



The complexity of FXL

Lars Ran & Monika Trimoska

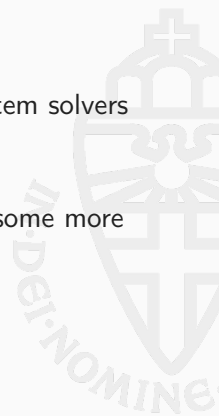
July 1, 2025, Summer School on real-world crypto and privacy



The XL and FXL algorithms are among the state-of-the-art polynomial system solvers

They currently hold some record computations

For random MQ systems, their complexity can be analyzed, in contrast to some more involved algorithms



A recap of XL



We consider polynomials

$$f_1, \dots, f_m \in \mathcal{R} = \mathbb{F}_q[x_1, \dots, x_n]$$

and assume the corresponding systems have a single solution, (a_1, \dots, a_n) , so that

$$\langle f_1, \dots, f_m \rangle = \langle x_1 - a_1, \dots, x_n - a_n \rangle$$



We consider polynomials

$$f_1, \dots, f_m \in \mathcal{R} = \mathbb{F}_q[x_1, \dots, x_n]$$

and assume the corresponding systems have a single solution, (a_1, \dots, a_n) , so that

$$\langle f_1, \dots, f_m \rangle = \langle x_1 - a_1, \dots, x_n - a_n \rangle$$

For example:

$$\begin{aligned} & \langle 8z^2 + 7x + 3y + 7z + 7, \\ & \quad 6yz + 5z^2 + 7x + 7, \\ & \quad 8xz + 6yz + 7y + 8z + 6, \\ & \quad 4x^2 + 5xz + 6yz + 7x + 2 \rangle \\ & \qquad \qquad \qquad = \langle x - 3, y - 1, z - 4 \rangle \end{aligned}$$

How to get the solution?

We "just" need to find $g_{ij} \in \mathcal{R} (= \mathbb{F}_q[x_1, \dots, x_n])$ such that

$$g_{11} \cdot f_1 + \dots + g_{1m} \cdot f_m = x_1 - a_1$$

$$\vdots$$

$$g_{n1} \cdot f_1 + \dots + g_{nm} \cdot f_m = x_n - a_n$$



How to get the solution?

We "just" need to find $g_{ij} \in \mathcal{R} (= \mathbb{F}_q[x_1, \dots, x_n])$ such that

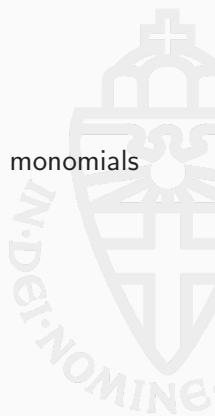
$$g_{11} \cdot f_1 + \dots + g_{1m} \cdot f_m = x_1 - a_1$$

$$\vdots$$

$$g_{n1} \cdot f_1 + \dots + g_{nm} \cdot f_m = x_n - a_n$$

In practice, there are many such g_{ij} . However, we want to limit the degree of $g_{ij} \cdot f_j$ so that there is still a solution

- (1) Determine the degree d_{solv} for which such g_{ij} should exist
- (2) Consider the Macaulay matrix which has as rows all f_i multiplied by all monomials g up to a certain degree
- (3) Find a right kernel element of this matrix



Analyzing the operating degree



First, let us define R_d to be the monomials of \mathcal{R} of degree d

$$R_d = \left\{ \prod_{j=1}^d x_{i_j} \mid 1 \leq i_1, \dots, i_d \leq n \right\} \quad \text{for } d \geq 0$$



First, let us define R_d to be the monomials of \mathcal{R} of degree d

$$R_d = \left\{ \prod_{j=1}^d x_{i_j} \mid 1 \leq i_1, \dots, i_d \leq n \right\} \quad \text{for } d \geq 0$$

For example:

$$R_0 = \{1\}$$

$$R_1 = \{x_1, \dots, x_n\}$$

$$R_2 = \{x_1^2, x_1x_2, \dots, x_{n-1}x_n, x_n^2\}$$



First, let us define R_d to be the monomials of \mathcal{R} of degree d

$$R_d = \left\{ \prod_{j=1}^d x_{i_j} \mid 1 \leq i_1, \dots, i_d \leq n \right\} \quad \text{for } d \geq 0$$

