

The computational complexity of the Chow form

Gabriela Jeronimo¹, Teresa Krick¹, Juan Sabia¹ and Martín Sombra^{1,2}

Abstract. We present a bounded probability algorithm for the computation of the Chow forms of the equidimensional components of an algebraic variety. In particular, this gives an alternative procedure for the effective equidimensional decomposition of the variety, since each equidimensional component is characterized by its Chow form.

The expected complexity of the algorithm is *polynomial* in the size and the geometric degree of the input equation system defining the variety. Hence it improves (or meets in some special cases) the complexity of all previous algorithms for computing Chow forms. In addition to this, we clarify the probability and uniformity aspects, which constitutes a further contribution of the paper.

The algorithm is based on elimination theory techniques, in line with the geometric resolution algorithm due to M. Giusti, J. Heintz, L.M. Pardo and their collaborators. In fact, ours can be considered as an extension of their algorithm for zero-dimensional systems to the case of positive-dimensional varieties.

The key element for dealing with positive-dimensional varieties is a new Poisson-type product formula. This formula allows us to compute the Chow form of an equidimensional variety from a suitable zero-dimensional fiber.

As an application, we obtain an algorithm to compute a subclass of sparse resultants, whose complexity is *polynomial* in the dimension and the volume of the input set of exponents. As another application, we derive an algorithm for the computation of the (unique) solution of a generic over-determined polynomial equation system.

Keywords. Chow form, equidimensional decomposition of algebraic varieties, symbolic Newton algorithm, sparse resultant, over-determined polynomial equation system.

MSC 2000. *Primary:* 14Q15, *Secondary:* 68W30.

Contents

Introduction	2
1 Preliminaries	6
1.1 The Chow form of a quasiprojective variety	7
1.2 Data and algorithm structure	8
1.3 Complexity of basic computations	14
1.4 Effective division procedures	16
2 The representation of the Chow form	21
2.1 Newton's algorithm	22
2.2 A product formula	25
2.3 The algorithm	29
3 The computation of the Chow form	31
3.1 Geometric resolutions	31
3.2 Intersection of a variety with a hypersurface	37
3.3 Separation of varieties	40
3.4 Equations in general position	42
3.5 Proof of Theorem 1	46

¹Partially supported by UBACyT EX-X198 and CONICET PIP 02461/01-03 (Argentina).

²Supported by a Marie Curie Post-doctoral fellowship of the European Community Program *Improving Human Research Potential and the Socio-economic Knowledge Base*, contract n° HPMFCT-2000-00709.

4 Applications	52
4.1 Computation of resultants	52
4.1.1 The classical d -resultant	52
4.1.2 Sparse resultants	53
4.2 Generic over-determined systems	55
References	57

Introduction

The Chow form of an equidimensional quasiprojective variety is one of the basic objects of algebraic geometry and plays a central role in elimination theory, from both the theoretical and practical points of view.

Let $V \subset \mathbb{P}^n$ be an equidimensional quasiprojective variety of dimension r and degree D defined over \mathbb{Q} . Its *Chow form* \mathcal{F}_V is a polynomial with rational coefficients —unique up to a scalar factor— which characterizes the set of over-determined linear systems over the projective closure \bar{V} . More precisely, let U_0, \dots, U_r denote $r+1$ groups of $n+1$ variables each, and set $L_i := U_{i0}x_0 + \dots + U_{in}x_n$ for the linear form associated to the group U_i for $0 \leq i \leq r$. Then $\mathcal{F}_V \in \mathbb{Q}[U_0, \dots, U_r]$ is the unique —up to a scalar factor— squarefree polynomial such that

$$\mathcal{F}_V(u_0, \dots, u_r) = 0 \iff \bar{V} \cap \{L_0(u_0, x) = 0, \dots, L_r(u_r, x) = 0\} \neq \emptyset$$

for $u_0, \dots, u_r \in \mathbb{C}^{n+1}$. This is a multihomogeneous polynomial of degree D in each group of variables U_i . Thus we can directly read the dimension and the degree of V from \mathcal{F}_V . In case V is closed in \mathbb{P}^n , its Chow form completely characterizes it, and it is possible to derive a complete set of equations for V from \mathcal{F}_V .

The main result of this work is that the computation of the Chow forms of all the equidimensional components of a quasiprojective variety defined by means of a given set of polynomials has a *polynomial* complexity in terms of the number of variables, the degree, and also the *length* and the *geometric degree* of the input system. The result is based on a new Poisson-type product formula for computing Chow forms of equidimensional varieties from zero-dimensional fibers.

The complexity of all known general algorithms in algebraic geometry is (at least) exponential in the worst case when the considered input parameters are just the number of variables and the degree of the input system, and there is strong evidence that this exponential behavior is unavoidable (see [34] and [12]). However, it has been observed that there are many particular instances which are much more tractable than the general case. This has motivated the introduction of parameters associated to the input system that identify these particular cases, and the design of algorithms whose performance is correlated to these parameters.

In this spirit, the *straight-line program* encoding of polynomials (slp for short) was introduced in the polynomial equation solving framework as an alternative data structure (see e.g. [23], [28], [27]) and its *length* is now considered to be a meaningful parameter measuring the input (see Section 1.2 below for the definition of these notions and [52], [32], [8] for a broader background).

Soon afterwards, the notion of *geometric degree* of the input polynomial system appeared naturally as another useful parameter classifying tractable problems. For a system of homogeneous polynomials $f_1, \dots, f_s, g \in \mathbb{Q}[x_0, \dots, x_n]$, the geometric degree $\delta(f_1, \dots, f_s; g)$ is defined as the maximum degree of the quasiprojective varieties $V(q_1, \dots, q_i) \setminus V(g)$, $1 \leq i \leq n+1$, successively defined by generic linear combinations q_1, \dots, q_{n+1} of the input polynomials (multiplied by suitable monomials in order to match their degrees); we refer to Section 3.4 below for the details. This is always bounded by the Bézout number d^n , where $d := \max_i \deg f_i$; however, there are many significant situations in which the geometric degree is much smaller than this upper bound (see [42, Section 4] for a particular example or [41, Proposition 2.12] for an analysis of the sparse case).

In [26] and [22], J. Heintz, M. Giusti, L.M. Pardo and their collaborators succeeded in classifying the tractability of polynomial equation solving *in the zero-dimensional case* in terms of the length and the geometric degree of the input system. They presented an algorithm, the *geometric resolution algorithm*, whose complexity is *polynomial* in the number of variables, the degree, the length and the geometric degree of the input system. Their algorithm (its structure and the tools they applied) represents a milestone in symbolic resolution of polynomial equation systems, and a lot of work has been done afterwards to improve it, make it implementable and extend it to other situations (see e.g. [29], [35]). Our main theorem can be seen as a general extension of their result to arbitrary varieties:

Theorem 1 *Let $f_1, \dots, f_s, g \in \mathbb{Q}[x_0, \dots, x_n]$ be homogeneous polynomials of degree bounded by d encoded by straight-line programs of length bounded by L . Set $V := V(f_1, \dots, f_s) \setminus V(g) \subset \mathbb{P}^n$ for the quasiprojective variety $\{f_1 = 0, \dots, f_s = 0, g \neq 0\}$ and let $V = V_0 \cup \dots \cup V_n$ be its minimal equidimensional decomposition. Set $\delta := \delta(f_1, \dots, f_s; g)$ for the geometric degree of the input polynomial system.*

Then there is a bounded probability algorithm which computes (slp's for) the Chow forms $\mathcal{F}_{V_0}, \dots, \mathcal{F}_{V_n}$ within (expected) complexity $s(n d \delta)^{\mathcal{O}(1)} L$. Its worst-case complexity is $s(n d^n)^{\mathcal{O}(1)} L$.

Let us make precise the formal setting for our computations. The basis of our algorithms is the model of bounded error probability *Blum-Shub-Smale machine* (BSS machine for short) over \mathbb{Q} : our algorithms are probabilistic BSS machines that manipulate slp's. A probabilistic BSS machine is the algebraic analogue of the classical notion of probabilistic Turing machine, in which the bit operations are replaced by the arithmetic operations $\{+, -, \cdot, /\}$ of \mathbb{Q} . It enables us to implement *uniform* procedures while “programming” using the basic operations of \mathbb{Q} . This model is well suited to control the algebraic complexity —that is the number of arithmetic operations— performed by the algorithm.

By *bounded error probability* we mean that the error probability of the machine is *uniformly* bounded from above by $1/4$. For us, the natural notion of complexity in this framework is then the expectation of the complexity seen as a random variable, and *not* its worst case.

The choice of the constant $1/4$ as error probability is not restrictive: for any given $N \in \mathbb{N}$ we can easily modify our machine (by running it $\mathcal{O}(\log N)$ times) so that the final error probability is bounded by $1/N$ (see Proposition 1.5 and Corollary 1.6 below).

We refer to Section 1.2 for a detailed description and discussion of the data structure and computational model. We consider that the error probability analysis developed here is another contribution of the paper, since the previous papers on the subject are often imprecise in this respect.

We note that in our situation, the dense encoding of polynomials does not admit a polynomial time algorithm: if V is an equidimensional variety of dimension r defined as the zero set of a family of polynomials of degrees bounded by d , its Chow form \mathcal{F}_V is a polynomial of degree $(r+1) \deg V$ in $(r+1)(n+1)$ variables. So the dense encoding of \mathcal{F}_V (i.e. the vector of its coefficients) has

$$\binom{(r+1)(n+1) + (r+1) \deg V}{(r+1)(n+1)} \geq \frac{(\deg V)^{(r+1)(n+1)}}{((r+1)(n+1))!}$$

entries, and hence it is not polynomial in $\deg V$ (which in the worst case equals d^{n-r}). In fact, Corollary 2.11 below shows that in the above situation the slp encoding of \mathcal{F}_V has length $L(\mathcal{F}_V) = (n d \deg V)^{\mathcal{O}(1)} L$.

For the problem of computing Chow forms, our algorithm fundamentally improves the complexity of the algorithms in [39], [9] and [24] (which use dense encoding of polynomials) and in [48] (which uses the slp representation for the output). The only previous algorithm for the task whose complexity is in some cases comparable to ours is the one by G. Jeronimo, S. Puddu and J. Sabia [36], which computes (an slp representation of) the Chow form of the component of maximal dimension of an algebraic variety within complexity $(s d^n)^{\mathcal{O}(1)}$. Here, we not only compute the Chow forms of *all* of

the equidimensional components, but we also replace the Bézout number d^n by $d\delta$, where δ denotes the geometric degree.

Our algorithm also provides an effective geometric *equidimensional decomposition*, since each equidimensional component is characterized by its Chow form. Moreover, we can easily derive equations or a geometric resolution of each equidimensional component from its Chow form (see Corollary 3.3 below).

The complexity of our algorithm meets or improves those of most of the previous equidimensional decomposition algorithms: [13], [24] and [19] (which use dense encoding of polynomials) and [37] and [44] (which use slp encoding). Its structure is similar to the one due to G. Jeronimo and J. Sabia [37]—which computes equations for each equidimensional component—and to G. Lecerf’s algorithm in [44]—which computes a geometric resolution of each equidimensional component. In fact, ours can be seen as a unification of these algorithms. Besides, we improve the complexity of [37] by replacing the quantity d^n by the geometric degree δ and with respect to [44], we improve the error probability estimate.

It is worth mentioning here the most recent equidimensional decomposition algorithm by G. Lecerf ([46], see also [45]), which improves the previous ones in several aspects. For instance, unlike his, our algorithm and the previous ones require a preprocessing of the input polynomials by taking generic linear combinations. This preprocessing may spoil the (potential) good behavior of the input and it also prevents the algorithm from being incremental. However, his algorithm still lacks an adequate error probability analysis.

On the other hand, it is by no means obvious how to obtain the Chow form from a geometric resolution within an admissible complexity. The difficulty lies in the fact that the involved morphism is not finite but just *dominant* (see Remark 2.7). In this paper, we exhibit a deterministic algorithm based on a new Poisson-type formula, which performs this task within complexity $(snd \deg V)^{\mathcal{O}(1)}L$ (Main Lemma 2.3). This is the key element in our algorithm and, probably, the main technical contribution of the present work. It might be interesting to remark that using Main Lemma 2.3 as a post-treatment to the output of the algorithm in [44] would have led to an algorithm for computing Chow forms with similar complexity results but a worse error probability estimate.

As a first application of our algorithm, we compute a particular class of sparse resultants.

The classical resultant $\text{Res}_{n,d}$ of a system of $n + 1$ generic homogeneous polynomials in $n + 1$ variables is a polynomial in the indeterminate coefficients of the polynomials that characterizes for which coefficients the system has a non-trivial solution. We show that $\text{Res}_{n,d}$ can be deterministically computed within complexity $(nd^n)^{\mathcal{O}(1)}$ (Corollary 4.1).

The sparse resultant $\text{Res}_{\mathcal{A}}$ —the basic object of sparse elimination theory—has been extensively applied as a tool for the resolution of polynomial equation systems (see e.g. [54], [49], [20]). Several effective procedures were proposed to compute it (see e.g. [54], [10], [11]). Recently, C. D’Andrea has obtained an explicit determinantal formula which extends Macaulay’s formula to the sparse case ([17]).

From the algorithmic point of view, the basic point of sparse elimination theory is that computations should be substantially faster when the input polynomials are sparse (in the sense that their Newton polytopes are restricted). Basically, the parameters which control the sparsity are the number of variables n and the normalized volume $\text{Vol}(\mathcal{A})$ of the convex hull of the set \mathcal{A} of exponents. None of the previous algorithms computing sparse resultants is completely satisfactory, as their predicted complexity is exponential in all or some of these parameters (see [11, Corollary 12.8]).

The precise definition of $\text{Res}_{\mathcal{A}}$ is as follows:

Let $\mathcal{A} = \{\alpha_0, \dots, \alpha_N\} \subset \mathbb{Z}^n$ be a finite set of integer vectors. We assume here that \mathbb{Z}^n is generated by the differences of elements in \mathcal{A} . For $0 \leq i \leq n$, let U_i be a group of variables indexed by the elements of \mathcal{A} , and set

$$F_i := \sum_{\alpha \in \mathcal{A}} U_{i\alpha} x^\alpha \in \mathbb{Q}[U_i][x_1^{\pm 1}, \dots, x_n^{\pm 1}]$$

for the generic Laurent polynomial with support equal to \mathcal{A} . Let $W_{\mathcal{A}} \subset (\mathbb{P}^N)^{n+1} \times (\mathbb{C}^*)^n$ be the incidence variety of F_0, \dots, F_n in $(\mathbb{C}^*)^n$, that is

$$W_{\mathcal{A}} := \{(\nu_0, \dots, \nu_n; \xi); F_i(\nu_i, \xi) = 0, 0 \leq i \leq n\},$$

and let $\pi : (\mathbb{P}^N)^{n+1} \times (\mathbb{C}^*)^n \rightarrow (\mathbb{P}^N)^{n+1}$ be the canonical projection.

The sparse \mathcal{A} -resultant $\text{Res}_{\mathcal{A}}$ is defined as the unique —up to a sign— irreducible polynomial in $\mathbb{Z}[U_0, \dots, U_n]$ which defines $\overline{\pi(W_{\mathcal{A}})}$, which is an irreducible variety of codimension 1 (see [21, Chapter 8, Proposition-Definition 1.1]). It coincides with the Chow form of the toric variety associated to the input set \mathcal{A} (see Section 1.1 below).

We show that the computation of $\text{Res}_{\mathcal{A}}$ in case $\mathcal{A} \subset (\mathbb{N}_0)^n$ and \mathcal{A} contains $0, e_1, \dots, e_n$ —the vertices of the standard simplex of \mathbb{R}^n — is an instance of our main algorithm (see Subsection 4.1.2). We thus obtain:

Corollary 2

Let $\mathcal{A} \subset (\mathbb{N}_0)^n$ be a finite set which contains $\{0, e_1, \dots, e_n\}$. Then there is a bounded probability algorithm which computes (an slp for) the \mathcal{A} -resultant $\text{Res}_{\mathcal{A}}$ within (expected) complexity $(n \text{Vol}(\mathcal{A}))^{\mathcal{O}(1)}$. Its worst-case complexity is $(nd^n)^{\mathcal{O}(1)}$, where $d := \max\{|\alpha|; \alpha \in \mathcal{A}\}$.

Hence our result represents a significant improvement in the theoretical complexity of computing the \mathcal{A} -resultant as we show it is *polynomial* in n and $\text{Vol}(\mathcal{A})$. We remark that to achieve this result, we had to abandon all matrix formulations. In fact, this polynomial behavior of the complexity is out of reach of the known matrix formulations, as in all of them the involved matrices have an exponential size.

It would be desirable to extend this algorithm in order to compute a general *mixed* resultant. This point will be the subject of our future research.

As another application, we compute the unique solution of a generic over-determined system over an equidimensional variety $V \subset \mathbb{P}^n$:

Set $r := \dim V$. Let $u = (u_0, \dots, u_r) \in (\mathbb{C}^{n+1})^{r+1}$ and set $\ell_i := L_i(u_i, x) = u_{i0}x_0 + \dots + u_{in}x_n$ for $0 \leq i \leq r$. The set of coefficients u such that the linear forms ℓ_0, \dots, ℓ_r have at least a common root in V contains a non-empty open subset for which there is exactly *one* common root. It turns out that the coordinates of the unique solution $\xi(u)$ are rational functions of u , and can be easily computed using the Chow form \mathcal{F}_V . We can successfully apply our algorithm to this situation (see Section 4.2 for the details).

Finally, let us mention a very recent new application of our algorithm: C. Blanco, G. Jeronimo and P. Solernó apply it for the computation of a set of generators of the ideal of a smooth equidimensional affine variety [4].

Now we briefly sketch our algorithm. The main structure follows that of the geometric resolution algorithm in [26, 22].

In a first step, we prepare the input data: We take $n + 1$ random linear combinations of the input polynomials so that —with high probability— these new polynomials define the same variety V and behave properly with respect to the dimensions and radicality of certain ideals they successively define. We also take a random change of variables to ensure good conditions for the considered projections.

After this preparatory step, we compute recursively the Chow forms of the components of a non-minimal equidimensional decomposition of V . For $0 \leq r \leq n - 1$, the algorithm deals with an equidimensional subvariety W'_r of the variety defined by the first $n - r$ polynomials.

The recursive step is as follows:

From a geometric resolution of a zero-dimensional fiber of W'_{r+1} , we compute the Chow form of the variety obtained by intersecting W'_{r+1} with the set of zeroes of the next polynomial. From this Chow form, we obtain the Chow form of an equidimensional variety of dimension r which is a subset

of V and contains the equidimensional component of dimension r of V together with a geometric resolution of the zero-dimensional fiber of W'_r that is needed for the next recursive step.

The recursion yields the Chow forms of the components of a non-minimal equidimensional decomposition of V . The required *minimality* of the equidimensional decomposition imposes a third step in which we remove the spurious components.

Finally, a word with respect to practical implementations: there is a Magma package called *Kronecker* written by G. Lecerf (see [43]) which implements —with remarkable success— the geometric resolution algorithm. As we already said, our algorithm is closely related to this one, and so it seems possible that a deforested version of it (in the spirit of [29] or [35]) might lead to an implementation using this package as support.

The outline of the paper is the following:

In Section 1 we recall the definition and basic properties of the Chow form, and we make precise the data structure and the computational model we will use. We also describe some basic subroutines that we need in the sequel, and we estimate their complexities.

In Section 2 we present a deterministic algorithm for the computation of the Chow form of an equidimensional variety from a particular zero-dimensional fiber, provided some genericity conditions are fulfilled.

In Section 3 we describe the algorithm underlying Theorem 1, and we estimate its complexity. First we establish the relationship between geometric resolutions and Chow forms, and then we present subroutines for computing Chow forms of intersections and of components outside hypersurfaces. Combined with the algorithm in Section 2, this yields the desired algorithm.

In Section 4 we apply the main algorithm to the computation of sparse resultants, and to the resolution of generic over-determined equation systems.

Acknowledgments. The authors wish to thank Agnes Szántó, Grégoire Lecerf and Joos Heintz for several helpful discussions. We also thank Mike Shub for suggesting us the problem of solving generic over-determined systems. Finally we are indebted to the referees for their careful reading and suggestions.

1 Preliminaries

Throughout this paper \mathbb{Q} denotes the field of rational numbers, \mathbb{Z} the ring of rational integers, \mathbb{R} the field of real numbers, and \mathbb{C} the field of complex numbers. We denote by \mathbb{N} the set of positive rational integers, and we also denote by \mathbb{N}_0 the set of non-negative integers.

We denote by \mathbb{A}^n and \mathbb{P}^n the n -dimensional affine space and projective space over \mathbb{C} , respectively, equipped with the Zarisky topology definable over \mathbb{C} . A *quasiprojective* variety V is an open set of a closed projective (not necessarily irreducible) variety (we refer to [51, Section I.4] for a complete exposition of this notion). We denote by $\bar{V} \subset \mathbb{P}^n$ the projective closure of V , that is the minimal closed projective variety which contains it. Rational maps between quasiprojective varieties are indicated by $--\rightarrow$.

If f_1, \dots, f_s are polynomials in $\mathbb{Q}[x_0, \dots, x_n]$, $V(f_1, \dots, f_s)$ will denote the set of common zeroes of f_1, \dots, f_s in \mathbb{P}^n . This notation will also be used in the affine case.

Let V be a quasiprojective variety and let $g \in \mathbb{Q}[x_0, \dots, x_n]$ be a homogeneous polynomial. Then we denote by V_g the basic open set $V \setminus V(g)$ of V .

We adopt the usual notion of degree of an irreducible projective variety. The degree of an arbitrary projective variety is here the sum of the degrees of its irreducible components. If the variety is quasiprojective, its degree is defined as the degree of its projective closure.

We only consider polynomials and rational functions with coefficients in \mathbb{Q} and varieties defined by polynomials with coefficients in \mathbb{Q} unless otherwise explicitly stated.

The determinant of a square matrix M is denoted by $|M|$.

1.1 The Chow form of a quasiprojective variety

We gather in this subsection some definitions and basic facts about Chow forms. For a more detailed account we refer to [51, Section I.6.5], [21, Chapter 3] and [16].

First we define the notion of Chow form of an equidimensional quasiprojective variety:

Let $V \subset \mathbb{P}^n$ be an equidimensional quasiprojective variety of dimension r .

For $i = 0, \dots, r$ let $U_i = (U_{i0}, U_{i1}, \dots, U_{in})$ be a group of $n + 1$ variables and set $U := (U_0, \dots, U_r)$. Then set

$$L_i := U_{i0}x_0 + \dots + U_{in}x_n \in \mathbb{Q}[U][x]$$

for the associated generic linear form, where x denotes the group of variables (x_0, \dots, x_n) . Let

$$\Phi_V = \{(u_0, \dots, u_r; \xi) \in (\mathbb{P}^n)^{r+1} \times \mathbb{P}^n; \xi \in V, L_0(u_0, \xi) = 0, \dots, L_r(u_r, \xi) = 0\} \subset (\mathbb{P}^n)^{r+1} \times \mathbb{P}^n$$

be the incidence variety of these linear forms in V , and let $\pi : (\mathbb{P}^n)^{r+1} \times \mathbb{P}^n \rightarrow (\mathbb{P}^n)^{r+1}$ be the projection $(u, \xi) \mapsto u$.

Lemma 1.1 *Under the previous assumptions and notations, $\overline{\pi(\Phi_V)} = \pi(\Phi_{\overline{V}})$.*

Proof.— Let $V = \cup_C C$ be the irreducible decomposition of V . From the definition above we deduce that $\Phi_V = \cup_C \Phi_C$ and so $\pi(\Phi_V) = \cup_C \pi(\Phi_C)$.

We also have that $\overline{V} = \cup_C \overline{C}$ is the irreducible decomposition of \overline{V} . Then $\pi(\Phi_{\overline{V}}) = \cup_C \pi(\Phi_{\overline{C}})$ and so, without loss of generality, we can assume that V is irreducible.

The map $\Phi_V \rightarrow V$ defined by $(u, \xi) \mapsto \xi$ makes Φ_V a fiber bundle over V with fiber $(\mathbb{P}^{n-1})^{r+1}$. Then Φ_V is an irreducible variety of codimension $n + 1$, and the same is true for $\Phi_{\overline{V}}$.

As $\overline{\Phi_V}$ is a closed set, $\overline{\Phi_V} \subset \Phi_{\overline{V}}$. These are irreducible projective varieties of the same dimension and, therefore, they are equal. The fact that π is a closed map implies that $\overline{\pi(\Phi_V)} = \pi(\overline{\Phi_V})$. \square

Then $\overline{\pi(\Phi_V)} \subset (\mathbb{P}^n)^{r+1}$ is a closed hypersurface [51, p.66]. We define a *Chow form of V* as any squarefree defining equation $\mathcal{F}_V \in \mathbb{Q}[U_0, \dots, U_r]$ of the Zariski closure $\overline{\pi(\Phi_V)} \subset (\mathbb{P}^n)^{r+1}$. Note that the Chow form of an equidimensional variety is uniquely determined up to a scalar factor. We extend this to dimension -1 defining a Chow form of the empty variety as any non-zero constant in \mathbb{Q} .

This definition extends the usual notion of Chow form of closed projective equidimensional varieties. In fact, Lemma 1.1 states that a Chow form of an equidimensional quasiprojective variety is a Chow form of its projective closure.

From this definition, we see that any Chow form of V characterizes the sets of over-determined linear systems over the variety \overline{V} which intersect it: for $u_0, \dots, u_r \in \mathbb{C}^{n+1}$ we have

$$\mathcal{F}_V(u_0, \dots, u_r) = 0 \Leftrightarrow \overline{V} \cap \{L_0(u_0, x) = 0\} \cap \dots \cap \{L_r(u_r, x) = 0\} \neq \emptyset.$$

A Chow form \mathcal{F}_V is a multihomogeneous polynomial of degree $\deg V$ in each group of variables U_i ($0 \leq i \leq r$). The variety \overline{V} is uniquely determined by a Chow form of V ([51, p. 66]). Moreover, it is possible to derive equations for the variety \overline{V} from a Chow form of V ([21, Chapter 3, Corollary 2.6]).

In case V is irreducible, \mathcal{F}_V is an irreducible polynomial and, in the general case, a Chow form of V is the product of Chow forms of its irreducible components.

Following [41] we avoid the indeterminacy of \mathcal{F}_V by fixing one of its coefficients under the following assumption on the equidimensional quasiprojective variety V :

Assumption 1.2 *If $\dim V = 0$, we assume $V \subset \{x_0 \neq 0\}$. If $\dim V = r > 0$, we assume that the projection $\pi_V : V \dashrightarrow \mathbb{P}^r$ defined by $x \mapsto (x_0 : \dots : x_r)$ verifies $\#\pi_V^{-1}((1 : 0 : \dots : 0)) = \deg V$.*

This assumption implies that $\bar{V} \cap \{x_1 = 0\} \cap \cdots \cap \{x_r = 0\}$ is a 0-dimensional variety lying in the affine chart $\{x_0 \neq 0\}$. In particular V has no components contained in the hyperplane $\{x_0 = 0\}$. We also note that, in case V is a closed affine variety, the hypothesis $\#(V \cap \{x_1 = 0\} \cap \cdots \cap \{x_r = 0\}) = \deg V$ implies that the map $\pi_V : V \rightarrow \mathbb{A}^r$ defined by $x \mapsto (x_1, \dots, x_r)$ is finite; that is, the variables x_1, \dots, x_r are in Noether normal position with respect to V ([41, Lemma 2.14]).

Set e_i for the $(i+1)$ -vector of the canonical basis of \mathbb{Q}^{n+1} and $D := \deg V$. Then, under Assumption 1.2, $\mathcal{F}_V(e_0, \dots, e_r)$ —that is, the coefficient of the monomial $U_0^D \cdots U_r^D$ — is non-zero. Then, we define *the (normalized) Chow form* Ch_V of V by fixing the choice of \mathcal{F}_V through the condition

$$Ch_V(e_0, \dots, e_r) = 1.$$

Note that if V satisfies Assumption 1.2 then each of its irreducible components also does. Therefore, the normalized Chow form of V equals the product of the normalized Chow forms of its irreducible components. The normalized Chow form of the empty variety equals the polynomial 1.

Here are some examples of Chow forms:

- In case $\dim V = 0$ we have

$$\mathcal{F}_V(U_0) = \prod_{\xi \in V} L_0(U_0, \xi) \in \mathbb{Q}[U_0].$$

Furthermore, if V satisfies Assumption 1.2, Ch_V is equal to the above expression provided we choose homogeneous coordinates of the type $\xi := (1 : \xi^t) \in \mathbb{P}^n$ for each point in V .

- In case V is a hypersurface of degree d , then $V = V(F) \subset \mathbb{P}^n$ where $F \in \mathbb{Q}[x_0, \dots, x_n]$ is a squarefree homogeneous polynomial of degree d . We consider the $n \times (n+1)$ -matrix $M := (U_{ij})_{\substack{1 \leq i \leq n, \\ 0 \leq j \leq n}}$, and, for $0 \leq j \leq n$, we set M_j for the maximal minor obtained by deleting its $(j+1)$ column. Then

$$\mathcal{F}_V = F(M_0, -M_1, \dots, (-1)^n M_n) \in \mathbb{Q}[U_0, \dots, U_n].$$

In this case, Assumption 1.2 is equivalent to the fact that $f := F(1, 0, \dots, 0, t)$ is a squarefree polynomial of degree d in t . Therefore, Ch_V is equal to the above expression if we choose F such that the coefficient of the monomial x_n^d is 1.

- The sparse resultant provides an important family of examples: let $\mathcal{A} = \{\alpha_0, \dots, \alpha_N\} \subset \mathbb{Z}^n$ be a finite set of integer vectors, such that the differences of elements in \mathcal{A} generate \mathbb{Z}^n . Consider the map

$$\varphi_{\mathcal{A}} : (\mathbb{C}^*)^n \rightarrow \mathbb{P}^N, \quad \xi \mapsto (\xi^{\alpha_0} : \dots : \xi^{\alpha_N}).$$

This is always well-defined as $\xi_i \neq 0$ ($1 \leq i \leq n$) for all $\xi \in (\mathbb{C}^*)^n$. The Zariski closure of the image of this map $X_{\mathcal{A}} := \overline{\varphi_{\mathcal{A}}((\mathbb{C}^*)^n)} \subset \mathbb{P}^N$ is the *toric variety* associated to the set \mathcal{A} . This is an irreducible variety of dimension n and degree $\text{Vol}(\mathcal{A})$ (the normalized volume of the convex hull of \mathcal{A}).

The \mathcal{A} -resultant equals the Chow form of this variety [21, Chapter 8, Proposition 2.1], that is:

$$\mathcal{F}_{X_{\mathcal{A}}} = \text{Res}_{\mathcal{A}}.$$

We refer to [21] and to [15, Chapter 7] for a broader background on sparse resultants and toric varieties.

1.2 Data and algorithm structure

First we specify our data structure:

The objects we deal with are polynomials with rational coefficients. The data structure we adopt to represent them concretely is the *straight-line program* encoding (slp's for short). The input, output

and intermediate objects computed by our algorithms are polynomials codified through slp's. We emphasize on the fact that in the present work the crucial feature of slp's is their role as data structures, rather than the more traditional functional role as programs without branchings for the evaluation of polynomials at given points.

For the standard terminology of slp's, see [8, Definition 4.2]. In this paper all slp's are defined over \mathbb{Q} , without divisions and expecting the variables x_1, \dots, x_n as input.

For completeness we restate the definition in our particular case:

Let $n \in \mathbb{N}$. We denote by $\{+, -, \cdot\}$ the addition, subtraction and multiplication in the \mathbb{Q} -algebra $\mathbb{Q}[x_1, \dots, x_n]$. We consider apart the addition and multiplication by scalars, that is for $\lambda \in \mathbb{Q}$ and $f \in \mathbb{Q}[x_1, \dots, x_n]$ we set $\lambda^a(f) := f + \lambda$ and $\lambda^m(f) := \lambda \cdot f$. We denote by \mathbb{Q}^a and \mathbb{Q}^m the set of all scalar additions and multiplications for $\lambda \in \mathbb{Q}$, respectively.

