

Solving Polynomial Equations

Teresa Krick
Universidad de Buenos Aires

Introduction

To “solve a polynomial equation system” usually means to describe in some understandable way the common zeroes of a finite set of polynomials in an arbitrary number of variables.

Are there solutions? In case there are, how many? A finite or an infinite number? What’s the shape, or the dimension of the solutions? How to describe them? These are natural questions when we deal with a system of equations.

If we consider polynomials with coefficients in the field of rational numbers $\mathbb{Q}$, or in the ring of integer numbers $\mathbb{Z}$ (this will be the context of these lectures) we can of course still apply all the numerical analysis techniques to approximate zeroes, but we can also proceed symbolically as much as we can in order to preserve exactness up to the last possible moment.

There exist many approaches which help to solve symbolically the questions raised above. For instance, a beautiful procedure, due to Buchberger (1965), is given by the search of a Gröbner Basis of a polynomial ideal ([Buch65], [Buch 85]), which modifies the generators of a given ideal and for certain orderings of monomials triangularizes them in such a way that one can actually read the information on the variety from the modified basis. Unfortunately, the complexity of Gröbner bases computations can be in the worst case extremely high, up to doubly exponential in the number of variables, so that some problems turn out to be intractable.

In any case, giving a better presentation of the generators of a zero-dimensional ideal in order to read better the information about the zeroes generally leads to the consideration of polynomials in a single variable, for which the roots have to be determined. The application of numerical analysis procedures is then unavoidable. However one can first try to factorize symbolically these polynomials, and postpone numerical analysis up to the end. Modern methods of factorization over $\mathbb{Q}[X]$ will be the topic of the first lecture.

The second lecture will analyze a useful presentation of the simplest possible case to be considered—the case of a zero-dimensional radical ideal—known as the Shape Lemma or the primitive element presentation.

In the last lecture, we’ll try to apply the zero-dimensional ideal presentation of Lecture 2 to the case of parametric ideals (some indeterminates are considered as parameters and the others are the true variables of the problem). The aim is to avoid working with too many variables because of the cost of operations it may imply. The idea is, given the total information on the variety in a good specialization of the parameters, to try to lift it in order to obtain the information for the original ideal.

1 Effective Factorization in $\mathbb{Z}[X]$

Unlike factoring a polynomial in $\mathbb{C}[X]$ or in $\mathbb{R}[X]$ (i.e. finding roots, which unavoidably leads to numerical analysis techniques), the question of factoring a polynomial in $\mathbb{Q}[X]$, or –what is essentially the same– in $\mathbb{Z}[X]$, has been shown to be effectively possible since a long time.

Around 1850, Kronecker established a very simple algorithm based on interpolation and divisibility in $\mathbb{Z}$; more precisely, the finiteness of the algorithm relies on the fact that any integer number has only a finite number of possible divisors. Although it is easy to see that the algorithm is absolutely intractable (its complexity is in the simplest cases singly exponential in the degree of the polynomial), the impact of the result comes from the fact that such an algorithm exists! Then, the problem was to find more tractable algorithms for factoring in $\mathbb{Z}[X]$.

Most modern algorithms of factorization over $\mathbb{Z}[X]$ rely on the following observation: If $f = gh \in \mathbb{Z}[X]$ is a factorization such that the absolute values of all the coefficients of f, g, h are bounded by $B \in \mathbb{N}$, then, when one considers all possible factorizations of $f \pmod{2B}$ —with the convention that “mod $2B$ ” means taking integers in the interval $[-B, B]$ — one of them must coincide with g and h .

The general structure of these algorithms is usually as follows:

1. — Factor $f \in \mathbb{Z}[X]$ in a product of squarefree polynomials. Then, without loss of generality, f can be assumed to be squarefree.
2. — Choose a prime p such that $f \pmod{p}$ is squarefree and factor $f \pmod{p}$.
3. — Via Newton–Hensel lifting, lift the factorization of $f \pmod{p}$ to the (unique) factorization of $f \pmod{p^n}$, for n big enough.
4. — In some way, choose candidates in the list of factors obtained mod p^n and recover from them possible factors in $\mathbb{Z}[X]$.

