RESOLUTION OF POLYNOMIAL SYSTEMS

D. LAZARD
LIP6, Université Paris VI, 75252 Paris Cedex 05, France
E-mail: Daniel.Lazard@lip6.fr

In this paper, the state of the art of solving polynomial systems by algebraic methods is sketched, and the main directions where future work is needed are indicated. Emphasis is given on input-output specification rather than on technical algorithmic aspects.

1 Introduction

For more than 20 years, much work has been devoted to algorithms for solving polynomial systems.

Most of the work in the field is devoted to four approaches: Gröbner bases, triangular systems, and, in the real context, cylindrical algebraic decomposition and "asymptotically fast algorithms". But few efficient implementations are available, and they only concern the first three approaches.

In this paper, we try to give a description of the state of the art on *solving polynomial systems* by focusing on practical efficiency. In order to keep in a small number of pages, we do not describe nor cite most of the relevant papers, limiting ourselves to trends and citing only the works well illustrating these trends.

Thus, we had to make many subjective choices, which have as a consequence that many important work or papers are not mentioned. We apologize for this.

One of these trends is that, for a long time, most authors tried to improve algorithms and very few of them took care on the specification of the output, i.e. on the kind of output that would fulfill the needs of the users of a solver. It is very recent that very efficient algorithms and implementations are available, especially for Gröbner basis computation and real zero isolation, which set this question of specification of the output as an essential one.

Therefore, the organization in sections of this paper is done by the different kinds of input-output. Most of the time, we mention only the most efficient algorithms and implementations without any description nor citing previous work incorporated therein.

2 Systems of Equations — Dimension 0

A *system of equations* is a formula[a]

$$P_1 = 0 \text{ and } P_2 = 0 \text{ and } \cdots \text{ and } P_k = 0 \tag{1}$$

where the P_i's are multivariate polynomials with coefficients in some given field K. A *solution* is a set of values for the variables of the P_i's, which satisfy (1); these values are searched in some field extension L of K.

In this paper we only consider the case where L is algebraically closed or is the field of the reals[b].

A system is said to be of *dimension* 0 or *zero-dimensional* if it has only a finite set of solutions in some algebraically closed field. In this case, it is well known that all solutions are algebraic on K and lie in the algebraic closure of K. Thus they are essentially independent of L.

The zero-dimensional case is essentially the only one which is accessible to purely numeric methods. Moreover, numeric methods require that the number of equations is equal to the number of variables, and, even in this case, numerical instability strongly limits the size of numerically solvable systems.

There are many ways for algebraically representing the solutions of a zero-dimensional system. One of the most useful is to represent them as one or several regular triangular systems

$$\begin{aligned} &g_1(x_1) = 0, \\ &g_2(x_1, x_2) = 0, \\ &\quad\cdots \\ &g_n(x_1, \ldots, x_n) = 0, \end{aligned} \tag{2}$$

where the initial[c] of g_i has no common zero with $g_1, \ldots, g_{i-1}$.

This representation is usually convenient for any subsequent work on the solutions, except for numerical evaluation, which needs to solve univariate polynomial equations with approximate coefficients.

Therefore, a RUR (Rational Univariate Representation) is numerically more convenient. In fact, a RUR is a system

$$\begin{aligned} &h_0(x_0) = 0, \\ &x_1 = h_1(x_0)/h_0'(x_0), \\ &\quad\cdots \\ &x_n = h_n(x_0)/h_0'(x_0), \end{aligned} \tag{3}$$

[a] Usually, the "and" are omitted and replaced by commas.

[b] Looking for the solutions in K is usually much more difficult: The 10th problem of Hilbert essentially asserts that the existence of solutions is undecidable when $K = L$ is the field of rationals.

[c] The *initial* is the leading coefficient of g_i when viewed as univariate polynomial in x_i.

where the h_i's are univariate polynomials, x_0 is an auxiliary variable and h_0' is the derivative of h_0.

Actually, one standard way for a complete resolution of a polynomial system consists in the following schema.

