Complexity of Gröbner bases computations and applications to cryptography

Elisa Gorla

Institut de mathématiques, Université de Neuchâtel

Journées nationales de calcul formel
March 2, 2021

Nonlinear algebra in the applications

Polynomial system solving is ubiquitous, as many models in the sciences and engineering can be described by non-linear polynomials. This includes:

- algebraic statistics,
- algebraic biology,
- chemical reaction networks,
- coding theory,
- computer vision,
- cryptography,
- networks modelling,
- neuroscience,
- robotics,
- string theory,
- topological data analysis via (multivariate) persistent homology.

Communication over a channel

$$\boxed{\text{source}} \longrightarrow \boxed{\text{encryptor}} \longrightarrow \boxed{\text{encoder}} \longrightarrow$$
$$\rightsquigarrow \text{ noise } \rightsquigarrow \text{ transmission } \rightsquigarrow \text{ noise } \rightsquigarrow$$
$$\longrightarrow \boxed{\text{decoder}} \longrightarrow \boxed{\text{decryptor}} \longrightarrow \boxed{\text{sink}}$$

Coding theory aims at correcting errors occurring during data transmission across a noisy channel. Cryptography aims at ensuring confidentiality over an insecure channel.

Communication over a channel

$$\boxed{\text{source}} \longrightarrow \boxed{\text{encryptor}} \longrightarrow \boxed{\text{encoder}} \longrightarrow$$
$$\rightsquigarrow \text{ noise } \rightsquigarrow \text{ transmission } \rightsquigarrow \text{ noise } \rightsquigarrow$$
$$\longrightarrow \boxed{\text{decoder}} \longrightarrow \boxed{\text{decryptor}} \longrightarrow \boxed{\text{sink}}$$

Coding theory aims at correcting errors occurring during data transmission across a noisy channel. Cryptography aims at ensuring confidentiality over an insecure channel.

Rank-metric codes are used over networks.

Definition

A rank-metric code is a vector subspace $C \subseteq \mathrm{Mat}_{k\times m}(\mathbb{F}_q)$.
The rank distance between $A, B \in \mathrm{Mat}_{k\times m}(\mathbb{F}_q)$ is

$$d(A, B) = \mathrm{rank}\,(A - B).$$

Rank-metric codes

Definition

A rank-metric code is a vector subspace $C \subseteq \mathrm{Mat}_{k\times m}(\mathbb{F}_q)$.
The rank distance between $A, B \in \mathrm{Mat}_{k\times m}(\mathbb{F}_q)$ is

$$d(A, B) = \mathrm{rank}\,(A - B).$$

The minimum distance of C is

$$d_{\min}(C) = \min\{d(A, B) \mid A, B \in C, A \neq B\}.$$

Rank-metric codes

Definition

A rank-metric code is a vector subspace $C \subseteq \mathsf{Mat}_{k\times m}(\mathbb{F}_q)$.
The rank distance between $A, B \in \mathsf{Mat}_{k\times m}(\mathbb{F}_q)$ is

$$d(A,B) = \operatorname{rank}(A-B).$$

The minimum distance of C is

$$d_{\min}(C) = \min\{d(A,B) \mid A, B \in C, A \neq B\}.$$

Example

$C = \left\langle \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix}, \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix} \right\rangle \subseteq \mathsf{Mat}_{3\times 3}(\mathbb{F}_2)$ has $d_{\min}(C) = 2$.

Decoding a rank-metric code

Let C be a rank-metric code. If $M \in C$ is sent and $N = M + E$ is received, then the error E can be corrected if

$$\operatorname{rank}(E) \leq \frac{d_{\min}(C)-1}{2}.$$

Decoding a rank-metric code

Let C be a rank-metric code. If $M \in C$ is sent and $N = M + E$ is received, then the error E can be corrected if

$$\mathrm{rank}\,(E) \leq \frac{d_{\min}(C)-1}{2}.$$

In fact, if that is the case, then

$$d(M, N) = \mathrm{rank}\,(E) \leq \frac{d_{\min}(C)-1}{2}$$

and if $L \in C$, $L \neq M$, then

$$d(L, N) \geq d(L, M) - d(M, N) \geq \frac{d_{\min}(C)+1}{2}.$$

Hence M is the only element of C s.t. $d(M, N) \leq \frac{d_{\min}(C)-1}{2}$.

Decoding a rank-metric code

Let C be a rank-metric code. If $M \in C$ is sent and $N = M + E$ is received, then the error E can be corrected if

$$\mathrm{rank}\,(E) \leq \frac{d_{\min}(C)-1}{2}.$$

In fact, if that is the case, then

$$d(M,N) = \mathrm{rank}\,(E) \leq \frac{d_{\min}(C)-1}{2}$$

and if $L \in C$, $L \neq M$, then

$$d(L,N) \geq d(L,M) - d(M,N) \geq \frac{d_{\min}(C)+1}{2}.$$

Hence M is the only element of C s.t. $d(M,N) \leq \frac{d_{\min}(C)-1}{2}$. Equivalently, M is the unique solution to the

Decoding Problem

Given $N \in \mathrm{Mat}_{k\times m}(\mathbb{F}_q)$, find $M \in C$ which minimizes $d(M,N)$.

The MinRank Problem

Assume that $C = \langle M_1, \ldots, M_n \rangle$, then the Decoding Problem becomes

Decoding Problem

Given $N \in \mathrm{Mat}_{\mathrm{k}\times\mathrm{m}}(\mathbb{F}_{\mathrm{q}})$, find $x_1, \ldots, x_n \in \mathbb{F}_q$ which minimize

$$\mathrm{rank}\ (N - \sum_{i=1}^{n} x_i M_i).$$

The MinRank Problem

Assume that $C = \langle M_1, \ldots, M_n \rangle$, then the Decoding Problem becomes

Decoding Problem

Given $N \in \mathrm{Mat}_{k \times m}(\mathbb{F}_q)$, find $x_1, \ldots, x_n \in \mathbb{F}_q$ which minimize

$$\mathrm{rank}\ (N - \textstyle\sum_{i=1}^{n} x_i M_i).$$

which, under our assumptions, is equivalent to the

MinRank Problem

Given $M_1, \ldots, M_n, N \in \mathrm{Mat}_{k \times m}(\mathbb{F}_q)$, find $x_1, \ldots, x_n \in \mathbb{F}_q$ s.t.

$$\mathrm{rank}\,(N - \textstyle\sum_{i=1}^{n} x_i M_i) \leq \frac{d_{\min}(C)-1}{2}.$$

The latter corresponds to a system of polynomial eqn's in $\mathbb{F}_q[x_1, \ldots, x_n]$.

Matrix completion

Matrix completion

Fill in the missing entries of a partially observed matrix in such a way that the matrix has least possible rank

Matrix completion

Matrix completion

Fill in the missing entries of a partially observed matrix in such a way that the matrix has least possible rank, or rank at most r.

The MinRank Problem is a generalization of matrix completion, where the unknown entries are linear forms instead of variables.

Matrix completion

Matrix completion

Fill in the missing entries of a partially observed matrix in such a way that the matrix has least possible rank, or rank at most r.

The MinRank Problem is a generalization of matrix completion, where the unknown entries are linear forms instead of variables. Matrix completion and the MinRank Problem arise in coding theory, cryptography, collaborative filtering, systems theory, IoT localization, and many others.

Matrix completion and the Netflix Problem

Matrix completion

Fill in the missing entries of a partially observed matrix in such a way that the matrix has least possible rank, or rank at most r.

The MinRank Problem is a generalization of matrix completion, where the unknown entries are linear forms instead of variables. Matrix completion and the MinRank Problem arise in coding theory, cryptography, collaborative filtering, systems theory, IoT localization, and many others.