We are going to analyze succinctly each step of this factorization algorithm. We refer to [Mig89] and [Coh93] for an exhaustive description:

Let $f \in \mathbb{Z}[X]$ be a primitive polynomial of degree d and denote by $H(f)$ the maximum of the absolute values of all coefficients of f . Furthermore, suppose without loss of generality, that f is unitary.

1. — This step presents no problems. There are several ways of splitting f in a product of squarefree polynomials. For instance, the more naïve one is to divide f by $\gcd(f, f')$ and so on. The algorithm involves $O(d^2)$ operations in $\mathbb{Z}$ (See [Yun76] for other procedures).
2. — In 1970, Berlekamp ([Ber70]) presented an efficient algorithm for factoring a squarefree polynomial modulo a prime p . For a polynomial of degree d , its total cost is $O(p d^3)$, polynomial in p and d . Berlekamp’s algorithm relies strongly both on the richness of the Chinese Remainder Theorem and on the structure (and unicity) of finite fields.
3. — A useful bound due to Mignotte is that if $g \mid f$ in $\mathbb{Z}[X]$, then $H(g) \leq B := 2^d \sqrt{d+1} H(f)$. So, applying the observation which motivates the procedure, we would like to get factorizations of $f \pmod{2B}$.

Now, Newton–Hensel lifting in its simplest form states essentially that a factorization mod p leads to a unique factorization mod p^n :

Theorem 1 *Let p be a prime, and let $f, g_0, h_0 \in \mathbb{Z}[X]$ be unitary polynomials such that*

- $f \equiv g_0 h_0 \pmod{p}$
- g_0 and h_0 are coprime mod p .

Then, for any $n \in \mathbb{N}$ there exist unitary polynomials $g_n, h_n \in \mathbb{Z}[X]$ (unique mod p^n) such that:

- $f \equiv g_n h_n \pmod{p^n}$
- $g_n \equiv g_0 \pmod{p}$ and $h_n \equiv h_0 \pmod{p}$. ■

Therefore, the natural procedure is to: choose a prime p small enough such that $f \pmod{p}$ is still squarefree, set $f \equiv g_0 h_0 \pmod{p}$, and finally, choosing $n \geq \log_p 2B$, recover the factorization $f \equiv f_n g_n \pmod{p^n}$, and check if $f = g_n h_n$ in $\mathbb{Z}[X]$.

None of these steps is expensive, and up to this point the procedures are all still polynomial in d and $\log_2 H(f)$.

4. — The Berlekamp-Zassenhaus algorithm (1970) ([Zas69]) chooses all possible factors (non necessarily irreducible) mod p , lifts each of them mod p^n , and checks if the obtained factor mod p^n is actually a factor in $\mathbb{Z}[X]$. If f is reducible in $\mathbb{Z}[X]$, eventually, due to the uniqueness of the lifting, one obtains a factor by this procedure. The problem is that it may happen that one has to check too many factors to finally conclude that f is irreducible!

In 1982, Lenstra, Lenstra and Lovasz ([LLL82]) obtained the first polynomial algorithm to factor primitive polynomials in $\mathbb{Z}[X]$:

Theorem 2 *Let $f \in \mathbb{Z}[X]$ be a primitive polynomial of degree d and size of the coefficients bounded by H . Then, there is an algorithm which factors f in $\mathbb{Z}[X]$ and performs $O(d^{12} + d^9 (\log_2 H)^3)$ bit operations.* ■

The LLL-algorithm begins like the other ones, but instead of checking all possible factorizations, it finds the irreducible factor in $g \in \mathbb{Z}[X]$ which, when considered mod p , is divisible by a fixed irreducible factor $g_0 \pmod{p}$. The polynomials in $\mathbb{Z}[X]$ of bounded degree and divisible by $g_0 \pmod{p}$ define a lattice and the authors apply their basis reduction algorithm to find a small enough vector in the lattice, which allows to determine g .