For example:

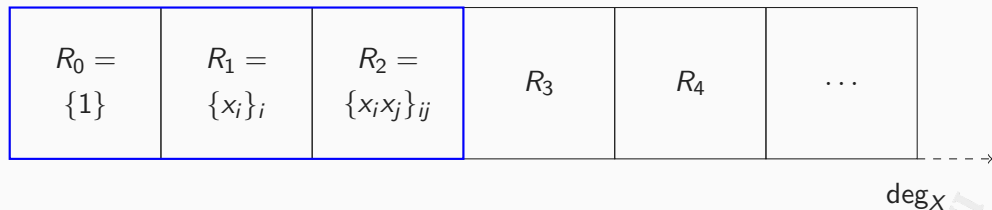
$$R_0 = \{1\}$$

$$R_1 = \{x_1, \dots, x_n\}$$

$$R_2 = \{x_1^2, x_1x_2, \dots, x_{n-1}x_n, x_n^2\}$$

Furthermore we define: $R_{\leq d} = R_d \cup \dots \cup R_0$.





A quadratic equation is then a linear combination of the union of the blue boxes $R_{\leq 2}$.

Then, let us define the rows of the Macaulay matrix

$$I_{\leq d} = \{uf_i \mid u \in R, 1 \leq i \leq m, \deg(uf_i) \leq d\}$$



Then, let us define the rows of the Macaulay matrix

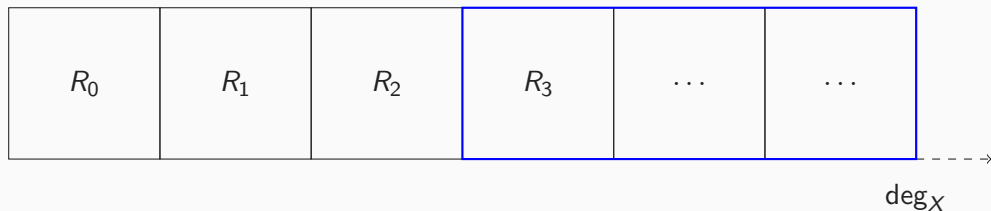
$$I_{\leq d} = \{uf_i \mid u \in R, 1 \leq i \leq m, \deg(uf_i) \leq d\}$$

Now we can describe these using $R_{\leq d}$

$$I_{\leq d} = f_1 \cdot R_{\leq d - \deg(f_1)} + \dots + f_m \cdot R_{\leq d - \deg(f_m)}$$

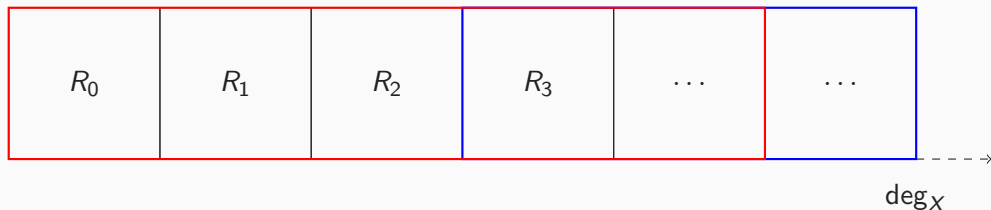


Pictorially again



For a quadratic polynomial f , the $f \cdot R_3$ polynomials are linear combinations of the union of the blue boxes.

Pictorially again



For a quadratic polynomial f , the $f \cdot R_3$ polynomials are linear combinations of the union of the **blue** boxes.

For all systems, the $I_{\leq 4}$ polynomials are linear combinations of the union of the **red** boxes.