1. Compute a lexicographical Gröbner basis, either directly or through a change of basis ordering[7]. This step checks zero-dimensionality.
2. Deduce from it a set of regular triangular systems using algorithm `Lex-triangular`[15].
3. For each of these triangular systems, compute a RUR[17].
4. For each RUR compute a numerical approximation of the solutions together with a bound on the error, either with the Uspensky algorithm implemented by Rouillier[18] in the real case or with Bini's implementation[3] in the complex case (the latter does not give a bound on the error, but it gives an output of arbitrary precision).

With the implementation of FGB and RS[6], this strategy commonly allows one to solve systems with more than thousand solutions.

The critical step lies frequently in the size of the Gröbner basis to be computed. It seems that algorithm F7[6] may avoid this problem by splitting the system during the computation of the Gröbner basis: The system Cyclic(9) has a Gröbner basis of 1.6 Gbytes whose computation needs 15 days and which is too big for subsequent computation, while F7 provides in a few hours a set of 113 Gröbner bases which allows a complete description of the solutions[5].

3 Triangular Sets

The most common alternative to Gröbner bases for algebraic resolution of polynomial systems consists in using triangular systems. They were introduced by Ritt under the name of characteristic sets and systematically used by Wen-tsün Wu and his followers for solving systems. However, two important facts were only recently well recognized and need to be described here: the essential role of regular triangular sets and the two different ways in which triangular sets may be used in solving systems.

For explaining this, we need to recall some technical notions.

We suppose that the variables of the polynomials are $x_1, \ldots, x_n$ sorted by increasing indices. The *main* variable of a polynomial is the highest one which effectively appears in it[d]. The *initial* of a polynomial is its coefficient of highest degree, when it is viewed as univariate polynomial in its main variable.

[d] Thus a constant does not have any main variable.

A *triangular set* is a sequence of polynomials with strictly increasing main variables. The *regular zeros* of a triangular set are its common zeros (in some algebraically closed field) which are not zeros of any initial.

To a triangular set T is associated an ideal $\mathrm{Sat}(T)$, the *saturated ideal* of T, which is the set of the polynomials, the product of which by some product of initials lies in the ideal generated by T.

Theorem 1 [2] *For a triangular set $T = (t_1, \ldots t_k)$ the following conditions are equivalent.*

1. *T is a characteristic set of $\mathrm{Sat}(T)$.*
2. *The ideal $\mathrm{Sat}(T)$ is the set of the polynomials which reduce to 0 by T with pseudo-division.*
3. *For any i the initial of t_i is not a zero-divisor in the ring $K[x_1, \ldots, x_n]/\mathrm{Sat}(t_1, \ldots, t_{i-1})$.*
4. *For any i, the iterated resultant r_0 defined as follows is not zero: $r_i = \mathrm{initial}(t_i)$; $r_j = \mathrm{resultant}(r_{j+1}, t_j, \mathrm{main\_var}(t_j))$ for $j = i-1, \ldots, 0$.*

A triangular set which satisfies these equivalent conditions is said to be *regular*. An ideal which is the saturated ideal of a regular triangular set is said to be *triangularizable*; it is always equi-dimensional of dimension $n - k$, where n is the number of variables and k the length of T. A prime ideal is always triangularizable, and the primes associated to a triangularizable ideal are simply obtained by factoring recursively each t_i in the field extensions defined by the factors of $t_1, \ldots, t_{i-1}$.

It should be remarked here that regular triangular sets are a good alternative to Gröbner basis for representing triangularizable and prime ideals on computers: The number of polynomials in a triangular set is always bounded by the number of variables, which is not the case for generating sets or Gröbner bases of prime ideals. Computing a Gröbner basis from a regular triangular set may be done by any Gröbner basis algorithm, and is usually not too difficult. The inverse transformation is very easy for prime ideals.

4 Solving by Prime Decomposition

For solving polynomial systems of positive dimension, there are two main approaches. We consider here the first one, leaving the second one to the next section.

A system of equations defines an ideal, and for solving the system (i.e. studying the possibly infinite set of solutions) one may decompose the radical of this ideal as an intersection of primes and then study separately the primes. This is equivalent to decomposing the set of solutions into irreducible components which are studied separately.

We do not precisely define here what we mean by an irreducible variety, because it strongly depends on the kind information about the solutions which is needed by the user of a solving program.