2 Zero-dimensional Radical Ideals

Suppose $I = (f_1, \dots, f_s)$ defines a zero-dimensional radical ideal in $\mathbb{Q}[X_1, \dots, X_n]$, i.e.

$$V(I) := \{x \in \mathbb{C}^n : f_1(x) = 0, \dots, f_s(x) = 0\}$$

is non-empty and finite, and $I = \sqrt{I} := \{f \in \mathbb{Q}[X_1, \dots, X_n] : \exists N \in \mathbb{N} \text{ s.t. } f^N \in I\}$ (which implies by Hilbert's Nullstellensatz that $I = I(V(I)) = \{f \in \mathbb{Q}[X_1, \dots, X_n] : f(x) = 0 \forall x \in V(I)\}$).

Under these conditions, we have the following *Shape Lemma*:

Lemma 3 Let $\pi : \mathbb{C}^n \longrightarrow \mathbb{C}$ denote the projection to the first coordinate $\pi(x_1, \dots, x_n) = x_1$. If it is true that for all $x, y \in V(I)$, $x \neq y \Rightarrow \pi(x) \neq \pi(y)$, then there exist $g_1, g_2, \dots, g_n \in \mathbb{Q}[X_1]$ with $g_1 \neq 0$ s.t.

$$I = (g_1(X_1), X_2 - g_2(X_1), \dots, X_n - g_n(X_1)).$$

Moreover, if $\#V(I) = D$, then $\deg(g_1) = D$ and $\deg(g_i) < D$ ($2 \leq i \leq n$). ■

The Shape Lemma gives a useful presentation of the radical ideal I , since it allows us to immediately solve the system, solving first the univariate polynomial g_1 to obtain the distinct values of x_1 and substituting then its roots in $g_2, \dots, g_n$ in order to obtain the values of $x_2, \dots, x_n$.

Let us observe that in our case, when π separates all roots of the system, the basis of I given by the Shape Lemma coincides exactly with the reduced Gröbner Basis of I obtained using any total lexicographic ordering with $X_1 < X_i$ ($2 \leq i \leq n$).

But sometimes, there are no coordinates $X_1, \dots, X_n$ verifying that the projection in that direction separates the distinct roots of the original system.

Example 4 Let $I := (X(X-1), Y(Y+1))$, Then $V(I) = \{(0,0), (1,0), (0,-1), (1,-1)\}$. In order to obtain a Shape Lemma for I , we need to find a direction of projection separating the four roots. Let us take $U := X + 2Y$: then $U(0,0) = 0, U(1,0) = 1, U(0,-1) = -2, U(1,-1) = -1$. Applying the change of coordinates $X = U - 2Y, Y = Y$, $I = ((U - 2Y)(U - 2Y - 1), Y(Y + 1))$ with roots $(0,0), (1,0), (-1,-1), (-2,-1)$, we obtain:

$$I = (U(U-1)(U+1)(U+2), Y - (-\frac{1}{3}U^3 - \frac{1}{2}U^2 + \frac{5}{6}U)).$$

Moreover, if we want to express X in terms of U , we can write, keeping in mind that $U = X + 2Y$:

$$I = (U(U-1)(U+1)(U+2), X - (-\frac{2}{3}U^3 + 3U^2 - \frac{10}{3}U), Y - (-\frac{1}{3}U^3 - \frac{1}{2}U^2 + \frac{5}{3}U)).$$

Or, more formally :

$$V(I) = \{ (x, y) \in \mathbb{C}^2 : x = -\frac{2}{3}u^3 + 3u^2 - \frac{10}{3}u, y = -\frac{1}{3}u^3 - \frac{1}{2}u^2 + \frac{5}{6}u / u(u-1)(u+1)(u+2) = 0 \}.$$

This leads us to the following general version of the *Shape Lemma* :

Lemma 5 Let U be a linear form in $\mathbb{Q}[X_1, \dots, X_n]$ s.t. for all $x, y \in V(I)$, $x \neq y \Rightarrow U(x) \neq U(y)$, then there exist $q, v_1, v_2, \dots, v_n \in \mathbb{Q}[U]$ with $q \neq 0$ s.t.