The Netflix Problem

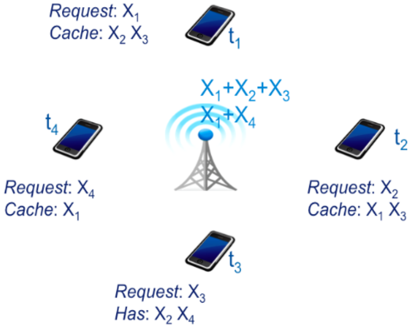
Given a ratings matrix whose entry (i, j) represents the rating of movie j by customer i if customer has watched movie j, and is otherwise missing, fill the remaining entries so that the matrix has low rank.

low rank $\leftrightsquigarrow$ user preferences depend on few factors

Matrix completion and index coding

Index coding

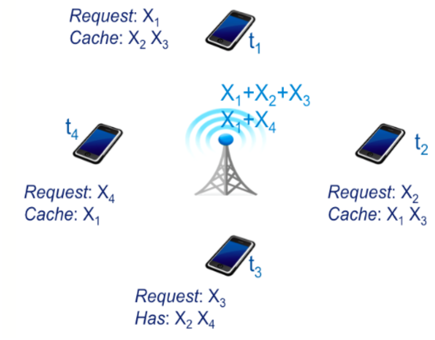
Find an optimal coding scheme for broadcasting multiple messages to receivers with different side information.



Matrix completion and index coding

Index coding

Find an optimal coding scheme for broadcasting multiple messages to receivers with different side information.



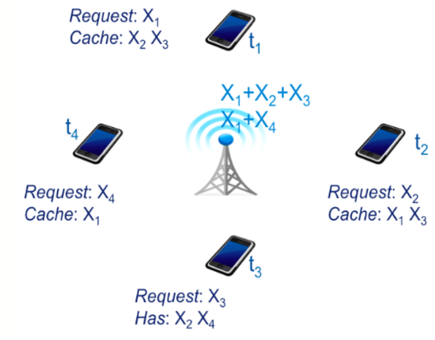
the corresponding incomplete matrix is

$$\begin{pmatrix} 1 & * & * & 0 \\ * & 1 & * & 0 \\ 0 & * & 1 & * \\ * & 0 & 0 & 1 \end{pmatrix}$$

Matrix completion and index coding

Index coding

Find an optimal coding scheme for broadcasting multiple messages to receivers with different side information.



the corresponding incomplete matrix is

$$\begin{pmatrix} 1 & * & * & 0 \\ * & 1 & * & 0 \\ 0 & * & 1 & * \\ * & 0 & 0 & 1 \end{pmatrix}$$

the rank of the completion is the number of messages to be broadcasted

Multivariate cryptography

$\mathbb{F}_q$ finite field, $q_1, \ldots, q_m \in \mathbb{F}_q[x_1, \ldots, x_n]$ usually quadratic

$$\begin{array}{cccc} \mathcal{Q}: & \mathbb{F}_q^n & \longrightarrow & \mathbb{F}_q^m \\ & \alpha = (\alpha_1, \ldots, \alpha_n) & \longmapsto & (q_1(\alpha_1, \ldots, \alpha_n), \ldots, q_m(\alpha_1, \ldots, \alpha_n)) \end{array}$$

$T : \mathbb{F}_q^m \longrightarrow \mathbb{F}_q^m$, $S : \mathbb{F}_q^n \longrightarrow \mathbb{F}_q^n$ random affine linear maps, $\mathcal{P} := T \circ \mathcal{Q} \circ S$

Private key: $\mathcal{Q}, S, T$ Public key: $\mathcal{P} = (f_1, \ldots, f_m)$

Multivariate cryptography

$\mathbb{F}_q$ finite field, $q_1, \ldots, q_m \in \mathbb{F}_q[x_1, \ldots, x_n]$ usually quadratic

$$\begin{array}{rccc} \mathcal{Q}: & \mathbb{F}_q^n & \longrightarrow & \mathbb{F}_q^m \\ & \alpha = (\alpha_1, \ldots, \alpha_n) & \longmapsto & (q_1(\alpha_1, \ldots, \alpha_n), \ldots, q_m(\alpha_1, \ldots, \alpha_n)) \end{array}$$

$T : \mathbb{F}_q^m \longrightarrow \mathbb{F}_q^m$, $S : \mathbb{F}_q^n \longrightarrow \mathbb{F}_q^n$ random affine linear maps, $\mathcal{P} := T \circ \mathcal{Q} \circ S$

Private key: $\mathcal{Q}, S, T$ Public key: $\mathcal{P} = (f_1, \ldots, f_m)$

Multivariate cryptosystem: Alice encrypts $\alpha \in \mathbb{F}_q^n$ to $\beta = \mathcal{P}(\alpha) \in \mathbb{F}_q^m$.
Bob knows $\mathcal{Q}, S, T$, so he can recover $\alpha = \mathcal{P}^{-1}(\beta) = S^{-1} \circ \mathcal{Q}^{-1} \circ T^{-1}(\beta)$.

Trapdoor: Construct $\mathcal{Q}$ so that $\mathcal{Q}^{-1}$ is efficiently computable.

Multivariate cryptography

$\mathbb{F}_q$ finite field, $q_1, \ldots, q_m \in \mathbb{F}_q[x_1, \ldots, x_n]$ usually quadratic

$$\begin{array}{rccc} \mathcal{Q}: & \mathbb{F}_q^n & \longrightarrow & \mathbb{F}_q^m \\ & \alpha = (\alpha_1, \ldots, \alpha_n) & \longmapsto & (q_1(\alpha_1, \ldots, \alpha_n), \ldots, q_m(\alpha_1, \ldots, \alpha_n)) \end{array}$$

$T : \mathbb{F}_q^m \longrightarrow \mathbb{F}_q^m$, $S : \mathbb{F}_q^n \longrightarrow \mathbb{F}_q^n$ random affine linear maps, $\mathcal{P} := T \circ \mathcal{Q} \circ S$

Private key: $\mathcal{Q}, S, T$ Public key: $\mathcal{P} = (f_1, \ldots, f_m)$

Multivariate cryptosystem: Alice encrypts $\alpha \in \mathbb{F}_q^n$ to $\beta = \mathcal{P}(\alpha) \in \mathbb{F}_q^m$.
Bob knows $\mathcal{Q}, S, T$, so he can recover $\alpha = \mathcal{P}^{-1}(\beta) = S^{-1} \circ \mathcal{Q}^{-1} \circ T^{-1}(\beta)$.

Trapdoor: Construct $\mathcal{Q}$ so that $\mathcal{Q}^{-1}$ is efficiently computable.

Multivariate digital signature: In order to sign $\beta \in \mathbb{F}_q^m$, Bob computes $\alpha \in \mathbb{F}_q^n$ s.t. $\mathcal{P}(\alpha) = \beta$.

Multivariate cryptography

$\mathbb{F}_q$ finite field, $q_1, \dots, q_m \in \mathbb{F}_q[x_1, \dots, x_n]$ usually quadratic

$$\begin{array}{rccc} \mathcal{Q}: & \mathbb{F}_q^n & \longrightarrow & \mathbb{F}_q^m \\ & \alpha = (\alpha_1, \dots, \alpha_n) & \longmapsto & (q_1(\alpha_1, \dots, \alpha_n), \dots, q_m(\alpha_1, \dots, \alpha_n)) \end{array}$$

$T : \mathbb{F}_q^m \longrightarrow \mathbb{F}_q^m$, $S : \mathbb{F}_q^n \longrightarrow \mathbb{F}_q^n$ random affine linear maps, $\mathcal{P} := T \circ \mathcal{Q} \circ S$

Private key: $\mathcal{Q}, S, T$ Public key: $\mathcal{P} = (f_1, \dots, f_m)$

Multivariate cryptosystem: Alice encrypts $\alpha \in \mathbb{F}_q^n$ to $\beta = \mathcal{P}(\alpha) \in \mathbb{F}_q^m$.
Bob knows $\mathcal{Q}, S, T$, so he can recover $\alpha = \mathcal{P}^{-1}(\beta) = S^{-1} \circ \mathcal{Q}^{-1} \circ T^{-1}(\beta)$.

Trapdoor: Construct $\mathcal{Q}$ so that $\mathcal{Q}^{-1}$ is efficiently computable.

Multivariate digital signature: In order to sign $\beta \in \mathbb{F}_q^m$, Bob computes $\alpha \in \mathbb{F}_q^n$ s.t. $\mathcal{P}(\alpha) = \beta$.