We set $\Omega_n := \mathbb{Q}^a \cup \mathbb{Q}^m \cup \{+, -, \cdot\}$ and denote by $\text{ar}(\omega)$ the arity of an operation $\omega \in \Omega_n$: that is 1 if it is a scalar operation and 2 if it is a binary one.

A *straight-line program* γ (over \mathbb{Q} and expecting x_1, \dots, x_n as input) is a sequence $\gamma := (\gamma_1, \dots, \gamma_L)$ of *instructions*

$$\gamma_i = \begin{cases} (\omega_i; k_{i1}) & \text{if } \text{ar}(\omega_i) = 1 \\ (\omega_i; k_{i1}, k_{i2}) & \text{if } \text{ar}(\omega_i) = 2 \end{cases}$$

where each $\omega_i \in \Omega_n$ is an operation and for every j , $k_{ij} \in \mathbb{Z}$ satisfies $-n + 1 \leq k_{ij} \leq i - 1$ and represents a choice of a previous index. The number of instructions L in γ is called the *length* of γ and is denoted by $L(\gamma)$. This is, in the standard terminology, the complexity induced by the cost function which charges 1 to each operation in Ω_n (see [8, Definition 4.7]).

Given an slp $\gamma = (\gamma_1, \dots, \gamma_L)$, its *result sequence* (f_{-n+1}, \dots, f_L) is classically defined as

$$f_{-n+1} := x_1, \dots, f_0 := x_n \quad \text{and for } 1 \leq i \leq L, \quad f_i := \begin{cases} \omega_i(f_{k_{i1}}) & \text{if } \text{ar}(\omega_i) = 1 \\ \omega_i(f_{k_{i1}}, f_{k_{i2}}) & \text{if } \text{ar}(\omega_i) = 2 \end{cases}.$$

Here we make a slight modification of this terminology. According to the data structure role played by slp's we consider only the final result of the slp γ , that is the final polynomial $f_L \in \mathbb{Q}[x_1, \dots, x_n]$. We call it the *result* of γ . Here is the precise definition: Let $\Gamma_{\mathbb{Q}}[x_1, \dots, x_n]$ denote the set of slp's over \mathbb{Q} expecting x_1, \dots, x_n as input. Then there is a well-defined surjective function

$$\text{Eval} : \Gamma_{\mathbb{Q}}[x_1, \dots, x_n] \rightarrow \mathbb{Q}[x_1, \dots, x_n] \quad , \quad \gamma \mapsto f_L \quad \text{where } L := L(\gamma).$$

In this way each slp defines precisely one polynomial (and not a finite set). We say that $\gamma \in \Gamma_{\mathbb{Q}}[x_1, \dots, x_n]$ *encodes* $f \in \mathbb{Q}[x_1, \dots, x_n]$ if f is the result of γ . Given a polynomial $f \in \mathbb{Q}[x_1, \dots, x_n]$ we define its *length* $L(f)$ as the minimal length of an slp which encodes f . (We always have $\deg f \leq 2^{L(f)}$.) For a finite set $\mathcal{P} \subset \mathbb{Q}[x_1, \dots, x_n]$ we define naively its *length* as $L(\mathcal{P}) := \sum_{f \in \mathcal{P}} L(f)$.

From the dense representation $\sum_{\alpha} a_{\alpha} x^{\alpha}$ of a polynomial $f \in \mathbb{Q}[x_1, \dots, x_n]$ we obtain an slp for f in a straightforward manner. First, it is easy to show inductively that for any $r \in \mathbb{N}$, there is an slp of length bounded by $\binom{n+r}{r}$ whose result sequence is the set of all monomials x^{α} of degree $|\alpha| \leq r$. This is due to the fact that once one has a list of all such monic monomials of degree bounded by $r - 1$, each one of the $\binom{n+r-1}{r}$ homogeneous monic monomials of degree r is simply obtained from one of the list multiplying by a single variable. Now set $d := \deg f$. We multiply all monic monomials of degree bounded by d by their coefficients and add them up, that is we add $2 \binom{n+d}{d}$ instructions to the slp, in order to obtain an slp which encodes f . Hence

$$L(f) \leq 3 \binom{n+d}{d} \leq 3(d+1)^n.$$

We call this particular slp the *dense slp* of f . The previous computation shows that in all cases, the length $L(f)$ of a polynomial f of degree d is linearly bounded by its number of monomials.

We can operate with the data structure slp, extending directly the operations in Ω_n : for instance for $*$ $\in \{+, -, \cdot\}$, given two slp's $\gamma, \delta \in \Gamma_{\mathbb{Q}}[x_1, \dots, x_n]$ we obtain the new slp

$$\gamma * \delta := *(\gamma, \delta) := (\gamma_1, \dots, \gamma_{L(\gamma)}, \delta_1, \dots, \delta_{L(\delta)}, (*; L(\gamma), L(\gamma) + L(\delta))),$$

where the choice of previous indexes for δ are suitable modified. This slp obviously encodes the $*$ of the two polynomials encoded by γ and δ , and its length is $L(\gamma) + L(\delta) + 1$.

More generally, for $\gamma \in \Gamma_{\mathbb{Q}}[y_1, \dots, y_m]$ and $\delta_1, \dots, \delta_m \in \Gamma_{\mathbb{Q}}[x_1, \dots, x_n]$, we can define the *composition slp* $\gamma \circ \delta := \gamma \circ (\delta_1, \dots, \delta_m) \in \Gamma_{\mathbb{Q}}[x_1, \dots, x_n]$. We have

$$L(\gamma \circ \delta) = L(\delta_1) + \dots + L(\delta_m) + L(\gamma).$$

This operation is compatible with the map Eval, that is $\text{Eval}(\gamma \circ \delta) = \text{Eval}(\gamma) \circ \text{Eval}(\delta)$. Hence for $f \in \mathbb{Q}[y_1, \dots, y_m]$ and $g_1, \dots, g_m \in \mathbb{Q}[x_1, \dots, x_n]$ we have that $L(f(g_1, \dots, g_m)) \leq L(g_1) + \dots + L(g_m) + L(f)$.

Now we specify the computational model that produces and manipulates our data structure: it is the *Blum-Shub-Smale machine* over \mathbb{Q} (BSS machine for short), which captures the informal notion of uniform algorithm over \mathbb{R} . We refer to [5, Chapters 3 and 4] for the definition, terminology and basic properties. However, there are again some slight modifications in our definition (restrictions on the operations –only over rational numbers– and the branches –only equality of numbers to zero), and we restate it for purpose of completeness:

We recall that a BSS machine \mathcal{M} over \mathbb{Q} has five types of nodes: input, computation, branch, shift, and output.

Set

$$\mathbb{Q}^{\infty} := \bigsqcup_{n \geq 0} \mathbb{Q}^n$$

for the disjoint union of the n -dimensional spaces \mathbb{Q}^n , i.e. the natural space to represent problem instances of arbitrarily large dimension. For $a \in \mathbb{Q}^{\ell} - \mathbb{Q}^{\ell-1} \subset \mathbb{Q}^{\infty}$ we call ℓ the *size* of a , and we denote it by $\text{size}(a)$.

On the other hand, let

$$\mathbb{Q}_{\infty} := \bigoplus_{m \in \mathbb{Z}} \mathbb{Q}$$

be the bi-infinite direct sum space over \mathbb{Q} . The elements $b \in \mathbb{Q}_{\infty}$ are of the form

$$b := (\dots, b_{-2}, b_{-1}, b_0 \cdot b_1, b_2, \dots),$$

where $b_i = 0$ for $|i| \gg 0$. The dot between b_0 and b_1 is a distinguished marker which enables one to visualize the position of the coordinates of b .

Now we define the computation nodes. There are of two types, operations between entries of b or scalar operations:

For each $\omega \in \{+, -, \cdot, /\}$ and $i, j, k \in \mathbb{N}$ there is a map

$$\mathbb{Q}_{\infty} \rightarrow \mathbb{Q}_{\infty}, \quad b \mapsto (\dots, b_{k-2}, b_{k-1}, \omega(b_i, b_j), b_{k+1}, \dots)$$

(observe that unlike in the case of our data structure, here we also allow divisions). A division will be performed only after a branch node, in case the divisor is known to be non-zero. Also for each $\lambda^a \in \mathbb{Q}^a$ or $\lambda^m \in \mathbb{Q}^m$ and $i, k \in \mathbb{N}$ there is in an analogous way a map

$$\mathbb{Q}_{\infty} \rightarrow \mathbb{Q}_{\infty}, \quad b \mapsto (\dots, b_{k-2}, b_{k-1}, \lambda^a(b_i), b_{k+1}, \dots) \quad \text{or} \quad b \mapsto (\dots, b_{k-2}, b_{k-1}, \lambda^m(b_i), b_{k+1}, \dots).$$

The only branch node we allow is the one which checks the equality $b_1 = 0$. In other words its associated map is $\mathbb{Q}_{\infty} \rightarrow \mathbb{Q}, \quad b \mapsto b_1$.

The shift nodes are of two types: shifting to the left or to the right, associated with the two maps $\mathbb{Q}_{\infty} \rightarrow \mathbb{Q}_{\infty}, \quad b \mapsto \sigma_l(b)_i = b_{i+1}$ or $\sigma_r(b)_i = b_{i-1}$.

The machine \mathcal{M} over \mathbb{Q} is a finite connected directed graph containing these five types of nodes (input, computation, branch, shift and output). The space \mathbb{Q}^{∞} is both the input space $\mathcal{I}_{\mathcal{M}}$ and the output space $\mathcal{O}_{\mathcal{M}}$, and \mathbb{Q}_{∞} is the state space $\mathcal{S}_{\mathcal{M}}$, that is the “working” space of the machine. The *dimension* $K_{\mathcal{M}}$ of the machine \mathcal{M} is the maximum dimension of the computation maps, which,

under our assumptions, coincides with the maximum of the natural numbers i, j and k involved in the computations.

We are interested in the *algebraic* complexity of these machines. We assume that the cost of each computation, branch and shift node is 1. Hence the *complexity* $C_{\mathcal{M}}(a)$ of the machine \mathcal{M} on an input a is just the number of computation, branch and shift nodes of the graph, from input to output.

Observe that any slp $\gamma \in \Gamma_{\mathbb{Q}}[x_1, \dots, x_n]$ is an example of a (finite-dimensional) BSS machine M_{γ} without branches or shift nodes for computing $f := \text{Eval}(\gamma) \in \mathbb{Q}[x_1, \dots, x_n]$ at any input point $a \in \mathbb{Q}^n$. The dimension of this machine is $n + L(\gamma)$ and its complexity is $L(\gamma)$.

Given $\ell \in \mathbb{N}$ we consider the *complexity* $C_{\mathcal{M}}(\ell)$ of the machine on inputs of size bounded by ℓ , that is

$$C_{\mathcal{M}}(\ell) := \sup \{C_{\mathcal{M}}(a) ; \text{size}(a) \leq \ell\}.$$

Since in particular, \mathcal{M} does arithmetic operations in \mathbb{Q} at unit cost, this computational model is a natural algebraic analogue of the notion of Turing machine. It provides a basis for the implementation of *uniform* procedures. The only difference with the Turing model is that one replaces bit operations by arithmetic ones. Moreover, since all the involved computations are done over the rational field, the machine \mathcal{M} can be effectively transformed into a classical Turing machine. However our complexity counting does not provide any reasonable control on the complexity of the resulting Turing machine.

Shift nodes can be regarded as the management done by the machine \mathcal{M} to execute the arithmetic operations and comparisons. In the sequel, we describe our algorithms as informal computer programs, that means that we just count the arithmetic operations and comparisons but *not* the shifts. The complexity of a formal BSS machine is polynomial in the complexity of the corresponding informal description (see [5, Section 4.4]) and so our results remain valid for the BSS model as described above.

As we have already anticipated, our algorithms are BSS machines that manipulate slp's. A machine \mathcal{M} receives as input a finite family of slp's $\gamma \in \Gamma_{\mathbb{Q}}[x_1, \dots, x_n]^M$ and gives back a finite family of slp's $\mathcal{M}(\gamma) \in \Gamma_{\mathbb{Q}}[y_1, \dots, y_m]^{M'}$.

A finite family of slp's $\gamma \in \Gamma_{\mathbb{Q}}[x_1, \dots, x_n]^M$ can be easily codified as an input element in $\mathcal{I}_{\mathcal{M}} = \mathbb{Q}^{\infty}$, in fact γ can be identified with a vector in $\mathbb{Q}^{M+3L(\gamma)}$ in the following way:

The first coordinate is for the dimension n , that is the number of variables. Then each instruction of the first slp γ_1 is codified as a triple: we enumerate the operations in Ω_n with numbers from 2 to 6, 2 and 3 corresponding to the operations in \mathbb{Q}^a and \mathbb{Q}^m , and 4 to 6 to $+$, $-$ and \cdot . For operations in $\mathbb{Q}^a \cup \mathbb{Q}^m$ we write the operation number in the first coordinate, the corresponding coefficient in the second one, and the position to which it applies in the third one. The binary operations are codified in a similar way, by writing first the operation number, and then the position of the two elements to which it applies. The positions are numbered from $1 - n$ to $L(\gamma)$ according to the definition of the result sequence. For instance the vector $(2, (3, 5, -1), (4, 0, 1), (6, 2, 2))$ codifies the slp $x_1, x_2, 5x_1, x_2 + 5x_1, (x_2 + 5x_1)^2$. The instruction to separate two consecutive slp's is an empty cell, that is a 0. The second slp γ_2 is now codified exactly as the first one. Therefore, $\gamma := (\gamma_1, \dots, \gamma_M) \in \Gamma_{\mathbb{Q}}[x_1, \dots, x_n]^M$ is codified as a vector in \mathbb{Q}^{∞} , in fact in $\mathbb{Q}^{M+3L(\gamma)}$ since we need to add "0" $M - 1$ times to separate two consecutive slp's.

The machine \mathcal{M} manipulates this input, the finite family of slp's $\gamma \in \Gamma_{\mathbb{Q}}[x_1, \dots, x_n]^M$: it operates on these slp's and gives as the output an element of $\mathcal{O}_{\mathcal{M}}$ corresponding to a finite family of slp's in $\Gamma_{\mathbb{Q}}[y_1, \dots, y_m]^{M'}$. As we have just seen, the input and output size is (essentially) the length of each of these families. Thus, we speak here of a finite family of slp's γ as the input of \mathcal{M} and we simply denote by $\mathcal{M}(\gamma)$ its output in $\Gamma_{\mathbb{Q}}[y_1, \dots, y_m]^{M'}$.

Remark 1.3 Let $\gamma \in \Gamma_{\mathbb{Q}}[x_1, \dots, x_n]^M$ be the input slp family of a BSS machine \mathcal{M} and let $\mathcal{M}(\gamma) \in \Gamma_{\mathbb{Q}}[y_1, \dots, y_m]^{M'}$ be its output. Then

$$L(\mathcal{M}(\gamma)) \leq 3L(\gamma) + C_{\mathcal{M}}(\gamma).$$

Proof.— As we do not know how the machine \mathcal{M} operates on γ , the only bound for $L(\mathcal{M}(\gamma))$ is the number of operations labelled from 2 to 6 of the representation of $\mathcal{M}(\gamma)$ in \mathcal{M} , which is bounded by the number of non-zero cells of this representation minus 1 (since the first cell of the output corresponds to the number of variables m of the output). This is bounded by $1 + 3L(\gamma) + C_{\mathcal{M}}(\gamma) - 1$, that is, the size of γ as an input of \mathcal{M} (excepting the $M - 1$ zero-cells separating different input slp's) plus the number of computation nodes $C_{\mathcal{M}}(\gamma)$ minus 1. \square

Our main algorithms are in fact probabilistic. For this reason we implement them in the model of *probabilistic BSS machine* over \mathbb{Q} [5, Section 17.1]. This is a BSS machine \mathcal{M} with an additional kind of node, called probabilistic. These are nodes that have two next nodes and no associated map and that “flip coins”, that is when a computation reaches a probabilistic node, it randomly chooses the next node between the two possible ones with probability $1/2$ for each of them.

In this probabilistic setting, each run —on the same given input γ — of the machine \mathcal{M} may lead to a different path computation. In our case, for any given input, the number of probabilistic nodes traversed is finite, that is, the number of possible paths is finite. We treat the probabilistic nodes as branches and charge cost 1 for each of them.

As every path \mathcal{P} of \mathcal{M} corresponds to a BSS machine of complexity $C_{\mathcal{P}}(\gamma)$, the algebraic complexity $C_{\mathcal{M}}(\gamma)$ of the machine \mathcal{M} on the input γ turns out to be a random variable, with finite sample set. Moreover, again in our context, every path is finite: it may happen that a branch ends in an error message but in any case the complexity of any path is bounded. Thus the random variable $C_{\mathcal{M}}(\gamma)$ satisfies

$$\text{Prob}(C_{\mathcal{M}}(\gamma) = C) := \sum \text{Prob}(\mathcal{P}; \mathcal{P} \text{ path such that } C_{\mathcal{P}}(\gamma) = C).$$

We are interested in the *worst-case complexity* $C_{\mathcal{M}}^{\max}(\gamma)$, the maximum complexity of the paths of \mathcal{M} on γ , and the *expected complexity* $E_{\mathcal{M}}(\gamma)$, defined as the (finite) expectation of this random variable, that is

$$E_{\mathcal{M}}(\gamma) := E(C_{\mathcal{M}}(\gamma)) = \sum_{C \in \mathbb{N}} C \cdot \text{Prob}(C_{\mathcal{M}}(\gamma) = C).$$

Observe that $C_{\mathcal{M}}^{\max}(\gamma) \geq E_{\mathcal{M}}(\gamma)$ always holds.

As before, we also consider the function $E_{\mathcal{M}} : \mathbb{N} \rightarrow \mathbb{N}$:

$$E_{\mathcal{M}}(\ell) := \sup \{E_{\mathcal{M}}(\gamma); n, M \in \mathbb{N}, \gamma \in \Gamma_{\mathbb{Q}}[x_1, \dots, x_n]^M \text{ and } M + 3L(\gamma) \leq \ell\}.$$

We define now the error probability of the machine on a given input. Again, there is here a modification with respect to traditional probabilistic BSS machines. Keeping in mind that for any run of the probabilistic machine \mathcal{M} on the input $\gamma \in \Gamma_{\mathbb{Q}}[x_1, \dots, x_n]^M$, its output (independently of the path randomly taken) encodes a finite family of polynomials $f := (f_1, \dots, f_M) \in \mathbb{Q}[y_1, \dots, y_m]^{M'}$ we define:

Definition 1.4 (*Bounded Probability Algorithm*) *Given $\gamma \in \Gamma_{\mathbb{Q}}[x_1, \dots, x_n]^M$ and given a set of polynomials $f := (f_1, \dots, f_{M'}) \in \mathbb{Q}[y_1, \dots, y_m]^{M'}$, the error probability $e_{\mathcal{M}}(\gamma, f)$ that \mathcal{M} computes f from the given input γ is the probability that the output of \mathcal{M} from γ does not encode f ; that is the probability that the computation finishes with an error message, or that it outputs $\delta \in \Gamma_{\mathbb{Q}}[y_1, \dots, y_m]^{M'}$ which does not encode f .*

We say that \mathcal{M} computes f if $e_{\mathcal{M}}(\gamma, f) \leq 1/4$. As this happens at most for one f , when it happens, we set $e_{\mathcal{M}}(\gamma) := e_{\mathcal{M}}(\gamma, f)$.

When $e_{\mathcal{M}}(\gamma) \leq 1/4$ for every input γ , we say that \mathcal{M} is a bounded probability machine for polynomial slp's, and we speak of a bounded probability algorithm.

Observe that our probabilistic machine is a little unusual since in fact, as different slp's may encode the same polynomial, the polynomial f computed by the probabilistic machine \mathcal{M} corresponds to

an equivalence class of outputs rather than a single one. In this paper, all machines are bounded probability machines for polynomial slp's in the sense of this definition.

In our setting, probability is introduced by choosing a random element with equidistributed probability in a set $[0, \ell)^n := \{0, \dots, \ell - 1\}^n$ for given natural numbers ℓ and n . Since probabilistic machines flip coins to decide binary digits, each of these random choices can be easily simulated with a machine with complexity $\mathcal{O}(n \lceil \log \ell \rceil)$, where here and in the sequel, \log denotes logarithm in base 2. This machine is denoted by $\text{Random}(n, \ell)$.

In this work, in each case, there is a non-zero polynomial $F \in \mathbb{Q}[x_1, \dots, x_n] \setminus \{0\}$ such that a random choice a is good—that is, leads to the computation of the desired output—if $F(a) \neq 0$. The error probability of this random choice is then estimated by the Zippel-Schwartz zero test ([57], [50, Lemma 1]):

$$\text{Prob}(F(a) = 0) \leq \frac{\deg F}{\ell}.$$

The choice of $1/4$ as a bound for the error probability is not restrictive and we can easily modify it in order to reduce the error probability as much as desired. The usual procedure is to run the machine \mathcal{M} many times and to declare that the polynomial family f is computed by \mathcal{M} if it is the output of more than half the times. There is a slight difficulty here, appearing from the fact that our machine computes slp's instead of polynomials, and two different runs may lead to different encodings of the same polynomials. That is why we need here to be more careful in our definition. We define it in the following way:

Given the bounded probability machine \mathcal{M} which on input $\gamma \in \Gamma_{\mathbb{Q}}[x_1, \dots, x_n]^M$ computes $f \in \mathbb{Q}[y_1, \dots, y_m]^{M'}$, and given $s \in \mathbb{N}$, the machine \mathcal{M}_s is the machine which performs the following tasks:

1. \mathcal{M}_s runs s times the machine \mathcal{M} on the given input $\gamma \in \Gamma_{\mathbb{Q}}[x_1, \dots, x_n]^M$: for $1 \leq i \leq s$ it obtains the output slp family $\delta_i \in \Gamma_{\mathbb{Q}}[y_1, \dots, y_m]^{M'}$ together with the complexity C_i of the path followed to compute δ_i .
2. Then \mathcal{M}_s chooses randomly $a \in [0, M' 2^{s+3L(\gamma)+C_1+\dots+C_s})^m$ and computes $\delta_i(a)$, $1 \leq i \leq s$.
3. For $j = 1$ to $\lceil s/2 \rceil$
 - it computes $\delta_j(a) - \delta_k(a)$, $j < k \leq s$, and compares the results to 0.
 - if 0 is obtained for strictly more than $s/2$ values of k , it returns the polynomial family $f \in \mathbb{Q}[y_1, \dots, y_m]^{M'}$ encoded by δ_j as the output and ends.
 - if not, it goes to $j + 1$.

If for no $j \leq \lceil s/2 \rceil$ the algorithm obtains 0 enough times, it outputs error and ends.

Proposition 1.5 *Given a bounded probability machine \mathcal{M} which on $\gamma \in \Gamma_{\mathbb{Q}}[x_1, \dots, x_n]^M$ computes $f \in \mathbb{Q}[y_1, \dots, y_m]^{M'}$ and given $s \in \mathbb{N}$, the worst case complexity, the expected complexity and the error probability of the machine \mathcal{M}_s on γ verify the following bounds:*

$$\begin{aligned} C_{\mathcal{M}_s}^{\max}(\gamma) &= \mathcal{O}((m+1)s(L(\gamma) + C_{\mathcal{M}}^{\max}(\gamma)) + m \log M') + M' \binom{s}{2}, \\ E_{\mathcal{M}_s}(\gamma) &= \mathcal{O}((m+1)s(L(\gamma) + E_{\mathcal{M}}(\gamma)) + m \log M' + M' \binom{s}{2}), \\ e_{\mathcal{M}_s}(\gamma) &\leq 2(3/4)^{s/2}. \end{aligned}$$

Proof.— Let us begin by describing the algebraic complexity C of a given run of the machine $C_{\mathcal{M}_s}$ on γ in terms of the complexities C_i of the paths followed by the machine \mathcal{M} on γ on the i -th run. Cost of Item 1 above: It has complexity $C_1 + \dots + C_s$.

Cost of Item 2: Producing the random choice a costs $\mathcal{O}(m(\log M' + s + L(\gamma) + C_1 + \dots + C_s))$, and, from Remark 1.3, computing $\delta_1(a), \dots, \delta_s(a)$ costs $3sL(\gamma) + C_1 + \dots + C_s$.
Cost of Item 3: As $\delta_j(a) \in \mathbb{Q}^{M'}$, to compute all $\delta_j(a) - \delta_k(a)$ and compare them to 0 costs $2M' \binom{s}{2}$.
Hence, the worst-case complexity of the machine \mathcal{M}_s on γ is

$$C_{\mathcal{M}_s}^{\max}(\gamma) = \mathcal{O}((m+1)s(L(\gamma) + C_{\mathcal{M}}^{\max}(\gamma)) + m \log M' + M' \binom{s}{2}),$$

while, as the complexity is an affine combination of the s independent random variables C_1, \dots, C_s , its expectation verifies

$$E_{\mathcal{M}_s}(\gamma) = \mathcal{O}((m+1)s(L(\gamma) + E_{\mathcal{M}}(\gamma)) + m \log M' + M' \binom{s}{2}).$$

The error probability is bounded by the probability that there is no group of more than $s/2$ vectors which coincide plus the probability that $\delta_j(a) = \delta_k(a)$ but the two polynomial families encoded by δ_j and δ_k do not coincide.

The first probability of error is bounded by $(3/4)^{s/2}$ as in [5, Sect.17.2, Lemma 1]. To estimate the second probability of error we apply Schwartz' lemma: for $1 \leq i \leq s$ the output δ_i encodes $f \in \mathbb{Q}[y_1, \dots, y_m]^{M'}$ where the degree of each component is bounded by $2^{3L(\gamma) + C_i}$. Thus the probability of error of one test is bounded by $(M' 2^{3L(\gamma) + C_i}) / (M' 2^{s+3L(\gamma) + C_1 + \dots + C_s}) \leq (1/2)^s$. As there are at most $\binom{s}{2}$ such independent tests, the total error probability verifies

$$e_{\mathcal{M}_s}(\gamma) \leq (3/4)^{s/2} + \binom{s}{2} (1/2)^s \leq 2(3/4)^{s/2}$$

for $s \geq 2$. □

Corollary 1.6 *Given a bounded probability machine \mathcal{M} which on $\gamma \in \Gamma_{\mathbb{Q}}[x_1, \dots, x_n]^M$ computes $f \in \mathbb{Q}[y_1, \dots, y_m]^{M'}$ and given $N \in \mathbb{N}$, $N \geq 4$, the error probability of the machine \mathcal{M}_s on γ for $s := \lceil 6(\log N + 1) \rceil$ is bounded by $1/N$ while its worst-case complexity is of order*

$$\mathcal{O}((m+1) \log N (L(\gamma) + C_{\mathcal{M}}^{\max}(\gamma)) + m \log M' + M' \log^2 N).$$

Proof.— As $(3/4)^3 < (1/2)$,

$$e_{\mathcal{M}_s}(\gamma) \leq 2(3/4)^{3(\log N + 1)} \leq (3/4)^{3 \log N} \leq (1/2)^{\log N} = 1/N. □$$

Corollary 1.6 will be used to decrease the error probability of intermediate subroutines of our main algorithm and keep control of the complexity in order that the error probability of the latter is bounded by $1/4$. Observe that the length of the output slp is of the same order that the length of the slp obtained when running any of the repetitions of the algorithm.

Given a bounded probability machine \mathcal{M} , any time we want to obtain the output of \mathcal{M} for an slp input family γ with error probability bounded by $1/N$, we run Subroutine 1 which gives a new probability machine $\mathcal{M}(\gamma; N)$ doing so. Any time we run \mathcal{M} for the input family γ , we will denote by $\text{Complexity}(\mathcal{M}(\gamma))$ the complexity of doing it this time.

1.3 Complexity of basic computations

We summarize the complexity of the basic operations on polynomials and matrices our algorithms rely on. As our interest is mostly theoretical, it will be sufficient for us to apply the more naive procedures for these operations. For the more advanced complexity results, we refer to [8] for a complete account, see also [29] for a brief survey of the existing literature.

Subroutine 1: Decreasing error probability of \mathcal{M}

```

procedure  $\mathcal{M}(\gamma; N)$ 
#  $\gamma$  is an slp input family for  $\mathcal{M}$ .
#  $N \in \mathbb{N}, N \geq 4$ .
# The procedure returns the output of  $\mathcal{M}$  with error probability bounded by  $1/N$ .

1.  $s := \lceil 6(\log N + 1) \rceil$ ;
2. for  $i$  from 1 to  $s$  do
3.    $(\delta_i, C_i) := (\mathcal{M}(\gamma), \text{Complexity}(\mathcal{M}(\gamma)))$ ;
4. od;
5.  $a := \text{Random}(m, M' 2^{s+3L(\gamma)+C_1+\dots+C_s})$ ;
6.  $(\Delta_1, \dots, \Delta_s) := (\delta_1(a), \dots, \delta_s(a))$ ;
7.  $j := 1$ ;
8. while  $j \leq \lceil s/2 \rceil$  do
9.    $k := j + 1$ ;
10.   $t := 0$ ;
11.  while  $k \leq s$  do
12.    if  $\Delta_j - \Delta_k = 0$  then
13.       $t := t + 1$ ;
14.    fi;
15.     $k := k + 1$ ;
16.  od;
17.  if  $t \geq s/2$  then
18.    return $(\delta_j)$ ;
19.  else
20.     $j := j + 1$ ;
21.  fi;
22. od;
23. return("error");

end.

```

Let R denote a commutative \mathbb{Q} -algebra and let $d \in \mathbb{N}$.

The multiplication of $d \times d$ -matrices with coefficients in R can be done with $\mathcal{O}(d^3)$ operations of R and no branches. The computation of the coefficients of the characteristic polynomial of a $d \times d$ -matrix—and in particular the computation of the adjoint and the determinant of this matrix—can be done with $\mathcal{O}(d^4)$ arithmetic operations and no branches, the same bounds hold for the inversion of an invertible matrix ([2], [1]).

Addition of univariate polynomials with coefficients in R and degree bounded by d can be done in $d + 1$ arithmetic operations, while the straightforward polynomial multiplication algorithm takes $\mathcal{O}(d^2)$ arithmetic operations and has no branches. Division with remainder—provided the divisor is a monic polynomial—has also in a straightforward manner complexity $\mathcal{O}(d^2)$ and no branches. The greatest common divisor can be computed through subresultants with $\mathcal{O}(d)$ branches (computing the degree of the gcd corresponds to checking the vanishing of the determinant of submatrices of

the Sylvester matrix) and complexity $\mathcal{O}(d^5)$ (solving a linear system) ([14], [7]).

Now we are going to consider some procedures involving polynomials encoded by slp's.

First, given an slp γ which computes $f \in \mathbb{Q}[x_1, \dots, x_n]$ and given $a \in \mathbb{Q}^n$, we can compute $f(a) \in \mathbb{Q}$ within complexity $L(\gamma)$ and so, we can also check $f(a) = 0$ within the same complexity. The derivative of the polynomial f with respect to one of its variables will be computed by means of the Baur-Strassen's algorithm (see [3]) within complexity $\mathcal{O}(L(\gamma))$.

For a group of variables $y := (y_1, \dots, y_m)$ and $a \in \mathbb{Q}^m$, we will denote by $\text{Expand}(f, y, a, d)$ the subroutine which, given an slp γ which encodes a multivariate polynomial f , computes as intermediate results slp's for the homogeneous components centered at a and of degree bounded by d of the polynomial f with respect to the given group of variables y : In [40, Lemma 13], [8, Lemma 21.25] are given slp's of length $\mathcal{O}(d^2 L(\gamma))$ in which all the homogeneous components of f of degree bounded by d appear as intermediate computations. These procedures can be easily modified within the same complexity to compute the homogeneous components centered at a and up to degree d of a polynomial with respect to the given group of variables. In particular, if y consists of a single variable and $a = 0$, this procedure computes the coefficients of the given polynomial with respect to y .