$$I = (q(U), X_1 - v_1(U), \dots, X_n - v_n(U)).$$

Or, equivalently

$$V(I) = \{ (x_1, \dots, x_n) \in \mathbb{C}^n : x_1 = v_1(u), \dots, x_n = v_n(u) \text{ for } q(u) = 0 \}.$$

Moreover, if $\#V(I) = D$, then $\deg(q) = D$ and $\deg(v_i) < D$ ($1 \leq i \leq n$). ■

Subject to these conditions, observe that :

$$\mathbb{Q}[V] \simeq \mathbb{Q}[X_1, \dots, X_n]/I \simeq \mathbb{Q}[U]/(q(U))$$

as $\mathbb{Q}$ -algebras. We call U (or more precisely its residue class in the algebra) a *primitive element* for the finite-dimensional algebra $\mathbb{Q}[V]$ of rank $\#V(I)$.

Moreover, the advantage of having the description of the algebra by means of a primitive element U is that one knows exactly the structure of the algebra, for example the product in the algebra by any polynomial $g \in \mathbb{Q}[X_1, \dots, X_n]$:

First, observe that the matrix of the multiplication by U in the basis $\{1, U, \dots, U^{\#V(I)-1}\}$ is the companion matrix C_q of $q(U)$. Hence, the matrix of the multiplication by $X_i = v_i(U)$ is $v_i(C_q)$ and finally, as all these matrices commute, the matrix of the product by any g is $g(v_1(C_q), \dots, v_n(C_q))$.

For example, the question “ $g \in (f_1, \dots, f_s)$?” reduces to checking if the matrix is the zero matrix or not. And all algebraic questions are translated to linear algebra questions.

The question is now: How to compute a primitive element U of $\mathbb{Q}[V]$ (i.e. a linear form which separates the elements of $V(I)$) and the polynomials $q, v_1, \dots, v_n$ without knowing explicitly $V(I)$? Clearly, almost all linear forms will be a primitive element. But in order to choose one, there is the following effective result (see [GH91] and [KP96]):

Theorem 6 *Let $I := (f_1, \dots, f_n) \in \mathbb{Q}[X_1, \dots, X_n]$ be a complete intersection radical ideal, with $\deg f_i \leq d$ ($1 \leq d \leq n$). Then there exists a polynomial $g \in \mathbb{Q}[Y_1, \dots, Y_n]$ of degree $d^{O(n)}$ s.t. for all $y \in \mathbb{Q}^n$ with $g(y) \neq 0$, $U := y_1 X_1 + \dots + y_n X_n$ is a primitive element of the algebra $\mathbb{Q}[V]$. The polynomial g is computable from the input performing $d^{O(n)}$ operations.* ■

Once the primitive element U is fixed, there are several ways of computing $q, v_1, \dots, v_n$, for instance applying Gröbner Bases computations for zero-dimensional ideals (which is less expensive than in general), or, if we wish to avoid Gröbner Bases techniques, a faster method is described in [GH91] and [KP96].

3 Parametric systems

The idea of using a symbolic adaptation of Newton–Hensel iteration for lifting fibers in this context was introduced in [GHM⁺98], and developed in [GHH⁺97], [Mor97], [Mat97] and [HKP⁺98].

Let $T := (T_1, \dots, T_m)$ be a set of *parameters* and $X := (X_1, \dots, X_n)$ be a set of *variables*. Assume $F_1, \dots, F_n \in \mathbb{Q}[T, X]$ are in the best possible conditions, i.e. that they span a zero-dimensional radical ideal in $\mathbb{Q}(T)[X]$.