Security: Eve's task is finding α s.t. $\beta = \mathcal{P}(\alpha)$, knowing $\mathcal{P}$ and β.
She may solve the system $f_1(x_1, \dots, x_n) = \beta_1, \dots, f_m(x_1, \dots, x_n) = \beta_m$.

The Multivariate Quadratic Problem and Gröbner bases

The security of multivariate cryptographic primitives relies on the computational hardness of solving a system of polynomial equations over a finite field.

Multivariate Quadratic (MQ) Problem

Compute the solutions of $f_1 = \ldots = f_m = 0$ over a field, where $\deg(f_i) = 2$.

The Multivariate Quadratic Problem and Gröbner bases

The security of multivariate cryptographic primitives relies on the computational hardness of solving a system of polynomial equations over a finite field.

Multivariate Quadratic (MQ) Problem

Compute the solutions of $f_1 = \ldots = f_m = 0$ over a field, where $\deg(f_i) = 2$.

Assumption

The system has a finite number of solutions over the algebraic closure, possibly zero.

Over $\mathbb{F}_q$, one can find the solutions by exhaustive search. Gröbner bases allow us to find the solutions of a system, under the assumption that they are finitely many. Computing a Gröbner basis has exponential complexity.

Cryptographic security

Systems coming from multivariate cryptographic schemes and digital signatures usually...

... consist of equations of small degree, often 2 or 3,

... are defined over small finite fields and contain the field equations,

... have large m and n, $m \geq n$.

Cryptographic security

Systems coming from multivariate cryptographic schemes and digital signatures usually...

... consist of equations of small degree, often 2 or 3,

... are defined over small finite fields and contain the field equations,

... have large m and n, $m \geq n$.

Systems coming from index calculus on elliptic curves (or on abelian varieties)...

... rarely have a solution,

... have fewer equations in fewer variables of larger degree (e.g. 8 equations of degree 12 in 6 variables), $m \geq n$,

... are defined over large fields, so adding the field equations is not feasible.

Cryptographic security

Systems coming from multivariate cryptographic schemes and digital signatures usually...

... consist of equations of small degree, often 2 or 3,

... are defined over small finite fields and contain the field equations,

... have large m and n, $m \geq n$.

Systems coming from index calculus on elliptic curves (or on abelian varieties)...

... rarely have a solution,

... have fewer equations in fewer variables of larger degree (e.g. 8 equations of degree 12 in 6 variables), $m \geq n$,

... are defined over large fields, so adding the field equations is not feasible.

The complexity of computing a Gröbner basis of a system gives an upper bound on the security of the corresponding cryptographic scheme.

Monomials and term orders

K a field, $R = K[x_1, \ldots, x_n]$

Definition

A monomial is a product of powers of variables $x^a := x_1^{a_1} \cdots x_n^{a_n} \in R$, where $a \in \mathbb{N}^n$.

E.g., $x^{(3,0,1,2)} = x_1^3 x_3 x_4^2 \in K[x_1, x_2, x_3, x_4]$ is a monomial.

Definition

A term order on R is a total order $<$ on the set of monomials such that:

- if $x^a < x^b$, then $x^{a+c} < x^{b+c}$ for any $c \in \mathbb{N}^n$ (multiplicative)
- $1 \leq x^a$ for any $a \in \mathbb{N}^n$ (well-ordering).

Monomials and term orders

K a field, $R = K[x_1, \ldots, x_n]$

Definition

A monomial is a product of powers of variables $x^a := x_1^{a_1} \cdots x_n^{a_n} \in R$, where $a \in \mathbb{N}^n$.

E.g., $x^{(3,0,1,2)} = x_1^3 x_3 x_4^2 \in K[x_1, x_2, x_3, x_4]$ is a monomial.

Definition

A term order on R is a total order $<$ on the set of monomials such that:

- if $x^a < x^b$, then $x^{a+c} < x^{b+c}$ for any $c \in \mathbb{N}^n$ (multiplicative)
- $1 \leq x^a$ for any $a \in \mathbb{N}^n$ (well-ordering).

Example

If $R = K[x]$, then we only have one term order $1 < x < x^2 < \ldots$

Two examples of term orders

Example (Lexicographic order – lex)

$x_1^{a_1} \cdots x_n^{a_n} >_{lex} x_1^{b_1} \cdots x_n^{b_n}$ if the first nonzero coordinate of $(a_1 - b_1, \ldots, a_n - b_n)$ is positive.

E.g., $x_1 x_3 >_{lex} x_2^d$ for any d, $x_1^2 x_2^2 >_{lex} x_1 x_2^2 x_3$, and $x_1 x_2^2 >_{lex} x_1 x_2 x_3$.

Example (Degree Reverse Lexicographic order – drl)

$x_1^{a_1} \cdots x_n^{a_n} >_{drl} x_1^{b_1} \cdots x_n^{b_n}$ if either $\sum_{i=1}^n a_i > \sum_{i=1}^n b_i$ or $\sum_{i=1}^n a_i = \sum_{i=1}^n b_i$ and the last nonzero coordinate of $(a_1 - b_1, \ldots, a_n - b_n)$ is negative.

E.g., $x_1 x_3 >_{drl} x_2$, $x_1 x_2^2 >_{drl} x_1 x_2 x_3$, and $x_1 x_2^2 x_3^2 <_{drl} x_1^2 x_2 x_3^2$.

For the sequel, we fix a term order.

Leading terms and Gröbner bases

$I = (f_1, \ldots, f_m) = \{\sum_{i=1}^m h_i f_i \mid h_i \in R\}$ ideal generated by $f_1, \ldots, f_m \in R$

Definition

The leading term of $f = \sum_{a \in \mathbb{N}^n} \alpha_a x^a \in R$ is $\mathrm{in}(f) = \max\{x^a \mid \alpha_a \neq 0\}$.

E.g., in $R = \mathbb{F}_3[x_1, x_2]$ with the lex term order, $\mathrm{in}(x_2^3 - x_1 x_2^2) = x_1 x_2^2$.

Leading terms and Gröbner bases

$I = (f_1, \ldots, f_m) = \{\sum_{i=1}^m h_i f_i \mid h_i \in R\}$ ideal generated by $f_1, \ldots, f_m \in R$

Definition

The leading term of $f = \sum_{a \in \mathbb{N}^n} \alpha_a x^a \in R$ is $\mathrm{in}(f) = \max\{x^a \mid \alpha_a \neq 0\}$.
The initial ideal of I is $\mathrm{in}(I) = (\mathrm{in}(f) \mid f \in I)$.
The polynomials $g_1, \ldots, g_s \in I$ are a Gröbner basis of I if

$$\mathrm{in}(I) = (\mathrm{in}(g_1), \ldots, \mathrm{in}(g_s)).$$

E.g., in $R = \mathbb{F}_3[x_1, x_2]$ with the lex term order, $\mathrm{in}(x_2^3 - x_1 x_2^2) = x_1 x_2^2$.

Leading terms and Gröbner bases

$I = (f_1, \ldots, f_m) = \{\sum_{i=1}^{m} h_i f_i \mid h_i \in R\}$ ideal generated by $f_1, \ldots, f_m \in R$

Definition

The leading term of $f = \sum_{a \in \mathbb{N}^n} \alpha_a x^a \in R$ is $\mathrm{in}(f) = \max\{x^a \mid \alpha_a \neq 0\}$.
The initial ideal of I is $\mathrm{in}(I) = (\mathrm{in}(f) \mid f \in I)$.
The polynomials $g_1, \ldots, g_s \in I$ are a Gröbner basis of I if

$$\mathrm{in}(I) = (\mathrm{in}(g_1), \ldots, \mathrm{in}(g_s)).$$

E.g., in $R = \mathbb{F}_3[x_1, x_2]$ with the lex term order, $\mathrm{in}(x_2^3 - x_1 x_2^2) = x_1 x_2^2$.
$I = (x_2^3 - x_1 x_2^2, x_1^2 + x_2^2) \ni -x_2^4 = (x_1 + x_2)(x_2^3 - x_1 x_2^2) + x_2^2(x_1^2 + x_2^2)$,

$$in_<(I) = (x_1^2, x_1 x_2^2, x_2^4)$$

and $x_2^3 - x_1 x_2^2, x_1^2 + x_2^2, x_2^4$ is a (lex) Gröbner basis of I.

