } \forall j > i, \alpha_j = \beta_j \ \ \& \ \alpha_i < \beta_i$$

Gröbner Basis

Need a monomial ordering

Lexicographic Order	DRL Order
$x_1^{\alpha_1}\dots x_n^{\alpha_n} > x_1^{\beta_1}\dots x_n^{\beta_n}$	$x_1^{\alpha_1}\dots x_n^{\alpha_n} \succ x_1^{\beta_1}\dots x_n^{\beta_n}$
$\Updownarrow$	$\Updownarrow$
$\exists i \in \{1,\dots,n\}, \text{st } \forall j < i,$	$\sum_i \alpha_i > \sum_i \beta_i$ or
$\alpha_j = \beta_j \ \& \ \alpha_i > \beta_i$	$\sum_i \alpha_i = \sum_i \beta_i \ \& \ \exists i \in \{1,\dots,n\},$
	$\text{st } \forall j > i, \alpha_j = \beta_j \ \& \ \alpha_i < \beta_i$

Leading Monomial of a polynomial : $LM(P)$

Gröbner Basis

Need a monomial ordering

Lexicographic Order	DRL Order
$x_1^{\alpha_1} \dots x_n^{\alpha_n} > x_1^{\beta_1} \dots x_n^{\beta_n}$	$x_1^{\alpha_1} \dots x_n^{\alpha_n} \succ x_1^{\beta_1} \dots x_n^{\beta_n}$
$\Updownarrow$	$\Updownarrow$
$\exists i \in \{1, \dots, n\}, \text{st } \forall j < i,$	$\sum_i \alpha_i > \sum_i \beta_i$ or
$\alpha_j = \beta_j \ \ \& \ \alpha_i > \beta_i$	$\sum_i \alpha_i = \sum_i \beta_i \ \& \ \exists i \in \{1, \dots, n\},$
	$\text{st } \forall j > i, \alpha_j = \beta_j \ \ \& \ \alpha_i < \beta_i$

Leading Monomial of a polynomial : $LM(P)$ Example:

$P = x_1^6 x_2^5 x_3^3 x_4^6 + x_1^4 x_2^9 x_3^4 x_4^5 + x_1^4 x_2^{10},$

Gröbner Basis

Need a monomial ordering

Lexicographic Order	DRL Order
$x_1^{\alpha_1}\dots x_n^{\alpha_n} > x_1^{\beta_1}\dots x_n^{\beta_n}$	$x_1^{\alpha_1}\dots x_n^{\alpha_n} \succ x_1^{\beta_1}\dots x_n^{\beta_n}$
$\Updownarrow$	$\Updownarrow$
$\exists i \in \{1,\dots,n\}$, st $\forall j < i$, $\alpha_j = \beta_j$ & $\alpha_i > \beta_i$	$\sum_i \alpha_i > \sum_i \beta_i$ or $\sum_i \alpha_i = \sum_i \beta_i$ & $\exists i \in \{1,\dots,n\}$, st $\forall j > i, \alpha_j = \beta_j$ & $\alpha_i < \beta_i$

Leading Monomial of a polynomial : $LM(P)$ Example: $P = x_1^6 x_2^5 x_3^3 x_4^6 + x_1^4 x_2^9 x_3^4 x_4^5 + x_1^4 x_2^{10}$, for Lexicographic order

Gröbner Basis

Need a monomial ordering

Lexicographic Order	DRL Order
$x_1^{\alpha_1} \dots x_n^{\alpha_n} > x_1^{\beta_1} \dots x_n^{\beta_n}$	$x_1^{\alpha_1} \dots x_n^{\alpha_n} \succ x_1^{\beta_1} \dots x_n^{\beta_n}$
$\Updownarrow$	$\Updownarrow$
$\exists i \in \{1, \dots, n\}, \text{st } \forall j < i,$	$\sum_i \alpha_i > \sum_i \beta_i$ or
$\alpha_j = \beta_j \;\; \& \;\alpha_i > \beta_i$	$\sum_i \alpha_i = \sum_i \beta_i \;\&\; \exists i \in \{1, \dots, n\},$
	$\text{st } \forall j > i, \alpha_j = \beta_j \;\; \& \;\alpha_i < \beta_i$

Leading Monomial of a polynomial : $LM(P)$ Example:

$P = x_1^6 x_2^5 x_3^3 x_4^6 + x_1^4 x_2^9 x_3^4 x_4^5 + x_1^4 x_2^{10}$, for DRL order

Gröbner Basis

Need a monomial ordering

Lexicographic Order	DRL Order
$x_1^{\alpha_1}\dots x_n^{\alpha_n} > x_1^{\beta_1}\dots x_n^{\beta_n}$	$x_1^{\alpha_1}\dots x_n^{\alpha_n} \succ x_1^{\beta_1}\dots x_n^{\beta_n}$
$\Updownarrow$	$\Updownarrow$
$\exists i \in \{1,\dots,n\}$, st $\forall j < i$, $\alpha_j = \beta_j$ & $\alpha_i > \beta_i$	$\sum_i \alpha_i > \sum_i \beta_i$ or $\sum_i \alpha_i = \sum_i \beta_i$ & $\exists i \in \{1,\dots,n\}$, st $\forall j > i, \alpha_j = \beta_j$ & $\alpha_i < \beta_i$

Leading Monomial of a polynomial : $LM(P)$ Example:
$P = x_1^6x_2^5x_3^3x_4^6 + x_1^4x_2^9x_3^4x_4^5 + x_1^4x_2^{10}$, for DRL order

The $\mathcal{S}$-polynomial of a pair of polynomials: $\mathcal{S}pol(f,g) = \frac{lcm(LM(f),LM(g))}{LT(f)}.f - \frac{lcm(LM(f),LM(g))}{LT(g)}.g$

Gröbner basis (2)

Gröbner basis : $G = \{g_1, \ldots, g_s\}$ of an ideal I is a Gröbner basis
if for all $f \in I$, there is g_i st $LM(g_i)$ divide $LM(f)$.
$G = \{g_1, \ldots, g_s\}$ of I is a Gröbner basis iff $\forall i, j$,
$\mathcal{S}pol(g_i, g_j) \xrightarrow[G]{} 0$.

Gröbner basis (2)

Gröbner basis : $G = \{g_1, \ldots, g_s\}$ of an ideal I is a Gröbner basis
if for all $f \in I$, there is g_i st $LM(g_i)$ divide $LM(f)$.
$G = \{g_1, \ldots, g_s\}$ of I is a Gröbner basis iff $\forall i, j$, $\mathcal{S}pol(g_i, g_j) \xrightarrow[G]{} 0$.

D-Gröbner basis : $G = \{g_1, \ldots, g_s\}$, g_i homogeneous, of I is a D-Gröbner basis iff $\forall i \neq j$ and $degree(lcm(LM(g_i), LM(g_j))) \leq D$,

$$\mathcal{S}pol(g_i, g_j) \xrightarrow[G]{} 0.$$

Solving systems over finite fields

Find solution of a system
$f_1(z_1, \ldots, z_n) = 0, \ldots, f_m(z_1, \ldots, z_n) = 0$ with
$(z_1, \ldots, z_n)$ in the field $\mathbb{F}_q$.

$$\Updownarrow$$

Consider the ideal I generated by $f_1, \ldots, f_m$ and field equations $X_i^q - X_i$.

Solving systems over finite fields

Find solution of a system
$f_1(z_1, \ldots, z_n) = 0, \ldots, f_m(z_1, \ldots, z_n) = 0$ with
$(z_1, \ldots, z_n)$ in the field $\mathbb{F}_q$.

$$\Updownarrow$$

Consider the ideal I generated by $f_1, \ldots, f_m$ and field equations $X_i^q - X_i$.

Important cases :

- The field $\mathbb{F}_2$.
- The field $\mathbb{F}_p$, $p \gg n$ and p prime number, the field equation are useless.

Gröbner basis and Gaussian Elimination

D. Lazard,

Gröbner bases, Gaussian Elimination and Resolution of Systems of Algebraic Equations,

1983.

Let us consider the Macaulay matrix for a degree $\leq d$.

$$\mathcal{M}_{d,m} = \begin{array}{c} \\ m'_1 \times f_{i_1} \\ m'_2 \times f_{i_2} \\ m'_3 \times f_{i_3} \\ \dots \end{array} \begin{array}{c} m_1 \succ m_2 \succ m_3 \dots \succ m_{L-2} \succ m_{L-1} \succ m_L \\ \left(\begin{array}{c} \dots \\ \dots \\ \dots \\ \dots \end{array} \right) \end{array}$$

with $i_1, i_2, i_3, \dots \leq m$ and $degree(m'_k) \leq d - degree(f_{i_k})$.

For d big enough, a Gaussian Elimination give a Gröbner basis.

Gröbner basis and Gaussian Elimination

D. Lazard,
Gröbner bases, Gaussian Elimination and Resolution of Systems of Algebraic Equations,
1983.

Let us consider the Macaulay matrix for a degree $\leq d$.

$$\mathcal{M}_{d,m} = \begin{array}{c} \\ m'_1 \times f_{i_1} \\ m'_2 \times f_{i_2} \\ m'_3 \times f_{i_3} \\ \dots \end{array} \begin{array}{c} m_1 \succ m_2 \succ m_3 \dots \succ m_{L-2} \succ m_{L-1} \succ m_L \\ \left(\begin{array}{c} \dots \\ \dots \\ \dots \\ \dots \end{array} \right) \end{array}$$

with $i_1, i_2, i_3, \dots \leq m$ and $degree(m'_k) \leq d - degree(f_{i_k})$.
For d big enough, a Gaussian Elimination give a Gröbner basis.
Theorem : Let be $< f_1, \dots, f_m >,\ m \leq n$, a regular sequence, $(X_1, \dots, X_n)$ generic coordonates, then a Gröbner basis with a DRL order is given with

$$d \leq d_1 + \cdots + d_m - n + 1.$$

F_4 and F_5 algorithm

- F_4 algorithm :
  Simultaneous reduction of all $\mathcal{S}-$polynomials.
  Combinaison of Buchberger criteria and very efficient linear algebra.

F_4 and F_5 algorithm

- F_4 algorithm :
  Simultaneous reduction of all $\mathcal{S}-$polynomials.
  Combinaison of Buchberger criteria and very efficient linear algebra.

- F_5 algorithm :
  Construct a matrice iteratively on *the degree* and on the *number of equations* and replace Buchberger criteria with new criteria to avoid reduction to zero

Semi-regular sequences

M. Bardet,J.-C. Faugère and B. Salvy

Complexity of Gröbner basis computation for Semi-regular Overdetermined sequences over GF(2) with solutions in GF(2).

Extend regular sequence on overdeterminate systems

Definition of a new familly of systems : Semi-regular sequences

Semi-regular sequences

M. Bardet,J.-C. Faugère and B. Salvy

Complexity of Gröbner basis computation for Semi-regular Overdetermined sequences over GF(2) with solutions in GF(2).

Extend regular sequence on overdeterminate systems

Definition of a new familly of systems : Semi-regular sequences

Conjecture : Almost all systems are a semi-regular sequences (random, generic,. . .).