Finally set $\mathbf{K}$ for an algebraic closure of $\mathbb{Q}(T)$ and

$$V_T(I) := \{x \in \mathbf{K}^n : F_1(T, x) = 0, \dots, F_n(T, x) = 0\}.$$

In these conditions, the Shape Lemma states:

Lemma 7 *Let U be a linear form in $\mathbb{Q}[X_1, \dots, X_n]$ s.t. for all $x, y \in V_T(I)$, $x \neq y \Rightarrow U(x) \neq U(y)$, then there exist $\rho \in \mathbb{Q}[T]$ and $Q, V_1, \dots, V_n \in \mathbb{Q}[T, U]$ with $\rho, Q \neq 0$ s.t.*

$$V_T(I) = \{(x_1, \dots, x_n) \in \mathbf{K}^n : \rho(T) x_1 = V_1(T, u), \dots, \rho(T) x_n = V_n(T, u) \text{ for } Q(T, u) = 0\}.$$

Moreover, if $\#V_T(I) = D$, then $\deg_U(Q) = D$ and $\deg_U(V_i) < D$ ($1 \leq i \leq n$). ■

Thus, for any value of $t := (t_1, \dots, t_m) \in \mathbb{C}^m$ such that $Q(t, U)$ is a separable polynomial of degree D and $\rho(t) \neq 0$, we deduce that :

$$\begin{aligned} V_t(I) &:= \{x \in \mathbb{C}^n : F_1(t, x) = 0, \dots, F_n(t, x) = 0\} \\ &= \{(x_1, \dots, x_n) \in \mathbb{C}^n : x_1 = \frac{V_1(t, u)}{\rho(t)}, \dots, x_n = \frac{V_n(t, u)}{\rho(t)} \text{ for } Q(t, u) = 0\}. \end{aligned}$$

Conversely, assume that $t \in \mathbb{Q}^m$ is given such that $\#V_t(I) = \#V_T(I)$, and that we know a primitive element $U \in \mathbb{Q}[X]$ and the polynomials $q, v_1, \dots, v_n \in \mathbb{Q}[U]$ of the Shape Lemma for $V_t(I)$. How can we recover a Shape Lemma for V_T ?

Clearly, U will also be a primitive element for $\mathbb{Q}(T)[X]/(F_1, \dots, F_n)$.

For sake of simplicity, we will assume in what follows that $t = 0$. By the Implicit Function Theorem:

Lemma 8 *For any $x \in V_0$, there exists a unique n -tuple $R^x := (R_1^x, \dots, R_n^x) \in \mathbb{C}[[T]]^n$ of formal power series such that:*

- $F_1(T, R^x(T)) = 0, \dots, F_n(T, R^x(T)) = 0$
- $R^x(0) = x$.

In the construction of the series R^x we use the wellknown Newton–Hensel operator. If $F := (F_1, \dots, F_n)$ and $D_F(T, X)$ denotes the Jacobian matrix of $F_1, \dots, F_n$ w.r. to X , then :

$$N_F(T, X)^t := X^t - D_F(T, X)^{-1} F(T, X)^t$$

(where ^{t} denotes tranposition.)

Now define the k –fold iterate application of Newton–Hensel operator

$$(R^x)^{(k)}(T) := N_F^k(T, x) \text{ for } k \geq 0.$$

Thus, the conditions on $F_1, \dots, F_n$ imply that the series are well defined and converge to a power series n -tuple R^x . ■

Now, observe that if $V_0(I) := \{x^{(1)}, \dots, x^{(D)}\}$, then $V_T(I) = \{R^{x^{(1)}}, \dots, R^{x^{(D)}}\}$. Moreover, as $U(R^{x^{(j)}})$, $1 \leq j \leq D$, are the different zeroes of the minimal polynomial Q of U in the algebra $\mathbb{Q}(T)[X]/(F_1, \dots, F_n)$, Q is the characteristic polynomial of the matrix

$$\begin{pmatrix} U(R^{x^{(1)}}) & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & U(R^{x^{(D)}}) \end{pmatrix}$$

We wish to compute Q from $q, v_1, \dots, v_n$, without knowing the matrix of multiplication by U in any basis of $\mathbb{Q}(T)[X]/(F_1, \dots, F_n)$.

For this purpose we will compute first an approximation of this matrix of multiplication, based on the symbolic adaptation of Newton–Hensel algorithm in $\mathbb{Q}[[T]]$, and then truncate the characteristic polynomial of this matrix at $\deg_T Q$.