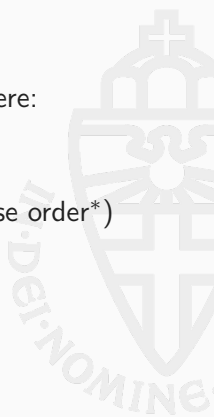
What about d_{solv} ?

d_{solv} is exactly the degree for which the Macaulay matrix has a right kernel of dimension 1



The degree d Macaulay matrix $\mathcal{M}(f_1, \dots, f_m)$ of a system is the matrix where:

- ▶ Its rows are labeled by the products $u \cdot f_i$ (the vectors that span $I_{\leq d}$)
- ▶ Its columns are labeled by monomials r in $R_{\leq d}$ (sorted in graded reverse order*)
- ▶ Its coefficients are the coefficient of r in $u \cdot f_i$



Macaulay matrices example $d = 3$

$$\begin{array}{l} \\ 1 \cdot f_1 \\ 1 \cdot f_2 \\ \vdots \\ x_1 \cdot f_1 \\ x_1 \cdot f_2 \\ \vdots \\ x_n \cdot f_m \end{array} \begin{pmatrix} x_1^3 & x_1^2 x_2 & \dots & x_n & 1 \\ 0 & 0 & \dots & 5 & 9 \\ 0 & 0 & \dots & 7 & 2 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 3 & 1 & \dots & 0 & 0 \\ 4 & 8 & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \dots & 6 & 0 \end{pmatrix}$$



Can we force this?

If there are enough linear independent rows in the matrix, we surely obtain a unique solution



Can we force this?

If there are enough linear independent rows in the matrix, we surely obtain a unique solution

The amount of columns is given by:

$$\dim(R_{\leq d}) = \binom{n+d}{d}$$

The amount of rows is given (for a quadratic system) by:

$$m \cdot \binom{n+d-2}{d-2}$$



Can we force this?

If there are enough linear independent rows in the matrix, we surely obtain a unique solution

The amount of columns is given by:

$$\dim(R_{\leq d}) = \binom{n+d}{d}$$

The amount of rows is given (for a quadratic system) by:

$$m \cdot \binom{n+d-2}{d-2}$$

So we can easily determine d_{solV} right? Not yet...



The rows that we generate might contain linear dependencies, called syzygies

In fact, the following syzygies always appear:

$$f_i f_j - f_j f_i = 0$$



The rows that we generate might contain linear dependencies, called syzygies

In fact, the following syzygies always appear:

$$f_i f_j - f_j f_i = 0$$

Not only that, but even syzygies between syzygies can appear!



The rows that we generate might contain linear dependencies, called syzygies

In fact, the following syzygies always appear:

$$f_i f_j - f_j f_i = 0$$

Not only that, but even syzygies between syzygies can appear!

When we account for these syzygies, and no other syzygies appear up to d_{solv} we can predict d_{solv}

Note that at that point syzygies must appear!

When we correctly account for these syzygies and no other syzygies appear, we obtain the following number of linear independent rows

$$\sum_{i=1}^n (-1)^{i+1} \binom{m}{i} \binom{n+d-2i}{d-2i}$$



Counting the linear independent equations

When we correctly account for these syzygies and no other syzygies appear, we obtain the following number of linear independent rows

$$\sum_{i=1}^n (-1)^{i+1} \binom{m}{i} \binom{n+d-2i}{d-2i}$$

Then d_{solv} is exactly the lowest d for which the alternating sum above is greater than the number of columns $\binom{n+d}{d}$, i.e.

$$\sum_{i=0}^n (-1)^{i+1} \binom{m}{i} \binom{n+d-2i}{d-2i} \leq 0$$

Lets plug in some numbers!

Let us consider a random quadratic system with 11 variables and 20 equations

We consider the numbers $\dim(R_{\leq d}) - \dim(I_{\leq d})$



Lets plug in some numbers!

Let us consider a random quadratic system with 11 variables and 20 equations

We consider the numbers $\dim(R_{\leq d}) - \dim(I_{\leq d})$

| | | | | | |
|---|----|----|-----|----|------|
| 1 | 12 | 58 | 124 | -5 | -623 |
|---|----|----|-----|----|------|

deg_x →

Lets plug in some numbers!