Quite frequently we use a mixed representation of f : instead of encoding it by means of a single slp, we consider f as a polynomial in a distinguished variable, and if d is a bound for the degree of f in this variable, we give a $(d+1)$ -uple of slp's, which encode the coefficients f_0, \dots, f_d of f with respect to the variable. The length of this mixed encoding does not essentially differ from the length of f ; denote by $L'(f)$ the length of the mixed encoding and by $L(f)$ the length of f , we have:

$$L(f) = \mathcal{O}(d + L'(f)) \quad \text{and} \quad L'(f) = \mathcal{O}(d^2 L(f)).$$

Sometimes we need to compute the exact degree of a polynomial with respect to a particular variable. We will call $\text{Deg}(f, d)$ the procedure which computes the degree of the univariate polynomial f given by its dense representation, where d is a bound for its degree. This computation is done by simply comparing the coefficients of f with 0. This procedure can be adapted to obtain a probabilistic algorithm $\text{Deg}(f_1, \dots, f_s, x, d; N)$ which computes, with error probability bounded by $1/N$, the total degrees of the polynomials f_1, \dots, f_s in the group of variables x , from slp's encoding f_1, \dots, f_s and an upper bound d for their degrees in the variables x . To do so, first we apply subroutine $\text{Expand}(f_i, x, 0, d)$ for $1 \leq i \leq s$, to obtain the homogeneous components of f_i . Then by choosing a random point in $[0, 1, \dots, sdN]^n$ we decide probabilistically which is the component of greatest degree different from zero of each polynomial f_1, \dots, f_s . If the given polynomials are encoded by slp's of length bounded by L , the worst-case complexity of this procedure is of order $\mathcal{O}(sd^2L + n \log(sdN))$.

1.4 Effective division procedures

Here, we gather the division subroutines we will need. Basically, they compute the division of multivariate polynomials and power series, and the greatest common divisor of multivariate polynomials. In all cases, the objects will be *multivariate polynomials encoded by slp's* and power series, whose known graded parts will be also *encoded by slp's*. The proposed procedure for multivariate power series division is new and plays an important role in Subroutine 7, which in turn is the key step of our main algorithm.

The following subroutine is the well-known Strassen's Vermeidung von Divisionen (division avoiding) algorithm ([53]). We re-prove it briefly in order to estimate its complexity.

Lemma 1.7 *Let $f, g \in \mathbb{Q}[x_1, \dots, x_n]$ be polynomials encoded by slp's of length bounded by L such that f divides g . Let $d \in \mathbb{N}$ be such that $\deg(g/f) \leq d$, and $a \in \mathbb{Q}^n$ such that $f(a) \neq 0$. Then PolynomialDivision (Subroutine 2) computes g/f within complexity $\mathcal{O}(d^2(d+L))$.*

Subroutine 2: Polynomial Division

procedure PolynomialDivision(f, g, d, a)

$f, g \in \mathbb{Q}[x_1, \dots, x_n]$ such that f divides g ,
 # $d \in \mathbb{N}$ an upper bound for the degree of the quotient g/f ,
 # $a \in \mathbb{Q}^n$ such that $f(a) \neq 0$.
 # The procedure returns $h := g/f$.

1. $\alpha := f(a)$;
2. $v := \frac{1}{\alpha} \sum_{i=0}^d \left(\frac{t}{\alpha}\right)^i$;
3. $H = g \cdot v(\alpha - f)$;
4. $(H_0, \dots, H_d) := \text{Expand}(H, x, a, d)$;
5. $h := \sum_{m=0}^d H_m$;
6. **return**(h);

end.

Proof.– The quotient polynomial $h := g/f \in \mathbb{Q}[x_1, \dots, x_n]$ can also be seen as a power series in $\mathbb{Q}[[x - a]]$. For $\alpha := f(a)$, we have

$$h = \frac{g}{f} = g \alpha^{-1} \left(1 - \frac{\alpha - f}{\alpha}\right)^{-1} = g \sum_{i=0}^{\infty} \frac{(\alpha - f)^i}{\alpha^{i+1}} \in \mathbb{Q}[[x - a]].$$

For

$$H := g \sum_{i=0}^d (\alpha - f)^i / \alpha^{i+1} \in \mathbb{Q}[x_1, \dots, x_n]$$

we have $h \equiv H \pmod{(x - a)^{d+1}}$. Thus, if $(H_m)_{m \leq d}$ are the homogeneous components of H centered at a and of degree bounded by d , we conclude $h = \sum_{m=0}^d H_m$.

The stated complexity is obtained as follows: We compute the univariate polynomial v with $\mathcal{O}(d+L)$ operations. Hence we compute H within complexity $\mathcal{O}(d+L)$. We compute its homogeneous components in $x - a$ up to degree d within complexity $\mathcal{O}(d^2(L(H))) = \mathcal{O}(d^2(d+L))$. Finally we obtain h as $\sum_{m=0}^d H_m$ within the same complexity bound. \square

Observe that the same procedure can be used to compute the graded parts centered at a and of a certain bounded degree of the rational function g/f , even in case f does not divide g . We denote this subroutine by $\text{GradedParts}(f, g, D, a)$, where the argument D corresponds to the bound for the degree of the graded parts to be computed. Its complexity is of order $\mathcal{O}(D^2(D+L))$.

Subroutine 2 converts slp's *with* divisions computing polynomials in $\mathbb{Q}[x_1, \dots, x_n]$ into ordinary slp's: Slp's with divisions are defined as ordinary slp's, but with the set of basic operations enlarged to include the division, which we denote by the bar $/$. A further requirement is that all divisions should be *well-defined*, that is, no intermediate denominator can be zero. In general, the result of an slp with divisions is a rational function in $\mathbb{Q}(x_1, \dots, x_n)$.

Observe that, given an slp with divisions γ which encodes a rational function h , we can easily compute *separately* a numerator g and a denominator f by means of two slp's ζ, η without divisions: for instance, for each *addition* $h_i := h_j + h_k$ in the result sequence of γ , if $h_j := h_{j1}/h_{j2}$ and $h_k := h_{k1}/h_{k2}$, we set $g_k := h_{j1} h_{k2} + h_{j2} h_{k1}$ and $h_k := h_{j2} h_{k2}$ for the corresponding result sequence in ζ and η respectively. We proceed analogously for the other operations in $\Omega'_n := \Omega_n \cup \{/\}$.

Subroutine 3: Greatest Common Divisor

```

procedure GCD( $f, g, x, d$ )

#  $f, g \in \mathbb{Q}[x_1, \dots, x_n]$  of degrees bounded by  $d$ ;
#  $x := (x_1, \dots, x_n)$ ;
# The procedure returns  $h := \gcd(f, g)$ .

1.  $a := \text{Random}(n, 8d(d+1))$ ;
2. if  $f(a) = 0$  then
3.   return("error");
4. else
5.    $(f_0, \dots, f_d) := \text{Expand}(f, x, a, d)$ ;
6.    $(g_0, \dots, g_d) := \text{Expand}(g, x, a, d)$ ;
7.    $e := 0$ ;
8.   while  $g_e(a) = 0$  and  $e \leq d$  do
9.      $e := e + 1$ ;
10.  od;
11.  if  $e = d + 1$  then
12.    return( $f$ );
13.  else
14.     $F := \sum_{k=0}^d f_k t^{d-k}$  and  $G := \sum_{k=0}^e g_k t^{e-k}$ ;
15.     $(q, Q) := \text{GCD1}(F, G, d, e)$ ;
16.     $h := \text{PolynomialDivision}(q(x), Q(x, 1), d, q(a))$ ;
17.    return( $h$ );
18.  fi;
19. fi;

end.

```

We have

$$h := \text{Eval}(\gamma) = \frac{\text{Eval}(\zeta)}{\text{Eval}(\eta)}.$$

Furthermore the slp's ζ and η can be computed within complexity $L(\zeta) \leq 3L(\gamma)$ and $L(\eta) \leq L(\gamma)$. In particular, given $a \in \mathbb{Q}^n$, we can check if γ is well-defined at a and, if that is the case, if $h(a) = 0$ within complexity $\mathcal{O}(L(\gamma))$.

In case h is a *polynomial* of degree bounded by d , the previous considerations together with Lemma 1.7 show that we can compute an slp *without divisions* for h with complexity $\mathcal{O}(d^2(d + L(\gamma)))$.

Now follows a bounded probability algorithm (in the sense of Definition 1.4) to compute the greatest common divisor of two multivariate polynomials encoded by slp's (for another algorithm solving this task see [38]). Herein, $\text{GCD1}(F, G, d, e)$ is the subresultant algorithm which computes a greatest common divisor of two univariate polynomials F and G of degrees d and e respectively with coefficients in a ring A . The output of GCD1 is (q, Q) , where $q \in A - \{0\}$ and Q is the multiple by q of the monic greatest common divisor of F and G over the fraction field of A .

Lemma 1.8 *Let $f, g \in \mathbb{Q}[x_1, \dots, x_n]$ be polynomials of degree bounded by d encoded by slp's of length bounded by L .*

Then GCD (Subroutine 3) is a bounded probability algorithm which computes (an slp for) the greatest common divisor between f and g . Its (worst-case) complexity is of order $\mathcal{O}(n \log d + d^4(d^2 + L))$.

Proof.— For $a \in \mathbb{Q}^n$ such that $f(a) \neq 0$ and t an additional variable, we set

$$F(x, t) := t^d f\left(\frac{x-a}{t} + a\right), \quad G(x, t) := t^d g\left(\frac{x-a}{t} + a\right) \in \mathbb{Q}[x][t].$$

Since $f(a) \neq 0$, F is monic —up to the scalar factor $f(a)$ — of degree d in t .

Set H for the gcd of F and G in $\mathbb{Q}(x)[t]$. Since F is monic in t , H belongs to $\mathbb{Q}[x, t]$, and it is easy to check that $\gcd(f, g) = H(x, 1)$ (up to a scalar factor).

The procedure runs as follows: First we observe that if $f = \sum_{\alpha} f_{\alpha}(x-a)^{\alpha}$, then

$$F = \sum_{0 \leq k \leq d} \left(\sum_{|\alpha|=k} f_{\alpha}(x-a)^{\alpha} \right) t^{d-k},$$

(and the same holds with g and G). So the homogeneous components of f and g centered at a turn out to be the coefficients of the monomial expansion of F and G with respect to t . Then, we apply the subresultant algorithm GCD1 to compute the multiple $Q \in \mathbb{Q}[x, t]$ and the superfluous factor q in $\mathbb{Q}[x]$ of their gcd H in $\mathbb{Q}(x)[t]$. Finally, we apply Subroutine 2 to avoid divisions in the computation of $\gcd(f, g) := H(x, 1) = Q(x, 1)/q(x)$.

Let us calculate the size of the sets of points we have to take to ensure that the algorithm has an error probability bounded by $1/4$:

We are going to choose randomly a point $a \in \mathbb{Q}^n$. This same point a will be used in each step we need a random point.

The first condition the point a must satisfy so that the algorithm computes a greatest common divisor of f and g is that $f(a) \neq 0$.

Then we use the point a to compute the degree of G in t . Finally it is used in the subresultant algorithm to compute the degree of the gcd (by deciding whether certain determinants are zero or not). Checking the degree of G involves testing an n -variate polynomial of degree bounded by d (the coefficients of G as a polynomial in $\mathbb{Q}[x][t]$) while checking the degree of the gcd involves testing n -variate polynomials of degree bounded by $2d^2$.

Thus, applying Schwartz bound for the set $[0, \ell]^n$, the conditional probability p of success verifies

$$p \geq \left(1 - \frac{d}{\ell}\right) \left(1 - \frac{d}{\ell}\right) \left(1 - \frac{2d^2}{\ell}\right) \geq 1 - \frac{d + d + 2d^2}{\ell} = 1 - \frac{2d(d+1)}{\ell}.$$

Therefore, taking $\ell := 8d(d+1)$ ensures that the error probability is bounded by $1/4$.

Now let us compute the worst-case complexity of the machine:

The cost of simulating the random choices here is $\mathcal{O}(n \log d)$. Computing the homogeneous components of f and g centered at a and checking the exact degree of G (that is finding the first non-zero coefficient of G with respect to t) can be done within complexity $\mathcal{O}(d^2(d+L))$. In Algorithm GCD1, to compute the degree of the gcd involves computing at most $d+1$ determinants of Sylvester-type matrices of size at most $2d \times 2d$, that is at most $(d+1) \mathcal{O}(d^4)$ operations. Once we know this degree, computing the gcd by means of an adjoint adds $\mathcal{O}(d^4)$ steps. That is, the complexity of computing $Q(x, t)$ (and $q(x)$ which is the non-vanishing determinant) is of order $\mathcal{O}(d^2(d^3 + L))$ while $L(Q(x, t), q(x)) = \mathcal{O}(d^2(d^2 + L))$ since the computation of the degree does not intervene in the length. Applying Subroutine 2 at $q(a)$ which is different from 0 (if not, subroutine GCD1 in line 15 would have returned error) we obtain a final complexity of order $\mathcal{O}(n \log d + d^2(d + L(Q(x, t), q(x)))) = \mathcal{O}(n \log d + d^4(d^2 + L))$. \square

The following procedure (Subroutine 4) computes the quotient —provided it is a polynomial of bounded degree— of two multivariate power series from their graded components up to a certain bound.

Subroutine 4: Power Series Division

procedure PowerSeries($n, m, d, \varphi_m, \dots, \varphi_{m+d}, \psi_m, \dots, \psi_{m+d}$)

$n \in \mathbb{N}$ is the number of variables,
$m \in \mathbb{N}_0$ is the order of the denominator $\varphi \in \mathbb{Q}[[x]]$,
$d \in \mathbb{N}$ is the degree of the quotient $h := \psi/\varphi \in \mathbb{Q}[x]$,
the φ_i 's and ψ_i 's are the graded parts of the power series φ and ψ respectively.
The procedure returns $q := \varphi_m^{d+1} h \in \mathbb{Q}[x]$.

1. $v := \sum_{i=0}^d y^{d-i} z^i \in \mathbb{Q}[y, z]$;
2. $P := (\sum_{i=0}^d \psi_{m+i} t^i) v(\varphi_m, -\sum_{j=1}^d \varphi_{m+j} t^j)$;
3. $(P_0, \dots, P_d) := \text{Expand}(P, t, 0, d)$;
4. $q := \sum_{i=0}^d P_i$;
5. **return**(q);

end.

Let $\varphi = \sum_{\alpha} a_{\alpha} x^{\alpha} \in \mathbb{Q}[[x_1, \dots, x_n]]$ be a power series. For $i \in \mathbb{N}_0$ we denote by $\varphi_i := \sum_{|\alpha|=i} a_{\alpha} x^{\alpha} \in \mathbb{Q}[x_1, \dots, x_n]$ the i -graded component of φ . Also we denote by $\text{ord } \varphi$ the order of φ , that is the least i such that $\varphi_i \neq 0$.

Proposition 1.9 *Let $\varphi, \psi \in \mathbb{Q}[[x_1, \dots, x_n]]$ be power series such that $h := \psi/\varphi \in \mathbb{Q}[x_1, \dots, x_n]$. Assume we are given $m := \text{ord } \varphi$, $d \geq \deg h$, and that the i -graded parts of φ and ψ for $i = m, \dots, m+d$ are encoded by slp's of lengths bounded by L . Then PowerSeries (Subroutine 4) computes $q := \varphi_m^{d+1} h$ within complexity $\mathcal{O}(d^3 L)$.*

Proof.– Set

$$\Phi(x, t) := \varphi(tx) = \sum_{i=0}^{\infty} \varphi_i(x) t^i, \quad \Psi(x, t) := \psi(tx) = \sum_{i=0}^{\infty} \psi_i(x) t^i \in \mathbb{Q}[x][[t]] \hookrightarrow \mathbb{Q}(x)[[t]].$$

Also set $H := h(tx) \in \mathbb{Q}(x)[t]$.

We first observe that $\text{ord } \Phi = m$, and thus $\text{ord } \Psi \geq m$ as $\Psi/\Phi = H \in \mathbb{Q}(x)[t]$ is a polynomial. Hence the following identity holds in $\mathbb{Q}(x)[[t]]$:

$$H = \frac{\Psi}{\Phi} = \frac{\Psi}{t^m} \frac{1}{\varphi_m} \left(1 - \frac{\varphi_m - \Phi/t^m}{\varphi_m}\right)^{-1} = \frac{\Psi}{t^m} \sum_{i=0}^{\infty} \frac{(\varphi_m - \Phi/t^m)^i}{\varphi_m^{i+1}}.$$

Thus, for

$$P := \left(\sum_{i=0}^d \psi_{m+i} t^i \right) \left(\sum_{i=0}^d \varphi_m^{d-i} \left(-\sum_{j=1}^d \varphi_{m+j} t^j \right)^i \right) \in \mathbb{Q}[x][t]$$

we have that $\varphi_m^{d+1} H \equiv P \pmod{t^{d+1}}$. Let $P = \sum_i P_i t^i$ be the monomial expansion of P . Then $\varphi_m^{d+1} H = \sum_{i=0}^d P_i t^i$, as the degree of H with respect to t is bounded by d . Hence $\varphi_m^{d+1} h = \sum_{i=0}^d P_i$.

The stated complexity is obtained as follows:

We compute an slp encoding of $v := \sum_{i=0}^d y^{d-i} z^i$ within complexity $\mathcal{O}(d)$. We compute P as $\sum_{i=0}^d \psi_{m+i} t^i$ times $v(\varphi_m, -\sum_{j=1}^d \varphi_{m+j} t^j)$ within complexity $\mathcal{O}(dL)$. We compute the expansion of P with respect to t up to degree d within complexity $\mathcal{O}(d^3 L)$. Finally we compute q as $\sum_{i=0}^d P_i$. The total complexity is of order $\mathcal{O}(d^3 L)$. \square

Remark 1.10 *In case that, in addition, we are given $b \in \mathbb{Q}^n$ such that $\varphi_m(b) \neq 0$, we can directly apply Subroutine 2 to compute the quotient polynomial h within total complexity $\mathcal{O}(d^5 L)$.*

2 The representation of the Chow form

This section presents an algorithm for the computation of the Chow form of an equidimensional variety from a 0-dimensional fiber and a set of local equations at a neighborhood of this fiber. This is the key step in our general algorithm (see Section 3), although it has independent interest; it shows that the Chow form and the geometric resolution are —up to a polynomial time computation— equivalent representations of a variety (see Subsection 3.1). As a further application, we give a non-trivial upper bound for the length of an slp representation of the Chow form (Corollary 2.11).

In order to state the result, we need the following definitions:

Definition 2.1 *Let $V \subset \mathbb{P}^n$ be an equidimensional variety of dimension r .*

We say that $f_{r+1}, \dots, f_n \in I(V)$ is a system of local equations at $\xi \in V$ if the polynomials f_{r+1}, \dots, f_n generate $I(V)$ at some neighborhood of ξ , i.e. $I(V)_\xi = (f_{r+1}, \dots, f_n)_\xi$ (where the subscript ξ denotes localization at the ideal of the point ξ).

If Z is a subset of V , we say that $f_{r+1}, \dots, f_n \in I(V)$ is a system of local equations (of V) at Z if it is a system of local equations at every $\xi \in Z$.

The existence of a system of local equations at a point $\xi \in V$ implies that $(\mathbb{C}[x]/I(V))_\xi$ is Cohen-Macaulay and thus, by [18, Theorem 18.15], for $f_{r+1}, \dots, f_n \in I(V)$ to be a system of local equations at ξ is equivalent to the fact that the Jacobian matrix of this system has maximal rank $n - r$ at ξ .

Definition 2.2 *Let $Z \subset \mathbb{A}^n$ be a 0-dimensional variety of cardinality D . A geometric resolution of Z consists of an affine linear form $\ell = c_0 + c_1 x_1 + \dots + c_n x_n \in \mathbb{Q}[x_1, \dots, x_n]$ and of polynomials $p \in \mathbb{Q}[t]$ and $v = (v_1, \dots, v_n) \in \mathbb{Q}[t]^n$ such that:*

- *The affine linear form ℓ is a primitive element of Z , that is $\ell(\xi) \neq \ell(\xi')$ for all $\xi \neq \xi'$ in Z .*
- *The polynomial p is monic of degree D and $p(\ell(\xi)) = 0$ for all $\xi \in Z$; that is, p is the minimal polynomial of ℓ over Z .*
- *$\deg v_i \leq D - 1$, $1 \leq i \leq n$, and $Z = \{v(\eta); \eta \in \mathbb{C} / p(\eta) = 0\}$; that is, v parametrizes Z by the zeroes of p .*

Observe that the minimal polynomial p and the parametrization v are uniquely determined by the variety Z and the affine linear form ℓ . We say that (p, v) is the *geometric resolution of Z associated to ℓ* .

In case $Z \subset \mathbb{P}^n$ is a zero-dimensional projective variety which satisfies that none of its points lie in the hyperplane $\{x_0 = 0\}$, Z can be identified to a 0-dimensional *affine* variety Z^{aff} , the image of Z under the rational map $\mathbb{P}^n \dashrightarrow \mathbb{A}^n$ defined by $(x_0 : \dots : x_n) \mapsto (x_1/x_0, \dots, x_n/x_0)$. By a geometric resolution of Z we then understand a geometric resolution —as defined before— of the affine variety $Z^{\text{aff}} \subset \mathbb{A}^n$. In homogeneous coordinates, the definition of geometric resolution states that the homogeneized linear form ℓ^h satisfies $(\ell^h/x_0)(\xi) \neq (\ell^h/x_0)(\xi')$ for all $\xi \neq \xi'$ in Z . The polynomial p is then the minimal monic polynomial of ℓ/x_0 over Z^{aff} . On the other hand, v defines a parametrization $V(p) \rightarrow Z$, $\eta \mapsto (1 : v_1(\eta) : \dots : v_n(\eta))$.

Now, we are able to state the lemma:

Main Lemma 2.3 *Let $V \subset \mathbb{P}^n$ be an equidimensional variety of dimension r and degree D which satisfies Assumption 1.2. Set $Z := V \cap V(x_1, \dots, x_r)$, and let $p \in \mathbb{Q}[t]$ and $v \in \mathbb{Q}[t]^n$ be a given geometric resolution of Z associated to a given affine linear form $\ell \in \mathbb{Q}[x_1, \dots, x_n]$. Let $f_{r+1}, \dots, f_n \in I(V)$ be a system of local equations at Z . Assume that f_{r+1}, \dots, f_n have degrees bounded by d and are encoded by slp's of length bounded by L .*

Then there is a deterministic algorithm (Procedure ChowForm (Subroutine 7) below) which computes Ch_V within complexity $\mathcal{O}(r^8 \log_2(rD) n^7 d^2 D^{11} L)$.

In Subsection 2.3 we present the complete proof of the correctness of the algorithm and its complexity estimate. The algorithm is essentially based on a new Poisson type product formula for the Chow form (see Proposition 2.5 below), which describes the Chow form as a quotient of products of norms of certain polynomials. We interpret this expression as a quotient of two power series, which can be approximated with the aid of a symbolic version of Newton's algorithm. Finally we apply Procedure PowerSeries (Subroutine 4 above) to compute the Chow form from the approximate quotient.

2.1 Newton's algorithm

In this subsection we present a symbolic version of Newton's algorithm for the approximation of roots of equations. Newton's algorithm is nowadays a widely used tool for polynomial equation solving, starting from [26] and [22] as we already mentioned. The situation in the present work is not much different from that in e.g. [33], [29]. Hence we just describe this procedure in order to adapt it to our setting and to estimate its complexity; its correctness follows directly from [33, Section 2] and the arguments therein.

First, we state the situation in which Newton's algorithm is applied.

Let $W \subset \mathbb{A}^r \times \mathbb{A}^n$ be an equidimensional variety of dimension r such that the projection map $\pi : W \rightarrow \mathbb{A}^r$ is dominant, that is, the image $\pi(W)$ is a Zariski dense set.

Set $A := \mathbb{Q}[t_1, \dots, t_r] = \mathbb{Q}[\mathbb{A}^r]$ and let K be its fraction field. Also let $B := \mathbb{Q}[W]$ and set $L := K \otimes_A B$. Then L is a finite K -algebra, and its dimension $D := [L : K]$ —that is the degree of π — equals the maximum cardinality of the *finite* fibers of π ([31, Proposition 1]).

The *norm* $N_\pi(h) \in K$ of a polynomial $h \in A[x_1, \dots, x_n]$ is defined as the determinant of the K -linear map $L \rightarrow L$ defined by $f \mapsto h f$.

Let $I(W)^e$ denote the extension of the ideal $I(W)$ to the polynomial ring $K[x_1, \dots, x_n]$, and set $W^e := V(I(W)^e) \subset \mathbb{A}^n(\bar{K})$, which is a 0-dimensional variety of degree D . Then

$$N_\pi(h) = \prod_{\gamma \in W^e} h(\gamma).$$

We also denote this norm by $N_{W^e}(h)$ when the projection map is clear from the context.

In various settings, we will be given a polynomial h and an equidimensional variety W and our aim will be to compute an approximation of $N_{W^e}(h)$. The input of the procedure will be the polynomial h , a geometric resolution of a 0-dimensional fiber of π and local equations at this fiber.

Let $F_1, \dots, F_n \in I(W) \subset A[x_1, \dots, x_n]$ and set $F := (F_1, \dots, F_n)$. Let

$$\mathcal{J}_F := \left(\frac{\partial F_i}{\partial x_j} \right)_{1 \leq i, j \leq n} \in A[x_1, \dots, x_n]^{n \times n}$$

be the Jacobian matrix of F with respect to the variables x_1, \dots, x_n , and let $\Delta_F := |\mathcal{J}_F| \in A[x_1, \dots, x_n]$ be its Jacobian determinant.

Let $Z \subset \mathbb{A}^n$ be such that $\pi^{-1}(0) = \{0\} \times Z$. We assume that Z is a 0-dimensional variety of cardinality D and that \mathcal{J}_F is non-singular at $\pi^{-1}(0)$, that is, $\Delta_F(0, \xi) \neq 0$ for all $\xi \in Z$. Observe that this means that $F(0, x)$ is a system of local equations at Z .

Under our assumptions, the implicit function theorem implies that the elements in W^e can also be considered as power series vectors: For $\xi \in Z$, since $\Delta_F(0, \xi) \neq 0$, there exists a unique $\gamma_\xi \in \mathbb{C}[[t_1, \dots, t_r]]^n$ such that

$$\gamma_\xi(0) = \xi \quad \text{and} \quad F(t_1, \dots, t_r, \gamma_\xi) = 0.$$

(See, for example, [6, Chapter 3, Section 4.5, Corollary 2].) It follows that $f(t_1, \dots, t_r, \gamma_\xi) = 0$ for all $f \in I(W)$ as F is a system of local equations at ξ , and so this also holds for all $f \in I(W^e) = I(W)^e$. Hence $\gamma_\xi \in W^e$ and, as $\#Z = \#W^e = D$, we conclude that the correspondence

$$Z \rightarrow W^e \quad , \quad \xi \mapsto \gamma_\xi$$

is one-to-one. In particular, since $N_{W^e}(h)$ is the determinant of a matrix in $\mathbb{Q}(t_1, \dots, t_r)$, $N_{W^e}(h) \in \mathbb{C}[[t_1, \dots, t_r]] \cap \mathbb{Q}(t_1, \dots, t_r) \subset \mathbb{Q}[[t_1, \dots, t_r]]$.

The given data —the description of the fiber and its local equations— suffices to determine W^e uniquely, and in particular allows us to compute a rational function q which approximates the norm $N_{W^e}(h)$ to any given precision κ (we understand by this that both Taylor expansions coincide up to degree κ , that is $N_{W^e}(h) \equiv q \pmod{(t_1, \dots, t_r)^{\kappa+1}}$). The rational function q can be obtained by a procedure based on an iterative application of the Newton operator. This operator, defined as

$$\mathcal{N}_F^t := x^t - \mathcal{J}_F(x)^{-1} F(x)^t \in K(x)^{n \times 1},$$

enables us to approximate the points in W^e from the points in the fiber Z . If we set $\mathcal{N}_F^{(m)} \in K(x)^{1 \times n}$ for the m -times iteration of \mathcal{N}_F , then, for every $\xi \in Z$,

$$\mathcal{N}_F^{(m)}(\xi) \equiv \gamma_\xi \pmod{(t_1, \dots, t_r)^{2^m}}$$

(see [33, Section 2]).

Procedure NumDenNewton (Subroutine 5) computes polynomials $g_1^{(m)}, \dots, g_n^{(m)}, f_0^{(m)}$ in $\mathbb{Q}[t_1, \dots, t_r]$ such that

$$\mathcal{N}_F^{(m)} = (g_1^{(m)} / f_0^{(m)}, \dots, g_n^{(m)} / f_0^{(m)}).$$

Herein, $\text{Homog}(f, d)$ is a procedure which computes the homogeneization of the polynomial f up to degree $d \geq \deg(f)$, $\text{JacobianMatrix}(F, x)$ is a procedure which constructs the Jacobian matrix with respect to the variables x associated to the system of polynomials F and $\text{Adjoint}(M)$ is a procedure which computes the adjoint of the matrix M . For the correctness and complexity of the whole procedure, see [22, Lemma 30].

Subroutine 5: Computation of numerators and denominators for the Newton operator

```

procedure NumDenNewton( $F, n, x, d, m$ )

#  $F_1, \dots, F_n \in A[x_1, \dots, x_n]$  such that  $J_F(x) \neq 0$ ;
#  $n$  is the number of dependent variables  $x$ ,
#  $d$  is an upper bound for the degrees of the polynomials  $F_1, \dots, F_n$ ,
#  $m$  is the number of iterations to be computed.
# The procedure returns  $g_1^{(m)}, \dots, g_n^{(m)}, f_0^{(m)}$  such that  $\mathcal{N}_F(x)^{(m)} = (g_1^{(m)}/f_0^{(m)}, \dots, g_n^{(m)}/f_0^{(m)})$ .

1.  $\mathcal{J}_F := \text{JacobianMatrix}(F, x)$ ;
2.  $\Delta_F := |(\mathcal{J}_F)|$ ;
3.  $A := \text{Adjoint}(\mathcal{J}_F)$ ;
4.  $\nu := nd + 1$ ;
5. for  $i$  from 1 to  $n$  do
6.    $g_i^{(1)} := \Delta_F x_i - \sum_{j=1}^n A_{ij} f_j$ ;
7.    $G_i := \text{Homog}(g_i^{(1)}, \nu)$ ;
8. od;
9.  $f_0^{(1)} := \Delta_F$ ;
10.  $F_0 := \text{Homog}(\Delta_F, \nu)$ ;
11. for  $k$  from 2 to  $m$  do
12.   for  $i$  from 1 to  $n$  do
13.      $g_i^{(k)} := G_i(f_0^{(k-1)}, g_1^{(k-1)}, \dots, g_n^{(k-1)})$ ;
14.   od;
15.    $f_0^{(k)} := F_0(f_0^{(k-1)}, g_1^{(k-1)}, \dots, g_n^{(k-1)})$ ;
16. od;
17. return( $g_1^{(m)}, \dots, g_n^{(m)}, f_0^{(m)}$ );

end.

```

We summarize the procedure that approximates the norm of a given polynomial h in Procedure Norm (Subroutine 6). Herein, CompanionMatrix is the procedure which constructs the companion matrix of a given univariate polynomial. We keep our notations.

Lemma 2.4 *Let notations be as before. Assume that $h, F_1, \dots, F_n \in A[x_1, \dots, x_n]$ are polynomials encoded by slp's of length bounded by L such that $\deg h \leq \delta$ and $\deg(F_i) \leq d$, $1 \leq i \leq n$. Then Norm (Subroutine 6) computes $f, g \in A$ with $f(0) \neq 0$ such that g/f approximates $\mathbb{N}_{W^\epsilon}(h)$ with precision κ , within complexity $\mathcal{O}((\log_2 \kappa) n^7 \delta^2 d^2 D^4 L)$.*

Proof.– For the correctness of the algorithm we refer to [33, Section 2] and the arguments given there. Now, we estimate its complexity:

First, the complexity of Subroutine 5 applied to our situation is of order $\mathcal{O}((\log_2 \kappa) n^7 d^2 L)$ (see [22, Lemma 30] and its proof).