Gröbner bases

(Finite) Gröbner bases always exists, since the initial ideal $\mathrm{in}(I)$ is finitely generated by Noetherianity.

Gröbner bases

(Finite) Gröbner bases always exists, since the initial ideal $\mathrm{in}(I)$ is finitely generated by Noetherianity.

There is a flat family whose special fiber is $\mathrm{in}(I)$ and whose general fiber is I. This implies that many algebraic invariants and properties are preserved when passing from I to $\mathrm{in}(I)$. This makes Gröbner bases an important tool in commutative algebra and algebraic geometry.

Gröbner bases

(Finite) Gröbner bases always exists, since the initial ideal $\mathrm{in}(I)$ is finitely generated by Noetherianity.

There is a flat family whose special fiber is $\mathrm{in}(I)$ and whose general fiber is I. This implies that many algebraic invariants and properties are preserved when passing from I to $\mathrm{in}(I)$. This makes Gröbner bases an important tool in commutative algebra and algebraic geometry.

Definition

A Gröbner basis $g_1, \ldots, g_s$ of I is reduced if $\mathrm{in}(g_1), \ldots, \mathrm{in}(g_s)$ are a minimal system of generators of $\mathrm{in}(I)$ and $\mathrm{in}(g_i)$ does not divide any monomial in the support of g_j for $j \neq i$.

The importance of being lex

Proposition (Shape Lemma)

Fix the lex term order on $R = K[x_1, \ldots, x_n]$, $I = (\mathcal{F}) = (f_1, \ldots, f_m) \subseteq R$.

Assume that $\mathcal{F}$ *has finitely many solutions over* $\overline{K}$ *and for any solutions* $\alpha, \beta \in \overline{K}^n$ $\alpha_n \neq \beta_n$. *If* $(\mathcal{F})$ *is radical, then the reduced Gröbner basis of* $(\mathcal{F})$ *has the form*

$$x_1 - h_1(x_n),\ x_2 - h_2(x_n), \ldots,\ x_{n-1} - h_{n-1}(x_n),\ h_n(x_n)$$

where $\deg(h_n) =$ *number of solutions of* $f_1 = \ldots = f_m = 0$.

The importance of being lex

Proposition (Shape Lemma)

Fix the lex term order on $R = K[x_1, \ldots, x_n]$*,* $I = (\mathcal{F}) = (f_1, \ldots, f_m) \subseteq R$*.*

Assume that $\mathcal{F}$ *has finitely many solutions over* $\overline{K}$ *and for any solutions* $\alpha, \beta \in \overline{K}^n$ $\alpha_n \neq \beta_n$*. If* $(\mathcal{F})$ *is radical, then the reduced Gröbner basis of* $(\mathcal{F})$ *has the form*

$$x_1 - h_1(x_n),\ x_2 - h_2(x_n), \ldots,\ x_{n-1} - h_{n-1}(x_n),\ h_n(x_n)$$

where $\deg(h_n) =$ *number of solutions of* $f_1 = \ldots = f_m = 0$*.*

Hence to solve the polynomial system $f_1 = \ldots = f_m = 0$ we:

- compute a reduced lex Gröbner basis of $(\mathcal{F})$,
- factor $h_n(x_n)$ to find its roots,
- for each a s.t. $h_n(a) = 0$ we have a solution $(h_1(a), \ldots, h_{n-1}(a), a)$.

The importance of being lex

Proposition (Shape Lemma)

Fix the lex term order on $R = K[x_1, \dots, x_n]$, $I = (\mathcal{F}) = (f_1, \dots, f_m) \subseteq R$.

Assume that $\mathcal{F}$ *has finitely many solutions over* $\overline{K}$ *and for any solutions* $\alpha, \beta \in \overline{K}^n$ $\alpha_n \neq \beta_n$. *If* $(\mathcal{F})$ *is radical, then the reduced Gröbner basis of* $(\mathcal{F})$ *has the form*

$$x_1 - h_1(x_n),\ x_2 - h_2(x_n), \dots,\ x_{n-1} - h_{n-1}(x_n),\ h_n(x_n)$$

where $\deg(h_n) =$ *number of solutions of* $f_1 = \ldots = f_m = 0$.

I is radical if $f^d \in I$ implies $f \in I$. E.g. (x) is radical and (x^2) is not.
If $K = \mathbb{F}_q$ and $\mathcal{F} = \{f_1, \dots, f_m\}$ contains the field equations, then $(\mathcal{F})$ is radical.

Hence to solve the polynomial system $f_1 = \ldots = f_m = 0$ we:

- compute a reduced lex Gröbner basis of $(\mathcal{F})$,
- factor $h_n(x_n)$ to find its roots,
- for each a s.t. $h_n(a) = 0$ we have a solution $(h_1(a), \dots, h_{n-1}(a), a)$.

The importance of being lex

Proposition (Shape Lemma)

Fix the lex term order on $R = K[x_1, \ldots, x_n]$, $I = (\mathcal{F}) = (f_1, \ldots, f_m) \subseteq R$.

Assume that $\mathcal{F}$ has finitely many solutions over $\overline{K}$ and for any solutions $\alpha, \beta \in \overline{K}^n$ $\alpha_n \neq \beta_n$. If $(\mathcal{F})$ is radical, then the reduced Gröbner basis of $(\mathcal{F})$ has the form

$$x_1 - h_1(x_n),\ x_2 - h_2(x_n), \ldots,\ x_{n-1} - h_{n-1}(x_n),\ h_n(x_n)$$

where $\deg(h_n) =$ *number of solutions of* $f_1 = \ldots = f_m = 0$.

I is radical if $f^d \in I$ implies $f \in I$. E.g. (x) is radical and (x^2) is not.
If $K = \mathbb{F}_q$ and $\mathcal{F} = \{f_1, \ldots, f_m\}$ contains the field equations, then $(\mathcal{F})$ is radical.

The assumption on the solutions is true after applying a change of coordinates to $\mathcal{F}$, possibly over a field extension.

Hence to solve the polynomial system $f_1 = \ldots = f_m = 0$ we:

- compute a reduced lex Gröbner basis of $(\mathcal{F})$,
- factor $h_n(x_n)$ to find its roots,
- for each a s.t. $h_n(a) = 0$ we have a solution $(h_1(a), \ldots, h_{n-1}(a), a)$.

Buchberger's Algorithm

It generalizes Gaussian elimination and the Euclidean Algorithm.

Example

$f_1 = x_1x_2 + x_2,\ f_2 = x_2^2 - 1$, lex order

$g_1 := f_1,\ g_2 := f_2,$

Buchberger's Algorithm

It generalizes Gaussian elimination and the Euclidean Algorithm.

Example

$f_1 = x_1 x_2 + x_2,\ f_2 = x_2^2 - 1$, lex order

$g_1 := f_1,\ g_2 := f_2,\quad \mathrm{lcm}\{\mathrm{in}(g_1), \mathrm{in}(g_2)\} = \mathrm{lcm}\{x_1 x_2, x_2^2\} = x_1 x_2^2$

Buchberger's Algorithm

It generalizes Gaussian elimination and the Euclidean Algorithm.

Example

$f_1 = x_1x_2 + x_2,\ \ f_2 = x_2^2 - 1$, lex order

$g_1 := f_1,\ \ g_2 := f_2,\ \ \operatorname{lcm}\{\operatorname{in}(g_1), \operatorname{in}(g_2)\} = \operatorname{lcm}\{x_1x_2, x_2^2\} = x_1x_2^2$

$S(g_1, g_2) = \frac{x_1x_2^2}{\operatorname{in}(g_1)} g_1 - \frac{x_1x_2^2}{\operatorname{in}(g_2)} g_2 = x_2(x_1x_2 + x_2) - x_1(x_2^2 - 1) = x_2^2 + x_1 \rightsquigarrow x_1 + 1$

Buchberger's Algorithm

It generalizes Gaussian elimination and the Euclidean Algorithm.