M. Kalkbrener[13] is the first to have provided an algorithm for computing a prime decomposition by mean of regular triangular sets, or more exactly for computing a decomposition in triangularizable ideal; in fact the prime decomposition may be computed at the end, using factorization in algebraic field extensions. This algorithm has been improved by Aubry[1] and seems to be the fastest existing algorithm for system solving by triangular sets.

A drawback of this algorithm is that, in some case, it computes a redundant decomposition which may only be cleaned by using Gröbner bases. Moreover, this algorithm extends trivially to differential algebra and in this context, no algorithm is known for removing redundancies, and one may conjecture that this is an undecidable problem (equivalent to the Ritt problem).

Recently, an experimental version of an algorithm called F7 has been developed by Faugère[6], which provides as output a decomposition in primes represented by Gröbner bases or regular triangular sets. This algorithm appears to be very efficient, but it is too early to determine which approach between F7 and Kalkbrener–Aubry is more efficient.

From an algebraic point of view, some information, like multiplicities, is lost by prime decomposition. Thus efficient algorithms for computing complete primary decompositions would be needed. Unfortunately, the algorithms which are known[10] are too inefficient for solving difficult problems. Thus a challenge for the next years would be to get efficient algorithms for primary decomposition.

5 Solving by Regular Zero Decomposition

There is another way of using triangular sets for solving polynomial systems of positive dimension: It consists in decomposing the sets of zeros into sets of regular zeros of triangular systems. It is the approach taken by Wen-tsün Wu and his followers[19].

The advantage of this approach is to allows non-redundant decompositions[14,16] without using external algorithms like Gröbner bases. On the other hand, some topological or geometric information may be difficult to extract without Gröbner bases, like the inclusion of one component in the closure of another.

It should be noted that this approach, as well as the one of preceding section, allows one to manage inequations of the form $Q \neq 0$. In both cases, the inequations of the input may be introduced only at the end of the computation, but it is usually better to use them during the computation for removing early empty components.

Despite the recent progresses on this approach, it is not yet clear in which cases it will be faster than using Gröbner bases. Moreover, no estimation is available of the complexity of the algorithms of decomposition by triangular sets, but one may conjecture a bound of $d^{O(n^2)}$ (see the paper[9] by Gallo and Mishra), while Gröbner bases may have output of size $d^{2^{O(n)}}$ (see the reference[12]).

It follows that competition is yet open between the triangular sets approach and the Gröbner basis one. Personally, I guess that the winner will be a mixture of both.

6 Real Solving in Positive Dimension

When working in the real field, one is usually concerned not only with equations and inequations but also with inequalities $Q > 0$ or $Q \geq 0$. In this context, one considers not only systems of equations and inequalities but more generally also formulae of first order logic, in which atomic formulae are equations, inequations and inequalities involving polynomials.

In this case, solving a system may ideally be split into three steps:

1. Eliminate quantifiers in order to get an equivalent quantifier-free formula, which may be written as a disjunction of systems of equations and inequalities.
2. Decompose the set of solutions of these systems into (algebraic, semi-algebraic or connected) components represented, for example, by some generalization of triangular systems.
3. Study the structure of these components and their inter-relation. For example describe the topology of the whole set of solutions.

In practice these steps are not completely distinct. For example, Collins' cylindrical algebraic decomposition (CAD) performs both first steps. Also, a projection of an irreducible variety may be viewed as a quantifier elimination as well as the study of the variety.

While many theoretical algorithms are known for performing these steps, the only one which is *in practice* efficient enough for being implemented is Collins' CAD[11]. Moreover many seemingly easy problems are outside the possibilities of this algorithm; among them some are solvable by hand.

Thus hard work is needed for finding better algorithms.

It should be remarked that the output of the above three steps is not well described. In fact, we need not only better algorithms but also specifications of input and output which together allow efficient computation and are useful in practice.

We end this paper by two examples for such restricted problems, which have been recently solved, and both need most of the recent progresses sketched in preceding sections.

The first is an efficient algorithm for providing at least one point for each connected component of the set of real solutions of a systems of equations (without inequations nor inequalities)[4].