Then, the algorithm computes the matrices $v_j(M_p)$ ($1 \leq j \leq n$) with complexity of order $\mathcal{O}(nD^3)$ (note that, as the companion matrix is *very* sparse, the multiplication by M_p can be done with complexity $\mathcal{O}(D^2)$). Now, the matrices $M_i := g_i(v(M_p))$ ($1 \leq i \leq n$) and $M_0 := f_0(v(M_p))$

Subroutine 6: Approximation of the Norm

procedure Norm($h, \delta, n, x, p, v, F, d, \kappa$)

$h \in A[x_1, \dots, x_n]$ is the polynomial whose norm we want to approximate,
 # δ is an upper bound for the degree of h ,
 # $n \in \mathbb{N}$ is the number of dependent variables x ,
 # $p \in \mathbb{Q}[t]$, $v \in \mathbb{Q}[t]^n$ is a given geometric resolution of Z ,
 # $F = (F_1, \dots, F_n)$ is a vector of polynomials in $I(W)$ such that $\Delta_F(0, \xi) \neq 0$ for all $\xi \in Z$,
 # d is an upper bound for the degrees of the polynomials F_1, \dots, F_n ,
 # $\kappa \in \mathbb{N}$ is the desired level of precision.
 # The procedure returns $f, g \in A$ with $f(0) \neq 0$ such that g/f approximates the norm $N_{W^e}(h)$
 # with precision κ .

1. $m := \lceil \log_2(\kappa + 1) \rceil$;
2. $(g_1, \dots, g_n, f_0) := \text{NumDenNewton}(F, n, x, d, m)$;
3. $M_p := \text{CompanionMatrix}(p)$;
4. **for** i **from** 1 **to** n **do**
5. $M_i := g_i(v(M_p))$;
6. **od**;
7. $M_0 := f_0(v(M_p))$;
8. $H := \text{Homog}(h, \delta)$;
9. $M := H(M_0, M_1, \dots, M_n)$;
10. $f := |M_0|^\delta$;
11. $g := |M|$;
12. **return**(f, g);

end.

are obtained within complexity $\mathcal{O}((\log_2 \kappa) n^7 d^2 D^3 L)$. As h is encoded by an slp of length L , its homogeneous components up to degree δ are encoded by slp's of length $\mathcal{O}(\delta^2 L)$. Therefore, the complexity of the computation of M is of order $\mathcal{O}(\delta^2 L D^3 + (\log_2 \kappa) n^7 d^2 D^3 L)$.

Finally, f and g can be computed within complexity $\mathcal{O}(D^4 + (\log_2 \delta) D^3 + (\log_2 \kappa) n^7 d^2 D^3 L)$ and $\mathcal{O}(D^4 + \delta^2 D^3 + (\log_2 \kappa) n^7 d^2 D^3 L)$ respectively. \square

2.2 A product formula

In what follows, we establish a product formula for the Chow form of an affine variety. This formula is an analogue of the classical Poisson formula for the resultant ([15, Chapter 3, Theorem 3.4]). It describes, under certain conditions, the Chow form in a recursive manner.

Let $V \subset \mathbb{A}^n$ be an equidimensional *affine* variety of dimension r and degree D which satisfies Assumption 1.2. Let U_0, \dots, U_r be $r + 1$ groups of $n + 1$ variables each and let $L_i := U_{i0} + U_{i1} x_1 + \dots + U_{in} x_n$, $0 \leq i \leq r$, be the affine linear forms associated to these groups of variables. Set $K := \mathbb{Q}(U_0, \dots, U_{r-1})$ and let $I(V)^e$ denote the extension of the ideal of V to the polynomial ring $K[x_1, \dots, x_n]$ (or to any other ring extension of $\mathbb{Q}[x_1, \dots, x_n]$ which will be clear from the context). We also set $V^0 := V(I(V)^e) \cap V(L_0, \dots, L_{r-1}) \subset \mathbb{A}^n(\overline{K})$, which is a 0-dimensional variety of degree D .

For $0 \leq i \leq r$, we set $V_i := V \cap V(x_{i+1}, \dots, x_r) \subset \mathbb{A}^n$, which is an equidimensional variety of dimension i and degree D as V satisfies Assumption 1.2. Observe that these varieties satisfy Assumption 1.2 as well.

Let $K_i := \mathbb{Q}(U_0, \dots, U_{i-1}) \hookrightarrow K$ and set

$$V_i^0 := V(I(V_i)^e) \cap V(L_0, \dots, L_{i-1}) \subset \mathbb{A}^n(\overline{K_i}).$$

Observe that V_i^0 is also a 0-dimensional variety of degree D .

Under these notations we have that $V_0^0 = V_0$, $K_r = K$ and $V_r^0 = V^0$.

Proposition 2.5 (*Product formula*) *Let $V \subset \mathbb{A}^n$ be an equidimensional variety of dimension r which satisfies Assumption 1.2. Let notations be as in the previous paragraph. Then*

$$\mathcal{C}h_V(U_0, \dots, U_r) = \frac{\prod_{i=0}^r \mathcal{C}h_{V_i^0}(U_i)}{\prod_{i=1}^r \mathcal{C}h_{V_i^0}(e_i)} \in \mathbb{Q}(U_0, \dots, U_{r-1})[U_r].$$

The proof of this fact is based on the following lemma:

Lemma 2.6 *Let $V \subset \mathbb{A}^n$ be an equidimensional variety of dimension r . Let $\mathcal{F}_V \in \mathbb{Q}[U_0, \dots, U_r]$ and $\mathcal{F}_{V^0} \in K[U_r]$ be Chow forms of V and V^0 respectively. Then there exists $\lambda \in K^*$ such that*

$$\mathcal{F}_V = \lambda \mathcal{F}_{V^0}.$$

Proof.— As before, we denote by $I(V)^e$ the extension of the ideal $I(V)$ to a ring extension of $\mathbb{Q}[x_0, \dots, x_n]$ which will be clear from the context. Let U_i^{lin} , $0 \leq i \leq r$, denote the group of n variables $U_i \setminus \{U_{i0}\}$. We consider the map

$$\mathbb{Q}[U_0, \dots, U_r][x_1, \dots, x_n]/(I(V)^e + (L_0, \dots, L_r)) \rightarrow \mathbb{Q}[U_0^{\text{lin}}, \dots, U_r^{\text{lin}}][x_1, \dots, x_n]/I(V)^e$$

defined by $U_{i0} \mapsto -(U_{i1}x_1 + \dots + U_{in}x_n)$, $U_{ij} \mapsto U_{ij}$ and $x_j \mapsto x_j$ for $0 \leq i \leq r$, $1 \leq j \leq n$.

As it is a ring isomorphism, $I(V)^e + (L_0, \dots, L_r)$ is a radical ideal. Following notations in Subsection 1.1, it follows that this ideal coincides with the defining ideal of the incidence variety Φ_V and, therefore,

$$(\mathcal{F}_V) = (I(V)^e + (L_0, \dots, L_r)) \cap \mathbb{Q}[U_0, \dots, U_r].$$

Similarly

$$(\mathcal{F}_{V^0}) = (I(V^0)^e + (L_r)) \cap K[U_r].$$

We have that $I(V)^e + (L_0, \dots, L_r) \subset I(V^0)^e + (L_r)$ and so $(\mathcal{F}_V) \subset (\mathcal{F}_{V^0})$, that is, there exists $\lambda \in K[U_r] \setminus \{0\}$ such that $\mathcal{F}_V = \lambda \mathcal{F}_{V^0}$. As $\deg_{U_r} \mathcal{F}_V = \deg V = \deg \mathcal{F}_{V^0}$, λ is an element in K^* . \square

Proof of Proposition 2.5.— Let $1 \leq i \leq r$. From Lemma 2.6, there exists $\lambda_i \in K_i^*$ such that

$$\mathcal{C}h_{V_i}(U_0, \dots, U_i) = \lambda_i \mathcal{C}h_{V_i^0}(U_i). \quad (1)$$

Hence $\mathcal{C}h_{V_i}(U_0, \dots, U_{i-1}, e_i) = \lambda_i \mathcal{C}h_{V_i^0}(e_i)$.

Now, it is easy to see that $\mathcal{C}h_{V_{i-1}}(U_0, \dots, U_{i-1})$ divides $\mathcal{C}h_{V_i}(U_0, \dots, U_{i-1}, e_i)$. From Assumption 1.2, it follows that $\deg V_{i-1} = \deg V_i = D$ and, therefore, both polynomials have the same degree. Moreover, the normalization imposed to both Chow forms implies that they coincide. So

$$\mathcal{C}h_{V_{i-1}}(U_0, \dots, U_{i-1}) = \lambda_i \mathcal{C}h_{V_i^0}(e_i). \quad (2)$$

From Identities (1) and (2) we deduce that

$$\frac{\mathcal{C}h_{V_i}(U_0, \dots, U_i)}{\mathcal{C}h_{V_{i-1}}(U_0, \dots, U_{i-1})} = \frac{\mathcal{C}h_{V_i^0}(U_i)}{\mathcal{C}h_{V_i^0}(e_i)}. \quad (3)$$

Multiplying these identities for $i = 1, \dots, r$ we obtain

$$\frac{\mathcal{C}h_V(U_0, \dots, U_r)}{\mathcal{C}h_{V_0}(U_0)} = \prod_{i=1}^r \frac{\mathcal{C}h_{V_i}(U_0, \dots, U_i)}{\mathcal{C}h_{V_{i-1}}(U_0, \dots, U_{i-1})} = \prod_{i=1}^r \frac{\mathcal{C}h_{V_i^0}(U_i)}{\mathcal{C}h_{V_i^0}(e_i)}$$

which gives the formula stated in Proposition 2.5. \square

Observe that $\mathcal{C}h_{V_i^0}(U_i) = \prod_{\gamma \in V_i^0} L_i(\gamma) = N_{V_i^0}(L_i)$ and $\mathcal{C}h_{V_i^0}(e_i) = \prod_{\gamma \in V_i^0} x_i(\gamma) = N_{V_i^0}(x_i)$. Thus, Proposition 2.5 can be restated as:

$$\mathcal{C}h_V(U_0, \dots, U_r) = \frac{\prod_{i=0}^r N_{V_i^0}(L_i)}{\prod_{i=1}^r N_{V_i^0}(x_i)} \in \mathbb{Q}(U_0, \dots, U_{r-1})[U_r]. \quad (4)$$

Also, observe that Lemma 2.6 implies the following:

Remark 2.7 *The Chow form $\mathcal{C}h_V$ is the numerator of $N_{V^0}(L_r)$.*

Unfortunately this norm is a rational function, due to the fact that the map π_r is not finite but just dominant. The product formula is the tool which enables us to overcome this difficulty, as it gives an expression for $\mathcal{C}h_V$ without any extraneous denominator.

Identity (4) enables us to compute $\mathcal{C}h_V$ as a quotient of power series. To do so, we state a technical lemma first, that we prove here for lack of reference.

Lemma 2.8 *Let $V \subset \mathbb{A}^n$ be an equidimensional variety of dimension r which satisfies Assumption 1.2. Assume that V is Cohen-Macaulay at every point of $Z := V \cap V(x_1, \dots, x_r)$. Then, the ideal $I(V) + (x_1, \dots, x_r) \subset \mathbb{C}[x_1, \dots, x_n]$ is radical.*

Proof.— Let \bar{V} denote the projective closure of $V \subset \mathbb{A}^n \hookrightarrow \mathbb{P}^n$. Let $Z := V \cap V(x_1, \dots, x_r)$. The fact that $\#Z = \deg V$ implies that $Z = \bar{V} \cap V(x_1, \dots, x_r)$.

Take $\xi \in Z$ and let Q_ξ be the primary component of the ideal $I(V) + (x_1, \dots, x_r) \subset \mathbb{C}[x_1, \dots, x_n]$ which corresponds to ξ .

We consider the length $\ell(\bar{V}, V(x_1, \dots, x_r); \xi)$ which under our assumptions can be defined as

$$\ell(\bar{V}, V(x_1, \dots, x_r); \xi) = \dim_{\mathbb{C}} \mathbb{C}[x_1, \dots, x_n]/Q_\xi. \quad (5)$$

By a suitable version of Bézout's theorem (see [56, Proposition 3.30])

$$\sum_{\xi \in Z} \ell(\bar{V}, V(x_1, \dots, x_r); \xi) \leq \deg V.$$

On the other hand, as $\ell(\bar{V}, V(x_1, \dots, x_r); \xi)$ is a positive integer for each $\xi \in Z$, and as $\#Z = \deg V$, it follows that

$$\sum_{\xi \in Z} \ell(\bar{V}, V(x_1, \dots, x_r); \xi) \geq \deg V.$$

Then $\ell(\bar{V}, V(x_1, \dots, x_r); \xi) = 1$ for all $\xi \in Z$, and so (5) implies that $Q_\xi = (x_1 - \xi_1, \dots, x_n - \xi_n)$ which is a prime ideal.

As $I := I(V) + (x_1, \dots, x_r)$ is 0-dimensional, it has no embedded components. Hence $I = \cap_{\xi} Q_\xi$ is a radical ideal. \square

The following corollary shows that the coordinates of all the points in V_i^0 belong to the subring $\mathbb{C}[[U_0 - e_1, \dots, U_{i-1} - e_i]] \cap \bar{K}_i$, and that there is a one-to-one correspondence between the points of $Z := V \cap V(x_1, \dots, x_r)$ and the points of V_i^0 .

Corollary 2.9 *Let notations and assumptions be as in Lemma 2.8 and before. Let $0 \leq i \leq r$ and $\xi \in Z$. Then there exists a unique $\gamma_\xi^{(i)} \in \mathbb{C}[[U_0 - e_1, \dots, U_{i-1} - e_i]]^n$ such that $\gamma_\xi^{(i)} \in V_i^0$ and $\gamma_\xi^{(i)}(e_1, \dots, e_i) = \xi$.*

Proof.– Suppose $I(V)$ is generated by the polynomials h_1, \dots, h_t . Since we are in the conditions of the previous lemma, the Jacobian criterion [18, Theorem 18.15] implies that the Jacobian matrix associated to the generators $h_1, \dots, h_t, x_1, \dots, x_r$ of the ideal $I(V) + (x_1, \dots, x_r)$ has maximal rank n at ξ . In other words, there are n polynomials g_1, \dots, g_n among $h_1, \dots, h_t, x_1, \dots, x_r$ such that the associated Jacobian determinant is non-zero. Now, as the rank of the Jacobian matrix of h_1, \dots, h_t at ξ is bounded by the codimension $n - r$ of V at ξ , we can assume w.l.o.g. that $g_1 := x_1, \dots, g_r := x_r$. Let

$$\Delta := \left| \left(\frac{\partial g_i}{\partial x_j} \right)_{r+1 \leq i, j \leq n} \right|$$

be the Jacobian determinant of g_{r+1}, \dots, g_n with respect to the variables x_{r+1}, \dots, x_n . Then $\Delta(\xi) \neq 0$ since Δ coincides with the Jacobian determinant of the system g_1, \dots, g_n .

On the other hand, let $\Delta_i \in K_i[x_1, \dots, x_n]$ denote the Jacobian determinant of the system $L_0, \dots, L_{i-1}, x_{i+1}, \dots, x_r, g_{r+1}, \dots, g_n$. An easy verification shows that $\Delta_i(e_1, \dots, e_{i-1})(\xi) = \Delta(\xi) \neq 0$. The statement follows from the implicit function theorem (see Section 2.1). \square

Now, set

$$\Psi := \prod_{i=0}^r N_{V_i^0}(L_i) \in K[U_r], \quad \Phi := \prod_{i=1}^r N_{V_i^0}(x_i) \in K^*,$$

so that, by Identity (4), $\mathcal{C}h_V := \Psi/\Phi$.

From Corollary 2.9, $\Psi \in \mathbb{Q}[[U_0 - e_1, \dots, U_{r-1} - e_r]][U_r]$ and $\Phi \in \mathbb{Q}[[U_0 - e_1, \dots, U_{r-1} - e_r]]$.

The following lemma gives the order of the denominator Φ at $E := (e_1, \dots, e_r) \in \mathbb{A}^{r(n+1)}$ together with its graded component of lowest degree:

Lemma 2.10 *Let notations be as in the previous paragraph and let $D := \deg V$. Then $\text{ord}_E(\Phi) = rD$ and its graded component of degree rD is*

$$\Phi_{rD} = \pm \prod_{i=1}^r \mathcal{C}h_{V_0}(U_{i-1}).$$

Proof.– Clearly, $\text{ord}_E(\Phi) = \sum_{i=1}^r \text{ord}_E(N_{V_i^0}(x_i))$.

Let $1 \leq i \leq n$. Recall that $N_{V_i^0}(x_i) = \mathcal{C}h_{V_i^0}(e_i)$. From Identity (3) in the proof of Proposition 2.5 we have:

$$\mathcal{C}h_{V_i^0}(e_i) \mathcal{C}h_{V_i}(U_0, \dots, U_i) = \mathcal{C}h_{V_i^0}(U_i) \mathcal{C}h_{V_{i-1}}(U_0, \dots, U_{i-1}).$$

As $\mathcal{C}h_{V_i^0}(e_0) = 1$, then $\mathcal{C}h_{V_i^0}(e_i) \mathcal{C}h_{V_i}(U_0, \dots, U_{i-1}, e_0) = \mathcal{C}h_{V_{i-1}}(U_0, \dots, U_{i-1})$.

We also have that

$$\mathcal{C}h_{V_i}(e_1, \dots, e_i, e_0) = \pm \mathcal{C}h_{V_i}(e_0, e_1, \dots, e_i) = \pm 1.$$

This shows that $\mathcal{C}h_{V_i}(U_0, \dots, U_{i-1}, e_0)$ is invertible in $\mathbb{Q}[[U_0 - e_1, \dots, U_{i-1} - e_i]]$ and, therefore, if $m := \text{ord}_E(\mathcal{C}h_{V_i^0}(e_i))$,

$$\mathcal{C}h_{V_i^0}(e_i) \equiv \pm \mathcal{C}h_{V_{i-1}}(U_0, \dots, U_{i-1}) \pmod{(U_0 - e_1, \dots, U_{i-1} - e_i)^{m+1}}.$$

By Lemma 2.6, there exists $\lambda_{i-1} \in \mathbb{Q}(U_0, \dots, U_{i-2}) \setminus \{0\}$ such that

$$\mathcal{C}h_{V_{i-1}}(U_0, \dots, U_{i-1}) = \lambda_{i-1} \mathcal{C}h_{V_{i-1}^0}(U_{i-1}).$$

As $\mathcal{C}h_{V_{i-1}^0}(U_{i-1})$ is a homogeneous polynomial of degree D in the group of variables U_{i-1} and does not depend on U_{i-1-i} , it is also homogeneous as a polynomial expanded in $U_{i-1} - e_i$. Then, the

order of $\mathcal{C}h_{V_{i-1}}$ at e_i with respect to the group of variables U_{i-1} equals D . On the other hand, we have that $\mathcal{C}h_{V_{i-1}}(e_1, \dots, e_{i-1}, U_{i-1}) = \pm \mathcal{C}h_{V_0}(U_{i-1}) \neq 0$. This implies that the series $\mathcal{C}h_{V_{i-1}}$ in $\mathbb{Q}[[U_0 - e_1, \dots, U_{i-1} - e_i]]$ has a term of degree D depending only on the group of variables $U_{i-1} - e_i$. We conclude that $m = D$ and

$$(\mathbb{N}_{V_i^0}(x_i))_D = (\mathcal{C}h_{V_i^0}(e_i))_D = \pm (\mathcal{C}h_{V_{i-1}}(U_0, \dots, U_{i-1}))_D = \pm \mathcal{C}h_{V_0}(U_{i-1}).$$

Therefore, $\text{ord}_E(\Phi) = \sum_{i=1}^r \text{ord}_E(\mathbb{N}_{V_i^0}(x_i)) = rD$ and the graded part of lowest degree of Φ is $\Phi_{rD} = \prod_{i=1}^r (\mathbb{N}_{V_i^0}(x_i))_D = \pm \prod_{i=1}^r \mathcal{C}h_{V_0}(U_{i-1})$. \square

2.3 The algorithm

Here, we are going to put the previous results together in order to obtain the algorithm underlying Main Lemma 2.3 and to estimate its complexity.

Let notations be as in Main Lemma 2.3. As we have already noted, the imposed conditions imply that both V and Z have no component in the hyperplane $\{x_0 = 0\}$. Hence V equals the projective closure of its affine part $V_{x_0} := V \setminus \{x_0 = 0\}$ and so both their Chow forms coincide. Hence we concentrate w.l.o.g. on the affine case. We use affine coordinates and keep the notation of the previous subsection.

From Identity (4), we have that

$$\mathcal{C}h_V = \frac{\prod_{i=0}^r \mathbb{N}_{V_i^0}(L_i)}{\prod_{i=1}^r \mathbb{N}_{V_i^0}(x_i)}.$$

Now, we approximate the norms appearing in this formula.

Set

$$\mathcal{V}_i := V(I(V_i)^e) \cap V(L_0, \dots, L_{i-1}) \subset \mathbb{A}^{i(n+1)} \times \mathbb{A}^n.$$

The map $\pi_i : \mathcal{V}_i \rightarrow \mathbb{A}^{i(n+1)}$ defined by $(U, x) \mapsto U$ is dominant of degree $D := \deg V$. We set $Z := V_0 = V \cap V(x_1, \dots, x_r) \subset \mathbb{A}^n$ and let $E_i := (e_1, \dots, e_i) \in \mathbb{A}^{i(n+1)}$. Then

$$\mathcal{Z}_i := \pi_i^{-1}(E_i) = \{E_i\} \times Z,$$

and so this fiber is a 0-dimensional variety of cardinality D . Furthermore, it is easy to check that

$$L_0, \dots, L_{i-1}, x_{i+1}, \dots, x_r, f_{r+1}, \dots, f_n \in \mathbb{Q}[U_0, \dots, U_{i-1}][x_1, \dots, x_n]$$

is a system of local equations of \mathcal{V}_i at \mathcal{Z}_i .

Since by definition, $\mathbb{N}_{\pi_i}(x_i)$ and $\mathbb{N}_{\pi_i}(L_i)$ coincide with $\mathbb{N}_{V_i^0}(x_i) \in \mathbb{Q}[[U_0 - e_1, \dots, U_{i-1} - e_i]]$ and $\mathbb{N}_{V_i^0}(L_i) \in \mathbb{Q}[[U_0 - e_1, \dots, U_i - e_{i+1}]]$ respectively, we can compute any approximation of the latter applying Procedure Norm (Subroutine 6) modulo a change of variables $(U_0, \dots, U_i) \mapsto (\tilde{U}_0 + e_1, \dots, \tilde{U}_i + e_{i+1})$ (in order to center the series at 0).

We multiply the computed approximations for $0 \leq i \leq r$ to obtain rational functions ψ and φ which approximate the power series

$$\Psi := \prod_{i=0}^r \mathbb{N}_{V_i^0}(L_i) \in \mathbb{Q}[[U_0 - e_1, \dots, U_{r-1} - e_r]][U_r], \quad \Phi := \prod_{i=1}^r \mathbb{N}_{V_i^0}(x_i) \in \mathbb{Q}[[U_0 - e_1, \dots, U_{r-1} - e_r]]^*$$

respectively.

From these approximations, we compute the graded parts of Φ and Ψ of degrees between rD and $(2r+1)D$ centered at $(E, 0) \in \mathbb{A}^{(r+1)(n+1)}$ (where $E := (e_1, \dots, e_r) \in \mathbb{A}^{r(n+1)}$) by applying Procedure GradedParts (see Subsection 1.4).

Subroutine 7: Chow form from a fiber and local equations

procedure ChowForm(n, x, r, D, p, v, f, d)

n is the number of variables $x := (x_1, \dots, x_n)$,
r, D are the dimension and the degree of V respectively,
$p \in \mathbb{Q}[t]$, $v \in \mathbb{Q}[t]^n$ is a given geometric resolution of the fiber Z ,
$f = (f_{r+1}, \dots, f_n) \in \mathbb{Q}[x_1, \dots, x_n]^{n-r}$ is a system of local equations of V at Z of degrees
bounded by d .

The procedure returns the normalized Chow form $\mathcal{C}h_V$.

1. **for** i **from** 1 **to** r **do**

2. $(\varphi_i^{(1)}, \varphi_i^{(2)}) := \text{Norm}(x_i, 1, n, x, p, v, L_0, \dots, L_{i-1}, x_{i+1}, \dots, x_r, f_{r+1}, \dots, f_n, d, (2r+1)D)$;

3. **od**;

4. **for** i **from** 0 **to** r **do**

5. $(\psi_i^{(1)}, \psi_i^{(2)}) := \text{Norm}(L_i, 1, n, x, p, v, L_0, \dots, L_{i-1}, x_{i+1}, \dots, x_r, f_{r+1}, \dots, f_n, d, (2r+1)D)$;

6. **od**;

7. $\varphi^{(1)} := \prod_{i=1}^r \varphi_i^{(1)}$, $\varphi^{(2)} := \prod_{i=1}^r \varphi_i^{(2)}$;

8. $\psi^{(1)} := \prod_{i=0}^r \psi_i^{(1)}$, $\psi^{(2)} := \prod_{i=0}^r \psi_i^{(2)}$;

9. $(\Phi_0, \dots, \Phi_{(2r+1)D}) := \text{GradedParts}(\varphi^{(1)}, \varphi^{(2)}, (e_1, \dots, e_r), (2r+1)D)$;

10. $(\Psi_0, \dots, \Psi_{(2r+1)D}) := \text{GradedParts}(\psi^{(1)}, \psi^{(2)}, (e_1, \dots, e_r, 0), (2r+1)D)$;

11. $Q := \text{PowerSeries}((r+1)(n+1), rD, (r+1)D, \Phi_{rD}, \dots, \Phi_{(2r+1)D}, \Psi_{rD}, \dots, \Psi_{(2r+1)D})$;

12. $\mathcal{C}h_V := \text{PolynomialDivision}(Q, \Phi_{rD}^{(r+1)D+1}, (r+1)D, (e_0, \dots, e_0))$;

13. **return**($\mathcal{C}h_V$);

end.

By Lemma 2.10, we have that $\text{ord}_{(E,0)}(\Phi) = \text{ord}_E(\Phi) = rD$. We also have $\deg \mathcal{C}h_V = (r+1)D$. We use this information together with the obtained graded parts in order to apply Procedure PowerSeries (Subroutine 4). This yields a polynomial $Q \in \mathbb{Q}[U_0, \dots, U_r]$ such that

$$Q = \Phi_{rD}^{(r+1)D+1} \mathcal{C}h_V.$$

Again, from Lemma 2.10, the denominator $\Phi_{rD}^{(r+1)D+1}$ does not vanish at $E^0 := (e_0, \dots, e_0) \in \mathbb{A}^{r(n+1)}$. We apply Procedure PolynomialDivision (Subroutine 2) to the polynomials Q and $\Phi_{rD}^{(r+1)D+1}$ and the point E^0 .

We summarize this procedure in Procedure ChowForm (Subroutine 7) which computes the Chow form of an affine equidimensional variety V satisfying Assumption 1.2.

Proof of Main Lemma 2.3. – As we have already observed, we may suppose w.l.o.g that V is an affine variety and that the polynomials f_{r+1}, \dots, f_n are in $\mathbb{Q}[x_1, \dots, x_n]$.

We apply Procedure ChowForm (Subroutine 7) to V in order to compute its normalized Chow form. The correctness of this procedure follows from our previous analysis. The announced complexity is a consequence of the complexity of the subroutines we call during this procedure:

- By Lemma 2.4, the complexity of lines 1 to 6 is of order $\mathcal{O}(r \log_2(rD) n^7 d^2 D^4 L)$. The products in lines 7 and 8 do not change this estimate.

- The computation of the graded parts in lines 9 and 10 has complexity $\mathcal{O}(r^3 \log_2(rD) n^7 d^2 D^6 L)$.
- Finally, the subroutines PowerSeries and PolynomialDivision in lines 11 and 12 add complexity $\mathcal{O}(r^8 \log_2(rD) n^7 d^2 D^{11} L)$.

We conclude that the overall complexity is $\mathcal{O}(r^8 \log_2(rD) n^7 d^2 D^{11} L)$. \square

We directly derive the following estimate for the length of an slp representation of the Chow form of an equidimensional variety:

Corollary 2.11 *Let $V \subset \mathbb{P}^n$ be an equidimensional variety of dimension r and degree D . Let $f_{r+1}, \dots, f_n \in I(V)$ be a system of local equations at a dense open subset of V , encoded by slp's of length bounded by L . Then, if $d := \max\{\deg(f_i) : r+1 \leq i \leq n\}$, we have*

$$L(\mathcal{F}_V) \leq \mathcal{O}(r^8 \log_2(rD) n^7 d^2 D^{11} L).$$

Proof.— Let $\ell_1, \dots, \ell_r \in \mathbb{Q}[x_0, \dots, x_n]$ be linear forms such that $Z := V \cap V(\ell_1, \dots, \ell_r)$ is a 0-dimensional variety of cardinality D . We can choose these linear forms so that Z lies in the dense open subset where f_{r+1}, \dots, f_n is a system of local equations.

Furthermore, let $\ell_0, \ell_{r+1}, \dots, \ell_n$ be linear forms which complete the previous ones to a change of variables such that $Z \cap \{\ell_0 = 0\} = \emptyset$.

Then V satisfies Assumption 1.2 with respect to these variables, and the statement follows directly from Main Lemma 2.3. \square

3 The computation of the Chow form

We devote this section to the description and complexity analysis of the algorithm underlying Theorem 1. The first subsections gather some results which lead to the proof of the theorem.

3.1 Geometric resolutions

Geometric resolutions were first introduced in the works of Kronecker and König in the last years of the XIXth century. Nowadays they are widely used in computer algebra, especially in the 0-dimensional case, but there are also important applications in the general case. We refer to [25] for a complete historical account.

In what follows we recall how to compute any —sufficiently generic— geometric resolution of an equidimensional variety from a Chow form in polynomial time. This computation and the procedure described in Section 2 imply that, from the point of view of complexity, Chow forms and geometric resolutions are equivalent representations of an equidimensional variety.

Let $V \subset \mathbb{A}^n$ be an equidimensional affine variety of dimension r and degree D . For $0 \leq i \leq r$, let L_i denote as usual the generic affine forms. Let $c_i \in \mathbb{Q}^{n+1}$. We set

$$\ell_i := L_i(c_i) = c_{i0} + c_{i1} x_1 + \dots + c_{in} x_n \in \mathbb{Q}[x_1, \dots, x_n].$$

We assume that the projection map

$$\pi_{(\ell_1, \dots, \ell_r)} : V \rightarrow \mathbb{A}^r \quad , \quad x \mapsto (\ell_1(x), \dots, \ell_r(x))$$

is *finite*, that is the affine linear forms ℓ_1, \dots, ℓ_r are in Noether position with respect to V . Let y_1, \dots, y_r be new variables. Set

$$K := \mathbb{Q}(y_1, \dots, y_r) \quad , \quad L = \mathbb{Q}(\ell_1, \dots, \ell_r) \otimes_{\mathbb{Q}[\ell_1, \dots, \ell_r]} \mathbb{Q}[V]$$

and consider the morphism

$$K \rightarrow L \quad , \quad y_i \mapsto \ell_i.$$

Then $K \hookrightarrow L$ is a finite extension of degree $[L : K] \leq D$. We assume furthermore that ℓ_0 is a primitive element of this extension, that is $L = K[\ell_0]$.

Then the *geometric resolution of V associated to $\ell := (\ell_0, \dots, \ell_r)$* is the pair

$$p := p_{V,\ell} \in K[t] \quad , \quad w := w_{V,\ell} \in K[t]^n$$

where p is the monic minimal polynomial of ℓ_0 with respect to the extension $K \hookrightarrow L$, and $w = (w_1, \dots, w_n)$ verifies $\deg w_i < [L : K]$ and $p'(\ell_0) x_i = w_i(\ell_0) \in L$ for $1 \leq i \leq n$, where $p' := \partial p / \partial t$. These polynomials are uniquely determined, and, because of the Noether position assumption, we have that p, w_i lie in fact in $\mathbb{Q}[y_1, \dots, y_r][t]$, see e.g. [29, Section 3.2].