Semi-regular sequences

M. Bardet,J.-C. Faugère and B. Salvy

Complexity of Gröbner basis computation for Semi-regular Overdetermined sequences over GF(2) with solutions in GF(2).

Extend regular sequence on overdeterminate systems

Definition of a new familly of systems : Semi-regular sequences

Conjecture : Almost all systems are a semi-regular sequences (random, generic,. . .).

- Matrix constructed by F_5 have full rank.

Semi-regular sequences

M. Bardet,J.-C. Faugère and B. Salvy

Complexity of Gröbner basis computation for Semi-regular Overdetermined sequences over GF(2) with solutions in GF(2).

Extend regular sequence on overdeterminate systems

Definition of a new familly of systems : Semi-regular sequences

Conjecture : Almost all systems are a semi-regular sequences (random, generic,. . .).

- Matrix constructed by F_5 have full rank.
- Example : For a system of n equation with n variables on $\mathbb{F}_2$, asymptotic degree reached :

$$d \simeq \frac{n}{11,11} + 1.00n^{\frac{1}{3}} + \mathcal{O}(\frac{1}{n^{\frac{1}{3}}})$$

The XL algorithm

Algorithm (The XL algorithm). *For a positive integer D, execute the following steps:*

- **Multiply:** Generate all the products $\prod_{j=1}^{r} x_{\ell_j} * f_i \in \mathcal{I}_{\mathcal{A}}$ with $r \leq D - \deg(f_i)$.
- **Linearize:** Consider each monomial in the x_i of *degree* $\leq D$ as a new variable and perform the Gaussian elimination on the equations obtained in Step 1. The ordering on the monomials must be such that all the terms containing one variable (say x_1) are eliminated last.
- **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).
- **Repeat:** Simplify the equations and repeat the process to find the values of the other variables.

Remark

- We can replace Step 1 of the XL algorithm by considering f_i^* the *homogenization* of f_i: $f_i^* = Z^d f(\frac{x_1}{Z}, \ldots, \frac{x_n}{Z}) \in k[\mathrm{x}, Z]$ and products mf_i^* with m a monomial with degree $D - \deg(f_i^*)$.
- All the computation is exactly the same. So the behavior of XL is the same on the homogenization of the system $\mathcal{A}$ as on $\mathcal{A}$.

Remarks

The two first steps correspond to methods in article :

D. Lazard, *Gröbner bases, Gaussian Elimination and Resolution of Systems of Algebraic Equations*, 1983.

Remarks

The two first steps correspond to methods in article :

D. Lazard,*Gröbner bases, Gaussian Elimination and Resolution of Systems of Algebraic Equations*, 1983.

Implementation in Magma to make practical test.

Remarks

The two first steps correspond to methods in article :

D. Lazard, *Gröbner bases, Gaussian Elimination and Resolution of Systems of Algebraic Equations*, 1983.

Implementation in Magma to make practical test.

Partial monomial order used in XL algorithm

Lemma :

$$\begin{array}{c}\text{XL terminates}\\ \text{for a degree } \boldsymbol{D}\end{array} \iff \begin{array}{c}\text{XL terminates}\\ \text{for a degree } \boldsymbol{D}\\ \text{with a Lexicographic order}\end{array}$$

XL computation and D-Gröbner basis

Lemma :

$\{f_1, \ldots, f_m\}$ ⟶ list of polynomials

Step 1&2 for XL

XL computation and D-Gröbner basis

Lemma :

$\{f_1, \ldots, f_m\}$ → list of polynomials

Step 1&2 for XL

homogenization

$\{f_1^*, \ldots, f_m^*\}$

homogenization of f : $f^* = z^d f(\frac{x_1}{z}, \ldots, \frac{x_n}{z})$ with $d = degree(f)$.

XL computation and D-Gröbner basis

Lemma :

$\{f_1, \ldots, f_m\}$ → list of polynomials

Step 1&2 for XL

homogenization ↓

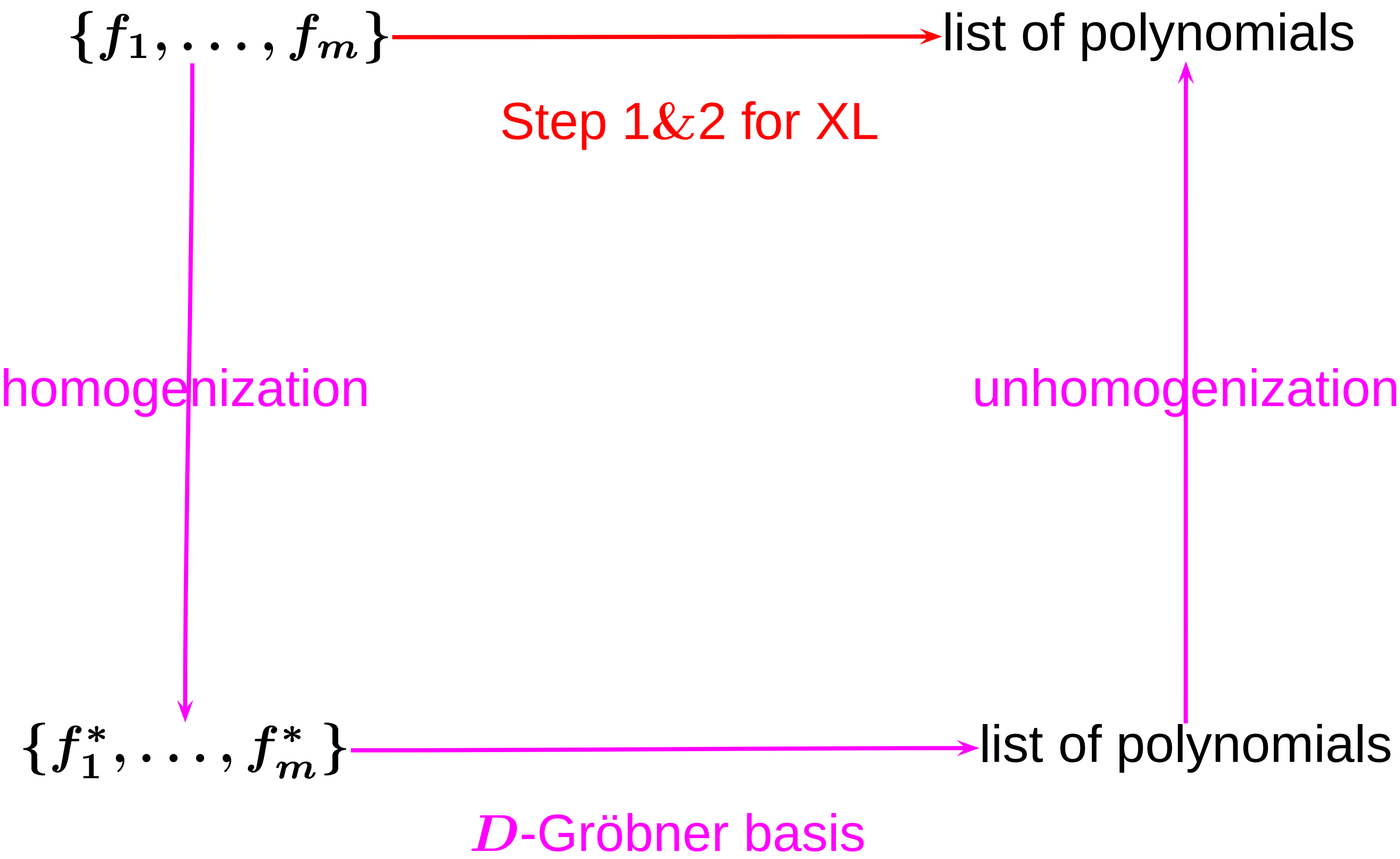
$\{f_1^*, \ldots, f_m^*\}$ → list of polynomials

D-Gröbner basis

homogenization of f : $f^* = z^d f(\frac{x_1}{z}, \ldots, \frac{x_n}{z})$ with $d = degree(f)$.

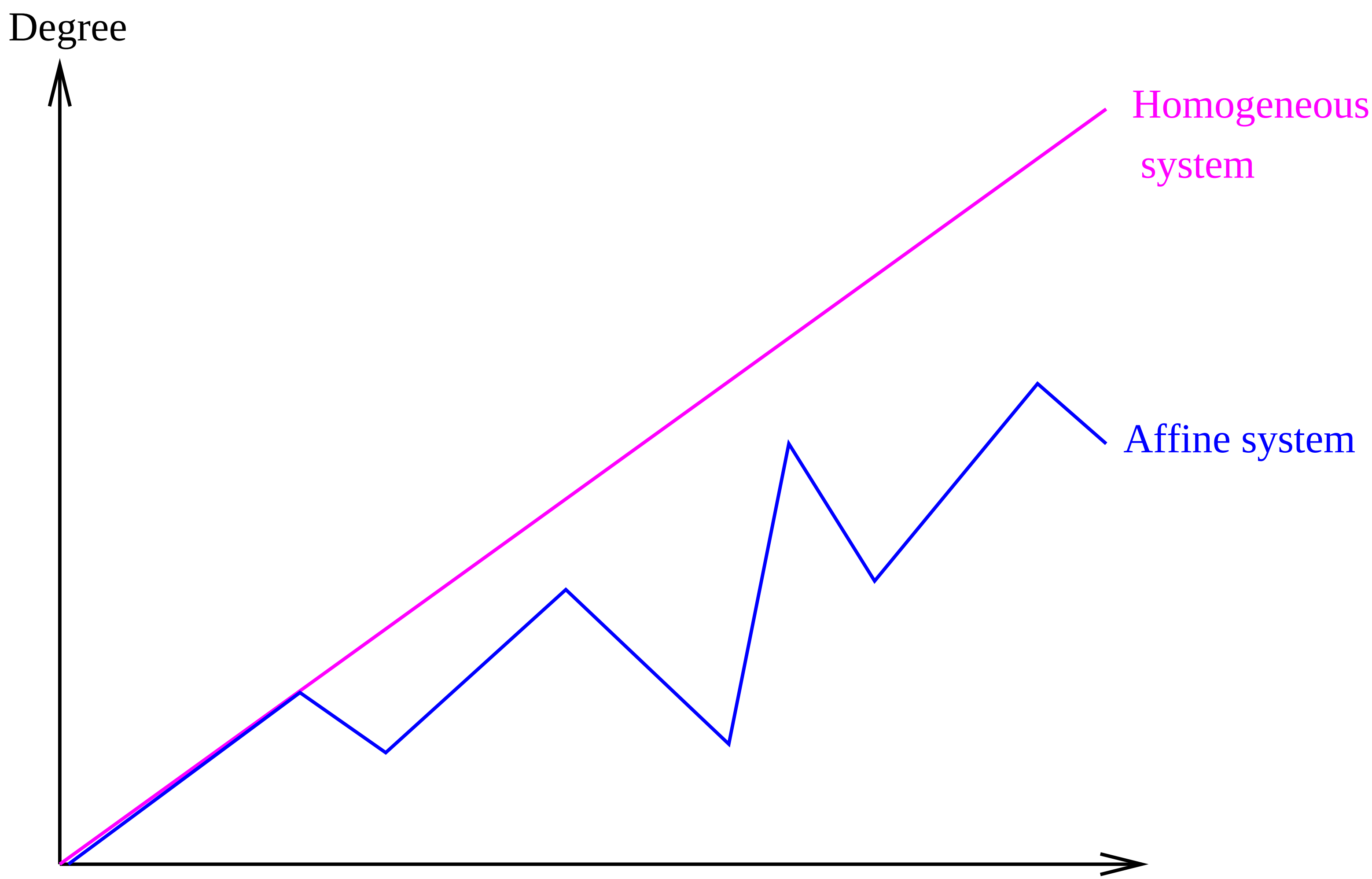
XL computation and D-Gröbner basis

Lemma :



homogenization of f : $f^* = z^d f(\frac{x_1}{z}, \ldots, \frac{x_n}{z})$ with $d = degree(f)$.