As the companion matrix $M \in \mathbb{Q}^{D \times D}$ of q is similar to the diagonal matrix with diagonal entries $U(x^{(1)}), \dots, U(x^{(D)})$, we will perform on M the Newton–Hensel operator N_F .

The k -fold iterate of the operator N_F can be computed by means of polynomials $g_1^k, \dots, g_n^k, h^k \in \mathbb{Q}[T, X]$ s.t.

$$N_F^k(T, X) = \left(\frac{g_1^k}{h^k}, \dots, \frac{g_n^k}{h^k} \right)$$

(See [GHH⁺97, Lemma 30]).

Lemma 9 *Define the matrix*

$$N := U \left(\frac{g_1^k}{h^k}(T, v_1(M), \dots, v_n(M)), \dots, \frac{g_n^k}{h^k}(T, v_1(M), \dots, v_n(M)) \right)$$

Then, the characteristic polynomial of N approximates Q with precision 2^k in $\mathbb{Q}[[T]]$. ■

Finally, a technique due to Strassen allows us to avoid divisions and to truncate the series at the required degree.

In order to define the rational functions $V_i \in \mathbb{Q}(T)[U]$ which define the coordinates X_i in terms of $U := \lambda_1 X_1 + \dots + \lambda_n X_n$ in the Shape Lemma for V_T , we compute in an analogous way the minimal polynomials of X_i and $U - \lambda_i X_i$ and then apply [KP96, Prop.27].

References

- [Ber70] E. Berlekamp: *Factoring polynomials over large finite fields*. Math. Comp. 24 (1970) 713–735.
- [Buch65] B. Buchberger: *Ein Algorithmus zum Auffinden der Basiselemente des Restklassenringes nach einem nulldimensionalen Polynomideal*. PhD thesis, Univ. Innsbruck, Austria (1965).
- [Buch 85] B. Buchberger: *Gröbner Bases: an algorithmic method in polynomial ideal theory*. Recent trends in multidimensional system theory. Reidel (1985) 184–232.
- [Coh93] H. Cohen: *A course in computational algebraic number theory*. Graduate Texts in Math. 138, Springer (1993).
- [GHH⁺97] M. Giusti, K. Hägele, J. Heintz, J.E. Morais, J.L. Montaña and L.M. Pardo: *Lower bounds for diophantine approximation*. J.P.A.A. 117-118 (1997) 277–317.
- [GH91] M. Giusti and J. Heintz: *La détermination des points isolés et de la dimension d’une variété algébrique peut se faire en temps polynomial*. Progress in Math. 94, Birkhäuser (1991) 169–193.
- [GHM⁺98] M. Giusti, J. Heintz, J.E. Morais, J. Morgenstern and L.M. Pardo: *Straight-line programs in geometric elimination theory*. J.P.A.A. 124 (1998) 101–146.
- [HKP⁺98] J. Heintz, T. Krick, S. Puddu, J. Sabia and A. Waissbein: *Deformation techniques for efficient polynomial equation solving*. Preprint Univ. Buenos Aires (1998).
- [KP96] T. Krick and L.M. Pardo: *A computational method for diophantine approximation*. Progress in Math. 143, Birkhäuser (1996) 193–253.
- [LLL82] A.K. Lenstra, H.W. Lenstra and L. Lovász: *Factoring polynomials with rational coefficients*. Math. Ann, 261 (1982) 515–534.

- [Mat97] G. Matera: *Sobre la complejidad en espacio y tiempo de la eliminación geométrica*. PhD thesis, Univ. Buenos Aires (1997).
- [Mor97] J.E. Morais: *Resolución eficaz de sistemas de ecuaciones polinomiales*. PhD thesis, Univ. Cantabria (1997).
- [Mig89] M. Mignotte: *Mathématiques pour le Calcul Formel*. Presses Universitaires de France (1989).
- [Yun76] D.Y.Y. Yun: *On square-free decomposition algorithms*. Proc. of the 1976 AMS Symp. on Symb. and Alg. Comp., Yorktown (1976).
- [Zas69] H. Zassenhaus: *On Hensel factorization I*. J.Number Theory 1 (1969) 291–311.