A geometric resolution gives a parametrization of a dense open set of V in terms of the points of a hypersurface in \mathbb{A}^{r+1} : there is a map

$$\begin{aligned} V(p(t, y_1, \dots, y_r)) \setminus V(p'(t, y_1, \dots, y_r)) &\rightarrow V \setminus V(p'(\ell_0(x), \ell_1(x), \dots, \ell_r(x))) \\ (t, y_1, \dots, y_r) &\mapsto \frac{w}{p'}(t, y_1, \dots, y_r). \end{aligned}$$

Note that, in case the considered variety is 0-dimensional, this definition of geometric resolution essentially coincides with the one given in Section 2: the passage from one to the other can be made by considering the resultant with respect to the variable t between p and p' .

The following construction shows that the geometric resolution associated to the generic affine linear forms L_0, \dots, L_r can be expressed in terms of the characteristic polynomial of the variety, and hence in terms of the Chow form:

Let U_0, \dots, U_r be $r+1$ groups of $n+1$ variables which correspond to the coordinate functions of $\mathbb{A}^{(r+1)(n+1)}$ and let $T := (T_0, \dots, T_r)$ be a group of $r+1$ variables which correspond to the coordinate functions of \mathbb{A}^{r+1} . We recall that a *characteristic polynomial* $\mathcal{P}_V \in \mathbb{Q}[U_0, \dots, U_r][T_0, \dots, T_r]$ of V is defined as any defining equation of the Zariski closure of the image of the map

$$\varphi_V : \mathbb{A}^{(r+1)(n+1)} \times V \rightarrow \mathbb{A}^{(r+1)(n+1)} \times \mathbb{A}^{r+1}, \quad (u_0, \dots, u_r; \xi) \mapsto (u_0, \dots, u_r; L_0(u_0, \xi), \dots, L_r(u_r, \xi))$$

which is a hypersurface. This is a multihomogeneous polynomial of degree D in each group of variables $U_i \cup \{T_i\}$. Its degree in the group of variables T is also bounded by D .

A characteristic polynomial of V can be derived from a Chow form \mathcal{F}_V . For $1 \leq i \leq r$ we set $\zeta_i := (U_{i0} - T_i, U_{i1}, \dots, U_{in})$. Then

$$\mathcal{P}_V = (-1)^D \mathcal{F}_V(\zeta_0, \dots, \zeta_r) \tag{6}$$

is a characteristic polynomial of V .

Set $\mathcal{P}_V := a_D T_0^D + \dots + a_0$ for the expansion of \mathcal{P}_V with respect to T_0 . Then a_D lies in $\mathbb{Q}[U_1, \dots, U_r] \setminus \{0\}$, and in fact it coincides with the coefficient of U_{00}^D in \mathcal{F}_V , that is

$$a_D(U_1, \dots, U_r) = \mathcal{F}_V(e_0, U_1, \dots, U_r).$$

In case V satisfies Assumption 1.2, we define *the* characteristic polynomial of V as

$$(-1)^D \text{Ch}_V(\zeta_0, \dots, \zeta_r)$$

where Ch_V is the normalized Chow form of V . We refer to [41, Section 2.3.1] for further details as well as for the proof of the stated facts.

Lemma 3.1 *Let $V \subset \mathbb{A}^n$ be an equidimensional variety of dimension r and degree D . Let U_0, \dots, U_r be $r+1$ groups of $n+1$ variables and let L_0, \dots, L_r be the generic affine forms associated to U_0, \dots, U_r . Set $E := \mathbb{Q}(U_0, \dots, U_r)$ and let V^e denote the Zariski closure of V in $\mathbb{A}^n(\overline{E})$. Let T_0, \dots, T_r be new indeterminates.*

Then the geometric resolution of V^e associated to L_0, \dots, L_r is given by

$$P := \frac{\mathcal{P}_V}{a_D} \in E[T_1, \dots, T_r][T_0] \quad , \quad W := -\frac{1}{a_D} \left(\frac{\partial \mathcal{P}_V}{\partial U_{01}}, \dots, \frac{\partial \mathcal{P}_V}{\partial U_{0n}} \right) \in E[T_1, \dots, T_r][T_0]^n$$

where \mathcal{P}_V is a characteristic polynomial of V and a_D is the leading coefficient of \mathcal{P}_V with respect to T_0 .

Proof.– Using the fact that the extended ideal $I(V)^e \subset E[x_1, \dots, x_n]$ is radical, it is easy to check that $I(V^e) = I(V)^e$. Consider then the morphism

$$\mathcal{A} := E[T_1, \dots, T_r] \rightarrow \mathcal{B} := E[x_1, \dots, x_n]/I(V)^e \quad , \quad T_i \mapsto L_i(U_i, x).$$

Our first aim is to prove that this is an integral inclusion, or in other words, that the projection map $\pi_{(L_1, \dots, L_r)} : V^e \rightarrow \mathbb{A}^r(\overline{E})$ is finite.

By definition

$$P(U_0, \dots, U_r)(L_0(U_0, x), \dots, L_r(U_r, x)) \equiv 0 \pmod{I(V) \otimes_{\mathbb{Q}[x]} \mathbb{Q}(U)[x]}. \quad (7)$$

Specializing U_0 by the $(i+1)$ -th element of the canonical basis e_i in this identity, we deduce that $P(e_i, U_1, \dots, U_r)(T_0, T_1, \dots, T_r) \in \mathcal{A}[T_0]$ is a monic equation for x_i for $i = 1, \dots, n$. Therefore $\mathcal{A} \hookrightarrow \mathcal{B}$ is an integral extension.

Set $\mathcal{K} := E(T_1, \dots, T_r)$ and $\mathcal{L} := \mathcal{K} \otimes_{\mathcal{A}} \mathcal{B}$.

It is immediate that $P := \mathcal{P}_V/a_D$ is a monic polynomial equation for L_0 with respect to the extension $\mathcal{K} \hookrightarrow \mathcal{L}$. As $\mathcal{A} \hookrightarrow \mathcal{B}$ is an integral extension, from the definition of \mathcal{P}_V we deduce that P is the minimal monic polynomial of L_0 . This implies that $[\mathcal{L} : \mathcal{K}] = D$ and that L_0 is a primitive element of this extension.

Write

$$Q(U, x) := \mathcal{P}_V(U_0, \dots, U_r)(L_0(U_0, x), \dots, L_r(U_r, x)) = \sum_{\beta} b_{\beta} U_0^{\beta}$$

with $b_{\beta} \in \mathbb{Q}[U_1, \dots, U_r][x_1, \dots, x_n]$.

As $b_{\beta} \in I(V)^e \subset \mathbb{Q}[U_1, \dots, U_r][x_1, \dots, x_n]$ for all β , $\frac{\partial Q(U, x)}{\partial U_{0i}} \in I(V)^e \subset \mathbb{Q}[U_0, \dots, U_r][x_1, \dots, x_n]$

for $i = 1, \dots, n$. Therefore, $\frac{\partial Q(U, x)}{\partial U_{0i}} = 0$ in \mathcal{L} for $1 \leq i \leq n$. Then the chain rule implies that the identity

$$\frac{\partial \mathcal{P}_V}{\partial T_0}(U, L(U, x)) x_i = -\frac{\partial \mathcal{P}_V}{\partial U_{0i}}(U, L(U, x))$$

holds in \mathcal{L} and the lemma follows. \square

Now we show how a particular geometric resolution can be obtained by direct specialization of the generic one.

Using the same notation as in the beginning of this subsection, we will assume that $V \cap V(\ell_1, \dots, \ell_r)$ is a 0-dimensional variety of cardinality D . This condition is satisfied provided that ℓ_1, \dots, ℓ_r are generic enough [41, Proposition 4.5]. After a linear change of variables, we may assume w.l.o.g. that $\ell_i = x_i$ for $i = 1, \dots, r$, so that the stated condition is Assumption 1.2.

Thus, for the rest of this section we fix the following notations:

$$Z := V \cap V(x_1, \dots, x_r) \quad , \quad K := \mathbb{Q}(x_1, \dots, x_r) \quad , \quad L := K \otimes_{\mathbb{Q}[x_1, \dots, x_r]} \mathbb{Q}[V].$$

We also assume that $\ell_0 = L_0(c_0, x) \in \mathbb{Q}[x_1, \dots, x_n]$ separates the points of Z . This is also a generic condition: if we set $\rho := \text{discr}_{T_0} \mathcal{P}_Z \in \mathbb{Q}[U_0] \setminus \{0\}$, this condition is satisfied provided that $\rho(c_0) \neq 0$.

These two conditions ensure the existence of the associated geometric resolution of V :

Lemma 3.2 *Let $V \subset \mathbb{A}^n$ be an equidimensional variety of dimension r and degree D which satisfies Assumption 1.2. Let $\ell_0 := L_0(c_0, x) \in \mathbb{Q}[x_1, \dots, x_n]$ be an affine linear form which separates the points of Z .*

Then the projection map $\pi : V \rightarrow \mathbb{A}^r$, $\pi(x) = (x_1, \dots, x_r)$ is finite and ℓ_0 is a primitive element of the extension $K \hookrightarrow L$. The geometric resolution of V associated to $\ell := (\ell_0, x_1, \dots, x_r)$ is given by

$$\begin{aligned} p &:= \mathcal{P}_V(c_0, e_1, \dots, e_r)(t, x_1, \dots, x_r) \in \mathbb{Q}[x_1, \dots, x_r][t], \\ w &:= -\left(\frac{\partial \mathcal{P}_V}{\partial U_{01}}, \dots, \frac{\partial \mathcal{P}_V}{\partial U_{0n}}\right)(c_0, e_1, \dots, e_r)(t, x_1, \dots, x_r) \in \mathbb{Q}[x_1, \dots, x_r][t]^n, \end{aligned}$$

where \mathcal{P}_V is the normalized characteristic polynomial of V .

Proof.— The fact that π is finite follows from [41, Lemma 2.14]. On the other hand, the normalization imposed on \mathcal{P}_V implies that

$$p_0(t) := p(t, 0, \dots, 0) = \mathcal{P}_V(c_0, e_1, \dots, e_r)(t, 0, \dots, 0) \in \mathbb{Q}[t]$$

is a monic—and thus non-zero—polynomial of degree D which vanishes on $\ell_0(\xi)$ for all $\xi \in Z$. The hypothesis that ℓ_0 separates the points of Z implies that p_0 is the minimal polynomial of ℓ_0 over Z ; in particular it is a *squarefree* polynomial of degree D and so, as p is monic,

$$0 \neq \text{discr } p_0 = (\text{discr}_t p)(0, \dots, 0).$$

In particular, $\text{discr}_t p \neq 0$, and thus p is also a squarefree polynomial which annihilates ℓ_0 over V . Now, as the map π is finite, the minimal polynomial $m_{\ell_0} \in K[t]$ of ℓ_0 lies in $\mathbb{Q}[x_1, \dots, x_r][t]$. Hence $m_{\ell_0}(0, \dots, 0, t)$ vanishes on $\ell_0(\xi)$ for all $\xi \in Z$. This implies that $\deg_t m_{\ell_0} = D$. As p is a monic polynomial of degree D in t , then $p = m_{\ell_0}$. So ℓ_0 is a primitive element of the extension $K \hookrightarrow L$, and p is its minimal polynomial.

Using the same notation of Lemma 3.1 we have

$$\frac{\partial \mathcal{P}_V}{\partial T_0}(U, L(U, x)) x_i = -\frac{\partial \mathcal{P}_V}{\partial U_{0i}}(U, L(U, x)) \in \mathcal{L}.$$

As this identity involves only polynomials in $\mathbb{Q}[U_0, \dots, U_r][x_1, \dots, x_n]$, it can be directly evaluated to obtain the parametrization w . \square

In particular this shows that the *total* degree of the polynomials in the geometric resolution is bounded by $\deg p \leq D$ and $\deg w_i \leq D$ (See also [29, Proposition 3]).

Lemma 3.2 can be applied directly to compute a geometric resolution of an equidimensional variety V which satisfies Assumption 1.2 from a given Chow form of V :

Corollary 3.3 *Let notations and assumptions be as in Lemma 3.2. Suppose that there is given a Chow form \mathcal{F}_V of V , encoded by an *slp* of length L . Then, there is an algorithm which computes a geometric resolution of V associated to ℓ within complexity $\mathcal{O}(nL)$. All polynomials arising in this geometric resolution are encoded by *slp*'s of length $\mathcal{O}(L)$. \square*

Lemma 3.2 also yields, from Ch_V , a geometric resolution of the fiber Z associated to an affine linear form ℓ_0 , as $\text{Ch}_Z(U_0) = \text{Ch}_V(U_0, e_1, \dots, e_r)$. This is summarized in Procedure *GeomRes* (Subroutine 8).

In Procedure *GeomRes* (Subroutine 8), as we do in all zero-dimensional situations, we use the definition of geometric resolution stated in Section 2 to avoid divisions by p' . In line 8 of this subroutine, $\text{Res}(f, g, d_1, d_2)$ is a procedure that, using basic linear algebra, computes (ρ, q_1, q_2) where ρ is the resultant between the univariate polynomials f and g of degrees d_1 and d_2 respectively, and q_1 and q_2 are polynomials of degrees bounded by $d_2 - 1$ and $d_1 - 1$ respectively satisfying $\rho = q_1 f + q_2 g$. In line 13, $\text{Mod}(f, g, d_1, d_2)$ is a procedure that computes the remainder of the division of the polynomial f of degree bounded by d_1 by the polynomial g of degree bounded by d_2 .

Subroutine 8: Computing a geometric resolution of a fiber

procedure GeomRes(n, r, D, Ch_V, ξ, c)

n is the number of variables,

r, D are the dimension and an upper bound for the degree of V respectively,

Ch_V is the normalized Chow form of V ,

$\xi := (\xi_1, \dots, \xi_r) \in \mathbb{A}^r$ such that $\#Z_\xi = \deg V$, where $Z_\xi := V \cap V(x_1 - \xi_1, \dots, x_r - \xi_r)$.

$c_0 \in \mathbb{Q}^{n+1}$ s.t. $\ell_0 := L_0(c_0, x)$ is the considered affine linear form.

The procedure returns (D_0, p, v) , where D_0 is the degree of V and $(p, v) \in \mathbb{Q}[t]^{n+1}$ is the

geometric resolution of Z_ξ associated to ℓ_0 in case ℓ_0 separates the points in Z_ξ .

Otherwise, it returns error.

1. $P(U_0, t) := Ch_V((U_{00} - t, U_{01}, \dots, U_{0n}), e_1 - \xi_1 e_0, \dots, e_r - \xi_r e_0)$;

2. $(p_0, \dots, p_D) := \text{Expand}(P(c_0, t), t, 0, D)$;

3. $D_0 := D$;

4. **while** $p_{D_0} = 0$ and $D_0 \geq 0$ **do**

5. $D_0 := D_0 - 1$;

6. **od**;

7. $p := (-1)^{D_0} P(c_0, t)$;

8. $(\rho, q_1, q_2) := \text{Res}(p, p', D_0, D_0 - 1)$;

9. **if** $\rho = 0$ **then**

10. **return** (“error”);

11. **else**

12. $(w_1, \dots, w_n) := ((-1)^{D_0+1} \partial P / \partial U_{01}(c_0, t), \dots, (-1)^{D_0+1} \partial P / \partial U_{0n}(c_0, t))$;

13. $(v_1, \dots, v_n) := (\text{Mod}(\frac{1}{p} q_2 w_1, p, 2D_0 - 1, D_0), \dots, \text{Mod}(\frac{1}{p} q_2 w_n, p, 2D_0 - 1, D_0))$;

14. **return** $(D_0, p, v_1, \dots, v_n)$;

end.

Proposition 3.4 *Let $V \subset \mathbb{A}^n$ be an equidimensional variety of dimension r and degree bounded by D . Let $(\xi_1, \dots, \xi_r) \in \mathbb{A}^r$ be such that $Z_\xi := V \cap V(x_1 - \xi_1, \dots, x_r - \xi_r)$ is a 0-dimensional variety of cardinality $\deg V$. Assume we are given both an slp of length L encoding Ch_V and the coefficients of an affine linear form $\ell_0 \in \mathbb{Q}[x_1, \dots, x_n]$ which separates the points in Z_ξ . Then, Procedure GeomRes (Subroutine 8) computes a geometric resolution of Z_ξ (in the sense of Section 2) within complexity $\mathcal{O}(nD^2L + D^4)$. \square*

On the other hand, our next result shows the converse of Corollary 3.3: To derive a Chow form from a given geometric resolution is quite standard in the zero-dimensional case, but it was by no means clear up to now how to generalize that for varieties of arbitrary dimension. Here we show how to do that within polynomial complexity. This is done by deriving, from the given geometric resolution of V , a geometric resolution of the fiber Z and a system of local equations for V at Z , which enables us to apply Procedure ChowForm (Subroutine 7).

Proposition 3.5 *Let $V \subset \mathbb{A}^n$ be an equidimensional variety of dimension r and degree D which satisfies Assumption 1.2. Let $\ell_0 \in \mathbb{Q}[x_1, \dots, x_n]$ be a linear form which separates the points of Z . Suppose that there is given a geometric resolution (p, w) of V associated to $\ell := (\ell_0, x_1, \dots, x_r)$,*

encoded by slp's of length L . Then there is a bounded probability algorithm which computes (an slp for) Ch_V within complexity $\mathcal{O}(n^{16}D^{19}(D+L))$.

Proof.— First we derive a geometric resolution of Z associated to ℓ_0 :
We know that $\text{Ch}_Z(U_0) = \text{Ch}_V(U_0, e_1, \dots, e_r)$. Thus,

$$\mathcal{P}_Z(U_0)(t) = \mathcal{P}_V(U_0, e_1, \dots, e_r)(t, 0, \dots, 0) \in \mathbb{Q}[U_0][t].$$

The geometric resolution (p, w) of V associated to ℓ is given by Lemma 3.2. Applying the same lemma to Z , we deduce that the geometric resolution (p_0, w_0) of Z associated to ℓ_0 is $p_0(t) := p(t, 0, \dots, 0) \in \mathbb{Q}[t]$ and $w_0(t) := w(t, 0, \dots, 0) \in \mathbb{Q}[t]^n$.

Now, let us derive a system of local equations of V at Z :

Let $c_i \in \mathbb{Q}^{n+1}$, $r+1 \leq i \leq n$, be such that the affine linear forms $\ell_i := c_{i0} + c_{i1}x_1 + \dots + c_{in}x_n \in \mathbb{Q}[x_1, \dots, x_n]$ are linearly independent and such that each of them separates the points of Z .

For $r+1 \leq i \leq n$ define

$$H_i := |p'(M_p)t - (p'\ell_i(w/p'))(M_p)|,$$

where $M_p \in \mathbb{Q}[x_1, \dots, x_r]^{D \times D}$ denotes the companion matrix of p . Since $p'\ell_i(w/p')$ belongs to $\mathbb{Q}[x_1, \dots, x_r][t]$, we see that $H_i \in \mathbb{Q}[x_1, \dots, x_r][t]$.

Observe that $x_i = (w_i/p')(\ell_0(x))$ in L implies that in L

$$p'(\ell_0)\ell_i = c_{i0}p'(\ell_0) + c_{i1}w_1(\ell_0) + \dots + c_{in}w_n(\ell_0) = (p'\ell_i(w/p'))(\ell_0).$$

Thus, as M_p is the matrix of multiplication by ℓ_0 with respect to $K \hookrightarrow L$, we conclude that $H_i = |p'(M_p)|m_{\ell_i}$ where m_{ℓ_i} is the minimal polynomial of ℓ_i over K .

Now, each m_{ℓ_i} belongs to $\mathbb{Q}[x_1, \dots, x_r][t]$ because ℓ_i separates the points of Z and the projection $\pi : V \rightarrow \mathbb{A}^r, x \mapsto (x_1, \dots, x_r)$ is finite.

Therefore, for $r+1 \leq i \leq n$, we can define

$$f_i := m_{\ell_i}(\ell_i) = \frac{1}{|p'(M_p)|} H_i(x_1, \dots, x_r)(\ell_i).$$

These are squarefree polynomials in separated variables which vanish over V , and so it is easy to verify from the Jacobian criterion that f_{r+1}, \dots, f_n is a system of reduced local equations of V at Z .

Now we show that $|p'(M_p)|(0, \dots, 0) \neq 0$, and hence we can use the point $(0, \dots, 0)$ to perform Procedure PolynomialDivision (Subroutine 2) in order to obtain division free slp's for f_{r+1}, \dots, f_n : There exist $a, b \in \mathbb{Q}[x_1, \dots, x_r][t]$ such that $\text{discr}(p) = a(t)p(t) + b(t)p'(t)$, thus $\text{discr}(p)\text{Id} = b(M_p)p'(M_p)$. On the other hand, the fact that $\deg Z = \deg V$ implies $\text{discr}(p)(0, \dots, 0) = \text{discr}(p_0) \neq 0$. Therefore, $|b(M_p)p'(M_p)|(0, \dots, 0) = (\text{discr}(p_0))^D \neq 0$.

Finally, we apply procedure ChowForm (Subroutine 7) to Z and $\{f_{r+1}, \dots, f_n\}$.

Let us decide now the random choices in order to ensure that the algorithm has an error probability bounded by $1/4$:

We need $c_{r+1}, \dots, c_n \in \mathbb{Q}^{n+1}$ satisfying the stated conditions of independence and separability. These conditions are satisfied provided that

$$\rho(c_{r+1}) \cdots \rho(c_n) \left| (c_{ij})_{\substack{r+1 \leq i \leq n \\ 1 \leq j \leq n-r}} \right| \neq 0,$$

where $\rho := \text{discr}_t \mathcal{P}_Z \in \mathbb{Q}[U_0] \setminus \{0\}$. As \mathcal{P}_Z is an homogeneous polynomial of degree D and $\deg_t \mathcal{P}_Z = D$, $\deg \rho \leq D(2D-1)$. Thus the degree of the polynomial giving bad choices is bounded by $(n-r)D(2D-1) + (n-r)$. We choose $\ell := 8nD^2$ in order to apply Schwartz lemma.

Now we compute the complexity of the algorithm:

The dense representation of the geometric resolution of Z associated to ℓ_0 is computed within complexity $\mathcal{O}(nD^2L)$ (using Procedure Expand).

The construction of the random choice for the affine linear forms $\ell_{r+1}, \dots, \ell_n$ is not relevant here. The computation of each polynomial H_i requires $\mathcal{O}(D^4)$ operations for the computation of the determinant plus the computation of each coefficient of the matrix, that is $\mathcal{O}(D^3 L)$ more operations. Hence, computing H_i requires $\mathcal{O}(D^3(D+L))$ operations.

By Lemma 1.7, taking into account that the total degree of each f_i is bounded by D (since it is the minimal polynomial of the affine linear form ℓ_i), and that the lengths of H_i and $|p'(M_p)|$ are of order $\mathcal{O}(D^3(D+L))$, the complexity of the final division for computing each f_i is $\mathcal{O}(D^2(D+D^3(D+L))) = \mathcal{O}(D^5(D+L))$.

Finally, Lemma 2.3 gives the final complexity $\mathcal{O}(r^8 \log_2(rD)n^7 D^{13} D^5(D+L)) = \mathcal{O}(n^{16} D^{19}(D+L))$. \square

3.2 Intersection of a variety with a hypersurface

Let $V \subseteq \mathbb{A}^n$ be an equidimensional variety defined over \mathbb{Q} , and let $f \in \mathbb{Q}[x_1, \dots, x_n]$ be a non-zero divisor modulo $I(V)$. In this section we compute, from the Chow form of V and the equation f , the Chow form of the set-theoretic intersection $V \cap V(f) \subset \mathbb{A}^n$. In order to do this, we use *generalized Chow forms*, which we define now. We refer to [47] and [41, Section 2.1.1] for a more extensive treatment of these generalized Chow forms.

We assume that $\dim V = r$ and that $\deg f \leq d$. As before, for $i = 0, \dots, r$, we introduce a group $U_i = (U_{i0}, \dots, U_{in})$ of $n+1$ variables; we introduce also a group $U(d)_r$ of $\binom{d+n}{n}$ variables. We set

$$L_i := U_{i0} + U_{i1}x_1 + \dots + U_{in}x_n \quad , \quad F_r := \sum_{|\alpha| \leq d} U(d)_{r\alpha} x^\alpha$$

for the generic affine linear forms in n variables associated to U_i and the generic polynomial of degree d in n variables associated to $U(d)_r$.

Set $N := r(n+1) + \binom{d+n}{n}$ and let $W \subset \mathbb{A}^N \times V$ be the incidence variety of L_0, \dots, L_{r-1}, F_r with respect to V , that is

$$W := \{(u_0, \dots, u_{r-1}, u(d)_r; \xi) \in \mathbb{A}^N \times \mathbb{A}^n; \\ \xi \in V, L_0(u_0, \xi) = 0, \dots, L_{r-1}(u_{r-1}, \xi) = 0, F_r(u(d)_r, \xi) = 0\}.$$

Let $\pi : \mathbb{A}^N \times \mathbb{A}^n \rightarrow \mathbb{A}^N$ denote the canonical projection onto the first coordinates. Then $\overline{\pi(W)}$ is a hypersurface in \mathbb{A}^N . A *generalized Chow form* or *d-Chow form* of V is any squarefree polynomial $\mathcal{F}_{d,V} \in \mathbb{Q}[U_0, \dots, U_{r-1}, U(d)_r]$ defining $\overline{\pi(W)} \subseteq \mathbb{A}^N$.

A *d-Chow form* $\mathcal{F}_{d,V}$ happens to be a multihomogeneous polynomial of degree $d \deg V$ in each group of variables U_i , and of degree $\deg V$ in the group $U(d)_r$.

If the variety V satisfies Assumption 1.2, we define the *normalized d-Chow form* of V as the unique *d-Chow form* $\mathcal{Ch}_{d,V} \in \mathbb{Q}[U_0, \dots, U_{r-1}, U(d)_r]$ of V satisfying $\mathcal{Ch}_{d,V}(e_0, \dots, e_{r-1}, e(d)) = 1$, where $e(d)$ is the vector of coefficients of the polynomial x_r^d .

Let \overline{V} and $\overline{V(f)}$ denote the closure in \mathbb{P}^n of V and $V(f)$ respectively. Set $\overline{V} \cap \overline{V(f)} = \bigcup_C C$ for the irreducible decomposition of $\overline{V} \cap \overline{V(f)} \subset \mathbb{P}^n$ and, for each irreducible component C , let $\mathcal{F}_C \in \mathbb{Q}[U_0, \dots, U_{r-1}]$ denote a Chow form of C . Then [47, Proposition 2.4] states that

$$\mathcal{Ch}_{d,V}(U_0, \dots, U_{r-1}, f) = \lambda \prod_C \mathcal{F}_C^{m_C} \quad (8)$$

for some $\lambda \in \mathbb{Q}^*$ and some positive integers $m_C \in \mathbb{N}$. (Here we wrote $\mathcal{Ch}_{d,V}(U_0, \dots, U_{r-1}, f)$ for the specialization of the group $U(d)_r$ into the coefficients of the polynomial f .)

On the other hand, as $V \cap V(f) = \bigcup_{C \not\subset \{x_0=0\}} C$, the polynomial

$$\prod_{C \not\subset \{x_0=0\}} \mathcal{F}_C$$

is a Chow form of $V \cap V(f)$.

Hence, in order to compute $\mathcal{C}h_{V \cap V(f)}$, the goal is to compute first $\mathcal{C}h_{d,V}(f) := \mathcal{C}h_{d,V}(U_0, \dots, U_{r-1}, f)$, then compute its squarefree part and finally eliminate the factors coming from the Chow forms of components contained in the hyperplane $\{x_0 = 0\}$.

The following result enables us to compute a d -Chow form from the standard one. We recall some of the notation of Subsection 2.2: for an equidimensional variety $V \subset \mathbb{A}^n$ of dimension r and degree D satisfying Assumption 1.2, we set $K := \mathbb{Q}(U_0, \dots, U_{r-1})$ and $I(V)^e$ for the extension of the ideal of V to $K[x_1, \dots, x_n]$. Also recall that

$$V^0 := V(I(V)^e) \cap V(L_0, \dots, L_{r-1}) \subset \mathbb{A}^n(\overline{K}),$$

is a 0-dimensional variety of degree D , and that N_{V^0} refers to the Norm as defined in Section 2.1.

Lemma 3.6 *Under Assumption 1.2, we have*

$$\mathcal{C}h_{d,V} = \mathcal{C}h_V(U_0, \dots, U_{r-1}, e_0)^d N_{V^0}(F_r).$$

Proof.— Let $\mathcal{C}h_{d,V^0} \in K[U_r]$ be the d -Chow form of V^0 . First one shows —exactly as in Lemma 2.2— that there exists $\lambda_d \in K^*$ such that $\mathcal{C}h_{d,V} = \lambda_d \mathcal{C}h_{d,V^0}$. Set $e(d)_0$ for the vector of coefficients of the polynomial x_0^d . Evaluating this identity at $U(d)_0 \mapsto e(d)_0$ we obtain

$$\mathcal{C}h_{d,V}(U_0, \dots, U_{r-1}, e(d)_0) = \lambda_d \mathcal{C}h_{d,V^0}(e(d)_0) = \lambda_d.$$

Consider the morphism $\varrho_d : \mathbb{Q}[U_0, \dots, U_{r-1}, U(d)_r] \rightarrow \mathbb{Q}[U_0, \dots, U_{r-1}, U_r]$ defined by $\varrho_d(L_i) = L_i$ for $0 \leq i \leq r-1$ and $\varrho_d(F_r) = L_r^d$. Then $\varrho_d(\mathcal{C}h_{d,V}) = \mathcal{C}h_V^d$ (see [41, Lemma 2.1]), which implies that

$$\mathcal{C}h_{d,V}(U_0, \dots, U_{r-1}, e(d)_0) = \varrho_d(\mathcal{C}h_{d,V})(U_0, \dots, U_{r-1}, e_0) = \mathcal{C}h_V(U_0, \dots, U_{r-1}, e_0)^d.$$

Therefore $\lambda_d = \mathcal{C}h_V(U_0, \dots, U_{r-1}, e_0)^d$. The statement follows immediately from this identity and the observation that $\mathcal{C}h_{d,V^0} = N_{V^0}(F_r)$. \square

To clean the components of $\overline{V \cap V(f)}$ lying in the hyperplane $\{x_0 = 0\}$ we use the following criterion:

Lemma 3.7 *Let $W \subset \mathbb{P}^n$ be an irreducible variety of dimension $r-1$. Then $W \subset \{x_0 = 0\}$ if and only if \mathcal{F}_W does not depend on the variable U_{00} .*

Proof.— In case \mathcal{F}_W does not depend on U_{00} we have that

$$\mathcal{F}_W(e_0, U_1, \dots, U_{r-1}) = 0,$$

which is equivalent to the fact that W is contained in the hyperplane $\{x_0 = 0\}$.

On the other hand, assume that $W \subset \{x_0 = 0\} \cong \mathbb{P}^{n-1}$. Then \mathcal{F}_W coincides with the Chow form of W considered as a subvariety of this linear space, see e.g. the proof of [41, Lemma 2.6]. Hence \mathcal{F}_W does not depend on U_{00} , and as a matter of fact, it does not depend on any of the variables U_{i0} for $0 \leq i \leq r-1$. \square

Let again $\mathcal{F}_C \in \mathbb{Q}[U_0, \dots, U_{r-1}]$ denote a Chow form of an irreducible component C of $\overline{V \cap V(f)} \subset \mathbb{P}^n$. Recalling Identity (8), set

$$\mathcal{F}_1 := \prod_{C \subset \{x_0=0\}} \mathcal{F}_C^{m_C} \quad \text{and} \quad \mathcal{F}_2 := \prod_{C \not\subset \{x_0=0\}} \mathcal{F}_C^{m_C}.$$

Then $\mathcal{C}h_{d,V}(f) = \lambda \mathcal{F}_1 \mathcal{F}_2$ for $\lambda \in \mathbb{Q}^*$, and the squarefree part $(\mathcal{F}_2)_{\text{red}}$ of \mathcal{F}_2 is a Chow form of $V \cap V(f)$.

By the previous lemma, \mathcal{F}_1 does not depend on U_{00} , while all the factors of \mathcal{F}_2 do.