Homogeneous/Affine system



Behavior of degree during Gröbner basis computation

Pre-assumption of the XL algorithm

Condition. 1
The system $\boldsymbol{\mathcal{A}}$ *has only one solution* $(\boldsymbol{x_1, \ldots, x_n}) = (\boldsymbol{a_1, \ldots, a_n})$ *in* $\boldsymbol{k^n}$*. (i.e.* $\boldsymbol{\mathcal{A}}$ *has a solution* $(\boldsymbol{a_1, \ldots, a_n})$ *in* $\boldsymbol{k^n}$ *and no other solution in* $\boldsymbol{k^n}$*.)*

Pre-assumption of the XL algorithm

Condition. 1
The system $\boldsymbol{\mathcal{A}}$ *has only one solution* $(\boldsymbol{x_1, \ldots, x_n}) = (\boldsymbol{a_1, \ldots, a_n})$ *in* $\boldsymbol{k^n}$*. (i.e.* $\boldsymbol{\mathcal{A}}$ *has a solution* $(\boldsymbol{a_1, \ldots, a_n})$ *in* $\boldsymbol{k^n}$ *and no other solution in* $\boldsymbol{k^n}$*.)*

- Most stream ciphers will satisfy **Condition. 1** with sufficiently large number of sequences

Pre-assumption of the XL algorithm

Condition. 1
The system $\boldsymbol{\mathcal{A}}$ *has only one solution* $(\boldsymbol{x_1, \ldots, x_n}) = (\boldsymbol{a_1, \ldots, a_n})$ *in* $\boldsymbol{k^n}$*. (i.e.* $\boldsymbol{\mathcal{A}}$ *has a solution* $(\boldsymbol{a_1, \ldots, a_n})$ *in* $\boldsymbol{k^n}$ *and no other solution in* $\boldsymbol{k^n}$*.)*

- Most stream ciphers will satisfy **Condition. 1** with sufficiently large number of sequences
- HFE satisfies **Condition. 1** with only 1 pair of (P/C).

Pre-assumption of the XL algorithm

Condition. 2
The reduced Gröbner basis of the ideal
$\widetilde{\mathcal{I}}_{\mathcal{A}} = \langle f_1, \ldots, f_m, x_1^q - x_1, \ldots, x_n^q - x_n \rangle$ *is*
$\{x_1 - a_1, \ldots, x_n - a_n\}$.

Pre-assumption of the XL algorithm

Condition. 2
The reduced Gröbner basis of the ideal $\widetilde{\mathcal{I}}_{\mathcal{A}} = \langle f_1, \ldots, f_m, x_1^q - x_1, \ldots, x_n^q - x_n \rangle$ *is* $\{x_1 - a_1, \ldots, x_n - a_n\}$.

Under **Condition. 2**, Gröbner bases may be obtained easily.

Pre-assumption of the XL algorithm

Theorem. *Let* $\mathcal{A}$ *be a system of multivariate equations* $f_j = 0$, $j = 1, 2, \ldots, m$ *in* $k[x_1, \ldots, x_n]$ *with* $k = \mathbb{F}_q$*.* *Let* $\widetilde{\mathcal{I}}_{\mathcal{A}}$ *be the ideal* $\langle f_1, \ldots, f_m, x_1^q - x_1, \ldots, x_n^q - x_n \rangle$*.* *Then,*

a solution $(x_1, \ldots, x_n) = (a_1, \ldots, a_n) \in k^n$
of $\mathcal{A}$ is unique in k^n

$$\Updownarrow$$

$$\widetilde{\mathcal{I}}_{\mathcal{A}} = \langle x_1 - a_1, \ldots, x_n - a_n \rangle.$$

Pre-assumption of the XL algorithm

Theorem. *Let* $\mathcal{A}$ *be a system of multivariate equations* $f_j = 0$, $j = 1, 2, \ldots, m$ *in* $k[x_1, \ldots, x_n]$ *with* $k = \mathbb{F}_q$. *Let* $\widetilde{\mathcal{I}}_{\mathcal{A}}$ *be the ideal* $\langle f_1, \ldots, f_m, x_1^q - x_1, \ldots, x_n^q - x_n \rangle$. *Then,*

a solution $(x_1, \ldots, x_n) = (a_1, \ldots, a_n) \in k^n$
of $\mathcal{A}$ is unique in k^n

$$\Updownarrow$$

$$\widetilde{\mathcal{I}}_{\mathcal{A}} = \langle x_1 - a_1, \ldots, x_n - a_n \rangle.$$

i.e.

Condition. 1 $\Longleftrightarrow$ Condition. 2

Various implementations of XL

- Begin with $D = 1$. Do XL described as in Definition for $\mathcal{A}$. If you cannot obtain the solution, set $D := D + 1$ and do XL again for $\mathcal{A}$ with the new D: Simple

Various implementations of XL

- Begin with $D = 1$. Do XL described as in Definition for $\mathcal{A}$. If you cannot obtain the solution, set $D := D + 1$ and do XL again for $\mathcal{A}$ with the new D: Simple
- Begin with $D = 1$. Iterate 'Multiply' and 'Linearize' described as in Definition for $\mathcal{A}$ by adding new equations obtained by 'Linearize' to $\mathcal{A}$. If you cannot solve the resulting system, then return to the original $\mathcal{A}$, set $D := D + 1$ and iterate the same procedure as for $D = 1$. Repeat until you obtain the solution: Iterative

Various implementations of XL

- Begin with $D = 1$. Do XL described as in Definition for $\mathcal{A}$. If you cannot obtain the solution, set $D := D + 1$ and do XL again for $\mathcal{A}$ with the new D: Simple
- Begin with $D = 1$. Iterate 'Multiply' and 'Linearize' described as in Definition for $\mathcal{A}$ by adding new equations obtained by 'Linearize' to $\mathcal{A}$. If you cannot solve the resulting system, then return to the original $\mathcal{A}$, set $D := D + 1$ and iterate the same procedure as for $D = 1$. Repeat until you obtain the solution: Iterative
- Begin with $D = 1$. Do XL described as in Definition for $\mathcal{A}$. If you cannot obtain the solution, then set $D := D + 1$, replace $\mathcal{A}$ by the resulting system obtained by 'Linearize' in the previous XL and do XL again for the new $\mathcal{A}$ and D. Repeat until you obtain the solution: Incremental

Various implementations of XL

- Begin with $D = 1$. Do XL described as in Definition for $\mathcal{A}$. If you cannot obtain the solution, set $D := D + 1$ and do XL again for $\mathcal{A}$ with the new D: Simple
- Begin with $D = 1$. Iterate 'Multiply' and 'Linearize' described as in Definition for $\mathcal{A}$ by adding new equations obtained by 'Linearize' to $\mathcal{A}$. If you cannot solve the resulting system, then return to the original $\mathcal{A}$, set $D := D + 1$ and iterate the same procedure as for $D = 1$. Repeat until you obtain the solution: Iterative
- Begin with $D = 1$. Do XL described as in Definition for $\mathcal{A}$. If you cannot obtain the solution, then set $D := D + 1$, replace $\mathcal{A}$ by the resulting system obtained by 'Linearize' in the previous XL and do XL again for the new $\mathcal{A}$ and D. Repeat until you obtain the solution: Incremental
- Begin with $D = 1$. Iterate 'Multiply' and 'Linearize' described as in Definition for $\mathcal{A}$ by adding new equations obtained by 'Linearize' to $\mathcal{A}$. If you cannot solve the resulting system $\mathcal{A}'$, then replace $\mathcal{A}$ by $\mathcal{A}'$, set $D := D + 1$ and iterate the same procedure as for $D = 1$. Repeat until you obtain the solution: Both iterative and incremental

F4-like Representation of XL algorithm

Definition. *A critical pair of two polynomials* (f_i, f_j) *is an element* $M^2 \times k[\mathbf{x}] \times M \times k[\mathbf{x}]$, $Pair(f_i, f_j) := (lcm_{ij}, t_i, f_i, t_j, f_j)$ *such that*

$$\begin{aligned} lcm(Pair(f_i, f_j)) &= lcm_{ij} = \mathrm{LM}(t_i f_i) = \mathrm{LM}(t_j f_j) \\ &= lcm(\mathrm{LM}(f_i), \mathrm{LM}(f_j)). \end{aligned}$$

(2) For a critical pair $p_{ij} = Pair(f_i, f_j)$, $\deg(lcm_{ij})$ *is called the degree of* p_{ij} *and denoted by* $\deg(p_{ij})$. *Let* P *be a list of critical pairs. For* $p = Pair(f, g) \in P$ *and* $d \in \mathbb{N}$, *we define two functions* $XLLeft(p, d) = \{(t, f) | t \in M, \deg(t * f) \leq d\}$, *and* $XLRight(p, d) = \{(t, g) | t \in M, \deg(t * g) \leq d\}$. *We write* $XLLeft(P, d) = \bigcup_{p \in P} XLLeft(p, d)$ *and* $XLRight(P, d) = \bigcup_{p \in P} XLRight(p, d)$.
$Left(p_{ij}) = (t_i, f_i)$, $Right(p_{ij}) = (t_j, f_j)$.
$Left(P) = \bigcup_{p_{ij} \in P} Left(p_{ij})$, $Right(P) = \bigcup_{p_{ij} \in P} Right(p_{ij})$.

F4-like Representation of XL algorithm

For a list of critical pairs P and a positive integer $d \in \mathbb{N}$, we set

$$Sel_{XL}(P, d) := \{p \in P \mid \deg(lcm(p)) \leq d\}.$$

For a list P of critical pairs of a given set,

$$Sel_{F_4}(P) := \{p \in P | \deg(lcm(p)) = d\}$$

where $d := min\{\deg(lcm(p)), p \in P\}$.