The second is an algorithm for discussing the number of real solutions of a system of equations and inequations depending on at most two parameters[8]. It has been developed for and applied to a problem of celestial mechanics, which consists in determining the number of symmetric configurations of 4 unequal masses which remain homothetic to themselves (two symmetric masses are supposed to be equal and are normalized to the unit).

This problem may be described by the following system

$$\begin{array}{l} b^3B^2-1=0, \qquad d^3D^2-1=0, \qquad f^3F^2-1=0, \\ (d-b)^2-2(b-d)+f+1=0, \\ (B-1)m+(D-F)(b-d-1)=0, \\ (D-1)n+(B-F)(d-b-1)=0, \end{array} \tag{4}$$

where the parameters are the two masses m and n, and all variables and parameters should be positive.

This problem, which seems rather small, gives an idea of the complexity of problems of positive dimension: the domains where the number of solutions are constant (1, 3 or 5) are delimited by two branches of an algebraic curve defined by a bivariate polynomial, which has total degree 424 with coefficients of more than 200 decimal digits and needs more than 10 Mbytes to be stored. An important part of the several days of computation which were needed was devoted to the drawing of this curve and to the certification of this drawing: computation of the 24 cusps, of the 22 asymptotes (some of which are multiple), of the points with tangents parallel to the axes, and of the way in which all of these are connected.

Note that in this case, the output of the resolution is essentially this curve.

References

1. P. Aubry. *Ensembles triangulaires de polynômes et résolution des systèmes algébriques.* Thèse de l'Université Pierre et Marie Curie (Paris VI) (1999).

2. P. Aubry, D. Lazard and M. Moreno-Maza. *On the theories of triangular sets.* J. Symbolic Comput. **28**, 105–124 (1999).
3. D. Bini and G. Fiorentino. *MPSolve: Numerical computation of polynomial roots v. 2.0.* FRISCO report (1998). `http://fibonacci.dm.unipi.it/~bini/papers/mps2.html`
4. P. Aubry, F. Rouillier and M. Safey El Din. *Real solving for positive dimensional systems.* Submitted to J. Symbolic Comput. (2000).
5. J.C. Faugère. *How my computer finds all the solutions of Cyclic 9.* Research report LIP6-2000-007 (2000). `http://www.lip6.fr/reports/lip6.2000.007.html`
6. J.C. Faugère. `//http://calfor.lip6.fr/~jcf/Software/index.html`
7. J.C. Faugère, P. Gianni, D. Lazard and T. Mora. *Efficient computation of zero-dimensional Gröbner bases by change of ordering.* J. Symbolic Comput. **16**, 329–344 (1993).
8. J.C. Faugère and D. Lazard. In preparation.
9. G. Gallo and B. Mishra. *Efficient algorithms and bounds for Wu–Ritt characteristic sets.* Progress in Math. **94**, 119–142, Birkhäuser (1991).
10. P. Gianni, B. Trager and G. Zacharias. *Gröbner bases and primary decomposition of polynomial ideals.* J. Symbolic Comput. **6**, 149–167 (1988).
11. H. Hong. *Comparison of several decision algorithms for the existential theory of the reals.* Research report, RISC-Linz (1991).
12. D.T. Huynh. *A superexponential lower bound for Gröbner bases and Church Rosser commutative Thue systems.* Information and Control **68**, 196–206 (1986).
13. M. Kalkbrener. *A generalized Euclidean algorithm for computing triangular representations of algebraic varieties.* J. Symbolic Comput. **15**, 143–167 (1993).
14. D. Lazard. *A new method for solving algebraic systems of positive dimension.* Discrete Appl. Math. **33**, 147–160 (1991).
15. D. Lazard. *Solving zero–dimensional algebraic systems.* J. Symbolic Comput. **13**, 117–131 (1992).
16. M. Moreno-Maza. *On triangular decomposition of algebraic varieties. Proceedings of MEGA'2000* (to appear).
17. F. Rouillier. *Solving zero-dimensional systems through the rational univariate representation.* Appl. Algebra Engrg. Comm. Comput. **9**, 433–461 (1999).
18. F. Rouillier and P. Zimmermann. *Efficient isolation of a polynomial real roots.* Preprint, 12 pages, INRIA (2000).
19. D. Wang. *Elimination methods.* Springer-Verlag (2000).