Therefore

$$\frac{\partial \mathcal{C}h_{d,V}(f)}{\partial U_{00}} = \lambda \mathcal{F}_1 \frac{\partial \mathcal{F}_2}{\partial U_{00}},$$

and so

$$\mathcal{F}_{V \cap V(f)} := \frac{\mathcal{C}h_{d,V}(f)}{\gcd(\mathcal{C}h_{d,V}(f), \partial \mathcal{C}h_{d,V}(f) / \partial U_{00})} \quad (9)$$

is a Chow form of $V \cap V(f)$.

Lemma 3.8 *Let $V \subset \mathbb{A}^n$ be an equidimensional variety of degree D which satisfies Assumption 1.2 and let $f \in \mathbb{Q}[x_1, \dots, x_n]$ of degree bounded by d be a non-zero divisor modulo $I(V)$. Assume that $\mathcal{C}h_V$ and f are encoded by slp's of length bounded by L .*

Then there is a bounded probability algorithm (Procedure Intersection (Subroutine 9) below) which computes the Chow form $\mathcal{C}h_{V \cap V(f)}$ of the intersection variety $V \cap V(f)$ within (worst-case) complexity $\mathcal{O}((ndD)^{12}L)$.

Proof.— Our first goal is to compute $\mathcal{C}h_{d,V}(f) \in \mathbb{Q}[U_0, \dots, U_{r-1}]$ by means of Lemma 3.6. To obtain $N_{V^0}(f)$ we derive first a geometric resolution of V^0 from its characteristic polynomial and Lemma 3.1. It is easy to check that the polynomial

$$p(t) := (-1)^D \mathcal{C}h_V(U_0, \dots, U_{r-1}, (U_{r0} - t, U_{r1}, \dots, U_{rn}))$$

is a characteristic polynomial of V^0 , with leading coefficient $a := \mathcal{C}h_V(U_0, \dots, U_{r-1}, e_0)$. Then, the geometric resolution of V^0 associated to L_r is given by

$$\frac{1}{a} p(t) \in K[U_r][t] \quad \text{and} \quad \frac{1}{a} w(t) \in K[U_r][t]^n \quad \text{where} \quad w := - \left(\frac{\partial p}{\partial U_{r1}}, \dots, \frac{\partial p}{\partial U_{rn}} \right).$$

For $\gamma \in V^0$, if we denote by f^h the homogeneization up to degree d of f with respect to a new variable x_0 and p' the derivative of p with respect to t , we have

$$p'(L_r(\gamma))^d f(\gamma) = f^h(p'(L_r(\gamma)), w_1(L_r(\gamma)), \dots, w_n(L_r(\gamma))).$$

Thus, if M denotes the companion matrix of $\frac{1}{a} p(t)$, we get

$$|p'(M)|^d N_{V^0}(f) = |f^h(p'(M), w_1(M), \dots, w_n(M))|.$$

In order to avoid divisions (since $M \in K[U_r]$), we replace M by $M_p := aM$ and p', w_1, \dots, w_n by their homogeneizations $(p')^h, w_1^h, \dots, w_n^h$ up to degree D such that $M_0 := a^D p'(M) = (p')^h(a \text{Id}, M_p)$ and for $1 \leq i \leq n$, $M_i := a^D w_i(M) = w_i^h(a \text{Id}, M_p)$. Therefore, multiplying both sides by $a^{dD^2} = |a^D \text{Id}|^d$, we obtain

$$|M_0|^d N_{V^0}(f) = |f^h(M_0, M_1, \dots, M_n)|.$$

Finally, from Lemma 3.6, we conclude that

$$\mathcal{C}h_{d,V}(f) = a^d N_{V^0}(f) = \frac{a^d |f^h(M_0, M_1, \dots, M_n)|}{|M_0|^d} \in \mathbb{Q}[U_0, \dots, U_{r-1}]. \quad (10)$$

We compute this quotient applying Procedure PolynomialDivision (Subroutine 2).

Next we apply Identity (9) to compute a Chow form $\mathcal{F} := \mathcal{F}_{V \cap V(f)}$ from $\mathcal{C}h_{d,V}(f)$: we first compute the polynomial

$$G := \gcd(\mathcal{C}h_{d,V}(f), \partial \mathcal{C}h_{d,V}(f) / \partial U_{00}) \quad (11)$$

applying Procedure GCD (Subroutine 3) and then perform the division $\mathcal{F} = \mathcal{C}h_{d,V}(f) / G$ applying again Procedure PolynomialDivision.

Finally, as Assumption 1.2 holds, $\mathcal{F}(e_0, \dots, e_r) \neq 0$ and we obtain the normalized Chow form $\mathcal{C}h_{V \cap V(f)} = \mathcal{F} / \mathcal{F}(e_0, \dots, e_r)$.

Now let us calculate the number of points necessary to ensure that the algorithm has an error probability bounded by $1/4$:

First, in order to compute $\mathcal{C}h_{d,V}(f)$ we need $u \in \mathbb{Q}^{(r+1)(n+1)}$ such that $|M_0|(u) \neq 0$. But let us observe that in fact $|M_0|(e_1, \dots, e_r, U_r) \neq 0 \in \mathbb{Q}[U_r]$ (so it is enough to choose randomly $u_r \in \mathbb{Q}^{n+1}$ such that $|M_0|(e_1, \dots, e_r, u_r) \neq 0$). This is due to the fact that $a(e_1, \dots, e_r) = \mathcal{C}h_V(e_1, \dots, e_r, e_0) = \pm 1$. Thus Assumption 1.2 implies that $\mathcal{C}h_V(e_1, \dots, e_r, U_r) = \pm \mathcal{C}h_Z(U_r)$. Hence, $p_Z(U_r, t) := p(e_1, \dots, e_r)(U_r, t)$ is a characteristic polynomial of Z , whose discriminant does not vanish, and then the polynomial $|M_0|(e_1, \dots, e_r, U_r) \neq 0 \in \mathbb{Q}[U_r]$.

Now, as $\deg |M_0|(e_1, \dots, e_r, U_r) \leq D^2$, taking $u_r := \text{Random}(n+1, 12D^2)$ we infer that, with probability at least $1 - 1/12$, the point (e_1, \dots, e_r, u_r) is a good base point to apply Procedure PolynomialDivision and obtain $\mathcal{C}h_{d,V}(f)$.

Next we compute G applying Procedure GCD $\lceil 6(1 + \log 12) \rceil = 26$ times (see Remark 1.6) so that its error probability is at most $1/12$.

Finally as G is a polynomial of degree bounded by rdD in $r(n+1)$ variables, choosing $u := \text{Random}(r(n+1), 12rdD)$ we also guarantee that the probability that u is a good base point to perform the last division is at least $1 - 1/12$.

Thus, the error probability of the whole algorithm is at most $1/4$.

Now let us compute the (worst-case) complexity of this algorithm:

The whole complexity of computing the numerator and denominator in Identity (10) is of order $\mathcal{O}(n(d^2D^3L + D^4)) = \mathcal{O}(nd^2D^4L)$. By Lemma 1.7 the complexity of computing $\mathcal{C}h_{d,V}(f)$ is of order $\mathcal{O}((ndD)^2(ndD + nd^2D^4)L) = \mathcal{O}(n^3d^4D^6L)$.

Then, we apply Lemma 1.8 and Proposition 1.5 to compute an slp of length $\mathcal{O}(n^7d^8D^{10}L)$ for G of Identity (11) within complexity $\mathcal{O}(n^9d^8D^{10}L)$.

Finally, when we perform the last division, the overall complexity of computing $\mathcal{C}h_{V \cap V}(f)$ is of order $\mathcal{O}((ndD)^{12}L)$. \square

We summarize the algorithm in Procedure Intersection (Subroutine 9).

3.3 Separation of varieties

Let $V \subset \mathbb{A}^n$ be an equidimensional variety of dimension r . Let $g \in \mathbb{Q}[x_1, \dots, x_n] \setminus \{0\}$, and let Y be the union of the irreducible components of V contained in $V(g)$ and W be the union of the other ones. Hence Y and W are equidimensional varieties of dimension r such that $V = Y \cup W$, $Y \subset V(g)$, and g is not a zero divisor modulo $I(W)$.

The following procedure (Subroutine 10) computes the Chow forms of Y and W from a Chow form of V and the polynomial g .

For the sake of simplicity we assume that V —and therefore Y and W — satisfy Assumption 1.2.

Lemma 3.9 *Let $V \subset \mathbb{A}^n$ be an equidimensional variety of degree bounded by D which satisfies Assumption 1.2. Let $g \in \mathbb{Q}[x_1, \dots, x_n] \setminus \{0\}$ of degree bounded by d and Y and W defined as above. Assume that $\mathcal{C}h_V$ and g are encoded by slp's of length bounded by L .*

Then there is a bounded probability algorithm (Procedure Sep (Subroutine 10) below) which computes the Chow forms $\mathcal{C}h_Y$ and $\mathcal{C}h_W$ within (worst-case) complexity $\mathcal{O}((ndD)^8L)$.

Proof.— Let $\mathcal{P}_V \in \mathbb{Q}[U_0, \dots, U_r][T_0, \dots, T_r]$ be the normalized characteristic polynomial of V , as defined in Subsection 3.1 and set $P' := \partial \mathcal{P}_V / \partial T_0$.

We consider the following map, already introduced in Subsection 3.1:

$$\varphi_V : \mathbb{A}^{(r+1)(n+1)} \times V \rightarrow V(\mathcal{P}_V) \quad , \quad (u_0, \dots, u_r; \xi) \mapsto (u_0, \dots, u_r; L_0(u_0, \xi), \dots, L_r(u_r, \xi)).$$

By Lemma 3.1 φ_V is a birational map which in fact is an isomorphism when restricted to

$$U := (\mathbb{A}^{(r+1)(n+1)} \times V) \setminus V(P'(L_0, \dots, L_r)) \rightarrow \mathcal{U} := V(\mathcal{P}_V) \setminus V(P'),$$

Subroutine 9: Intersection with a hypersurface

procedure Intersection(n, r, D, f, d, Ch_V)

n is the number of variables,

r, D are the dimension and the degree of V respectively,

$f \in \mathbb{Q}[x_1, \dots, x_n]$ is a non-zero divisor modulo $I(V)$ of degree bounded by d ,

Ch_V is the normalized Chow form of V ,

The procedure returns the normalized Chow form $Ch_{V \cap V(f)}$ of the intersection variety $V \cap V(f)$.

1. $p := (-1)^D Ch_V(U_0, \dots, U_{r-1}, (U_{r0} - t, U_{r1}, \dots, U_{rn}))$;
2. $a := Ch_V(U_0, \dots, U_{r-1}, e_0)$;
3. $w := -\left(\frac{\partial p}{\partial U_{r1}}, \dots, \frac{\partial p}{\partial U_{rn}}\right)$;
4. $M_p := a \text{CompanionMatrix}(p/a)$;
5. **for** i **from** 1 **to** n **do**
6. $w_i^h := \text{Homog}(w_i, D)$;
7. $M_i := w_i^h(a, M_p)$;
8. **od**;
9. $(p')^h := \text{Homog}\left(\frac{\partial p}{\partial t}, D\right)$;
10. $M_0 := (p')^h(a, M_p)$;
11. $f^h := \text{Homog}(f, d)$;
12. $M_f := f^h(M_0, M_1, \dots, M_n)$;
13. $H_1 := |M_f|$;
14. $H_2 := |M_0|$;
15. $u_r := \text{Random}(n+1, 12D^2)$;
16. **if** $H_2(e_1, \dots, e_r, u_r) = 0$ **then**
17. **return**("error");
18. **else**
19. $Ch_{d,V}(f) := \text{PolynomialDivision}(a^d H_1, H_2^d, rdD, (e_1, \dots, e_r, u_r))$;
20. $G := \text{GCD}(Ch_{d,V}(f), \partial Ch_{d,V}(f) / \partial U_{00}, (U_0, \dots, U_{r-1}), rdD; 12)$;
21. $u := \text{Random}((r+1)(n+1), 12rdD)$;
22. **if** $G(u) = 0$ **then**
23. **return**("error");
24. **else**
25. $\mathcal{F} := \text{PolynomialDivision}(G, Ch_{d,V}(f), rDd, u)$;
26. $Ch_{V \cap V(f)} := \mathcal{F} / \mathcal{F}(e_0, \dots, e_r)$;
27. **return**($Ch_{V \cap V(f)}$);

end.

with inverse

$$\psi_V : (u_0, \dots, u_r; t_0, \dots, t_r) \mapsto \left(u_0, \dots, u_r; -\frac{1}{P'} \frac{\partial \mathcal{P}_V}{\partial U_{01}}, \dots, -\frac{1}{P'} \frac{\partial \mathcal{P}_V}{\partial U_{0n}}\right).$$

Define

$$G := (P')^d \psi_V^*(g) = g^h \left(P', -\frac{\partial \mathcal{P}_V}{\partial U_{01}}, \dots, -\frac{\partial \mathcal{P}_V}{\partial U_{0n}} \right),$$

where $g^h := \text{Homog}(g, d)$. Thus φ_V induces an isomorphism between $V(g) \cap U$ and $V(G) \cap \mathcal{U}$. Hence $V(\mathcal{P}_Y)$ equals the union of the components in $V(\mathcal{P}_V)$ which are contained in the hypersurface $V(G) \subset \mathbb{A}^{(r+1)(n+1)+(r+1)}$, and $V(\mathcal{P}_W)$ is the union of the other ones. As \mathcal{P}_V is a squarefree polynomial we conclude that

$$\mathcal{P}_Y := \gcd(G, \mathcal{P}_V) \quad , \quad \mathcal{P}_W = \frac{\mathcal{P}_V}{\gcd(G, \mathcal{P}_V)},$$

and therefore, from Identity (6) of Section 3.1, we obtain that

$$\mathcal{F}_Y = \mathcal{P}_Y(U)(0) \quad \text{and} \quad \mathcal{F}_W = \mathcal{P}_W(U)(0)$$

are Chow forms of Y and W respectively.

Note that as $\mathcal{P}_Y | \mathcal{P}_V$, $\mathcal{P}_Y(e_0, \dots, e_r, 0, \dots, 0) \neq 0$, thus $e := (e_0, \dots, e_r, 0, \dots, 0)$ is a good base point to apply Procedure PolynomialDivision. Thus the only probability step of this algorithm is the computation of the Greatest Common Divisor between \mathcal{P}_V and G .

Now we estimate the (worst-case) complexity of the algorithm:

The characteristic polynomial \mathcal{P}_V can be computed from $\mathcal{C}h_V$ with complexity $\mathcal{O}(L)$ using Identity (6) in Section 3.1. Its partial derivatives with respect to T_0 and U_{01}, \dots, U_{0n} can be computed within complexity $\mathcal{O}(nL)$.

The polynomial G is obtained within complexity $\mathcal{O}(d^2(d+nL))$.

As $\deg \mathcal{P}_V = (r+1)D$ and $\deg G \leq d((r+1)D-1)$, both bounded by $(r+1)dD$, the gcd computation of \mathcal{P}_V requires $(ndD)^6(d+L)$ additional arithmetic operations.

From Lemma 1.7, the polynomial division for \mathcal{P}_W is then performed within complexity $\mathcal{O}((ndD)^8L)$.

The final specialization $T \mapsto 0$ does not change this estimate.

Therefore, the (worst-case) complexity of the algorithm is of order $\mathcal{O}((ndD)^8L)$. \square

We summarize the algorithm in Procedure Sep (Subroutine 10).

Remark 3.10 *In case the variety V does not satisfy Assumption 1.2, this procedure can be modified within the same bounds of complexity so that, from a Chow form of V , we obtain Chow forms of W and Y . The only problem that may appear in the previous lemma is that $\mathcal{P}_Y(e)$ may be zero and we will not be able to accomplish the polynomial division. To solve this, we can modify Subroutine 10 in the following way: we choose a random point so that we can apply the polynomial division subroutine with error probability bounded by $\frac{1}{8}$ and we change the error probability of the GCD computation also by $\frac{1}{8}$ (by repeating it several times) in order that the error probability of the whole procedure is still bounded by $\frac{1}{4}$.*

3.4 Equations in general position

The algorithm we construct in Subsection 3.5 works under some genericity hypotheses on the input polynomial system. This is one of the main reasons—but not the only one—for the introduction of non-determinism in our algorithm: there are no known efficient deterministic procedures to obtain these hypotheses from a given system. In order to achieve them we replace the system and the variables by *random* linear combinations. Effective versions of Bertini's and Noether normalization theorems enable us to estimate the probability of success of this preprocessing.

The complexity of our algorithm is controlled by the geometric degree of the input system, that is the maximum degree of the varieties successively cut out by the equations obtained by this preprocessing. To define this parameter, which is a suitable generalization of the geometric degree of a 0-dimensional system introduced in [26], we first give the following definition:

Subroutine 10: Separation of varieties

procedure Sep(n, r, D, g, d, Ch_V)

n is the number of variables,
r, D are the dimension and an upper bound for the degree of V respectively,
$g \in \mathbb{Q}[x_1, \dots, x_n] \setminus \{0\}$,
d is a bound for the degree of g ,
Ch_V is the normalized Chow form of V .
The procedure returns the normalized Chow forms Ch_Y and Ch_W .

1. $\mathcal{P}_V := Ch_V((U_{00} - T_0, U_{01}, \dots, U_{0r}), \dots, (U_{r0} - T_r, U_{r1}, \dots, U_{rn}))$;
2. $g^h := \text{Homog}(g, d)$;
3. $G := g^h \left(P', -\frac{\partial \mathcal{P}_V}{\partial U_{01}}, \dots, -\frac{\partial \mathcal{P}_V}{\partial U_{0n}} \right)$;
4. $\mathcal{P}_Y := \text{GCD}(G, \mathcal{P}_V, (U_0, \dots, U_r, T_0, \dots, T_r), (r+1)dD)$;
5. $\mathcal{P}_W := \text{PolynomialDivision}(\mathcal{P}_Y, \mathcal{P}_V, (r+1)D, (e_0, \dots, e_r; 0, \dots, 0))$;
6. $Ch_Y := \mathcal{P}_Y(U)(0)/\mathcal{P}_Y(e_0, \dots, e_r; 0, \dots, 0)$;
7. $Ch_W := \mathcal{P}_W(U)(0)/\mathcal{P}_W(e_0, \dots, e_r; 0, \dots, 0)$;
8. **return**(Ch_Y, Ch_W);

end.

Definition 3.11 Let $g \in \mathbb{Q}[x_0, \dots, x_n]$ be a homogeneous polynomial, $I_g \subset \mathbb{Q}[x_0, \dots, x_n]_g$ be a homogeneous ideal, and $V \subset \mathbb{P}^n$ be a projective variety. We say that I_g is radical of dimension r outside $V_g := V \setminus V(g)$ if every primary component \mathcal{Q} of I_g such that $V(\mathcal{Q})_g := V(\mathcal{Q}) \setminus V(g) \not\subset V_g$ is prime of dimension r .

An analogous definition holds for an ideal in $\mathbb{Q}[x_1, \dots, x_n]_g$ and an affine variety $V_g \subset \mathbb{A}_g^n$.

Let $f_1, \dots, f_s, g \in \mathbb{Q}[x_0, \dots, x_n]$ be homogeneous polynomials of degree bounded by d , and set $V_g = V(f_1, \dots, f_s)_g \subset \mathbb{P}_g^n$. We assume that $V_g \neq \mathbb{P}_g^n$, that is $f_j \neq 0$ for some j . We also assume w.l.o.g $\deg f_j = d$ for every j : if this is not the case, we replace the input system by

$$x_i^{d-\deg f_j} f_j, \quad 0 \leq i \leq n, 1 \leq j \leq s.$$

For $a_1, \dots, a_{n+1} \in \mathbb{Q}^s$ we set

$$Q_i(a_i) := a_{i1} f_1 + \dots + a_{is} f_s$$

for the associated linear combination of f_1, \dots, f_s , which —by the assumption that $\deg f_j = d$ — is also a system of homogeneous polynomials of degree d .

Let Δ be the set of all $(n+1) \times s$ -matrices $A = (a_1, \dots, a_{n+1})^t \in \mathbb{Q}^{(n+1) \times s}$ such that the ideals $I_i(A) := (Q_1(a_1), \dots, Q_i(a_i)) \subset \mathbb{Q}[x_0, \dots, x_n]$, $1 \leq i \leq n+1$, satisfy:

- $V(I_{n+1}(A))_g = V_g$ in \mathbb{P}_g^n .
- For $1 \leq i \leq n$, if $V(I_i(A))_g \neq V_g$, then $I_i(A)_g$ is a radical ideal of dimension $n - i$ outside V_g .

(These are the first genericity hypotheses the polynomials should verify in order that our algorithm work.)

For every $A \in \Delta$ we set $\delta(A) := \max\{\deg V(I_i(A))_g; 1 \leq i \leq n+1\}$.

Definition 3.12 Keeping these notations, the geometric degree of the system

$$f_1 = 0, \dots, f_s = 0, \quad g \neq 0$$

is defined as

$$\delta := \delta(f_1, \dots, f_s; g) := \max\{\delta(A); A \in \Delta\}.$$

Note that the Bézout inequality implies $\delta \leq d^n$.

Remark 3.13 For a system of polynomials $F_1, \dots, F_s, G \in \mathbb{Q}[x_1, \dots, x_n]$ (not necessarily homogeneous) of degree bounded by d , the affine analogue δ_{aff} of the geometric degree is defined in exactly the same manner, but without preparing the polynomials to make their degrees coincide.

In fact, if for $1 \leq i \leq s$, $d_i := \deg F_i$, $d := \max_i d_i$ and $F_i^h, G^h \in \mathbb{Q}[x_0, \dots, x_n]$ are the homogenizations of F_i and G respectively, then

$$\delta_{\text{aff}}(F_1, \dots, F_s; G) = \delta(x_0^{d-d_1} F_1^h, \dots, x_0^{d-d_s} F_s^h; x_0 G^h).$$

Let

$$V_g = V_0 \cup \dots \cup V_{n-1}$$

be the equidimensional decomposition of V_g in \mathbb{P}_g^n , where V_i is either empty or of dimension i , and let $A = (a_1, \dots, a_{n+1})^t \in \Delta$.

For $i = 1, \dots, n+1$, as $I_i(A) \subseteq (f_1, \dots, f_s)$, $V_g \subseteq V(I_i(A))_g$ always holds. Moreover, if $V(I_i(A))_g = V_g$ for some i , then $V(I_j(A))_g = V_g$ for all $j \geq i$. Also, observe that the ideal $I_i(A)$ is generated by i polynomials, so every irreducible component of $V(I_i(A))$ has dimension at least $n - i$. Thus, we infer that for $r := n - i$, $0 \leq r \leq n - 1$, we have

$$V(I_{n-r}(A))_g = V'_r \cup V_r \cup \dots \cup V_{n-1},$$

where V'_r is an equidimensional variety of dimension r . (We set $V'_r = \emptyset$ for every r such that $V(I_{n-r}(A))_g = V_g$ since in these cases $V_g = V_r \cup \dots \cup V_{n-1}$.)

From now on, $Q_i(a_i)$ will be denoted simply by Q_i .

The condition that $A \in \Delta$ implies that, in case $V'_r \neq \emptyset$, Q_{n-r+1} is not a zero divisor modulo $I(V'_r)_g$. In this case, we have

$$\begin{aligned} V'_{r-1} \cup V_{r-1} \cup V_r \cup \dots \cup V_{n-1} &= V(Q_1, \dots, Q_{n-r+1})_g \\ &= (V'_r \cup V_r \cup \dots \cup V_{n-1}) \cap V(Q_{n-r+1}) \\ &= (V'_r \cap V(Q_{n-r+1})) \cup V_r \cup \dots \cup V_{n-1}, \end{aligned}$$

as for all i , $V_i \subset V(Q_{n-r+1})$. Hence, since $\dim(V'_r \cap V(Q_{n-r+1})) = r - 1$, we deduce that

$$V'_r \cap V(Q_{n-r+1}) = V'_{r-1} \cup V_{r-1} \cup \tilde{V}_{r-1}, \quad (12)$$

where

$$\tilde{V}_{r-1} = \bigcup \{C; C \text{ component of } V'_r \cap V(Q_{n-r+1}) \cap (V_r \cup \dots \cup V_{n-1}) \text{ of dimension } r - 1\}$$

is an equidimensional subvariety of $V_r \cup \dots \cup V_{n-1}$ of dimension $r - 1$. We set $\tilde{V}_{n-1} := \emptyset$ and $V'_{-1} := \emptyset$.

Now for $b_0, \dots, b_n \in \mathbb{Q}^{n+1}$ we consider the linear change of variables

$$y_k(b_k) := b_{k0} x_0 + \dots + b_{kn} x_n \quad , \quad 0 \leq k \leq n.$$

We say that (b_0, \dots, b_n) is *admissible* if, under this linear change of variables, for $0 \leq r \leq n - 1$,

- the varieties $V'_r \cup V_r \cup \tilde{V}_r$ satisfy Assumption 1.2,

- the polynomials $Q_1, \dots, Q_{n-r} \in I(V'_r)_g$ are a system of local equations of V'_r at $Z_r := V'_r \cap V(y_1, \dots, y_r)$.

We construct the polynomials Q_1, \dots, Q_{n+1} and the variables y_0, \dots, y_n by choosing the coefficient vectors a_i , $1 \leq i \leq n+1$, and b_k , $0 \leq k \leq n$, at random in a given set. In what follows we estimate the error probability of this procedure:

Lemma 3.14 *Let notation be as in the previous paragraphs. Let N be a positive integer and let*

$$a_i \in [0, 8N(d+1)^{2n}]^s \quad 1 \leq i \leq n+1, \quad , \quad b_k \in [0, 2Nn^2d^{2n}]^{n+1} \quad 0 \leq k \leq n$$

be chosen at random. Then the error probability of $A := (a_1, \dots, a_{n+1})^t$ being in Δ and (b_0, \dots, b_n) being admissible is bounded by $1/N$.

Proof.– The set of matrices Δ contains in fact a non-empty open set of $\mathbb{Q}^{(n+1) \times s}$; by the effective Bertini's theorem in [44, Lemmas 1 and 2] or a local version of [41, Proposition 4.3 and Corollary 4.4] there is a non-zero polynomial F with $\deg F \leq 4(d+1)^{2n}$ such that $F(a_1, \dots, a_{n+1}) \neq 0$ implies that $A = (a_1, \dots, a_{n+1})^t \in \Delta$.

Assume now that $A \in \Delta$. By the effective Noether theorem version of [41, Proposition 4.5] there is a non-zero polynomial $G \in k[U_0, \dots, U_n]$ with

$$\deg G \leq 2 \sum_{r=0}^{n-1} r \deg(V'_r \cup V_r \cup \tilde{V}_r)^2$$

such that $G(b_0, \dots, b_n) \neq 0$ implies that under the linear change of variables given by (b_0, \dots, b_n) , the varieties $V_r \cup V'_r \cup \tilde{V}_r$ satisfy Assumption 1.2. Since, from Identity (12),

$$\deg(V'_r \cup V_r \cup \tilde{V}_r) \leq d \deg V_{r+1} \leq d^n,$$

$$\deg G \leq n(n-1)d^{2n}.$$

Now we will define a polynomial $H \in k[U_1, \dots, U_{n-1}]$ such that $H(b_1, \dots, b_{n-1}) \neq 0$ implies that the second condition for admissibility holds.

Fix r , $0 \leq r \leq n-1$. We know that $(Q_1, \dots, Q_{n-r})_g$ is a radical ideal of dimension r outside V_g whose associated variety coincides with V'_r outside V_g . Thus, localizing at any $\xi \in V'_r$, $\xi \notin V_g$, we get $((Q_1, \dots, Q_{n-r})_g)_\xi = I(V'_r)_\xi$, that is, Q_1, \dots, Q_{n-r} is a system of local equations of V'_r at ξ . Therefore, it suffices to take new variables y_0, \dots, y_n such that $V'_r \cap V_g \cap V(y_1, \dots, y_r) = \emptyset$.

From the definition of V'_0 , it is clear that $V'_0 \cap V_g = \emptyset$.

For $1 \leq r \leq n-1$, as V_g is definable by polynomials of degrees bounded by d and no irreducible component of V'_r is contained in V_g , there exists a polynomial $g_r \in k[x_1, \dots, x_n]$ with $\deg(g_r) \leq d$ such that $V_g \subset V(g_r)$ and $V'_r \cap V(g_r)$ is equidimensional of dimension $r-1$. Let $\mathcal{F}_r \in k[U_1, \dots, U_r]$ be a Chow form of $V'_r \cap V(g_r)$.

Set $H := \prod_{r=1}^{n-1} \mathcal{F}_r \in k[U_1, \dots, U_{n-1}]$. The condition $H(b_1, \dots, b_{n-1}) \neq 0$ implies that, for every $1 \leq r \leq n-1$, $\mathcal{F}_r(b_1, \dots, b_r) \neq 0$ and so,

$$V'_r \cap V_g \cap V(y_1(b_1), \dots, y_r(b_r)) \subset V'_r \cap V(g_r) \cap V(y_1(b_1), \dots, y_r(b_r)) = \emptyset.$$

Observe that H is a non-zero polynomial with

$$\deg H = \sum_{r=1}^{n-1} \deg \mathcal{F}_r = \sum_{r=1}^{n-1} \deg V'_r \cap V(g_r) \leq \sum_{r=1}^{n-1} d \deg V'_r \leq \sum_{r=1}^{n-1} d^{n-r+1} \leq (n-1)d^n.$$

Therefore, there exists a non-zero polynomial condition of degree bounded by $n(n-1)d^{2n} + (n-1)d^n \leq n^2d^{2n}$ which ensures that the matrix (b_0, \dots, b_n) is admissible.

The conclusion follows as usual from the Zippel-Schwartz test. \square

3.5 Proof of Theorem 1

Let $f_1, \dots, f_s, g \in \mathbb{Q}[x_0, \dots, x_n]$ be homogeneous polynomials of degree bounded by d , and set $V_g := V(f_1, \dots, f_s)_g \subseteq \mathbb{P}_g^n$. Set δ for the geometric degree of the system $f_1 = 0, \dots, f_s = 0, g \neq 0$.

The algorithm is iterative and consists of two main steps, besides the preparation of the input (equations and variables).

The preparation of the input enables us to work with an affine variety W instead of the input quasiprojective variety V_g and local systems of equations of certain auxiliary varieties appearing in the process.

The first main step computes recursively the Chow forms of a non-minimal equidimensional decomposition of W . Here the crucial point which controls the explosion of the complexity is that the size of the input of an iteration does not depend on the size of the output of the previous step: the input of each recursive step has the same controlled size.

The second main step clears out extra components and computes the Chow forms of the equidimensional components of the minimal decomposition of W from which the Chow forms of the equidimensional components of V_g are obtained straightforwardly.

This is a bounded error probability algorithm whose expected complexity is of order $s(nd\delta)^{\mathcal{O}(1)}L$. Its worst-case complexity is $s(nd^n)^{\mathcal{O}(1)}L$.

For the rest of this proof, we set $N := d^{56n}$.

Input Preparation

Set $V_g = V_0 \cup \dots \cup V_n$ for the minimal equidimensional decomposition of V_g , where each V_r is either empty or of pure dimension r .

First, applying Procedure Deg described at the end of Section 1.3 to f_1, \dots, f_s , we compute with error probability bounded by $1/6N$ the exact degree of the polynomials f_1, \dots, f_s within complexity $\mathcal{O}(sd^2L + n \log(sdN))$. This also states whether these polynomials are the zero polynomial and, therefore, whether $V_g = \mathbb{P}_g^n$. In that case $\mathcal{F}_{V_n} = |(U_0, \dots, U_n)|$ and for $i < n$, $\mathcal{F}_{V_i} = 1$.

Thus, with error probability bounded by $1/6N$ we can assume we know the exact degree of the polynomials f_1, \dots, f_s , and that $V_n = \emptyset$ and $\dim V_g \leq n - 1$.

We consider the polynomials

$$\tilde{f}_{ij} := x_i^{d - \deg f_j} f_j, \quad 0 \leq i \leq n, \quad 1 \leq j \leq s;$$

hence we have now $t \leq (n + 1)s$ polynomials \tilde{f}_{ij} of degree d , that we rename $\tilde{f}_1, \dots, \tilde{f}_t$.