F4-like Representation of XL

F_4-like representation of XL	F_4
Input: $\begin{cases} F : \text{a finite subset of } k[\mathbf{x}] \\ Sel = Sel_{XL} \end{cases}$	Input: $\begin{cases} F : \text{a finite subset of } k[\mathbf{x}] \\ Sel = Sel_{F_4} \end{cases}$
Output: a finite subset of $k[\mathbf{x}]$.	Output: a finite subset of $k[\mathbf{x}]$.
$G := F$, $\tilde{F}_0^+ := F$ and $d := 0$	$G := F$, $\tilde{F}_0^+ := F$ and $d := 0$
$P := \{Pair(f, g) \mid f, g \in G$ with $f \neq g\}$	$P := \{Pair(f, g) \mid f, g \in G$ with $f \neq g\}$
While $P \neq \phi$ Do	While $P \neq \phi$ Do
$d := d + 1$	$d := d + 1$
$P_d := Sel(P, d)$	$P_d := Sel(P, d)$
$L_d :=$ $\mathbf{XLLeft}(P, d) \cup \mathbf{XLRight}(P, d)$	$L_d :=$ $\mathbf{Left}(P_d) \cup \mathbf{Right}(P_d)$
$P := P \setminus P_d$	$P := P \setminus P_d$
$\tilde{F}_d^+ := \mathbf{Reduction}(L_d)$	$\tilde{F}_d^+ := \mathbf{Reduction}(L_d)$
For $h \in \tilde{F}_d^+$ Do	For $h \in \tilde{F}_d^+$ Do
$P := P \cup \{Pair(h, g) \mid g \in G\}$	$P := P \cup \{Pair(h, g) \mid g \in G\}$
$G := G \cup \{h\}$	$G := G \cup \{h\}$
Return G	Return G

F4-like Representation of XL algorithm

Reduction (same in F_4-like representation of XL and in F_4)

Input: a finite subset L of $M \times k[\mathbf{x}]$

Output: a finite subset of $k[\mathbf{x}]$ (possibly an empty set).

$F :=$ **Symbolic Preprocessing**(L)

$\tilde{F} :=$ Reduction to Row Echelon Basis of F w.r.t. $<$

$\tilde{F}^+ := \{f \in \tilde{F} | \mathrm{LM}(f) \notin \mathrm{LM}(F)\}$

Return $\tilde{F}^+$

F4-like Representation of XL algorithm

Symbolic Preprocessing

F_4-like representation of XL	F_4
Input: a finite subset L of $M \times k[\mathbf{x}]$	Input: a finite subset L of $M \times k[\mathbf{x}]$
Output: a finite subset of $k[\mathbf{x}]$	Output: a finite subset of $k[\mathbf{x}]$
$F := \{t * f \mid (t, f) \in L\}$	$F := \{t * f \mid (t, f) \in L\}$
Return F.	$Done := \mathrm{LM}(F)$
	While $M(F) \neq Done$ Do
	$Done := Done \cup \{m\}$
	$(m \in M(F) \setminus Done)$
	If m is top reducible modulo G Then
	$m = m' * \mathrm{LM}(f)$
	for some $f \in G$ and some $m' \in M$
	$F := F \cup \{m' * f\}$
	Return F

XL and Gröbner bases algorithms

Theorem. *Let $\boldsymbol{F}$ be a finite set of polynomials in $\boldsymbol{k}[\mathbf{x}]$. Then XL algorithm computes a Gröbner basis $\boldsymbol{G}$ for the ideal $\langle \boldsymbol{F} \rangle$ in $\boldsymbol{k}[\mathbf{x}]$ such that $\boldsymbol{F} \subseteq \boldsymbol{G}$.*

Semi-regular sequences

Consider a system of m quadratic equations on n variables

Semi-regular sequences

Consider a system of m quadratic equations on n variables

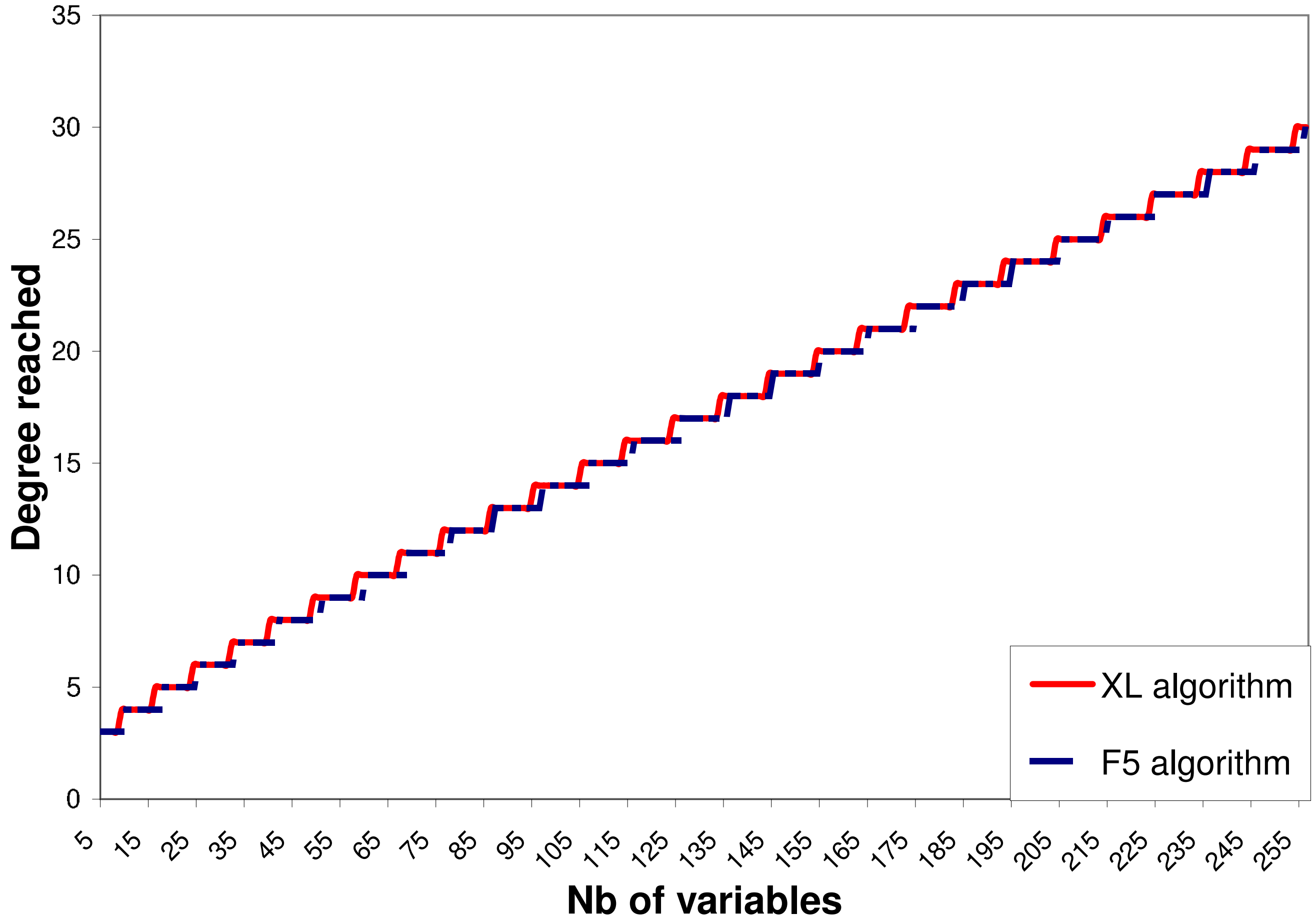
For F_5 algorithm :

$$\frac{(1+y)^n}{(1+y^2)^m}$$

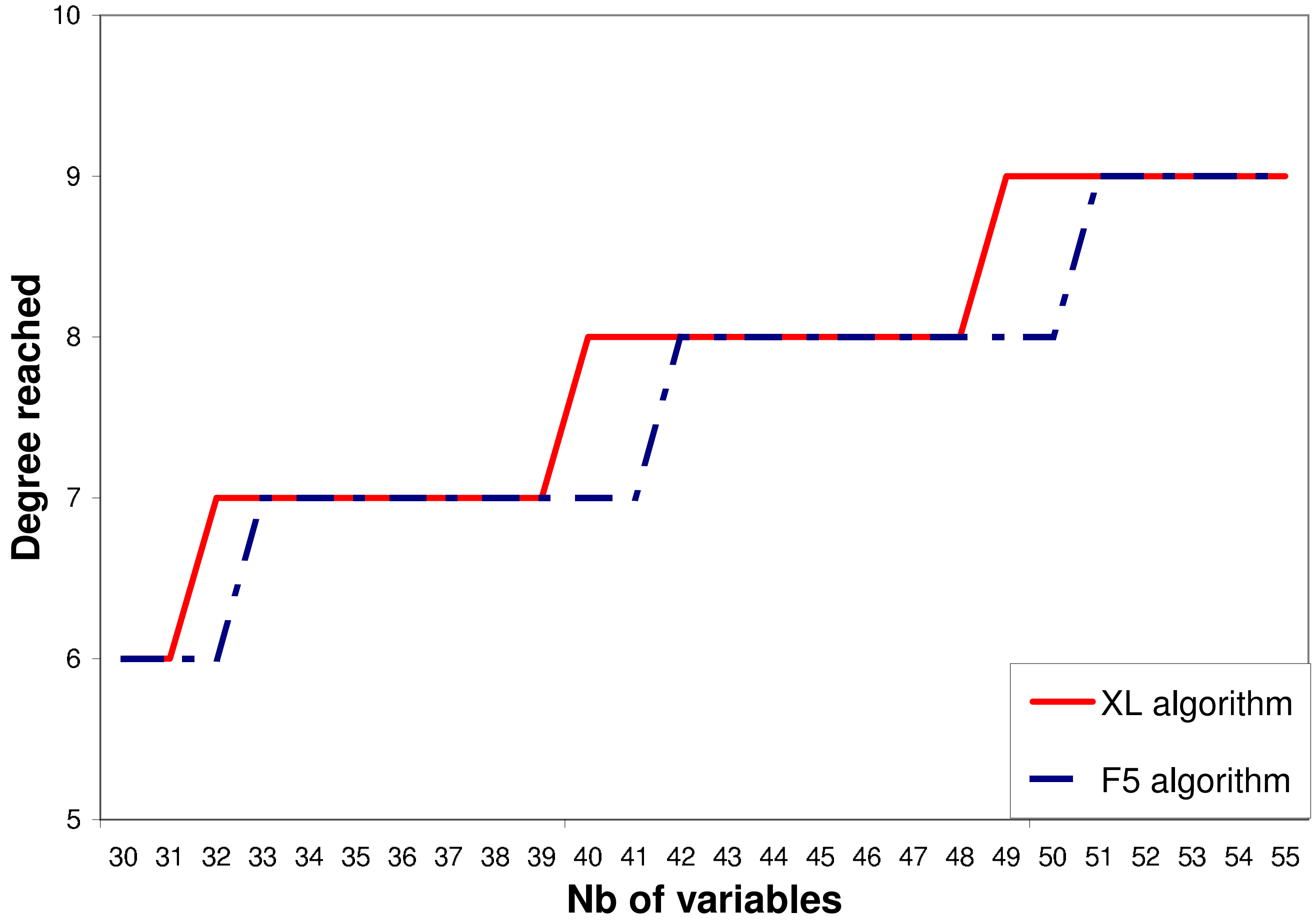
For XL algorithm :