Let us consider a random quadratic system with 11 variables and 20 equations

We consider the numbers $\dim(R_{\leq d})$

| | | | | | |
|---|----|----|-----|------|------|
| 1 | 12 | 78 | 364 | 1365 | 4368 |
|---|----|----|-----|------|------|

$\deg_X \rightarrow$

The complexity



The cost of reducing a matrix

Recall, that we have a matrix of size $m \cdot \binom{n+d-2}{d-2} \times \binom{n+d}{d}$ that we want to reduce.
This has a complexity of

$$\mathcal{O} \left(\binom{n+d-2}{d-2} \binom{n+d}{d}^2 \right) \text{ field operations}$$



The cost of reducing a matrix

Recall, that we have a matrix of size $m \cdot \binom{n+d-2}{d-2} \times \binom{n+d}{d}$ that we want to reduce. This has a complexity of

$$\mathcal{O} \left(\binom{n+d-2}{d-2} \binom{n+d}{d}^2 \right) \text{ field operations}$$

It turns out that we can randomly remove rows to get a square matrix of the same rank, for a cost of

$$C_{XL}(q, n, m) = \mathcal{O} \left(\binom{n+d}{d}^\omega \right) = \mathcal{O} \left(n^{\omega d} \right) \text{ field operations}$$

The cost of reducing a matrix

Recall, that we have a matrix of size $m \cdot \binom{n+d-2}{d-2} \times \binom{n+d}{d}$ that we want to reduce. This has a complexity of

$$\mathcal{O} \left(\binom{n+d-2}{d-2} \binom{n+d}{d}^2 \right) \text{ field operations}$$

It turns out that we can randomly remove rows to get a square matrix of the same rank, for a cost of

$$C_{XL}(q, n, m) = \mathcal{O} \left(\binom{n+d}{d}^\omega \right) = \mathcal{O} \left(n^{\omega d} \right) \text{ field operations}$$

The parameter d is of a large influence in determining the complexity!

Because d has such a large influence, it would be worthwhile to find ways to lower it.

One such way is by guessing variables, essentially reducing n .

This has a cost, but might result in a good tradeoff.

$$C_{FXL}(q, n, m) = \min_{0 \leq k \leq n} q^k C_{XL}(q, n - k, m)$$



The matrix that we construct is generally really sparse. I.e. it has a lot of zero entries. We can use algorithms optimized for such systems such as Wiedemann. They have cost

$$3 \cdot \rho \cdot N^2 \text{ field operations}$$



The matrix that we construct is generally really sparse. I.e. it has a lot of zero entries. We can use algorithms optimized for such systems such as Wiedemann. They have cost

$$3 \cdot \rho \cdot N^2 \text{ field operations}$$

Here ρ is the density of the matrix, which is $\binom{n+2}{2}$ for MQ, and N the size of the square matrix



The matrix that we construct is generally really sparse. I.e. it has a lot of zero entries. We can use algorithms optimized for such systems such as Wiedemann. They have cost

$$3 \cdot \rho \cdot N^2 \text{ field operations}$$

Here ρ is the density of the matrix, which is $\binom{n+2}{2}$ for MQ, and N the size of the square matrix

$$C_{FXL}(q, n, m) = \min_{0 \leq k \leq n} q^k \cdot 3 \cdot \binom{n+2}{2} \cdot \binom{(n-k) + d_k}{d_k}^2$$

Hilbert series and semi-regularity



In our computation of the number of linear independent rows, we assumed that there are no additional syzygies.

This assumption is called the semi-regularity assumption of random polynomial systems. For random systems, this is believed to be true (but not yet proven!¹)

For structured systems, this is often very much not the case. If we cannot correctly predict the solving degree, we can not correctly predict the complexity!

¹This is Fröberg's conjecture

We described the rank of the Macaulay matrix using an alternating sum constraint.

In the literature, this is often done with the Hilbert series

$$\frac{(1-t^2)^m}{(1-t)^{n+1}} = \sum_{d \geq 0} \sum_i (-1)^i \binom{m}{i} \binom{n+d-2i}{d-2i} \cdot t^d$$



We described the rank of the Macaulay matrix using an alternating sum constraint.

In the literature, this is often done with the Hilbert series

$$\frac{(1 - t^2)^m}{(1 - t)^{n+1}} = \sum_{d \geq 0} \sum_i (-1)^i \binom{m}{i} \binom{n + d - 2i}{d - 2i} \cdot t^d$$

Semi-regularity now claims that random polynomial systems exactly follows the Hilbert series

What did we do?

We explored the theory of Macaulay matrices for XL

We computed the solving degree for semi-regular polynomial systems

We computed the complexity for XL and FXL



Thanks for listening!