We apply Lemma 3.14 to choose randomly a matrix $A = (a_1, \dots, a_{n+1})^t \in \mathbb{Q}^{(n+1) \times t}$ and a matrix $B = (b_0, \dots, b_n) \in \mathbb{Q}^{(n+1) \times (n+1)}$ such that the error probability that $A \in \Delta$ and B is admissible is bounded by $1/6N$.

We can assume thus that the linear combinations $(Q_1, \dots, Q_{n+1}) = A(\tilde{f}_1, \dots, \tilde{f}_t)$ satisfy

$$V_g = V(Q_1, \dots, Q_{n+1})_g$$

and, for $0 \leq r \leq n - 1$,

- $(Q_1, \dots, Q_{n-r})_g$ is either empty outside V_g or a radical ideal of dimension r outside V_g .
- $V(Q_1, \dots, Q_{n-r})_g = V'_r \cup V_r \cup \dots \cup V_{n-1}$, where V'_r is either empty or an equidimensional variety of dimension r with no irreducible component included in $V_r \cup \dots \cup V_{n-1}$.
- $V'_r \cap V(Q_{n-r+1})_g = V'_{r-1} \cup V_{r-1} \cup \tilde{V}_{r-1}$, where \tilde{V}_{r-1} is either empty or an equidimensional variety of dimension $r - 1$ included in $V_r \cup \dots \cup V_{n-1}$. We set $V'_n := \mathbb{P}_g^n$ to extend this property to $r = n$.

We can assume moreover that the change of coordinates $y = Bx$ verifies

- $BV'_r \cup BV_r \cup B\widetilde{V}_r$ satisfies Assumption 1.2.
- $Q_1(B^{-1}y), \dots, Q_{n-r}(B^{-1}y)$ is a system of local equations of BV'_r at $BV'_r \cap V(y_1, \dots, y_r)$.

The complexity of constructing the random matrices A and B and the inverse of the matrix B is of order $\mathcal{O}(sn^4(\log N + \log d))$.

Now, Assumption 1.2 implies that the varieties have no irreducible component at infinity. Hence we restrict to the affine space: we set $y_0 = 1$ and denote by q_1, \dots, q_{n+1}, h the set of polynomials in the new variables obtained from Q_1, \dots, Q_{n+1}, g , that is:

$$(q_1, \dots, q_{n+1}) = AF(B^{-1}(1, y_1, \dots, y_n)) \quad , \quad h = g(B^{-1}(1, y_1, \dots, y_n)),$$

where $F := (\tilde{f}_1, \dots, \tilde{f}_t)$.

We define

$$W := \overline{V(q_1, \dots, q_{n+1})_h} = \overline{BV \cap \mathbb{A}_h^n} \subset \mathbb{A}^n.$$

Let $W = W_0 \cup \dots \cup W_{n-1}$ be the minimal equidimensional decomposition of W , where for $0 \leq r \leq n-1$, W_r is either empty or of dimension r , and let W'_r and \widetilde{W}_r defined by the same construction as V'_r and \widetilde{V}_r before, that is

- $V(q_1, \dots, q_{n-r})_h = W'_r \cup W_r \cup \dots \cup W_{n-1}$
- $W'_r \cap V(q_{n-r+1})_h = W'_{r-1} \cup W_{r-1} \cup \widetilde{W}_{r-1}$

As the identity

$$W_r = \overline{B\widetilde{V}_r \cap \mathbb{A}_h^n}$$

holds, from a Chow form of W_r we obtain a Chow form of the corresponding V_r by means of the change of variables:

$$\mathcal{F}_{V_r}(U_0, \dots, U_r) = \mathcal{C}h_{W_r}(U_0B^{-1}, \dots, U_rB^{-1}).$$

We observe that $W'_r = \overline{BV'_r \cap \mathbb{A}_h^n}$, and then q_1, \dots, q_{n-r} is a system of local equations of W'_r at $W'_r \cap V(y_1, \dots, y_r)$.

The error probability of this preparation step is bounded by $1/3N$. Once the matrices A and B are fixed, we have that the complexity of computing the polynomials q_1, \dots, q_{n+1}, h is of order $\mathcal{O}(sn^2dL)$, as well as their length.

First Main Step

From $r = n-1$ to 0, the algorithm computes the Chow form of $W_r \cup \widetilde{W}_r$ and a geometric resolution of the fiber $Z_r := W'_r \cap V(y_1, \dots, y_r)$ (which also gives the degree D_r of W'_r). The former will be the input of the second main step while the latter is the input of the next step in this recursion. Each step of this recursion is a bounded probability algorithm whose error probability is bounded by $1/3nN$ provided that the input of the iteration step was correct.

We begin with the fiber $Z_n = V(y_1, \dots, y_n) = (0, \dots, 0)$ and its geometric resolution $(t, (t, \dots, t))$ associated to $\ell = x_1$. We also set $D_n := 1$.

Now, we are going to describe a step of the recursion. From a geometric resolution of Z_{r+1} we compute a Chow form for $W_r \cup \widetilde{W}_r$ and a geometric resolution of Z_r , which is the input of the next recursive step. Set D_{r+1} for the given estimate of $\deg W'_{r+1}$.

- Computation of $\mathcal{C}h_{W'_{r+1}}$:

From the geometric resolution $(p_{r+1}, (v_1, \dots, v_n))$ associated to the affine linear form ℓ_{r+1} of Z_{r+1} , and the system of local equations q_1, \dots, q_{n-r-1} of W'_{r+1} at Z_{r+1} , we compute the Chow form of W'_{r+1} applying Procedure ChowForm (Subroutine 7). This step of the algorithm is deterministic and computes $\mathcal{C}h_{W'_{r+1}}$ provided that the polynomials and variables satisfy the genericity conditions and that the geometric resolution of Z_{r+1} is accurate. Observe that by

Main Lemma 2.3 applied to the local system of equations q_1, \dots, q_{n-r-1} of degree d and length $\mathcal{O}(sn^2dL)$, the complexity and the length of the output are both of order

$$L(\mathcal{C}h_{W'_{r+1}}) = \mathcal{O}((r+1)^8 \log_2((r+1)D_{r+1})n^7 d^2 D_{r+1}^{11} (sn^2dL)) = \mathcal{O}(sn^6(ndD_{r+1})^{12}L).$$

- Computation of $\mathcal{C}h_{W'_{r+1} \cap V(q_{n-r})}$:

Now we apply sufficiently many times Procedure Intersection (Subroutine 9) to compute the Chow form of $W'_{r+1} \cap V(q_{n-r})$ with error probability bounded by $1/18nN$: by Lemma 3.8, the length of the output Chow form and the complexity of one iteration are both of order

$$L(\mathcal{C}h_{W'_{r+1} \cap V(q_{n-r})}) = \mathcal{O}((ndD_{r+1})^{12}L(\mathcal{C}h_{W'_{r+1}})) = \mathcal{O}(sn^6(ndD_{r+1})^{24}L),$$

while, from Corollary 1.6 for the choice $s = \lceil 6(\log(18nN) + 1) \rceil$, the complexity of this step is of order

$$\begin{aligned} \mathcal{O}(((r+1)(n+1)+1) \log(18nN)(L(\mathcal{C}h_{W'_{r+1}}) + L(\mathcal{C}h_{W'_{r+1} \cap V(q_{n-r})})) + \log^2(18nN)) &= \\ = \mathcal{O}(\log^2(N) s n^9 (ndD_{r+1})^{12}L). \end{aligned}$$

- Computation of $\mathcal{C}h_{W_r \cup W_r \cup \widetilde{W}_r}$:

Observe that each irreducible component of $W'_{r+1} \cap V(q_{n-r})$ is either an irreducible component of $W'_r \cup W_r \cup \widetilde{W}_r$ or an irreducible variety included in $V(h)$. Therefore, we apply sufficiently many times Procedure Sep (Subroutine 10) to compute the Chow form of $W'_r \cup W_r \cup \widetilde{W}_r$ with error probability bounded by $1/18nN$: by Lemma 3.9, the length of the output Chow form and the complexity of one iteration are both of order

$$L(\mathcal{C}h_{W'_r \cup W_r \cup \widetilde{W}_r}) = \mathcal{O}((nd(dD_{r+1}))^8 L(\mathcal{C}h_{W'_{r+1} \cap V(q_{n-r})})) = \mathcal{O}(sn^6 d^8 (ndD_{r+1})^{32}L),$$

while the complexity of this step is of order

$$\mathcal{O}(\log^2(N) sn^9 d^8 (ndD_{r+1})^{32}L).$$

- Computation of $\mathcal{C}h_{W'_r}$ and $\mathcal{C}h_{W_r \cup \widetilde{W}_r}$:

Next, since $V(q_{n-r+1})$ contains $W_r \cup \widetilde{W}_r$ but does not contain any component of W'_r , we use q_{n-r+1} to separate $\mathcal{C}h_{W'_r}$ from $\mathcal{C}h_{W_r \cup \widetilde{W}_r}$. We apply sufficiently many times Procedure Sep (Subroutine 10) to compute the Chow forms of W'_r and $W_r \cup \widetilde{W}_r$ with error probability bounded by $1/18nN$: the length of the output Chow forms and the complexity of one iteration are both of order

$$L(\mathcal{C}h_{W'_r}, \mathcal{C}h_{W_r \cup \widetilde{W}_r}) = \mathcal{O}((nd(dD_{r+1}))^8 L(\mathcal{C}h_{W'_{r+1} \cap V(q_{n-r})})) = \mathcal{O}(sn^6 d^{16} (ndD_{r+1})^{40}L),$$

while the complexity of this step is of order

$$\mathcal{O}(\log^2(N) sn^9 d^{16} (ndD_{r+1})^{40}L).$$

- Computation of a geometric resolution of $Z_r := W'_r \cap V(y_1, \dots, y_r)$:

We apply here Procedure GeomRes (Subroutine 8). It requires a random choice of the coefficients of a separating linear form ℓ_r . We do that in order to ensure that the error probability is $1/6nN$. The condition that a linear form separates the points of the fiber Z_r is given by a polynomial of degree bounded by $\binom{\deg Z_r}{2} \leq \frac{d^{2(n-r)}}{2}$ as $\deg Z_r \leq d^{n-r}$. So we choose the set of coefficients of ℓ_r in $[0, 3nNd^{2(n-r)})^{n+1}$. The complexity of constructing these coefficients is thus of order $\mathcal{O}((n+1)(\log(nN) + (n-r)\log d)) = \mathcal{O}(n^2(\log N + \log d))$ and the complexity of computing afterwards the geometric resolution of Z_r (that is, all its constant coefficients) adds, as $D_r \leq dD_{r+1}$,

$$\mathcal{O}(n(dD_{r+1})^2 L(\mathcal{C}h_{W'_r}) + d^4 D_{r+1}^4) = \mathcal{O}(sn^5 d^{16} (ndD_{r+1})^{42}L)$$

operations.

Summarizing, from the geometric resolution of Z_{r+1} and the polynomials q_1, \dots, q_{n-r} , the algorithm produces, within complexity $\mathcal{O}(\log^2(N)sn^7d^{16}(ndD_{r+1})^{42}L)$, all the coefficients of the geometric resolution of Z_r and an slp of length $\mathcal{O}(sn^6d^{16}(ndD_{r+1})^{40}L)$ for the Chow form of $W_r \cup \widetilde{W}_r$. The error probability that the computed objects are not the correct ones, provided that the input was right, is bounded by $1/3nN$.

Therefore, provided that the input preparation was correct, this algorithm is expected to compute $\mathcal{C}h_{W_r \cup \widetilde{W}_r}$, for $0 \leq r \leq n-1$, with error probability bounded by $1/3N$, within complexity of order

$$\mathcal{O}(\log^2(N)sn^7d^{16}(\sum_{k=r+1}^{n-1}(ndD_k)^{42})L),$$

and, by the iterative character of the algorithm, to compute all $\mathcal{C}h_{W_r \cup \widetilde{W}_r}$, $0 \leq r \leq n-1$, within the same complexity as that of computing $\mathcal{C}h_{W_0 \cup \widetilde{W}_0}$.

Second Main Step

For $0 \leq r \leq n-1$, in order to extract from the Chow form $\mathcal{C}h_{W_r \cup \widetilde{W}_r}$ the factor $\mathcal{C}h_{W_r}$, we define a hypersurface $V(G_r)$ such that, probabilistically, \widetilde{W}_r is exactly the union of all the irreducible components of $W_r \cup \widetilde{W}_r$ contained in $V(G_r)$, and then we apply Procedure Sep (Subroutine 10) to compute $\mathcal{C}h_{W_r}$.

Fix k , $1 \leq k \leq n-1$. We define a polynomial $H_k \in \mathbb{Q}[y_1, \dots, y_n]$ such that, with error probability bounded by $1/6(n-1)N$, the following conditions hold:

1. $W_k \cup \widetilde{W}_k \subseteq V(H_k)$,
2. no irreducible component of W_r is contained in $V(H_k)$ for $r = 0, \dots, k-1$.

Let \mathcal{P} be the characteristic polynomial of $W_k \cup \widetilde{W}_k$. For any affine linear form $\ell_0 = L_0(c_0, x)$, we have that $H_k := \mathcal{P}(c_0, e_1, \dots, e_k)(\ell_0, y_1, \dots, y_k)$ vanishes on $W_k \cup \widetilde{W}_k$. We determine now randomly ℓ_0 such that Condition 2 holds with error probability bounded by $1/6(n-1)N$. This is a standard argument that can be found for instance in [24, Section 2.3.5]:

For any irreducible component C of $W_0 \cup \dots \cup W_{k-1}$ there exists $\xi_C := (\xi_1^C, \dots, \xi_n^C) \in C - (W_k \cup \widetilde{W}_k)$. Now, if a linear form ℓ_0 satisfies that for any $\xi \in (W_k \cup \widetilde{W}_k) \cap V(y_1 - \xi_1^C, \dots, y_k - \xi_k^C)$ (which is a zero-dimensional variety of degree bounded by $d\delta$), $\ell_0(\xi) \neq \ell_0(\xi_C)$, then

$$\mathcal{P}(c_0, e_1, \dots, e_k)(\ell_0(\xi_C), \xi_1^C, \dots, \xi_k^C) \neq 0.$$

Hence C is not included in $V(H_k)$.

The condition to be satisfied is thus given by

$$\prod_{C, \xi} (\ell_0(\xi) - \ell_0(\xi_C)) \neq 0,$$

where C runs over the irreducible components of $W_0 \cup \dots \cup W_{k-1}$ and $\xi \in (W_k \cup \widetilde{W}_k) \cap V(y_1 - \xi_1^C, \dots, y_k - \xi_k^C)$. The polynomial has degree bounded by $d\delta^2 \leq d^{2n+1}$ since $\deg W_0 \cup \dots \cup W_{k-1} \leq \delta$. Choosing $c_0 := (0, c_{01}, \dots, c_{0n}) \in [0, 6(n-1)Nd^{2n+1}]^n$, the probability that H_k does not satisfy Condition 2 is bounded by $1/6(n-1)N$. Therefore the probability that, for $1 \leq k \leq n-1$, at least one H_k does not satisfy Condition 2 is bounded by $1/6N$.

Now, for $r = 0, \dots, n-2$ we define $G_r := \prod_{k=r+1}^{n-1} H_k$. Clearly, as $\widetilde{W}_r \subset W_{r+1} \cup \dots \cup W_{n-1}$, G_r vanishes on \widetilde{W}_r by Condition 1. On the other hand, as, by Condition 2, no irreducible component of W_r is contained in $V(H_k)$ for $r+1 \leq k \leq n-1$, G_r splits W_r and \widetilde{W}_r .

For $1 \leq k \leq n-1$, $\deg H_k \leq dD_{k+1}$ and $L(H_k) = \mathcal{O}(L(\mathcal{C}h_{W_k \cup \widetilde{W}_k}))$ since we derive \mathcal{P} from the corresponding Chow form by Identity (6). Hence $L(H_k) = \mathcal{O}(sn^6d^{16}(ndD_{k+1})^{40}L)$. Thus, for

$0 \leq r \leq n-2$, $\deg G_r \leq d \sum_{k \geq r+1} D_{k+1}$ and

$$L(G_r) = \mathcal{O}(sn^6 d^{16} (\sum_{k \geq r+1} (ndD_{k+1})^{40})L).$$

The computation of all H_k , $1 \leq k \leq n-1$, involves the computation of the random coefficients of each linear form ℓ_0 , that is $\mathcal{O}(n^2(\log N + n \log d))$ operations for each one of them, plus the complexity of computing and specializing each characteristic polynomial. Thus the total complexity of computing all H_k is of order $\mathcal{O}(n^2 \log N + sn^6 d^{16} (\sum_{k \geq 2} (ndD_k)^{40})L)$. We conclude that the complexity of computing all G_r , $0 \leq r \leq n-2$, is also of the same order.

This algorithm is expected to compute the right polynomials G_0, \dots, G_{n-2} , provided that the Input Preparation and the First Main Step were correct, with error probability bounded by $1/6N$.

Now we apply sufficiently many times Procedure Sep (Subroutine 10) to $\mathcal{C}h_{W_r \cup \widetilde{W}_r}$ and G_r in order to compute $\mathcal{C}h_{W_r}$ with error probability bounded by $1/6nN$: the length of the output Chow forms and the complexity of one iteration are both of order

$$L(\mathcal{C}h_{W_r}) = \mathcal{O}((n(d \sum_{k=r+1}^{n-1} D_{k+1})(dD_{r+1}))^8 L(\mathcal{C}h_{W_r \cup \widetilde{W}_r}, G_r)) = \mathcal{O}(sn^7 d^{16} (nd\overline{D})^{56} L),$$

where $\overline{D} = \max\{D_k : 1 \leq k \leq n-1\}$, while the total complexity of computing all $\mathcal{C}h_{W_r}$ with error probability bounded by $1/6N$, provided that the polynomials G_0, \dots, G_{n-2} were correct, is of order

$$\mathcal{O}(\log^2(N)sn^{11}d^{16}(nd\overline{D})^{56}L + s \log^2(s) n^2 \log(d)L).$$

Thus, the total error probability of the second main step is bounded by $1/3N$.

Finally, the Chow form \mathcal{F}_{V_r} is obtained by changing variables back. This computation does not change the order of complexity involved.

The total error probability of the whole algorithm is bounded by $1/N$. Moreover, in case each of the random choices was right, $D_k \leq \delta$ for every k , and therefore the Chow forms \mathcal{F}_{V_r} of the equidimensional components V_r of V_g are encoded by slp's of length

$$L(\mathcal{F}_{V_r}) = \mathcal{O}(sn^7 d^{16} (nd\delta)^{56} L),$$

and computed within complexity

$$\mathcal{O}(\log^2(N)sn^{11}d^{16}(nd\delta)^{56}L).$$

Since in any case, $D_k \leq d^{n-k} \leq d^{n-1}$ for every $1 \leq k \leq n-1$, the worst-case complexity of the computation is of order

$$\mathcal{O}(\log^2(N)sn^{67}d^{16}d^{56n}L).$$

Therefore the expected complexity of the algorithm is

$$\mathcal{O}((1 - \frac{1}{N})(\log^2(N)sn^{11}d^{16}(nd\delta)^{56}L) + \frac{1}{N}(\log^2(N)sn^{67}d^{16}d^{56n}L)).$$

Fixing $N := d^{56n}$, we conclude that the expected complexity of our bounded probability algorithm is of order

$$\mathcal{O}(\log^2(d^{56n})sn^{11}d^{16}(nd\delta)^{56}L + \log^2(d^{56n})sn^{67}d^{16}L) = s(nd\delta)^{\mathcal{O}(1)}L,$$

while the error probability is bounded by $1/N$.

We summarize in Procedure Equidim (Subroutine 11) the algorithm underlying the Proof of Theorem 1.

Subroutine 11: Equidimensional decomposition

```

procedure Equidim( $n, d, f_1, \dots, f_s, g, x$ )
#  $f_1, \dots, f_s, g$  are homogeneous polynomials in  $\mathbb{Q}[x_0, \dots, x_n]$  and  $x := (x_0, \dots, x_n)$ ;
#  $d$  is an upper bound for the degrees of  $f_1, \dots, f_s, g$ .
# The procedure returns the Chow forms  $\mathcal{F}_{V_0}, \dots, \mathcal{F}_{V_n}$  of the equidimensional components
# of  $V_g := V(f_1, \dots, f_s)_g \subset \mathbb{P}_g^n$ .
1.  $N := d^{56n}$ ;
2.  $(d_1, \dots, d_s) := \text{Deg}(f_1, \dots, f_s, x, d; 6sN)$ ;
3. if  $(d_1, \dots, d_s) := (-1, \dots, -1)$  then
4.    $(\mathcal{F}_{V_0}, \dots, \mathcal{F}_{V_{n-1}}, \mathcal{F}_{V_n}) := (1, \dots, 1, |(U_0, \dots, U_n)|)$ ;
5. else
6.    $F := (x_0^{d-d_1} f_1, \dots, x_n^{d-d_1} f_1, \dots, x_0^{d-d_s} f_s, \dots, x_n^{d-d_s} f_s)$ ;
7.    $A := \text{RandomMatrix}(n+1, s(n+1), 48N(d+1)^{2n})$ ;
8.    $B := \text{RandomMatrix}(n+1, n+1, 12Nn^2 d^{2n})$ ;
9.    $(y_0, \dots, y_n) := B(x_0, \dots, x_n)$ ;
10.   $(q_1, \dots, q_{n+1}) := AF(B^{-1}(1, y_1, \dots, y_n))$ ;
11.   $h := g(B^{-1}(1, y_1, \dots, y_n))$ ;
12.   $\mathcal{F}_{V_n} := 1$ ;
13.   $(c^{(n)}, D_n, p_n, v^{(n)}) := (e_1, 1, t, (t, \dots, t))$ ;
14.  for  $i$  from 1 to  $n$  do
15.     $r := n - i$ ;
16.     $\mathcal{C}h_{W'_{r+1}} := \text{ChowForm}(n, r+1, D_{r+1}, c^{(r+1)}, p_{r+1}, v^{(r+1)}, q_1, \dots, q_{n-r-1}, d)$ ;
17.     $\mathcal{F} := \text{Intersection}(n, r+1, D_{r+1}, q_{n-r}, d, \mathcal{C}h_{W'_{r+1}}; 18nN)$ ;
18.     $\mathcal{C}h_{W'_r \cup W_r \cup \widetilde{W}_r} := (\text{Sep}(n, r+1, dD_{r+1}, h, d, \mathcal{F}; 18nN))_2$ ;
19.     $(\mathcal{C}h_{W'_r}, \mathcal{C}h_{W_r \cup \widetilde{W}_r}) := \text{Sep}(n, r, dD_{r+1}, d, \mathcal{C}h_{W'_r \cup W_r \cup \widetilde{W}_r}, q_{n-r+1}; 18nN)$ ;
20.     $c^{(r)} := \text{Random}(n+1, 3nNd^{2(n-r)})$ ;
21.     $(D_r, p_r, v^{(r)}) := \text{GeomRes}(n, r, dD_{r+1}, \mathcal{C}h_{W'_r}, (0, \dots, 0), c^{(r)})$ ;
22.  od;
23.  for  $k$  from 0 to  $n-1$  do
24.     $\mathcal{P}_k := \mathcal{C}h_{W_k \cup \widetilde{W}_k}((U_{00} - T_0, U_{01}, \dots, U_{0n}), \dots, (U_{k0} - T_r, U_{k1}, \dots, U_{kn}))$ ;
25.     $u^{(k)} := \text{Random}(n+1, 6(n-1)Nd^{2n+1})$ ;
26.     $H_k := \mathcal{P}_k(u^{(k)}, e_1, \dots, e_k)(u_0^{(k)} + u_1^{(k)} y_1 + \dots + u_n^{(k)} y_n, y_1, \dots, y_k)$ ;
27.  od;
28.  for  $r$  from 0 to  $n-2$  do
29.     $G_r := \prod_{k=r+1}^{n-1} H_k$ ;
30.     $\mathcal{C}h_{W_r} := \text{Sep}(n, r, dD_{r+1}, G_r, d(D_{r+2} + \dots + D_n), \mathcal{C}h_{W_r \cup \widetilde{W}_r}; 6nN)$ ;
31.     $\mathcal{F}_{V_r} := \mathcal{C}h_{W_r}(U_0 B^{-1}, \dots, U_r B^{-1})$ ;
32.  od;
33. fi;
34. return $(\mathcal{F}_{V_0}, \dots, \mathcal{F}_{V_n})$ ;

end.

```

4 Applications

We present some algorithmical applications of our results, concerning the computation of resultants and the resolution of generic over-determined systems.

4.1 Computation of resultants

4.1.1 The classical d -resultant

As a first application of our results, we compute an slp for the classical resultant of $n + 1$ generic homogeneous polynomials of degree d in $n + 1$ variables. The algorithm follows directly from Lemma 2.3 and is therefore deterministic. For the definition and basic properties of the classical resultant we refer for instance to [15, Chapter 3]).

Corollary 4.1 *There is a deterministic algorithm which computes (an slp for) the classical resultant $\text{Res}_{n,d}$ of $n + 1$ generic homogeneous polynomials of degree d in $n + 1$ variables within complexity $(nd^n)^{\mathcal{O}(1)}$.*

Proof.– It is a well-known fact that the resultant $\text{Res}_{n,d}$ is the Chow form of the Veronese variety $V(n, d)$ defined as the image of the morphism

$$\varphi_{n,d} : \mathbb{P}^n \rightarrow \mathbb{P}^N \quad , \quad \xi \mapsto (\xi^\alpha)_{\alpha \in \mathbb{N}_0^{n+1}, |\alpha|=d},$$

where $N := \binom{n+d}{n} - 1$. We recall that $V_{n,d}$ is an irreducible variety of dimension n and degree d^n . We compute here the resultant by defining a system of local equations at an appropriate fiber of $V(n, d)$ in order to apply Lemma 2.3.

Let $\{y_\alpha : \alpha \in \mathbb{N}_0^{n+1}, |\alpha| = d\}$ be a set of homogeneous coordinates of \mathbb{P}^N and consider the projection

$$\pi : V(n, d) \rightarrow \mathbb{P}^n \quad , \quad (y_\alpha)_\alpha \mapsto (y_{de_0} : \cdots : y_{de_n})$$

where e_i is as usual the $(i + 1)$ -vector of the canonical basis of \mathbb{Q}^{n+1} . This projection is finite [51, Chapter 1, Theorem 5.3.7]. Moreover, $Z := \pi^{-1}((1 : 1 : \cdots : 1))$ verifies that $Z = \varphi_{n,d}(Z_0)$ with $Z_0 := \{(1 : \omega_1 : \cdots : \omega_n); \omega_i^d = 1 \text{ for } 1 \leq i \leq n\}$. Thus $\#Z = d^n = \deg V(n, d)$, and the n -dimensional variety $V(n, d)$ satisfies Assumption 1.2 for the fiber Z .

Let us define now a system of local equations of $V_{n,d}$ at Z : For every $\alpha = (\alpha_0, \dots, \alpha_n) \in (\mathbb{N}_0)^{n+1}$ such that $|\alpha| = d$ and $\alpha \neq (d - 1)e_0 + e_i$ ($0 \leq i \leq n$) we consider the polynomial

$$f_\alpha := y_{de_0}^{d-1-\alpha_0} y_\alpha - y_{(d-1)e_0+e_1}^{\alpha_1} \cdots y_{(d-1)e_0+e_n}^{\alpha_n}.$$

These are $N - n$ non-zero homogeneous polynomials of degree $d - \alpha_0$ which vanish at $V_{n,d}$ since

$$f_\alpha((\xi^\beta)_\beta) = \xi_0^{d(d-1-\alpha_0)} \xi^\alpha - \xi_0^{(d-1)(\alpha_1+\cdots+\alpha_n)} \xi_1^{\alpha_1} \cdots \xi_n^{\alpha_n} = 0.$$

From the Jacobian criterion one also checks that, as $\frac{\partial f_\alpha}{\partial y_\alpha} = y_{de_0}^{d-1-\alpha_0}$ and $\frac{\partial f_\alpha}{\partial y_\beta} = 0$ for $\beta \neq \alpha$ and $\beta \neq (d - 1)e_0 + e_i$, the Jacobian matrix of the system has maximal rank $N - n$ at any $\xi \in Z$.

Observe that the equations f_α can be encoded by slp's of length $\mathcal{O}(d)$.

The next step in order to apply Lemma 2.3 is to compute a geometric resolution of the fiber Z . For that purpose we compute its characteristic polynomial (considering it as an affine variety in $\{y_{de_0} \neq 0\}$) and apply Lemma 3.2 for a separating linear form.

Let $L := \sum_{|\alpha|=d} U_\alpha y_\alpha$ be a generic linear form in $N + 1$ variables, and let $P = \sum_{|\alpha|=d} U_\alpha x^\alpha$ be the generic homogeneous polynomial of degree d in $n + 1$ variables associated to L .

The characteristic polynomial of Z is

$$\begin{aligned} \mathcal{P}_Z(U, T) &= \prod_{\xi \in Z} (T - L(U, \xi)) = \prod_{(1:\omega) \in Z_0} (T - L(U, \varphi_{n,d}(1, \omega))) \\ &= \prod_{(1:\omega) \in Z_0} (T - P(U, (1, \omega))) = \prod_{\omega: (1:\omega) \in Z_0} (T - P^a(U, \omega)), \end{aligned}$$

where $P^a(U, \omega) = P(U, (1, \omega))$. Therefore, if we set $A := \mathbb{Q}[x_1, \dots, x_n]/(x_1^d - 1, \dots, x_n^d - 1)$, \mathcal{P}_Z is then computed as the characteristic polynomial of the linear map $A \rightarrow A$ defined by $g \mapsto P^a g$ within complexity $d^{\mathcal{O}(n)}$.

Finally, an easy computation shows that the linear form $\ell = y_{de_0} + d y_{(d-1)e_0+e_1} + \dots + d^n y_{(d-1)e_0+e_n}$ separates the points in Z . Thus ℓ yields a geometric resolution of Z and we apply Lemma 2.3 to compute $\text{Res}_{n,d}$ within the stated complexity. \square

4.1.2 Sparse resultants

Let $\mathcal{A} = \{\alpha_0, \dots, \alpha_N\} \subset \mathbb{Z}^n$ be a finite set of integer vectors. We assume that \mathbb{Z}^n is generated by the differences of elements in \mathcal{A} .

For $0 \leq i \leq n$ let U_i be a group of $N + 1$ variables indexed by the elements of \mathcal{A} , and set

$$F_i := \sum_{\alpha \in \mathcal{A}} U_{i\alpha} x^\alpha \in \mathbb{Q}[U_i][x_1^{\pm 1}, \dots, x_n^{\pm 1}]$$

for the generic Laurent polynomial with support in \mathcal{A} . Let $W_{\mathcal{A}} \subset (\mathbb{P}^N)^{n+1} \times (\mathbb{C}^*)^n$ be the incidence variety of F_0, \dots, F_n in $(\mathbb{C}^*)^n$, that is

$$W_{\mathcal{A}} = \{(\nu_0, \dots, \nu_n; \xi) \in (\mathbb{P}^N)^{n+1} \times (\mathbb{C}^*)^n : F_i(\nu_i, \xi) = 0, 0 \leq i \leq n\},$$

and let $\pi : (\mathbb{P}^N)^{n+1} \times (\mathbb{C}^*)^n \rightarrow (\mathbb{P}^N)^{n+1}$ be the canonical projection. Then $\overline{\pi(W_{\mathcal{A}})}$ is an irreducible variety of codimension 1. The \mathcal{A} -resultant $\text{Res}_{\mathcal{A}}$ is defined as the unique —up to a sign— irreducible polynomial in $\mathbb{Z}[U_0, \dots, U_n]$ which defines this hypersurface (see [21, Chapter 8, Proposition-Definition 1.1]).

This is a multihomogeneous polynomial of degree $\text{Vol}(\mathcal{A})$ in each group of variables U_i , where $\text{Vol}(\mathcal{A})$ denotes the (normalized) volume of the convex hull $\text{Conv}(\mathcal{A})$, which is defined as $n!$ times its volume with respect to the Euclidean volume form of \mathbb{R}^n .