$$\frac{(1+y)^n}{(1-y)(1+y^2)^m} - \frac{1+y}{1-y}$$

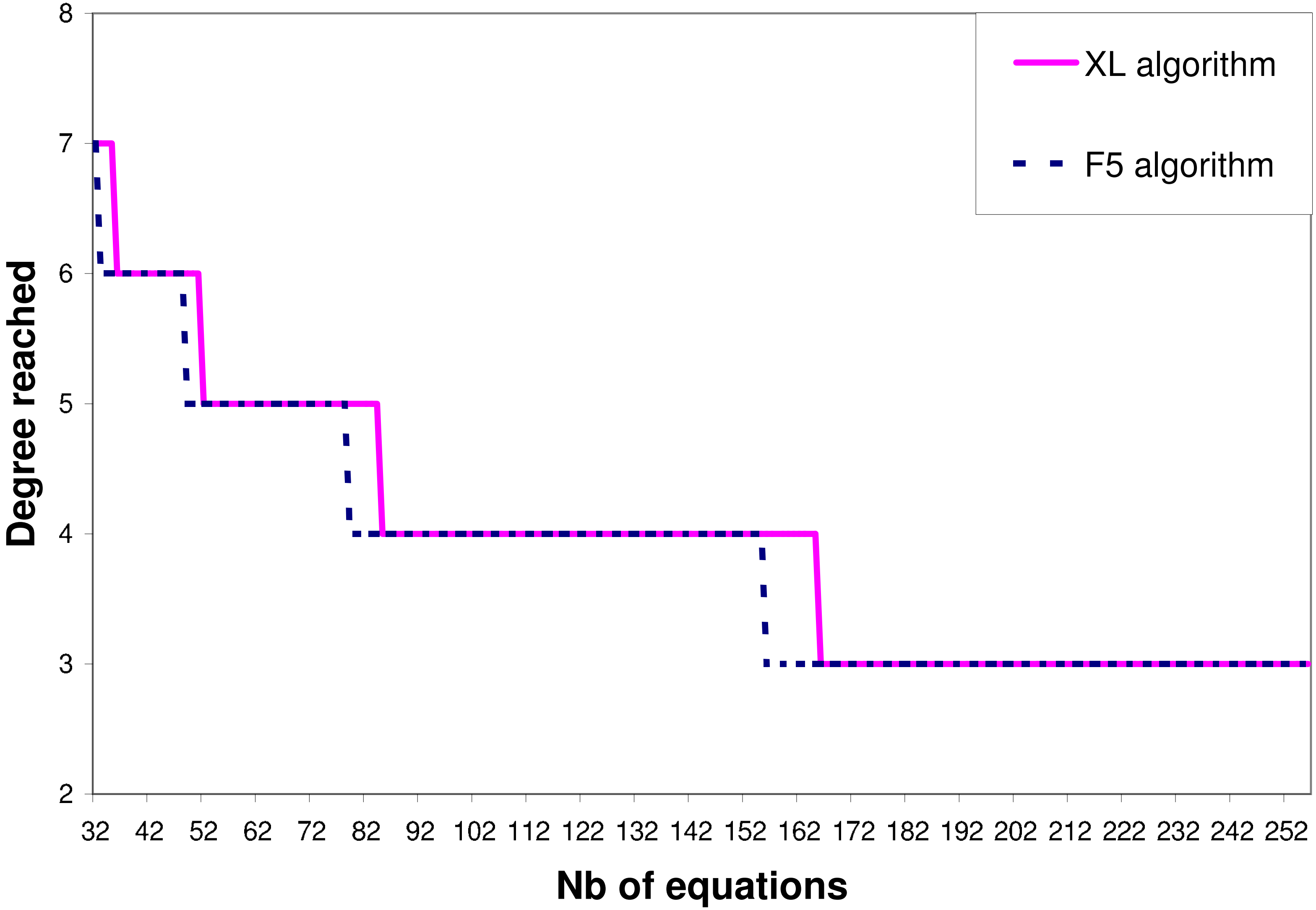
Semi-regular sequences: $m = n + 2$



Semi-regular sequences: $m = n + 2$



Semi-regular sequences: $m = n + 2$



Semi-regular sequences: $m = n + 2$

Complexity : $(\text{size of the matrix})^{\boldsymbol{w}}$

XL algorithm

Matrix

size

Semi-regular sequences: $m = n + 2$

Complexity : $(\text{size of the matrix})^{\boldsymbol{w}}$

XL algorithm		$\boldsymbol{F_4}$ algorithm
Matrix size	$\geq$	Matrix size

Buchberger criteria

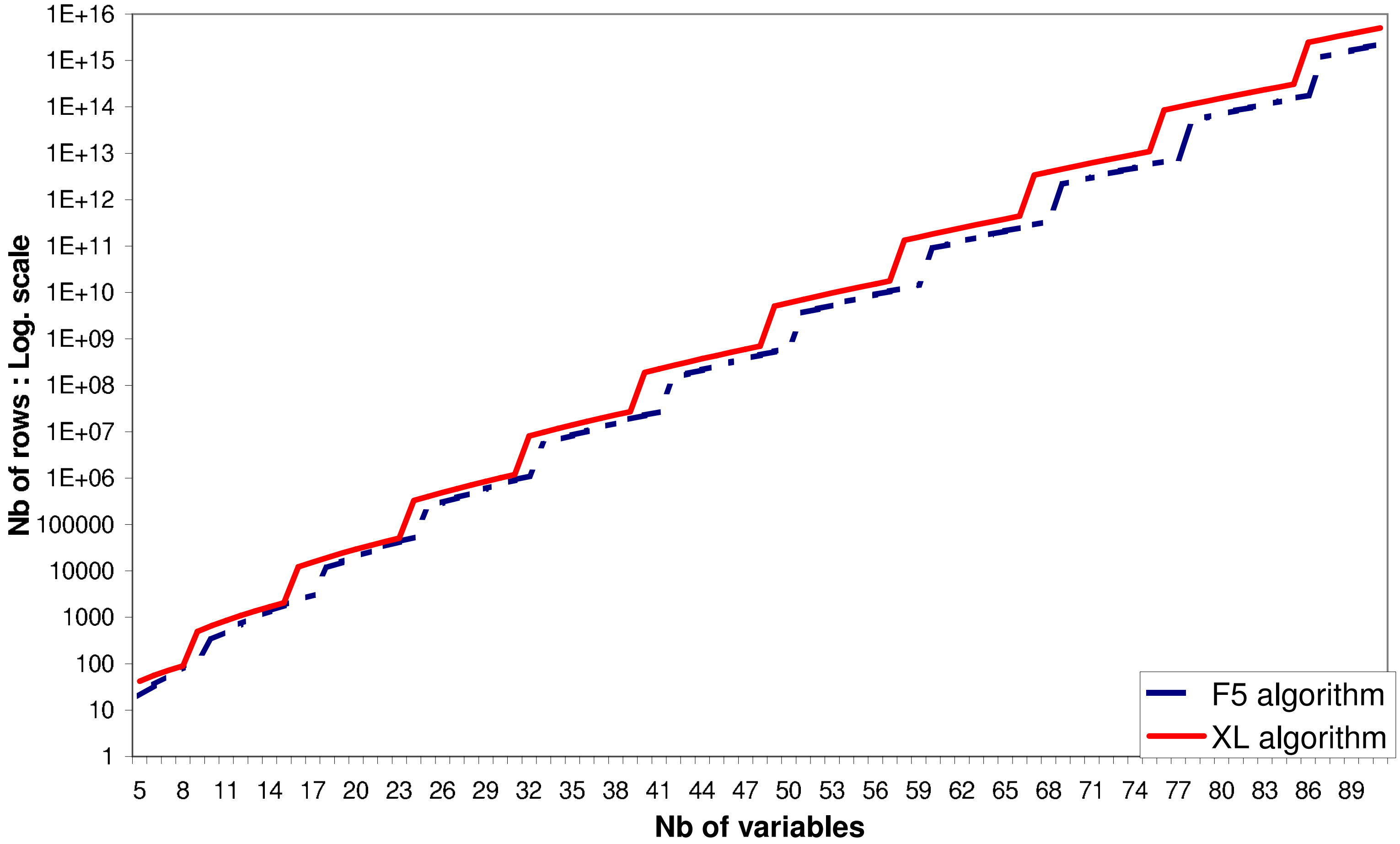
Semi-regular sequences: $m = n + 2$

Complexity : $(\text{size of the matrix})^{\omega}$

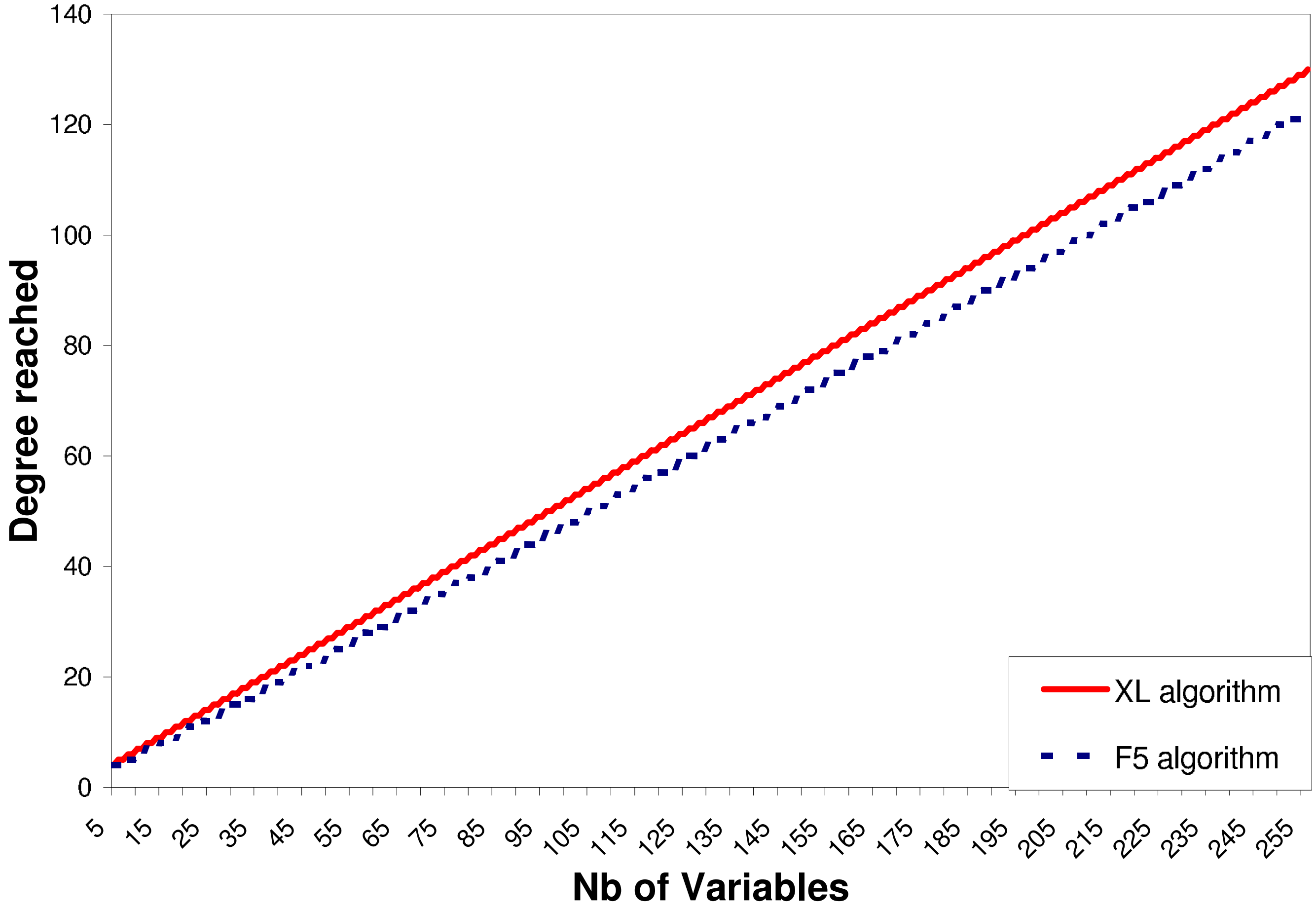
XL algorithm		F_4 algorithm		F_5 algorithm
Matrix size	$\geq$	Matrix size	$\geq$	Matrix size
	↑		↑	Full rank matrix
	Buchberger criteria		F_5 criteria	