Example

$f_1 = x_1x_2 + x_2,\ f_2 = x_2^2 - 1$, lex order

$g_1 := f_1,\ g_2 := f_2,\ \ \mathrm{lcm}\{\mathrm{in}(g_1), \mathrm{in}(g_2)\} = \mathrm{lcm}\{x_1x_2, x_2^2\} = x_1x_2^2$

$S(g_1, g_2) = \frac{x_1x_2^2}{\mathrm{in}(g_1)}g_1 - \frac{x_1x_2^2}{\mathrm{in}(g_2)}g_2 = x_2(x_1x_2 + x_2) - x_1(x_2^2 - 1) = x_2^2 + x_1 \rightsquigarrow x_1 + 1$

$g_3 := x_1 + 1$

Buchberger's Algorithm

It generalizes Gaussian elimination and the Euclidean Algorithm.

Example

$f_1 = x_1x_2 + x_2,\ f_2 = x_2^2 - 1$, lex order

$g_1 := f_1,\ g_2 := f_2,\ \ \mathrm{lcm}\{\mathrm{in}(g_1), \mathrm{in}(g_2)\} = \mathrm{lcm}\{x_1x_2, x_2^2\} = x_1x_2^2$

$S(g_1, g_2) = \frac{x_1x_2^2}{\mathrm{in}(g_1)}g_1 - \frac{x_1x_2^2}{\mathrm{in}(g_2)}g_2 = x_2(x_1x_2 + x_2) - x_1(x_2^2 - 1) = x_2^2 + x_1 \rightsquigarrow x_1 + 1$

$g_3 := x_1 + 1$

$S(g_1, g_3) = g_1 - x_2g_3 = x_2^2 - 1 = 0$

Buchberger's Algorithm

It generalizes Gaussian elimination and the Euclidean Algorithm.

Example

$f_1 = x_1x_2 + x_2,\ \ f_2 = x_2^2 - 1$, lex order

$g_1 := f_1,\ \ g_2 := f_2,\quad \mathrm{lcm}\{\mathrm{in}(g_1), \mathrm{in}(g_2)\} = \mathrm{lcm}\{x_1x_2, x_2^2\} = x_1x_2^2$

$S(g_1, g_2) = \frac{x_1x_2^2}{\mathrm{in}(g_1)} g_1 - \frac{x_1x_2^2}{\mathrm{in}(g_2)} g_2 = x_2(x_1x_2 + x_2) - x_1(x_2^2 - 1) = x_2^2 + x_1 \rightsquigarrow x_1 + 1$

$g_3 := x_1 + 1$

$S(g_1, g_3) = g_1 - x_2g_3 = x_2^2 - 1 = 0$

$S(g_2, g_3) = x_1g_2 - x_2^2g_3 = -x_1 - x_2^2 \rightsquigarrow -x_1 - 1 \rightsquigarrow 0.$

Hence $x_1x_2 + x_2, x_2^2 - 1, x_1 + 1$ are a lexicographic Gröbner basis of (f_1, f_2).

Buchberger's Algorithm

It generalizes Gaussian elimination and the Euclidean Algorithm.

Example

$f_1 = x_1x_2 + x_2,\ \ f_2 = x_2^2 - 1$, lex order

$g_1 := f_1,\ \ g_2 := f_2,\ \ \mathrm{lcm}\{\mathrm{in}(g_1), \mathrm{in}(g_2)\} = \mathrm{lcm}\{x_1x_2, x_2^2\} = x_1x_2^2$

$S(g_1, g_2) = \frac{x_1x_2^2}{\mathrm{in}(g_1)} g_1 - \frac{x_1x_2^2}{\mathrm{in}(g_2)} g_2 = x_2(x_1x_2 + x_2) - x_1(x_2^2 - 1) = x_2^2 + x_1 \rightsquigarrow x_1 + 1$

$g_3 := x_1 + 1$

$S(g_1, g_3) = g_1 - x_2g_3 = x_2^2 - 1 = 0$

$S(g_2, g_3) = x_1g_2 - x_2^2g_3 = -x_1 - x_2^2 \rightsquigarrow -x_1 - 1 \rightsquigarrow 0.$

Hence $x_1x_2 + x_2, x_2^2 - 1, x_1 + 1$ are a lexicographic Gröbner basis of (f_1, f_2).

Buchberger's Algorithm computes and reduces S-pairs for each pair of elements in the Gröbner basis and adds the results to the Gröbner basis. When all the S-pairs reduce to zero, a Gröbner basis has been found.

Linear algebra based algorithms

They are the most efficient. They include F_4/F_5 and XL and its variants.

Linear algebra based algorithms

They are the most efficient. They include F_4/F_5 and XL and its variants.

Definition

For each degree d one has a Macaulay matrix:

- columns $\leftrightarrow$ monomials of degree $\leq d$
- rows $\leftrightarrow$ polynomials $x^a f_i$ of degree $\leq d$
- the entry (i, j) is the coefficient of the monomial corresponding to column j in the polynomial corresponding to row i

The matrix is brought in RREF. If the rows are not a Gröbner basis of $I = (f_1, \ldots, f_m)$, then one looks at the Macaulay matrix in the next degree.

Some variants add new rows to the matrix, whenever a degree drop occurs.

Example

$f_1 = x_1x_2 + x_2, \; f_2 = x_2^2 - 1$, lex order

	x_1^2	x_1x_2	x_1	x_2^2	x_2	1
f_1	0	1	0	0	1	0
f_2	0	0	0	1	0	-1

Example

$f_1 = x_1x_2 + x_2, \; f_2 = x_2^2 - 1$, lex order

	x_1^2	x_1x_2	x_1	x_2^2	x_2	1
f_1	0	1	0	0	1	0
f_2	0	0	0	1	0	-1

	x_1^3	$x_1^2x_2$	x_1^2	$x_1x_2^2$	x_1x_2	x_1	x_2^3	x_2^2	x_2	1
x_1f_1	0	1	0	0	1	0	0	0	0	0
x_1f_2	0	0	0	1	0	-1	0	0	0	0
x_2f_1	0	0	0	1	0	0	0	1	0	0
x_2f_2	0	0	0	0	0	0	1	0	-1	0
f_1	0	0	0	0	1	0	0	0	1	0
f_2	0	0	0	0	0	0	0	1	0	-1

Example

$f_1 = x_1x_2 + x_2,\ \ f_2 = x_2^2 - 1$, lex order

	x_1^2	x_1x_2	x_1	x_2^2	x_2	1
f_1	0	1	0	0	1	0
f_2	0	0	0	1	0	-1

	x_1^3	$x_1^2x_2$	x_1^2	$x_1x_2^2$	x_1x_2	x_1	x_2^3	x_2^2	x_2	1
x_1f_1	0	1	0	0	1	0	0	0	0	0
x_1f_2	0	0	0	1	0	-1	0	0	0	0
x_2f_1	0	0	0	1	0	0	0	1	0	0
x_2f_2	0	0	0	0	0	0	1	0	-1	0
f_1	0	0	0	0	1	0	0	0	1	0
f_2	0	0	0	0	0	0	0	1	0	-1

Example

$f_1 = x_1x_2 + x_2,\ \ f_2 = x_2^2 - 1$, lex order

	x_1^2	x_1x_2	x_1	x_2^2	x_2	1
f_1	0	1	0	0	1	0
f_2	0	0	0	1	0	-1

	x_1^3	$x_1^2x_2$	x_1^2	$x_1x_2^2$	x_1x_2	x_1	x_2^3	x_2^2	x_2	1
	0	1	0	0	0	0	0	0	-1	0
	0	0	0	1	0	0	0	0	0	1
	0	0	0	0	1	0	0	0	1	0
	0	0	0	0	0	1	0	0	0	1
	0	0	0	0	0	0	1	0	-1	0
	0	0	0	0	0	0	0	1	0	-1

$x_1 + 1$ and $x_2^2 - 1$ are a reduced lexicographic Gröbner basis of (f_1, f_2).