Consider the map

$$(\mathbb{C}^*)^n \rightarrow \mathbb{P}^N, \quad \xi \mapsto (\xi^{\alpha_0} : \dots : \xi^{\alpha_N}).$$

The Zariski closure of the image of this map is called the *affine toric variety* $X_{\mathcal{A}} \subset \mathbb{P}^N$ associated to \mathcal{A} . This is an irreducible variety of dimension n and degree $\text{Vol}(\mathcal{A})$. Its Chow form coincides —up to a scalar factor— with the sparse resultant $\text{Res}_{\mathcal{A}} \in \mathbb{Z}[U_0, \dots, U_n]$ (see [21, Chapter 8, Proposition 2.1] and [15, Chapter 7, Theorem 3.4]).

For a broader background on toric varieties and sparse resultants we refer to [21] and [15].

We apply the algorithm underlying Theorem 1 to compute the sparse resultant $\text{Res}_{\mathcal{A}}$ for the particular case that $\mathcal{A} \subset (\mathbb{N}_0)^n$ and the elements $0, e_1, \dots, e_n$ —that is the vertices of the standard simplex of \mathbb{R}^n — lie in \mathcal{A} . To do so, we construct a set of equations which define $X_{\mathcal{A}}$ in the open chart $(\mathbb{P}^N)_{y_0}$, where $(y_0 : \dots : y_N)$ is a system of homogeneous coordinates of \mathbb{P}^N , and compute a Chow form of this variety.

Corollary 4.2 *Let $\mathcal{A} \subset \mathbb{N}_0^n$ be a finite set which contains $\{0, e_1, \dots, e_n\}$. Then there is a bounded probability algorithm which computes (an slp for) a scalar multiple of the \mathcal{A} -resultant $\text{Res}_{\mathcal{A}}$ within (expected) complexity $(n + \text{Vol}(\mathcal{A}))^{\mathcal{O}(1)}$.*

Proof.– W.l.o.g. we assume that in \mathcal{A} , $\alpha_0 = 0$ and $\alpha_i = e_i$ for $i = 1, \dots, n$. Set $d := \max_{\alpha \in \mathcal{A}} |\alpha|$. For $n + 1 \leq j \leq N$ we set

$$f_j := y_0^{d-1} y_j - y_0^{d-|\alpha_j|} y_1^{\alpha_{j1}} \dots y_n^{\alpha_{jn}} \in \mathbb{Q}[y_0, \dots, y_N].$$

Then, $X_{\mathcal{A}} \setminus \{y_0 = 0\} = V := V(f_{n+1}, \dots, f_N)_{y_0} \subset (\mathbb{P}^N)_{y_0}$. Therefore the Chow form of $X_{\mathcal{A}}$ coincides with the one of V and can be computed by application of Procedure Equidim (Subroutine 11) to the polynomial system $f_{n+1}, \dots, f_N; y_0$.

Each polynomial f_j , $n + 1 \leq j \leq N$, can be encoded by an slp of length $\mathcal{O}(d)$. Moreover, as for each $\alpha \in \mathcal{A}$, $|\alpha| = \text{Vol}(\{0, e_1, \dots, e_n, \alpha\}) \leq \text{Vol}(\mathcal{A})$ since $\{0, e_1, \dots, e_n, \alpha\} \subset \mathcal{A}$, then $d \leq \text{Vol}(\mathcal{A})$. Therefore $L(f_j) \leq \mathcal{O}(\text{Vol}(\mathcal{A}))$ for $n + 1 \leq j \leq N$.

Now, as the toric variety $X_{\mathcal{A}}$ is non-degenerated (that is, it is not contained in any hyperplane in \mathbb{P}^N), [30, Corollary 18.12] implies that

$$N + 1 \leq \dim X_{\mathcal{A}} + \deg X_{\mathcal{A}} = n + \text{Vol}(\mathcal{A}).$$

This gives an estimation for the parameter N .

Finally, we have to estimate the geometric degree $\delta(f_{n+1}, \dots, f_N; y_0)$. As we want to compute this degree outside $\{y_0 = 0\}$ it is enough to deal with linear combinations of the dehomogenized polynomials \hat{f}_j obtained by specializing $y_0 = 1$ in the original f_j for $n + 1 \leq j \leq N$.

For $1 \leq i \leq N$, $n + 1 \leq j \leq N$ and $a_{ij} \in \mathbb{Q}$ we set

$$q_i := a_{in+1} \hat{f}_{n+1} + \dots + a_{iN} \hat{f}_N.$$

For every i , the support $\text{Supp}(q_i)$ —that is the set of exponents of its non-zero monomials—is contained in $(\mathcal{A} \times \{0\}) \cup \mathcal{S} \subset \mathbb{Z}^N$, where $\mathcal{S} := \{e_{n+1}, \dots, e_N\} \subset \mathbb{Z}^N$ and then, by [41, Proposition 2.12],

$$\deg V(q_1, \dots, q_i) \leq \text{Vol}((\mathcal{A} \times \{0\}) \cup \mathcal{S}).$$

As we have that

$$\text{Vol}((\mathcal{A} \times \{0\}) \cup \mathcal{S}) = N! \text{vol}_{\mathbb{R}^N} \text{Conv}((\mathcal{A} \times \{0\}) \cup \mathcal{S}) = n! \text{vol}_{\mathbb{R}^n} \text{Conv}(\mathcal{A}) = \text{Vol}(\mathcal{A})$$

(where $\text{vol}_{\mathbb{R}^N}$ and $\text{vol}_{\mathbb{R}^n}$ denote the standard Euclidean volume forms) we infer that

$$\delta := \delta(f_{n+1}, \dots, f_N; y_0) \leq \text{Vol}(\mathcal{A}).$$

We conclude that $\text{Res}_{\mathcal{A}}$ can be probabilistically computed by means of subroutine Equidim within complexity $(N - n)(N d \delta)^{\mathcal{O}(1)} L(f_{n+1}, \dots, f_N) \leq (n + \text{Vol}(\mathcal{A}))^{\mathcal{O}(1)}$. \square

Remark 4.3 *It would be interesting to improve this algorithm in order to compute $\text{Res}_{\mathcal{A}}$ without any extraneous scalar factor. It would suffice to compute this factor as the coefficient of any extremal monomial of $\mathcal{F}_{X_{\mathcal{A}}}$, as we know a priori that the corresponding coefficient in $\text{Res}_{\mathcal{A}}$ equals ± 1 ([21, Chapter 8, Theorem 3.3], see also [55, Corollary 3.1]).*

Example 4.4 *We take the following example from [41, Example 4.13]: Set*

$$\mathcal{A}(n, d) := \{0, e_1, \dots, e_n, e_1 + \dots + e_n, 2(e_1 + \dots + e_n), \dots, d(e_1 + \dots + e_n)\} \subset \mathbb{Z}^n.$$

It is easy to check that $\text{Vol}(\mathcal{A}(n, d)) = n d$, and so the previous algorithm computes an slp for (a scalar multiple of) $\text{Res}_{\mathcal{A}(n, d)}$ within $(n d)^{\mathcal{O}(1)}$ arithmetic operations.

4.2 Generic over-determined systems

Our last application concerns the computation of the unique solution of a generic over-determined system.

Let $f_0, \dots, f_n \in \mathbb{Q}[x_0, \dots, x_n]$ be homogeneous polynomials of degree d . The associated equation system is *generically* inconsistent, where generically means if and only if the vector of the coefficients of the polynomials does not lie in the hypersurface $V(\text{Res}_{n,d}) \subset (\mathbb{P}^N)^{n+1}$ defined by the classical resultant $\text{Res}_{n,d}$ of $n+1$ homogeneous $(n+1)$ -variate polynomials of degree d , and $N := \binom{d+n}{n} - 1$.

Now assume that the system *is* consistent. In this case the system is said to be *over-determined*, in the sense that its solution set can be defined—at least locally—with fewer equations.

Under this condition the system has generically exactly one solution, which is a rational map of the coefficients of the polynomials f_0, \dots, f_n (see Corollary 4.7 below). A natural problem is to compute this rational parametrization. In what follows we show that this parametrization can be easily derived from the resultant, and therefore can be computed with our algorithm.

In fact we treat the more general case of an over-determined linear system on a variety. The following result seems to be classical although we could not find it in the existing literature. The closest statement we found is given in [21, Chapter 3, Corollary 3.7] and has a mistake as we discuss below.

Lemma 4.5 *Let $V \subset \mathbb{P}^n$ be an equidimensional variety of dimension r definable over \mathbb{Q} . Let $\mathcal{F}_V(U_0, \dots, U_r)$ be a Chow form of V , and let $u := (u_0, \dots, u_r) \in V(\mathcal{F}_V) \subset (\mathbb{P}^n)^{r+1}$ be such that $\partial \mathcal{F}_V / \partial U_{i_0 j_0}(u) \neq 0$ for some $0 \leq i_0 \leq r$, $0 \leq j_0 \leq n$. For $0 \leq i \leq r$, let $L_i(U_i, x) := U_{i_0} x_0 + \dots + U_{i_n} x_n$ denote the generic linear form associated to U_i . Then $V \cap V(L_0(u_0, x), \dots, L_r(u_r, x))$ consists of exactly one element $\xi(u)$, and*

$$\xi(u) = \left(\frac{\partial \mathcal{F}_V}{\partial U_{i_0 0}}(u) : \dots : \frac{\partial \mathcal{F}_V}{\partial U_{i_0 n}}(u) \right).$$

Proof.—As the formula stated by the Lemma is invariant under linear changes of variables, we can assume w.l.o.g. that no irreducible component of V is contained in any hyperplane $\{x_j = 0\}$, $0 \leq j \leq n$.

For $0 \leq i \leq r$ we set $\ell_i(x) := L_i(u_i, x) = u_{i_0} x_0 + \dots + u_{i_n} x_n \in \mathbb{C}[x_0, \dots, x_n]$ for the linear form associated to $u_i \in \mathbb{C}^{n+1}$. Then $V \cap V(\ell_0, \dots, \ell_r) \neq \emptyset$ because of the assumption $\mathcal{F}_V(u) = 0$. Let ξ be a point in this variety. Suppose $\xi_0 \neq 0$.

Set $V^{\text{aff}} \subset \mathbb{A}^n$ for the image of V under the rational map $\psi : \mathbb{P}^n \dashrightarrow \mathbb{A}^n$ defined by $(x_0 : \dots : x_n) \mapsto (x_1/x_0, \dots, x_n/x_0)$. Let $T := \{T_0, \dots, T_r\}$ be a group of $r+1$ additional variables, and let $P := P_{V^{\text{aff}}} \in \mathbb{Q}[U][T]$ be the characteristic polynomial of V^{aff} , as defined in Subsection 3.1. Then, for $0 \leq j \leq n$,

$$0 = \frac{\partial P(U, L)}{\partial U_{i_0 j}}(u, \xi) + \frac{\xi_j}{\xi_0} \frac{\partial P(U, L)}{\partial T_{i_0}}(u, \xi) = \frac{\partial \mathcal{F}_V}{\partial U_{i_0 j}}(u) - \frac{\xi_j}{\xi_0} \frac{\partial \mathcal{F}_V}{\partial U_{i_0 0}}(u). \quad (13)$$

The first equality was shown in Lemma 3.1, while the second follows directly from formula (6) in Subsection 3.1, and the fact that $L_i(u_i, \xi) = 0$ for $0 \leq i \leq r$.

From Identity (13) and the assumption $\partial \mathcal{F}_V / \partial U_{i_0 j_0}(u) \neq 0$, we infer that $\partial \mathcal{F}_V / \partial U_{i_0 0}(u) \neq 0$ and $\frac{\xi_j}{\xi_0} = \frac{\partial \mathcal{F}_V / \partial U_{i_0 j}}{\partial \mathcal{F}_V / \partial U_{i_0 0}}(u)$. Therefore

$$\xi = \left(\frac{\partial \mathcal{F}_V}{\partial U_{i_0 0}}(u) : \dots : \frac{\partial \mathcal{F}_V}{\partial U_{i_0 n}}(u) \right).$$

This shows in particular that ℓ_0, \dots, ℓ_r have exactly one common root in $V \setminus \{x_0 = 0\}$. Moreover, as the formula for the coordinates of ξ does not depend on the chosen affine chart, we conclude that ξ is the only common root of ℓ_0, \dots, ℓ_r in V . \square

Corollary 3 of [21, Chapter 3] would imply that the formula of Lemma 4.5 holds in case $\xi(u)$ is a simple common root of ℓ_0, \dots, ℓ_r in V . Denoting by $\mathcal{O}_{V,\xi}$ the local ring of V at ξ , this is equivalent to the fact that $\mathcal{O}_{V,\xi}/(\ell_0, \dots, \ell_r) \cong \mathbb{C}$.

The following counterexample shows that this is not true: let

$$F(t, x, y) := x^2(x+t) - ty^2 \in \mathbb{Q}[t, x, y] \quad , \quad C := V(F) \subset \mathbb{P}^2.$$

C is an elliptic curve with a node at $(1:0:0)$. The linear forms $\ell_0 := L_0((0:1:0), (t:x:y)) = x$ and $\ell_1 := L_1((0:0:1), (t:x:y)) = y$ have a single common root $(1:0:0)$ in C , which is a simple root of ℓ_0, ℓ_1 in C .

On the other hand, as C is a hypersurface, $\mathcal{F}_C = F(M_0, -M_1, M_2)$, where M_j denotes the maximal minor obtained by deleting the $(j+1)$ column of the matrix

$$\begin{pmatrix} U_{00} & U_{01} & U_{02} \\ U_{10} & U_{11} & U_{12} \end{pmatrix}.$$

A straightforward computation shows that $\partial\mathcal{F}_C/\partial U_{ij}((0:1:0), (0:0:1)) = 0$ for every i, j .

The proof given in [21] is based on the biduality theorem and on Cayley's trick, and it holds in case V is *smooth*, and in case $u = (u_0, \dots, u_r)$ does not lie in the singular locus of the hypersurface $V(\mathcal{F}_V)$. This last condition is equivalent to ours.

Let $V \subset \mathbb{P}^n$ be an equidimensional variety of dimension r , and set $\Omega_V := V(\mathcal{F}_V) \subset (\mathbb{P}^n)^{r+1}$ for the set of (coefficients of) over-determined linear systems over V . As \mathcal{F}_V is squarefree and each of its irreducible factors depends on every group of variables,

$$\gcd(\mathcal{F}_V, \partial\mathcal{F}_V/\partial U_{i_0}, \dots, \partial\mathcal{F}_V/\partial U_{i_n}) = 1$$

for $0 \leq i \leq r$. Then $\Theta_i := \Omega_V \setminus V(\partial\mathcal{F}_V/\partial U_{i_0}, \dots, \partial\mathcal{F}_V/\partial U_{i_n})$ is a dense open set of Ω_V and so

$$\Psi_V : \Omega_V \dashrightarrow \mathbb{P}^n \quad , \quad u \mapsto \xi(u) := \left(\frac{\partial\mathcal{F}_V}{\partial U_{i_0}}(u) : \dots : \frac{\partial\mathcal{F}_V}{\partial U_{i_n}}(u) \right) \quad \text{if } u \in \Theta_i$$

is a rational map well-defined on $\Theta_0 \cup \dots \cup \Theta_r$.

Now let $V \subset (\mathbb{P}^n)_g$ be an arbitrary variety of dimension r , and let $V = V_r \cup \dots \cup V_0$ be its equidimensional decomposition. In what follows, for sake of clarity, we keep the same notations as previously for different objects sharing analogous properties.

Set (again) $\Omega_V \subset (\mathbb{P}^n)^{r+1}$ for the set

$$\Omega_V = \{(u_0, \dots, u_r) \in (\mathbb{P}^n)^{r+1} : \exists \xi \in V / L_0(u_0, \xi) = 0, \dots, L_r(u_r, \xi) = 0\}$$

of generic over-determined linear systems over V , which is a quasiprojective variety of codimension 1 in $(\mathbb{P}^n)^{r+1}$.

For every $0 \leq k \leq r$, let Ω_{V_k} be the set of the coefficients of $r+1$ linear forms which have a common root in V_k . If \mathcal{F}_{V_k} is a Chow form of V_k , we have that

$$\Omega_{V_k} \subset \bigcap_{0 \leq i_0 < \dots < i_k \leq r} V(\mathcal{F}_{V_k}(U_{i_0}, \dots, U_{i_k}))$$

and, therefore, Ω_{V_k} has codimension at least 2 for $0 \leq k \leq r-1$.

Let $\Theta_i := \Omega_{V_r} \setminus V(\partial\mathcal{F}_{V_r}/\partial U_{i_0}, \dots, \partial\mathcal{F}_{V_r}/\partial U_{i_n})$ for $i = 0, \dots, r$.

Then every over-determined linear system over V with coefficients in the open set $(\Theta_0 \cup \dots \cup \Theta_r) \setminus (\Omega_{V_0} \cup \dots \cup \Omega_{V_{r-1}})$ of Ω_V has a unique solution in V which, in fact, lies in V_r . As before, this solution can be given by the rational map

$$\Psi_V := \Psi_{V_r} : \Omega_{V_r} \dashrightarrow (\mathbb{P}^n)_g \quad , \quad u \mapsto \xi(u) := \left(\frac{\partial\mathcal{F}_{V_r}}{\partial U_{i_0}}(u) : \dots : \frac{\partial\mathcal{F}_{V_r}}{\partial U_{i_n}}(u) \right) \quad \text{if } u \in \Theta_i.$$

As an immediate consequence of Theorem 1 and Lemma 4.5 we obtain:

Corollary 4.6 Let $f_1, \dots, f_s, g \in \mathbb{Q}[x_0, \dots, x_n]$ be homogeneous polynomials of degree bounded by d encoded by straight-line programs of length bounded by L .

Set $V := V(f_1, \dots, f_s) \setminus V(g) \subset \mathbb{P}^n$ for the quasiprojective variety $\{f_1 = 0, \dots, f_s = 0, g \neq 0\}$ and let $V = V_0 \cup \dots \cup V_n$ be its minimal equidimensional decomposition. Let $\delta := \delta(f_1, \dots, f_s; g)$ be the geometric degree of the input polynomial system.

Then there is a bounded probability algorithm which computes (slp's for) the coordinates of the rational map Ψ_V defined above within (expected) complexity $s(nd\delta)^{O(1)}L$.

The previous result can be applied directly to compute the solution of a generic over-determined system of $n + 1$ homogeneous polynomials in $n + 1$ variables of degree d by means of $\text{Res}_{n,d}$:

Corollary 4.7 Let $u = (u_0, \dots, u_n) \in (\mathbb{P}^N)^{n+1}$ where $N := \binom{d+n}{n} - 1$, and for $0 \leq i \leq n$, set

$$f_i := \sum_{|\alpha|=d} u_{i\alpha} x^\alpha.$$

Assume that $\text{Res}_{n,d}(u) = 0$ and that $\partial \text{Res}_{n,d} / \partial U_{i_0 \beta}(u) \neq 0$ for some $0 \leq i_0 \leq n, \beta = (\beta_0, \dots, \beta_n) \in (\mathbb{N}_0)^{n+1}$ with $|\beta| = d$.

Then $V(f_0, \dots, f_n)$ consists of exactly one element $\xi(u) \in \mathbb{P}^n$, and

$$\xi(u) = \left(\frac{\partial \text{Res}_{n,d}}{\partial U_{i_0, (d-1) e_j + e_0}}(u) : \dots : \frac{\partial \text{Res}_{n,d}}{\partial U_{i_0, (d-1) e_j + e_n}}(u) \right)$$

for any $0 \leq j \leq n$ such that $\beta_j \neq 0$.

Proof.- From Lemma 4.5 applied to the Veronese variety $V(n, d) \subset \mathbb{P}^N$ (see Section 4.1.1) we have that $V(f_0, \dots, f_n)$ has only one point $\xi(u)$ and that

$$(\xi(u)^\alpha)_{|\alpha|=d} = \left(\frac{\partial \text{Res}_{n,d}}{\partial U_{i_0 \alpha}}(u) \right)_{|\alpha|=d}.$$

Let $\beta = (\beta_0, \dots, \beta_n)$ be such that $|\beta| = d$ and $\partial \text{Res}_{n,d} / \partial U_{i_0 \beta}(u) \neq 0$, and let $0 \leq j \leq n$ be such that $\beta_j \neq 0$. The previous identity implies that $\xi \in \{x_j \neq 0\}$. Then

$$\xi(u) = (\xi_j^{d-1} \xi_0 : \dots : \xi_j^{d-1} \xi_n) = \left(\frac{\partial \text{Res}_{n,d}}{\partial U_{i_0, (d-1) e_j + e_0}}(u) : \dots : \frac{\partial \text{Res}_{n,d}}{\partial U_{i_0, (d-1) e_j + e_n}}(u) \right).$$

□

As an immediate consequence of this result and Proposition 4.1 we obtain:

Corollary 4.8 Let notation be as in Corollary 4.7. Then the rational map $(\mathbb{P}^N)^{n+1} \dashrightarrow \mathbb{P}^n, u \mapsto \xi(u)$ can be (deterministically) computed within complexity $(nd^n)^{O(1)}$.

References

- [1] J. ABDELJAOUED, *Algorithmes rapides pour le calcul du polynôme caractéristique*. Ph.D. dissertation, Univ. Franche-Comté, Besançon, France, 1997.
- [2] S.J. BERKOWITZ, *On computing the determinant in small parallel time using a small number of processors*, Inform. Process. Lett. **18** (1984), 147-150.
- [3] W. BAUR, W. STRASSEN, *The complexity of partial derivatives*, Theoret. Comput. Sci. **22** (1983), 317-330.
- [4] C. BLANCO, G. JERONIMO, P. SOLERNÓ *Computing generators of the ideal of a smooth affine algebraic variety*, Preprint Universidad de Buenos Aires, 2003.

- [5] L. BLUM, F. CUCKER, M. SHUB, S. SMALE, *Complexity and real computation*, Springer, 1998.
- [6] N. BOURBAKI, *Éléments de mathématique. Algèbre commutative*, Hermann, 1961.
- [7] W.S. BROWN, J.F. TRAUB, *On Euclid's algorithm and the theory of subresultants*, J. ACM **18** (1971), 505-514.
- [8] P. BÜRGISSER, M. CLAUSEN, M.A. SHOKROLLAHI, *Algebraic complexity theory*, Springer, 1997.
- [9] L. CANIGLIA, *How to compute the Chow form of an unmixed ideal in single exponential time*, Appl. Algebra Engrg. Comm. Comput. **1** (1990), 25-41.
- [10] J.F. CANNY, I.Z. EMIRIS, *An efficient algorithm for the sparse mixed resultant*, Lecture Notes in Comput. Sci. **263** (1993), 89-104.
- [11] J.F. CANNY, I.Z. EMIRIS, *A subdivision-based algorithm for the sparse resultant*, J. ACM **47** (2000), 417-451.
- [12] D. CASTRO, M. GIUSTI, J. HEINTZ, G. MATERA, L.M. PARDO, *The hardness of polynomial equation solving*, To appear in FoCM Journal (2003).
- [13] A.L. CHISTOV, D.Y. GRIGORIEV, *Subexponential time solving systems of algebraic equations*. LOMI preprint E-9-83, E-10-83, Steklov Institute, Leningrad, 1983.
- [14] G.E. COLLINS, *Subresultants and reduced polynomial remainder sequences*, J. ACM **14** (1967), 128-142. Univ. Nice-Sophia Antipolis, 1996.
- [15] D. COX, J. LITTLE, D. O'SHEA, *Using algebraic geometry*, Grad. Texts in Math. **185**, Springer-Verlag, 1998.
- [16] J. DALBEC, B. STURMFELS, *Introduction to Chow forms*, Invariant methods in discrete and computational geometry (Curaçao, 1994), Kluwer, 1995, 37-58.
- [17] C. D'ANDREA, *Macaulay's style formulas for sparse resultants*, Trans. Amer. Math. Soc. **354**, No. 7, 2002, 2595-2629 (electronic).
- [18] D. EISENBUD, *Commutative algebra with a view toward algebraic geometry*, Grad. Texts in Math. **150**, Springer-Verlag, 1995.
- [19] M. ELKADI, B. MOURRAIN, *A new algorithm for the geometric decomposition of a variety*, Proc. ISSAC'1999 (ACM) (1999), 9-16.
- [20] I. EMIRIS, B. MOURRAIN, *Matrices in elimination theory*, J. Symb. Comput. **28** (1999), 3-44.
- [21] I.M. GELFAND, M.M. KAPRANOV, A.V. ZELEVINSKY, *Discriminants, resultants, and multidimensional determinants*, Birkhäuser, 1994.
- [22] M. GIUSTI, K. HÄGELE, J. HEINTZ, J.L. MONTAÑA, L.M. PARDO, J.E. MORAIS, *Lower bounds for Diophantine approximation*, J. Pure Appl. Algebra **117 & 118** (1997), 277-317.
- [23] M. GIUSTI, J. HEINTZ, *La détermination des points isolés et de la dimension d'une variété algébrique peut se faire en temps polynomial*, Computational Algebraic Geometry and Commutative Algebra (D. Eisenbud, L. Robbiano, eds.) Sympos. Math. Vol. **XXXIV** (1993), 216-256.
- [24] M. GIUSTI, J. HEINTZ, *Algorithmes — disons rapides — pour la décomposition d'une variété algébrique en composantes irréductibles et équidimensionnelles*, Progress in Math. **94**, Birkhäuser, 1991, 164-194.
- [25] M. GIUSTI, J. HEINTZ, *Kronecker's smart, little black-boxes*, Proceedings of Foundations of Computational Mathematics, Oxford 1999 (FoCM'99), A. Iserles and R. DeVore, eds., Cambridge University Press **284**, 2001, 69-104.
- [26] M. GIUSTI, J. HEINTZ, J.E. MORAIS, J. MORGENSTERN, L.M. PARDO, *Straight-line programs in geometric elimination theory*, J. Pure Appl. Algebra **124** (1998), 101-146.

- [27] M. GIUSTI, J. HEINTZ, J.E. MORAIS, L.M. PARDO, *When polynomial equation systems can be “solved” fast?*, Proc. 11th. International Symposium Applied Algebra, Algebraic Algorithms and Error-Correcting Codes, AAEC-11, Paris 1995, G. Cohen, M. Giusti and T. Mora, eds., Lecture Notes in Comput. Sci. **948** (1995), 205-231.
- [28] M. GIUSTI, J. HEINTZ, J. SABIA, *On the efficiency of effective Nullstellensätze*, Comput. Complexity **3** (1993), 56-95.
- [29] M. GIUSTI, G. LECERF, B. SALVY, *A Gröbner free alternative for polynomial system solving*, J. Complexity **17** (2001), 154-211.
- [30] J. HARRIS, *Algebraic Geometry*, Grad. Texts in Math. **133**, Springer, 1992.
- [31] J. HEINTZ, *Definability and fast quantifier elimination in algebraically closed fields*, Theoret. Comput. Sci. **24** (1983), 239-277.
- [32] J. HEINTZ, *On the computational complexity of polynomials and bilinear maps. A survey*, Lecture Notes in Comput. Sci. **356** (1989), 269-300.
- [33] J. HEINTZ, T. KRICK, S. PUDDU, J. SABIA, A. WAISSBEIN, *Deformation techniques for efficient polynomial equation solving*, J. Complexity **16** (2000), 70-109.
- [34] J. HEINTZ, G. MATERA, L.M. PARDO, R. WACHENCHAUSER, *The intrinsic complexity of parametric elimination methods*, Electronic J. SADIO **1** (1998), 37-51.
- [35] J. HEINTZ, G. MATERA, A. WAISSBEIN, *On the time-space complexity of geometric elimination procedures*, Appl. Algebra Engrg. Comm. Comput. **11**(4) (2001), 239-296.
- [36] G. JERONIMO, S. PUDDU, J. SABIA, *Computing Chow forms and some applications*, J. Algorithms **41** (2001), 52-68.
- [37] G. JERONIMO, J. SABIA, *Effective equidimensional decomposition of affine varieties*, J. Pure Appl. Algebra **169** (2002), 229-248.
- [38] E. KALTOFEN, *Greatest common divisors of polynomials given by straight-line programs*, J. ACM **35**, N° 1, (1988) 234-264.
- [39] T. KRICK, *Complejidad para problemas de geometría elemental*, Ph.D. dissertation, Univ. Buenos Aires, Buenos Aires, Argentina, 1990.
- [40] T. KRICK, L.M. PARDO, *A computational method for Diophantine approximation*, Progress in Math. **143**, Birkhäuser, 1996, 193-253.
- [41] T. KRICK, L.M. PARDO, M. SOMBRA, *Sharp estimates for the arithmetic Nullstellensatz*, Duke Math. J. **109**, No. 3 (2001), 521-598.
- [42] T. KRICK, J. SABIA, P. SOLERNÓ, *On intrinsic bounds in the Nullstellensatz*, Appl. Algebra Engrg. Comm. Comput. **8** (1997), 125-134.
- [43] G. LECERF, *Kronecker, a Magma package for polynomial system solving*, available at <http://kronecker.medicis.polytechnique.fr/>.
- [44] G. LECERF, *Computing an equidimensional decomposition of an algebraic variety by means of geometric resolutions*, Proc. ISSAC'2000, ACM (2000), 209-216.
- [45] G. LECERF, *Quadratic Newton iteration for systems with multiplicity*, FoCM Journal, **2**(3) (2002) 247-293.
- [46] G. LECERF, *Computing the equidimensional decomposition of an algebraic closed set by means of lifting fibers*, To appear in J. Complexity (2003).
- [47] P. PHILIPPON, *Critères pour l'indépendance algébrique*, Inst. Hautes Études Sci. Publ. Math. **64** (1986), 5-52.

- [48] S. PUDDU, J. SABIA, *An effective algorithm for quantifier elimination over algebraically closed fields using straight line programs*, J. Pure Appl. Algebra **129** (1998), 173-200.
- [49] J.M. ROJAS, *Toric laminations, sparse generalized characteristic polynomials, and a refinement of Hilbert's tenth problem*, Proc. FoCM'97, Springer-Verlag, 1997, 369-381.
- [50] J.T. SCHWARTZ, *Fast probabilistic algorithms for verification of polynomial identities*, J. ACM, **27** (1980), 701-717.
- [51] I. SHAFAREVICH, *Basic algebraic geometry*, Springer-Verlag, 1972.
- [52] V. STRASSEN, *Berechnung und Programm I, II*, Acta Inform. **1** (1972), 320-355; *ibid.* **2** (1973), 64-79.
- [53] V. STRASSEN, *Vermeidung von Divisionen*, J. Reine Angew. Math. **264** (1973) 182-202.
- [54] B. STURMFELS, *Sparse elimination theory*, in D. Eisenbud and L. Robbiano, eds., Computational algebraic geometry and commutative algebra, Cambridge Univ. Press, 1993, pp. 377-396.
- [55] B. STURMFELS, *On the Newton polytope of the resultant*, J. Algebraic Combinatorics **3** (1994), 207-236.
- [56] W. VOGEL, *Lectures on results on Bezout's theorem*, Tata Institute of Fundamental Research Lectures on Mathematics and Physics, 74, Springer-Verlag, Berlin, 1984.
- [57] R. ZIPPEL, *Probabilistic algorithms for sparse polynomials*, Proceedings EUROSAM'79, Lecture Notes in Comput. Sci. **72**, Springer (1979), 216-226.

GABRIELA JERONIMO, TERESA KRICK, AND JUAN SABIA: Departamento de Matemática, Universidad de Buenos Aires, Ciudad Universitaria, 1428 Buenos Aires, Argentina.
 E-mail: jeronimo@dm.uba.ar (G.J.), krick@dm.uba.ar (T.K.), jsabia@dm.uba.ar (J.S.)

MARTÍN SOMBRA: Université de Paris 7, UFR de Mathématiques, Équipe de Géométrie et Dynamique, 2 place Jussieu, 75251 Paris Cedex 05, France; and Departamento de Matemática, Universidad Nacional de La Plata, Calle 50 y 115, 1900 La Plata, Argentina.
 E-mail: sombra@math.jussieu.fr