Semi-regular sequences: $m = n + 2$

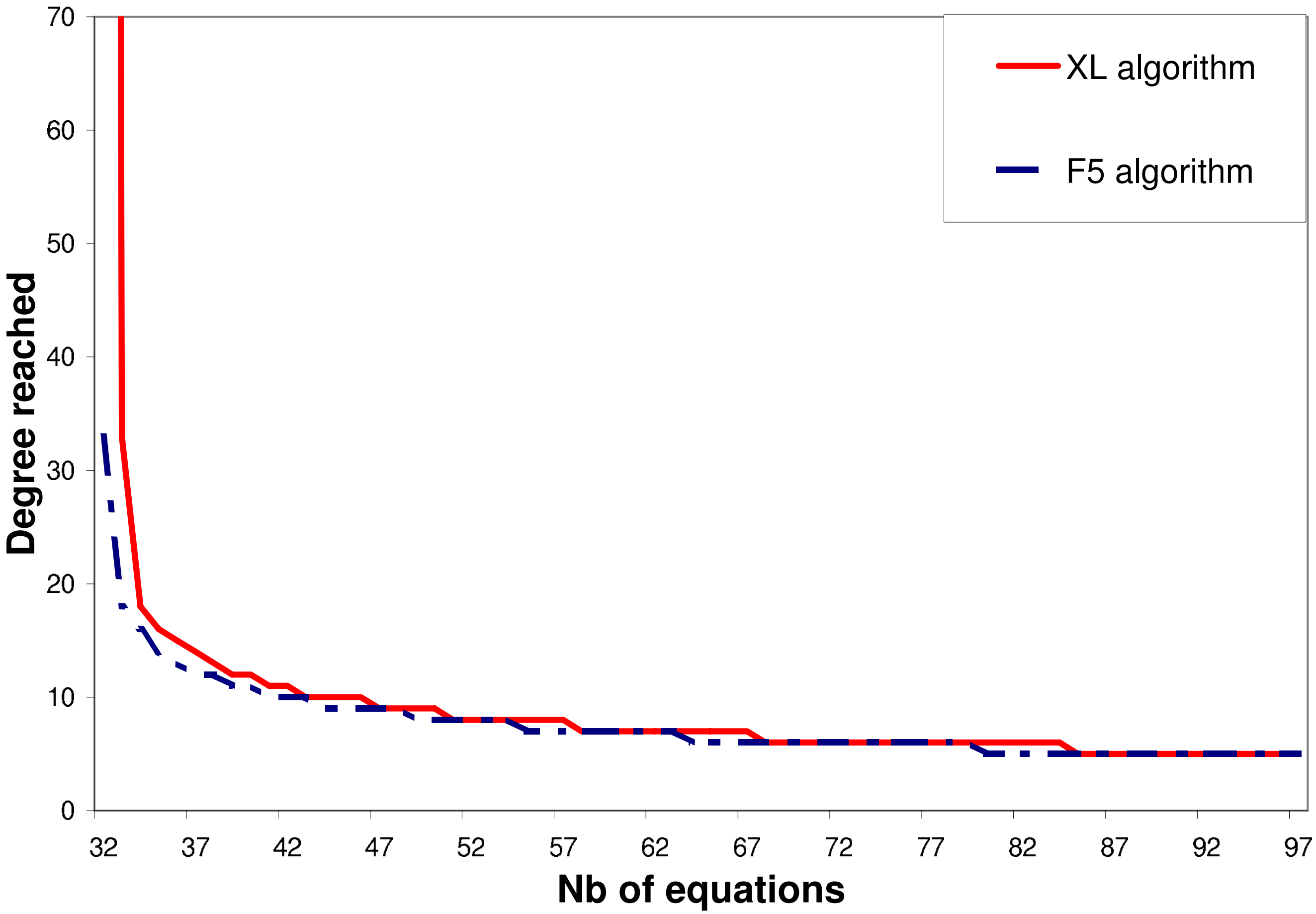
Complexity : $(\text{size of the matrix})^{\boldsymbol{w}}$