Example

$f_1 = x_1x_2 + x_2,\ \ f_2 = x_2^2 - 1$, lex order

	x_1^2	x_1x_2	x_1	x_2^2	x_2	1
f_1	0	1	0	0	1	0
f_2	0	0	0	1	0	-1

	x_1^3	$x_1^2x_2$	x_1^2	$x_1x_2^2$	x_1x_2	x_1	x_2^3	x_2^2	x_2	1
	0	1	0	0	0	0	0	0	-1	0
	0	0	0	1	0	0	0	0	0	1
	0	0	0	0	1	0	0	0	1	0
	0	0	0	0	0	1	0	0	0	1
	0	0	0	0	0	0	1	0	-1	0
	0	0	0	0	0	0	0	1	0	-1

$x_1 + 1$ and $x_2^2 - 1$ are a reduced lexicographic Gröbner basis of (f_1, f_2).

Gröbner bases computations and change of order

The complexity of computing a Gröbner basis...

... is usually largest for the lexicographic order and smallest for the degree reverse lexicographic order

Gröbner bases computations and change of order

The complexity of computing a Gröbner basis...

... is usually largest for the lexicographic order and smallest for the degree reverse lexicographic order

Algorithm (Faugère, Gianni, Lazard, Mora)

A Gröbner basis for $I = (f_1, \ldots, f_m)$ wrt a given term order can be converted into a Gröbner basis for I wrt a different term order with $\mathcal{O}(n^2 d^3)$ operations, where d is the number of solutions of $f_1 = \ldots = f_m = 0$.

Polynomial systems of cryptographic interest typically have $d = 1$ or d very small.

Gröbner bases computations and change of order

The complexity of computing a Gröbner basis...

... is usually largest for the lexicographic order and smallest for the degree reverse lexicographic order

... is dominated by the cost of Gaussian elimination in the largest matrix

Algorithm (Faugère, Gianni, Lazard, Mora)

A Gröbner basis for $I = (f_1, \ldots, f_m)$ wrt a given term order can be converted into a Gröbner basis for I wrt a different term order with $\mathcal{O}(n^2 d^3)$ operations, where d is the number of solutions of $f_1 = \ldots = f_m = 0$.

Polynomial systems of cryptographic interest typically have $d = 1$ or d very small.

Computing a lex Gröbner basis in practice

- compute a drl Gröbner basis using a linear algebra based algorithm
- convert it into a lex one using the FGLM Algorithm

Computing a lex Gröbner basis in practice

- compute a drl Gröbner basis using a linear algebra based algorithm
- convert it into a lex one using the FGLM Algorithm

For cryptographic systems, the complexity is dominated by the first step.

Theorem

The complexity of Gaussian elimination in an $a \times b$ matrix is $\mathcal{O}(a^2 b)$ operations in K.

If we compute matrices up to degree s, then the largest has

$$a = \sum_{i=1}^{m} \binom{n+s-d_i-1}{s-d_i} \quad \text{and} \quad b = \binom{n+s-1}{s}$$

where $d_i = \deg(f_i)$.

Solving degree

Let $\mathcal{F} = \{f_1, \ldots, f_m\}$, fix the degree reverse lexicographic order.

Definition

The solving degree of $\mathcal{F}$, denoted $\mathrm{solv.deg}(\mathcal{F})$, is the least degree for which Gaussian elimination in the drl Macaulay matrix of degree d yields a Gröbner basis of $(\mathcal{F}) = (f_1, \ldots, f_m)$.
$\mathrm{max.GB.deg}(\mathcal{F})$ denotes the largest degree of a polynomial in a reduced drl Gröbner basis of $(\mathcal{F})$.

Solving degree

Let $\mathcal{F} = \{f_1, \ldots, f_m\}$, fix the degree reverse lexicographic order.

Definition

The solving degree of $\mathcal{F}$, denoted $\operatorname{solv.deg}(\mathcal{F})$, is the least degree for which Gaussian elimination in the drl Macaulay matrix of degree d yields a Gröbner basis of $(\mathcal{F}) = (f_1, \ldots, f_m)$.
$\operatorname{max.GB.deg}(\mathcal{F})$ denotes the largest degree of a polynomial in a reduced drl Gröbner basis of $(\mathcal{F})$.

Remark

$\operatorname{solv.deg}(\mathcal{F}) \geq \operatorname{max.GB.deg}(\mathcal{F})$
$\operatorname{solv.deg}(\mathcal{F}) = \operatorname{max.GB.deg}(\mathcal{F})$ if $f_1, \ldots, f_m$ are homogeneous

Solving degree

Let $\mathcal{F} = \{f_1, \ldots, f_m\}$, fix the degree reverse lexicographic order.

Definition

The solving degree of $\mathcal{F}$, denoted $\operatorname{solv.deg}(\mathcal{F})$, is the least degree for which Gaussian elimination in the drl Macaulay matrix of degree d yields a Gröbner basis of $(\mathcal{F}) = (f_1, \ldots, f_m)$.
$\operatorname{max.GB.deg}(\mathcal{F})$ denotes the largest degree of a polynomial in a reduced drl Gröbner basis of $(\mathcal{F})$.

Remark

$\operatorname{solv.deg}(\mathcal{F}) \geq \operatorname{max.GB.deg}(\mathcal{F})$
$\operatorname{solv.deg}(\mathcal{F}) = \operatorname{max.GB.deg}(\mathcal{F})$ if $f_1, \ldots, f_m$ are homogeneous

Example

The Gröbner basis of $f_1 = x_1x_2 + x_2,\; f_2 = x_2^2 - 1$ wrt the lexicographic order is $x_1 + 1, x_2^2 - 1$, so $\operatorname{max.GB.deg}(\mathcal{F}) = 2 < 3 = \operatorname{solv.deg}(\mathcal{F})$.

Homogeneous polynomials and homogenization

Definition

A polynomial f is homogeneous if all the monomials in the support of f have the same degree.

E.g., $x_1^2x_3 - 2x_2^3$ is homogeneous, but $x_1^2x_3 - 2x_2$ is not.

Homogeneous polynomials and homogenization

Definition

A polynomial f is homogeneous if all the monomials in the support of f have the same degree.

E.g., $x_1^2 x_3 - 2x_2^3$ is homogeneous, but $x_1^2 x_3 - 2x_2$ is not.

Definition

The homogenization of $f = \sum_{a \in \mathbb{N}^n} \alpha_a x^a \in K[x_1, \ldots, x_n]$ wrt x_0 is

$$f^h = \sum_{a \in \mathbb{N}^n} \alpha_a x^a x_0^{\deg(f) - |a|} \in K[x_0, \ldots, x_n],$$

where $|a| = a_1 + \ldots + a_n = \deg(x^a)$.

E.g., the homogenization of $f = x_1^2 x_3 - 2x_2$ wrt x_0 is $f^h = x_1^2 x_3 - 2x_0^2 x_2$.

A provable bound for the solving degree

Let $I = (f_1, \ldots, f_m) \subseteq R = K[x_1, \ldots, x_n]$, $\deg(f_i) = d_i$, $d_1 \geq \ldots \geq d_m$
$\mathcal{F}^h = \{f_1^h, \ldots, f_m^h\}, \qquad \subseteq S = K[x_0, \ldots, x_n]$.

Theorem (Lazard)

Suppose that $(\mathcal{F}^h)$ is in generic coordinates, then
$\operatorname{solv.deg}(I) \leq d_1 + \ldots + d_{n+1} - n$.

A provable bound for the solving degree

Let $I = (f_1, \ldots, f_m) \subseteq R = K[x_1, \ldots, x_n]$, $\deg(f_i) = d_i$, $d_1 \geq \ldots \geq d_m$
$\mathcal{F}^h = \{f_1^h, \ldots, f_m^h\}$, $\subseteq S = K[x_0, \ldots, x_n]$.

Theorem (Lazard)

Suppose that $(\mathcal{F}^h)$ is in generic coordinates, then
$\mathrm{solv.deg}(I) \leq d_1 + \ldots + d_{n+1} - n.$

Theorem (Caminata, G.)