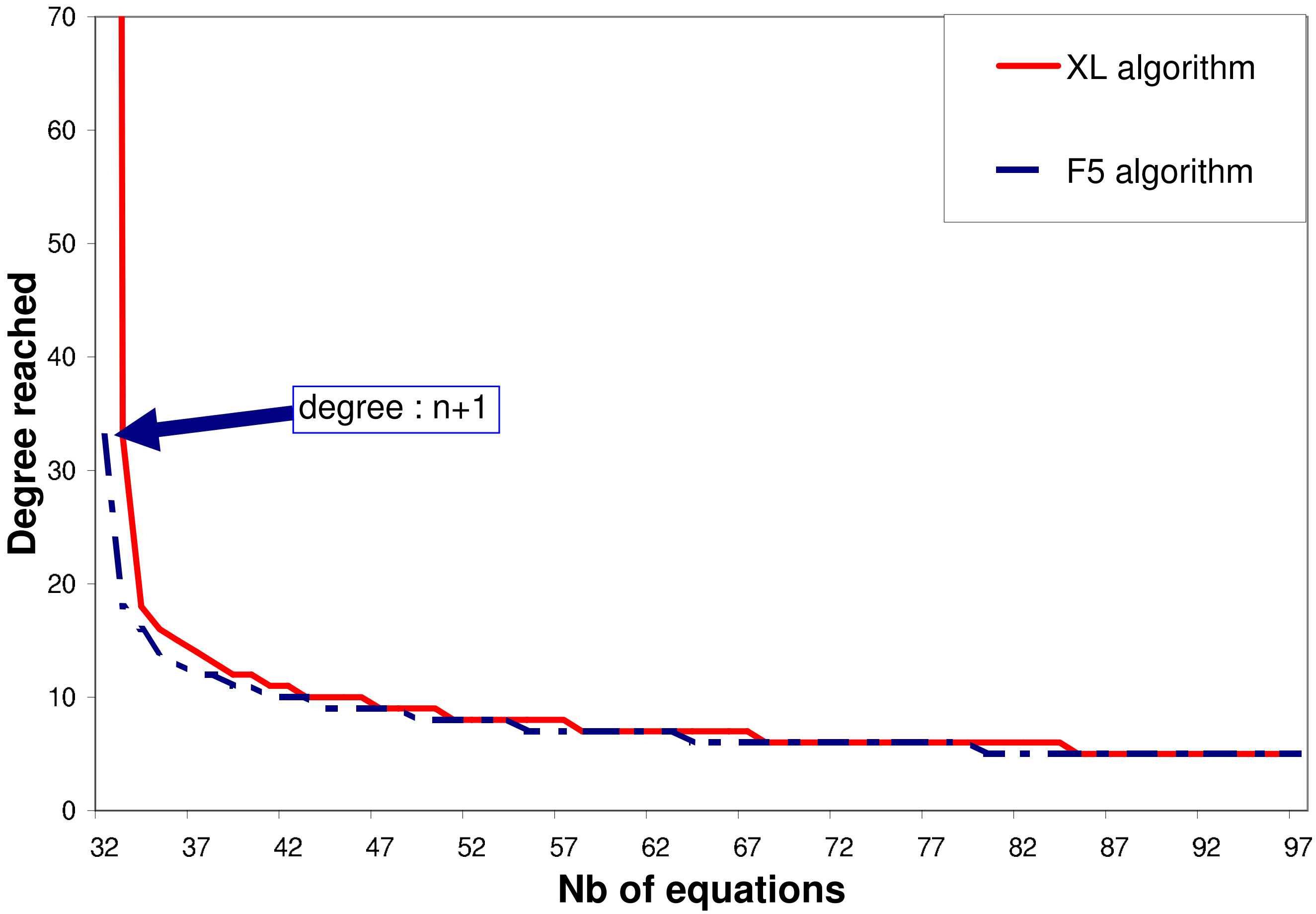
Semi-regular sequences: $m = n + 2$



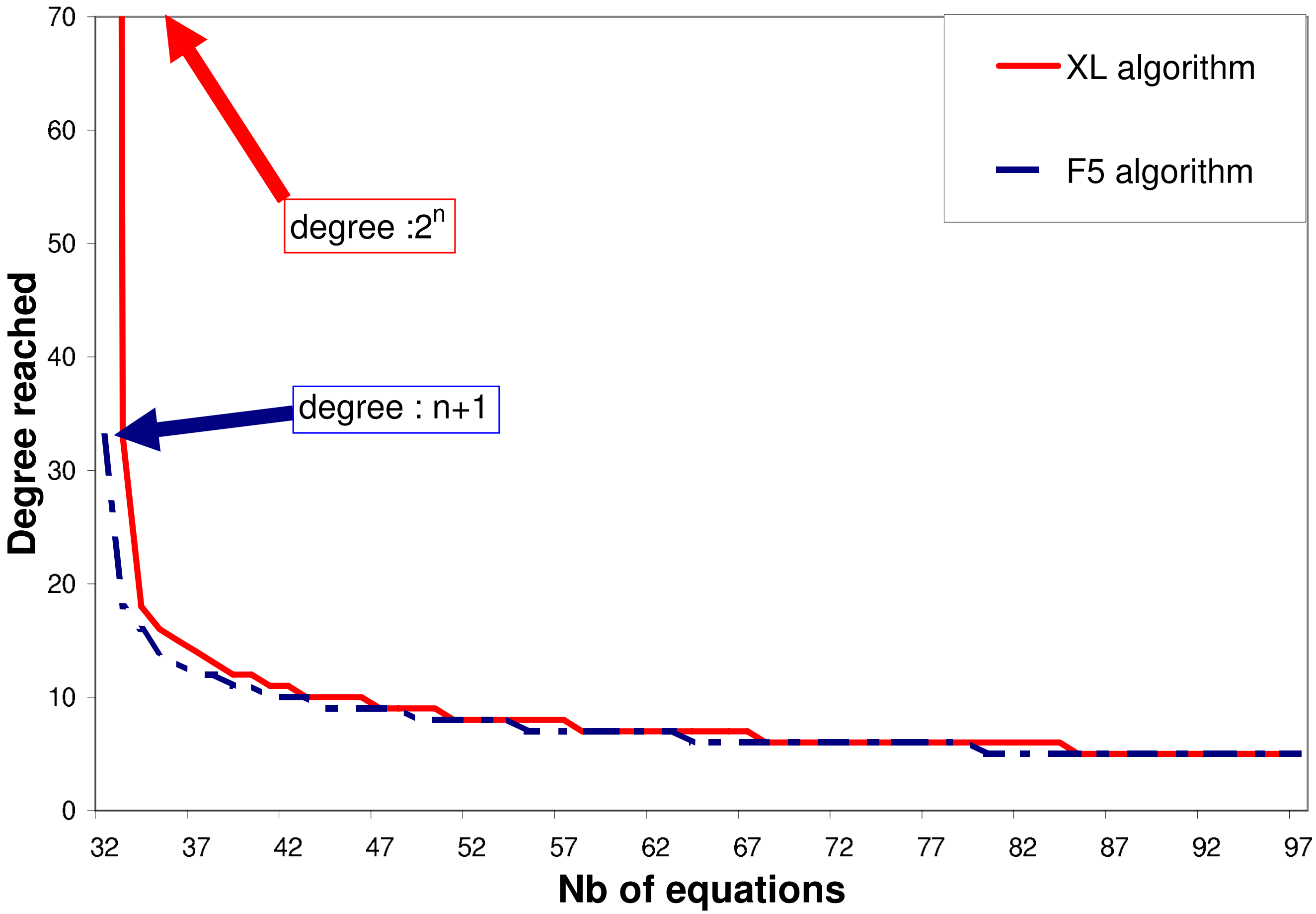
Semi-regular sequences: $n = 32$



Semi-regular sequences: $n = 32$



Semi-regular sequences: $n = 32$



For random system of n quadratic equations on n variables, univariate polynomial will have a degree 2^n.

On HFE systems

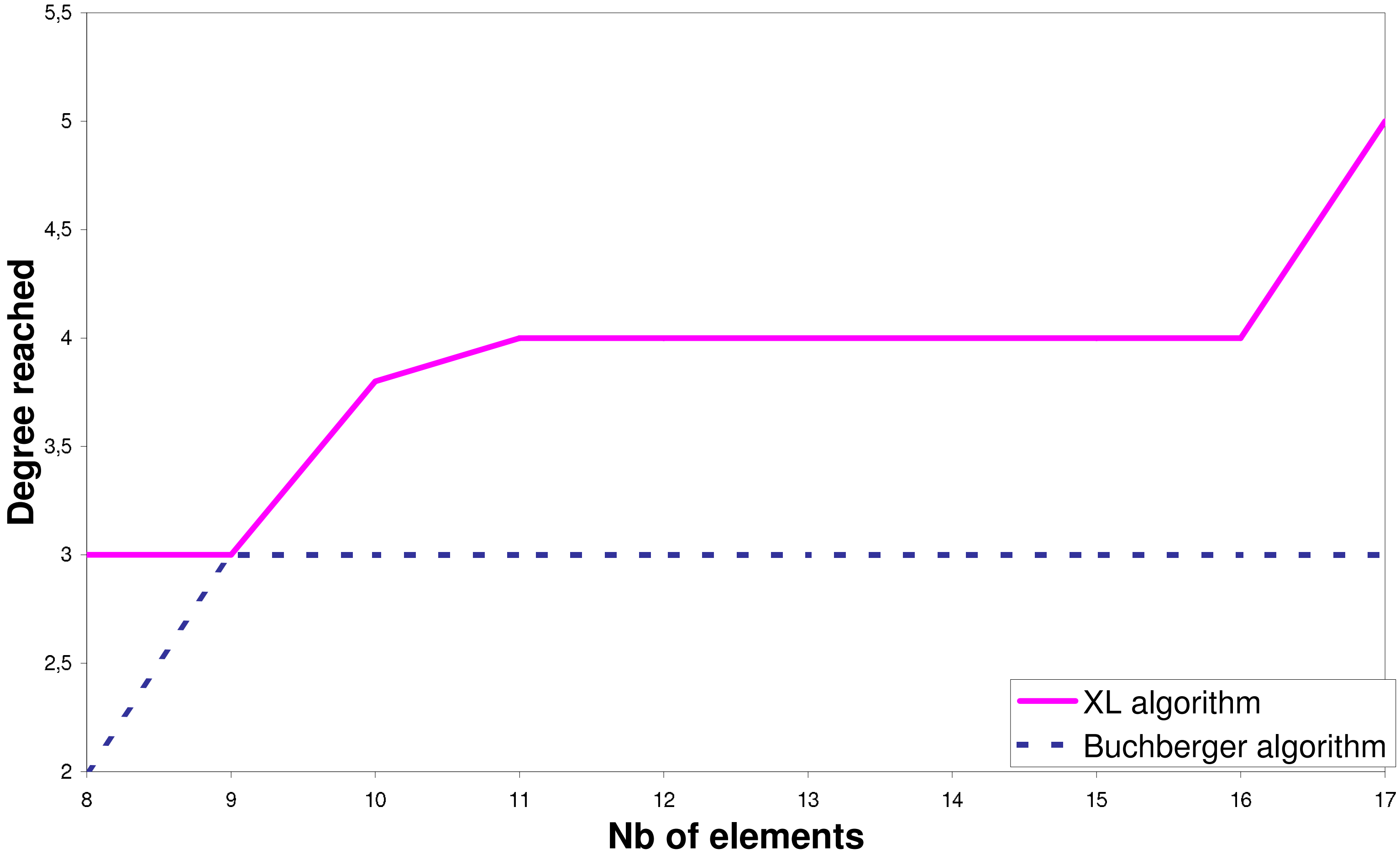
For real system :

Lower degree reached

Example : Public Key Cryptosystem HFE proposed by J. Patarin composed by a system of n quadratic equations with n variables.

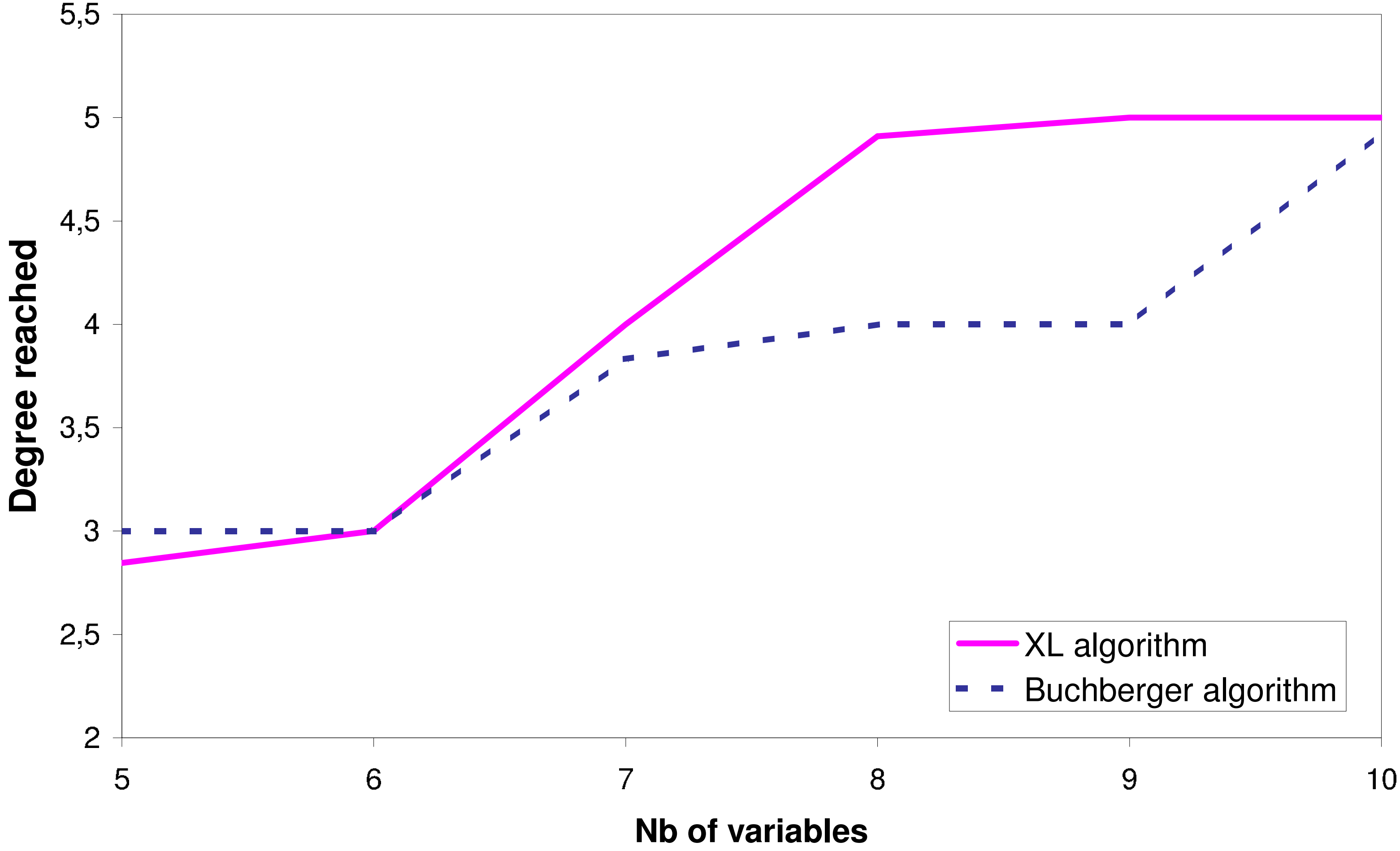
On HFE systems

HFE : n variables and degree 24 for univariate polynomial.



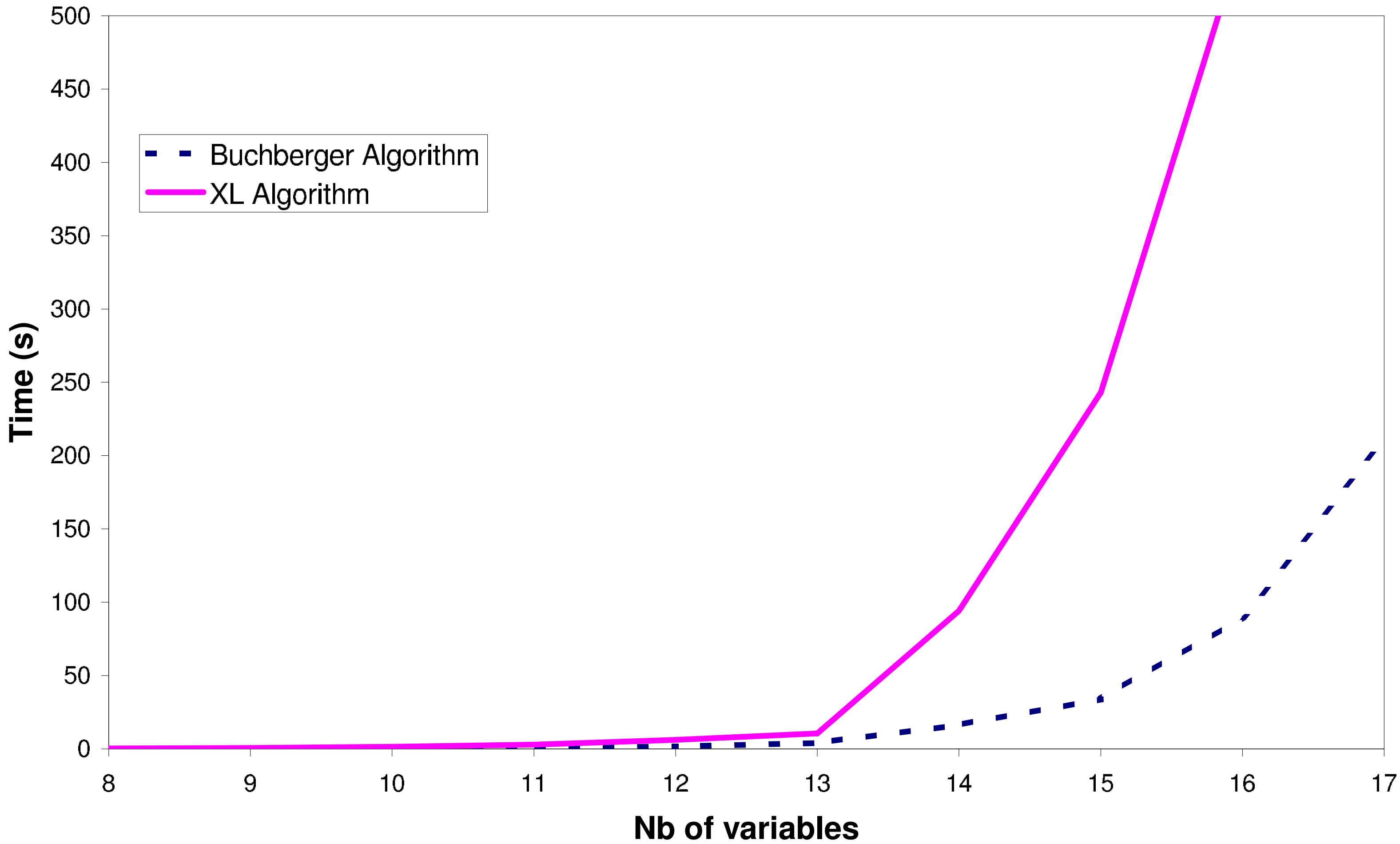
On HFE systems

HFE : n variables and degree 24 for univariate polynomial.



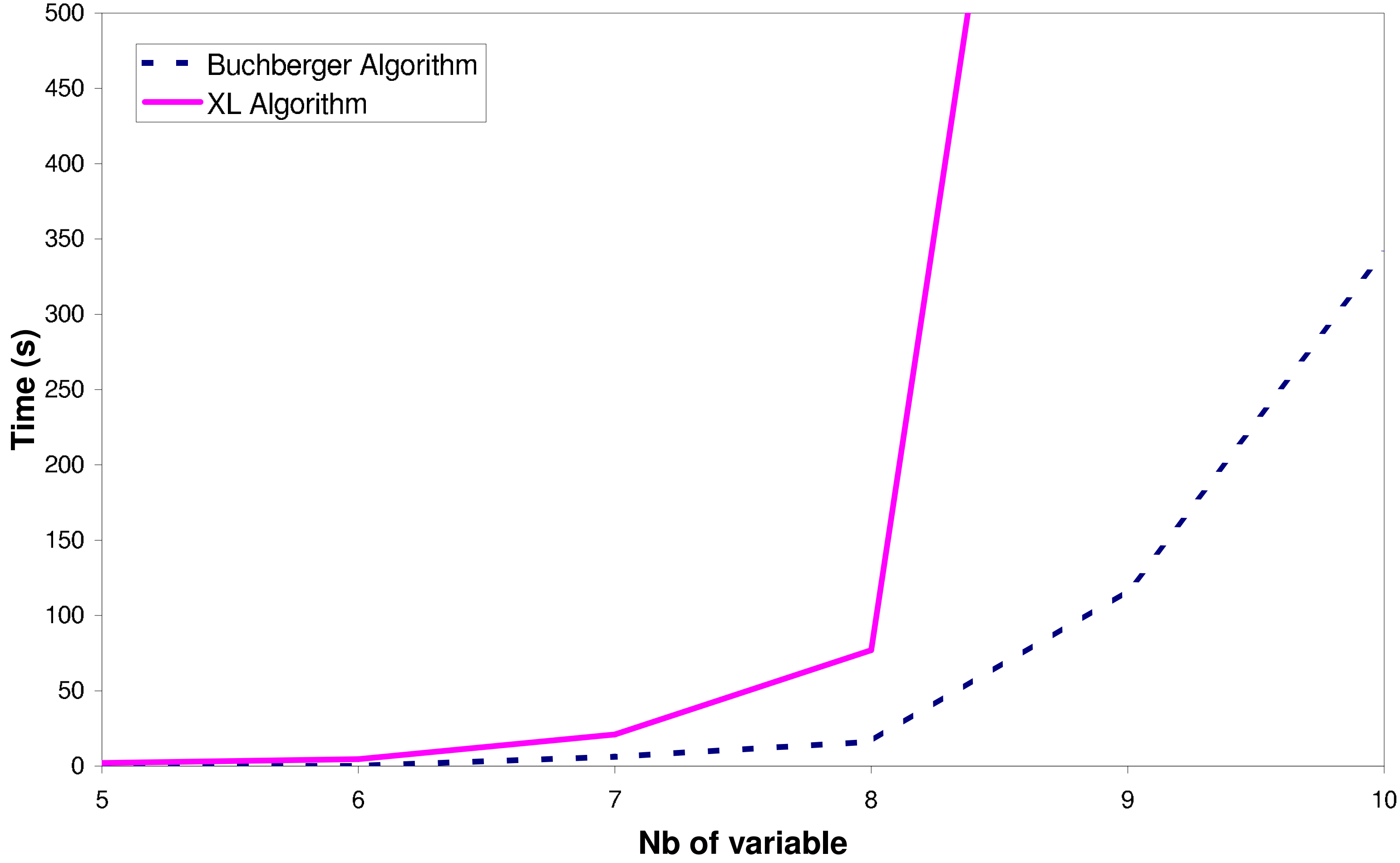
HFE: Time computation on F_2

HFE : n variables and degree 24 for univariate polynomial with Magma 2.10.



HFE: Time computation on F_{16}

HFE : n variables and degree 24 for univariate polynomial with Magma 2.10.



Conclusion

- We compared XL with known Gröbner basis algorithms.

Conclusion

- We compared XL with known Gröbner basis algorithms.
- We showed that to solve a system of algebraic equations treated in XL is equivalent to calculate the reduced Gröbner basis of the ideal associated with the system.

Conclusion

- We compared XL with known Gröbner basis algorithms.
- We showed that to solve a system of algebraic equations treated in XL is equivalent to calculate the reduced Gröbner basis of the ideal associated with the system.
- We showed XL is a redundand version of the $\boldsymbol{F_4}$ algorithm.

Conclusion

- We compared XL with known Gröbner basis algorithms.
- We showed that to solve a system of algebraic equations treated in XL is equivalent to calculate the reduced Gröbner basis of the ideal associated with the system.
- We showed XL is a redundand version of the F_4 algorithm.
- We showed the result of simulations comparing XL with F_5, which is an improved version of F_4.

Conclusion

- We compared XL with known Gröbner basis algorithms.
- We showed that to solve a system of algebraic equations treated in XL is equivalent to calculate the reduced Gröbner basis of the ideal associated with the system.
- We showed XL is a redundand version of the F_4 algorithm.
- We showed the result of simulations comparing XL with F_5, which is an improved version of F_4.
- Our results imply that XL is not so efficient as it was expected.

Conclusion

XL algorithm Matrix size	$\geq$	F_4 algorithm Matrix size	$\geq$	F_5 algorithm Matrix size

Conclusion

- XL algorithm Matrix size $\geq$ F_4 algorithm Matrix size $\geq$ F_5 algorithm Matrix size
- XL algorithm Time Experiments $\geq$ F_4 algorithm Time Experiments $\geq$ F_5 algorithm Time Experiments

Homogeneous semi-regular sequence

Definition. Homogeneous semi-regular sequence :

Let $f_1, \ldots, f_m$ be a sequence of m homogeneous polynomials (i.e. for all monomial t of f_i, $\deg(t) = \deg(f_i)$ in $\mathcal{R}_n^h := \mathbb{F}_2[x_1, \ldots, x_n]/\langle x_1^2, \ldots, x_n^2\rangle$ or $\mathbb{Q}[x_1, \ldots, x_n]$), and $\mathcal{I} = \langle f_1, \ldots, f_m\rangle$ an ideal of $\mathcal{R}_n^h$ or $\mathbb{Q}[x_1, \ldots, x_n]$.

- The *degree of regularity* of $\mathcal{I}$ is the minimal degree d such that $\{LT(f) \mid f \in \mathcal{I}, \deg(f) = d\}$ is exactly the set of monomials of degree d in $\mathcal{R}_n^h$, denoted by $D_{reg}(\mathcal{I})$.
- $f_1, \ldots, f_m$ is a *homogeneous semi regular sequence on* $\mathbb{F}_2$ if $\mathcal{I} \neq \mathcal{R}_n^h$ and for $i \in \{1, \ldots, m\}$, if $g_i f_i = 0$ in $\mathcal{R}_n^h/\langle f_1, \ldots, f_{i-1}\rangle$ and $\deg(g_i f_i) < D_{reg}(\mathcal{I})$ then $g_i = 0$ in $\mathcal{R}_n^h/\langle f_1, \ldots, f_{i-1}, f_i\rangle$.
- $f_1, \ldots, f_m$ is a *homogeneous semi regular sequence on* $\mathbb{Q}$ if $\mathcal{I} \neq \mathbb{Q}[x_1, \ldots, x_n]$ and for $i \in \{1, \ldots, m\}$, if $g_i f_i = 0$ in $\mathbb{Q}[x_1, \ldots, x_n]/\langle f_1, \ldots, f_{i-1}\rangle$ and $\deg(g_i f_i) < D_{reg}(\mathcal{I})$ then $g_i = 0$ in $\mathbb{Q}[x_1, \ldots, x_n]/\langle f_1, \ldots, f_{i-1}\rangle$.

Affine semi-regular sequence

Affine semi-regular sequence : Let $f_1, \dots, f_m$ be a sequence of m polynomials, and $\mathcal{I} = \langle f_1, \dots, f_m \rangle$ an ideal of $\mathbb{F}_2[x_1, \dots, x_n]/\langle x_1^2 - x_1, \dots, x_n^2 - x_n \rangle$ or $\mathbb{Q}[x_1, \dots, x_n]$. Let f_i^h the homogeneous part of the largest degree of f_i.

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