Suppose that $(\mathcal{F}^h)$ is in generic coordinates, then
$\mathrm{reg}(\mathcal{F}^h) \geq \mathrm{max.GB.deg}(\mathcal{F}^h) = \mathrm{solv.deg}(\mathcal{F}^h) \geq \mathrm{solv.deg}(\mathcal{F})$

where $\mathrm{reg}(\mathcal{F}^h)$ *is the Castelnuovo-Mumford regularity of* $(\mathcal{F}^h)$.

A provable bound for the solving degree

Let $I = (f_1, \ldots, f_m) \subseteq R = K[x_1, \ldots, x_n]$, $\deg(f_i) = d_i$, $d_1 \geq \ldots \geq d_m$
$\mathcal{F}^h = \{f_1^h, \ldots, f_m^h\}$, $(\mathcal{F}^h) \subseteq I^h = (f^h \mid f \in I) \subseteq S = K[x_0, \ldots, x_n]$.

Theorem (Lazard)

Suppose that $(\mathcal{F}^h)$ is in generic coordinates, then
$\operatorname{solv.deg}(I) \leq d_1 + \ldots + d_{n+1} - n.$

Theorem (Caminata, G.)

Suppose that $(\mathcal{F}^h)$ is in generic coordinates, then

$$\operatorname{reg}(\mathcal{F}^h) \geq \operatorname{max.GB.deg}(\mathcal{F}^h) = \operatorname{solv.deg}(\mathcal{F}^h) \geq \operatorname{solv.deg}(\mathcal{F}) \geq \operatorname{max.GB.deg}(\mathcal{F}) = \operatorname{max.GB.deg}(I^h) = \operatorname{solv.deg}(I^h)$$

where $\operatorname{reg}(\mathcal{F}^h)$ *is the Castelnuovo-Mumford regularity of* $(\mathcal{F}^h)$.

The Castelnuovo-Mumford regularity

$J = (F_1, \ldots, F_m)$, $F_i \in S = K[x_0, \ldots, x_n]$ homogeneous of $\deg(F_i) = d_i$

J

The Castelnuovo-Mumford regularity

$J = (F_1, \ldots, F_m)$, $F_i \in S = K[x_0, \ldots, x_n]$ homogeneous of $\deg(F_i) = d_i$

$$\bigoplus_{i=1}^{m} S(-d_i) \xrightarrow{(F_1,\ldots,F_m)} J \to 0$$

The Castelnuovo-Mumford regularity

$J = (F_1, \ldots, F_m)$, $F_i \in S = K[x_0, \ldots, x_n]$ homogeneous of $\deg(F_i) = d_i$

$$\bigoplus_{i=1}^{\ell_1} S(-b_{1,i}) \to \bigoplus_{i=1}^{m} S(-d_i) \xrightarrow{(F_1,\ldots,F_m)} J \to 0$$

The Castelnuovo-Mumford regularity

$J = (F_1, \ldots, F_m)$, $F_i \in S = K[x_0, \ldots, x_n]$ homogeneous of $\deg(F_i) = d_i$

$$\cdots \to \bigoplus_{i=1}^{\ell_1} S(-b_{1,i}) \to \bigoplus_{i=1}^{m} S(-d_i) \xrightarrow{(F_1,\ldots,F_m)} J \to 0$$

The Castelnuovo-Mumford regularity

$J = (F_1, \ldots, F_m)$, $F_i \in S = K[x_0, \ldots, x_n]$ homogeneous of $\deg(F_i) = d_i$

$$0 \to \bigoplus_{i=1}^{\ell_p} S(-b_{p,i}) \to \cdots \to \bigoplus_{i=1}^{\ell_1} S(-b_{1,i}) \to \bigoplus_{i=1}^{m} S(-d_i) \xrightarrow{(F_1,\ldots,F_m)} J \to 0$$

The Castelnuovo-Mumford regularity

$J = (F_1, \ldots, F_m)$, $F_i \in S = K[x_0, \ldots, x_n]$ homogeneous of $\deg(F_i) = d_i$

$$0 \to \bigoplus_{i=1}^{\ell_p} S(-b_{p,i}) \to \cdots \to \bigoplus_{i=1}^{\ell_1} S(-b_{1,i}) \to \bigoplus_{i=1}^{m} S(-d_i) \xrightarrow{(F_1,\ldots,F_m)} J \to 0$$

Definition

The Castelnuovo-Mumford regularity of J is $\operatorname{reg}(J) = \max\{b_{j,i} - j, d_i\}$.

The Castelnuovo-Mumford regularity

$J = (F_1, \ldots, F_m)$, $F_i \in S = K[x_0, \ldots, x_n]$ homogeneous of $\deg(F_i) = d_i$

$$0 \to \bigoplus_{i=1}^{\ell_p} S(-b_{p,i}) \to \cdots \to \bigoplus_{i=1}^{\ell_1} S(-b_{1,i}) \to \bigoplus_{i=1}^{m} S(-d_i) \xrightarrow{(F_1,\ldots,F_m)} J \to 0$$

Definition

The Castelnuovo-Mumford regularity of J is $\mathrm{reg}(J) = \max\{b_{j,i} - j, d_i\}$.

How to compute the Castelnuovo-Mumford regularity of $\mathcal{F}^h$?

The Castelnuovo-Mumford regularity

$J = (F_1, \ldots, F_m)$, $F_i \in S = K[x_0, \ldots, x_n]$ homogeneous of $\deg(F_i) = d_i$

$$0 \to \bigoplus_{i=1}^{\ell_p} S(-b_{p,i}) \to \cdots \to \bigoplus_{i=1}^{\ell_1} S(-b_{1,i}) \to \bigoplus_{i=1}^{m} S(-d_i) \xrightarrow{(F_1,\ldots,F_m)} J \to 0$$

Definition

The Castelnuovo-Mumford regularity of J is $\mathrm{reg}(J) = \max\{b_{j,i} - j, d_i\}$.

How to compute the Castelnuovo-Mumford regularity of $\mathcal{F}^h$?

We can compute it from a Gröbner basis of $\mathcal{F}^h$.

The Castelnuovo-Mumford regularity

$J = (F_1, \ldots, F_m)$, $F_i \in S = K[x_0, \ldots, x_n]$ homogeneous of $\deg(F_i) = d_i$

$$0 \to \bigoplus_{i=1}^{\ell_p} S(-b_{p,i}) \to \cdots \to \bigoplus_{i=1}^{\ell_1} S(-b_{1,i}) \to \bigoplus_{i=1}^{m} S(-d_i) \xrightarrow{(F_1,\ldots,F_m)} J \to 0$$

Definition

The Castelnuovo-Mumford regularity of J is $\mathrm{reg}(J) = \max\{b_{j,i} - j, d_i\}$.

How to compute the Castelnuovo-Mumford regularity of $\mathcal{F}^h$?

We can compute it from a Gröbner basis of $\mathcal{F}^h$.

Have we made any progress?

The Castelnuovo-Mumford regularity

$J = (F_1, \ldots, F_m)$, $F_i \in S = K[x_0, \ldots, x_n]$ homogeneous of $\deg(F_i) = d_i$

$$0 \to \bigoplus_{i=1}^{\ell_p} S(-b_{p,i}) \to \cdots \to \bigoplus_{i=1}^{\ell_1} S(-b_{1,i}) \to \bigoplus_{i=1}^{m} S(-d_i) \xrightarrow{(F_1,\ldots,F_m)} J \to 0$$

Definition

The Castelnuovo-Mumford regularity of J is $\mathrm{reg}(J) = \max\{b_{j,i} - j, d_i\}$.

How to compute the Castelnuovo-Mumford regularity of $\mathcal{F}^h$?

We can compute it from a Gröbner basis of $\mathcal{F}^h$.

Have we made any progress?

Yes, because we know a lot on the Castelnuovo-Mumford regularity.

Example – The complexity of MinRank

MinRank Problem

Given $M_1, \ldots, M_n, N \in \mathrm{Mat}_{k\times m}(\mathbb{F}_q)$ and $r < \min\{k, m\}$, find $x_1, \ldots, x_n \in \mathbb{F}_q$ s.t.

$$\mathrm{rank}\left(N - \sum_{i=1}^{n} x_i M_i\right) \leq r.$$

Example – The complexity of MinRank

Generalized MinRank Problem

Given $M \in \mathrm{Mat}_{k\times m}(K[x_1,\dots,x_n])$ and $r < \min\{k,m\}$, find $x_1,\dots,x_n \in K$ s.t.

$$\mathrm{rank}\,(M) \leq r.$$

Example – The complexity of MinRank

Generalized MinRank Problem

Given $M \in \mathrm{Mat}_{k\times m}(K[x_1,\dots,x_n])$ and $r < \min\{k,m\}$, find $x_1,\dots,x_n \in K$ s.t.

$$\mathrm{rank}\,(M) \le r.$$

The next result was shown by Faugère, Safey El Din, and Spaenlehauer for $d_{ij} = d \ge 1$.

Theorem (Caminata, G.)

Let $M \in \mathrm{Mat}_{k\times m}(R)$, let $r < k \le m$ and $n \ge (m-r)(k-r)$.
Assume that the entries of M are generic of degree d_{ij} with $d_{ij} > 0$ and $d_{ij} + d_{h\ell} = d_{i\ell} + d_{hj}$ for all i,j,h,ℓ.
Let $\mathcal{F}$ be the homogeneous polynomial system of the minors of size $r+1$ of M. Then

$$\mathrm{solv.deg}(\mathcal{F}) \le (m-r)\sum_{i=1}^{r} d_{i,i} + \sum_{i=r+1}^{k}\sum_{j=r+1}^{m} d_{ij} - (m-r)(k-r) + 1.$$

Algebra and geometry

K field, $\mathcal{F} = \{f_1, \ldots, f_m\} \subseteq R = K[x_1, \ldots, x_n]$, $I = (\mathcal{F})$

Definition

The affine variety associated to I is

$$V(I) = \{P = (x_1, \ldots, x_n) \in K^n \mid f_1(P) = \ldots = f_m(P) = 0\} \subseteq K^n.$$

Algebra and geometry

K field, $\mathcal{F} = \{f_1, \dots, f_m\} \subseteq R = K[x_1, \dots, x_n]$, $I = (\mathcal{F})$

Definition

The affine variety associated to I is

$$V(I) = \{P = (x_1, \dots, x_n) \in K^n \mid f_1(P) = \dots = f_m(P) = 0\} \subseteq K^n.$$

Theorem (Hilbert's Nullstellensatz)

If $K = \overline{K}$, then we have a one-to-one correspondence between radical ideals and affine varieties.

Algebra and geometry

K field, $\mathcal{F} = \{f_1, \dots, f_m\} \subseteq R = K[x_1, \dots, x_n]$, $I = (\mathcal{F})$

Definition

The affine variety associated to I is

$$V(I) = \{P = (x_1, \dots, x_n) \in K^n \mid f_1(P) = \dots = f_m(P) = 0\} \subseteq K^n.$$

Theorem (Hilbert's Nullstellensatz)

If $K = \overline{K}$, then we have a one-to-one correspondence between radical ideals and affine varieties.

Affine varieties in K^n are the closed sets of the Zarisky topology on K^n.

If $K = \mathbb{F}_q$, then the Zarisky topology is the discrete topology.

If K is infinite, then any $\emptyset \neq U \subseteq K^n$ open is dense, i.e. $\overline{U} = K^n$.

Genericity

Definition

A property is generic if it holds on a nonempty Zarisky-open set.

Over a finite field this is meaningless, but over an infinite field this means that the property holds "almost everywhere".

Genericity

Definition

A property is generic if it holds on a nonempty Zarisky-open set.

Over a finite field this is meaningless, but over an infinite field this means that the property holds "almost everywhere". However, when one can describe the open set via the equations of its complement, then one can check whether any given point belongs to the open set.

Example

Genericity conditions for the statement on the complexity of MinRank:

- the homogenization of the minors of M are the minors of the matrix obtained from M by homogenizing its entries,
- the zero locus of the minors has codimension $(m-r)(k-r)$.

Ideals in generic coordinates

$K = \overline{K}$, $S = K[x_0, \ldots, x_n]$, $J \subseteq S$ homogeneous
$G = \mathrm{GL}_{n+1}(K)$ acts on S as changes of coordinates

Theorem (Galligo)

There is a nonempty open $U \subseteq G \subseteq K^{(n+1)^2}$ s.t. $\mathrm{in}(gJ) = \mathrm{in}(hJ)$ for $g, h \in U$.

Definition

$\mathrm{gin}(J) := \mathrm{in}(gJ)$ for $g \in U$ is the generic initial ideal of J wrt the chosen term order.

Ideals in generic coordinates

$K = \overline{K}$, $S = K[x_0, \ldots, x_n]$, $J \subseteq S$ homogeneous
$G = \mathrm{GL}_{n+1}(K)$ acts on S as changes of coordinates

Theorem (Galligo)

There is a nonempty open $U \subseteq G \subseteq K^{(n+1)^2}$ s.t. $\mathrm{in}(gJ) = \mathrm{in}(hJ)$ for $g, h \in U$.

Definition

$\mathrm{gin}(J) := \mathrm{in}(gJ)$ for $g \in U$ is the generic initial ideal of J wrt the chosen term order.

Theorem (Bayer, Stillman)

Fix the degree reverse lexicographic order, then

$$\mathrm{reg}(J) = \mathrm{reg}(\mathrm{gin}(J)).$$

Hence, if J is in generic coordinates, then

$$\mathrm{reg}(J) = \mathrm{reg}(\mathrm{in}(J)).$$

Are we in generic coordinates?

I do not know of any deterministic algorithm that does that.

Are we in generic coordinates?

I do not know of any deterministic algorithm that does that.

One could check whether $\mathrm{in}(J) = \mathrm{in}(gJ)$ for a random $g \in G$, but this only shows that J is in generic coordinates with high probability.

Are we in generic coordinates?

I do not know of any deterministic algorithm that does that.

One could check whether $\mathrm{in}(J) = \mathrm{in}(gJ)$ for a random $g \in G$, but this only shows that J is in generic coordinates with high probability.

Theorem (Caminata, G.)

$\mathcal{F} \subseteq \mathbb{F}_q[x_1, \ldots, x_n]$. Assume that

$$x_1^q - x_1, \ldots, x_n^q - x_n \in \mathcal{F} \quad \text{or} \quad x_1^q - x_2, \ldots, x_{n-1}^q - x_n, x_n^q - x_1 \in \mathcal{F}.$$

Then $(\mathcal{F}^h)$ is in generic coordinates.

Corollary (Macaulay Bound)

$\mathcal{F} = \{f_1, \ldots, f_m\} \subseteq R = \mathbb{F}_q[x_1, \ldots, x_n]$, $\deg(f_i) = d_i$, $d_1 \geq \ldots \geq d_m$, $m \geq n+1$. *Assume that $(\mathcal{F}^h)$ is in generic coordinates, or that $\mathcal{F}$ contains the field equations. Then*

$$\mathrm{solv.deg}(\mathcal{F}) \leq d_1 + \ldots + d_{n+1} - n.$$

Summary

- polynomial systems arise in many models from engineering and the sciences
- they can be solved over finite fields by computing a Gröbner basis
- the complexity of linear algebra algorithms for computing Gröbner bases is upper bounded by a function of the solving degree, which is the least degree for which Gaussian elimination in the Macaulay matrix yields a Gröbner basis
- the Castelnuovo-Mumford regularity of the homogenization of a system is an upper bound for its solving degree
- the arguments to prove this use the concept of genericity from algebraic geometry

Summary

- polynomial systems arise in many models from engineering and the sciences
- they can be solved over finite fields by computing a Gröbner basis
- the complexity of linear algebra algorithms for computing Gröbner bases is upper bounded by a function of the solving degree, which is the least degree for which Gaussian elimination in the Macaulay matrix yields a Gröbner basis
- the Castelnuovo-Mumford regularity of the homogenization of a system is an upper bound for its solving degree
- the arguments to prove this use the concept of genericity from algebraic geometry

Thank you for your attention!